

北京科海培训中心

---

SCO UNIX  
系统管理员实用指南

鹤 立 编

中央民族大学出版社

## 内 容 简 介

本书全面介绍了SCO UNIX System V / 386 (Release 3.2.2) 系统管理员的任务, 目的在于指导 SCO UNIX System V / 386 的系统管理人员对系统进行安装、管理与维护, 进而为系统用户提供一个高效、安全的工作环境。全书根据系统的几个比较独立的功能块分成五大部分: 第一部分系统安装, 主要介绍系统安装环境及安装过程; 第二部分日常管理, 对用户帐号、文件系统及数据安全等方面的管理分别作了详细的阐述; 第三部分外设管理, 介绍了系统支持的外设(打印机、硬盘、鼠标器等)安装和操作方法; 第四部分网络通信, 讲述 Micnet、UUCP、调制解调器、电子邮件系统几部分, 讲述系统的通信环境; 最后一部分归纳了系统管理员在工作中遇到的各种问题, 并提供了有效的解决办法。

本书注重实用性, 覆盖面广, 组织合理, 是SCO UNIX系统管理员必备的工具书, 也可作为 SCO UNIX 系统普通用户的参考书, 同时, 对有关大专院校师生学习了解 UNIX 系统也有相当的参考价值。

## 概 述

UNIX 系统是由许多程序的集合，利用这些程序可以做高级开发和汇编语言程序、生成文件、编辑和排版等一系列工作。为了保证系统的正常运行，必须对其运行进行认真的控制和定期维护。本书介绍如何在计算机上运行和维护这个操作系统，以实现最佳性能，并尽量减少问题的发生。

书中提到的既有不需要对 UNIX 系统有太多了解就能完成的工作，也有需要对本操作系统和你的计算机有深入了解才能完成的工作。各章分别介绍了为完成在本章所讲述的任务所需要的相应工具知识。

本书包括一些有关计算机硬件的章节，这些章节全面介绍了各种设备的使用及其与操作系统的相互作用。

本书还介绍本系统如何在局部网和远程网中工作。靠串行线连接的局部网 micnet 可以把 UNIX 系统连接到工作现场，而靠电话线连接的广域网 uucp 可以使你的 UNIX 工作现场与全世界进行通信。

系统运行中的一项重要任务是保护系统中的数据。本系统为数据保护设计了灵活的机制。

最后请重视“解决系统问题”这一章，它为保证系统正常运行提供了非常有力的手段。

### 系统管理员及其任务

每个 UNIX 系统都需要至少一个人来负责系统的维护和运行，这样的人就叫做系统管理员。其责任是保证系统正常运行，并从事特权工作，这就要求系统管理员熟练掌握系统的许多不同功能。

系统管理员可以有一个或多个。如果有多个，可分别负责不同方面的工作，并且每个人都严格地限制在本领域中。系统管理员必须忠实执行各项所需要的维护任务，粗心的维护肯定会影响系统的工作。

系统管理员必须准备一个日志来记载对系统所做的修改和系统中发生的重大事件。对于每一个事件、信息、备份或修改，都必须注明当时的日期、时间、注册用户姓名以及当时的环境等。例如，当向系统中加入一个新的应用软件时，就应该在日志中增加一个记录，这个记录必须包括时间、日期、软件安装人的姓名及一些关于此软件使用和安装的有关说明。准确的日志对于诊断系统中出现的问题和制定系统发展和使用计划是非常有帮助的。

本书中提及的任务都是从系统管理员的角度介绍的，但其中许多工作也可以由普通用户完成。由于其中有些工作会改变系统的运行，因此这些工作一定要由系统管理员来做。然而，无论谁做了何种操作，都应该被记入系统日志中，这样才能防止对系统不良的或不必要的修改。

系统管理员的日常工作如下：

- 做适当的备份（系统上文件的正常拷贝），供以后使用。

- 处理由于计算机有限资源的使用（如盘空间、进程数目等）而遇到的问题。
- 排除由于连接问题而造成的系统通信（网络）阻塞。
- 进行操作系统的升级和维修。
- 为用户提供支持和帮助。

系统管理员的工作还可以根据其频率被分为几组。下表中列出了管理员从每天要做一次的工作到一个月可能也不必做的各种工作，当然，根据系统的大小和复杂度的不同，其中的一些工作可能要做得或多或少一些。

表 1.1 系统管理员的任务

根据需要所做的工作	详细介绍的章节
在日志中记录系统的修改和重大事件	本章前面的段落
随时解决由于外界原因产生的问题(如电压不稳等)以及用户提出的问题	参阅“解决系统问题”一章
硬件、软件、数据文件访问的安全性维护	参阅“系统安全性维护”一章

每天做一次的工作	详细介绍的章节
进行文件备份	参阅“文件系统的备份”一章
系统运行效率检查	参阅“系统性能的调整”一章
失控进程的检查	参阅“解决系统问题”一章的“失控的进程”
检查磁盘空间	参阅“文件系统的维护”一章中的“显示空闲空间”
检查通信状况和连接状况	参阅“建立电子邮件”一章中的“MMDF 系统的维护”
检查打印机状态	参阅“打印机”一章中“检查打印机的配置”
检查审计报告	参阅“审计系统的使用”一章中的“报告的生成”
检查通信的连接	参阅“用 uucp 建立远程网络”和“建立 Micnet 局部网络”两章
查找无主的注册区	参阅“用户帐号的管理”一章
删除 core 和 *.out 文件	参阅“文件系统的维护”一章中“维护文件系统中的空闲空间”

每星期做一次的工作	详细介绍的章节
对整个文件系统运行一次 fsck(ADM)	参阅“文件系统的维护”一章中的“文件系统的完整性”
检查打印机缓冲池状态的报告	(从 lpcc)帐号的信箱中查找信息
检查记录文件,如 / etc / wtmp 和那些在目录 / usr / adm 或 / usr / spool 中的文件,并对它们进行清理。	参阅“文件系统的管理”一章中的“文件系统中空闲空间的维护”
使用 sar(ADM)命令生成系统活动报告	参阅“系统性能的调整”一章
建立用户磁盘使用情况的详细报告	参阅“文件系统的管理”一章中“显示磁盘的使用情况”
删除临时文件和丢失后又找到的文件(lost and found files)	参阅“文件系统的管理”一章中的“文件系统空间的维护”

每月做一次的工作	详细介绍的章节
做整个系统的备份	参阅“文件系统的备份”一章
恢复被修改的文件	参阅“使用软盘和磁带机”一章
如果需要,重新调整系统和重新分配资源	参阅“调整系统性能”一章
进行硬件维护	参阅“使用软盘和磁带驱动器”一章及硬件文档
如果需要,修改拨入口令	参阅“使用调制解调器”一章
如果需要,修改根(root)的口令	参阅“用户帐号的管理”一章中的“修改用户口令”
根据需要对操作系统和应用软件进行升级	参阅各软件新版本的安装说明
对软件的使用权限进行审定	使用 fixper 来实现
为文件系统重新分配空间	参阅“解决系统问题”和“文件系统的管理”两章
寻找 SUID 或 SGID 文件,检查文件的所有者和文件大小	使用 find 命令
查找超过 64MB 的巨型文件,并检查它们的功能	参阅“文件系统的管理”一章中的“查找文件”
寻找无主文件	参阅“文件系统的管理”一章中的“查找文件”

### 使用 sysadmsh 简化系统管理

sysadmsh 命令是为简化系统管理而设计的一个菜单接口程序。它所提供的菜单、子菜单及各个屏幕可以在其提供的各选项上进行点取或填空。sysadmsh 可以帮助缺少经验的系统管理员执行 UNIX 命令,否则,系统管理员就必须记住许多东西或不断地翻阅手册。sysadmsh 还提供了上下文相关的帮助信息,在任意一个菜单上,只要按一下 <F1> 键,该菜单项的进一步的解释就会显示出来。为了有助于使用 Sysadmsh,在本书中,所有 UNIX 命令行的下面都附加有该指令用 sysadmsh 来实现的方法参考。例如,custom 这个命令,它是用来往系统上增加软件的,在它的实际的命令下面就是一个 sysadmsh 菜单选择序列。

输入下列命令:

```
custom
```

```
sysadmsh
```

使用者选择: System → Software

这就表示为了实现 custom 这个命令的功能可以先在 sysadmsh 主菜单中选择 System,然后在下一级菜单中选择 Software。这些选择可以通过下列方法来实现:

- 按 <Space> 键在各菜单选择项上做标记,再按 <Return> 键选出你想要的。
- 使用箭头键左右移动光标,再按 <Return> 键,选中希望得到的选项。
- 按要选的选项的第一个字母键,这是最快的方法。还用上面的例子,只要键入 ss (不必按 <Return> 键),就能够进入 custom 菜单。

更多的说明请参阅第三章“使用系统管理工具: sysadmsh”。

### 超级用户

超级用户目录（也就是根 root）是一个用来执行系统维护任务的特殊帐号。它赋予系统管理员以普通用户所没有的不同一般的特权，例如存取系统中所有的文件、执行特权命令等。在本书中所提供的许多任务都要求系统管理员作为超级用户注册进入系统后才能执行。因此，系统管理员必须知道超级用户的口令，这个口令是在系统安装时建立的（参见“系统安装”一章）。

只有在执行系统维护任务时，才以超级用户的身份进行注册。即使只有系统管理员一个人使用这个系统，这个人也必须建立一个普通用户帐号来进行日常工作，保留超级用户帐号仅用来做系统维护工作。

必须只有很少的人知道超级用户口令。缺乏经验的人如果不正确地使用了超级用户的权力，就会造成数据、程序的破坏乃至系统本身的丢失。

### 键盘

在 UNIX 操作系统中，许多键和键的组合有特殊的作用。它们的名称与键盘上所标的不一定相对应。表 1.2 列出了在典型的终端上对于 UNIX 系统有特殊作用的键。许多键还可以由用户用 stty 命令来修改。

表 1.2 特殊键

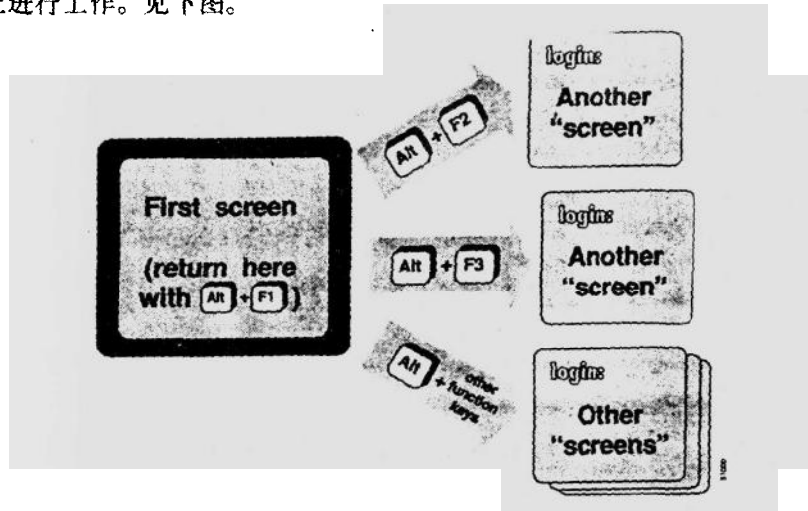
在 UNIX 上的名称	作用
<Return>	结束一个命令行并开始执行此命令。也叫 Enter 键。
<ESC>	退出当前状态;例如,在编辑器 vi 中退出插入状态。
<DEL>	停止当前程序的执行,回到 shell 提示符下。这个键也叫 INTERRUPT 或 DELETE 键。
<BKSP>	删除光标左侧的字符。
<CTL>d	表明键盘输入结束;从当前的 shell 退出,如果当前的 shell 是注册 shell,就会注销。它与 BREAK 键不等价。
<CTL>h	删除光标左侧的第一个字符,也叫 ERASE 键。
<CTL>q	在按<CTL>S 键停止打印后,按此键,则重新开始打印。
<CTL>s	在标准输出设备上(如终端)停止打印。此时程序并未停止执行。
<CTL>u	删除当前行上的所有字符。也叫做 KILL 键。
<CTL>\	退出当前命令的执行,产生一个 core 文件。(建议仅用于调试)

### 使用多屏幕页技术

使用多屏幕页技术，可以同时也在控制台上运行多个程序。可以在不同的屏幕页上查看各程序的显示，但却不必离开控制台。只要按一个组合键，就可以从一个屏幕页切换到另一个屏幕页，并且各屏幕页之间是相互独立的。

在正常注册后，所看到的屏幕页是与 <F1> 键相关的。要另开辟一个屏幕页，只要按住 <Alt> 键，再按 <F2> 或其它功能键。在按了这样一种键的组合之后，屏幕上的内容就被清掉，并且出现等待注册的提示符。再注册一次，就可以在这第二个屏幕页上开始工作。按 <Alt> <F1> 又会立刻切换回第一个屏幕页。像这样在屏幕页之间进行切换，

可以加快速度执行两个不同程序的过程。可以同时执行两个程序，并且可以在任何时间在任何一个是进行工作。见下图。



多屏幕页

例如，可以在第一屏幕页上开始执行 `sysadmsh`，然后按 `<Alt> <F2>` 生成第二屏幕页。再注册一次并且用 `vi` 编辑一个记录文件。使用 `<Alt> <F1>` 和 `<Alt> <F2>` 就可以在这两个屏幕页间来回切换，在 `sysadmsh` 中进行操作，同时在记录文件中进行记录。

也可以同时开两个以上屏幕页。必须先从所有其它的屏幕页注销后，再从第一个屏幕页注销，这样就可以避免剩下没有依附的注册对话区。如果同时有多个屏幕页，按 `<CTL> <PrtSc>` 就能循环访问它们，如果忘记了哪个屏幕页对应哪个进程，那么这个办法是很有用的。

除 `<Alt> + <功能键>` 以外，还可以用 `<CTL> <Alt> + <功能键>` 来改变屏幕页。这在把 `<Alt> + <功能键>` 保留给其它用途的时候是很有用的。这可以用 `mapkey` 命令来设置。

### 系统的安全性

一个非常重要的考虑就是防止系统及其数据未经许可就被非法访问。在这个系统中有一套在其它 UNIX 系统上所没有的安全保护机制，这套机制符合由 the Trusted Computer System Evaluation Criteria (也称作 the Orange Book) 定义的 C2 级“安全级别”。作为系统管理员，可以根据环境的要求来设置这个安全保护机制。也可以建立一个有效的审计系统来提供有关用户注册和系统使用状况的详细报告。在“用户帐号管理”一章中将介绍如何在系统中增加新用户以及如何设置缺省的安全级别，在第二部分第六章“系统安全性维护”中讲述了各种可靠性操作以及审计子系统的特性。

### 用户的培训

为了使用户更好地利用系统的各种资源，不至于因资源超负荷而引起系统故障，系统管理员应该事先向用户讲解后面所列出的事项。用户对系统及其工作限度懂得越多，系统

管理员遇到的麻烦就越少。如果用户对 UNIX 系统是完全陌生的，系统管理员就必须对他们进行一次培训，至少要让他们仔细地阅读一下用户手册，培训内容如下：

- 系统安全级——告知用户系统是否是完全开放，如果不是，哪些目录或文件是可以访问的，哪些系统功能是用用户可以使用的。请参阅“系统安全性维护”一章。

- 如何注册——向用户讲述，如果允许，应如何建立用户口令。必须强调口令要保密以及口令记忆的方法。向用户说明口令长度及类型的限制和修改口令的周期安排、规则、试敲入的次数。告诉用户如何指定他们的终端的类型，如果他们使用的是同一种终端，就帮助他们在 .login 或 .profile 文件中预设定好终端的类型。请参阅“用户帐号管理”一章。

- 如何管理文件和目录——要使用户熟悉基本的命令，如果需要的话，对用户进行培训，或提供命令一览表。

- 如何编辑文件——要使用户切实了解基本的 vi 命令，或是缺省编辑器的基本命令，如果需要，对用户进行培训或提供命令一览表。

- 如何调用程序——向用户介绍必须要用的程序名，所处的位置以及调用命令。

- 如何打印文件——向用户介绍常用程序输出的打印命令，告诉用户作为缺省打印设备的打印机位置，演示添装纸张等操作，如果需要，提供有关打印机的文档资料。

- 良好的通信格式——向用户介绍如何阅读和发送邮件，如何查看邮件表，以及如何通过往信箱文件中增加消息来组织发布信息。演示如何用 mail-f 命令来阅读这些文件，并解释如何清除不想要的信息。特别要强调保持一定量空闲磁盘空间的重要性。

- 良好的文件系统设计——要向用户说明目录的大小是有限的，注册目录和工作目录中的登记项的数目必须小于 30 个，数据文件的存放目录中登记项的数目必须小于 286 个。告诫用户目录是不会变小的，即使其中的登记项被删去也不影响目录大小。不鼓励用户把邮件信息存于其它的文件中，而希望把邮件信息加到现存的信箱文件中。请参阅“文件系统的管理”一章。

- 如何把文件归档——告诉用户如何用 tar 和 cpio 命令把不使用的文件或目录归档存于磁带或软盘上。参见第三部分第三章“使用软盘和磁带驱动器”。

- 如何重新启动混乱的终端——向用户讲解如何从一些最常用的程序中退出，如何用 tset 和 stty same 命令重新启动终端。告诉用户如何在需要的时候对终端重新加电或注销。使用户在遇到终端混乱情况时，能先试用上述几种方法，在全部无效后再来向系统管理员请求帮助。请参阅“解决系统问题”一章。

- 如何消除一个挂起的进程——告诉用户如何在相邻的终端上使用 ps-flu 命令来找到在终端上挂起的进程，并且在系统不造成什么危害的前提下消除这个进程。详见“解决系统问题”一章。

## 机房的环境

在安装或扩展系统之前，请先看一看下面有关系统物理安装的考虑，这可以使得系统上的资源充分发挥效用，也可以使用户更加直接和彻底地使用这些资源。如果要对系统进行扩展，先计划一下，使得各种资源以及连接它们的通信线路适合于这种扩展。

- 把计算机放置在不会被碰撞或挪动的地方。如果可能，它们应被放在自己专用的房间里，没有过多的人员经过。任何一个工作站都应放在稳固的平台上，并不要把电缆线暴



露在过道中。

- 保持计算机房的凉爽并使每台机器都享有良好的通风条件；不要使机器紧挨着墙壁，如果可能，在机房中应有单独的空调设备，并具备非常好的制冷能力。

- 要装备合适的防火设施。

- 把存有文件备份的介质放置在另外一个房间里，不要与计算机放在同一房间，并且那个房间要有防火设施。

如果要安装一个局部网络，就必须认真设计所有机器和外设的连接及位置。合理的设计、使用适当的连接介质和合适的硬件是保证网络长期正常工作的基本条件。

- 如果要把一台打印机连到计算机的一个并行端口上，请把它放在这台计算机附近，但不要把计算机放在通向打印机的过道上。

- 如果要把终端、打印机或其它外设连接到串行端口上，特别是还想继续在规模和复杂程度上扩展系统时，建议选用电话电缆和切换硬件来进行连接。

# 目 录

## 概 述

### 第一部分 安装指南

第一章 安装之前 .....	( 1)
1.1 关于安装介质 .....	( 1)
1.2 安装要求 .....	( 1)
1.3 必须知道的内容 .....	( 3)
1.4 建立和格式化 DOS 分区 .....	( 3)
1.5 响应提示信息 .....	( 3)
1.6 计划磁盘布局 .....	( 4)
第二章 安装过程 .....	( 8)
2.1 安装指令 .....	( 8)
附录 A 安装中可能出现的问题 .....	(22)
附录 B 安装和移去附加软件 .....	(23)
B.1 安装附加产品 .....	(23)
B.2 安装 UNIX 系统软件包 .....	(25)
B.3 从系统中移去软件 .....	(28)
附录 C 建立紧急启动软盘组 .....	(28)
C.1 建立软盘 .....	(28)
附录 D 硬盘配置 .....	(30)
附录 E 设置北美之外的时区 .....	(35)
附录 F 建立附加文件系统 .....	(36)
附录 G 配置非缺省参数的磁带机 .....	(38)
G.1 开始磁带配置 .....	(38)
G.2 标准卡式磁带 .....	(38)
G.3 SCSI 卡式磁带驱动器 .....	(39)
G.4 回到 UNIX 系统安装 .....	(41)
附录 H 安装词汇解释 .....	(42)

### 第二部分 日常管理指南

第一章 启动和停止系统 .....	(45)
1.1 启动系统 .....	(45)

1.2	作为超级用户注册	( 47)
1.3	停止系统	( 48)
1.4	关于启动显示信息	( 49)
1.5	改变引导进程	( 50)
<b>第二章</b>	<b>系统管理工具——sysadmsh</b>	<b>( 51)</b>
2.1	sysadmsh 的屏幕组织	( 51)
2.2	sysadmsh 的操作	( 52)
2.3	sysadmsh 的其它功能	( 60)
<b>第三章</b>	<b>用户帐号的管理</b>	<b>( 69)</b>
3.1	帐号管理	( 69)
3.2	帐号的缺省配置	( 80)
3.3	终端注册管理	( 88)
3.4	活动报告的生成	( 90)
<b>第四章</b>	<b>文件系统和和管理</b>	<b>( 94)</b>
4.1	文件系统的安装和拆卸	( 94)
4.2	文件系统的类型	( 95)
4.3	文件系统的可配置特征	( 96)
4.4	文件系统中空闲空间的维护	( 96)
4.5	维持高效的文件系统组织	(103)
4.6	增加磁盘空间的文件系统	(104)
4.7	文件系统的完整性	(105)
<b>第五章</b>	<b>在同一硬盘上使用 UNIX 和 DOS</b>	<b>(110)</b>
5.1	用 fdisk 进行硬盘分区	(110)
5.2	在 DOS 系统上安装 UNIX	(112)
5.3	在两个硬盘的环境下使用 UNIX 和 DOS	(113)
5.4	从硬盘上删除一个操作系统	(113)
5.5	访问 DOS 的实用程序	(114)
5.6	在 UNIX 系统上安装 DOS 文件系统	(115)
<b>第六章</b>	<b>系统安全性维护</b>	<b>(118)</b>
6.1	什么是可靠系统	(118)
6.2	运行一个可靠系统	(121)
6.3	保护系统上的数据	(125)
6.4	检查系统所受的损害	(128)
6.5	处理文件系统错误	(129)
6.6	有关安全性的错误信息	(136)
6.7	可靠系统上所运行的守护程序	(139)
<b>第七章</b>	<b>审计子系统</b>	<b>(140)</b>
7.1	数据采集	(148)
7.2	文件管理	(154)

7.3	报告生成	(158)
7.4	授予用户审计权力	(171)
7.5	审计术语	(172)
<b>第八章</b>	<b>UNIX 目录和特殊设备文件</b>	<b>(174)</b>
8.1	UNIX 目录	(174)
8.2	记录文件	(177)
8.3	特殊设备文件	(177)
<b>第九章</b>	<b>文件系统和备份</b>	<b>(180)</b>
9.1	使用 sysadmssh 进行备份的策略	(180)
9.2	例行备份的准备	(181)
9.3	进行例行备份	(186)
9.4	进行非例行备份	(187)
9.5	备份的核查	(189)
9.6	获取备份表	(189)
9.7	从备份中恢复特定的文件或目录	(190)
9.8	恢复完整的文件系统	(191)
9.9	备份级的概念	(192)
<b>第十章</b>	<b>系统性能的调整</b>	<b>(195)</b>
10.1	利用 configure 重新分配核心资源	(196)
10.2	修改系统配置	(197)
10.3	性能的重新配置	(198)
10.4	规定有效的系统使用模式	(201)
10.5	对低效系统进行诊断	(202)
10.6	调整磁盘效率	(206)
10.7	可调系统参数的说明	(207)
<b>附录 A</b>	<b>系统启动的预设置</b>	<b>(217)</b>
<b>附录 B</b>	<b>作业调度命令的使用授权</b>	<b>(222)</b>

### 第三部分 外设使用指南

<b>第一章</b>	<b>增加多功能卡、内存卡及其它总线卡</b>	<b>(226)</b>
1.1	安装总线卡	(226)
1.2	增加存储器	(227)
1.3	增加及配置串行端口	(228)
<b>第二章</b>	<b>打印机</b>	<b>(230)</b>
2.1	安装打印机	(230)
2.2	打印机配置选项	(233)
2.3	用户命令综述	(242)
2.4	管理命令综述	(243)

2.5	增加本地打印机	(244)
2.6	启动和停止打印服务	(246)
2.7	删除打印请求	(246)
2.8	使打印机有效 / 无效	(247)
2.9	在一组打印机中增加一台	(247)
2.10	设置系统缺省打印执行者	(248)
2.11	安装表格或打印字型轮	(248)
2.12	移去打印机或打印机组	(249)
2.13	管理打印负载	(250)
2.14	管理队列优先级	(252)
2.15	表格与处理程序	(255)
2.16	使用请求登记中的信息	(256)
2.17	调整打印服务	(257)
2.18	建立 RTS / CTS 协议的串行打印机	(264)
2.19	使用不带假脱机程序的打印机	(266)
2.20	建立初始设备文件	(266)
<b>第三章</b>	<b>使用软盘和磁带驱动器</b>	<b>(267)</b>
3.1	使用磁带驱动器	(267)
3.2	使用软磁盘	(272)
<b>第四章</b>	<b>使用鼠标</b>	<b>(277)</b>
4.1	安装硬件	(277)
4.2	安装鼠标	(277)
4.3	使用鼠标	(280)
<b>第五章</b>	<b>增加硬盘</b>	<b>(281)</b>
5.1	在开始之前	(282)
5.2	安装硬盘	(284)
5.3	增加新的文件系统	(290)
5.4	操作系统核心的重连接	(292)
5.5	将用户帐号移出主硬盘	(292)
<b>第六章</b>	<b>终端的管理</b>	<b>(293)</b>
6.1	增加终端	(293)
6.2	改变缺省的终端线特性	(295)
6.3	设置终端类型	(299)
6.4	自动设置终端类型	(300)
6.5	去掉一个终端	(300)
6.6	建立串行控制台	(301)
6.7	利用 mscreen 使用串行多屏幕	(301)
<b>第七章</b>	<b>硬件的兼容性</b>	<b>(307)</b>
7.1	兼容的硬件	(307)

7.2	如何使用本章	(307)
7.3	硬件支持	(307)
7.4	标准结构的 386 类机器	(307)
7.5	微通道结构的 386 机器	(309)
7.6	一般兼容性标准	(310)
7.7	标准结构机器的硬件兼容性	(315)
7.8	微通道结构硬件的兼容性	(322)
附录 A	使用系统控制台和彩色显示器	(325)

## 第四部分 网络通信指南

<b>第一章</b>	<b>使用调制解调器</b>	(330)
1.1	选择串行端口	(330)
1.2	配置调制解调器	(330)
1.3	安装拨号输出调制解调器	(331)
1.4	建立拨号输入调制解调器	(334)
1.5	安装拨号输入/输出共享的调制解调器	(335)
1.6	拨号输入你的计算机	(335)
1.7	从你的计算机拨号输出	(336)
<b>第二章</b>	<b>建立 Micnet 局部网络</b>	(337)
2.1	网络设计	(337)
2.2	建立网络	(340)
2.3	启动网络	(343)
2.4	测试 Micnet 网络	(343)
2.5	停止网络	(345)
2.6	修改 Micnet 网络	(345)
<b>第三章</b>	<b>用 UUCP 建立远程网络</b>	(346)
3.1	简介	(346)
3.2	将两个本地系统直接线连	(350)
3.3	用调制解调器连接远程 UUCP 系统	(352)
3.4	在系统中配置 UUCP	(356)
3.5	特殊的 UUCP 配置选项	(369)
3.6	管理 UUCP 系统	(370)
3.7	完整的 UUCP 举例	(372)
3.8	UUCP 的错误信息	(375)
<b>第四章</b>	<b>建立电子邮件</b>	(378)
4.1	引言	(378)
4.2	MMDF 是如何工作的	(380)
4.3	配置 MMDF	(390)

4.4	MMDF 系统的维护 .....	(407)
4.5	改变机器名 .....	(408)
4.6	Internet 地址 .....	(408)

## 第五部分 实用参考

<b>第一章</b>	<b>解决系统问题 .....</b>	<b>(410)</b>
1.1	解决引导问题 .....	(410)
1.2	解决控制台问题 .....	(415)
1.3	解决文件系统的问题 .....	(417)
1.4	解决安装中的问题 .....	(422)
1.5	解决注册问题 .....	(423)
1.6	解决邮件故障问题 .....	(424)
1.7	使用调制解调器可能出现的问题 .....	(426)
1.8	打印机系统故障排除 .....	(430)
1.9	解决进程管理中的问题 .....	(434)
1.10	解决调度问题 .....	(436)
1.11	解决磁带驱动器的问题 .....	(437)
1.12	解决终端问题 .....	(439)
1.13	解决 UUCP 连接的问题 .....	(441)
1.14	排除其它系统故障 .....	(448)
<b>第二章</b>	<b>常用命令解释 .....</b>	<b>(453)</b>

# 第一部分 安装指南

## 第一章 安装之前

本章介绍如何在计算机上安装 UNIX 操作系统。在开始安装工作之前应熟悉每一部分的步骤及其相关术语。

### 1.1 关于安装介质

UNIX 系统可存储于卡式磁带或软盘上，分别描述如下。

存储 UNIX 系统的软盘包括以下内容：

- 两张标志为 N1（根）和 N2（文件系统）的磁盘，加上一些标有 N3, N4, N5... 的 N 类软盘；
- 一张标志为 M1 的软盘，即主安装盘；
- 一系列标志为 B1, B2, B3.....的基本实用程序软盘；
- 一系列标志为 X1, X2, X3.....的扩展实用程序软盘。

UNIX 的磁带系统包括以下内容：

- 两张标志为 N1（根）和 N2（文件系统）的软盘，加上几张标志为 N3, N4.....的 N 类盘；
- 一张标志为 M1（主安装盘）的软盘；
- 一盘 1/4 英寸 QIC 卡式磁带。

虽然 UNIX 系统的存储介质不同，但本书中的指令对两种介质的 UNIX 系统都适用。在必要的地方，会出现提示信息以区分这两种安装指令。一般来讲，使用磁带安装的方法只处理关于 N 类盘和主安装盘的提示信息，其余可忽略。

### 1.2 安装要求

以下是安装 UNIX 所必要的物质条件：

- 带有足够内存的计算机，对系统 V 版本 3.2.2 来说至少要 2M 内存，推荐内存大小为 3M 字节或更多。
- 安装软件所必需的磁盘空间，对系统 V 版本 3.2.2 来说至少要 40M 空间。具体使用多大的磁盘空间还取决于计划安装多大规模的 UNIX 系统。
- 如果希望将系统全部装入，则需要极大的磁盘空间；如果只是安装系统的子集，则为这部分软件留出磁盘空间即可。

下面给出 UNIX 系统各部分软件包的大小：

package	Size(单位为块,每块 512 字节)	Use / Contents
All	33842	Entire Operating System set
Backup	256	System back up and recovery tools
BASE	1406	Basic extended utility set



package	Size(单位为块,每块 512 字节)	Use / Contents
CSH	116	The C-shell
KSH	282	The K-shell
DOS	368	DOS utilities
EX	408	The ex and Vi editors
FILE	480	File manipulation tools
LAYERS	194	System V Layers
LPR	3202	Multiple line printer spooler
MAIL	4944	Electronic mail and micnet
MAN	3764	Operating System Manual Pages
MAPCHAN	144	International character set mapping
MOUSE	68	Mouse and graphic input devices files
SysAdM	818	Additional system administration tools
TPLOT	350	Tplot Graph and Spline
UUCP	2280	UUCP and Cu communication utilities
LINK	6816	System V Link kit files

各软件包的内容解释如下:

**BACKUP:** 备份文件所必需的实用程序。使用本软件包,可以备份所有文件或者只备份最近修改的文件,也可以恢复备份文件的任一部分。

**BASE:** 通常在实时系统安装的基本系统实用程序。

**CSH:** C-shell 命令解释程序。

**KSH:** 即 korn shell, 是支持作业控制、命令行编辑以及其他许多高级特性的命令解释程序。

**DOS:** 支持将 MS-DOS 文件移动、拷贝到 MS-DOS 磁盘或 MS-DOS 分区,以及从 MS-DOS 磁盘或 MS-DOS 分区拷贝文件的实用程序集合。

**EX:** 包括 ex (c) 编辑器及 vi (c) 屏幕编辑器。

**FILE:** 用以比较和操作文件的一系列有用程序。

**LAYERS:** 用于 AT&T BLit 终端的窗口机制。要使用本软件包,驱动程序必须连接到核心。

**LPR:** 允许多份打印作业排成队列以便自动打印的打印池。

**MAIL:** 包括允许电子邮件在系统之间以及系统之内进行传输的多个子系统。允许文件转换及远程命令执行的 Micnet 局部网络以及 MMDF 邮件处理系统都在本软件包内。

**MAN:** 本软件包包含系统管理 (ADM), Commands (C), File Formats(F),Hardware Dependent(HW)和 Miscellaneous(M) 指南。

**MAPCHAN:** 允许使用采用不同字符集的硬件。例如,使用某一字符集的终端可以将一个文件传送到使用另一个字符集的打印机去而不会导致混乱。

**MOUSE:** 支持鼠标等设备的实用程序。值得注意的是在这些实用程序使用之前应先使用 mkdev mouse 命令重建核心 kernel。

**SYSADM:** 用以管理系统的 sysadm (ADM) 系统管理菜单接口及其他工具。

**TPLOT:** 产生图形的实用程序。这些程序只能在一部分硬件上工作。

**UUCP:** 用以局部地连接系统或者将系统连到由其他系统构成的广域网络的实用程序。

**LINK:** 如果你打算给系统增加设施,如磁带驱动器、多端口串行卡、鼠标、

MS-DOS 文件系统支持, 则必须重建系统核心, 这时就会用到本软件包提供的工具。

安装 UNIX 系统还会用到:

- 一个软盘驱动器
- 如果 UNIX 系统存在于磁带上, 则还需一个 1/4 英寸的卡式磁带驱动器
- UNIX 系统存储介质, 磁盘或磁带
- 一个串行号 (在 N1 盘上给出)
- 一个 activation 键 (在 N1 盘上给出)

### 1.3 必须知道的内容

在开始安装过程之前, 首先保证计算机是完全装配好的 (能工作的), 并且应熟悉其操作。特别应该知道:

- 如何开关机器
- 如何把软盘插入软盘驱动器
- 如何复位计算机

如果是首次装配机器, 或者对以上所列项目尚无把握, 请有重点地查阅机器和硬盘的随机手册。必须按照制造商的技术说明把硬盘连到计算机上, 并且建议按照操作员指南运行系统自检程序, 以便检查可能的硬件故障。

---

注意: 如果计划对硬盘建立 DOS 分区, 必须首先安装 DOS。

---

### 1.4 建立和格式化 DOS 分区

如果想在系统上拥有一个 DOS 分区, 则必须在安装 UNIX 系统之前建立。用随机的实用程序建立及格式化 DOS 分区。DOS 分区至少应 2.5M 字节大小。有一些 DOS 拷贝-保护机制不能安装于小于 2.5M 字节的分区上。

如果已经安装了 UNIX 系统而没有物理 DOS 分区, 必须先备份硬盘, 建立 DOS 分区, 然后重新安装 UNIX 系统及应用程序。

### 1.5 响应提示信息

按以下规则响应提示信息:

- 按照要求输入名字数字等
- 一行以按 <RETURN> 键为结束

---

注意: <RETURN> 有时标为 <ENTER> 键

---

- 用 <BKSP> 键或 Ctrl-h 来删除一个字符
- 按 ctrl-u 来删除全行

---

注意: 按 ctrl-? 键的意思是在按 ctrl 键的同时按另一个键, 这与 <shift> 键的使用是一样的。

---

另外,

- (y/n)? 是在问“yes or no”, 通常要求用“y”或“n”作出肯定或否定的响应
- 文中出现的命令常在后面的括弧里标注出参考部分, 如 Cat (C)。

## 1.6 计划磁盘布局

如果系统拥有大量的用户，开发任务很重或者计划使用一个数据库程序，则建议认真安排硬盘的布局。负荷较重的系统中，布局决策会影响性能。

本节讨论硬盘的两个片段（非根 / u 文件系统和对换区）及与其密切联系的 UNIX 根文件系统。由于 DOS 分区影响可用硬盘空间的大小，所以也将提及。

### • 对换区 (swap space)

对换区是硬盘的一部分，充当物理内存 RAM 的扩展部分。处于激活 (active) 状态但目前处于等待状态的程序的某部分可以转移到该区（称为对换出去）以便让其他程序可以在 RAM 中运行。如果系统拥有许多用户或者要执行象数据库这样的大的应用程序，就要求对换区必须很大。

### • 独立的非根文件系统 (/u)

一个拥有许多用户的容量较大 (140M 字节以上) 的硬盘，如果根文件系统太大，速度就会下降。为用户建立独立的非根 (/u) 文件系统可以提高性能，使备份工作变得容易，并帮助保护文件系统免受许多意外事故的袭击和破坏。

### • DOS 分区

希望由 DOS 来分享硬盘的人应该查阅本书相应章节，首先在系统硬盘上装载好 DOS。DOS 分区的存在对计划好磁盘资源更为重要，因为在磁盘布局上微小的变化都要求重新安装操作系统。在安装 DOS 之前应仔细阅读本节，作好磁盘的布局。

在本节结尾处有几张简单的表格，用以计算磁盘分区的大小。虽然这几张表格似乎令人望而生畏，但总的来讲计算过程是相当机械的，只要求填入几个简单数值即可。

### 1.6.1 决定对换空间分配

当物理存储器不够存放系统上运行的所有进程时，处于 sleeping 的时程就被移到称为“对换空间”的区域。对换空间是硬盘保留的一块，其大小应该与你安装的 RAM 大小一样。对换空间是在操作系统安装之时分配的。以下是计算对换空间大小的两种办法。

#### 1. 让 divvy (ADM) 为你作决定

如果系统处于典型的多用户操作状态（而不是在运行大的应用程序或扩展系统），则可用在安装过程中推荐使用的缺省对换区大小。

在 badtrk (ADM) 执行之后，系统提示你选择分配给对换空间的空间大小。divvy 根据硬盘大小计算出对换空间大小的最大值和最小值，将要用到下面的表格（其中一块为 1024 字节，即 1K 字节）：

如果 UNIX 分区大于	但小于	缺省对换区分配为
40M 字节 *	50M 字节	分区的 20%
51M 字节	139M 字节	10M 字节
140M 字节	—	$[10M \text{ 字节} + (\text{分区大小} - 140) \times 0.1] * *$

上表中，\* 指希望 UNIX 分区不小于 40M 字节，\*\* 指希望对换区不大于 64M 字节。

如果计划安装扩展系统，则对换区大小至少应比默认值大 1000 块，特别是当计划使用像 databases 和 spreadsheets 那样的大型应用程序，或基于性能上的考虑，打算将频繁使用的程序常驻内存时，这一点就更为重要。

## 2. 自己计算对换空间大小

使用大的应用程序（例如一个大的关系数据库）的系统应该根据下面公式计算对换空间大小（使用本节尾处理提供的表格。）

(1) 用系统中的用户数乘以通常在系统中的最大进程大小（单位为 1K 字节）。

(2) 用机器上安装的内存数量加上 256K，与第一步计算的结果进行比较，取较大值作为对换空间大小。

(3) 如果多个用户正在运行不同的大型应用程序，则应调整第二步的结果。每个不同的大型应用程序增加 1M 字节。（认为 Development System 是一个大的应用程序）。

例如，计算 8 个用户的机器，带有 4M 字节内存，使用电子表格 (spreadsheets)、数据库 (databases) 字处理和图形包 4 个应用程序，此时的对换空间大小：

(1)  $8 \text{ users} \times 1\text{M 字节} = 8\text{M 字节}$

(2)  $4\text{M 安装内存} + 1\text{M 字节} = 5\text{M 字节}$ ，小于第一步的结果。

(3) 有 4 个用户使用不同的大型应用程序， $4 \times 1\text{M 字节} = 4\text{M 字节}$ 。将此数加上第一步和第二步结果中较大者即得到合理的对换空间大小： $8\text{M 字节} + 4\text{M 字节} = 12\text{M 字节}$ 。因为操作系统使用 1K 大小的块，这就意味着对换空间分配的大小应为 12000 块。

## 1.6.2 决定是否要独立的非根文件系统

如果是在大于 240Mbytes 硬盘上安装系统，就会被问及是否要把 UNIX 分区分成两个独立的文件系统。这时如果 UNIX 分区大小减去根文件系统大小再减去对换空间大小之差大于 20M 字节，则应使用独立的非根文件系统。

即使分区大小小于 240M 字节，也可以建立附加文件系统，这一文件系统通常称为 /u，不过可以把它称作除 root 之外的任何名字。/u 文件系统常用作用户帐号和文件。使用 /u 文件系统既有优点也有缺点。

如果是在一个实时系统（即系统中没有安装或几乎没有安装操作系统软件包）上工作，应用程序很简单或者不用存储用户文件，则根本没有必要建立 /u 文件系统。

建立独立的文件系统有几大优点：

### • 更易于系统备份

在一般系统中，用户文件经常变动，而 UNIX 实用程序和系统文件一旦建立好，则极少变动。大多数用户只是在自己的帐号下频繁地建立和修改文件，所以用户帐号下内容应经常备份，而根文件系统则只需要偶尔加以备份。建立了独立的文件系统之后，对单个文件系统进行备份只需要少量的存储介质（例如，较少的软磁盘，较少的磁带，较小的磁带容量等）。

### • 较小的文件系统速度较快

由于在文件访问时操作系统要搜索的文件标识符和数据块少了，速度随之加快。每个文件系统都有自己的标识节点和数据块区域。另一个原因是当系统非正常关闭（遇到意外事故）时，要用 fsck (ADM) 检查和清理整个文件系统，而处理两个较小的文件系统所

花时间少于处理一个较大文件系统所花时间。

- 当出现意外事故时，非当前文件系统所遭受的损失要小于当前文件系统。因此，减少在根文件系统下操作，可以使根文件系统遭受破坏的概率大大减小。而非根文件系统即使遭到破坏，也可以很容易地从备份中恢复，得以重建。

/u (非根) 文件系统有两个缺陷，但都与管理而不是性能有关：

- 硬盘划分一旦完成，要想改变一个文件系统的大小或者删除一个文件系统就几乎不可能。如果一定要做，就要耗费大量时间。

因为为了做到这一点，必须备份整个 UNIX 分区，再用 divvy (ADM) 实用程序改变文件系统大小，然后再将整个系统从备份中恢复。这几步工作只能由有经验的系统管理员来做。有时还不如重新安装系统容易。

- 由于有额外的文件系统，使操作系统复杂度大大增加。这样一来系统管理任务对新手来说就更复杂。如果增加的文件系统不是在启动时自动安装的，则必须记住安装。

#### 确定 /u 文件系统空间

要确定硬盘上是否有足够的空间留给 /u 文件系统，应按以下描述的步骤完成计算。(公式很繁琐，先在这里浏览一下，再使用本节结尾处给出的表格)。

- 1) 将你打算安装在系统上的所有操作系统软件包大小加在一起。软件包的大小在本章的前部已有说明。

- 2) 计算 /tmp 目录的大小。/tmp 通常被许多 UNIX 程序用作暂时存储区。为每个用户至少安排 500K 字节的空间。如果有的用户要有大型数据库，编辑大文件等，则再为每个用户留出 500K 字节的空间并累加在一起。

- 3) 注意给根文件系统上的应用程序留出必需的磁盘空间。大多数应用程序都会在其手册或使用说明注明所需硬盘空间的大小。

- 4) 将第 1)、2)、3) 步的结果加在一起，总和表示完整的根文件系统大小。

- 5) 另外，在所有软件安装完毕之后必须留出根文件系统的 10-20% 作为空闲空间。将第 4) 步所得根文件系统大小加上自身的 20%。例如，假设根文件系统大小为 40M 字节，为了保持 20% 空间空闲，应该在根文件系统大小 40M 字节上加上 8M 字节，得到总数为 48M 字节。如果文件系统的空间空闲量低于这个数量，操作系统在需要空闲空间时将花费大量时间去寻找，这就带来性能上的损失。

- 6) 从 UNIX 分区 (如果 UNIX 系统是唯一的分区，则从整个硬盘) 大小中减去根文件系统的需求空间大小 (按第 5) 步的方法计算)。在考虑增加 /u 文件系统之前硬盘上至少应剩下 10M 字节的空闲空间。

- 7) 计算 /u 文件系统的需求。在 /u 文件系统上应至少为每个用户留出 1.5M 字节的磁盘空间。加上 /u 文件系统中数据库的需求，该需求决定于文件所能达到的最大量，决定于多少用户将使用数据库以及使用数据库的用户之间将要共享什么文件等因素。再加上将驻留 /u 文件系统的应用程序大小。如果 /u 文件系统需要的空间小于第 5) 步计算所得到的结果，那么你可以放心地增加一个 /u 文件系统了。

- 8) 将 /u 文件系统大小的兆 (M) 字节数乘以 1000，得到分配给 /u 文件系统的块数。(1K 字节/块)。

### 1.6.3 磁盘规划计算表格

用以下表格完成上面讨论的计算。

表 1.1 对换空间分配

1.用户数	_____ × 1M 字节	= _____
2.安装的内存量	_____ + 1M 字节	= _____
3.(1)和(2)中较大的		= _____
4.大的应用程序数目	_____ × 1M 字节	= _____
5.(3)和(4)的总数和		= _____

表 1.2 硬盘划分

DOS 分区: _____
UNIX 分区: _____
对换空间: _____
/u 文件系统: _____

表 1.3 /u 文件系统分配

指 令	计 算	结 果
1.要安装的操作系统软件包大小(将块数转化为 M 字节)	_____ / 2000	= _____
2.用户数目 × $\frac{1}{2}$ M 字节 / 用户	_____ × 0.5M 字节	= _____
3.使用数据库或 DOS 的用户数乘以 $\frac{1}{2}$ M 字节 / 用户	_____ × 0.5M 字节	= _____
4.将(2)、(3)结果相加	.....	= _____
5.应用程序所需磁盘空间	.....	= _____
6.根文件系统大小:从(1)加到(5)再增加 20%	_____ × 1.2	= _____
7.从 UNIX 分区(或整个硬盘)减去根文件系统大小	_____	= _____
8.如果(7)小于 5-10M 字节,则不能建立 /u 文件系统	.....	= _____
9.用户数 × $\frac{1}{2}$ M 字节 / 用户	_____ × 1.5M 字节	= _____
10.驻于 /u 中的应用程序大小	.....	= _____
11.要驻于 /u 是的数据库文件大小	.....	= _____
12.从(9)加到(11)得 /u 文件系统大小	.....	= _____
13.如果(12)结果小于(7)的结果,则继续,否则不能建 /u 文件系统	.....	.....
14.将(12)的结果转化成块	_____ × 1000	= _____

## 第二章 安装过程

在开始安装过程以前应仔细阅读本章，以便熟悉各部分不同的步骤及术语。本章描述了如何安装 UNIX 系统。

安装过程包含以下几个阶段：

- 用软盘 N1 和 N2 启动系统
- 准备好硬盘以便安装
- 用经过初始化的硬盘启动操作系统
- 安装所需要的 UNIX 系统的部分内容，并装载自己的应用程序
- 用菜单驱动的 Sysadmsh 系统管理 Shell 程序来配置你的系统，包括增加用户帐号，选择自己所希望的安全级别等
- 准备好系统以备一般的使用，包括设置系统根口令，并重新启动系统使得改变过的配置信息生效。

### 2.1 安装指令

详细指导 UNIX 系统的安装过程。如有必要，会指导你参考本书附录中内容。

---

注意：在硬盘分区时如果希望建立 MS-DOS 分区，必须在安装 UNIX 系统之前建立 DOS 分区。如果没有这样做，UNIX 分区就会不可启动。建立 DOS 分区时参考 MS-DOS 的安装资料。参考第二部分第五章中“建立与格式物理 DOS 分区”的内容。

---

按照以下步骤执行，记住在键盘上键入响应之后应按 <RETURN> 键。

- (1) 找出标有 N1 (或“boot”盘) 和 N2 (或“filesystem”盘) 的软盘
- (2) 将 N1 (boot) 软盘插入驱动器。如果有不止一个的驱动器，则应用主驱动器 (有时称为引导驱动器)。如果不能确认哪个驱动器是主驱动器可查阅计算机硬件手册
- (3) 打开计算机电源开关。计算机把 UNIX bootstrap 程序装入内存并执行。在屏幕的左上角显示计算机所安装的内存总数。接着显示：

```
sco System v / 386
```

```
Boot
```

```
:
```

```
Press <Return> to boot from the floppy drive.
```

---

注意：如果你需要从头开始安装过程，在提示符“boot”出现时应输入“restart”，系统即会从头开始安装过程。

---

- (4) 接着屏幕显示：

```
fd(64)unix root = fd(64) swap = ram(0) pipe = ram(1) swablo = 0 nswap = 16 ronly
```

(5) 过一会儿之后, 屏幕显示:

```
loading .text
```

```
.....
```

```
.....
```

```
.....
```

```
loading .data
```

```
.....
```

```
loading .bss
```

在软件装载时进展情况会显示到屏幕上。

(6) 接着提示你插入下一张软盘:

```
Insert N2 (Filesystem) floppy and press <RETURN>
```

取出 N1(boot)软盘, 插入 N2 (FILESYSTEM) 软盘然后回车。

(7) 在操作系统核心装入内存之后, 系统显示内存分配信息和其他配置信息, 类似:

```
device      address          vector dma comment
%serial     0x03F8-0x03FF  04    -- unit = 0  type= Standard nports= 1
%serial     0x03F8-0x02FF  03    -- unit = 1  type= Standard nports= 1
%floppy     0x03F2 0x02F7  06    2  unit= 0  type= 96ds15
%console    ..             -      unit= ega  type= 0 12 screens= 68K
%parallel  0x0378-0x037A  07    -  unit= 0
%disk       0x01F0-0x01F7  16    type= W0  unit= 0  cyls= 286 hds= 16 secs= 63
mem: total= 3712k, kernel= 1436k < user= 2276k
rootdev= 1 / 40, swapdev= 1 / 41, pipedev= 1 / 40
kernel: i / o bufs= 4681
```

系统执行自检程序以确定是否检测到硬件错误。每一步都产生从 A 至 Z 的一个字母, 显示后覆盖的速度太快以致于看不太清。

(8) 如果在“Z”出现之前显示停止, 并且没有启动提示符出现, 则应按计算机手册中讲述的方法运行硬件诊断程序, 改正发现的错误, 然后从头重新开始安装过程。

(9) 系统一旦开始运行, 会显示如下菜单:

```
Setting installation display environment...
```

#### Keyboard Selection

1. Ameican
2. British
3. French
4. German
5. Italian
6. Spanish

Use the Numeric Keypad if present, using <Num Lock> if necessary, to select one of the above options:

选择与你键盘对应的数字。

(10) 下面是硬盘初始化过程:

```
System V Hard Disk Initialization
```

```
The primary hard disk in the system will now be initialized.
```



This will create a UNIX partition on the disk and divide it into filesystem (s) and a swap space.

You can choose a fully configurable disk initialization, which requires you to set the disk parameters, specify the size of the UNIX partition, and control the layout of filesystems and swap area. You will be presented with defaults at each selection.

You can also choose an automatic disk initialization that creates a complete disk layout and configuration using system default values.

Choose one of the following:

1. Fully Configurable Initialization
2. Automatic Initialization ( use system defaults )
3. Exit Installation

Enter your choice:

如果决定使用“Automatic”任选项，应输入“2”并按回。这一选项初始化硬盘而不提示输入磁盘参数及文件系统大小等等信息。如果选择了“Fully Configurable”选项，则应按照附录 D“Running the Configurable Initialization”的步骤进行。

(11) 接着会显示确认信息:

Verify Automatic Disk Initialization

You have chosen to initialize the primary hard disk automatically using system defaults.

This option requires that your system was set up to recognize the hard disk at the factory, or that you have run the proper setup floppy disk prior to system V Operating System installation.

The hard disk initialization will preserve any pre-existing DOS partition, but will overwrite any non-DOS partitions.

A single, active UNIX partition will be created and divided into a root filesystem, a swap area, and if the UNIX partition is 245 megabytes or larger, a user ( / u ) filesystem.

Setting up the hard disk may take as long as one minute per megabyte of space on the disk.

Are you sure you wish to do this ( y or n ):

回答“y”并回车。

(12) 首先，会看到描述所安装硬盘的一行信息。类似:

```
%disk 0x01F0-0x01F7 16 - type= W0 unit = 0 cyl = 286 hds = 16 secs = 63
```

对 SCSI 硬盘，将显示两行信息。

(13) 接着会看到:

Creating UNIX partition

---

注意: 如果用的是 SCSI 或 SMS-OMTI ESDI 控制器, 则硬盘不用扫描坏道, 应该跳过下一步。所有其他 EDSI 硬盘或控制器都应进行扫描。

---

(14) 如果所安装的硬盘是 ST506 或者 ESDI 磁盘, 硬盘要扫描 flaws 并把它们记录

到坏道表中。将看到如下信息:

```
Scanning disk...
```

```
Destructively scanning track x / y. n % of scan completed
```

这一步花费的时间较长: 大约每兆硬盘空间需要一分钟。

(15) 显示以下提示信息:

```
Dividing UNIX partition into filesystem and swap divisions...
```

```
Making filesystems
```

```
Setting up hard disk root filesystem...
```

每条信息之间要隔几分钟。

(16) 这一步完成之后, 将看到:

```
Initialization of the hard disk is now complete.
```

注意: 如果要增加附加硬盘, 应首先完成安装过程。然后再按本过程后部给出的指令进行操作。

(17) 系统关闭, 显示新初始化的硬盘启动的指令:

```
AFTER You see the message * * Safe to Power Off * *
```

```
or
```

```
* * Press Any Key to Reboot * *
```

```
remove the N2 (Filesystem) floppy from the floppy drive and
```

```
insert the N1 (Boot) floppy. (The floppy light may stay on.)
```

```
Close the floppy door, then press any key.
```

```
The screen will clear and you will see the boot message:
```

```
Boot
```

```
:
```

```
Press <Return> to reboot the system and continue the installation.
```

(18) 按指令移去 N2 软盘, 并插入 N1 盘。按回车键重新启动系统。将会看到如下提示:

```
SCO System V / 386
```

```
Boot
```

```
:
```

(19) 确信 N1 (boot) 软盘是在驱动器中并按回车键, 将看到以下信息:

```
fd (64) unix root = hd(40) swap = hd(41) pipe = hd(40)
```

后跟在安装开始时显示过的引导点信息。

(20) 过一会儿之后显示版权信息, 之后是系统的配置信息。

和以前一样, 系统执行自检程序以检测出可能存在的硬件故障。显示从 A 至 Z 的字母, 每一个都覆盖上一个字母。如果在“Z”出现之前字母显示停止, 应按计算机手册中讲述的方法运行硬件诊断程序。纠正发现的错误, 从头开始安装过程。

(21) 接着, 再次显示环境信息:

```
Setting installation display environment...
```

(22) 现在要求确信 N1 (boot) 软盘是在软盘驱动器中。将看到如下信息:

```
Verify that the N1 (Boot) floppy is in the drive and the
```

floppy door is closed and press < Return >

(23) 接着要求插入 M1 (主安装盘) 盘:

Insert Master Installation Floppy

and press < Return > or enter q to quit:

注意, 如果出现错误, 则显示信息:

Extraction error, try again? (y / n)

确信使用的软盘是正确的一张, 且软盘驱动器的门是关闭的, 然后输入“y”之后回车, 如果错误信息仍然出现, 则参阅附录 A“Troubleshooting Your Installation”中“读盘错”部分。

(24) 系统提示插入标志为 B1 至 B4 的软盘。提示信息类似:

Insert SCO System V Runtime System Floppy Volume B1

and press < Return > or enter q to quit:

按提示插入相应的 B 盘。

(25) 接着系统提示插入标志为 N1-N3 的软盘, 如果插入软盘的次序不对, 则会看到:

Error: Incorrect volume in drive

从驱动器中移去该软盘, 插入正确标志的软盘, 并回车。

(26) 在插入 N3 之后, 会看到:

SCO System V Operating System Serialization

When prompted, use the serial number and activation key included with the SCO System V Operating System distribution

Enter your serial number or enter q to quit:

将 serialization 卡上出现的一串数字输入并按回车键。会看到如下信息:

Enter your activation key or enter q to quit:

输入 serialization 卡上的 activation 键, 回车。如果输入出错, 显示:

Error: Invalid activation key

Do you wish to try activation again? (y / n)

输入“y”并回车, 重新输入代码。如果输入“n”, 则安装失败。

(27) 如果 UNIX 系统的存储介质是磁带, 则提示确定磁带机是否已配置好。

---

注意: 如果标准卡式磁带配置的地址不是下面推荐值之一, 则会显示 SCSI 的信息而不是卡式磁带的信息, 因为系统不能识别到底在用哪一个。

---

对标准的卡式磁带机来说, 会出现:

A tape has been detected at address 0x0338-0x033C. Is it set at the default interrupt of 05 and the default dma channel of 1? (y / n)

对 SCSI 卡式磁带机来讲, 会看到:

Do you have a SCSI tape at the default setting of

Host Adapter 0, ID 2 and LUN 0? (y / n)

如果磁带机的信息是正确的，输入“y”并回车。如果磁带机的配置不是缺省参数，则输入“n”然后回车，参照附录 G, “Configuring a non-default Tape Drive”的内容执行。然后系统要求插入卡式磁带：

Insert SCO System V Tape Volume 1

and press < Return > or enter q to quit:

按要求插入磁带并回车。过一会儿之后，系统要求你插入部分 N 盘。

(28) 接着，系统提示根据所处的地域设置时区，如果是在北美洲之外的地区，则应参考附录 E “Setting Time Zones Outside North America”的说明，设置各自的时区。首先看到：

Time zone initialization

Are you in North America? (y / n)

如果不在北美洲，输入“n”然后按附录 E 的要求设置时区。否则，应输入“y”。将看到如下菜单：（假定输入了“n”）

- (1) NST-Newfoundland standard Time
- (2) AST-Atlantic standard Time
- (3) EST-Eastern standard Time
- (4) CST-Central standard Time
- (5) MST-Mountain standard Time
- (6) PST-Pacific standard Time
- (7) YST-Yuron standard Time
- (8) HST-Hawaiian / Alaskan standard Time
- (9) NST-Nome that represents your time zone or enter q to quit

如果时区是非标准时间，就应该输入数字 9 并且回车。

(29) 可看到如下信息：

Does daylight savingst time (summer time) apply at your location? (y / n)

如果所在的地区采用了夏令时和标准时间的更替，就输入“y”；否则，输入“n”，在 /etc/TZ 中的时区变量作相应变化。时区设置完毕。

(30) 接着，进行 terminfo (M) 数据库的编译，这将要花费几分钟：

Setting up terminfo database. This will take a moment...

(31) 这一步工作完成之后，显示下列信息：

Checking file permissions ...

(32) 系统询问所希望有的安全类型：

Trusted System Configuration

As distributed, the system is designed to meet the requirements for the C2 class of trust, which describes the level of protection given to prevent unauthorized access to a system and its data.

If you not wish to follow the C2 guidelines, you can choose

here to configure the system to behave in a manner consistent with standard System V security. The Relaxed defaults include less stringent password aging, fewer login restrictions, and fewer limits on the kind of programs that users can run. Note that a system that has been relaxed cannot be reliably restored to the C2 level. Refer to "Maintaining System Security" in your System Administrator's Guide for more information.

Choose one of the following:

- 1) Trusted C2 Defaults
- 2) Relaxed Defaults

Enter your choice or q to quit:

除非有特别的安全要求，应输入“2”的并回车。

(33) 如果计算机上没有安装显示适配器（系统假设是以串行方式使用终端），会看到如下的提示信息：

Please enter your terminal type as listed on the terminals (M) manual page in your User's Reference. For example, if you are using a VT100 terminal, the proper terminal type is vt100.

Enter terminal type:

如果不知道终端类型，可去查阅手册。系统支持数百种类型的终端，许多终端都有“仿真”的工作方式，可模拟常用的终端类型。

(34) 接着，显示下列信息，概述当前磁盘的使用情况：

You have now installed the SCO System V minimum Run Time System. The Run Time System will support most application programs, so you may choose to stop installation now and preserve the maximum available disk space for user files.

The remainder of the SCO System V product ( including operating and Development System sets) are installable in small packages. You may either install each set entirely or selectively choose which packages of the set to install.

(35) 还会看到一个菜单，提供选择停止安装或继续安装：

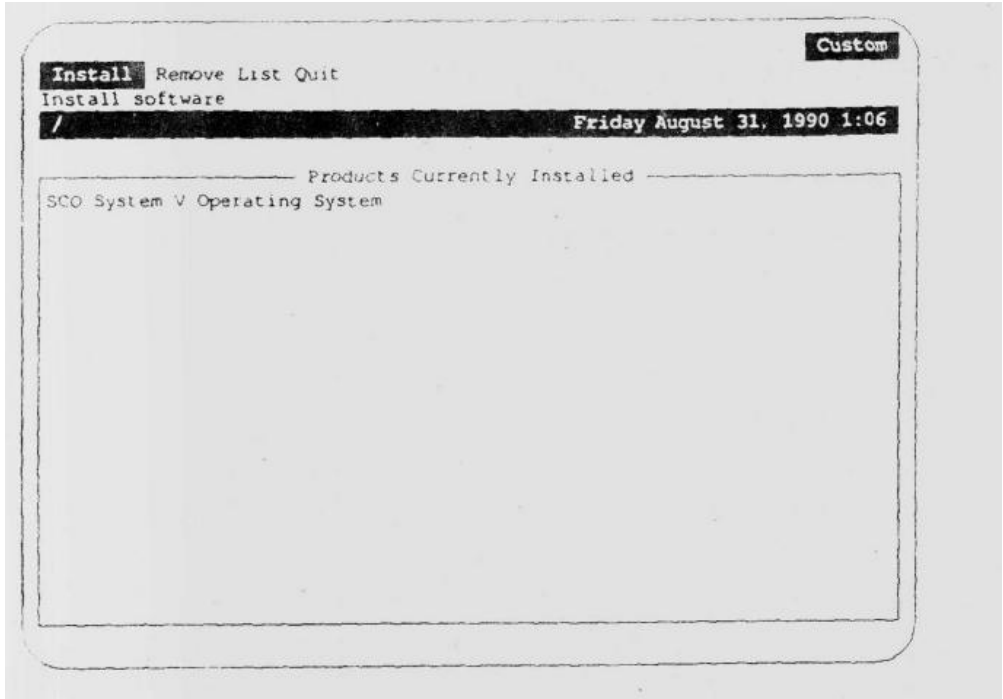
1. Install additional software (Extended Utilities or applications)
2. Continue installation

到目前为止已经安装好一个最小的，或称“实时”系统。如果安装 UNIX 操作系统只是为了作为某一应用程序的平台，就不必安装其他软件了。但是，如果需要广泛地管理系统，拥有用户帐号，使用系统管理外壳，就应该完成更多软件的安装工作。如果还想安装操作系统软件包或应用程序，应选择选项 1 并回车。

(36) 到此为止只是安装了操作系统中必须有的部分。还可以用安装程序 Custom

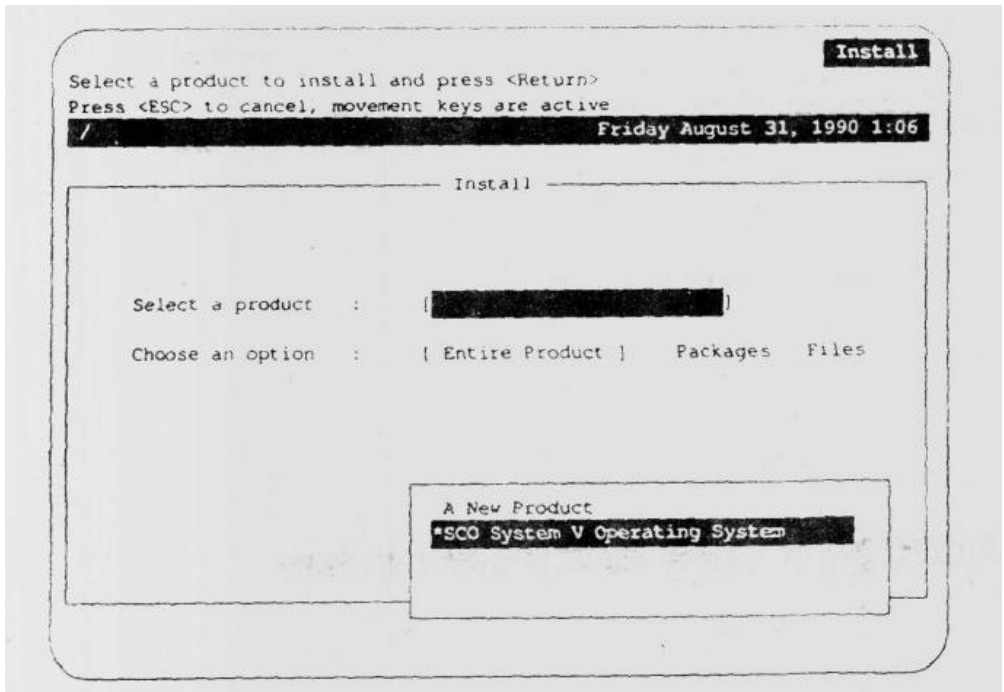
(ADM) 来自动安装 UNIX 软件包及其他软件包和应用程序。

(37) Custom 的主菜单形式如下：



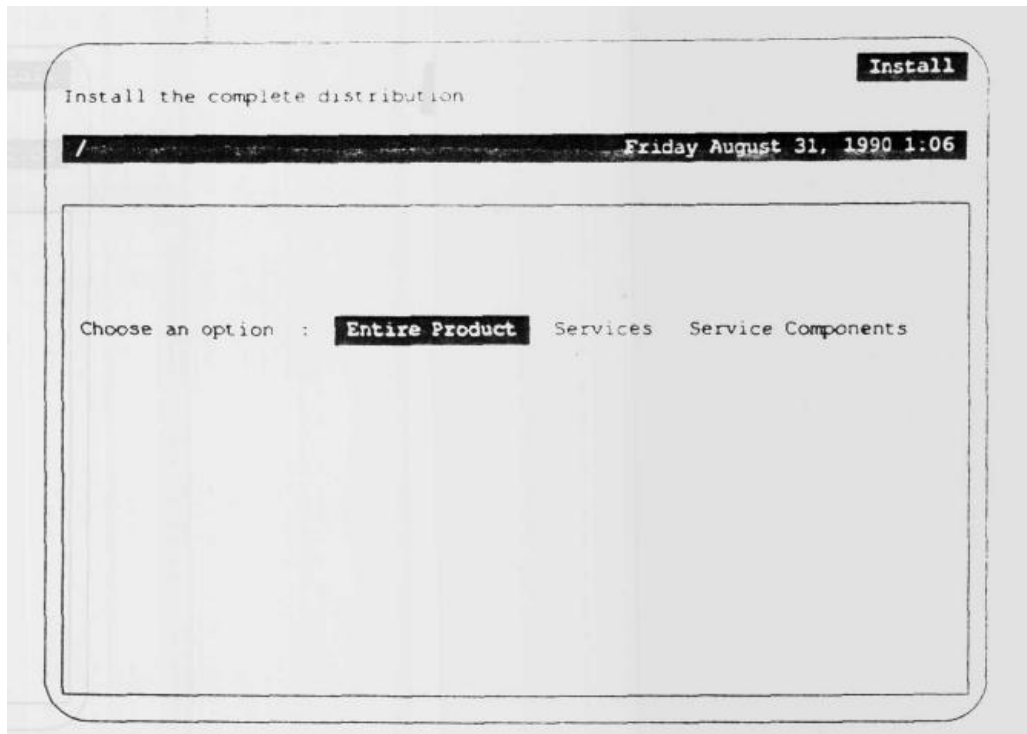
菜单已经选到了“安装”一项，要安装软件，按回车键可继续。

(38) 会看到菜单进入下一步：



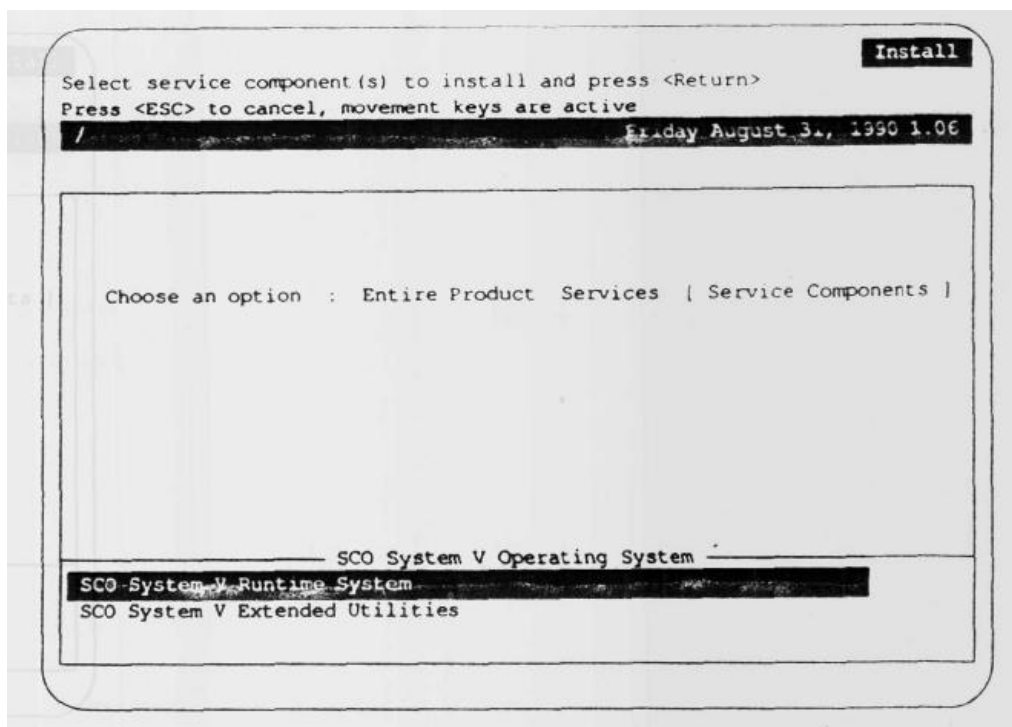
如图所示使用上下箭头将 SCO system V Operating system 一项加亮，按回车键

(39) 过一会儿之后显示：



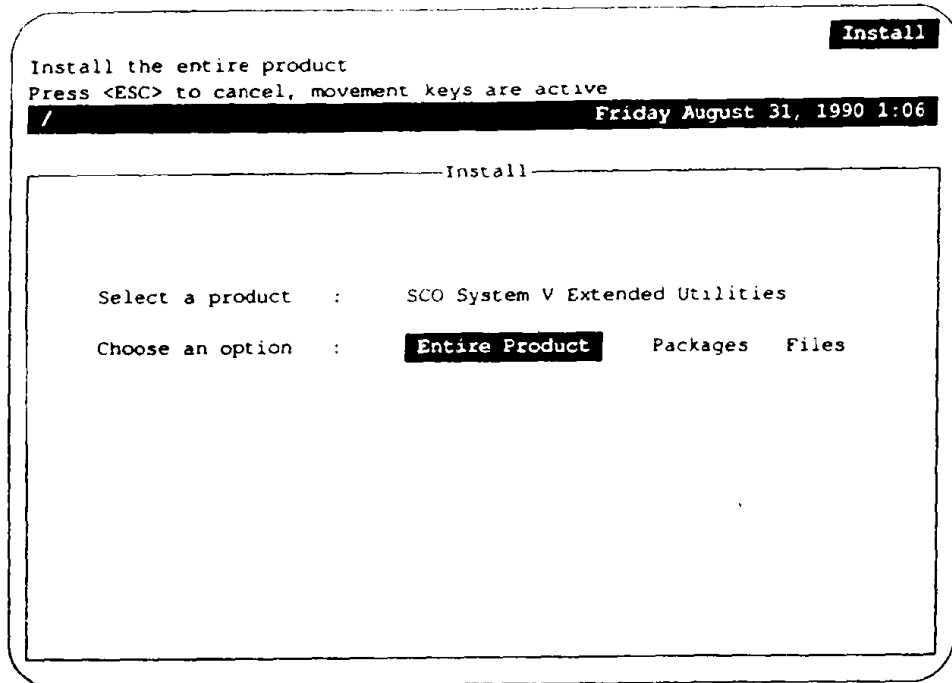
如果希望安装整个系统，按回车键，会要求插入从 X1 开始的盘，接着应转第 44 步执行。如果只打算安装操作系统的一部分，则使用左右箭头将 Service Components 一项加亮，然后回车。

(40) 会见到如下菜单：

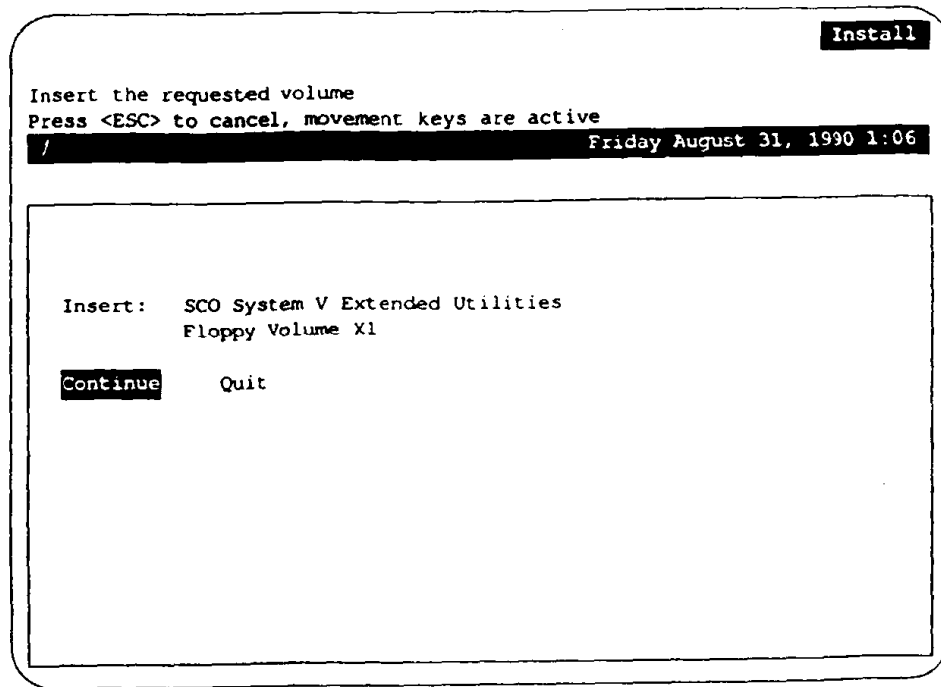


使用上下选择 SCO System V Extended utilities 一项并且回车。

(41) 这一步菜单如下：



Entire Product 一项被加亮，用右箭头键加亮 Packages 一项并回车。  
(42) 过一会儿见到如下菜单：



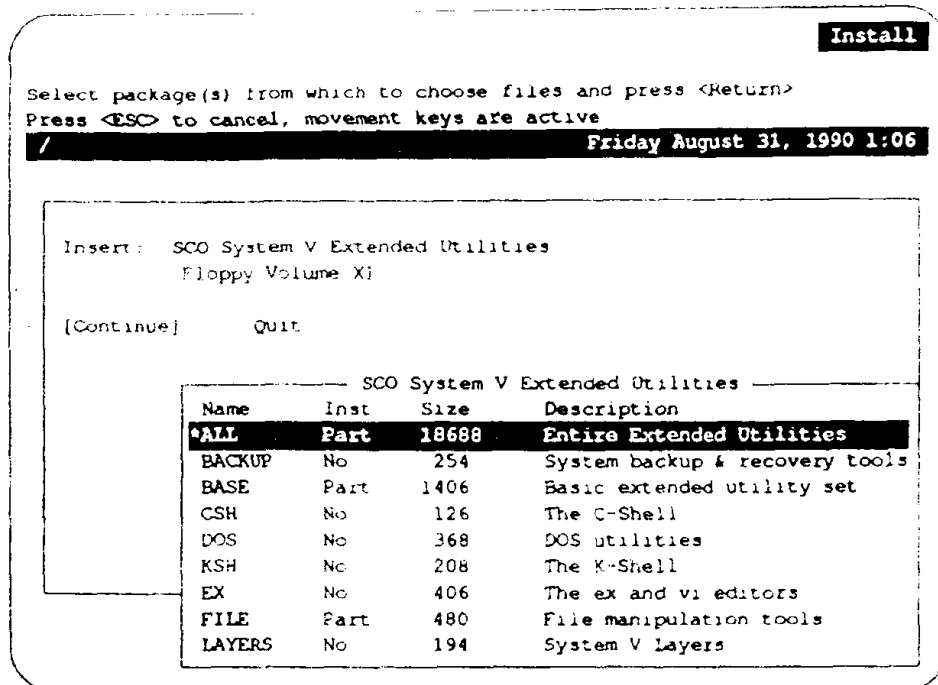
按照指令插入软盘。

(43) 下面装入数据文件，见到下表：

这个菜单是操作系统软件包的目录。

可以用箭头在这个列表上上下下移动。如果希望选择安装几个软件包，则用空格键为每一个选择上的软件包加上星号（\*）。按下回车键时，标上\*号的每一个软件包都要安装。值得注意的是，如果要安装列表中的所有软件包，则选择 All 一项即可。





在第三章中已经列出了软件包的目录，到目前为止已经安装在你的系统上的软件包只是 RTS，即 RUN TIME SYSTEM，这是一个能工作的操作系统所配置的最必需的软件。其他的软件包也可以增加到系统中以满足系统要求。如果系统只是一个提供给电子表格、字处理程序、数据库这样的应用程序的工作平台，就应该安装以下的软件包：

**SYSADM:** 用于系统管理的、用户使用很方便的接口软件。

**BACKUP:** 执行备份任务的实用程序。

**LPR:** 形式打印机池。

**MAIL:** 给其他用户发送消息的邮件程序。

另外，对使用传统的 UNIX 环境、使用 ex(c)或 Vi(c)文件编辑软件、不同的文件实用程序以及 c-shell 程序的系统来说，像 BASE，EX 和 Csh 等软件包也是非常有用的。

这里，向你推荐安装 EX，FILE，BASE 和 LINK 软件包，因为这些软件包含有对维护维护系统非常有用的实用程序。特别推荐 EX 软件包，因为它包含有 vi 全屏幕编辑程序，这是本系统中典范的编辑程序，在许多地方都用得到。有些软件(比如网络软件)的驱动程序需要连接到操作系统核心中才能应用，所以应该安装 Link kit(Link)软件包。

如果硬盘容量非常大，空间不成问题，可以选择 ALL 项，将 UNIX 操作系统的每一个软件包都安装到系统中。

(44) 如果安装 MAIL 或 ALL，(或者选择了 Entire Product) 有时会出现如下的提示信息：

Executing SCO System V Operating System Init Script

Your system name is set to scosysv. Do you wish

the mail system to use a different name? (y / n)

如果回答“y”，系统会提示输入新名字。

系统用以下信息表示确认了你的选择：

The mail system will use the name UUCP as the local machine name.

If you want to change this, please edit the file /usr/mmdf/mmdftailor, and any files in the directory /usr/mmdf/table that contain the old name, and then run /usr/mmdf/table/dbmbuile.

关于电子邮件的内容将以专门章节讨论。

(45) 可以通过选择 custom 安装菜单中“A NEW PRODUCT”一项来安装应用程序。系统会提示指定不同的安装内容（包括 UNIX 操作系统的某一部分）。这和安装 UNIX 软件包的方法是一样的。

(46) 在完成 UNIX 软件包及某些应用程序的安装之后，选择 Quit 项退出 Custom 程序，在要求确认你的选择时输入“y”。显示如下菜单：

The system software is now installed and the system configured for a single user (root).

At this point, You may use the System Administration Shell to add users, configure the system, or change system defaults.

1. Run System Administration Shell
2. Continue installation

选择第 2 项并按回车键。

(47) 系统显示警告信息：

NOTE: Be sure to use the Quit option to exit the System Administration Shell, so that the remainder of the installation will be completed.  
Press <Return> to continue

按回车键。

(48) sysadmsh 主菜单显示如下：

```

SysAdmSh
System Backups Accounts Printers Media Jobs Dirs/Files Filesystems Quit
Administer and configure system resources and report system status
/ Friday August 31, 1990 1:06

```

如何使用 sysadmsh (system Administration shell) 接口的细节，将另辟一章加以讨论。现在，参照下面的功能键表格来进行选择：

表 2.1 主菜单功能键

功能键	作用
箭头键或用作右箭头的空格键	在菜单上移动到选择项
选择项的第一个字母或移动加亮条到选项，然后回车；<ESC>; <F1>	选择菜单上某选项，回到上一级菜单，取得帮助信息

可以试着在菜单上用箭头键或空格键将加亮条从一个选项移到另一个选项，熟悉这一过程。每次将加亮条移到一个新的选项，关于该选项的描述就会出现在信息描述行上。

(49) 如果在安装的早期硬盘上建立了另外的文件系统（例如 /u），则在操作之前应按照附录下“If yw Created Additional Filesystems”中的命令执行，附录 F 使得附加文件系统准备好，可供使用。

(50) 如果需要，现在就可以在系统上增加用户帐号。在 sysadmsh 中应做以下选择：Accounts → User → Create

每个使用系统的人都应该有自己的登录帐号和口令。系统的安全性在很大程度上决定于对用户帐号使用所加的限制。详细的内容可参考本书中“管理用户帐号”部分。还可以改变用来建立和管理帐号的缺省值。

(51) 工作完成之后，可以通过选择最高级菜单中 Quit 一项来退出 sysadmsh。

(52) 首先系统提示你提供根口令：

Please assign a password for the super-user account. "root".

Setting password for user: root

Last successful password change for root: NEVER

Last unsuccessful password change for root: NEVER

Choose password

You can choose whether you pick your own password,  
or have the system create one for you.

1. Pick your own password

2. Pronounceable password will be generated for you

Enter choice (default is 1):

输入“1”来选择自己的口令。

(53) 然后，系统提示你输入口令两次。

Please enter new password (at least 5 characters):

Please choose a password which contains a mixture of lower-  
and upper-case letters, digits (0-9), and non-alphanumeric  
characters (e.g., !, #, @, ;, %, or /.)

Please do NOT choose a password that is an English word,  
or which is the name of a person, person, place, or thing, or which  
contains the string "SCO", "XENIX", or "UNIX" (in either case).

Re-enter password:

新的口令可以是字母，数字和标点符号的组合，但至少应该有 5 个字符长。输入新的口令并按回车键。在输入口令时系统并不显示输入的内容（主要是出于安全的考虑），所以输入时必须小心仔细。

现在设置好超级用户口令。以后每次试图作为超级用户访问系统就会要求输入这个口令。该口令使得系统免遭未经授权操作的破坏，因为作为超级用户登录注册时很容易犯一些对系统产生破坏的错误。一般只是在做系统安装及系统维护工作时才作为超级用户注册。对超级用户的完整的描述见本书第六章“启动和停止系统”。

千万不要将超级用户口令遗忘。要恢复遗忘的超级用户口令，只能重新安装 UNIX 系统，而这样做的工作量是很大的。

(54) 系统显示如下信息:

Store the password in a safe place. You will need it to log in.

The password for "root" can be changed by logging in as "root"  
and invoking the command 'passwd.'

Installation and configuration of the System V Operating System is now complete.

AFTER you see the message \* \* Safe to Power Off \* \*

or --

\* \* Hit Any Key to Reboot \* \*

reboot the system by opening the floppy door and pressing any key.

The screen will clear and you will see the boot message:

Boot

Press RETURN to reboot the newly installed system.

(55) 按指令执行, 你会看到:

\* \* Safe to Power Off \* \*

or --

\* \* Hit Any Key to Reboot \* \*

按回车键, 系统即关闭。记住移走软盘驱动器中的软盘片。

(56) 按任意键重新启动系统, 并在启动提示符出现时按回车键:

Boot

(57) 以下信息显示出来:

hd (4) unix

屏幕清除, 可以看到部分自检诊断操作。

(58) 看到如下信息:

INIT: SINGLE USER MODE

Type CONTROL d to continue with normal startp.

(or give the root password for system maintenance):

按 ctrl-d 继续。

(59) 系统要求设置系统时间。如果你时钟电池是正常的, 就不必设置时间。按回车键继续。

(60) 在系统启动时, 每个系统设备的信息逐一显示。系统提示你检查 tcb (Trusted Computing Base)时, 输入“n”并回车即可。

当启动结束后, 系统显示一个注册提示符, 准备让用户登录。

(61) 如果打算使用电子邮件, 则需要为 MMDF 帐号建立口令。作为超级用户注册并输入以下命令:

passwd mmdf

系统就会提示如同建立超级用户口令一样建立一个口令, 该口令用于管理邮政事务系统。

(62) 现在应该为系统建立紧急启动软盘组。在系统遭到破坏, 已不能用硬盘启动

时，这一组盘片会帮助迅速恢复根文件系统。参照附录 C，“Creating an Emergency Boot Floppyset”的内容。这么做在系统故障时使得可以不必从原盘重新安装系统，从而节省许多时间。注意，应时常备份文件系统，以便在故障发生时可以很方便地恢复。

(63) 如果希望增加新的硬盘，请参照本书中第三部分第五章的内容执行。

## 附录 A 安装中可能出现的问题

在安装过程中有时会出现错误。一般来说软件或安装过程本身没有什么问题。偶尔硬件会有一些小问题，但大多数情况都不难解决，例如电缆连接不对等。本附录讲述在出现这些经常发生的故障时该怎么办。

在安装过程中，不要认为能预见到将要发生的一切，即使以前做过安装工作也不一定会一帆风顺。所以请仔细阅读本附录。

下面列出最常出现的错误，给出如何避免错误以及在错误出现时如何解决的方法，在安装软件遇到问题时，也许能有帮助。

- 有些硬件设备（如磁盘驱动器）似乎不能工作，但在另外的操作系统下又能工作得很好：

某些硬件设备不能在 UNIX 系统下工作。参阅本书第三部分第七章的内容，看一看出故障的设备是否与 UNIX 系统兼容。

- UNIX 系统在 DOS 之后安装，现在二者都不能工作：

也许只能备份下 DOS 文件，重新安装 DOS，再重新安装 UNIX 系统。

- 读软盘错

确认在驱动器中的软盘是正确的一张，而且正确插入驱动器中。

确认软盘插入之后驱动器小门已关好。

如果确信正确的盘已正确地插到软盘驱动器中而读盘错误仍然存在，应在一个坚硬的表面如桌面上轻轻顿顿软盘的边缘。这一工作应特别仔细地进行，因为一不小心即会损坏磁盘。

- 系统不能从启动软盘启动

确信已插入启动软盘。如果插入的是另外一张软盘，将不会看到出错信息，而系统仍然不能启动。

启动软盘送到手中时贴有写保护标签，所以在安装过程中不致由于误操作将盘损坏。如果损坏了启动盘或抹去了盘的内容而且又没有留下备份，就只好再索取一张新的启动盘。

- 系统不能从硬盘启动

可能会看到如下的或者另外一种错误信息：

PANIC: init

有时这是由于没有运行坏道检测，而在安装时又把引导块放到坏道上。不管是什么原因，都只能重新安装 UNIX 系统。如果操作系统仍然不能启动，则应运行随计算机和硬件一道送来的硬件诊断程序。

## 附录 B 安装和移去附加软件

custom (ADM) 实用程序用于在你的系统上安装和移去 UNIX 系统软件和 UNIX 应用程序。custom 共有三级操作：Entire Product, Service 和 Service Component。这三个级别的功能如下：

Entire Product: 安装所有设备软件

Service: 安装组成某功能块的一组设备

Service Component: 单个设备的软件包和单个文件的安装。比如，UNIX 系统扩展程序包含若干个软件包，如 MAIL 和 LPR (打印实用程序)

例如，Entire Product 可以包含若干个 Service，同样，一个 Service Component 包含许多软件包，一个软件包则是多个单独文件的集合。文件以软件包的形式提取或删除。

UNIX 系统包含多个软件包，本附录解释如何做到以下几条：

- 安装附加的产品
- 安装 UNIX 系统的软件包
- 从系统中移去软件

如果以前只选择安装了 UNIX 系统的部分软件包，现在希望安装其余软件包，请查阅本附录后面“安装 UNIX 系统软件包”。

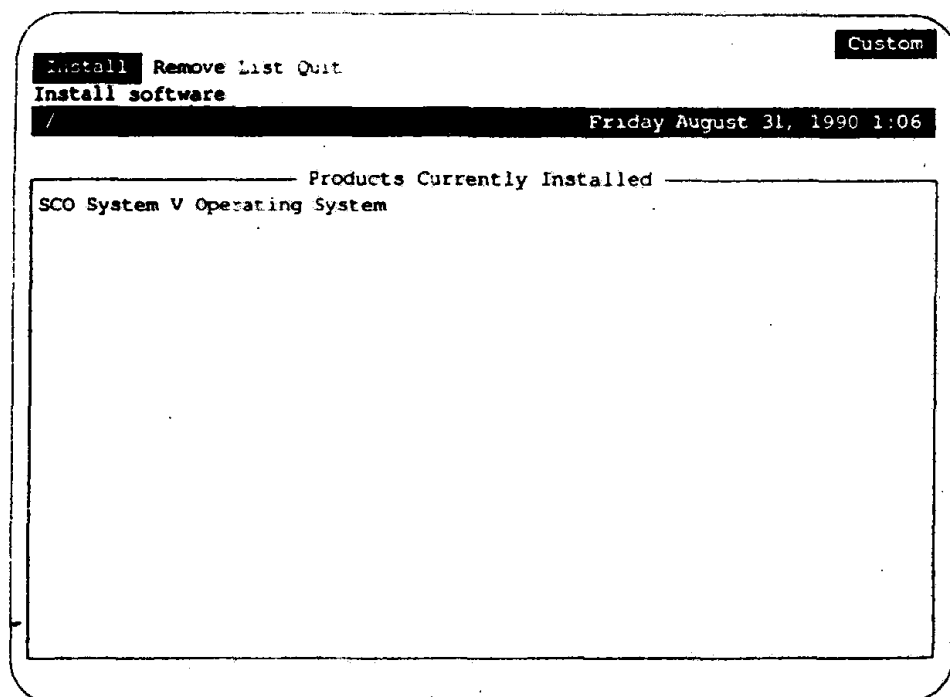
### B.1 安装附加产品

本节描述如何安装能够用 Custom 进行安装的附加产品或应用程序。

1. 为了安装其他应用程序，应做如下的 stsadmsh 选择：

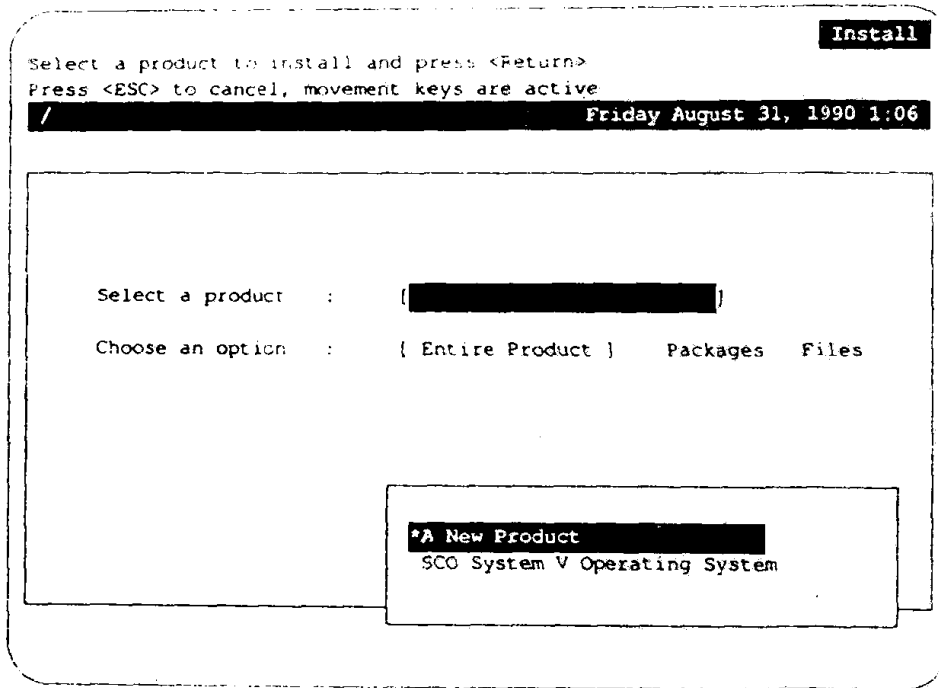
System >Software

2. custom 主菜单显示如下：



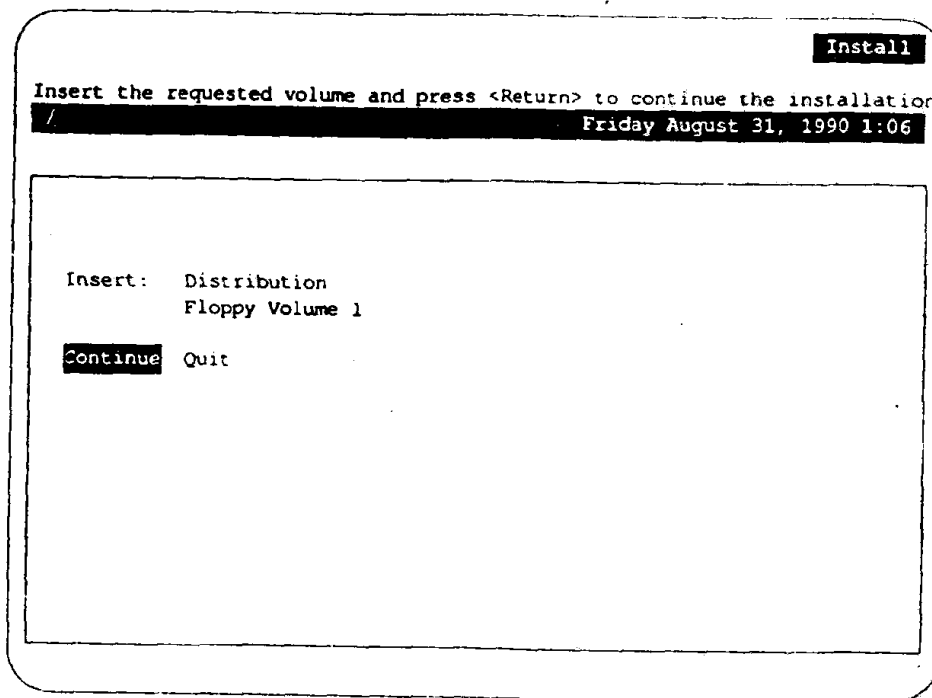
- 菜单已设置到了“安装”一项，按回车键继续。

3. 系统显示下一层菜单:



加亮条指向 A New Product 一项。按回车键，Entire Product 一项即被加亮，再次按回车。

4. 屏幕显示:



按照命令插入所安装产品的第一张软盘并按回车键。这样做之后不久，系统提示插入第 1 号盘，这一次用的是实际软件的名称。按回车键并根据提示插入要求的软盘。

5. 产品安装完毕之后回到 Custom 菜单。如果安装工作已经完成，则选主菜单中 Quit 一项并确认选择以退出 Custom。

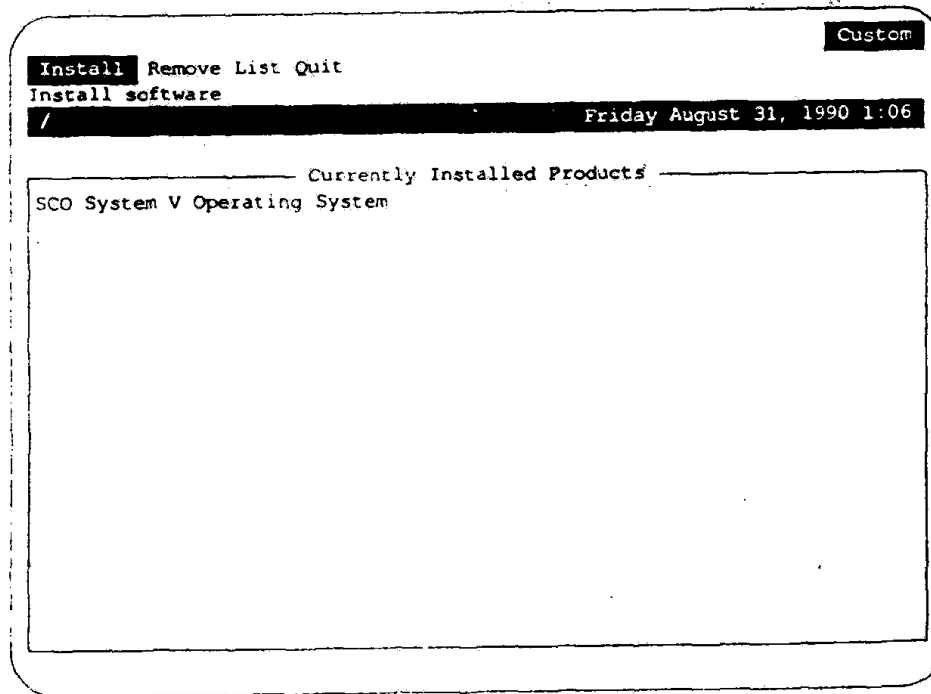
## B.2 安装 UNIX 系统软件包

本节介绍如何安装 UNIX 系统软件包。

1. 为执行安装，在 sysadmsh 中应做如下选择：

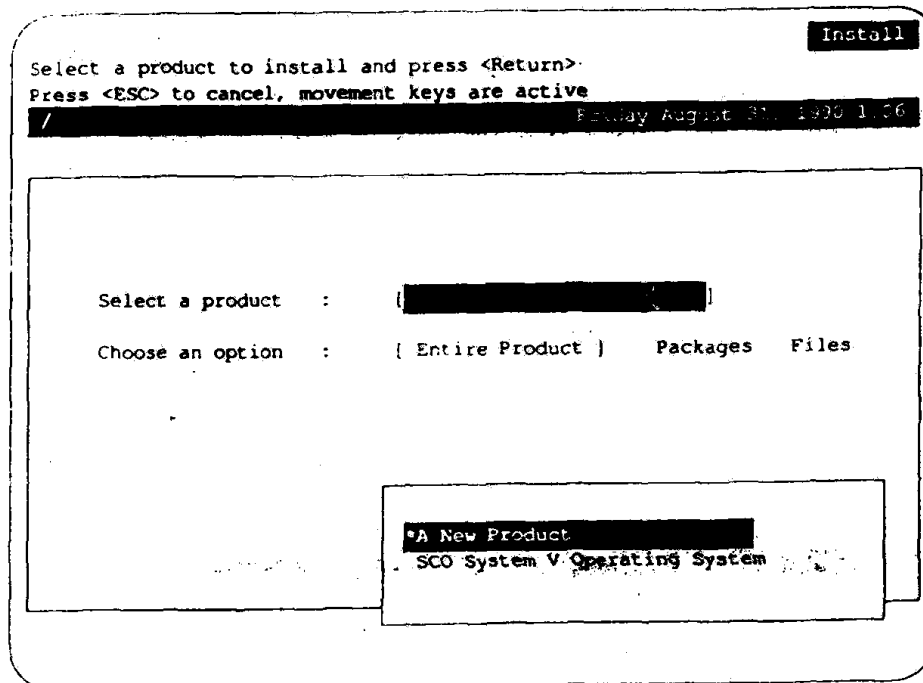
System → Software

2. 主 Custom 菜单显示如下：



菜单已经设置到 Install 一项，即可以安装软件，按回车键继续。

3. 下一屏菜单为：

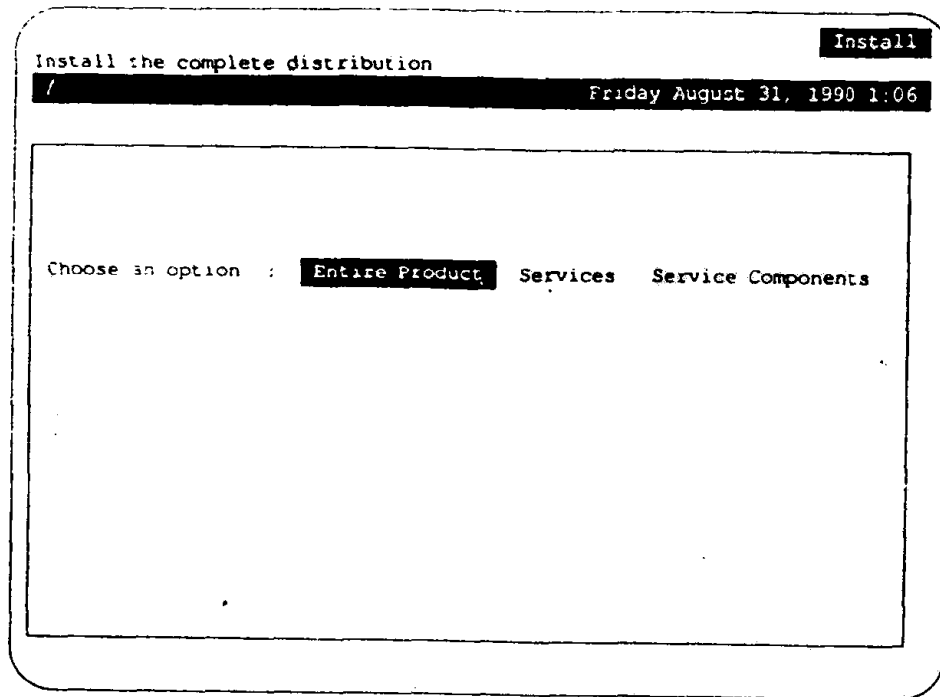


用下箭头键将 SCO System V Operating System 一项加亮并按回车键，接着 Entire Product 一项被加亮。如果希望安装整个操作系统，应该再次按回车键。如果希望选择安



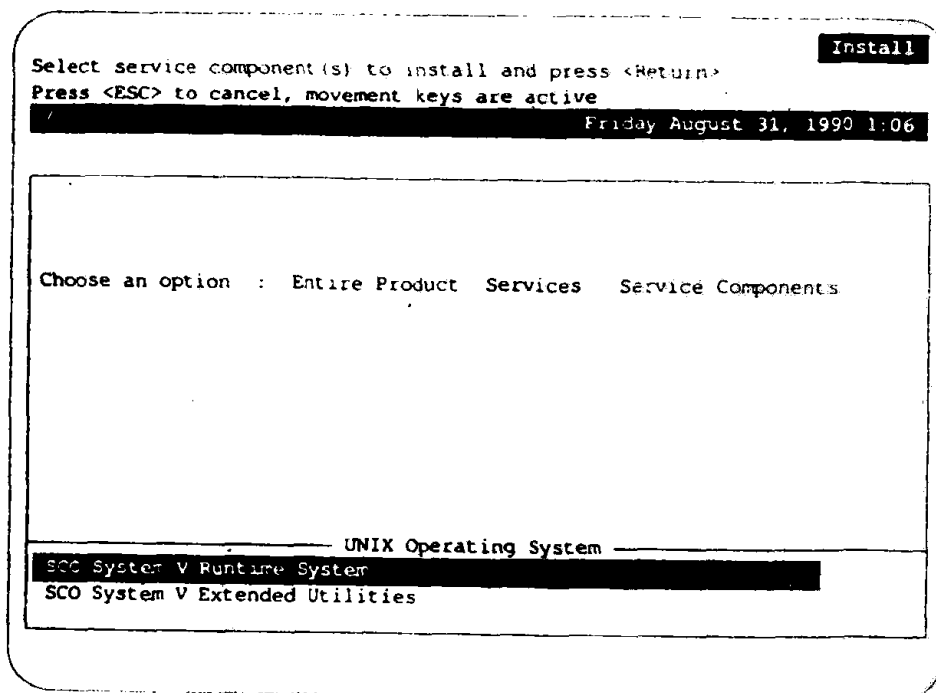
装软件包，应用左箭头键将 Packages 一项加亮并按回车键。

4. 这一会儿之后将看到：



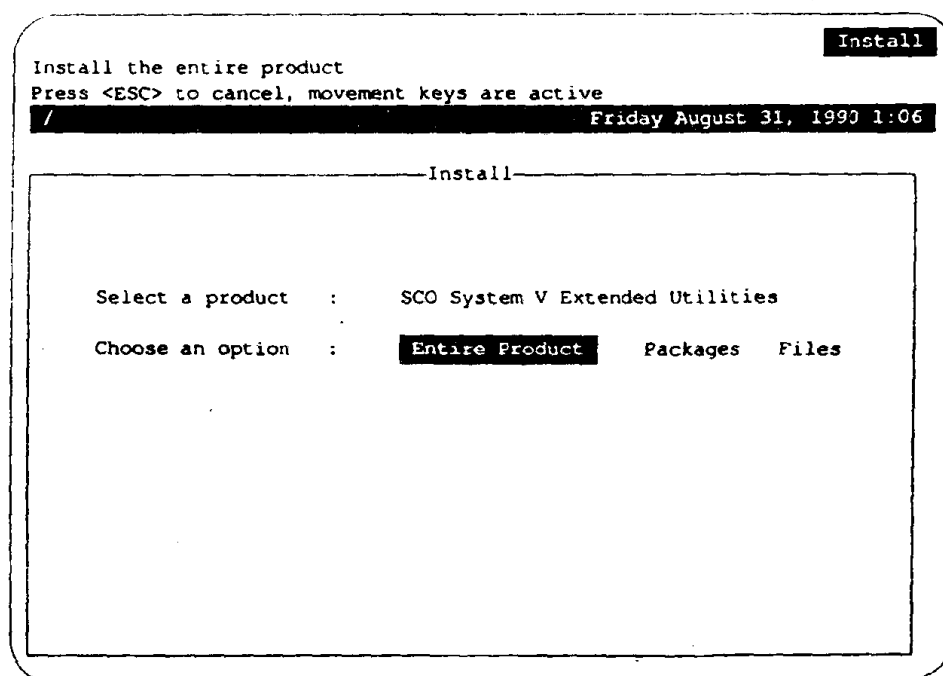
用左箭头键将 Service Components 一项加亮并按回车键。

5. 屏幕上出现：



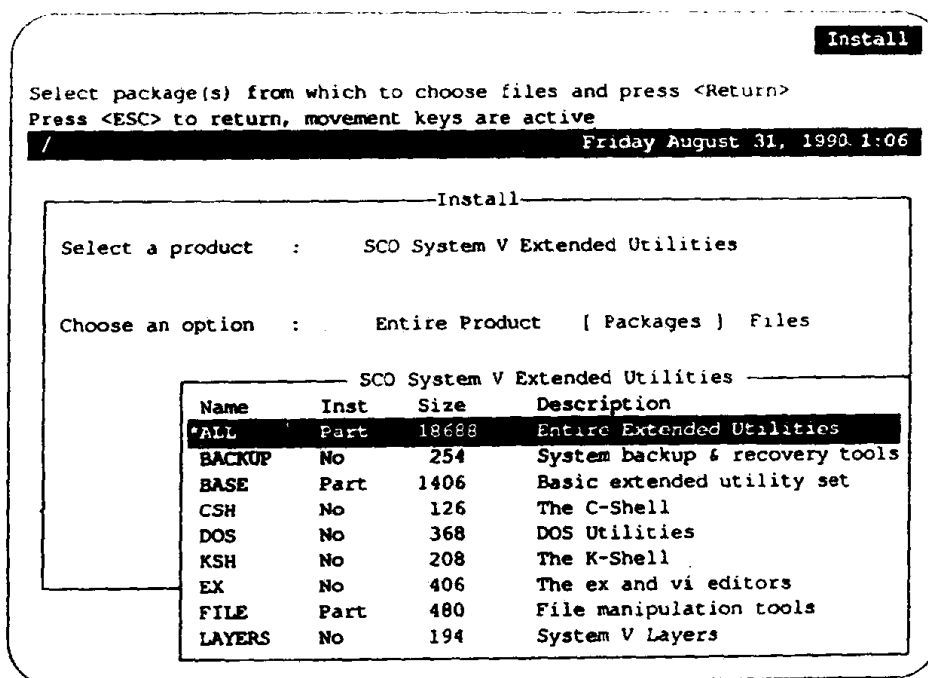
用下箭头键选择 SCO System V Extended Utilities 一项并按回车键。

6. 会看到如下屏幕:



• Entire Product 一项被加亮。用左箭头键加亮 Packages 一项并按回车键。

7. 装载数据文件, 将看到类似下面的操作系统软件包列表:



可以用箭头键在列表中上下移动, 如果希望选择安装若干个软件包, 应用空格键为每一个软件包加上 (\*) 号。按回车键时, 每一个标上 (\*) 号的软件包都将被安装。选择 ALL 项, 则可以安装所有软件包。

8. 按提示信息的要求插入相应软盘。当安装完毕之后, 回到 Custom 菜单。

9. 这时可以选择主菜单中 Quit 一项退出 Custom。

### B.3 从系统中移去软件

大体上讲，从系统中移去软件，不管是软件产品、设施、设备或软件包，其过程与安装也差不多。只需要选择 custom 菜单的 Remove 一项，再像安装时那样选择要处理的软件即可。特殊驱动程序那样的产品将重新连接核心程序，以便移去这些驱动程序。另外，不同的产品也许要求不同的过程。

## 附录 C 建立紧急启动软盘组

在系统管理中紧急启动软盘组是一个非常重要的工具。在系统遭受灾难性的破坏后，如果重新启动系统而系统不作反应，这时即可用紧急启动软盘组来恢复系统。可以用这些软盘来恢复遭到破坏的根文件系统而不用重新安装系统。如果你的系统不止一个，则应该为每台计算机作一组紧急启动软盘。因为每台机器的紧急启动软盘组是独有的，在一个系统上所作的紧急启动软盘组在另一个系统上是不能用的；如果将一台机器的紧急启动软盘组用到另一台机器上，不但不能工作，而且还可能给机器带来更严重的损坏。

用来建立紧急启动软盘组的实用程序称为 mkdev fd，这是一个菜单驱动的程序，可产生三类磁盘：一种是一般文件系统，另两种用于紧急启动软盘组，一种只能启动，另一种只有根文件系统。建立紧急启动软盘组必须建立一张可启动软盘和一张根文件系统软盘。

### C.1 建立软盘

要建立所需软盘组应按照以下过程：

1. 作为超级用户注册并输入：

```
mkdev fd
```

sysadmsh 用户应选：

Filesystems→Floppy

2. 会看到以下内容：

Floppy Disk Filesystem Creation Program

Choices for type of floppy filesystem.

1. 48tpi, double sided, 9 sectors per track
2. 96 tpi, double sided, 15 sectors per track
3. 135tpi, double sided, 9 sectors per track
4. 135tpi, double sided, 18 sectors per track

Enter an option ro q to quit:

输入想要的磁盘类型号并回车。

3. 接着会看到：

Choices for contents of floppy filesystem.

1. Filesystem
2. Bootable only (96ds15 and 135ds18 only)
3. Root filesystem only (96ds15 and 135ds18 only)

Enter an option or enter q to quit:

首先建立可启动软盘，输入 2 并回车

4. 屏幕显示：

Insert a type floppy into drive 0.

Press Return to continue or enter q to quit:

按回车键。

5. 系统显示：

Would you like to format the floppy first? (y / n)

如果软盘已经格式化好，输入“n”，文件系统会立即建立。如果软盘尚未格式化，输入“y”，会看到：

formatting / dev / type

track 00 head 0

磁道数和磁头号随格式化的进行而变化。如果 / etc / default / format 文件包含 VERIFY = Y，则格式化结束之后需要进行校验。

6. 屏幕显示如下信息：

Successfully created filesystem.

Copying files to / dev / type ...

建立了文件系统，将有关文件从根文件系统中拷贝到软盘上即建立起可启动磁盘。mkdev 还用 fsck (ADM) 检查文件系统，系统显示与启动时检查文件系统时类似的信息。

7. 当磁盘准备好时显示如下信息：

type floppy created and checked successfully

8. 然后回到主菜单。现在应该建立根文件系统软盘。输入“3”并按回车。

9. 你会看到如下信息：

Insert a type floppy into drive 0.

Press Return to continue or enter q to quit:

按回车键。

10. 屏幕显示提示信息：

would you like to format the floppy first? (y / n)

如果软盘已经格式化，输入“n”，文件系统立即建立。如果软盘尚未格式化，输入“y”你会看到与上面描述一样的格式化信息。

11. 下列信息出现在屏幕上：

copying file to / dev / type root filesystem ...

copying special files to / dev / type root filesystem

12. 与建立可启动软盘一样，mkdev 也要用 fsck (ADM) 检查文件系统；显示信息很相似。

13. 磁盘准备好之后显示下列信息：

type floppy created and checked successfully

好好保存这些软盘。一旦系统遭到破坏，不能启动，就用得着这些盘。

## 附录 D 硬盘配置

本附录描述如何手工初始化硬盘，包括配置非标准硬盘（不存在 ROM BIOS 入口的硬盘）。具体讲，包括确定硬盘上操作系统避免使用的坏道的分布情况，以及如何将硬盘划分成分区（由不同操作系统占据的独立区域）和文件系统（由同一操作系统下目录所占据的独立区域）。

共有三种类型的硬盘和控制器：ST506（标准硬盘），ESDI 和 SCSI。设置 ESDI 和 SCSI 硬盘和控制器的步骤与适用于标准硬盘的步骤不尽相同。这些步骤详述如下。

可配置硬盘的初始化过程如下：

1. 系统首先检测出所安装的硬盘和控制器类型并显示类似下例的信息：（本例为 SCSI 配置）

```
%disk - - - type=Sha=0 id=0 lun=0
%Sdsk - - - cyls=08 hds=64secs=32
```

2. 屏幕上显示：

During installation you may choose to overwrite all  
or part of the present contents of your hard disk.

Do you wish to continue? (y / n)

如果错误地重新开始了安装进程，此时就有机会停止这一进程。输入“y”继续。

3. 如果用的是 SCSI 控制器，就将看到下列信息：

The hard disk installation program will now invoke / etc / fdisk.  
Entering 'g' at the following menu will exit / etc / fdisk.  
and the hard disk installation will continue.

If you wish to exit the entire installation at this menu,  
press the <DEL> key.

跳到第 9 步。

---

注意：SCSI 的安装跳过第 4~8 步。

---

4. 如果用的是 ST506（标准接口）控制器，将看到如下信息和提示：

The hard disk installation will now invoke / etc / dkinit.  
Entering 'q' at the following menu will exit / etc / dkinit,  
and the hard disk installation will continue.

If you wish to exit the entire installation at this menu,  
Press the <DEL> key.

Hard Disk Drive 0 Configuration

1. Display current disk parameters
2. Modify current disk parameters

### 3. Select default disk parameters

Enter an option or 'q' to quit:

5. 如果控制器是 OMTI, 会看到:

Caution: Consult the ESDI installation Release Notes if you wish to modify the disk parameters the / etc / default will display.

如果输入“q”, 会见到如下信息:

The hard disk installation program will now invoke two disk preparation utilities: fdisk and badtrk.

Selecting 'q' at the main menu for each utility will exit that utility and continue with the hard disk installation.

转第 9 步。

6. dkinit 菜单用于为非标准硬盘设置参数, 如果硬盘是标准的, 即为计算机硬件或特殊的母板 ROM 所支持, 则输入“3”并回车以继续安装过程。另外, 如果硬盘是 SCSI 的, 还必须输入“q”。参数已经设置好。

此时输入“q”为硬盘选定缺省参数。除非知道你的硬盘是非标准的, 否则假设它是标准的, 输入 q 继续安装工作。转第 9 步。

如果磁盘是非标准的, 必须输入信息以覆盖 ROM 磁盘配置信息。

如果输入“1”或“2”, 屏幕显示:

Disk parameters	Values
1. Cylinders	value
2. Heads	value
3. Write Reduce	value
4. Write Precomp	value
5. Ecc	value
6. Control	value
7. Landing Zone	value
8. Sectors / track	value

当出现该显示时, “value”被该变量的缺省值替代。

---

注意: “Cylinders”值是指整个硬盘上的柱面数目, 不能与分配给 (或打算分配给) 某一指定分区的柱面数目混为一谈。

---

如果输入“1”, 再次显示第一个菜单。如果输入“2”, 提示:

Enter a parameter to modify or 'q' to return to the main menu

输入从 1 到 8 之间的某一数字以改变磁盘参数, 或输入“q”回到上一级菜单。

7. 将看到:

Enter the new value or <RETURN> to use the existing value

如果要修改这个值, 现在就可输入新值或按回车键使用原来的值。

8. 在完成磁盘参数的修改后, 输入 q 退回主菜单。然后再一次输入 q, 将修改的参

数存下来。输入 q 退出 dkinit, 经过改变的新值即会重写相应的参数。如果进行修改之后又想恢复缺省参数, 应在按“q”退出 dkinit 之前在第一个菜单上打入“3”。

9. 下一步安装程序将运行 fdisk(ADM)实用程序对硬盘进行分区。可以在同一个硬盘上支持 DOS (如果 DOS 已经安装到硬盘上), 或者将整个硬盘都分配给 UNIX 系统使用。

过一会儿之后, 在屏幕上出现 fdisk 菜单, 将看到如下的选项表:

1. Display Partition Table
2. Use Entire Disk for UNIX
3. Use Rest of Disk for UNIX
4. Create UNIX Partition
5. Activate Partition
6. Delete Partition

Enter your choice or 'q' to quit:

选择选项“1”并按回车键。

如果从未在硬盘上安装操作系统, 会看到类似下面的表格:

Current Hard Disk Drive: /dev/rhd00

Partition	Status	Type	Start	End	Size

Total disk size: 1220 tracks (5 reserved for masterboot and diagnostics)

Press <RETURN> to continue

如果以前在硬盘上安装过操作系统, 这张 fdisk 表格就是填入了值的。DOS 通常作为 4 号分区显示。

10. 按回车键回到 fdisk 主菜单。如果希望让 UNIX 分区占据整个硬盘, 选择选项“2”。在选择完成之后, 输入“q”以退出 fdisk 菜单。如果以前在系统上安装过任何其他系统, 还将看到如下警告信息:

Warning! All data on your disk will be lost!

Do you wish to continue? (y / n)

必须在确认让 UNIX 占据整个硬盘时才输入“y”并回车。这保证了 fdisk 对整个硬盘进行分区。

---

注意: 如果选择了选项“3”, 即把硬盘的剩余部分分给 UNIX 系统, 必须接着选择选项“5”以激活(activate) UNIX 分区。否则, 被激活的将是第一个分区。

---

大多数计算机硬盘的最后一个柱面上都写有诊断程序, 这意味着不应该把最后一个柱面分配给任何一个分区。在从 fdisk 菜单选择选项 2 时, 最后一个柱面没有分配, 在选择选项 4 时, 也不应该把硬盘的最后一个柱面分配给 UNIX 分区。

11. 按回车键, 将显示 fdisk 菜单。现在硬盘上分区已经设置好。输入“q”并回车, 继续进行安装过程。

如果硬盘控制器是 ST506 或 ESDI, 则从第 12 步开始执行。

如果用的是 SCSI 控制器或 SMS-OMTI — ESDZ 控制器, 转第 21 步执行。

---

注意: SCSI 和 SMS OMTI 安装过程不执行第 12 步至 20 步的 badtrk 程序, SMS OMTI 是唯一不执行 badtrk 的 ESDI 控制器。

---

12. 现在看到 badtrk (ADM) 程序的菜单。通过 badtrk 程序, 可以检测出硬盘上有缺陷的磁道。该程序将有缺陷的磁道映射到硬盘别的地方的好的磁道上, 并产生一个报告硬盘上坏道情况的坏道表。

badtrk 的主菜单如下:

1. Print Current Bad Track Table
2. Scan Disk (You may choose Read-Only or Destructive later)
3. Add Entries to Current Bad Track Table by Cylinder / Head Number
4. Add Entries to Current Bad Track Table by Sector Number
5. Delete Entries Individually from Current Bad Track Table
6. Delete All Entries from Bad Track Table

Enter your choice or 'q' to quit:

输入“2”并回车。

13. 屏幕显示下列子菜单:

1. Scan entire UNIX partition
2. Scan a specified range of tracks
3. Scan a specified filesystem

Enter an option or 'q' to quit:

选择选项 1。

14. 选择想扫描的区域之后, 提示:

1. Quick scan (approximately 7 megabytes / min)
2. Thorough scan (approximately 1 megabyte / min)

Enter an option or 'q' to quit:

选择选项 2。

15. 系统提示:

Do you want this to be a destructive scan? (y / n)

输入“y”, 出现警告信息:

This will destroy the present contents of the region you are scanning.

Do you wish to continue? (y / n)

输入“y”并回车, 看到:

Scanning in progress, press 'q' to interrupt at any time.

16. 在响应上面的提示之后, 该程序扫描硬盘的活动分区。硬盘越大, 扫描过程所需时间越长。所以一个很大的磁盘可能会需要一段时间才能完成扫描。



当 badtrk 扫描磁盘时，显示它检查的每个磁道号和已扫描硬盘的百分比。按“q”键可随时中断扫描（不用按回车键），系统将询问继续扫描还是回到主菜单。无论何时 badtrk 发现坏道，就以柱面号和磁头号的形式显示这个磁道的位置。坏道信息填入表格并显示在屏幕上。这里是一个坏道的例子：

```
WARNING:wd:on fixed disk ctlr=0 dev=0 / 47 block=31434 cmd = 00000020
status=00005180, sector=62899, cylinder / head =483 / 4
```

17. 扫描完成之后，菜单重新出现。选择选项“1”，看看扫描的结果。看到的坏道表类似：

Defective Tracks

	Cylinder	Head	Sector Number (s)
1.	190	3	12971-12987

Press <RETURN> to continue

按回车键返回主菜单。

---

注意：如果在 UNIX 分区的头几个磁道上出现了坏块，则回到 fdisk 实用程序（见前面的安装步骤）。用 fdisk 重新对硬盘进行分区，使 UNIX 分区不再含有坏道。必须自己确定需要排除多少磁道，不要把这些有缺陷的磁道分配给任何一个操作系统。退出 fdisk 之后，运行 badtrk 程序，扫描硬盘以检测其他错误。

这一过程不断继续，直到 badtrk 程序在 UNIX 系统的头几个磁道上没有检测出错误为止。

---

18. 如果盘出现了坏块图，则应该将其中的坏块填到坏道表中。

因为大多数坏块属于边界的或间歇性的，任一坏块图含有的坏块几乎必然比扫描时所发现的多。如果是这样，现在就应该把有坏块的磁道加到坏道表中。根据坏道格式，在任选项 3“以柱面 / 头形式在当前坏块表中增加坏道”和任选项 4“以扇区号形式在当前坏块表中增加坏道”中选择一项。在每一行上输入一个坏道。如果发生错误，就输入 q 并回车。当看到 badtrk 主菜单时，选择任选项 5 以删除一个磁道。

19. 如果磁盘没有坏块图，或者已完成了对坏道表的改变，输入 q 并回车。

20. 系统提示输入分配用来替换坏磁道的磁道数，至少应分配不少于建议用作替换空间的磁道的数目。输入该数目或者直接回车，直接选用显示的推荐数目：

Enter the number of bad tracks to allocate space for

(or press <RETURN> to use the recommended value of n):

如果直接回车，没有输入磁道数目，则 badtrk 按推荐的数目分配磁道作为替换空间，推荐数目的确定基于当前坏道表中坏道数目，加上为未来可能出现的坏道所留富余量。如果坏道数超过了留作坏道替换的磁道数，就只能重新安装系统了。

21. 现在已经根据硬盘配置对其做了相应的初始化处理，接着对硬盘文件系统布局进行规划。下一步，安装过程运行 divvy (ADM) 实用程序。该程序将硬盘分区的一部分空间分配给根和对换空间，还将一部分空间分配作恢复区，这在用 fsck (ADM) 程序进行自动引导时使用。divvy 程序首先提示指定分配给对换空间的空间。

提示信息中的实际数目随硬盘大小而变化。除非你有特别的需要，否则应选择显示于屏幕上的缺省值。（如果计划安装 Development System，则应输入至少比缺省值大 1000 块的对空区大小）。现在输入对换区大小或回车取缺省值：

There are n 1K blocks in the UNIX area.

Between x and y 1K blocks should be reserved for the swap area.

Please enter the swap space allocation, or press <RETURN>

to get the default allocation of z 1K blocks:

如果磁盘空间足够安装 / u 文件系统，会看到如下提示信息：

Do you want a separate / u file system? (y / n)

如果想建立一个独立的用户文件系统，输入“y”并回车，如果不希望建立，则回答“n”转而执行下一步。

接着看到：

Enter block allocation for the / u file system.

(min to max)

在实际的显示中，min 和 max 被能够分配给文件系统的块数目的最小值和最大值所替代。

## 22. 系统提示：

The layout of the file systems and swap area is now prepared.

Do you wish to make any manual adjustments to the sizes or names of the file systems or swap area before they are created on the hard disk? (y / n)

输入“n”并回车。本选项选择文件系统和对换区域大小的精确值，以满足特殊的需求。对大多数用户来讲并不需要如此精确的值，所以对此提示都回答“n”，让 divvy 根据你硬盘的大小选择使用缺省值。

23. 现在系统在硬盘上建立文件系统和对换区。这件工作要几分钟。屏幕显示：

Making file systems

Setting up hard disk root file system...

24. 这一过程完成之后，显示下面信息：

Initialization of the hard disk is now complete.

现在应该返回到第 4 章从第 19 步开始安装过程。

## 附录 E 设置北美之外的时区

本附录介绍如果你不在北美洲，该如何设置时区的方法。按照以下步骤：

1. 在时区设置过程的初始提示符出现之后，会看到如下信息：

What is the abbreviation of your standard time zone?

Enter 1-9 characters or enter q to quit:

输入所在标准时区的缩写符号。

## 2. 屏幕显示:

How many hours west of Greenwich Mean Time are you?

Enter hh [:mm:ss] (e.g. 10:30:00 or 10:30, use negative numbers for locations east of GMT) or enter q to quit:

输入一个数值。

## 3. 显示下面的提示信息:

Does summer time (daylight saving time) apply at your location (y / n)?

如果没有采用夏令时, 输入“n”并转第 7 步。

如果采用了夏令时, 输入“y”, 会看到:

What is the summer abbreviation of your time zone?

Enter 1-9 characters or enter q to quit:

输入所在时区的夏令时缩写。

## 4. 接着显示:

1. Week of the year (1-52).

2. Week of a specific month (eg. 1st week of April).

3. Day of the year, ie. julian date (1-365).

Select the method your time zone uses to convert from standard time to summer time (daylight saving time)

\ or enter q to quit:

选择一种方法并按提示执行。

## 5. 会看到:

At what time of day is the conversion made (use 24 hour clock)?

Enter hh [:mm:ss] or press RETURN for default value of 2 am

or enter q to quit:

输入一个数或按回车键。

## 6. 关于所在时区采用的夏令时的方法, 将看到下面提示信息:

How many hours does your timezone adjust for summer time (daylight saving time)?

Enter hh [:mm:ss] or press RETURN for the default value of 1 hour

or enter q to quit:

输入一个数或按回车键。

7. 时区变量 / etc / TZ 作相应变化。你的时区设置好了。没有必要为用户改变 profile, 除非从不同的时区访问系统, 并且希望取代当地标准。

现在可以回到第二章“安装过程”, 从第 30 步继续执行安装过程。

## 附录 F 建立附加文件系统

如果在安装过程中建立了附加文件系统 (比如 / u 文件系统), 必须按照本附录中的命令执行, 才能将附加文件系统准备好, 可供使用。

1. 作下面的 sysadmsh 选择:

Filesystem ▶Add

2. 会看到:

Filesystem Initialization Program

This program performs maintenance tasks required to add or delete an existing file system. would you like to:

1. Add a new file system to system.
2. Remove a file system.

Select an option or enter q to quit:

输入“1”并回车。

3. 然后会提示你输入设备名:

Enter a device name and press <Return> or q to quit:

输入设备在 /dev 中的全名。比如, 要增加一个名为 /u 的文件系统, 应输入 /dev / u。

4. 现在系统提示输入将要使用的安装点的名字:

Enter a directory name and press <Return> or q to quit:

这里的目录是指安装文件系统的地点。比如, 名为 /u 的文件系统被安装在目录 /u 处。

5. 显示如下信息:

Reserving slots in lost+found directory ...

When entering multiuser mode:

1. Always mount file system
2. Never mount file system
3. Prompt before mounting file system

Select an option:

如果希望文件系统在系统启动时自动进行安装, 应输入“1”, 如果希望在系统管理员提出要求时才安装文件系统, 则应选择“2”。如果选择了“3”, 那么在系统启动时就会询问是否要安装文件系统。

6. 然后系统询问是否允许用户装配文件系统:

Do you want to allow users to mount this file system? (y / n)

必须回答“y”, 只有这样系统备份程序才能根据需要装卸文件系统。

7. 当这一过程完成之后, 显示如下信息:

Updating system files ...

Filesystem has been successfully added.

8. 然后, 应该使用下面的 sysadmsh 选择安装 /u 文件系统:

Filesystem ▶Mount

9. 为保证系统正确识别出新的文件系统，按下惊叹号(!)，然后，在提示信息出现时，输入以下命令（命令被分号分隔开来，所有可以在同一行上输入）：

```
chmod 755 /dev / u; chgrp auth / dev / u
```

现在新的文件系统准备好，可供使用。

## 附录 G 配置非缺省参数的磁带机

本附录解释在不愿意使用安装 UNIX 系统前推荐的缺省参数时应该如何配置磁带驱动器。

### G.1 开始磁带配置

应根据以下步骤来开始配置磁带机的工作：

1. 首先，将看到如下信息：

```
You do not have a cartridge tape present at a default setting.  
mkdev tape will onw be called to allow you to configure your  
tape drive.
```

2. 接着，配置菜单显示到屏幕上：

```
Tape Drive Configuration Program
```

1. Install a Tape Drive
2. Remove a Tape Drive
3. Change default tape drive

```
Enter an option or 'q' to quit:
```

3. 如果想增加一个标准的卡式磁带机，应该输入“1”，直接加上磁带机；如果想加上一个 SCSI 卡式磁带机，则应先选择“2”，删除当前配置。然后根据正在配置的磁带机的种类转到相应的部分。需要说明的是，在执行完配置磁带机的命令之后应参照本附录后面“回到 UNIX 系统安装”一节的内容执行。

### G.2 标准卡式磁带

这一部分处理标准卡式磁带机的配置。

1. 首先要求说明选择安装的磁带驱动器的类型

```
Tape Drive Installation Menu
```

1. Install Cartridge Tape Driver
2. Install Mini-Cartridge Tape Driver
3. Install QIC-40 Tape Driver
4. Install SCSI Tape Driver

```
Enter an option or 'q' to quit:
```

输入“1”并回车。

2. 看到:

The Cartridge Tape Driver is already present in system configuration files.

Do you wish to continue? (y / n)

输入“y”并回车。

3. 下面的菜单显示到屏幕上:

QIC Cartridge tape Configuration

1. Display current tape parameters
2. Modify current tape parameters
3. Select previous tape parameters
4. Select default tape parameters

Enter an option or 'q' to quit:

4. 如果不打算改变厂商以缺省值形式设定的 DMA 通道和中断向量, 应输入“q”并回车。然后转而执行第 6 步。

5. 如果想改变缺省的设定, 应输入改变的值。选择选项 2 会显示下面菜单:

<u>Tape Parameters</u>	<u>Values</u>	<u>Comments</u>
1. Controller Type0	1 = type A, 3 = type w, 4 = type E. 5 = type M, 6 = type or 7 = typex	
2. DMA Channel 0	1 or 3	
3. Interrupt Vector	0	logical vecctor number
4. Base Address 0x0	1 / o	addresses start here

Zero Values Imply Auto-Configuration

Enter a parameter to modify or 'q' to return to the main menu:

在此应改变已在控制板上变动过的参数值。方法是输入参数号, 系统会询问新值。值得一提的是, 在改变基地址时, 如果是十六进制地址形式必须首先输入“0x”, 然后才是数字, 否则系统将认为地址是十进制的。另外, 如果选择控制器中断 2, 在修改磁带参数时应指明是 25 号中断, 因为 25 号软中断对应于 2 号硬件中断。除此之外, 其他中断的软件中断号和硬件中断号是一致的。这些工作完成之后, 输入“q”并回车。

6. 会看到下列信息:

System files have been successfully updated.

7. 接下来显示一条处理 SCSI 磁带驱动器的信息, 接着是一个设备名表, 这与标准卡式磁带驱动器无关, 可以忽略。

8. 然后回到配置菜单。输入“q”并回车。转到“回到 UNIX 系统安装”一节。

### G.3 SCSI 卡式磁带驱动器

本节处理 SCSI 磁带驱动器的配置。要配置 SCSI 设备, 必须知道 ID 号和主适配器号。关于 SCSI 配置问题的详细内容, 见书中“使用软盘和磁带驱动器”部分。

1. 首先必须移去当前配置的标准卡式磁带驱动器:

Tape Drive Removal Menu

1. Remove Cartridge Tape Driver
2. Remove Mini-Cartridge Tape Driver
3. Remove QIC-40 Tape Driver
4. Remove SCSI Tape Driver

Enter an option or 'q' to quit:

输入“1”并回车。

2. 会看到:

Updating system files to effect removal of driver ...

System files have been updated.

Cartridge Tape Driver and associated devices have been removed

3. 这时会看到主菜单:

Tape Drive Configuration Program

1. Install a Tape Drive
2. Remove a Tape Drive
3. Change default tape drive

Enter an option or 'q' to quit:

输入“1”并回车。

4. 屏幕上显示安装菜单:

Tape Drive Installation Menu

1. Install Cartridge Tape Driver
2. Install Mini-Cartridge Tape Driver
3. Install QIC-40 Tape Driver
4. Install SCSI Tape Driver

Enter an option or 'q' to quit:

输入“4”并回车。

5. 屏幕显示下面的信息:

The SCST Tape Drive must be configured before use.

Do you wish to configure the SCSI Tape Drive now? (y / n)

回答“y”并回车。

6. 接着要求输入磁带驱动器的位置:

What is the ID of the controller for this device?

Select 0-7, or 'h' for help, or 'q' to quit:

输入主适配器上控制器的 ID 号并回车。

7. 以下是主适配器号:

Which SCSI host adapter supports this device?

Select 0 or 1, or 'h' for help, or 'q' to quit:

输入“0”或“1”并回车。

8. 最后，系统要求输入 LUN（逻辑部件号）：

What is the LUN of this device?

Select 0-7, or 'h' for help, or 'q' to quit:

因为磁带驱动器及其控制器是同一个部件，LUN 值为 0。输入“0”并回车。

9. 看到：

The SCSI configuration files have been updated.

The following special devices have been created:

    /dev / rStp0          / dev / nStp0          / dev / xStp0

Default special devices have been created with the following links:

/ dev / xct0	linked to	/ devxStp0	and / dev / rmt / 0b
/ dev / rct0	linked to	/ dev / rStp0	and / dev / rmt / 0bn
/ dev / nrct0	linked to	/ dev / nrStp0	

10.这时回到主配置菜单。输入“q”并回车，转到下一节“回到 UNIX 系统安装”。

#### G.4 回到 UNIX 系统安装

在选择要安装的磁带驱动器之后，操作系统核心必须重新连接，系统必须重新启动，否则安装工作不能继续进行。

1. 首先看到：

The UNIX operating system will now be rebuilt.

This will take a few minutes. Please wait.

Root for this system build is / .

这一过程需要几分钟。

2. 这一步工作完成之后，看到：

You must reboot the system to continue the installation.

AFTER you see the message \* \* Safe to Power Off \* \*

-or-

\* \* Press Any Key to Reboot \* \*

reboot the system by opening the floppy door and pressing any key.

The screen will clear and you will see the boot message:

    Boot

    :

Press < Return > to reboot the newly configured kernel

3. 接着出现信息：

\* \* Safe to Power Off \* \*

- or -

\* \* Hit Any Key to Reboot \* \*

记住移走软盘驱动器中的盘片，按回车键，系统关闭。



4. 启动提示符出现在屏幕上:

Boot

:

回车。

5. 显示下面信息:

hd (40)unix

屏幕清除, 会看到一些自检诊断。

6. 系统提示放入卡式磁带。

这时应转到安装过程一章, 从第 28 步开始继续安装过程。

## 附录 H 安装词汇解释

如果对一般操作系统或者 UNIX 系统不太熟悉, 有些概念的含义不太清楚, 那么本附录对部分词汇的解释就相当有用。

**备份:** 硬盘上文件的拷贝件, 存于软磁盘或磁带上, 在硬盘数据丢失时可以用来恢复。

**坏道:** 硬盘上丧失数据存储能力的一小块空间。

**坏道表:** 已知的硬盘上坏道的列表, 也包括将来可能坏的磁道空间。

**badtrk:** 用来扫描硬盘并记录坏道的UNIX实用程序。

**BIOS:** 基本输入/输出设备。该数据表描述系统识别出的、已连接到计算机主电路板(母板)上的硬件设备。

**bit:** “0”或者“1”。在二进制系统中, 只有这两种可能性。

**块(block):** 根据具体的说明表示不同的量大小。对操作系统和划分硬盘的实用程序(比如在 divvy 中讨论的那样), 一块表示 1024 字节, 在大多数实用程序以及磁盘空间使用的场合, 一块为 511 字节, 如在 custom 实用程序中。

**启动:** 意为开始startup过程。常用的说法为“启动系统”。

**启动软盘:** 指从软盘驱动器启动系统的操作系统核心的那张软盘。

**bootstrap:** 将操作系统核心装入内存的程序。

**字节:** 8位一组, 是基本信息单位。一个单字节可以描述一个字符, 如“a”或“5”或“#”, 例如在 ASCII 字符集中, 字符“A”用包含 0100.0001 这 8 位的字节来表示, 在十进制中值为 65。

**Custom:** 安装操作系统产品和任选软件包的程序。

**设备驱动程序:** 允许核心和某一特定设备通信的程序。有些驱动程序, 如软盘和硬盘的驱动程序, 已经存于核心中。其他的驱动程序则必须用 linkkit 附加或连接到核心中对应设备才能使用。

**divvy:** 将某一分区划分成独立的文件系统的程序。

**dkinit:** 使得UNIX系统能够识别出不包含ROM BIOS的硬盘的程序。

**驱动程序:** 见设备驱动程序条目。

**fdisk:** 将硬盘划分成分区, 或者说划分成驻存不同操作系统的几个部分的程序。

**文件系统:** 独立于操作系统和实用程序部分的分区的一部分。文件可以安装和拆卸,就象软盘和其他设备一样。

**坏块图:** 由厂商提供的已知坏道表。

**fsck:** 在系统遭受破坏之后检查和清理文件系统的程序,可将各种系统表格恢复原貌。

**千兆字节(gigabyte):** 一千兆字节,缩写为“G”。换言之,是 $1024 \times 1024 \times 1024$ 字节或者 $1024 \times 1,048,576$ 字节或 $1,073,741,824$ 字节或者2的30次方,是巨大数量的信息,你可以建立和使用达4千兆的文件,而硬盘容量可为1千兆或者更多。

**硬盘:** 实际是一组磁盘,或磁碟,由一组磁头进行扫描,如同磁带录音机一般。

**核心程序:** 操作系统的中心,总是驻留内存,所有的应用程序、实用程序都在它上面运行。

**kilobyte:** 通常写作k(kilo),意为1000。1K是1024字节。1K不是1000而是1024的原因在于2的10次方是1024,而在二进制中,所有的数值都基于2的方次而不是10的方次。

**linkkit:** 用来将设备驱动程序连接到操作系统核心的一组程序。参考“设备驱动程序”条。

**megabyte:** 有时简写成“meg”或“M”或“Mbytes”,1M是1024K,换言之,是 $1024 \times 1024$ 字节,或者 $1,048,576$ 字节或2的20次方,为相当大数量的信息,通常用来量度像硬盘之类的介质的尺寸大小。

**mkdev:** 建立与设备对应的设备文件的程序。对一个系统来说,一个设备就是一个写入或读出的文件。这样,主硬盘对应于文件/dev/hd00。

**分区:** 硬盘上驻存整个操作系统的一部分。如果整个硬盘都用作UNIX系统,则硬盘上只有一个分区。

**进程:** 当前正运行于系统上的程序或程序的一部分。

**RAM:** Random Access Memory的缩写,是主要存储介质(主存),而硬盘存储器称为辅存。在带虚拟存储器的系统中,RAM和硬盘辅存实际上是一个部件。

**重启动:** 关闭系统之后又将它打开的操作。

**根:** UNIX系统中最高级别的帐号,可以不受任何约束地访问所有文件和实用程序,该帐号仅用于系统维护。

**根文件系统:** 包含操作系统及其实用程序的文件系统。

**实时系统RTS:** RTS是UNIX系统正常运动所必需的最小量的软件,在它之外的其他软件包可以用custom装入。

**扇区:** 磁盘的一部分。

**shutdown:** 是关闭开关或重启动之前用以关闭系统的程序。

**单用户方式:** 见系统维护方式。

**超级用户:** 根的另一名称。

**对换区:** 用作内存扩展的硬盘(不是文件系统)的一块区域,运行着但正在等待输入或设备访问的程序被移到这一区域,使得其他程序得以在内存

中运行。

**系统维护方式:** 又称为单用户方式,是系统建立阶段执行维护任务的方式,其它用户不允许登录于该方式,如根或超级用户帐号,本方式在访问系统时不受任何权限的限制。

**tpi:** 每英寸磁道数,是软盘存储密度大小的量度。

**磁道:** 硬盘划分成的同心圆环。

**实用程序:** 有时被看作操作系统的一部分,但不是操作系统核心的一部分,是如同应用程序一般地在系统上运行的独立的用户程序。用户自己可以编制新的实用程序,称为“工具”。

## 第二部分 日常管理指南

### 第一章 启动和停止系统

本章介绍了实现以下几项任务的方法：

- 启动和停止系统
- 作为超级用户（根）登录
- 改变系统引导过程
- 使用在引导时显示出的设备和系统配置信息

#### 1.1 启动系统

启动系统的工作当然不只是把电源打开这么简单。还必须执行一系列步骤初始化系统。这些步骤包括：

- 加载操作系统
- 检查文件系统（检查系统是否被不正常地停止）
- 选择系统操作方式

##### 1.1.1 加载操作系统

启动系统的第一步是从计算机硬盘上加载操作系统。步骤如下：

(1) 打开计算机和硬盘电源。计算机加载 UNIX 引导程序并显示以下信息：

```
SCO system V / 386
```

```
Boot
```

```
:
```

(2) 按 RETURN 键。引导程序加载操作系统。系统加载完成后，显示有关系统自身的信息，并检查根文件系统（包括所有文件与目录）是否状态正常，是否已被侵害。如果状态正常未被侵害，则称文件系统是“干净”的。如果根文件系统干净，就可以选择操作方式，否则，必须首先清理文件系统。

##### 1.1.2 清理文件系统

如果系统显示如下信息：

```
fsstat: root file system needs checking
```

```
OK to check the root file system (/dev/root) (y/n)?
```

就必须清理文件系统。这条信息表示系统没有按本章稍后“停止系统”一节中所介绍的方法停止系统运行。操作系统想正常工作，文件系统必须是“干净”的。如果上述信息没有出现，就表示文件系统是干净的。

如果要清理文件系统，则键入 Y 作为对于提示信息的肯定答复，然后按

〈RETURN〉键。fsck (ADM) 实用程序就会清理文件系统，修复被损坏的文件、检测出不能修复的文件。每一步工作完成之后都要报告工作进程。到某一步时，可能会问及是否希望保存文件，这时键入 y 或 n 然后按回车键作为肯定或否定的答复。在本书中“文件系统的管理”部分对 fsck 实用程序详细解释。

清理完成之后，系统要求你选择操作方式。

### 1.1.3 检查安全数据库

每次重新启动系统时（假定文件系统是干净的，如果文件系统不干净，则假设已经执行过 fsck），系统都自动检查关键的安全数据库文件。提示信息为：

```
checking tcb...
```

```
check protected subsystem databases (y/n)?
```

这一检查工作是为了避免系统出现问题。如果文件丢失（这是极少发生的），则提醒你注意并从备份中恢复该文件。关于这一点的详细描述见本书“系统安全性维护”一章。

### 1.1.4 选择系统操作方式

当一看到如下信息就可开始选择操作方式：

```
INIT: SINGLE USER MODE
```

```
Type CONTROL-d to continue with normal startup,
```

```
(or give the root password for system maintenance);
```

系统有两种方式：普通操作方式和系统维护方式（普通操作方式又称为多用户方式，而系统维护方式称为单用户方式）。普通操作方式主要完成系统中的普通工作。若允许多用户注册并开始工作，则应选这种方式。系统维护方式是留给系统管理员使用的特殊方式，它不允许多用户工作。

要选择普通操作方式，应按 Ctrl-d，系统即显示启动信息，并提示输入系统时间（见下一节）。系统接着执行 /etc/rc 文件和目录中找到的命令。/etc/rc 包括 /etc/rc 文件，/etc/rc0.d，/etc/rc1.d 和 /etc/rc2.d 目录，逐次往下，统称为 /etc/rc。执行不同文件中的命令产生不同系统设备的启动信息，比如说打印机，网络设施的启动信息。接着，系统显示提示信息“login:”，这时就可以作为普通用户登录注册，或者按稍后将谈到的办法作为超级用户登录。

要选择系统维护方式，应键入超级用户口令（又叫做根口令）然后回车。

注意：超级用户口令是在系统安装时指定的。如果不知道，可以问给你安装系统的管理员。

在口令正确之后，系统要求设置时间。之后即出现超级用户提示符（#）。不执行 /etc/rc 中的命令。（只有当要求所有其他用户都退出系统以便做系统维护工作时才选择系统维护方式）。用 Ctrl-d 可退出系统维护方式，系统自动进入普通操作方式。

要从普通操作方式进入系统维护方式，应作为超级用户登录，并发如下命令关闭系统：

```
/etc/shutdown-gn
```

（使用 sysadmsh 的用户应选择：System→Terminate）

此处 n 是多用户方式停止的分钟数。接着系统要求确认你的选择并按任意键重新启动。

如果希望从普通操作方式直接进入单用户方式，可作为 root 登录注册，并发如下命令：

```
/ etc / shutdown -gn su
```

su 参数表示不必关闭系统而直接进入系统维护方式。

(sysadmsh 用户不能直接进入单用户方式)

### 1.1.5 设置时间和日期

一旦启动了普通操作方式，系统会询问正确的时间和日期。先显示当前时间及日期以及如下信息：

```
INIT: NEW run level: 2
```

```
Current System Time is Wed Nov 29 08:19:00 pst 1989
```

```
Enter new time ([yymmdd]hhmm):
```

如果系统的时钟电池正常，则无必要改变日期。如果时间和日期都不改动，按 RETURN 键即可，否则，输入新的时间，表示方法为两对或多对连续数字串，数字串可能是如下几种情形的一种或几种：

yy (任选项)：表示当前年，形式为 2 位数字串，从 00 至 99 表示 1900 年至 1999 年；

mm (任选项)：表示当前月份，形式为 2 位数字串，从 01 至 12 代表 1 月至 12 月；

dd (任选项)：表示当天日期，形式为 2 位数字串，从 01 到当月的天数。

hh：表示当时小时数，形式为 2 位数字串，从 00 至 23。小时的表示以军事时间为准，即午前时间从 00 至 11，午后时间从 12 至 23。

mm：代表当时分钟数。形式为 2 位数字串，从 00 至 59。

例如，要将时间和日期改为 1991 年 3 月 31 日正午，则应输入：9103311200，然后回车，系统即会显示新的时间和日期如下：

```
Sun Mar 31 12:00 PST 1991
```

如果输入了不正确的值，系统会提示你再输入一次。如果没有输入任选项的值，则该项的当前值保持不变。如果输入了新的年值，则必须同时输入月值和日值，类似地，如果输入了新的月值，则必须输入日值。

在时间和日期之后是设备启动信息以及“login”信息。

## 1.2 作为超级用户注册

许多系统维护工作都要求在超级用户状态下进行。例如，要停止系统运行就必须作为超级用户注册。

在作为超级用户注册之前，需要知道超级用户口令，并要在终端屏幕上看到“login”信息。如果没有见到这条信息，则按 ctrl-d 键直至该条信息出现为止。

以超级用户注册时，步骤如下：

1. 当看到“Login”时，打入超级用户注册名：

root

并按回车键。系统提示超级用户口令。

2. 打入超级用户口令并按回车键（超级用户口令是在系统安装时指定的）。为保密起见，系统不显示你打入的口令，所以必须仔细地打入每个字符。系统打开超级用户帐号并显示日期信息以及超级用户提示符（#）。

由于超级用户在系统中具有以上的权力，当你作为超级用户注册时必须特别小心。特别是，当删除或修改文件或目录时更要非常仔细。因为超级用户可以不受任何限制地访问所有文件，可能由于误操作删去或修改了某个对系统非常重要的文件。应避免在文件名中使用通配符并且应经常检查自己的工作目录。在超级用户状态下一个小的错误也可以导致系统或文件出现不可挽回的损失。

任何时候均可按 ctrl-d 键，退出超级用户状态。

### 1.3 停止系统

停止 UNIX 系统运行的工作并不仅是关掉电源。不管是用 shutdown (ADM) 命令还是在某种条件下用 haltsys (ADM) 命令停止系统都必须先使系统做一些准备工作。以下对命令加以说明。

在用 shutdown (ADM) 停止系统之前，必须通知用户“系统将要停止”的信息，此外可能还要告诉用户其他信息，如系统何日再次启动等。

用 wall (ADM) (“write to all”) 命令可以把全系统范围内的信息送往当前注册的所有用户终端上：

wall

然后回车，接着输入信息，如果一行不够，可回车另起一行。输入完成之后，按 ctrl-d，系统的所有终端上就会显示输入信息。

shutdown 命令是用以停止系统的通常办法，在系统处于普通操作方式时，即应该使用该命令，它提示用户“系统将要关闭”并给用户时间做善后工作。提示信息一般是固定的。（如有必要，系统管理员在执行 shutdown 命令之前也可以先用 wall 命令将系统将要关闭的信息告诉用户）。

用 shutdown (ADM) 命令停止系统运行时，应遵循以下步骤：

1. 作为超级用户注册，方法可参阅本章前部。系统打开超级用户帐号并显示日期信息和超级用户提示符（#）。

2. 敲入以下命令并回车

(sysadmsh 用户应选择：

system → Terminate)

/etc/shutdown-n

其中 n 是系统停止运行之前的延迟分钟数。系统在每个终端上显示警告信息，要求注册用户完成善后工作并退出注册。在所有的用户都已退出或到了限定的延迟时间之后，系统就关闭所有帐号并显示如下信息：

\* \* safe to power off \* \*

—or—

\* \* Press Any Key to Reboot \* \*

3. 关闭计算机电源或按任意键重新启动。

haltsys (ADM) 命令可立即停止系统运行, 该命令一般只在系统维护方式下使用。如果在使用 halt 命令时还有其他用户注册, 则这些用户被强制退出, 来不及做善后工作, 可能导致损失。另外, 网络服务程序和其他程序非正常退出, 以后重新启动时会遇到麻烦。

为 haltsys 命令停止系统运行时应遵循以下步骤:

1. 必须处于单用户状态。(切记 haltsys 命令不可在多用户状态下使用)。
2. 输入命令:

/etc/haltsys

然后回车, 系统显示如下信息:

\* \* Safe to Power Off

—or—

\* \* Press Any Key to Reboot

3. 关闭计算机电源, 或按任意键重新启动。

#### 1.4 关于启动显示信息

启动时, 在显示版权信息之前, 会显示一个给出硬件信息的表格。该表将操作系统识别出的用户配置情况显示到屏幕上。下面给出这样一个表格:

device	address	vector	dma	comment
fpu	—	35	—	type = 80387
fioppy	0x03F2-0x03F7	06	2	unit = 0 type = 96ds15
serial	0x02F8-0x02FF	03	-	unit = 1 type = standard nports = 1
parallel	0x0378-0x037A	07	-	unit = 0
console	-	-	-	unit = ega type = 0 12 screens = 68k
disk	0x01F0-0x01F7	36	-	type = W0 unit = 0 cyls = 791 hds = 16 secs = 48

下面对表格的有关内容加以解释:

device, address, vector, dma, comment: 分别是硬件设备名, 十六进制地址, 中断向量, DMA 通道以及关于硬件设备的其他信息。

fpu: 浮点运算部件, 专制 Intel 80387 芯片。

floppy: 高密软盘驱动器。

serial: COM1。COM1 有一个端口 (没有安装多端口卡)。

parallel: 并行口

console: 控制台包括一个与 (0 类) IBM EGA 设计相兼容的 EGA 视频适配器

disk: Western Digital st506 控制器号 0 (W0), 0 号硬盘驱动器 (unit 0), 以及柱面数, 头数以及扇区数

hwconfig (c) 实用程序可以随时使用 /usr/adm/hwconfig 文件中的配置信息来显



示或访问这类信息。

## 1.5 改变引导进程

每次启动计算机时，系统都运行 boot 程序。除非在出现提示时发不同的指令，boot 程序都会用根文件系统下 /etc/default/boot 文件中指明的配置值来装载缺省核心程序。可以编辑 /etc/default/boot 文件，改变缺省配置值，使以后的启动过程装载不同的配置，也可以在 /etc/default/boot 文件中设置某一任选参数，允许系统自动进行启动。

/etc/default/boot 文件中的缺省参数值可以在 boot (HW) 手册中查到。通过编辑 /etc/default/boot 文件，可以改变任选参数值。

在出现启动提示符时如果只按回车键，到底装载哪一个程序将决定于 /etc/default/boot 文件中的 DEFBOOTSTR (default bootstring) 任选项。例如，在 /etc/default/boot 文件中如下的设置将使启动程序默认从硬盘上装载操作系统核心：

```
DEFBOOTSTR=hd (40) Unix
```

参数的前两个字母指明设备名 (hd 表示硬盘或用 fd 表示软盘)，括弧里的值表示次要设备号 (40 表示硬盘上的根文件系统)。括弧之后是要装载程序的路径名。

在 /etc/default/boot 文件中对任选值 AUTOBOOT, TIMEOUT 和 PANICBOOT 的设置控制是否要自动启动以及什么时候自动启动等。

如果 AUTOBOOT=NO, 则引导程序无限期地等待对提示符的响应，也可以设置 AUTOBOOT=YES, 允许系统在提示符出现一定时间之后还未有响应的情况下自动启动。这一定时间的默认值为 60 秒，但可以通过 TIMEOUT 任选参数任意设置等待的时间数。当设置的时间到了而没有收到响应，引导程序就如同接收到回车那样，用在 /etc/default/boot 文件中指定的配置值来执行默认引导进程。

还可以通过设置 PANIC 为 YES 或 NO 来指定系统在遭受意外事故时是否应该重新启动 (意外事故通常会使得系统停止运行)。

## 第二章 系统管理工具——sysadmsh

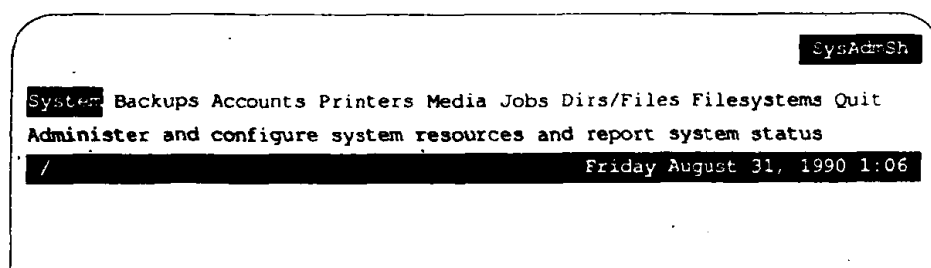
sysadmsh (system administration shell) 是一个为了简化系统管理工作而设计的菜单接口工具。sysadmsh 可以使管理员通过在各菜单上进行各种选择，而不是运用传统的 UNIX 命令行的方式，来执行许多系统管理命令。

本章介绍如何使用 sysadmsh 这一工具。为了更好地利用 sysadmsh，还必须了解一些能被 sysadmsh 调用的 UNIX 命令。在本章末尾有一张这些命令及与其相对应的 sysadmsh 菜单选项的表格。如果在学习本章的同时还在机器上实际运行所看到的例子，会感到 sysadmsh 是很好学的。在使用 sysadmsh 之前，请先了解本章所讲的各种概念。

### 2.1 sysadmsh 的屏幕组织

为了执行 sysadmsh，必须在根 (root) 下注册，然后键入命令：sysadmsh。

sysadmsh 主菜单显示如下：



```

SysAdmSh
System Backups Accounts Printers Media Jobs Dirs/Files Filesystems Quit
Administer and configure system resources and report system status
/ Friday August 31, 1990 1:06
```

下面是 sysadmsh 屏幕示意图，屏幕上的黑色区域表示高亮度区域或是正文亮带，每个区域都显示不同种类的信息：

- 屏幕右上角的高亮度正文亮带是上下文指示。它显示当前菜单名。sysadmsh 第一屏的上下文指示显示 SyAdmsh。

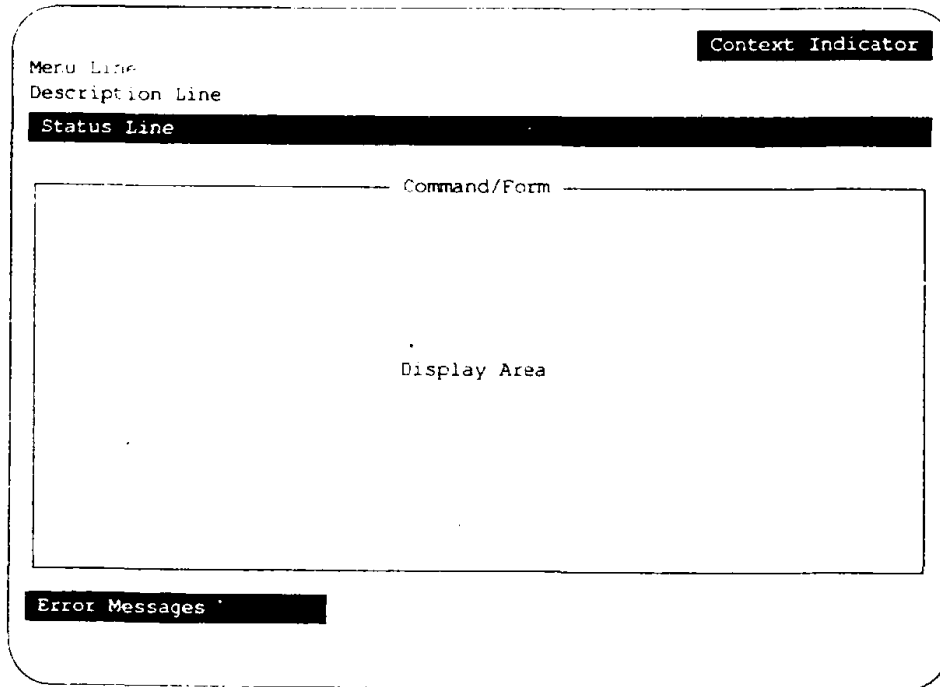
- 菜单行 (Menu Line) 显示该菜单当前所提供的菜单选项。sysadmsh 的主菜单所包括的九个选项是：System、Backups、Accounts、Printers、Media、Jobs、Dirs / Files、Filesystems 以及 Quit。

- 描述行 (Description Line) 提供了当前处于高亮度显示的那个菜单选项的简单说明。

- 把菜单行、描述行与显示窗口 (Display Window) 隔开的那个高亮度正文显示带是状态行 (status Line)。sysadmsh 第一屏的状态行显示的是日期、时间、以及当前的工作记录。在一个 UNIX 命令被执行时，这个命令的名字以及所选到的所有选项都在状态行的最左端简略地显示出来。

- 命令 / 表格 (Command / Form) 行显示的是显示区域 (Display Area) 中内容的标题。这个标题可以是一个 UNIX 命令名，也可以是一个 sysadmsh 的表名。当显示的是一个命令名时，这个命令在用户手册中的位置也被放在一个括号中紧跟在其后显示。譬如，当选了 System→Report→Users，在命令行就显示“who (C)”。这表示这个命令可以在用户手册 (User's Reference) 中的命令部分 (Commands (C) section) 找到。

- 显示区域 (Display Area) 显示 sysadmsh 的表格和搜索窗口。表格 (forms) 和搜索窗口 (scan windows) 将在本章的后面详细的解释。
- 错误信息 (Error Message) 和恢复指令以高亮度正文显示方式出现在屏幕最后一行。



## 2.2 Sysadmsh 的操作

### 2.2.1 菜单项的选择

表 2.1 列出了在菜单上移动光标的方式。请注意有好几种方法可以用来选择菜单选项，如果以前曾用过基于菜单驱动的程序，那就还可用最熟悉的方法。

可以通过使用箭头或空格键把光标从一个选项移到另一个选项上，熟悉一个菜单的各个选项。每次当把光标移到一个新的选项上时，那个选项说明就会在描述行显示出来。

Sysadmsh 具有分级的菜单结构。许多菜单选项能使用户进入下一级菜单。例如，当在主菜单中选了 Jobs 这一项，一个包含更多的选择项的子菜单就显示出来，可使用户检查并操作机器上的进程。这种菜单分级可以很方便地通过从一个菜单进入下一级菜单来查找所需要的命令，最后选中了一个菜单选择项，这个选项或是执行了一个 UNIX 命令或是显示一个表格让你填入该命令所需要的详细参数。请注意键入选项名的第一个字母是进入某一级菜单的最快的办法；可以根据记忆敲入三四个字母，就能够立即找到所需要的功能。（在本章末尾的表 2.9 列出了 Sysadmsh 所提供的所有选择项和速记形式。）

表 2.1 基本的菜单敲键方式

敲键方式	作用
箭头键或空格键（与右箭头键相同）	移动到某菜单选项上
各选项的第一个字母键或把光标移至某选项上再按回车键	选中该菜单选项
<ESC>	退回到前一个菜单
<F1>	请求帮助

学习如何使用菜单的最好办法就是运用上述的敲键方式来实践菜单选择过程。如果错误地选了一项，可以按 <ESC> 键退回到前一个菜单。无论在哪一级菜单中，都可以按 <F2> 键后再按 n，回到主菜单。<F2> 是退出的选择，n 就代表回到主菜单。（如果已经在执行一个 UNIX 命令，如 vi，那么 <F2> 键就没用了）为了帮助读者找到在 sysadmsh 菜单中的路径，表 2.2 列出了第二级菜单的各种选项。

表 2.2 第二级菜单选项表

System	Backups	Accounts	Printers	Media	Jobs	Dirs / Files	Filesystems	Quit
↓	↓	↓	↓	↓	↓	↓	↓	↓
Report	Create	User	Configure	List	Report	List	Check	Yes
Configure	Restore	Defaults	Schedule	Extract	Terminate	View	Mount	No
Hardware	Schedule	Terminal	Request	Archive	Authorize	Copy	Umount	
Software	View	Report	Ausiliary	Format		Edit	Add	
Audit	Integrity	Check	Priorities	Duplicate		Modify	Floppy	
Execute				Tapedump		Print	DOS	
Terminate						Archive		
						Differences		
						Remove		
						UseDOS		

本章使用了一种句法规则来表示一个菜单选择串。例如，为了打印一个文件，必须从主菜单中选择 Dirs / files 这个选项，然后再从 Dirs / files 菜单中选出 Print 这个选项。这个序列就可以用速记符号 Dirs / files → Print 来表示，它也可通过键入 dp 来执行。

当选中了一个菜单选项，下列三种情况之一必将发生：

- 显示下一级菜单
- 显示一张表格
- 执行了一个 UNIX 命令并且在一个搜索窗口里显示了一个结果。

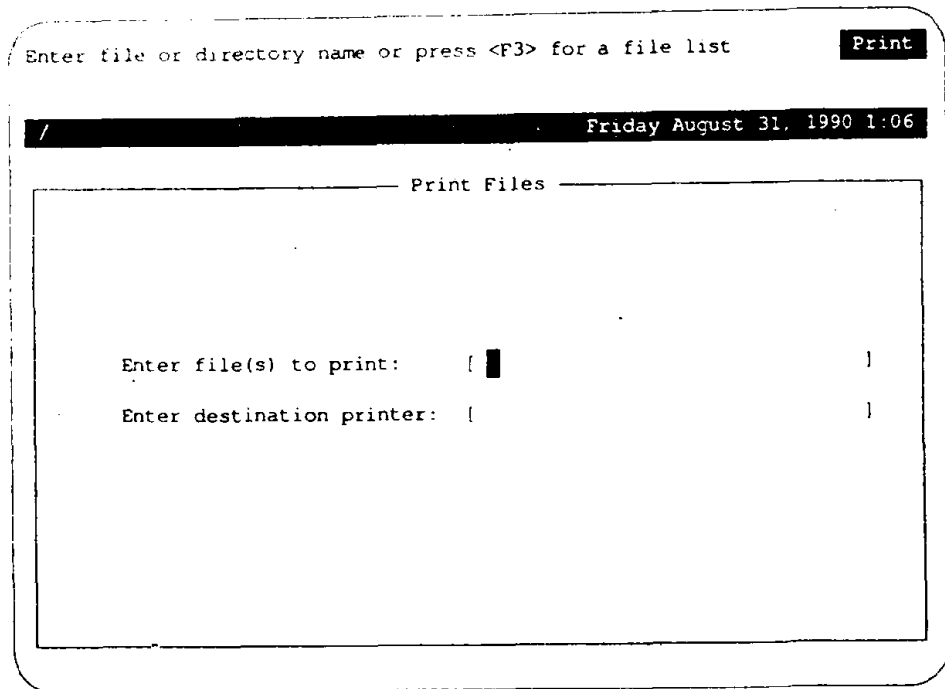
下面两节将讲述表格和搜索窗口。

### 2.2.2 表格 (form) 的使用

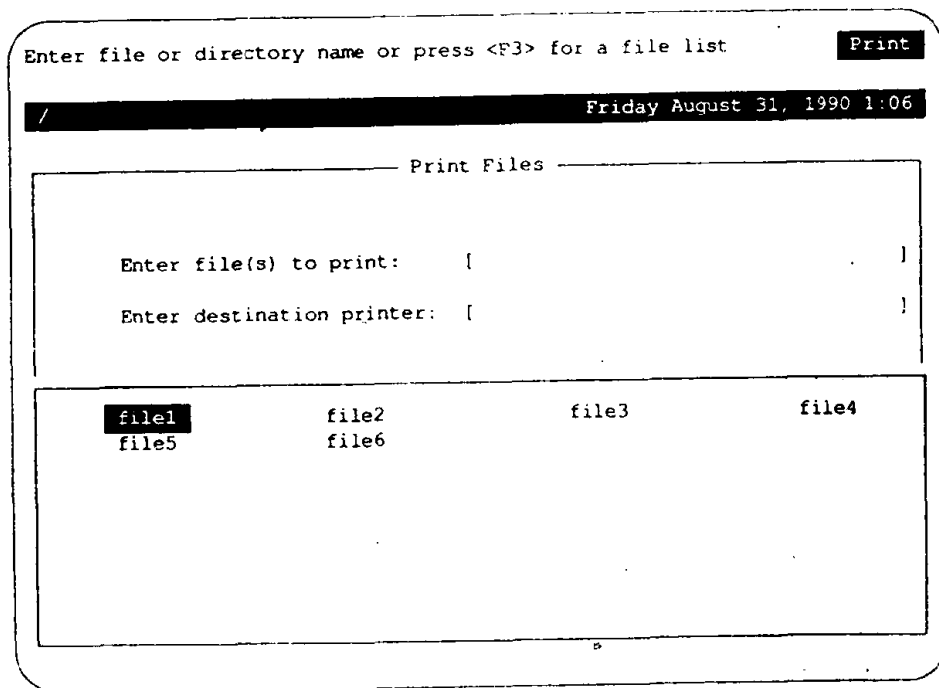
有些菜单选项还需要另外的信息才能正确地工作。例如，对于 Print 这个选项，如果不告诉它要打印哪个文件和用哪个打印机，它就不能工作。当选了这种选项后，在屏幕上就会出现一个表格。正是通过填这个表格，把这个命令所需要的信息告诉了它。

下面这个例子是打印当前目录下的一个文件，该例子示范了怎样使用表格。这个例子之后的表 2.3、表 2.4、表 2.5 列出了一些键的使用方法，这些键是用来在表格中移动光标、对表格进行编辑、以及进行“点菜”选择的。

要打印一个文件，先要选择 Dirs / Files → Print。这时 Print 表格显示出来：

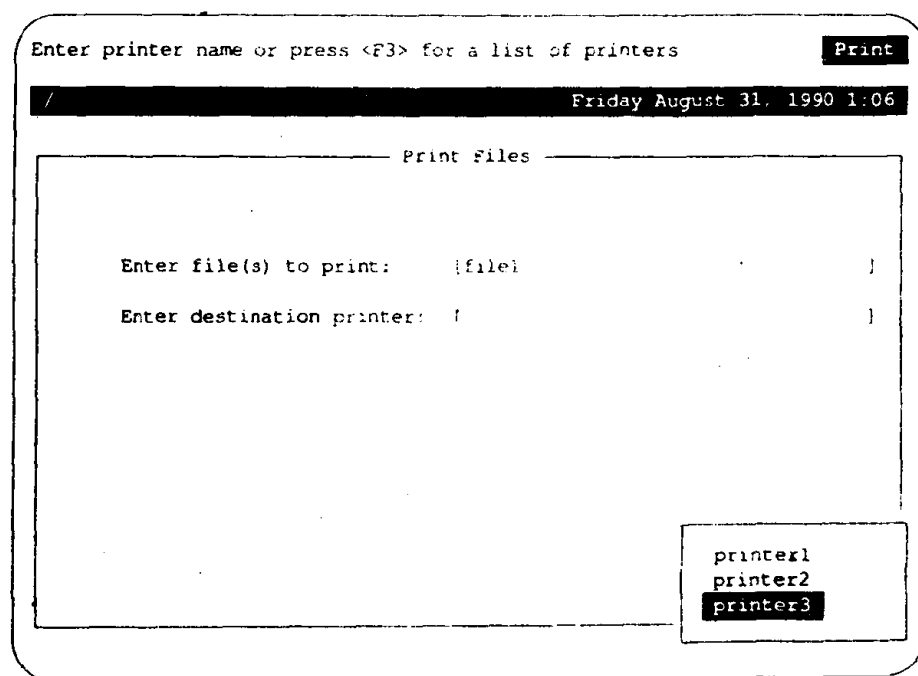


请注意光标是在表格的第一项上。可以直接填表，也可以按<F3>键而得到一系列供选择的内容。如果知道要打印的文件名，就直接填入。如果需要去找文件名，就按<F3>，此时就在原来的Print表格上又叠加了一个窗口。这个窗口中列出了一系列文件名供选择。要选其中的某一个，只要把光标移到这个文件名上，然后按回车键把它选出来。这就是所谓的“点菜”，它可以用来在任何一列显示出来的内容中进行选择操作。选择完后，这个窗口就消失了，又回到原来的Print表格中。



请注意此时所选中的文件名已经填到表格中。这时也可以利用编辑键来修改这个名字（编辑键在后面的表 2.4 中列出），或者按回车键到下一个表项中。

现在该填要用的打印机的名字了。如果不知道打印机名，就按 <F3>，一个小窗口出现，其中列出了所有安装在该机器上的打印机。可以像选文件名一样选一个打印机，此后，又回到 Print 表格中。



在下面的表 2.3 中列出了一些键的使用，使用这些键，可以很方便地利用 sysadmsh 的表格。

表 2.3 表格键

键的使用	作用
<ESC>	告诉程序不想继续填表了。这个表格被删除，什么也不做，又回到了原来的菜单下。
上、下箭头键	把光标移到表格中的其它项中。有些表格项是有限制的，不要求填任何内容，箭头键就会跳过它们，而其它的项目是必须填的。当在最后一个项目上继续按下箭头，光标就会回到第一个项目上。
左、右箭头键	在当前项目上左、右移动光标。可以不必重输入整个一行，而只修改其中的几个字符就行。
回车键	完成一个项目的数据输入而把光标移到下一个表项中。当填完最后一个表项按回车键表示完成了整个表格，并通知 shell，其中的数据可以使用。
<CTL>x	不管处于哪一个表项，都可以退出并执行此表格的功能。可以把“x”看作“执行 (execute)”。<F10> 也可以起到同样的作用。
<F4>	在表格中调用拼写检查工具。如果怀疑一个单词拼错了，可以把光标移到此单词上并按 <F4>，一系列拼写正确的单词就显示出来，在其中选一个代替拼错的单词。

表 2.4 编辑键

键的使用	作用
<CTL>y	删除当前行，并重新开始。
<CTL>w	删除当前单词。
<CTL>g-<CTL>h	把光标移到一行的起始处。
<CTL>g-<CTL>l	把光标移到一行的末尾。
<CTL>v	进入或退出打印状态。
<DEL>	删除光标处的字符。
<BKSP>	删除光标左侧的一个字符。
<CTL>u	翻到前一页。
<CTL>d	翻到下一页。
<CTL>n	移到下一个单词。
<CTL>p	移到前一个单词。
左、右箭头键	在一个编辑行中左、右移动光标。

表 2.5 点菜键

键的使用	作用
回车键	选中此项
<ESC>	停止此选择过程。该项目表被消去，并不做任何动作。
<CTL>v	选择表中列出的所有项或一项也不选的切换开关
上、下 箭头键	把光标移到另一个选项上
左、右 箭头键	在各列之间移动光标
<Space>	当需要选择多于一个选项时，在各所要选的项目上作标记。这个标记是在所选中的项目的左侧标一个“*”。还可以在此项上再按一个 <space> 把这个标记去掉。当所有的标记都做完后，按回车键把它们全选出来。
<F5>	在很多项目中寻找项目，这个键也叫“查找”键。按 <F5> 后就会出现一个提示符，这时填入要找的一串字符，然后按回车键。如果这项找到了，光标就会移到它的上面，再按回车键就把它选出来。如果没找到，光标就不会移动。“;”和“:”分别使你继续向前找和向后找。
第一个字母	用其第一个字母来进行选项是一种最快的选项方法。再按回车键就把光标所在的项选出（如果只有一个项目是由所敲的字母开头的，只敲第一个字母键就行了，不必再按回车键）。如果有几项都是由这个字母开头，光标只移到其中的第一项。

### 2.2.3 收音机按钮 (Radio Buttons) 的使用

收音机按钮就是 sysadmsh 的表格 (form) 中的一行或几行选择单元，它们可以像菜单中的选择项一样来选择。收音机按钮的目的是在一个表格中提供第二级选择，请看一个

例子，它是从 sysadmsh 的 Accounts 菜单得到的。

在这个例子中，“Username”这个域必须在选择收音机按钮前填上。sysadmsh 的句法中为收音机按钮提供了一个标记，——一个冒号(:)，请看下边的表示：

Accounts→User→Examine: Audit

sysadmsh 的这种表示形式就表示在选择 Audit 之前必须先提供信息（这里是用户名）。

```
Name of an existing user (<F3> for list) Examine
-----
Friday August 31, 1990 1:06
-----
View/Modify an existing user's account
-----
Username : '[ ]'
-----
[ Audit ] Expiration Identity Logins Password Privileges
```

#### 2.2.4 搜索窗口 (scan windows) 的使用

在通过从 sysadmsh 菜单中选择一个选项而执行了一个 UNIX 命令之后，其执行结果一般都是在搜索窗口中显示出来。搜索窗口也可以显示文件或目录的内容。为了看看如何使用搜索窗口，让我们假设有看看现在有哪些用户在系统中注册了，选 System→Report→Users（这就执行了 UNIX 中的 who 命令），这时就会出现一个搜索窗口，who 命令的输出结果就在这个显示区域中：

```
<ESC> to exit; Movement keys are active
-----
who -B Friday August 31, 1990 1:06
-----
who(C)
-----
```

NAME	LINE	TIME
root	tty01	24 May 10:23
faithz	tty02	24 May 11:03
stevem	tty03	24 May 8:16
naomib	tty04	24 May 8:00
terib	tty08	24 May 8:16
martinm	tty11	24 May 9:09
mattb	tty14	24 May 7:49
teresae	tty16	24 May 10:29
danju	tty20	24 May 10:05
docadm	tty23	24 May 11:05
stuartc	tty27	24 May 8:26



请注意这个命令名及其在参考手册中的位置的描述符 (C) 都在这个窗口的顶部显示出来。再请注意这个命令的可选参数 (-H) 也在状态行的命令名的右侧给出。

在窗口的最右边有一条垂直的卷屏标志带 (scroll bar), 如果当前窗口中的内容是全部输出内容的最前面一部分, 则 (+) 号就处在卷屏标志带的最上端; 如果当前内容是最后一部分, 则 (+) 号就在最下端。如果在窗口中包括所有的内容输出, 两端的两个 (+) 号就都显示出来。这个卷屏标志带还可以指示这个窗口中的内容属于整个输出内容的哪部分。卷屏标志带的高亮度显示部分就表示当前窗口中的内容, 当向上或向下卷屏时, 这个高亮度显示部分也在标志带上跟着移动。

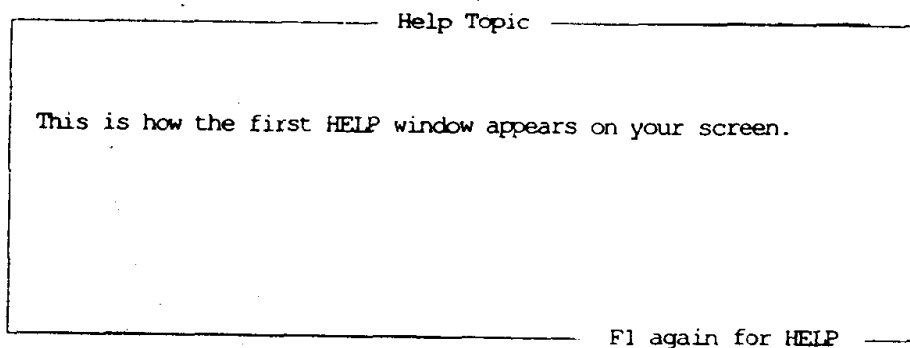
在搜索窗口中可使用表 2.6 中所列的键。

表 2.6 搜索键

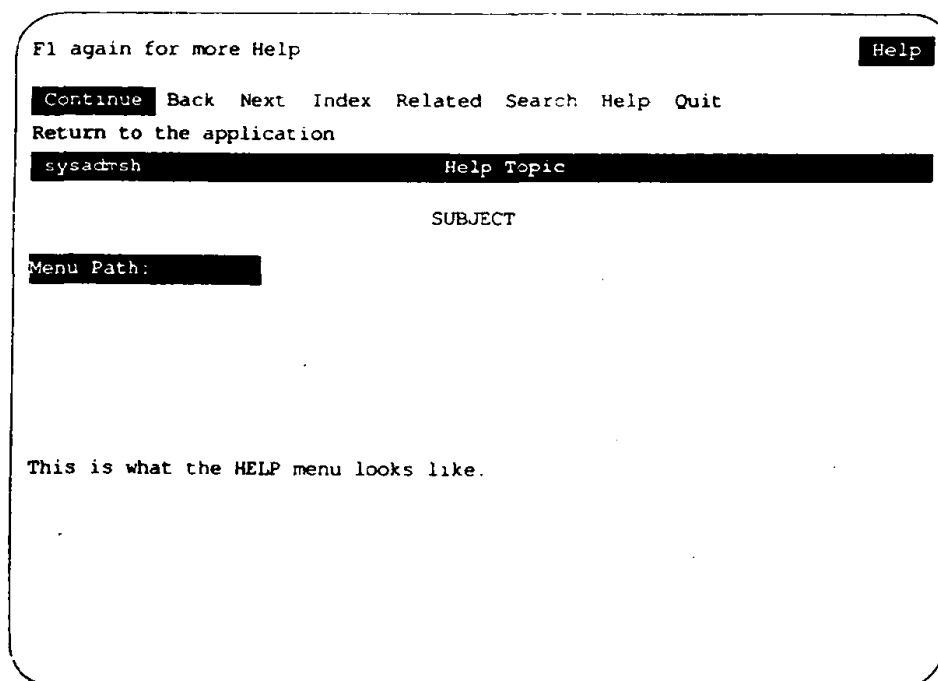
键	作用
<ESC>	退出文件
上箭头键	向上移一行
下箭头键或回车键	向下移一行
<PgDn> 或 <Space>	向下翻一页
<PgUp>	向上翻一页
<Home>	移动显示的最顶端
<End>	移动显示的最尾端
<F5>	进行串匹配。(; 和: 分别表示继续向前或向后匹配)
<F7>	打印在当前窗口中的命令执行结果或文件。

### 2.2.5 请求帮助

可以按 <F1> 键, 它可以给选择提供更多的信息。当按了 <F1> 键, 在当前屏幕上就会出现一个 Help 窗口, 就像下面这样:



这个窗口包含一些基本信息, 如果需要更多的帮助, 就再按一次 <F1>, 这时一个完整的 Help 菜单就显示出来:



当看完之后，在 Help 菜单中选择 Quit 就又回到了在 sysadmsh 菜单中原来的地方。Help 菜单选项在表 2.7 中列出：

表 2.7 帮助选择项

选择项	作用
Continue	在下一页继续显示功能说明。所有的纵向移动键都有效：上、下箭头键、上、下翻页键以及 <Home> 键、<End> 键。如果没有再多的信息，高亮度显示状态移到 Quit 选项上并在说明行上显示“Return to the application”。
Back	回到前一个标题页。这个选项没有与之相反的“Forward”选项。这个选项也可回到一个更概括的标题，并可以一直回到最高一层的介绍标题。
Index	从标题索引表中选择一个新的标题。
Related	选择一个与当前标题相关的新标题。
Search	通过匹配寻找一个新标题。首先要说明在哪儿找（在标题中还是在正文行中，或是在两者中同时找），然后给出匹配串。这个匹配串可以是一个简单的关键字（如“create”或“date”）或是更复杂一些的如“regular expression”，这时一些包含有匹配串的标题就列出来了。
Help	说明“帮助工具”本身是如何使用的。在屏幕上显示一张与这张表类似的表。如果还想要更多的信息，请读有关 Index、Related、和 Search 的标题内容。
Quit	退出 Help 回到 sysadmsh。按 <F2> 或 <ESC> 是快速退出的另外的方法。

每一个 Help 屏幕都包含一个总的信息以及有关各菜单选项的特殊信息。每一个描述段的前面都有一个相关的菜单行，而后面都跟着其在此操作系统文档中的参考位置。

注意：正在执行一个真正的 UNIX 命令时，是不能够读取“帮助工具”的。例如当选择 Dirs / Files → Edit，就进入了 vi，这时 sysadmsh 的各键就失效了。当退出该命令回到 sysadmsh，各键的功能就又恢复了。如果在屏幕上看不到 sysadmsh 的一点痕迹（菜单行、框、上下文指示等），那么 Help 就可能失去作用了。如果需要帮助，就得从当前进程中退出，再按 <F1> 键来读 Help。总之，最好是先读 Help，再执行菜单选项。

## 2.3 Sysadmsh 的其它功能

### 2.3.1 在 sysadmsh 中改变当前目录

为了使用某些文件和命令，就经常必须发出当前目录的情况。可以按 <F6> 键来改变当前目录。当前目录名在屏幕的顶端显示出来，用 <BKSP> 键删除当前目录名（再重写一个），也可以对当前目录名进行部分的增加或修改。按回车键后，就进行目录修改，并在状态行反映出来。

### 2.3.2 功能键

功能键提供了许多省时的特点，请看表 2.8:

表 2.8 功能键

键	作用
<F1> 帮助键	在使用中显示当前文本的帮助信息。 再按一次 <F1> 还可获得进一步的信息。
<F2> 退出键	激活菜单上的 Quit 选项，再按 n 键，就回到 sysadmsh 中。
<F3> 弹出键	（在表格（form）中使用），显示一系列供填当前域的选择内容。
<F4> 拼写键	（在表格（form）中使用），给出填在当前域内的单词的几种可能正确的拼写形式，在其中选择一个并按回车键，这时这个词就填到了当前域中。
<F5> 搜索键	（在表格（form）中使用）给出等待输入串的提示符，当输入一个字符串并按回车键后，高亮度显示就移到在所有项目中与输入串相匹配的那个上。没有相匹配的，这次搜索就失败了，高亮度显示的位置也不移动。另外，分号(;)表示继续向前找，冒号(:)表示向后找。
<F6> 改变目录键	为改换当前工作目录提供机会。请注意，这并不会改变从 sysadmsh 返回后所回到到的目录。
<F7> 打印键	把在一个搜索窗口（scan window）中所显示的某一个命令的执行结果打印出来。

### 2.3.3 使用 shell 逃逸键得到一个 UNIX 命令行

在 sysadmsh 中也可以直接执行一个 UNIX 命令，方法是在 sysadmsh 的任意一个菜单中按一下 shell 逃逸键——一个感叹号(!)。这时菜单就被一个子 shell 代替，一个文本输入行和一个等待命令输入的提示符出现了。当输入命令并按回车键后，这个命令就由 shell 执行。命令执行结束后，结果在屏幕上显示出来，并提示按任意键回 shell。

注意：一个 UNIX 命令行只能从 shell 菜单中得到，而不能从表格 (form) 或“点菜”项目表中得到。

### 2.3.4 sysadmsh 的环境变量

sysadmsh 目前使用下列环境变量，这些变量是在用户的 .login 或 .profile 文件中给定的：

- SA\_\_EDITOR 如果没有设置，当安装了 lyrix，则缺省编辑器就是 lyrix，当 lyrix 没安装，缺省编辑器就是 vi。
- SA\_\_MALL 如果没有设置，当安装了 SCO Port-folio email，缺省通信命令就是 email，当它没安装，缺省通信命令就是普通 UNIX 的 mail
- SA\_\_PRINT 如果没有设置，缺省打印机设备就是 /dev/lp。

### 2.3.5 sysadmsh 菜单选项

在表 2.9 中列出了 sysadmsh 各菜单中的所有选项。主菜单的选项写在表的最左边。在主菜单选项的下边是各级子菜单的选项。带一个箭头的是第一级子菜单的选项，带两个箭头的是第二级子菜单的，依此类推。例如，在表的开关，Report 是 System 子菜单这个第一级子菜单的一个选项，而 Activity 是 Report 子菜单这个第二级子菜单中的一个选项。

一旦熟悉了各级菜单的选项，就可以通过速记或菜单路径名来直接到达某一级子菜单，而不必看所有的高层菜单。在表 2.9 中的 path 这一列给出了各菜单选项的速记或菜单路径名。这种路径名是由为了到达某一级子菜单而所经过的所有菜单选项的第一个字母组成的。例如，要选 Activity 选项，只要输入 system、report 和 activity 这三个单词的头一个字母，即 sra 就可以了，并且不必按回车键。如果在所选中的菜单选项下再也没有其它菜单选项，就会出现一张表格 (form) 或显示某种结果。例如当选了 Activity 这一项，就会出现一个上卷显示或一个搜索窗口，显示出系统中的各进程。

表 2.9 sysadmsh 菜单结构图

选项	路径	功能
System	s	系统全局的报告和配置情况
→ Report	sr	报告系统的当前状态
→ → Activity	sra	报告系统的当前活动
→ → Users	sru	报告当前已注册的用户

(续表)

选项	路径	功能
→→Printers	srp	报告打印机的当前状态
→→Disk	srđ	报告当前磁盘的使用情况
→→Network	srn	报告当前的网络文件
→→→Xnet	srnx	报告 XENIX-NET 的状态
→→→Micnet	srnm	报告 Micnet 的状态
→→→UUCP	srnu	报告 UUCP 的状态
→→Messages	srn	阅读系统信息
→→Software	srš	检查已安装的软件(包)的状态
→Configure	sc	配置系统文件
→→Security	scs	修改系统安全级
→→→Relax	scsr	把安全级恢复为缺省的安全级
→→Kernel	sck	修改核心参数或性能
→→→Parameters	sckp	配置可调核心参数
→→→Rebuild	sckr	按照当前的设置重新连接核心
→→→DOS	sckd	给核心增加 DOS 文件系统的支持
→→→Streams	scks	给核心增加数据流的支持
→→→Layers	sckl	给核心增加 shell 层次支持
→→Logout	scl	设置用户注销之前的空闲时间
→→Defaults	scđ	修改系统缺省参数
→→→Home	Scdh	修改缺省的用户主目录
→→→Message	scđm	修改工作日文件的信息
→→→Checklist	scđc	列出在启动时所检查的文件系统
→→→Other	scđo	修改其它缺省文件
→→International	sci	配置国际性使用的系统
→→→System	scis	设置系统局部变量
→→→Individual	scii	修改某个用户的缺省地点
→→→Display	scid	修改终端字符集的映象
→→→Keyboard	scik	修改控制台键盘的映象
→→Network	scn	配置网络文件
→→→UUCP	scnu	配置 UUCP 文件, 使能/禁止 tty
→→Time	sct	设置系统时间

(续表)

选项	路径	功能
→Hardware	sh	向系统中加入或从系统中移去硬件
→HardDisk	shh	向系统中增加一个硬盘
→Tape	sht	向系统中增加或从系统中移去磁带机
→Printer	shp	向系统中增加或从系统中移去打印机
→Card_Serial	shs	向系统中增加一个串行卡
→Mouse	shm	向系统中增加鼠标器
→Video	shv	配置视频卡的图形参数
→Software	ss	向系统中增加或从系统中移去软件
→Audit	sa	管理 / 检查系统的审计数据
→Enable	sae	使用已存在的参数文件使能审计
→Disable	sad	禁止审计 (停止收集进程数据)
→Collection	sac	显示或修改审计收集规则
→Directories	sacd	显示或修改审计目录表
→List	sacd1	显示审计目录表
→Create	sacdc	建立新的审计目录
→Delete	sacdd	删除现存的审计目录
→Add	sacda	向审计目录表中增加项目
→Remove	sacdr	从审计目录表中移去项目
→Events	sace	显示 / 修改审计事件表
→View	sacev	显示系统审计收集掩码
→Modify	sacem	修改系统审计收集掩码
→IDs	saci	显示 / 修改用户 / 组审计表
→View	saciv	显示用户和组审计表
→Modify	sacim	修改用户和组审计表
→Parameters	sacp	显示或修改审计参数
→View	sacpv	显示审计参数
→Modify	sacpm	修改审计参数
→Reset	sacr	修改缺省值的收集规则
→Statistics	sacs	显示当前审计阶段的统计结果
→Report	sar	报告所保存的审计阶段数据
→List	sarl	列出所有的报告模板

(续表)

选项	路径	功能
→→→View	sarv	查看某个报告模板中的参数
→→→Create	sarc	建立一个新的报告模板
→→→Modify	sarm	修改一个现存的报告模板
→→→Delete	sard	删除一个现存的报告模板
→→→Generate	sarg	生成一个审计阶段的报告
→→Files	saf	操作审计阶段文件
→→→List	safl	列出系统上的审计阶段文件
→→→Backup	saftb	把审计阶段文件备份到可移动介质上
→→→Delete	saftd	删除审计阶段文件
→→→Restore	saft	从可移动介质上恢复审计文件
→Execute	se	执行系统专用程序
→Terminate	st	关闭系统以关机或重新引导
Backups	b	进行文件备份
→Create	bc	建立备份
→→Scheduled	bcs	进行例行的文件系统备份
→→→Unscheduled	bcu	进行非例行的文件系统备份
→Restore	br	恢复文件系统和文件
→→Partial	brp	恢复指定的目录和文件
→→Full	brf	恢复整个文件系统
→Schedule	bs	修改例行备份的频率
→View	bc	查看备份的内容
→Integrity	bi	检查一个备份的完整性
Accounts	a	控制用户帐号的功能
→User	au	改变 / 建立 / 废除用户帐号
→→Examine	aue	设置口令 / ID / 权限 / 审计参数
→→Create	auc	建立一个新的用户帐号
→→Retire	aur	关闭一个现存的帐号
→Defaults	ad	设置系统全局 (缺省) 参数
→→Authorization	ada	查看 / 修改缺省核心 / 子系统权限
→→Password	adp	查看 / 修改缺省口令生命期 / 选择
→→Logins	adl	查看 / 修改缺省注册控制

(续表)

选项	路径	功能
→Terminal	at	管理终端数据库登记项
→→Examine	ate	查看 / 现存的终端登记项
→→Create	atc	建立一个新的终端登记项
→→Delete	atd	删除一个现存的终端登记项
→→Lock	atl	封锁一个特定的终端
→→Unlock	atu	清除一个特定终端上的所有封锁
→→Assign	ata	管理设备名等价数据库
→→→Examine	atae	查看 / 修改现存的设备登记项
→→→Create	atac	建立一个新的设备登记项
→→→Delete	atad	删除一个现存的设备登记项
→Report	ar	报告口令 / 终端 / 注册活动
→→Password	arp	报告帐号的口令状态
→→→Impending	arpi	报告口令将要退役的帐号
→→→Expired	arpe	报告口令已经退役的帐号
→→→Dead	arpd	报告口令已经死亡的帐号
→→→User	arpu	报告单个用户的口令
→→→Group	arpg	报告单个用户组的口令
→→→Full	arpf	列出口令数据库中的所有登记项
→→Terminal	art	报告终端的访问状态
→→Login	arl	报告用户 / 组 / 终端的注册活动
→→→User	arlu	报告一个、一系列、或所有用户的注册情况
→→→Group	arlg	报告一个或一系列用户组的注册情况
→→→Terminal	arlt	报告一个 / 一系列 / 所有终端的注册情况
→Check	ac	检查 tcb 文件的内容来找错
→→Datebases	acd	检查子系统数据库的一致性
→→Password	acp	检查 / etc / passwd 和 / etc / group
Printers	P	管理打印系统
→Configure	pc	为打印服务配置打印机
→→Add	pca	向系统中增加打印机
→→Modify	pcm	修改打印机配置



(续表)

选项	路径	功能
→→Remove	pcr	从打印服务中移走打印机接收地
→→Default	pcd	修改系统缺省目标打印机
→→Parameters	pcm	修改打印机控制和参数
→→Errors	pce	设置错误警告报告/恢复模式
→→Content	pcc	指定在打印机上可打印内容的类型
→→Users	pcu	指定谁能够使用打印机
→Schedule	ps	启动/停止打印服务, 处理请求
→→Begin	psb	启动打印服务
→→Stop	pss	关闭打印服务
→→Accept	psa	允许请求使用目标打印机
→→Reject	psr	拒绝请求使用目标打印机
→→Enable	pse	开放打印机
→→Disable	psd	禁止打印机
→Request	pr	移动或取消打印服务请求
→→Move	prm	在目标打印机之间移动请求
→→Cancel	prc	取消对打印服务的请求
→Auxiliary	pa	处理打印轮/过滤器/预打印格式
→→Alert	paa	设置/列出打印轮报警信号
→→Filter	paf	管理用于打印服务的过滤器
→→→Change/ Add	pafc	增加或修改用于打印服务的过滤器
→→→Remove	pafr	从打印服务中移去过滤器
→→→List	pafl	列出一个过滤器的描述
→→→Original	pafo	恢复原始的过滤器描述
→→PPforms	pap	管理打印服务的预打印格式
→→→Configure	papc	为预打印格式修改打印机设置
→→→Modify	papm	增加或修改预打印格式
→→→Remove	papr	从打印服务中取消预打印格式
→→→List	papl	列出现存格式的属性
→→→Users	papu	允许或拒绝用户访问一个格式
→→→Alerts	papa	修改已安装格式的报警方法
→→→→Specify	papas	指定报警方法

(续表)

选项	路径	功能
→→→→List	papal	列出当前的报警信息
→→→→Terminate	papat	中断现存的活动报警
→→→→Remove	papar	删除报警定义
→Priorities	pp	设置打印队列优先级
→→Default	ppd	设置系统全局性优先级缺省值
→→Highest	pph	为用户设置缺省的最高优先级
→→Remove	ppr	从指定的优先级中删除用户
→→List	ppl	列出缺省优先级和限制
Media	m	对软盘/磁带进行读、拷贝、比较、格式化等操作
→List	ml	列出软盘或磁带的內容
→Extract	me	选取软盘或磁带的內容
→Archive	ma	把文件/目录/文件系统保存到介质上
→Format	mf	格式化一张 UNIX 或 DOS 软盘
→Duplicate	md	建立软盘或磁带的另一个副本
→Tapedump	mt	显示磁带的物理內容
Jobs	j	查看/控制进程
→Report	jr	报告当前进程(快照)
→Terminate	jt	中断当前运行的进程(kill)
→Authorize	ja	给用户授权来运行作业
→→Scheduled	jas	授权建立正常的调度作业
→→→Default	jasd	为调度作业设置缺省权限
→→→User	jasu	允许/禁止用户调度作业
→→→View	jasv	检查谁能够创建调度作业
→→Delayed	jad	授权用户创建延期的作业
→→→Default	jadd	为延期作业设置缺省权限
→→→User	jadu	允许/禁止用户建立延期作业
→→→View	jadv	检查有谁被授权建立延期作业
→→Environment	jae	修改延期作业环境
→→→At	jaea	为正被控制的作业修改环境
→→→Batch	jaeb	为 batch (c) 作业修改环境
Dirs / Files	d	对文件和目录进行操作

(续表)

选项	路径	功能
→List	dl	列出在当前目录中的文件
→View	dv	查看文件的内容
→Copy	dc	拷贝目录或文件
→Edit	de	编辑一个或多个文件
→Modify	dm	修改文件参数
→→Permissions	dmp	修改文件操作权限
→→Ownership	dmo	修改文件所有权
→→Group	dmg	修改文件组所有权
→→Name	dmn	重新命名或移动文件
→→Size	dms	压缩文件
→→Format	dmf	修改文件格式
→Print	dp	打印文件
→Archive	da	保存文件
→Differences	dd	比较两个文本文件或目录
→Remove	dr	删除特定的文件或目录
→UseDOS	du	使用 DOS 工具对 DOS 文件进行操作
→→List	dul	列出当前目录中的 DOS 文件
→→Remove	dur	删除一个 DOS 文件或目录
→→MakeDir	dum	建立一个 DOS 目录
→→Copy	duc	在 DOS 和 UNIX 之间拷贝文件
→→View	duv	显示 DOS 文件
→→Format	duf	格式化 DOS 介质
Filesystems	f	对文件系统进行检查 / 操作
→Check	fc	检查和修理文件系统 (fsck)
→Mount	fm	安装文件系统
→Unmount	fu	拆卸已安装的文件系统
→Add	fa	给一个新文件系统增加合适的信息
→Floppy	ff	在软盘上建立文件系统
→DOS	fd	为 DOS 文件系统增加支持
Quit	q	退出系统管理程序
→Yes	qy	离开 sysadmsh
→No	qn	取消退出命令

## 第三章 用户帐号的管理

用户帐号用来跟踪系统使用者的使用情况并控制他们对系统资源的存取。每一个帐号都包含一个不同的“注册名”(login name) 和一个“口令”(password), 供用户进入系统使用, 还有一个“主目录”(home directory), 这是供用户工作的。另外, 系统还有某些缺省值, 用来规定口令的长度, 是否允许用户自选口令以及最多允许用户不成功注册的次数。

为所有的用户建立帐号并在需要的时候对口令进行修改, 修改注册小组及其它帐号参数等维护工作, 这些都是系统管理员的工作。

必须确定, 系统是需要几个管理员分别负责系统几方面的操作, 还是只需要一个管理员, 进行所有的维护工作。(不作任何规定, 在根目录下工作就具备超级用户的能力。)

本章将介绍如下几方面的内容:

- 帐号管理: 增加、修改和删除用户帐号, 另外建立新的用户组。
- 帐号缺省设定: 设定系统的用户注册符和口令缺省值。
- 终端注册管理: 根据各终端特性, 调节在各终端上注册的过程。
- 生成报告: 产生用户注册, 终端的使用和口令等状态的报告。

在建立用户帐号后立即检查帐号的缺省设置是很重要的。这些都已在本章的“缺省帐号设置”中叙述。系统管理员必须检查一下这些缺省设置对系统的需求来说是否合适。

注意, 在任何情况下, 都不要用文本编辑器对 `/etc/passwd` 进行编辑。在其它的 UNIX 系统中这样做是很正常的, 但这是一种不可靠的增加用户的方法。这样做的结果是, 在屏幕上显示错误信息, 并且还会使系统不再接受更多的用户注册。要修改或增加用户帐号, 请用 `sysadmsh` 中的 `Accounts` 选项。 `/etc/passwd` 被存放在一个附加的受保护 (protected Password) 数据库中, 这个库中存有各用户口令的加密形式以及其它保密参数。

### 3.1 帐号管理

这一节介绍如何建立和管理用户帐号。

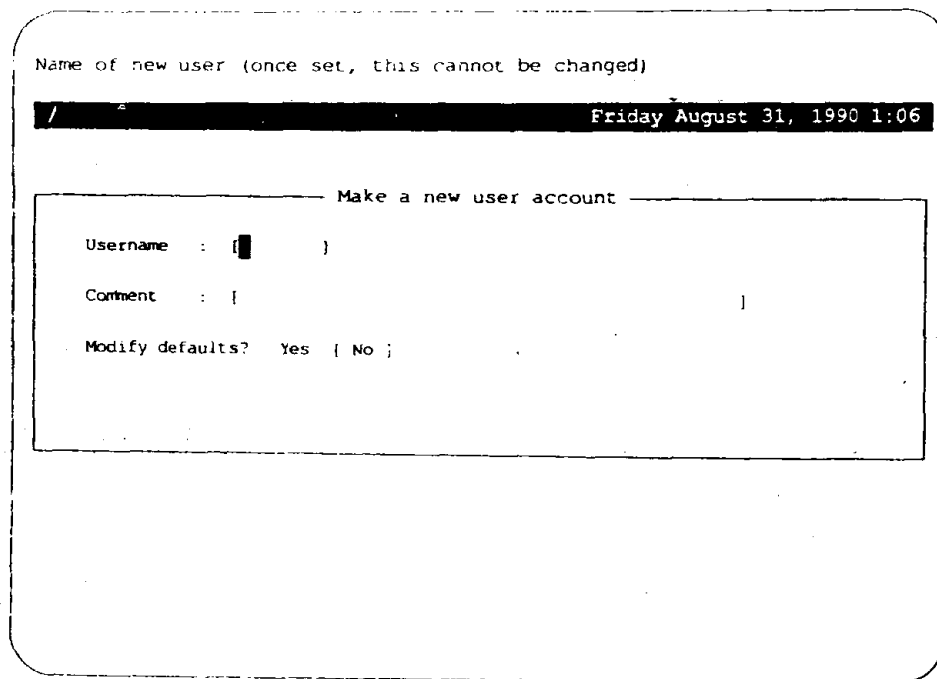
#### 3.1.1 增加用户

用 `sysadmsh` 可以在系统中增加新的用户帐号, 同时在帐号数据库中增加一条新记录。在这个数据库中包含有新用户的有关信息 (如注册名和初始口令), 系统凭此信息允许用户注册并开始工作。 `sysadmsh` 还为用户建立一个主目录, 用 `mail` 命令为用户建立一个信箱, 还要建立一个初始化文件 (例如, 对 Bourne 的 shell 来说就是 `.profile`, 对 C-shell 来说就是 `.login`), 在这个文件中包含有在用户一注册就立即执行的 UNIX 命令。

在建立一个用户帐号时, 请做如下 `sysadmsh` 选择:

Accounts → User → Create

这时的屏幕显示如下:



增加用户步骤如下:

1. 填写“username”域: 如需要, 还要填“comment”域。
2. 如果想修改缺省设置, 就选择“**Yes**”, 并且给定在“修改用户帐号缺省设置”一节中所列出的各域值。按需要填写各域, 也可以按<F3>后从所列出的各供选值中选出所需值。当按了回车键之后, 所选的值就填入到该域中。
3. 当从表格中退出时, 就会弹出一个窗口, 要求对所填的内容给予确认。如果确认了, 就显示如下信息:

```
Created home directory: pathname
Created shell file: filename
Greeting mail sent to user: name
```

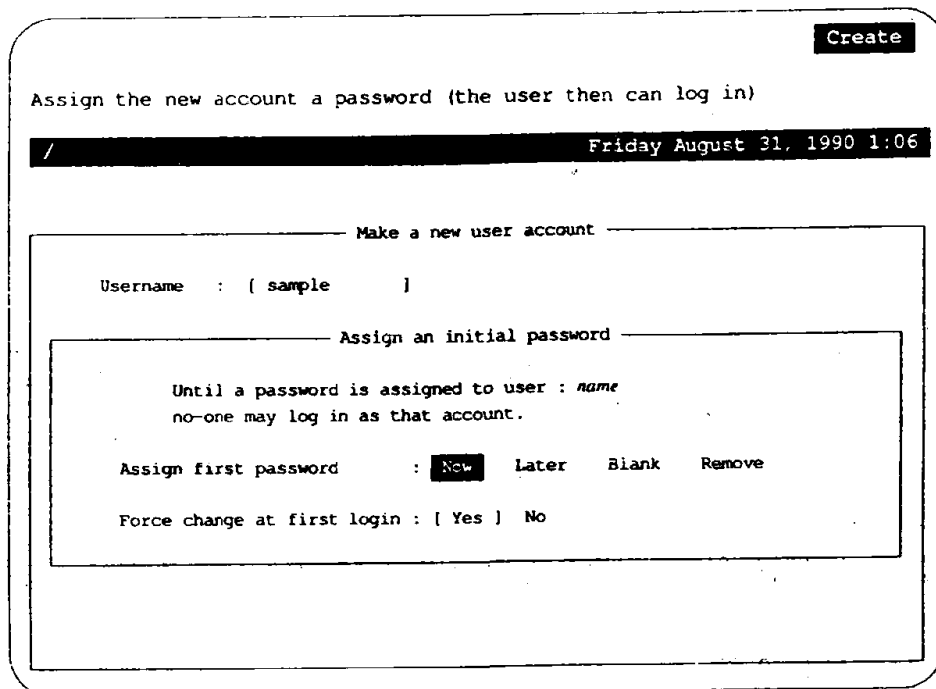
这表示所有所需的文件和目录都建立了 (这个缺省信息是从 / use / bin / mkuser 中取得的)。

注意, 如果看到了类似如下的信息:

```
useshell: file access control database inconsistency
useshell: not authorized
```

那么这表示数据库文件的权限或内部的一致性出了问题。这时应该执行 integrity 命令, 这在“系统安全性维护”一章中的“系统文件的完整性检查: integrity”有详细说明。而 integrity 执行后所给出的信息应该指出问题的原因。

4. 最后, sysadmsh 提示是否要建立一个原始口令, 此时会显示下面的菜单:



对“Assign first password”可以有如下几种可能的选择：

Now 给新帐号赋一个口令。

Later 不给新帐号赋口令（用户不能注册）。

Blank 给新帐号赋一个空口令（用户必须在第一次注册后设置一个口令）。

Remove 根本不给新帐号设口令（用户不必通过口令就可以注册）。

“Force change at first login”规定用户是否必须在第一次注册后就修改口令。请注意如果把这项设为 Yes，就会否定 Remove 的结果，这就是说，用户必须在第一次注册后就设置一个口令。

5. 如果选择了为新用户建立一个口令，就会看到如下提示：

Last successful password change for user: date

Last unsuccessful password change for user: NEVER

choose password

you can choose whether you pick your own password,

or have the system create one for you.

1. pick your own password

2. pronounceable password will be generated for you

Enter choice (default is 1):

6. 如果选择1，就会要求输入口令两次：

please enter new password (at least 5 characters):

please choose a password which contains a mixture of lower- and upper-case letters, digits (0-9), and non-alphanumeric characters (e.g., !, #, @, :, %, or /.)

please do NOT choose a password that is an English word, of which is the name of a person, place, or thing, or which

contains the string “SCC”, “XENIX”, or “UNIX” (in either case)

Re-enter password:

请注意，在输入口令时，它不会在屏幕上显示出来。如果所输入的口令不被接受，

就会被要求输入另外的口令。

7. 如果选择2，就会显示如下：

```
Gencrating random pronounceable password for user.
The password, along wity the hyphenated version, is shown.
Hit <RETURN> or <ENTER> until you like the choice.
When you have chosen the password you want, type it in.
Note: Type your interrupt character or 'quit' to abort at any time.
Password: xxxxxxxx Hyphenation: xx-xx-xx Enter password:
```

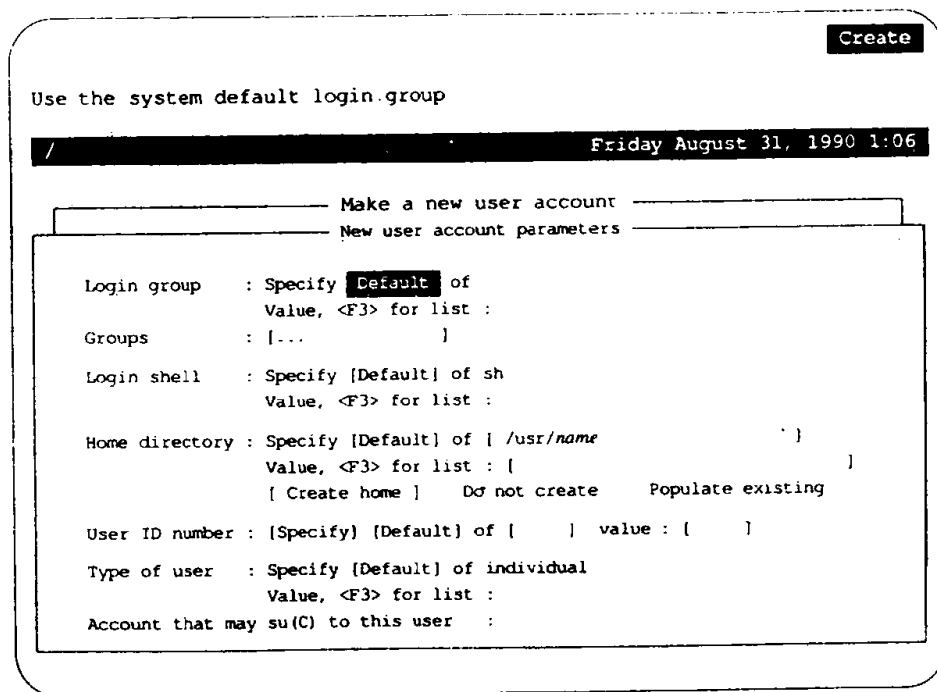
所产生的口令以一种带连接符的形式显示，连接符把口令分割成几个可发音的音节，以帮助对这个口令的记忆。

8. 把新口令给用户。如果选择了强制修改口令，那么用户就必须在第一次注册后马上修改口令。

这时，这个新帐号就可用了，除非给用户规定了特殊的值，否则就一直缺省的安全参数对其进行维护。

### 修改用户帐号缺省设置

对大多数用户来说，这种缺省设置是足够的。如果选择“Modify Defaults”，就会出现下面的表格：



光标开始处于“Login group”域。上面表格中的某些域只能在帐号建立时修改——而在修改（Modify）状态下，这些域只能作为信息显示出来，而其值却不能修改。各域定义如下：

- Login group（注册组）

当一个用户登记到系统中，这个组就与这个帐号发生了联系。这个域是可以修改的，但不能为空。这个域中的值还被填到 / etc / passwd 文件中的此用户记录的 group 域。按 <F3> 键后，当前所存在的所有小组的组名就会列出来供选择。

- Groups (组)

本用户所在的所有的组。

- Login shell (注册进入的 shell)

本用户将使用的shell。(缺省值由 / etc / default / authsh规定) 如果其值是一个完整的路径名 (譬如“ / bin / sh”), 则由这个路径名所说明的 shell 就是用户注册后所处的 shell。然而, 如果所用来说明 shell 的不是一个路径名 (譬如“sh”), 那么它就以一个预定义过的 shell 的名字作为名字, 这些曾定义过的 shell 名都记录在一个叫 / usr / lib / mkuser 的子目录中。选择一个预定义过的 shell 就在这个帐号建立时把一些与 shell 相关的文件 (如“sh”的 . profile 文件) 拷贝到该用户的主目录中。(有关建立自己的预定义 shell 的内容, 请看本章后面的“增加预定义 shell”。)

- Home directory (主目录)

定义用户文件所驻留的位置。缺省目录选项是高亮度显示时, 按回车键就会得到一个主目录的缺省位置。主目录选项如下:

Create 为该用户建立一个新的目录。

Do not create 不为该用户建立新目录。

Populate existing 以指定的已经存在的目录作用该用户的目录。

如果要让许多用户同用一个主目录, 具体指令请看下面的“共享主目录”。

- User ID (用户标识符)

用户标识符。一旦设立, 这个用户标识符就不可以再改变, 否则会破坏审计跟踪。

- Type of user (用户类型)

在多数情况下, 这个域值为“individual” (个人) 或“pseudo-user” (假用户)。如果缺省, 则被设定为“individual”。个人用户都是带有真实姓名的人, 而假用户都是匿名帐号, 例如 mmdf。但每一个假用户帐号都必须有一个对该帐号负责的人。

- Account that may su (c) to this user

对本帐号负责的人。当且仅当该用户类型不是“individual”时, 本域才可以修改。对于“individual”型的用户, 本域为空, 而对于非“individual”型的用户, 本域必须填另外一个帐号 (资源尚未用尽) 的用户名。例如, 根 (root) 帐号必须有一个对此帐号负责的人的名字。这就保证了没有任何帐号是匿名的: 每一个帐号都必须对应一个真正的人。按 <F3> 就会列出系统中所有的用户, 供选择。

### 共享主目录

还可以建立共享同一个注册目录的用户帐号。这一般都是在增加新用户目录时做的。

然后就可以退出 sysadmsh 并输入下列命令 (用真实的目录名来替换 homedir):

```
cd homedir
```

```
chmod 775
```

```
chown auth
```

另外, 根据帐号所使用的注册 shell, 输入下列命令之一:

Bourne 或 Korn shell

```
chmod 660.profile
```

```
chmod 660.kshrc (korn shell only)
```



C-shell

```
chmod 660.login .cshrc
```

这就保证了在同一注册的成员都可以共享这个目录。注意，如果给用户已经赋予了另一个注册组名，他们就不能共享这个目录了。要增加一个共享目录的用户，就必须在增加用户的表格中选“Modify Defaults”，在缺省值修改表格的 Home directory 一项选 specify 和 “populate existing”；在回答是否重写如.profile 等文件时，回答 n。

### 增加管理用户

除了标准的标识信息以外，那些作为管理员来管理打印机、帐号等的用户，可以通过选择特权而被赋予相应的职责。详见本章后面的“改变权限缺省”。

### 所显示的用户帐号缺省值的修改

通过编辑 /etc/default/authsh 文件可以修改用户生成菜单中的缺省选择。下面的缺省项可以重新定义：

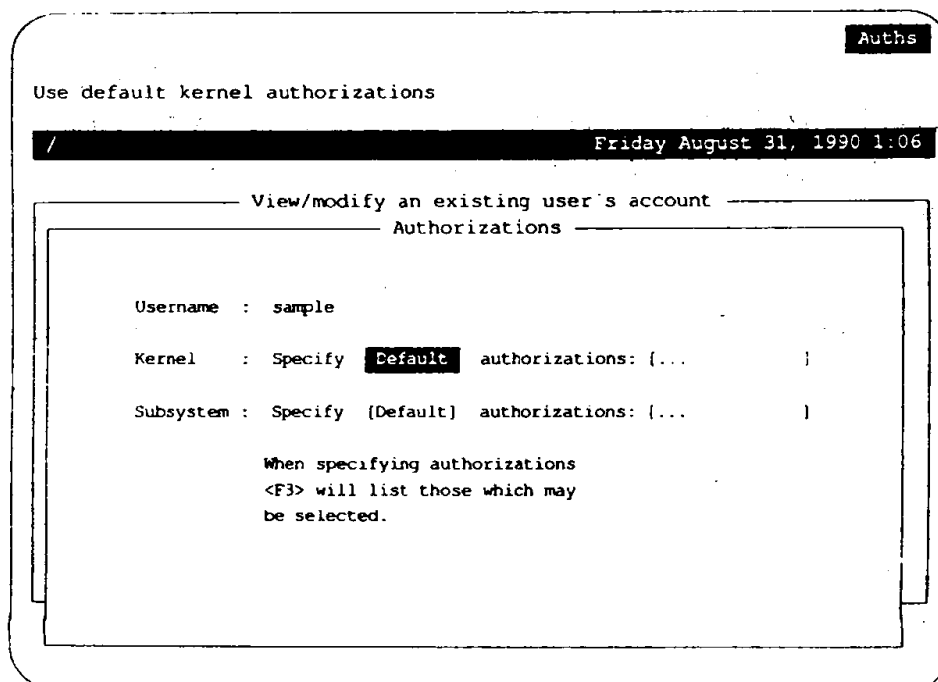
- login group
- groups
- login shell
- home directory
- range of user IDs
- type of user

### 3.1.2 修改或分配用户授权

授权仅赋予那些能够可靠地对某一个子系统进行管理的用户。要给一个用户赋予一个新的授权，请在 sysadmsh 中进行如下选择：

Accounts→User→Examine: Privileges

(这个冒号表示在选择 Privileges 之前，必须先填一个域 (这里是用户名)) 这时就显示出如下表格：



可以选择 Specify, 并按 <F3> 产生一个窗口, 其中列出了所有的授权。

注意, 如果从 default 切换到 specify, 那个用户的缺省值就被取消, 而只有那些所指定的授权有效。

### 3.1.3 删除用户帐号

从严格的意义上讲, 一个用户是永远不能从系统上删除的。一个用户标识符一旦被用过一次, 就不能再用了。只能说, 某一个用户帐号从业务上被删去。要删去一个用户帐号, 在 sysadmsh 上进行如下选择:

Accounts → User → Retire

注意, 一个被删除了的帐号永远不能再使用, 删除是永久的。

删除一个用户帐号并不能同时删去该用户的文件; 系统管理员必须再动手删除文件。

### 3.1.4 对用户帐号进行封锁或解锁

系统管理员可以封锁一个帐号而禁止其使用。另外, 如果某一个用户帐号的注册参数已经超支了, 它就会被自动封闭。一旦一个用户或终端被封闭了, 只有系统管理员能够对其启封。要封锁或解锁一个帐号, 在 sysadmsh 中做如下选择:

Accounts → User → Examine: logins

(这个冒号表示在选择 Logins 之前, 必须先填一个域 (这里是用户名))。这时类似下面的表格显示出来。

把光标下移至 lock status 这个域, 然后按要求把它定在 Apply Administrative lock 上或 Clear All Lock 上。

The screenshot shows a terminal window titled 'Logins'. At the top, it says 'Use the system default limit on unsuccessful login attempts'. Below that is a date and time stamp: 'Friday August 31, 1990 1:06'. The main content is a window titled 'View/Modify an existing user's account' with a sub-header 'Login history and locks'. It displays the following information for user 'sample':

Last login attempt	Location	Date/time
successful	tty01	Thu 3 May 1990 08:22:06 AM
unsuccessful	tty2b	Mon 7 May 1990 02:12:39 AM
Last logout	tty2b	Mon 7 May 1990 03:19:24 AM

Below the table, it shows: 'Number of unsuccessful login attempts since last successful login: 1' and 'Maximum number of unsuccessful attempts before account is locked: Specify Default of 6 Value:'. At the bottom, it says 'Account locked: NO LOCKS' and 'Lock status [No change] Apply administrative lock Clear all locks'.

### 3.1.5 修改用户的组 (group)

要修改用户的注册组, 在 sysadmsh 中进行如下选择:

Accounts → User → Examine: Identity

(这里的冒号表示在选择 Identity 之前要先填一个域 (这里是用户名)) 这时就会显示类似下面的一张表格:

```
Identity
Group associated with this account when the user logs in (<F3> for list)
/ Friday August 31, 1990 1:06
View/Modify an existing user's account
Identity
Username      : sample
User ID       : 246                Type of user : individual
Account that may su(C) to this user : NONE
Login group   : [ pub          ]
Groups        : [ ...          ]
Login shell   : [ /bin/sh      ]
Home directory : [ Keep ] Edit  Create  Move  Restore
              path : /usr/sample
Comment       : [
Priority       : Specify [Default] of 0 Value:
```

然后按需要修改 Login group 域。

### 3.1.6 修改用户的主目录 (home directory)

要修改一个用户的主目录,包括移动该用户的文件,请在 sysadmsh 中进行如下选择:

Accounts → User → Examine: Identity

(这里冒号表示在选择 Identity 之前要先填一个域 (这里是用户名)) 这时一个类似下面的表格就显示出来:

```
Identity
Group associated with this account when the user logs in (<F3> for list)
/ Friday August 31, 1990 1:06
View/Modify an existing user's account
Identity
Username      : sample
User ID       : 246                Type of user : individual
Account that may su(C) to this user : NONE
Login group   : [ pub          ]
Groups        : [ ...          ]
Login shell   : [ /bin/sh      ]
Home directory : [ Keep ] Edit  Create  Move  Restore
              path : /usr/sample
Comment       : [
Priority       : Specify [Default] of 0 Value:
```

在把光标向下移到 Home directory 这个域后，就可以选择做下面的工作：

- keep        不改变这个用户的主目录
- Edit        修改这个用户主目录的路径名，但并不移动任何文件。
- Create      为这个用户建立一个主目录。
- Move        为这个用户的主目录改名，并把旧的主目录下的文件全移至新目录下。
- Restore     把路径名改回为原来的值（文件不移动）。

### 3.1.7 修改用户口令或口令参数

系统管理员可以在任何时候改变用户的口令。口令生成参数也可以针对个别用户进行修改，就像针对全系统进行修改一样，以控制如何修改用户口令。修改口令，请在 sysadmsh 中做如下选择：

Accounts→User→Examine: password

（这里的冒号表示在选择 password 之前先填写一个表格域（在这里是用户名域））之后，下面的表格就会显示出来：

口令选取参数与在本章后面的“改变缺省的口令限制”中所描述的全系统的参数是相似的。下面的参数定义了每个用户的口令限制：

Password required

The screenshot shows a terminal window titled "Passwords". At the top, it says "Use the system default maximum password length for this user". Below that is a date and time stamp: "Friday August 31, 1990 1:06". The main menu is "View/Modify an existing user's account" with a sub-menu "Password selection". The options are:

Username	:	sample			
Password required	:	Yes	No	Default	of Yes
User can choose own	:	Yes	No	[Default]	of Yes
User can run generator	:	Yes	No	[Default]	of Yes
Maximum generated length	:	Specify	[Default]	of 10	Value :
Checked for obviousness	:	Yes	No	[Default]	of No
Current password status	:	[ Keep ]	Change	Disable	Remove
Change password at login	:	Yes	[ No ]		

若答Yes,则用户帐号可不需口令；若答No,则这个帐号必须有口令才能允许注册。

User can choose own

这个参数规定了用户自己是否可以选择口令。一个可靠的（trusted）系统要求系统必须自动地为用户生成口令。这就防止用户选取一些明显的（obvious）口令，所谓明显的口令就是指一些有见识的入侵者能够根据该用户的个人的特征而猜出来的口令。而其它的 UNIX 系统是允许用户去选择他们自己的口令的。这个参数如果置为 Yes，那个非可靠 UNIX 系统的规则就生效了，即允许用户自己选取口令；如果该参

数置为 No, 则系统就会通过一个随机口令生成过程给用户生成口令。

User can run generator

这个参数允许用户运行口令生成程序。请注意, 这个参数不允许用户选取口令, 只允许生成一个新的随机口令。

Maximum generated password length

口令的最大长度是由系统规定的。这个最大长度是80。

Checked for obviousness

这个参数指明系统是否要对所产生的口令进行通常的检查。这些检查与命令 goodpw 所进行的检查一起保证了口令不会出现在联机目录中。把这个参数设为 Yes, 这会保证所有想通过输入真实的单词作为口令而渗透进入系统的企图都归于失败, 但是用限制用户和终端注册的手段来进行这种控制会更加有效。实际上, 通常的检查增加了修改一个口令所需的时间。所有这三个参数都可以由用户的参数值来对其进行重写。

Current password status

可以选择下列四项之一:

keep 不修改口令。

change 调用在“增加用户”的末尾所讲述的口令修改过程。

Disable 使口令失效, 其结果是封锁该用户帐号。

Remove 删除口令, 使用户可以不通过口令就注册到系统中。

请注意, 如果最小的口令修改时间间隔还没过完, 即使是超级用户也不能够修改口令。但是可以通过暂时修改 Accounts→User→Examine: Expiration 下的“Minimum number of days between password changes”绕过这一限制。

Change password at login

这个选择可以强制用户在下次注册后修改其口令。

### 修改用户口令的退役参数

有时, 把某一用户的退役参数改为与系统的缺省值不同的值是很有必要的。要做这件事, 在 sysadmsh 中选择:

Accounts→User→Examine: Expiration

(这里的冒号表示在选择 Expiration 之前要先填写一个表格域 (这里是用户名域)) 这时类似下面的表格显示出来:

这里用户参数的描述与本章后面的“改变缺省口令限制”中所描述的全系统的参数是相似的。描述是不同的, 但参数是相同的。所显示的参数控制一个口令有效的时间, 还控制当口令失效时修改口令的过程。口令的生命期可分为三个阶段:

- 口令有效
- 口令已经到截止期, 但用户还可以注册并修改它 (如果授权允许这样做)。
- 口令已经死亡, 帐号被封锁, 系统管理员必须对该帐号进行解锁, 再由用户修改口令。

为了阻止用户在口令将到截止时间时修改口令, 然后再立即改回他们所记得的口令, 系统还规定了在两次口令修改之间的最小时间间隔, 在这个间隔没有超过之前, 用户是不能修改口令的。口令生命期方案是这样实现的:

Minimum number of days between password changes

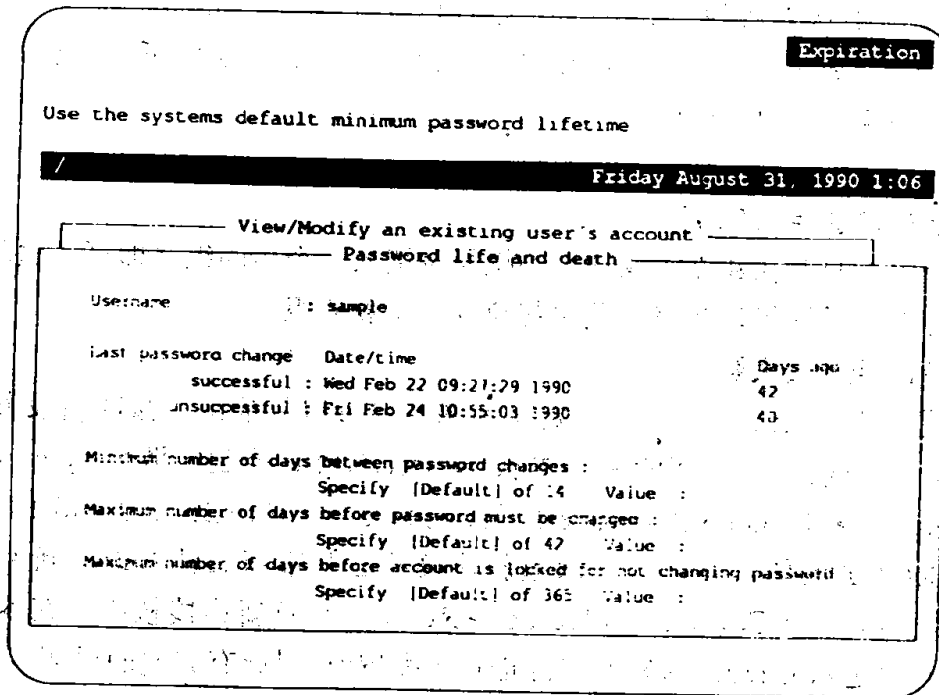
用户在再一次改变口令之前所必须等待的天数。)

Maximum number of days before password must be changed

这规定了一个口令有效的时间长度。

Maximum number of days before account is locked for not.....

这规定了从最后一次改变口令到这个口令死亡之间的时间间隔。

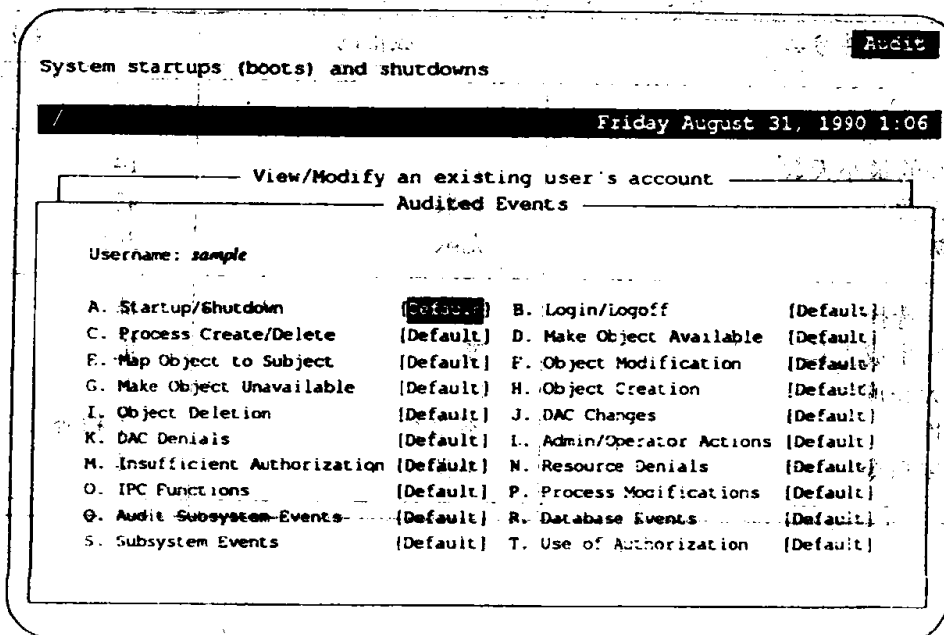


### 3.1.8 改变用户的审计参数

可以像规定整个系统的审计参数那样规定个别用户的审计参数,为用户所设置的任何参数值都可以覆盖系统的缺省值。为了规定或改变审计参数,在 sysadmsh 中做如下选择:

Accounts → User → Examine: Audit

(这里的冒号表示在选择 Audit 之前必须先填一个域 (这里是用户名域)) 这时一个类似下面的表格显示出来:



更加详细的审计项目的说明可以在本书的“使用审计子系统”一章中看到。

存在三种可能的设置：

Default 使用全系统的帐号缺省值。

Always 对本项进行一贯性审计（重写全系统缺省值）。

Never 永远不对本项进行审计（重写全系统缺省值）。

按 <F3> 就会显示出这几种设置供选择，也可以手工填写。简写也是认可的（如“n”“nev”和“N”都表示 Never）。要执行这个表格，只要按 <CTL>x 即可。（如果填完了表格中的最后一项，它就会自动执行）。

### 3.1.9 增加或改变组 (Group)

要增加一个组，只要在建立或改动用户帐号时填入一个新的组名即可。这时系统就会提示这个组不存在，并让管理员确认确实想建立一个新的组。

#### 改变所增补的组的最大数目

增补的组是指在 /etc/group 中所定义的基本设置以外的组。按缺省设置，增补组的最大数目是 8。这个数目由可调的核心参数 NGROUPSP 控制。调用 sysadmsh 的选择 System→Configure→Kernel→Parameters，再选第 3 类“Files, Inodes and Filesystems”，就可改变 NGROUP 的值。这时，必须重新连接核心并重新启动系统，新值才能生效。用 sysadmsh 的 System→Configure→Kernel→Rebuild 来重新连接核心，在本书“调整系统性能”一章中“通过 configure 对核心资源进行重配置”里对此有完整的说明。

## 3.2 帐号的缺省配置

本带介绍如何改换系统的安全性缺省设置，包括缺省口令的模式，子系统的授权和允许用户注册失败的次数。系统事先已经为用户帐号的安全模式做了缺省的预配置。表 3.1 包含了这些缺省配置，还包括放松值（由其它的不很可靠的 UNIX 系统使用），在下一部分“选取放松的安全性缺省设置”对此做了讲述。

表 3.1 系统安全性参数缺省设置

安全性参数	放松的	C2
口令		
两次改变之间的最小天数	0	14
退役时间 (天数)	无限	42
生命期 (天数)	无限	365
用户能否自己选择	能	能
用户能否运行生成工具	能	能
所生成的最大长度	8	10
检查其可见性	不检查	不检查
注册时是否要口令	不要	要

注册		
最多的不成功注册次数 (帐号和终端) 两次注册尝试之间的延时 完成注册的时间	99 0 秒 60 秒	5 2 秒 40 秒
审计项目类型	没有	A、B、F、H、I、J、K、 L、M、N、Q、R、S、T
授权		
子系统	queryspace, printerstat, printqueue, mem, terminal, audittrail, su	queryspace printerstat, printqueue
核心	execsuid, chmodsugid, chown, nopromain	exesuid, chown, nopromaim

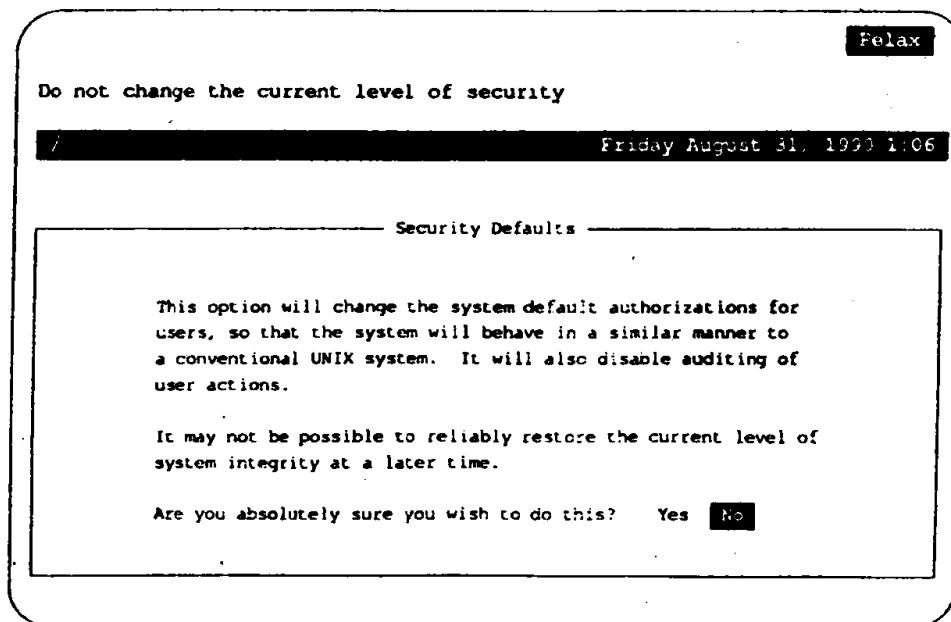
在本书的“系统安全性维护”一章对审计项目的类型有详细说明。

### 3.2.1 选取放松的安全性缺省设置

在系统安装时，安全性缺省设置就选定了，如果选了 C2 的缺省设置，那么它们还是可以改变的，如果愿意把系统降级为类似其它 UNIX 系统的功能，就可以在 sysadmsh 中做如下选择：

System → Configure → Security → Relax

这时就把安全性策略更换为非可靠 UNIX 系统的缺省设置。当做了这一选择之后，就会显示如下警告信息：





如果对安全性缺省设置做了一些改变，就会显示如下的另外警告：

The default file has changed since installation

Previous changes will be lost

Press <Return> to continue, or <Escape> to abort

请注意：还可以个别确定这些缺省设置，甚至可以根据需要把它们改为 C2 安全级所要求的值。

### 3.2.2 选取 C2 安全性缺省设置

在选取了放松的安全性缺省设置后，还可以再恢复为 C2 级安全性缺省设置，虽然这并不意味着系统可以自动地符合一个 C2 安全系统的要求。（按规定，只有在系统安装时选取 C2 级安全性缺省设置才能真正符合要求）要再恢复为 C2 级安全系统，请以表 3.1 为指导，在 sysadmsh 中选择 Accounts → User → Examine，然后再分别选择各项参数，手工重新进行设置。

### 3.2.3 动态地改变安全性参数

下面的系统全局性帐号参数是可以特殊规定的：

- 权限
- 口令
- 注册

系统全局性安全性参数对用户注册的方式进行控制，并且在其建立了一个对话区后，还控制系统赋予的终端和权限环境。所有能够使用 sysadmsh 接口程序改变的参数都在这里讨论，而影响系统操作的其它参数都安排在以后讨论。

应该先使用系统全局性功能特性来作为自己的缺省的系统行为参数，然后再对有不同要求的用户分配特定的功能，来对其行为进行调整。就像所希望的那样，对用户特定的参数设置将覆盖系统的缺省设置。

### 3.2.4 改变缺省的注册限制

系统中许多与建立注册对话区相关的缺省设置都可以重新设置，这些设置包括注册特性、口令的生成和执行方式。注册参数还规定了帐号和终端封锁特征。当用户注册时，他们必须给出注册名和口令。另外，用户不成功注册次数也是有限的，当不成功注册的次数超过这个限度后，帐号或终端就被封锁。这个特性防止了有恶意者通过多次尝试而打入系统的企图。要访问注册限制参数，请做如下选择：

Accounts → Defaults → Logins

这时如下表格显示出来。

其中各参数说明如下：

Maximum number of unsuccessful attempts before locking

这是对用户和终端所规定的不成功尝试的缺省数目，如果某一个特定用户或终端需要更少或更多的数目，就必须修改用户帐号或改变终端配置来覆盖系统的缺省设置。

Delay (in seconds) between login attempts on a terminal

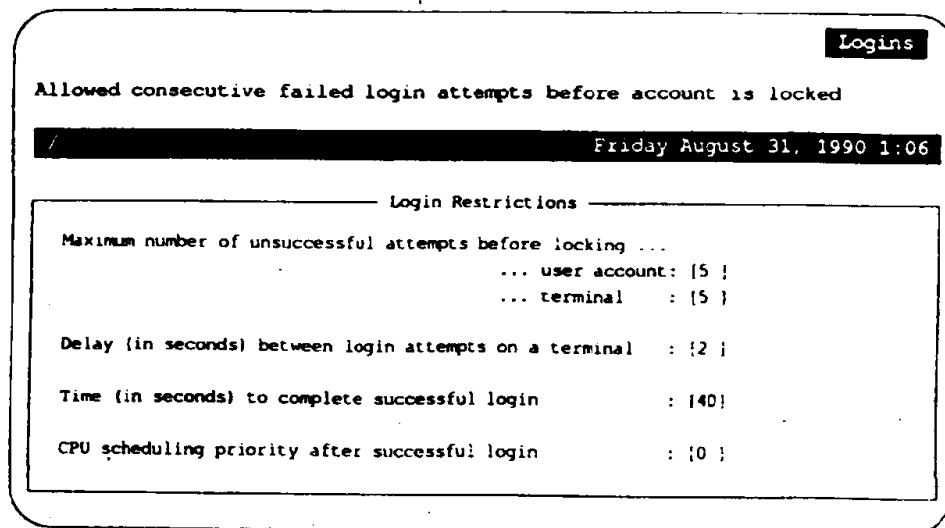
这个参数用来控制在两次不成功的注册尝试之间必须经过的时间。为了进一步减少非法打入系统的可能性，系统还可通过加长这个间隔时间来增加试图打入系统而多次重复所需的总的时。可以通过增大这个参数来控制出现“login:”提示符的周期。依靠本参数与终端的不成功尝试参数的配合,就可以挫败在某些终端上反复试口令的企图。

Time (in seconds) to complete successful login

这个参数规定了在注册完成之前用以输入用户名和口令的时间长度限制。

CPU scheduling priority after successful login

这设定了与用户进程有关的 nice (C) 值。

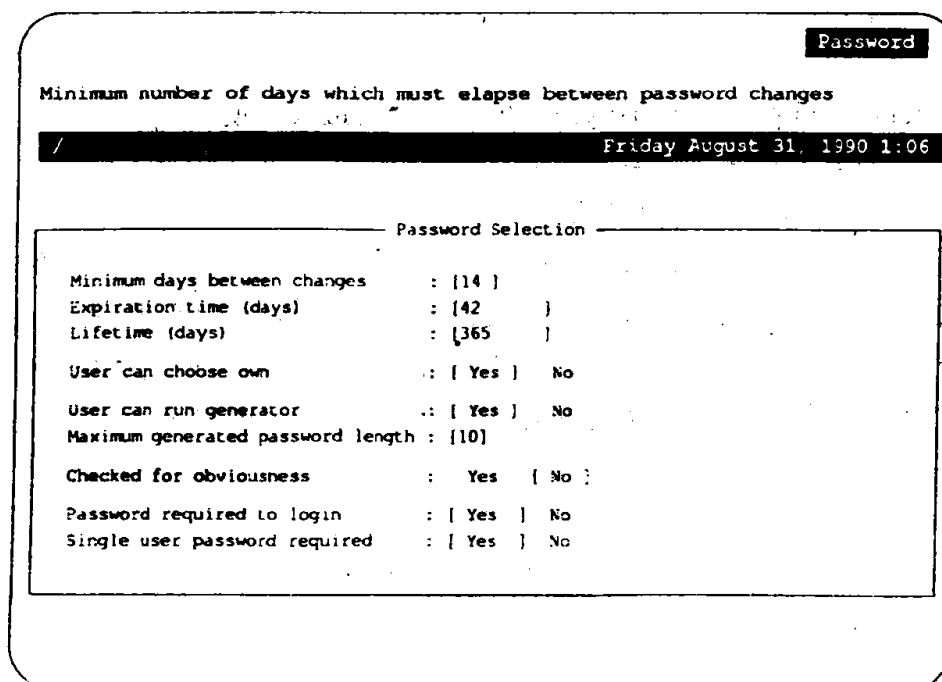


### 3.2.5 改变缺省的口令限制

前面已经给出了对入侵者猜口令次数的控制办法，下面的工作是控制口令自身的复杂度。为了访问口令限制的参数，请在 sysadmsh 中选取：

Accounts→Defaults→Password

这时就显示如下的屏幕内容：



系统所进行的口令检查类型就是由此屏幕上所显示的参数来控制的。这些参数控制着口令有效的时间以及一旦这个口令失效而改变它的过程。一个口令一直是有效，直到它“退役”或“死亡”（“expires” or “dies”）。一个退役的口令可以由任何一个有权改变此帐号口令的人来改变。当一个口令到了退役时间（expiration time），它就退役了。这个退役时间可以由管理接口程序对整个系统或对某一个用户进行设置，它是以前该口令最后一次被修改之日起的天数来表示的。口令死亡，就使得该用户帐号被封闭，此时，只有系统管理员可以开启这个帐号，并且要在用户再一次注册之前改变这个口令。

为了防止用户在其口令退役时改变口令后又立即改回他们原来的口令，系统还设置了在两次改变口令之间的最短时间这个参数。在这个最短时间没有过完之前，口令是不能改变的。这个参数即可以针对整个系统来设定也可以针对各用户而单独设定。

如下各参数定义对口令的限制：

Minimum days between password change（两次口令修改之间的最短天数）

在下一次改变口令之前必须等待的天数。

Expiration time days（退役时间（天））

口令有效的时间长度。

Lifetime（days）（生命期（天））

从最后一次改变口令到口令死亡之间的时间间隔。

User can choose own（用户可以自己选择）

这个参数用来规定用户是否可以自己选择口令。一个“可靠”的系统要求用户口令必须由系统自动生成。这就防止了因用户选取了“显而易见”的口令而使有经验的人可根据用户的特征而猜出其口令。然而其它 UNIX 系统是允许用户自己选择口令的。如果这个参数被置为 Yes，那么非可靠 UNIX 系统的规则就生效，即允许用户自己选取口令。如果这个参数被置为 No，那么口令必须由系统根据随机口令生成过程来产生。

User can run generator（用户可运行生成器）

这个参数允许用户运行口令生成器。请注意这不是允许用户自己选择口令，而只是允许生成一个新的随机口令。

Maximum generated password length（生成口令的最大长度）

所生成的口令的最大长度是由系统规定的，这个最大长度是80个字符。

Checked for obviousness（可见性检查）

这个参数用以规定系统是否要对所生成的口令进行平凡性的检查，以确保其不会出现在在线的词典中，而这项工作要与命令 goodpw（ADM）配合进行。如果把这一参数置为 Yes，就会使所有以真实单词为口令而企图打入系统的尝试都归于失败，但通过限制用户和终端注册会对此实现更有效的控制。平凡性检查实际上增加了改变口令所需的时间。所有这三个参数都可以由为用户特殊给定的参数值所覆盖。

Password required to login（注册所需的口令）

如果置为 No，用户帐号就可以不要口令；如果置为 Yes，则该帐号必须有口令才能允许注册。

Single user password required（所需的单用户口令）

这个参数规定了在以单用户(维护)状态下启动系统时是否需要口令。当一个帐号被系统封闭了，只有根或帐号管理员才能对它启封，同时也必须改变口令。可以按前面“修改用户口令或口令参数”中的描述对任何用户的上述参数进行重写。

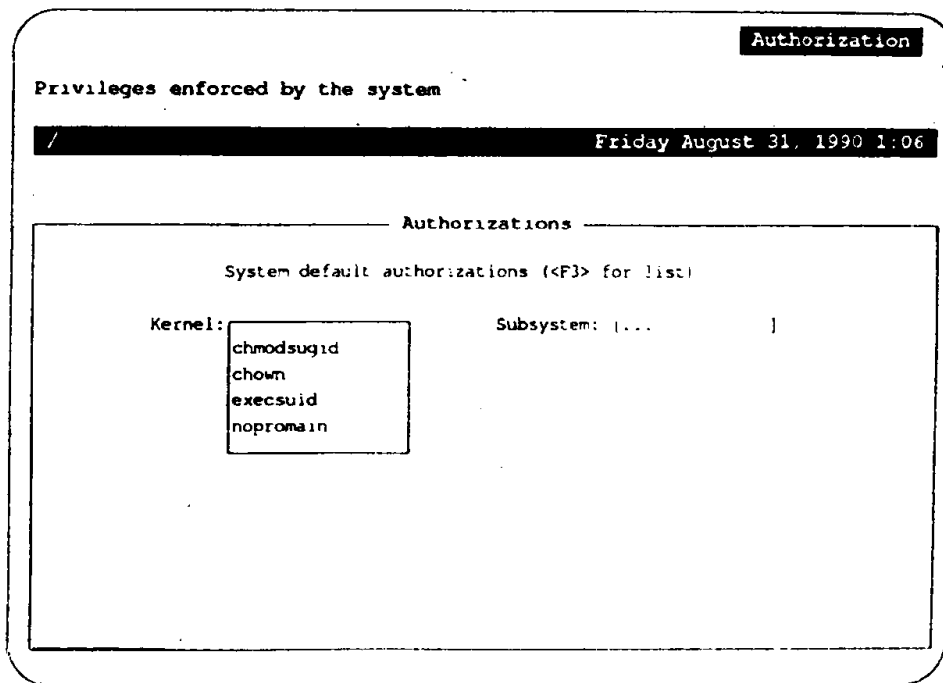
### 3.2.6 改变缺省权限

系统中定义了两种类型的权限：核心权限和子系统权限。子系统权限 (Subsystem authorizations) 是与用户相关的，它使用户能够执行可靠的工具软件。核心权限 (Kernel authorizations) 是与进程相关的，如果某进程有必需的权限，它就可以进行某种动作。每一个用户对话区都有一组核心权限和一组子系统权限。

为了访问权限参数，请在 sysadmsh 中进行如下选择：

Accounts→Defaults→Authorizations

就会显示如下屏幕：



可以使用 <TAB> 键在核心权限和子系统权限之间移动。使用 <F3> 键就会得到一个弹出窗口，其中列出了各组权限。表 3.2 列出了子系统权限的说明。

子系统权限决定了用户通过运行可靠的工具而承担的管理任务。对于一般的系统用户，是不能有任何子系统权限的。管理人员按照各自的职责被授予子系统权限；即帐号管理员被授予 auth 权限，打印机管理员被授予 lp 权限。

表 3.2 子系统权限

权限	子系统	权力
mem	内存	存取“保密的”系统数据，列出所有进程。
terminal	终端	可以不受任何限制地使用 write (C) 命令。
lp	行式打印机	打印机管理。
backup	备份	进行文件备份。

权限	子系统	权力
auth	帐号	帐号管理员：增加用户，改变口令，等等。
audit	审计	审计管理员：运行系统审计并生成报告。
cron	作业调度	控制 cron (C) .at (C) 和 batch (C) 命令的使用。

在系统缺省数据库中有一组缺省的命令权限，这是提供给所有那些在自己的帐号信息中没有任何命令权限的用户的。就 C2 级安全性缺省设置而言，在系统缺省数据库中的子系统权限为空；而在用户专用条目中，如果某用户有系统管理的任务，则根据其任务设置相应的权限。

具有 sysadmin 子系统权限就有运行 integrity (ADM) 程序的控制权，可以检查列在文件控制数据库中的文件许可权。(更详细的内容请参见本书中“系统安全性维护”一章中的“系统文件完整性检查”：integrity (ADM))。

第二级权限允许用户对资源进行有限的访问，否则，这些资源会被严密地控制起来(例如，如果没有 printqueue 权限，用户在使用 lpstat 命令时就只能看到他自己的作业)。这些权限给用户提供一些更接近其它 UNIX 操作系统的工作行为能力。参见表 3.3。

对于放松的缺省设置，这些第二级权限都被赋予用户。如果正在使用的是放松的缺省设置，但又不想不给用户过些权限，就必须一个一个地按前面所说明的那样对这些缺省设置进行重新规定。

表 3.3 第二级权限

第二级权限	子系统	说明
audittrail	audit	有能力对其自己的活动生成审计报告。
queryspace	backup	使用 df 命令来查询磁盘空间。
printqueue	lp	使用 lpstat 命令查看队列中的所有作业。
printerstat	lp	能够使用打印机使能 / 禁止命令。
su	auth	有权访问根目录 (超级用户) 帐号。(仍需要根的口令)

注：当一个子系统的主权限被授予给用户，则其次权限也就被授予给用户。(例如，lp 权限就支持 printquene 和 printerstat 权限。)

### 超级用户与被授权的管理员

前面已经讨论了一般应由超级用户行使的权利，在安全性要求不太高的系统上，被分配给了受保护的子系统。但是，一些功能仍需由根用户来完成，在分配权限时牢记这一点是很重要的。赋予了 su 权限的管理员能够对超级用户帐号执行 su (C) 命令，但他并不具备根的权限集——用户自己的权限集依然起作用。

超级用户必须执行下述任务：

- 安装软件
- 磁盘分区和文件系统维护
- 文件的重新装入、覆盖和权限设置
- 系统停机

## • 故障排除

### 核心权限

核心权限用以控制用户所具有的执行特定的操作系统任务的权力。例如，改变一个文件的所有权的能力是由 `chown` 权限管理的。（权限 `chown` 允许使用命令 `chown (C)`。）当用户的核心权限没有特殊设定时，就被赋予缺省核心权限。因此，需要更多权限的用户就会有特定的登记项用以授予他们那些权限，而一般用户的权限就设置为包含在系统缺省数据库中的缺省值。

表 3.4 核心权限

权限	动作
<code>configaudit</code>	构造审计子系统的参数
<code>writeaudit</code>	向审计跟踪文件中写入审计记录
<code>execsuid</code>	具备运行 UID—设置程序的能力
<code>chmodsugid</code>	具备在文件中设置 UID—设置和 GID—设置位的能力
<code>chown</code>	具备改变一个对象的所有者的能力
<code>suspendaudit</code>	暂停操作系统的进程审计

这些权限所产生的各种限制是复杂的：它们按缺省设置装配，并依照 C2 可靠级的要求进行工作。审计参数是专门提供给审计操作的，不能分配给用户；在本书的“审计子系统”一章对这些参数做了解释。

权限 `execsuid` `chmodsugid` 和 `chown` 在本书“系统安全性维护”一章中的“分配核心权限”和“在安全系统上的守护程序的操作”中有详细说明。

### 核心权限和参与管理的用户

必须随子系统权限分配某些核心权限，在系统中大多数核心权限已经通过缺省设定分配过了，它们都列在表 3.5 中，以供对缺省设定进行修改。有一个例外就是审计子系统，它需要另外的 `configaudit` 和 `suspendaudit` 权限。这些权限绝不能缺省设定，也不能分配给普通用户。另一个例外是 `sysadmin` 权限，它需要配合有 `chmodsugid` 核心权限，虽然作为根运行 `integrity (ADM)` 程序会更简单些。

表 3.5 子系统的核心权限需求

子系统权限	所需的核心权限
<code>audit</code>	<code>configaudit</code> , <code>suspendaudit</code> , <code>execsuid</code>
<code>auth</code>	<code>chown</code> , <code>execsuid</code>
<code>backup</code>	<code>execsuid</code>
<code>lp</code>	<code>chown</code>
<code>cron</code>	<code>execsuid</code> , <code>chown</code> , <code>chmodsugid</code>
<code>sysadmin</code>	<code>execsuid</code> , <code>chmodsugid</code> , <code>chown</code>

### 3.2.7 增加注册 shell 和配置文件

可以使用 `sysadmsh` 通过加入一些文件为用户选取并建立另一种注册 shell。例如，当

把 csh 选为某用户的注册 shell 时，就可以使用 csh (C) 命令在这个用户的目录中安装原型文件 .login 和 .cshrc。每个 shell 都在 /usr/mkuser 中有一个这些原型文件的目录。可以查看一下这些文件并按其中的例子进行操作。一定要使它们的操作权和所属权与其它文件一致。当安装了这些文件，这个新的 shell 就在用户标识表 (Identity form) 中的 Login shell 域登记了。

### 3.3 终端注册管理

终端控制数据库 (Terminal Control database) 保存着有关系统终端的参数。这个数据库可以使管理员控制做多少次不成功注册之后就封锁终端。它也保存有在各终端上注册活动状况。当安装了一个终端或打印机之后，有关信息就被自动加到终端控制数据库中。然而，必须对这些登记项进行修改，以控制如何使用它们以及应遵循何种安全性规程。

“Accounts”菜单的 Terminals 选项中有如下选择：

- Examine 查看或修改一个现存的终端登记项。
- Create 建立一个新的终端登记项。
- Delete 删除一个现存的终端登记项。
- Lock 封锁特定的终端。
- Unlock 给特定的终端解锁。
- Assign 管理设备名等价数据库。

当向系统中加入 tty 设备时，就自动在终端控制数据库中建立基本的登记项。最经常使用的选择是 Examine、lock 和 Unlock。其它的选择，Create、Delete 和 Assign，都是特殊的事件，当系统需要人工配置特殊的硬件或软件时才使用这些选择项。

#### 3.3.1 检查终端登记项

要想修改终端的设置，可进行如下选择：

Accounts → Terminal → Examine

这时会显示这样一个屏幕表格：

Examine

Enter the name of a terminal device in /dev

---

Friday August 31, 1990 1:06

Terminal Database Entry

Terminal device	:	[		]	
					User
					Date/time
Last login	:	[		]	]
Last logout	:	[		]	]
Last failed login	:	[		]	]
Consecutive unsuccessful logins	:	[		]	]
		[Specify]	Default of	[	] Value: [ ]
Delay between attempts	:	[Specify]	Default of	[	] Value: [ ]
Time to complete login	:	[Specify]	Default of	[	] Value: [ ]

这个屏幕可以用来检查终端的当前状态。终端名是必须要给出的，它是在 /dev 目录中的该终端的目录登记项。如果把“Consecutive unsuccessful logins”的值定为“INFINITE”，那么该终端就不会因多次不成功注册而被封锁。这个终端控制登记项与该设备分配数据库有关，这将在本章后面进行讲述。

### 3.3.2 对注册次数的限制进行重新设定

如果某一终端的注册限制有问题，或者是漏洞太大，那么可以使用如下的 sysadmsh 选择对这些限制进行重新设置：

Accounts → Terminal → Examine

所显示的屏幕表格请看前一节。按需要修改“Consecutive unsuccessful logins”和“Delay between attempts”。

### 3.3.3 对终端进行封锁或解锁

要想对终端进行封锁或解锁，可分别使用如下两个 sysadmsh 选择：

Accounts → Terminal → Lock

Accounts → Terminal → Unlock

当出现输入终端名的提示时，就输入所要的终端名，例如 tty01。若某终端被封锁了，当有人想在该终端上登录时，就会显示如下信息：

Terminal is disabled--see Authentication Administrator

### 3.3.4 建立设备等价数据库

设备分配数据库（device assignment database）是用来记录物理上相同但要以不同的路径名引用的终端的。这种等价映射对于终端来说是很重要的，因为这保证了注册历史和终端的封锁状态能够与系统上的设备路径名正确地对应。

要想修改设备分配登记项，可进行如下选择：

Accounts → Terminals → Assign → Create

这时会显示如下屏幕表格：

The screenshot shows a terminal window titled 'Create'. At the top, it says 'Name of a character special device (<F3> for a list)'. Below that is a status bar showing 'Friday August 31, 1990 1:06'. The main area is titled 'Device Assignment Entry' and contains three lines of input fields: 'Device name: [ ]', 'Device type: [ Terminal ] Printer Removable', and 'Path names: [... ]'.



输入一个能够在 /dev 中找到的设备名。然后选择设备类型，是终端，打印机，或是可更换设备，例如可卸盒式硬盘。最后还应填写与这个设备相连接的所有的完整路径名。

### 3.4 活动报告的生成

操作系统能够对其三个重要方面的状态建立报告：

Passwords 报告帐号的口令状态

Terminal 报告终端的访问状态

Login 报告用户、组、或终端的注册活动可以使用这些报告来保证系统的安全性（例如，列出受保护口令和终端控制数据库中的参数）。因为这些报告给出了系统和外设的使用情况，所以它们对于更好地调整和重新配置系统是很有用的。

对于所有的报告，都会从屏幕上得到一个询问，是要把报告结果送到屏幕上，打印机上、还是一个文件中。

可以使用任何一个系统分页程序对屏幕输出进行过滤。这个分页程序的缺省设置为 PAGER 环境；如果 PAGER 没有定义，那么就使用 more (C)。对于打印机输出，可以给出打印机的名字；如果没有给打印机名，系统就会使用缺省打印机。如果把输出定向为一个文件，就要使用完整的路径名。不管选择哪一种报告、种类，都会被要求定义以何种形式显示数据：在屏幕上、送到打印机上、还是送到一个文件中。

输出屏幕如下：

```
Enter the name of the character device in /dev
/
Friday August 31, 1990 1:06

Output Selection

Send to: Terminal Printer File
Pager/Printer/File name:
{/usr/bin/more
Page length: [24]
Page width : [80]

Create
```

#### 3.4.1 报告口令状态

要想报告口令状态，可以进行如下 sysadmsh 选择：

Accounts→Report→Password

所报告的口令状态可被分为几类：

- Impending 报告口令将要过期的帐号。
- Expired 报告口令已经退役的帐号。
- Dead 报告口令已经死亡的帐号。
- User 对一个用户提出报告。
- Group 对一个用户组提出报告。
- Full 列出口令数据库中的所有登记项。

Impending 选项对于口令将要退役的或已经退役帐号提出报告。这包括所有的口令已经退役的帐号，也包括口令将在一周内退役的帐号。虽然口令退役的迫近本身并没有什么错误，但这个报告可以显示那些等到最后时刻才修改口令的用户。管理员也许想根据由此得到的信息来修改全系统的或各用户的口令有效期。

Expired 选项对于所有口令已经退役的帐号提出报告。这些口令可能是也可能不是已经死亡的口令。所有这种帐号在能够再次起用之前都必须经过系统管理员处理；至少是口令必须修改。

Dead 选项对于那些已经超出口令生命期的帐号提出报告，这使得这种帐号不能再接受注册。

User 选项对于所指定的特定用户提出报告。输入用户注册名就可激活这个选项。

Group 选项对于特定的一个用户组提出报告。这个报告包括属于这一组的所有用户。

最后是 Full 选项，对系统上的所有用户提出统计报告。

报告中使用的如下简写形式：

Dflt Default

Y, N, D Yes, No, Default。有些选择有三个可能的值：Yes, No, 以及系统所用的缺省值，其中任何一个都可以。

下面是“hamster”小组口令报告的例子。“Password Parameters”下的简写形式对应于全系统的口令缺省参数。

```

password Database Report
System unix
Wed Mar 22 10: 56: 29 1990
password parameters
[1] User  Name  Type  Min  Exp  Life  Rnd?  Pck?  Rst?  Lck?
-----
                Last Changes      Last Logins      Consec
[2] Success  Failed      Success  Failed      #Failed
-----
[3] Kernel  Authorizations
-----
[1] alvin   general  Dflt  Dflt  Dflt  D  D  D  Y
[2] 05 / 22 / 90  NEVER      05 / 22 / 90      NEVER      -
[3] DEFAULT
[1] slmon   general  Dflt  Dflt  Dflt  D  D  D  N
[2] 05 / 22 / 90  NEVER      05 / 22 / 90      NEVER      -

```

```

[3] DEFAULT
[1] theodore    general  Dflt  Dflt  Dflt  D   D   D   N
[2] 05/22/90  NEVER          05/22/90    05/22/90    -
[3] DEFAULT

```

### 3.4.2 报告终端活动

要想报告终端活动，可进行如下 sysadmsh 选择：

Accounts→Report→Terminal

这可以使你得到一个终端控制数据库的统计结果。报告中包含封锁条件、在终端上的不成功登录的记录，以及在两次注册之前必需的时间间隔。与口令报告类似，可以选择对单个终端、一指定系列终端，或对所有终端提出报告。

当选择了一个用户或用户组，在报告中就包含最后一次成功的和最后一次不成功的注册记录。不成功注册的次数也报告出来。因为这个数接近于对一个帐号的最大注册次数，可以由此判断问题的原因。大多数帐号的每一次注册的最大尝试次应该很小。

当选取了一个或多个终端时，报告中就包括最后一次成功、最后一次不成功的注册记录，以及最后一次注销的记录。该终端上的最大不成功注册次数也报告出来。这两种报告类型都能够提供关于系统使用状况的有价值的的数据。

这里是一个终端报告的输出的例子：

```

Terminal Database Report
System unix
Wed Mar 22 10: 58: 42 1990

Admin Login Unsucc Max Unsuc
Tty      Name Lck?   Delay      Attempts    Attempts
-----
console  D         Dflt       2           Dflt
tty02    D         Dflt       None        Dflt
tty03    D         Dflt       None        Dflt
tty04    D         Dflt       None        Dflt
tty05    D         Dflt       1           Dflt
tty06    D         Dflt       None        Dflt
tty07    D         Dflt       None        Dflt
tty08    D         Dflt       None        Dflt
tty09    D         Dflt       None        Dflt
tty10    D         Dflt       None        Dflt
tty11    D         Dflt       None        Dflt
tty12    D         Dflt       None        Dflt

```

### 3.4.3 报告注册活动

要想建立一个注册活动的报告，应做如下的 sysadmsh 选择：

Accounts→Report→Login

注册报告的生成可分为三类：根据用户、根据用户组、和根据终端。

这里是依据在一个终端上的注册活动所建立的报告输出:

Login Activity Report

System unix

Wed Mar 22 14: 43: 53 1990

Tty	Name	User Name	Last Good Login Date	User Name	Last Bad Login Date	User Name	Last Logout Date	#Failed
console		alvin	05 / 22 / 90	KNOWN	05 / 22 / 90	alvin	05 / 22 / 90	2
tty02		root	05 / 21 / 90	root	05 / 22 / 90	root	05 / 19 / 90	0
tty05		maryt	05 / 21 / 90	UNKNOWN	05 / 21 / 90	root	05 / 19 / 90	0
tty04		root	05 / 19 / 90	root	05 / 13 / 90	root	05 / 19 / 90	0
tty05		UNKNOWN	NEVER	root	05 / 13 / 90	UNKNOWN	NEVER	1
tty06		UNKNOWN	NEVER	UNKNOWN	NEVER	UNKNOWN	NEVER	0
tty07		UNKNOWN	NEVER	UNKNOWN	NEVER	UNKNOWN	NEVER	0
tty08		UNKNOWN	NEVER	UNKNOWN	NEVER	UNKNOWN	NEVER	0
tty09		UNKNOWN	NEVER	UNKNOWN	NEVER	UNKNOWN	NEVER	0
tty10		UNKNOWN	NEVER	UNKNOWN	NEVER	UNKNOWN	NEVER	0
tty11		UNKNOWN	NEVER	UNKNOWN	NEVER	UNKNOWN	NEVER	0
tty12		UNKNOWN	NEVER	UNKNOWN	NEVER	UNKNOWN	NEVER	0

## 第四章 文件系统的管理

这一章介绍系统管理员的最重要的职责之一：建立和维护文件系统。这里将讲述通用的维护活动，譬如自由空间的维护策略，还要介绍一下“文件系统完整性 (file system integrity)”的概念，以及操作系统是如何修复损坏了的文件系统的。文件系统的建立在本书的“增加硬盘”一章进行了讨论，而有关文件的使用权限及其它的安全性考虑，请看本书的“系统安全性维护”一章。

什么是“文件系统 (file system)”呢？

文件系统是操作系统的一个特定部分，它由文件、目录以及寻找定位和访问它们所需要的信息组成。一个文件系统可以被看作一个由目录和文件构成的结构。每一个 UNIX 系统在其主硬盘上至少有一个文件系统，这个文件系统被称为“根文件系统 (the root file system)”，并且由“/”符号来表示。根文件系统包含构成操作系统的程序和目录。在小容量硬盘上，根文件系统还包括所有的用户目录。主硬盘也可以被划分成多个文件系统，这在“系统的安装”一章中有详细介绍。一个最常用的划分是 /u 文件系统，它是用来隔离用户帐号和根文件系统的，有关这些文件系统的内容，请看第一部分“安装指南”。

一个 UNIX 系统还可以包括其它一些文件系统，这些文件系统包括一些特殊的目录和应用程序。把主硬盘分成多个文件系统可以保护数据，并可以使得维护更加容易。通过安装另外的硬盘来增加更多的文件系统可以扩展系统的存储空间。新的文件系统可以由系统管理员特别地建立，然后在需要的时候安装到系统上或从系统上卸下来，也可以用同样的方法来访问软盘。这个过程在本书的“增加硬盘”一章中介绍。

### 4.1 文件系统的安装和拆卸

mount (ADM) 命令是用来安装和拆卸文件系统的。在安装时，必须指明文件系统的类型。例如，要在 /u 上安装或拆卸 /dev/u，就要分别使用下面的两个命令：

```
mount /dev/u /u
```

△sysadmsh 使用者可以选择 Filesystems → Mount

```
umount /dev/u
```

△sysadmsh 使用者可以选择 Filesystems → unmount

只有超级用户可以使用 mount 命令。

---

注：

仅当某一个文件系统被安装上之后，其中的文件才能被访问。如果当某一个文件系统还没有安装上时，就往这个文件系统中的目录拷贝或在其中建立文件，那么这些文件看起来是在那个文件系统中存在了，但实际上却不存在。当这个文件系统被安装上之后，这些文件看起来就消失了。

---

虽然只有超级用户可以使用 mount 命令，但是超级用户可以通过设置参数来规定哪些文件系统可以由用户使用 mnt (C) 命令进行安装，如果需要，还可以包括访问口令的

使用。

每一个文件系统都必须在 /etc/default/filesys 中有一个登记项。图 4-1 中包括了登记项中的一组样本。

```
bdev = / dev / root cdev = dev / rroot mountdir = / \  
desc = "The Root File system" romount = no mount = no  
  
bdev = / dev / u cdev = / dev / ru mountdir = / u romount = yes \  
fsckflags = -y desc = "The user File system"  
  
bdev = / dev / x cdev = / dev / rx mountdir = / x mount = yes \  
romount = yes fsckflags = -y desc = "The Extra File system"
```

图 4-1 /etc/default/filesys 文件实例

简单地说，实例当中的各登记项决定了表 4.1 中所给出的各种行为。

表 4.1 文件系统安装说明

文件系统	安装时间	用户是否能安装
root	启动时	不能
/u	多用户状态	不能
/x	任何时候	能

如果希望某一个非根文件系统可由用户安装，只要在这个给定的文件系统的登记项中加入“mount=yes”即可。另外，如果在调用 mnt 命令时不给任何参数（无文件系统名），那么程序就将检查所有的非根文件系统，看它们是否可安装，如果可以，就进行安装。如果带“mount=prompt”选项，那么当遇到一个文件系统允许用户安装时，就会向用户提问是否要对其进行安装。

## 4.2 文件系统的类型

在系统中可以设置如下四种不同类型的文件系统：

- XENIX
- UNIX
- DOS
- AFS (Acer Fast Filesystem)

UNIX 与 XENIX 文件系统在内部结构上有微小的差别，但这并不会造成严重的后果。缺省文件系统是一种 UNIX 文件系统的快速变体，叫做 AFS (Acer Fast Filesystem)，它的速度要比 UNIX 和 XENIX 文件系统的速度明显地快。这些文件系统的块的大小都是 1K。DOS 文件系统将在本书中的“DOS 和 OS/2 的使用”一章中进行讨论。

一个 UNIX 文件系统在任何时候都可以被转化为一个 AFS 文件系统。这是通过使用带选项 -Cclustersize 的 fsck (ADM) 命令来实现的，而这个一般是用来检查和修复文件系统的。这个 clustersize 参数必须是 2 的乘方，并且必须小于 16，(推荐值是 8)。

AFS 文件系统的真正优点只有当它是一个新的文件系统时才能看出来。经转化而得来的 AFS 文件系统在速度上的增加并不能立即显现出来，只有当有新文件增加到该文件系统后，这个优势才会真正起作用。对一个空间几乎充满了的文件系统进行转化，作用是很小甚至根本没有的，如果某一个 UNIX 文件系统处于几个完全充满的存储块中，那么向 AFS 文件系统的转化就根本不能实现。（有关 fsck 命令的完全的讨论请参见“文件系统的完整性”一节）

---

注：在运行 fsck 之前，必须把该文件系统卸下来。

---

fsck 命令格式如下：

```
Fsck -S -Cclustersize device
```

这里的 device 是在 /dev 中的该文件的设备名。

注意，-s 选项也是必须要的。

### 4.3 文件系统的可配置特征

文件系统的某些特征是可以配置的，包括：

- 新建立文件的组 ID (Group ID)：一个新文件或是承袭其父目录的组 ID 或是承袭其建立用户的组 ID。
- 文件名的截取：若企图建立一个超过 14 个字符长的文件名，将怎样处理。

#### 4.3.1 设置目录的 SGID 位

按照缺省，一个新建立文件的 GID (组 ID) 将被设置为其创建进程的或用户的 GID。而这一行为可以通过设置一个目录的 SGID 位来改变。通过设置某一目录的 GID，可以使得在其中建立的新文件承袭该目录的 GID。为设置某目录的 GID 位，请输入下面的命令，其中 directory 是该目录名：

```
chmod g+s directory
```

要删去该位，只要把上面的命令中的“+”用“-”代替。

#### 4.3.2 设置文件名截取

按照缺省，如果建立了一个长于 14 个字符的文件名，就会显示错误信息“Filename too long.”这可以通过把过长的文件名截为 14 个字符来解决。这种缺省行为是根据 POSIX FIPS 的要求而规定的，并由 ETRUNC 核心参数控制。这个参数可以改变，方法是这样的，调用 sysadmsh，选择 System → Configure → Kernel → Parameters，然后选取第三类：“Files, Inodes and Filesystems,”再把 ETRUNC 的值改为 1。之后，核心必须重新连接，并重新启动系统。这样新的行为规范才能起作用。使用 sysadmsh 的 System → Configure → Kernel → Rebuild 选择来做核心的重新连接。更详细的说明请看本书中“系统性能的调整”一章中的“利用 configure 重新分配核心资源”一节。

### 4.4 文件系统中空闲空间的维护

系统管理员的一项重要工作就是进行文件系统的维护，其中包括维持系统的顺利运行，维持文件系统的清洁，并保证所有的用户都有足够的空间。要维护文件系统，系统管理员必须对每一个文件系统的空闲空间进行监视，并当空闲空间变得很小时，采取适当的行动。

这部分将介绍文件系统维护的命令，这些命令能够报告有多少空间已经被使用了，找出不常用的文件，以及删除或修复被损坏的文件。

在每个文件系统都至少有 15% 的空间空闲的情况下，UNIX 系统的运行处于最佳状态。任何系统中，空闲空间的总量决定于包含有文件系统的磁盘的大小以及盘上文件的数量。由于每个磁盘的空间总量是一定的，因此控制存在盘上的文件数量是十分重要的。

如果一个文件系统中空闲空间少于 15%，系统的运行就会变得缓慢。如果没有一点儿空闲空间，系统就会拒绝执行某些对该文件系统进行的写操作。也就是说，在这台计算机上用户的正常工作（建立一个新文件或扩大一个现存的文件）就会停止。

对一个空闲空间少于 15% 的文件系统唯一的解决办法就是从该文件系统中删除一个或多个文件。下面几部分将介绍维护空闲空间的策略。

#### 4.4.1 空闲空间的维护策略

系统管理员应该经常对所安装的所有文件系统的空闲空间总量进行检查，并提醒用户把他们目录中的无用文件删除。可以通过在 `/etc/motd`（当日信息）文件中包含一段提示信息来提醒用户。

另外，`cleantmp`（ADM）命令是由系统运行来清洁 `/tmp` 目录的。可以编辑 `/etc/default/cleantmp` 文件来规定间隔多长时间对关键目录（缺省为 `/tmp`）中的文件进行一次清洗。

如果空闲空间的总量滑降为 15% 以下，系统管理员应该进行下面几步工作：

1. 对全系统发送一条信息要求用户删除无用的文件。
  2. 找出特别大的目录和文件，并向它们的主人发送一个邮件要求他们删除不必要的文件。
  3. 找出并删除暂时文件和名为 `core` 的文件。
  4. 清除系统记录文件（log files）的内容。
  5. 通过做该文件系统的完全备份来减少磁盘中的存储碎片，删除所有文件，再把它们从备份中重新恢复到盘上。
  6. 如果系统长期缺少空闲空间，那就可能有必要建立和安装另一个文件系统。
- 上述建议都在下面的各部分中详细说明。

#### 4.4.2 显示空闲空间

可以用 `df`（表示“disk free”）命令来查看在某个文件系统中所存在的空闲空间的大小。这个命令可以显示出在一个指定的文件系统中空闲空间的“块”数。每一块是 512 个字符（或字节）的数据。

`df` 命令的格式如下：

```
df specialfile
```



△sysadmsh 用户可以选择：System→Report→Disk

这里 specialfile 可以是一个 UNIX 特殊文件名，它与包含该文件系统的磁盘驱动器相对应。如果没有给出这个特殊文件名，那么所有正常安装的文件系统的空闲空间都会给出来。

例如，要显示根文件系统 /dev/root 的空闲空间，就输入：

```
df /dev/root
```

按回车键。这个命令就会显示出特殊文件名和空闲块的数目。

还可以利用下面的命令来获得在系统中空闲空间占总空间的百分比：

```
df-v
```

#### 4.4.3 向全系统发送信息

如果空闲空间少了，可以用 wall（表示“write to all”）命令向系统中的所有用户发送一个信息。这个命令把在终端上键入的信息拷贝到当前注册了的所有用户的终端上。

要发送一条信息，键入：

```
wall
```

按回车键。键入信息，如果需要，可以按回车键开始新的一行。信息输入完毕，按 <CTL>d。这样就把该信息显示到系统中所有终端上。要想脱离 wall 命令，再按 <CTL>d，这就中断了与其它终端的联系。

#### 4.4.4 显示磁盘的使用状况

可以利用 du 命令来显示某个目录所使用的块的数目。这个命令可以发现特大的目录和文件。

du 命令格式如下：

```
du directory
```

这里的选项 directory 必须是一个安装了的文件系统中的目录名。如果不给出目录名，这个命令就会显示当前目录的块的数目。

例如要显示目录 /usr/johnd 所使用的块的数目，就键入：

```
du /usr/johnd
```

按回车键。这个命令就显示出在 /usr/johnd 目录中的每一个文件和目录的名字及其所使用的块的数目。

#### 4.4.5 显示各用户所占用的块

可以利用 quot（表示“quota”）命令来显示用户清单及其所占用块的数目。命令格式如下：

```
quot specialfile
```

这里 specialfile 必须是对应于某一个文件系统的特殊文件名。

例如，要显示 /dev/u 上的文件系统中文件的所有者，只要键入：

```
quot /dev/u
```

按回车键。这个命令显示出在这个文件系统中拥有文件的用户以及这些文件所占的块数。

#### 4.4.6 给一个用户邮递信息

如果一个用户有特别大的目录和文件，可以利用 mail 命令给这个用户发送一条信息。

利用邮件发送信息，开始应键入：

```
mail login-name
```

按回车键。这里 login-name 是接受者的注册名。要发送一条信息，应先键入信息，按回车键，再按 <CTL>d。如果信息超过一行，可以在每一行的末尾按回车键。mail 命令把这个信息拷贝到用户的信箱中，在那里，它可以被用户通过 mail 命令看到。

#### 4.4.7 查找文件

可以利用 find 命令根据文件名、权限设置、大小、类型、拥有者或最后一次访问或修改时间来找到所有的文件。这个命令对于查找不常用的或特别大的文件或由某个特定用户拥有的文件是非常有用的。

find 命令格式如下：

```
find pathname option
```

这里 pathname 是要搜索的目录的路径名。find 命令在这个指定的目录下向下搜索所有的目录，循环查找符合由 option（选项）所指定的标准的文件。有些选项也为 find 指定某些作用于所找到的文件上的动作。

表 4.2 介绍了一些对系统管理员非常有用的 find 的选项。

表 4.2 有用的 find 选项

选 项	说 明
-atime number	查找在最近指定的天数 (number of days) 内未曾访问过的文件。
-exec cmd	查找符合指定标准的文件并对那些文件执行 cmd。命令参数 ({ }) 由 find 所找到文件的当前路径名所取代。一个逃逸的分号 (\;) 必须跟在 cmd{ } 结构的后面。
-group gname	寻找属于组 gname 的文件。如果 gname 是数字并没有在 /etc/group 中出现，那么它就被当作一个 GID。
-name file	寻找带有指定名字的文件。
-ok cmd	它类似于 -exec，不同的是它还以提示的形式显示所产生的命令；只有当用户对于提示键入 y 时 cmd 才执行。
-perm onum	查找完全符合 onum (chmod (C) 所用到的八进制数) 所规定的访问权限的所有文件。
-print	显示 find 所找到的任何文件的位置。
-size number	查找大小大于指定块 (每块 512 个字节) 数的文件。
-type x	查找指定类型的文件；例如目录 d 型或文件 f 型。
-user uname	查找所有属于用户 uname 的文件。如果 uname 是一个数字并且没在 /etc/passwd 中出现，那么它就被解释为注册 ID。

---

注：如果不包含 `-print` 选项，`find` 就不会显示符合查找标准的文件清单。

---

#### (1) 寻找临时文件

要在 `/usr` 目录中重复查找并显示名为 `temp` 的所有文件，应使用下面的命令：

```
find /usr -name temp -print |more
```

#### (2) 寻找指定大小的文件

可以使用 `find` 来寻找指定大小或类型的文件。例如，要找到并显示出所有目录中（`/` 和其下的类型为 `-type d`）大小大于 3 块（`-size +3`）的目录，就键入：

```
find / -type d -size +3 -print
```

#### (3) 根据访问权限查找文件

对 `find` 使用 `-perm onum` 选项，可以找到所有完全符合 `onum`（`chmod (C)` 所使用的八进制数）访问权限的文件。例如查找并显示在 `/u` 目录中的赋予所有用户读、写和执行权限（`onum` 是 `0777`）的所有文件，键入如下命令：

```
find /u -perm 0777 -print
```

#### (4) 在 `find` 的输出上执行命令

使用 `-exec` 选项，可以对 `find` 所找到的文件执行一条指定的 `shell` 命令。`-exec` 的最普通的用处是找到一组文件并把它们删掉。例如，当一个用户退役之后，可以使用 `find` 命令找到那个用户所拥有的所有文件，对它们进行备份，并把它们从系统中删除。要用 `find` 来实现上述工作，键入：

```
find / -user edwarda -print |cpio -ovBc> /dev/rfd096
```

```
find / -user edwarda -exec rm {} \;
```

第一个命令找到用户 `edwarda` 所拥有的全部文件，并把这些文件拷贝到一个大容量外存上。第二个命令找到这些文件并把它们删除。更多的有关把文件拷贝到大容量外存储器的知识请看 `cpio (C)` 命令的说明。

还可以使用 `-ok` 选项替代 `-exec` 来指示 `find` 命令在对每个文件执行这个 `shell` 命令之前先显示一个提示（`find` 所产生的命令行）。当使用 `-ok` 时，`find` 就以所产生的命令行提示：

```
<rm... /u / edwarda / billboard>?
```

要执行这个命令（在这里是 `rm`），就键入 `y`。如果键入了 `y` 以外的其它字符或是回车键，这个命令就不会执行。

带 `-exec` 选项的 `find` 的另一个常用之处是找到属于某一个特定组的全部文件并改变它们。例如，如果一个用户换了组，就可以使用 `find` 找到他的所有文件并把它们换到新组中：

```
find / -user edwarda -exec chgrp moms {} \;
```

如果一个用户退役了，要想把他的文件的所有权交给另外一个用户，就可以使用下面的命令：

```
find / -user edawrda -exec chown earnestc {} \;
```

使用这一指令对很大的一组文件执行一条命令可能是很慢的，这是因为 `-exec` 选项要

给这一系列文件中的每一个都派生一个单独的进程。做这件事的一个更有效的方法是用 `xargs (c)` 来代替 `-exec`。 `xargs` 命令在对整个一组文件执行一个命令时会派生出少一些的进程。下面的例子说明了如何与 `find` 一起使用 `xargs` 指令：

```
find ./ -user edwarda -print |xargs chown earnestc
```

这个命令与前面的例子完成同样的工作，但效率更高。

---

注：如果利用 `xargs` 所执行的这个命令的句法与标准顺序（命令选项参数）有偏差，那么就必须使用 `-exec`。

---

#### (5) 查找 core 文件

`find` 的一个很有效的应用是利用 `-name` 选项来查找 core 文件和临时文件，并把它们删除。

core 文件包含一个终止了的程序的备份，当一个程序产生了一个它自己不能解决的错误时，UNIX 系统有时就会建立这样一个文件。临时文件包含一些数据，这些数据是一个程序在执行过程中所建立的中间步骤，当一个程序有错，或是被用户提前停止，这个文件就会被留下。临时文件名是依建立它的程序而定的。在大多数情况下，core 文件或临时文件对用户来说都是没用的，删除它们并没有什么危险。

要找 core 文件或临时文件，最好是找那些在合适的时间范围内未曾被访问过的文件。例如，要在 `/usr` 目录中查找在一个星期里没有访问过的所有的 core 文件，就键入：

```
find /usr -name core -atime +7 -print
```

一旦找到了 core 文件，就可以很容易地利用下面两条命令之一把它们删除：

```
find /usr -name core atime +7 -exec rm { } \;
```

```
find /usr -name core -atime +7 -print |xargs rm
```

这些命令在 `/usr` 目录中查找在最近七天内未曾被访问过的所有 core 文件，并把它们删除。

#### 4.4.8 检查并清理记录文件 (Log Files)

UNIX 系统要保存一些包含有系统使用状况信息的记录文件。当产生新信息时，系统就把它附加到恰当的记录文件中，同时也保留文件中原来的内容。由于有些记录文件会很快地变得很大，所以定期地检查这些文件是很重要的，并且在需要的时候，删掉其中的内容。也许需要在一些记录文件中保留最近所增加的信息，譬如在 `/etc/date` 中。在这种情况下，可以通过删掉过去的的数据而保留最新的记录项来对这些文件进行修剪。

有些记录文件仅在特殊情况下增长。例如，如果要记录所有使用 `su` 命令的过程，就可以在 `/etc/default/su` 上设置 `SULOG` 选项，这时记录文件 `/usr/adm/sulog` 就会迅速增大。

表 4.3 列出经常需要清理或修剪的一些记录文件。系统中可能还有一些不同于所列出的记录文件，这依赖于它的配置和所安装的各种工具及应用软件。也可以比所指定的频率或高或低地检查这些文件，这依赖于系统的活动。可以使用带 `-size` 标志的 `find (c)` 命令来查找系统中在下表中未列出的记录文件。

表 4.3 可管理的记录文件

记录文件路径名	目的	检查频率
/etc/wtmp	历史注册记录	每周一次
/usr/adm/pacct	进程记帐记录文件	每周一次
/usr/adm/messages	系统信息记录文件	每周一次
/tcb/audittmp/*	审计系统临时文件	每周一次
/etc/ddate	记录每次备份的日期	每月一次或每年一次
/usr/spool/micnet/remote/ <name>/LOG	记录指定名字的远程机器的网络活动	每月一次
/usr/spool/uucp/LOGFILE	记录 UUCP 的作业请求, 文件传输, 系统状态	每月一次
/usr/spool/uucp/.LOG/.Old/*	由 uudemmon.dean 保存的老的 UUCP 记录文件	每月一次
/usr/spool/lp/logs/request	打印机请求记录	自动 <sup>①</sup>

①这个文件(和其它文件)的维护可以自动完成; 详见“检查和清理记录文件”下的“自动清理: 打印请求记录”。

若要清除一个记录文件的内容, 并且还要保留文件的访问权限:

1. 把这个文件拷贝到一个新文件中。例如

```
cp /usr/adm/messages /usr/adm/messages.old
```

2. 如果是在 Bourne shell 下工作, 就用如下命令清除这个文件:

```
> /usr/adm/messages
```

若要监视当前正在向系统记录文件中增加的新信息, 可以像下面这样使用 tail (C) 命令:

```
tail -f /usr/adm/messages
```

使用 -f (follow) 选项, tail 命令可以显示出这个文件的最后十行, 后面还跟着从发出 tail 命令之时起至按 <DEL> 键杀掉这个命令之时止的这段时间中所增添到文件中的各行。

#### 自动清理: 打印请求记录

自动进行记录文件的清除也是可能的。这里就是一个使用 /usr/spool/lp/logs/requests 文件的例子。这个以及其它文件都可以利用像 cron 这样的工具来进行周期性的清除。下面是使用 crontab 命令所看到的有关打印设备的缺省设定信息:

```
13 3 * * * cd /usr/spool/lp/logs; if [ -f
requests]; then /bin/mv requests xyzy; /bin/cp
xyzy requests; > xyzy; /usr/lbin/agefile -c2
requests; /bin/mv xyzy requests; fi
```

这是 crontab 命令所显示的一行，但这里为了便于阅读把它分成了几行。这是什么意思呢？简单地说，这是给文件规定年龄。先把文件名改为 requests-1，再把以前几天的拷贝移到 requests-2 中。选项 -c 里的 2 告诉 agefile 程序从前两天起保留记录文件，同时丢弃旧文件。改变这个数目，能够改变保存的信息量。另外，如果希望每天保存信息或删除文件的次数更多，而不止一次，那么可以通过在 crontab 运行时修改头两个数来改变时间。目前的值 13 和 3，使得文件清除在每天的 3:13AM 进行。

这里所提供的缺省的 crontab 设定项足以防止在假脱机文件系统中积累的老的请求记录。有关请求记录的更多的内容请看本书“打印机”一章中的“使用请求登记中的信息”。

## 4.5 维持高效的文件系统组织

在文件系统的使用过程中，有两种情况会降低文件系统的效率：

- 由于不断地使用和重新使用文件系统的存储块而造成的磁盘碎片空间。
- 目录变得越来越大，而增加了搜索文件的时间。

这一节介绍如何避免这些问题而维持文件系统高效率地工作。

### 4.5.1 磁盘碎片空间

如果磁盘已经用了一定的时间，文件的不断建立和删除就会造成一种叫做磁盘碎片空间 (disk fragmentation) 的情况。这个意思就是文件系统中的文件被写到硬盘上的一个个小块空间中。当一个文件被写在磁盘的多个区域中时，就会多占用一些磁盘空间。要想恢复这些多占的空间（一般大约占整个文件系统空间的 5% 到 10%），可以先把文件系统中的所有文件做一个完整的备份，然后把所有的文件从磁盘上删去，最后再从备份中把文件恢复到磁盘上。上述操作的具体说明请看本书中“解决系统问题”一章中的“减少磁盘碎片空间”。

由于文件又被重新完全写到了磁盘上，每个文件都被写到连续空间中，碎片空间就减少了，并恢复了一小部分空间。对于一个利用率比较高的系统，最好每年做一次这个工作，而对于一个利用率比较低的系统，做这个工作的频率可以少一些。请注意，在文件被删除之前，必须确认是否已经做了完整、准确、可读的备份。

### 4.5.2 大目录

避免目录过大是明智的。有必要掌握一些目录大小的限制。包括 30 多个文件（加上所需的 . 和 .. 登记项）的目录适合于存在一个单独的磁盘存储块中，在其中查找文件的效率也是很高的。一个目录最多可以有 286 个登记项，这时依然能够正常工作，只要它只用来存储数据；如果它作为工作目录使用，那么只要其中的一个东西再大一些，就会造成灾难。保持小的注册目录是特别重要的，最好是不超过一个数据存储块。

如果很多文件被移进或移出一个目录，譬如一个假脱机 (spool) 目录，这个目录就可能变得很大。这是因为，一般来说，目录是不会缩小的。当在注册目录中有不少于 30 个文件登记项或在一个假脱机目录或数据存储目录中有不少于 286 个文件登记项时，即使删除了文件，系统在目录中的搜索效率也会降低。这是因为所删掉的登记项的 i 节点依然存在。

查看目录是否太大，即使其中的文件看起来并未超过 30 个，可以使用命令 `hd|wc -l`。这个命令可以告诉你在 i 节点表中的行（i 节点）数。

解决大目录这个问题可以从两方面入手。一方面是教用户保持小的注册目录，另一方面是使用 `cpio` 命令对目录进行备份和恢复，以消除删掉或移去文件后所剩下的多余的 i 节点。

#### (1) 限制注册目录的大小

为了限制注册目录的大小，每个用户必须建立一个子目录分级结构，并使这些子目录名都能反映所包含文件的用途和目的。这可以使用户十分容易地找到所需的文件。这里还必须强调一点，建立一个简洁、清晰的目录层次结构是很重要的，并且在每一级应该只有五至十个目录，这样在以后增加文件时就不必要再改变目录的分级结构或在各目录间移动文件了。

#### (2) 删去多余的 i 节点

为了删去 i 节点数的“影子”（当文件从子目录中删除或移走后所剩下的 i 节点数），可以对一个过大的目录使用 `cpio` 来进行备份和恢复。然后再使用 `hd` 命令列出这个目录中的文件，并且看一看是否有在标准文件目录表中看不到的文件。这些“影子”文件出现在文件表的末尾，并有空的 i 节点数；00 和 00 出现在最前面两列 16 进制数中。可以使用下面的过程从目录中删去这些空文件登记栏。请注意同样的过程也可以用在假脱机目录中。

1. 进到这个大目录的上一层目录中，并建立一个备份目录。例如，如果一个叫做“tracy”的用户有一个过大的主目录，就键入：

```
mkdir tracy.old
```

2. 进到这个过大的目录中并对其进行备份：

```
find.-print |cpio -pdl ../tracy.old
```

这个命令对这个目录及其所有子目录进行拷贝，在一些合适的地方也可以连接文件而不必拷贝。

3. 使用 `l` 和 `hd` 命令来对备份目录进行检查并且会看到所有的文件都存在，但所有的“影子”i 节点数都消失了。

4. 进到上一层记录中，删掉这个过大的记录。

```
rm -rf tracy
```

5. 对备份目录改名代替那个过大的目录。

```
mv tracy.old tracy
```

对于用户来说这个记录看起来还与原来一样，但是对它的搜索要比以前快多了。这可以提高系统性能，特别是如果这个过程被用于工作目录上。

## 4.6 增加磁盘空间和文件系统

如果空闲空间总是很少，那么最好是像“增加硬盘”一章中所说明的那样增加第 2 个硬盘以扩大系统的存储能力。一旦安装上新的硬盘，就可以使用新文件系统的空闲空间进行工作，或者干脆把用户或系统目录拷贝到它上面。

假设由于一个文件系统已满而使主硬盘的空闲空间变少，如果这时在其它文件系统中还有空闲空间，或是在这个硬盘上还有未用的空间，也可以改变主磁盘的布局，但这并不

像增加第 2 个硬盘那样简单。一般来说最好像安装部分中所讲的那样事先安排好硬盘的布局。

经常出现空间不足通常是由于在系统中有超过当前硬盘所能够合理负担的用户数量，或是有太多的目录或文件。无论在哪种情况下，在新的硬盘上建立一个新的文件系统都允许把一些用户和目录从主硬盘上传送到这个新加的硬盘上，而从已存在的文件系统中释放很大的一部分空间，进而改善系统运行。

如果已决定改变主硬盘上文件系统的数量或是在文件系统之间进行磁盘空间的重新分配，就必须备份系统并且按“系统安装”一章所讲的规程重新安装系统。应该使用“全配置”磁盘初始化，只有这样才能够对磁盘的布局进行手工控制。在安装过程中，请按要求手工控制磁盘空间的重新分配。

#### 4.7 文件系统的完整性

操作系统的最重要的工作就是维护文件系统数据的完整性。实际的数据丢失是很少发生的；UNIX 文件系统对于讹误的抵御也是很有力的。这是因为有一定量的冗余以特殊的结构存在着，它是不为用户所见的。正是这些结构保证了文件系统的完整性。例如，当系统遇到了电源中断这种意外情况时，就会有一小部分信息丢失。一两个文件就会受到某种破坏，使它们不能再被访问了。几乎在所有情况下，这个操作系统都能够恢复被破坏的文件。只有很少的情况，使得整个文件系统变得不可访问。

这个操作系统使用 fsck（意思是“filesystem check”）程序来修复被破坏的文件系统。fsck 程序检查文件系统的一致性。假使在某处文件的内容丢失了（极少），唯一的办法就是从文件系统备份中恢复所丢失的数据。在引导的时候（boot time）fsck 在根文件系统上自动执行。fsck 状态信息显示如下：

```
* * Phase 1--Check Blocks and Sizes
* * Phase 2--Pathnames
* * Phase 3--Connectivity
* * Phase 4--Reference Counts
* * Phase 5--Check Free List
```

如果系统异常中止（电源中断），就会显示用以报警的另外的信息：

```
FREE INODE COUNT WRONG IN SUPERBLK (FIX?)
```

实际上，这种信息是一个系统在非正常停机时的例行程序，这时只有键入 y 后，fsck 才能继续进行其恢复工作。这项工作不需要系统管理员的干预就可以进行，但一般最好在发生问题后弄清楚文件系统到底发生了什么问题。

为了讨论文件系统完整性这个概念以及 fsck 是如何工作的，有必要解释一下位于文件、目录、文件系统这些简单概念下面的结构。虽然并不是非常有必要懂得文件存储的原理，但知道上面信息的意思也是有用的，这样它们看起来就不那么神秘了。

这部分要讲一些 UNIX 操作系统的最基本原理。“用 fsck 修复文件系统”这部分会介绍使用 fsck 命令的简单机理。下面各分标题将讲解与 fsck 有关的文件系统结构。

##### 4.7.1 UNIX 系统对文件的维护



每一个文件系统都包含一种特别的结构，操作系统正是利用这种结构对文件系统中的文件和所保存的数据进行访问和维护的。

一个文件系统的结构基于硬盘保存数据的方法。虽然硬盘上包含有系统所使用的的数据，但是它并不是仅存在于与特定的文件相对应的单一的一小块区域内。谁都不能指着硬盘上的一点有把握地说：“我的文件就存在盘上的这部分。”事实上，数据很可能分散在磁盘上的各个地方。操作系统使用一种复杂的寻址方案来访问文件所分散到的每一块，并把这些数据作为一个整体单位送给用户。

数据被分散到各处，这是因为操作系统并不是真正对文件进行操作，而是对数据单元进行操作。例如，假定建立了一个文件，并且这个文件确实存在磁盘的一个部分上。然后，假设又对这个文件进行了编辑，在这里或那里删除了一些句子。这时所使用的磁盘空间就要比开始时少一点。这些少用的空间就是文件所存在的区域中的一系列空隙。磁盘空间是一种宝贵的资源，是不能浪费的。所以，那些少量的存储块就要被其它文件占用。

#### 4.7.2 UNIX 系统对文件系统的维护

文件系统包括文件和目录，它们是由一些叫“i 节点”和“数据块”的特殊结构来表示的。这些结构使得操作系统建立和跟踪文件系统成为可能。

**数据块 (data blocks):**

一个数据块是一个存在于磁盘上的1024个字节的数据单元。数据块中可以是目录登记项也可以是文件数据。一个目录登记项包含一个 i 节点数和一个文件名。

**i 节点 (inode):**

i 节点可以被看作是图书馆卡片目录中的卡片。每一个 i 节点包含一个文件的有关信息，就像一张卡片包含一本书的有关信息一样，包括其位置、大小、文件的类型以及连接着该文件的目录登记项的数目。应该记住的很重要的一点是在 i 节点中并不包括文件名；而文件名是包含在目录中的。在 i 节点中包含着构成文件的全部数据的地址，这就使得操作系统在需要时能够把数据全部聚积起来。

数据块不仅仅存在于硬盘上。为了减少在硬盘上寻找数据，最近使用过的数据块就被保存在一种叫做缓冲区特殊记忆结构的高速缓存器中。这些结构使得操作系统具有更高的效率。当有足够的数数据积聚起来足以写满磁盘上的一块或多块时，缓冲区就把其中的信息写到磁盘上。在掉电时，会有一小部分信息丢失，这是因为刚刚被改变的数据还没有被写到磁盘上。

硬盘中存有数据、i 节点、目录、文件和曾在内存中暂存过的数据块，操作系统是怎样跟踪它们的呢？答案是，所有这些维持文件与目录之间有效联系的结构可以使被割断了的联系重新建立起来。

还有一个特殊的数据块，这个“超级”块包含着关于文件系统的全面信息，而不是用来放置一个文件的一个特定块的。这个超级块中存放有安装这个文件系统和访问其数据所需的信息。它包括文件系统的大小、空闲 i 节点数以及现有的空闲空间的信息。

当对文件系统进行安装、维护或当它在系统上进行各种活动而被修改时，就要从磁盘的超级块中读出信息。update 命令还会以一定的时间间隔把信息写回磁盘上。update 命令每 30 秒钟调用一次 sync (C) 命令，这可以使得内存中的超级块和缓冲区内容被写到

磁盘上。如果系统瘫痪了并且存在磁盘上的信息被不适当改变了，文件系统就可能会产生讹误。

#### 4.7.3 文件系统讹误的原因

讹误会影响在这一节中所提到的所有的结构。这就是说，数据或是用来查询数据的结构都可能被破坏。这可能由几种原因造成：

硬件故障：

硬件故障是比较少见的。处理硬件故障的最好办法就是认真地按照所推荐的诊断和维护过程来进行操作。

程序中断：

能够造成程序运行失败的错误也很可能造成数据丢失。但这种错误并不容易发生，这是因为系统的容错性是很强的。

人为错误：

虽然承认这一点是很痛苦的，但这确实是造成文件系统讹误的最主要原因。所以必须遵守文件系统管理的各项规程。

#### 4.7.4 文件系统检查的规则

在检查和修复文件系统时，必须记住如下规则：

- 在拆卸文件系统（或关闭系统）前要先运行sync（ADM）命令。sync命令把高速缓存中的数据写回磁盘上。
- 在拆卸文件系统前要把系统降为单用户（维护）模式。
- 在对文件系统进行检查或修复之前要先用etc / umount（ADM）命令拆卸该文件系统。
- 在再次安装文件系统之前必须用 / etc / fsck（ADM）命令对其进行检查。
- 要先拆卸文件系统，譬如是在软盘上的一个文件系统，然后才能在物理上把它取走。
- 在对文件系统进行了检查和修复之后，一定要记住把它再安装上。

对文件系统定期进行的备份对于以后文件系统的完整性是最好的保证。

#### 4.7.5 利用 fsck 命令修复文件系统

为了检查和修复文件系统，可以使用 / etc / fsck（ADM）。fsck 命令对磁盘上的各种结构进行检查，并努力使它们重新和谐起来。在可能的地方，fsck 重新建立起联系并分解不正确的关联；它使文件系统变得“干净”。

在用 fsck 修复文件系统之前，要先并闭系统并把系统降为单用户（维护）模式。为了进行这一操作而又不致彻底关闭系统，应该利用 shutdown 的 su 参数：

```
/etc/shutdown -g10 su
```

第一个参数指明了系统关闭之前应经历的分种数。

---

注：shutdown 命令自动调用 sync。

一旦系统处于单用户模式，就可以使用下面的命令对文件系统进行检查和修复了：

```
fsck filesystem
```

△sysadmsh 用户可以选择：Filesystems ▶ Check

其中 filesystem 是这个文件系统设备名所对应的特殊文件名。例如， /dev/u 设备文件对应 /u 文件系统。

为了检查根文件系统，可使用 fsck 的 -b 参数：

```
fsck -b /dev/root
```

在对根文件系统运行 fsck -b 之前，系统一定要处于单用户模式。这个命令在检查结束之后可以自动重新安装根文件系统。

---

注：

fsck 程序实际上只是一个前端，它可以使不同类型的文件系统调用不同版本的 fsck。例如，fsck 可以为修复 DOS 文件系统调用一种特殊的版本。

---

例如，在掉电之后，应先使系统进入单用户模式，在进入多用户模式之前可以使用 fsck 命令来检查 /u 文件系统。做这件事，可键入下面的命令：

```
fsck /dev/u
```

fsck 命令对文件系统进行检查，同时以如下信息对其执行进程进行报告：

- \* \* Phase 1—Check Blocks and Sizes
- \* \* Phase 2—Pathnames
- \* \* Phase 3—Connectivity
- \* \* Phase 4—Reference Counts
- \* \* Phase 5—Check Free List

如果在其中的某一个阶段 (phase) 发现了一个被破坏的文件，fsck 就会问是否要对它进行修复或抢救。这时只要键入 y 就可以对这个被破坏的文件进行修复。应该允许系统对被破坏的文件进行修复，即使已经在其它某处有它的副本或打算把它删除。

请注意 fsck 会删去它认为被破坏得无法修复的任何文件。管理员可以为 fsck 决定是否要进行修复。也可以让 fsck 忽略不相容性错误，因为这种问题是很严重的，以至于必须使用 fsdb (ADM) 工具来修理它或者从系统备份中重新恢复它。如果管理员不能使用 fsdb，就必须让 fsck 来解决这种不相容，否则这个文件系统就不能使用了。(fsdb 在本书的“解决系统问题”一章中的“在 fsck 在大小检查处停止时修复文件系统”中有具体讲解)。

在整个文件系统干净之前，可能需要运行若干次 fsck。

#### 4.7.6 fsck 阶段的概要介绍

fsck 程序搜索并检查前面提到的每一个结构。每个阶段 (phase) 都要对各组成进行比较，并且检查这些组成之间是否相互适合。

阶段 1 对块及其大小进行检查。fsck 程序读出 i 节点表来确定各块的大小，并找到所使用的各文件的块。检查 i 节点是要检查 i 节点的类型、零连接数、i 节点大小、以及坏

的或重复的块。(坏块是指块值在文件系统边界以外的块)。当 fsck 问是否要清除 i 节点时, 其意思就是要把 i 节点中的坏信息用零清掉。这就会把与它相联系的文件或目录删掉。重复块是指两个 i 节点指向磁盘上的同一块。fsck 命令要寻找这个重复块的原始 i 节点, 为在阶段 2 时对其进行改正做准备。

阶段 2 检查路径名。对于在阶段 1 所移走文件, 还必须要把它们在目录中的登记项移走。阶段 2 要清除由于不适当的 i 节点状态而造成的错误条件、越界的 i 节点指针, 以及如前所述的指向坏 i 节点的目录。对于在阶段 1 发现的带有重复块的文件, fsck 要把两个文件都删掉 (这是少数几个需要系统管理员干预的领域之一)。

阶段 3 对连接进行检查。阶段 2 把不指向有效文件的目录删去。阶段 3 把从目录结构上割下来的文件又重新连接上。所有的没有关联的但又是有效的文件都被放到一个叫做 lost+found 的特殊目录中。

阶段 4 对关联记数进行检查。fsck 对于从阶段 2 和 3 生存下来的每一个登记项的连接记数进行检查。在一些情况下, 那些没有被指定位于一个目录结构下但却有一个 i 节点的文件能够被再次连接到 lost+found 的文件系统中。

阶段 5 对空闲表进行检查。fsck 命令检查由文件系统维护的空闲块表。当检查出一个不相容性错误时, fsck 就会提示对它进行重新建造。

阶段 6 抢救空闲表。如果在阶段 5 发现了一个坏的空闲表, 系统就会根据变动后的文件系统重新建造一个空闲块表。

#### 4.7.7 文件系统的自动检查

操作系统有时要求在刚开机时检查文件系统。这种情况一般发生在没有正常关机 (例如掉电) 之后。文件系统会检查在关机过程中被破坏的文件。

## 第五章 在同一硬盘上使用 UNIX 和 DOS

许多用户的计算机采用 MS-DOS 或其他与之相兼容的操作系统。本章介绍了在安装了 UNIX 操作系统以后继续使用 DOS 的实用程序、文件和应用程序的方法。用户甚至还可以在自己的 UNIX 系统上访问 DOS 文件和目录，或者安装 DOS 文件系统并直接访问文件。UNIX 系统的这种特性使得用户不必废弃用于 DOS 软件的投资，或者另外购买一台计算机来专门运行 UNIX 系统。

UNIX 系统的若干程序使得 DOS 和 UNIX 系统并存于一台计算机上成为可能。dos (C) 实用程序允许存取软盘上的或在硬盘 DOS 分区上的 DOS 文件。这些实用程序将在本章的稍后部分讨论。划分硬盘的实用程序为 fdisk (ADM)，在 DOS 和 UNIX 版本中一样可用。下一节介绍在同一硬盘上如何用 fdisk 来建立 DOS 分区和 UNIX 分区。另有一节讨论在装有 DOS 的硬盘上安装 UNIX 分区，还有一节将向主要使用 UNIX 系统的用户和主要使用 DOS 的用户介绍不同的引导配置。

---

注：安装在系统上的 DOS 必须是 DOS3.3 或者更低的版本。不支持扩展 DOS 分区。

---

### 5.1 用 fdisk 进行硬盘分区

在各自的操作手册中都有不同版本的 fdisk 的资料。如果没有特别注明的话，本章中所说的 fdisk 是指 fdisk 的 UNIX 版本。

fdisk 是交互方式的，它利用菜单来显示选择项。此处给出了 fdisk 菜单：

1. Display Portition Table
2. Use Entire Disk For UNIX
3. Use Rest of Disk For UNIX
4. Create UNIX Portition
5. Activate Portition
6. Delete Portition

Enter your choice or 'q' to quit

fdisk 实用程序允许用户在硬盘上为自己的操作系统建立独立的区域（分区），硬盘被分为若干磁道，磁道的数量取决于硬盘的大小。

一个分区由一组磁道组成，一个硬盘包含的分区数可以多达 4 个，各个分区可以有不同的操作系统和相应的目录及文件。

fdisk 命令允许用户指明当前“活动”分区，其含义是：当用户打开计算机（引导）时，安装在活动分区中的操作系统将开始运行。如果想使用 UNIX 系统，则必须将 UNIX 的分区指定为活动分区。

fdisk 命令允许用户指定每个分区的磁道数，这将随用户硬盘的大小而变化。运行 UNIX 系统时，建议使用不小于 40M 字节的硬盘。UNIX 分区的大小也取决于想建立的

软件包的数量。除了必须建立的软件包之外，还应为用户文件和其他软件包留出空间。

fdisk 命令允许用户指定分区从哪里开始，但不允许构造重叠的分区。并非一定要把 UNIX 系统安装于第一个分区上。

DOS 分区总应该位于硬盘起始处，从 1 柱面开始。之所以不从 0 柱面开始 DOS 分区是因为 DOS 的主引导块在 0 柱面，如果从 0 柱面紧接着主引导块的地方开始 DOS 分区，则 DOS 分区会变为不可访问分区。

如果在一个硬盘上安装了 DOS 之后还想建立 UNIX 分区，那就从该硬盘下一个柱面的起始处开始 UNIX 分区。确定下一个柱面起点位置的方法是：查出 DOS 分区的最后一个磁道号，从下一个磁道顺次往下找，直到磁道号能被硬盘的磁头数整除为止。该磁道即是 UNIX 分区的起始磁道。例如，如果用户的硬盘有 5 个磁头，而 DOS 分区结束于 103 磁道，则应从 105 磁道开始 UNIX 分区。

在运行 UNIX 系统时，UNIX 分区的设备名是 /dev/hd0a。

fdisk 的一种选择是将分区的当前状态以表格形式显示（显示选择分区表），对于每个分区，此任选项表格列出分区是否“活动”，开始磁道，结束磁道，使用磁道的条数以及相应的操作系统是什么。如果输入显示分区表任选并按下 RETURN 键，则可以看到这样的分区表：

Current Hard Disk Drive: /dev/hd00

Partition	status	Type	Start	End	Size
1	Inactive	DOS	005	398	393
2	Active	UNIX	400	1219	819

Total disk size: 1229 tracks (9 tracks reserved for masterboot and diagnostics)

在建立了独立的 UNIX 和 DOS 分区之后，就有三种方法转换操作系统（从 UNIX 到 DOS）：

- 在开机引导提示时输入“dos”字样
- 使用含有引导 DOS 所必需文件的软盘来引导 DOS 操作系统
- 用 fdisk 命令来改变当前活动分区

在此推荐用户使用前两种方法来引导 DOS。从软盘引导或者在引导提示时输入 dos 字样的方法比起经常使用 fdisk 命令修改活动分区的方法来，具有简单、迅速的优点，而且使用起来更为安全。

在使用前两种方法引导 DOS 时，虽然操作系统已经转换，但 UNIX 分区仍然是活动分区。如果用的是 fdisk 方法，则 UNIX 分区成为不活动分区，直到将操作系统转换回来。

使用引导提示方法时，在引导提示时输入 dos 字样：

```
sco system V / 386
```

```
Boot
```

```
:
```

在使用软盘引导 DOS 时应遵循以下步骤：

1. 确保全部用户都已退出系统。

2. 运行 shutdown (ADM) 关闭 UNIX 系统。这个命令通知所有用户系统要关闭。终止所有进程，然后关闭系统。

3. 一旦 UNIX 系统关闭，就将 DOS 引导盘插入主（引导）驱动器。

4. 引导 DOS。

5. 要回到 UNIX 分区，应把软盘从软盘驱动器中拿出，按下 <CTRL> <ALT> - <DEL> 键，或者 Reset 键，或者关闭主机，再打开。因为此时 UNIX 分区仍是活动的，所以将引导 UNIX 系统。

请记住，如果有一个活动的 UNIX 分区，并且从软盘引导 DOS，则可以转换到 C 盘以便使用 DOS 文件。

转换操作系统的另外一种方法是运行 fdisk，将当前活动分区由 UNIX 转换成 DOS，然后在关闭 UNIX 之后（见上面的步骤）从硬盘引导 DOS。这时不再需要 DOS 的启动软盘，因为 DOS 在硬盘的 DOS 分区中。

要转回 UNIX 系统，应在 DOS 下运行 fdisk，使 UNIX 分区活动，然后按下 <CTL> <ALT> <DEL> 键，或者按 Reset 键，或者关闭主机再打开，重新引导 UNIX 分区。

因为要运行 UNIX，UNIX 分区必须是活动的，所以不能用引导软盘来启动 UNIX 系统。这第二种方法适用于偶尔改变一下操作系统的情况。

硬盘设备名如下：

/dev/dsk/0sd

/dev/rdisk/0sd

/dev/dsk/1sd

/dev/rdisk/1sd

名字很类似 /dev/hd0a（活动磁盘分区），由磁盘驱动程序确定哪个分区是 DOS 分区，并把它用作 hd0d 和 hd1d。这意味着从 UNIX 分区运行的、使用 UNIX 分区的软件不必知道哪个分区是 DOS 分区，这由硬磁盘驱动程序来确定。

## 5.2 在 DOS 系统上安装 UNIX

如果想在以前已安装 DOS 的硬盘上安装 UNIX，则要遵循以下的步骤：

1. 把硬盘上所有的 DOS 文件和目录拷贝到软盘上，也可以拷贝到其它的存储介质，如磁带、磁鼓上。

2. 在 DOS 下运行 fdisk。如果硬盘上有足够的自由空间留给 UNIX 分区，则转第 4 步。否则删除 DOS 分区，然后再重建。在硬盘上留出足够的空间给 UNIX 系统及其他打算安装的 UNIX 软件。

3. 将备份存储介质（软盘、磁带等）上的 DOS 文件拷贝回硬盘上新建立的 DOS 分区。保存备份存储介质，以便在发生错误时不至于丢失数据。

4. 关闭计算机。

5. 按步骤安装 UNIX 系统。（见第一部分第二章）

安装时将看到一条警告信息，指出硬盘内容将被破坏。不过不必担心，因为已经拷贝了 DOS 文件并已将它送至新的 DOS 分区。正在建立的新分区包括了 UNIX，安装过程

只是把信息写到 UNIX 分区中。

6. 在安装过程中将调用 fdisk 对硬盘进行分区。应注意给 UNIX 指定一个足够大的分区。

7. 在 fdisk 的“活动分区”任选项下选择 UNIX，以指定 UNIX 作为活动分区。

8. 完成 UNIX 系统的安装。

---

注：UNIX 下的 fdisk 显示 DOS 分区为“DOS”，而 DOS 下的 fdisk 显示 UNIX 分区为“other”。

只能用 DOS fdisk 来建立 DOS 分区，同样，只能使用 UNIX fdisk 来建立 UNIX 分区。

需要注意的是，DOS fdisk 以柱面为单位报告分区大小，而 UNIX fdisk 则是以磁道为单位来报告大小。具体硬盘的柱面数目在硬盘手册上都可以查到。

---

### 5.3 在两个硬盘的环境下使用 UNIX 和 DOS

计算机总是从第一个硬盘上的活动分区中引导操作系统的。UNIX 系统必须从第一个硬盘上引导。如果有两个硬盘，则有若干种方法来配置系统。这里对其中两种加以讨论。

第一种方法是将第一个硬盘全部划分给 UNIX 分区，然后使用 DOS 启动盘片引导 DOS，并且指定：

A>A: C:

将 DOS 分区转换到第二个硬盘上，这里 C: 表示第二个硬盘。这种方法只适用于 DOS 较新版本，因为早期的版本只能处理一个硬盘。

---

注：如果将一个硬盘指定 DOS 使用，该硬盘必须在 DOS 配置中说明。

---

另一种实现方法是在第一个硬盘上保留一块小的 DOS 分区，并且指定它为活动分区。在这种配置下，计算机总是引导 DOS。如果想从硬盘引导 UNIX 系统，则必须首先改变活动分区。

如果将第二个硬盘全部用于 DOS，这时如果想使用 UNIX 的 DOS 实用程序 (doscp, dosls, doscat 等等)，就运行 mkdev hd 来为第二个硬盘建立设备文件。如果不打算使用这些实用程序来访问第二个硬盘上的 DOS 文件，则不必运行 mkdev hd。

---

注：如果要使用引导盘来引导第二个操作系统，必须先做一个引导盘的备份。

---

### 5.4 从硬盘上删除一个操作系统

在某种情况下也许不再需要已经安装在硬盘上的某一操作系统了。这时如果想将它删除，应使用相应版本的 fdisk。删除 UNIX 分区应使用 fdisk 的 UNIX 版本，删除 DOS 分区应使用 fdisk 的 DOS 版本。删除分区即意味着删除分区中的内容，所占空间即成为未分配空间。



这部分未分配空间可以用来另外增加一个 UNIX 分区或 DOS 分区，也可以用来扩大现有的分区。扩大现有分区要求重新安装操作系统，对 UNIX 分区来讲，还必须使用 divvy (ADM) 来重新建立分区中的文件系统。

### 5.5 访问 DOS 的实用程序

在使用 UNIX 的存取 DOS 实用程序之前，必须保证硬盘上有一个可启动的 DOS 分区或者有一张 DOS 软盘。比如，只有在 DOS 软盘可引导或者硬盘上有 DOS 分区的条件下才能将硬盘上 UNIX 分区中的文件转移到 DOS 软盘上。

还可以使用 UNIX 的 dd (C) 和 diskcp (C) 命令来拷贝和比较 DOS 软盘。UNIX 的 dtype (C) 命令用来判别软盘的类型。(不同的 DOS 和 UNIX 类型)

另外，文件 /etc/default/msdos 描述哪一个 DOS 文件系统 (例如，A: B: C: ...) 对应于哪一个 UNIX 设备。

---

注：不能从 UNIX 系统直接运行 DOS 程序。

---

UNIX 系统不能记录硬盘 DOS 区的坏道情况。如果 DOS 区域出现坏道，那么企图访问坏道区的操作，例如 dscp，就会失败。这种情况下，将显示信息指出“固定盘错误”。

对于较小的文件，可以将出错区域的文件拷贝到 DOS 下的另外一个地点，然后访问拷贝的新文件。

---

注：在使用 DOS 实用程序访问 DOS 分区中的文件时，可能会出现错误信息“bad media byte”，这条信息表示硬盘的 DOS 分区不可引导。要使得该分区可以引导，应先将 DOS 分区的所有文件备份下来，用软盘启动 DOS，然后使用下面命令对 DOS 分区进行格式化：

```
format C: /S
```

现在重新安装 DOS 文件即可。

---

DOS 文件的文件和目录参数格式为：

设备：名字

其中设备是指包含 DOS 软盘或 DOS 分区的特殊设备文件的 UNIX 路径名，而“名字”是指 DOS 文件或目录的路径名，例如：

```
/dev/fd0: /john/memos
```

表示文件 memos 是在目录 john 中，而二者都在设备文件 /dev/fd0 (主驱动器的 UNIX 特殊设备文件) 中。没有“设备:”参数的文件作为 UNIX 文件处理。

为方便起见，用户设置缺省文件 /etc/default/msdos 能够定义可能用到的 DOS 软盘驱动器名，以替代不太方便的 UNIX 特殊设备文件路径名。例如，可以定义如下：

```
A = /dev/fd096ds15
```

```
B = /dev/fd048ds9
```

```
C = /dev/hd0d
```

D = /dev/hd1d

定义完成之后，在指定 DOS 文件或目录路径时即可用驱动器标志“A:”来替代特殊设备文件 /dev/fd0 (96ds15 是缺省值)。如：

/dev/fd0: /john/memos

即可替换为：

A: /john/memos

驱动器标志 B 表示低密 (48ds9) 的主驱动器，而标志“C:”和“D:”分别表示主硬盘与第二硬盘上的 DOS 分区。

---

注：如果出现信息“cannot open /dev/hd0d” (不能打开 /dev/hd0d) 或者类似信息，就检查所涉及特殊设备文件的用户权限。作为超级用户，可以用 chmod 命令来改变权限，如：

chmod 666 /dev/hd0d

这条命令授权所有用户对特殊设备文件 /dev/hd0d (即主硬盘的 DOS 分区) 进行读写操作。

---

## 5.6 在 UNIX 系统上安装 DOS 文件系统

除了随操作系统一起提供的对 DOS 文件进行操作的 DOS 实用程序之外，还可以通过安装 DOS 文件系统的方法从当前 UNIX 系统随意访问 DOS 文件。

这意味着 DOS 文件可以不必拷贝到 UNIX 文件系统中，可以直接进行编辑或检查。需要注意的是 DOS 文件 and 应用程序不能直接在 UNIX 下运行。如果想在 UNIX 下运行，就要用 VP/ix 或者引导 DOS 分区。当然，数据文件、字符文件可以直接检查、拷贝或编辑。

---

注：建立 DOS 文件系统不能使用 mkfs (ADM) 命令，因为安装是针对已经存在的 DOS 文件系统而言的。

---

要安装 DOS 文件系统，要求内核中有某些支持，如果没有，必须首先用 mkdev (ADM) 命令将这些支持加入内核中。确信自己是从根目录登录的，然后键入以下命令：

mkdev dos

如果使用 sysadmsh，则从以下路径进行：

System → Configure → Kernel → Dos

这一命令加入安装所需的特性到内核中并且重新连接内核。(如果连接部分还未装入就会要求用户装入)。重新启动之后即可以按照下面讲到的方法安装 DOS 文件系统。

操作系统在处理 DOS 文件系统时加上了 UNIX 文件系统特有的某些属性，而没有改变 DOS 文件系统的实际文件。由于 UNIX 文件系统结构较为高级并且在多用户环境下工作，所以包含许多在 DOS 下没有任何意义的限制条件，它们是：

- 文件的所有权；
- 文件的访问权限；

- 特殊文件（管道、设备文件，等）；
- 链接（Links）。

---

注：其它应用程序和操作系统在安装和访问 DOS 文件系统时也采用给 DOS 文件增加某些属性的方法，但其中很大一部分都在某种程序上修改了 DOS 文件。考虑到可移植性，在这里没有对 DOS 文件做任何修改或补充。安装文件系统完全是通过文件系统开关（FSS）机制实现的。

---

为了使 DOS 文件可以随意访问，在安装 DOS 文件系统时对访问权限、文件所有权等都采取“全开放”方式。

DOS 文件系统安装命令格式如下：

```
mount -r -f DOS /dev/hdxy /mountpoint
```

在此各参数含义如下：

x: 硬盘号

y: 硬盘分区号

mountpoint: root 文件系统中 DOS 文件系统将要安装于其中的目录名。

r: 该标志使文件系统以只读方式安装，这可以防止在安装过程中破坏 DOS 文件系统。

在使用 mount 命令时，必须指定硬盘号和所用硬磁盘分区号。

也可以安装 DOS 软盘，下例给出了用 96tpi 软盘安装于 /mnt 目录的情况：

```
mount -r -f DOS /dev/fd096 /mnt
```

操作系统中包括 fsck (ADM) 实用程序可工作于 DOS 文件系统。该实用程序将 DOS 的文件分配表 FAT 与文件系统中的文件协调起来。当执行 fsck 时，可以自动检测 DOS 文件系统。

只有根目录才能安装文件系统。用户对文件系统的访问由根目录指定到 DOS 文件系统的权限和所有权进行控制。由于前面讨论过的 DOS 的局限，DOS 不能识别访问权限或所有权。安装在 UNIX 系统上的 DOS 文件有以下特性：

- 文件系统的权限和所有权由安装点控制。例如，若根建立了一个权限值为 777 的安装点，则所有的用户都可以读 / 写文件系统的内容。如果该安装点为根所有，则 DOS 文件系统内的所有文件或其他用户创建的文件均为根所有。

- 一般文件的权限为 0777，表示读 / 写文件，若为 0555 表示只读文件。这保持了 DOS 文件系统的一致性。如果一个用户能够访问文件系统，他就会受到 DOS 目录结构下的访问权限的限制。访问权限为只读或读 / 写。在建立一个文件时，其权限值由建立者的 umask 决定。例如，假设用户的 umask 为 022，则所建立文件的权限即为 777。举例如下：

例 1、建立一个文件。文件的权限由属主者的 umask 所决定。如果 umask 为 022，则在 DOS 分区上产生一个权限为 777 的文件。因为属主没有将写位屏蔽掉，所以文件是可读 / 写的。

例 2、检查一个已经存在于 DOS 分区的文件。所看到的权限是 UNIX 安装点权限值和 DOS 文件权限值逻辑“与”运算的结果。如 UNIX 安装点权限为 750 而 DOS 文件权限

为 555，那么所得到的权限将是 550。这与 umask 毫无关系。

- 在 DOS 文件系统中每个文件只有一种连接；
- 在 UNIX 系统中，一个文件可以同时为多个用户所访问，这是通过锁控制机制实现。在安装的 DOS 文件系统中也可通过这种机制使得两个用户同时编辑同一个文件。

由于 DOS 文件没有作任何改动，所以在 UNIX 系统上编辑 DOS 文件时就会看到回车符（^ M）。（UNIX 系统中在一行结束时有换行符，而 DOS 系统中多一个回车符）。dtox（C）和 xtod（C）命令是在行尾格式上统一二者的最简单的方法。dtox 用于从 DOS 格式转化成 UNIX 格式，而反之则为 xtod 命令。

另外还必须注意到在文件名、修改时间和备份实用程序几个方面的要求。

#### 文件名

文件名及其转换规则在 dos（C）手册中可以查到。标准 DOS 对于文件名中非法字符的规定在此也适用。

#### 修改时间

在从 UNIX 分区访问 DOS 文件时，文件的建立、修改和访问时间采用格林威治时间。这预示着 DOS 文件系统中建立的文件在 DOS 和 UNIX 下的时间是不一致的。

#### UNIX 备份实用程序

backup（ADM）和 xbackup（ADM）实用程序不能用来实现安装在 UNIX 上的 DOS 文件系统的备份。可以使用 DOS 实用程序以及其他拷贝程序。

#### 非标准盘上的 UNIX 和 DOS

UNIX 系统为“非标准”硬盘提供支持。关于“非标准”一词的意思是指在计算机的 ROM 中没有正确的磁盘参数登记项。

为非标准硬磁盘所指定的正确参数存放在主引导块中，即引导硬盘的第一个扇区中。在安装 UNIX 过程中，可以指定硬盘的特性并且把这些特性写入主引导块的剩余部分。不管哪个操作系统是“活动的”，专用的主引导块都将重置硬盘参数为指定值。这种机制为 UNIX 和 DOS 提供了对非标准硬盘的支持。

虽然在 DOS 下专用的主引导程序支持非标准硬盘，但不能用 UNIX 系统在硬盘上安装 DOS。如果正在使用的是非标准磁盘，那么就认为用户已经有某种方法将 DOS 文件传输到硬盘上了。

除非正在改变当前活动分区，否则应该只使用 UNIX fdisk 来处理硬盘分区表。在 UNIX 安装之后，使用由硬盘厂商提供的 DOS fdisk 或者通常的 fdisk 可能会破坏非标准硬盘的特性，使之不能使用。

## 第六章 系统安全性维护

每个计算机系统都需要防止对计算机、磁盘和系统文件进行非法访问。本系统提供了比 UNIX 操作系统基本安全措施更强的安全措施。在设计上符合 Department of Defense's Trusted Computer System Evaluation Criteria (也叫做 TCSEC 或 the Orange Book) 定义的 C2 可靠级。

本章介绍如何运用安全措施来维护可靠的系统。包括如下内容：

- 系统安全性概述
- 运行可靠的系统
- 保护系统的数据
- 检测系统所受的损害
- 处理文件系统的讹误
- 与安全性有关的信息
- 在可靠系统上的守护操作
- 允许用户安装文件系统

---

注：如果硬件和介质不保护起来，那么操作系统的安全措施也是没有用的。必须保护好计算机本身和所分配的软盘以及其它各种备份介质，以防非授权人员访问。应该遵守如下规则：

- 当操作员不在时，必须把系统锁起来。
  - 整理并锁好所有的备份介质。
  - 保护通信线 (uucp, 以太网, 和终端), 以防非授权访问。
- 

### 6.1 什么是可靠系统

因为计算机系统不可能完全没有危险，所以我们只能把系统称为“可靠的”而不是“安全的”。可靠系统对于信息的访问具有很高控制权，并且还提供了防止（或至少是能够发现）非授权访问的机制，同时还必须确保这些机制正常地发挥作用。C2 可靠级的意思是系统的设计在安全性策略、责任性、质量保证、检测以及资料等方面都符合特定的规则。

本系统上所具备的安全特征比标准 UNIX 系统有所扩展。在增加对用户和系统信息保护等特征的同时，本系统依然保持与普通 UNIX 完全兼容的机制。本章讲述的对系统信息的维护和保护在管理工作中占很大的比例。

在系统安装时，要设定在系统上使用的安全性缺省设置（可靠的或放松的）。另外也可以在系统上根据需要，自己设定任何一种组合的缺省设置。

作为管理员，为维护一个可靠系统，各种动作都必须很严格，任何一点小错都会造成系统的疏漏。为了在管理位置上更加称职，必须懂得系统的安全策略，懂得系统是如何在系统信息（数据库）的控制下工作，懂得自己所做的任何操作是如何影响用户和管理员动作的。

下面将给出一个可靠系统的基本概念。作为一个管理员，必须了解这些概念并知道为了正常运行系统，那些与安全性相关的信息都保存在哪儿。

#### (1) 可靠计算基础库 (Trusted Computing Base) 或 (TCB)

有一组被称为可靠计算基础库 (即 TCB) 的软件负责维护与安全性相关的那部分系统状态。TCB 由 UNIX 核心 (操作系统的核心) 及引用和维护与安全性相关的数据的可靠工具组成。这个可靠计算基础库执行系统的安全策略。安全策略监视和保卫“主体” (譬如进程, 它是一段正在系统上运行的程序) 和“客体” (譬如文件, 设备以及在各进程间所交换的对象) 之间的相互作用。对于 C2 级, 这组策略包括分点访问控制 (Discretionary Access Control, 也可略为 DAC) 和目标复用, 后者是指在一个存储物体被分配之前必须把其中的信息清除掉。管理员所用到的许多软件都是系统 TCB 的一部分。sysadmsh 中也提供了帮助管理员维护可靠计算基础库的菜单驱动的管理界面。

#### (2) 责任性 (accountability)

如果一个动作可以追溯到某一特定的人, 那么这个动作就是可负责的。在可靠系统上, 所有的动作都可追溯到一个可负责的人。大多数 UNIX 系统缺少好的责任性, 因为有一些动作不能追溯到一个人。例如, 假用户帐号, 像 lp 或 cron, 都是匿名执行; 它们的动作只有在转化为系统信息后才能被发现。正如后面将讲到的, 可靠的 UNIX 系统可以通过把每一个帐号都与一个真实用户联系起来, 通过对每一个动作进行审计, 和把每一个动作与一个特定的系统上的用户联系起来, 来改善其责任性。

在典型的 UNIX 系统上, 每一个进程都有一个真实而有效的用户 ID, 同时还有一个真实而有效的组 ID。一个带根 (root) 有效用户 ID 的进程可以把这些标识符设置给任意用户。C2 可靠级要求 TCB 能够识别出不同的用户并由此确定每个人的责任。可靠的 UNIX 系统对用户标识符的概念进行了扩展, 增加了一个叫做“注册用户标识符” (login user identifier, 简略为 LUID) 的另一个标识符。LUID 是一个在与一个用户相关的进程上的永久标记。LUID 标识出对某进程对话区负责的用户。一旦被打上了标记, LUID 就不会被任何人改变。子进程继承其父进程的 LUID。

#### (3) 分立访问控制 (discretionary access control)

分立访问控制 (discretionary access control, 简略为 DAC) 用来判断一个用户是否有权访问他所希望得到的信息。那个信息在用户进程想使用的一个“目标” (文件、设备等) 中。在大多数 UNIX 系统中, 目标保护是根据进程的用户和组与这个目标的主人和组及其它状态位之间的关系来实施的。这些目标的属性都在其主人的分立掌握下, 这些主人可以改变这个文件的保护位, 甚至可以丢弃这个文件 (修改所有权)。可靠的 UNIX 系统对用于 UNIX 文件访问权限的标准分立访问控制进行了扩展, 并对如下各项进行了限制:

- 给文件设置 SUID (set user ID on execution) 位的能力
- 改变文件所有权 (使用 chown (C)) 的能力
- 可能用错 SUID 和 SGID, 和通过在文件写入之后清除这些位来固定其访问权限。

#### (4) 授权

授权是执行某种操作所需的用户属性。大多数 UNIX 系统是根据简单的文件访问权限或根据请求访问的那个进程是否隶属于根来决定进行所有的访问。根帐号能够执行其它

进程所不能做的系统动作。可靠的 UNIX 系统定义了两类授权权限：核心和子系统。核心权限是与进程有关的。它们允许进程执行某种动作，只要这个进程具备所要求的特权。子系统权限是与用户有关的。它们允许用户利用子系统命令（可靠的工具和程序）来执行一个特殊的动作。“子系统”是一组相关的文件、设备和能够完成一种特定功能的命令的集合。例如，lp 子系统包括要打印的假脱机文件、打印机设备和一些命令（譬如用于协助维护子系统的 lpadm (ADM) 命令)。

每个进程都有一个“权限集合”，核心权限就保存在其中。这个权限集合是一个特权表，如果一个权限在表中，那么相应的动作就被允许执行，否则就不能执行。权限可以由系统默认设置，也可为专门的用户特殊设置。

#### (5) 身份识别和口令鉴别 (Identification and Authentication, 简略为 I&A)

当用户在一个非可靠 UNIX 系统上注册时，要进行有限的身份识别和口令鉴别。系统在口令数据库 (/etc/passwd) 中查找这个用户名。如果用户名找到了，系统就会在口令数据库的登记项中与加密的口令版本进行口令比较。还有一些关于口令的字符以及所实施的修改口令的规则，但是这些规则看起来都不能十分有效地防止非法入侵。

可靠系统对标准 UNIX 的 I&A 机制进行了扩展。有更多的规则用来规定能够使用的口令种类。有新的建立和修改口令的过程。对口令数据库中的某一部分进行定位和保护也不同于其它的 UNIX 系统。管理员对于注册进程可以实施更多的控制。还有一个专门的角色，叫做鉴别 (帐号) 管理员 (子系统权限是 auth)，负责系统这方面的维护。

#### (6) 审计 (Auditing)

大多数 UNIX 系统在其记帐子系统中只保留有限的系统动作记录。这个记帐子系统为每一个完成的进程写一个单独的记帐记录。可靠的操作系统提供了一系列扩展的报告 (或称动作的“足迹”)。在这个足迹中包含主体与客体之间进行的每一次访问 (成功的和不成功的) 以及主体、客体和系统特性每一次变化的记录。这个审计子系统是由一个叫做审计管理员的独立的角色控制的 (子系统权限为 audit)。审计管理员决定有多少信息要记录、决定所记录信息的可信度，还要对所搜集的信息进行维护。这个审计子系统给审计管理员提供了一份系统动作的全面历史。这可以帮助管理员识别在系统中发生了什么、什么时候发生的、有谁参与等等。

#### (7) 受保护子系统

UNIX 系统提供了设置用户 ID (SUID) 和设置组 ID (SGID) 机制。(用户 ID 和组 ID 的设置能力是通过 setuid 和 setgid 系统调用和文件的 setuid 和 setgid 访问权限控制位来实现的。) 利用这些可以建立维护专用信息的程序。这些信息也只能由在这个程序中所完成的操作来访问和修改。可靠的 UNIX 系统定义了几个“受保护子系统”。每一个这种子系统由一组专用信息 (文件和 / 或数据库)、一些相关设备以及用来维护这些信息的工具和命令组成。受保护子系统利用 SUID / SGID 机制来保护其专用文件、数据库和设备不受非法访问。可靠系统在如下几方面扩展了受保护子系统的概念：

- 它对于用户和组实施更加精确的控制，这些用户和组对某组系统资源 (专用信息) 进行维护。

- 它保留了一个单独的用户数据库，这些用户被允许运行程序来维护其专用信息。

- 它并不要求用户以子系统管理员的身份注册，但是又允许他们使用这个数据库对子

系统权限进行检查。这符合对子系统程序所执行的所有动作的全部责任性要求。(如果用户都注册到匿名帐号中去执行系统管理任务,那么就无法确定是谁执行这些动作。)

## 6.2 运行一个可靠系统

现在已经选择了在本系统上所使用的安全体制(可靠的或放松的)。即使没有选择运行可靠系统,管理员依然可以考虑下面的选择:

- 指派单独的人管理系统,或指派一组人管理系统,使每个人分别负责一个子系统。
- 给需要附加特权的用户分配核心权限。
- 决定希望以何种严格程度来控制 and 监视对系统的访问。
- 决定是否使用审计特征。

### 6.2.1 分配管理角色

必须首先确定的是谁来维护这个可靠系统。可以只有一个能注册于根的全权的超级用户,也可以把部分管理职权赋予其它用户,但只能分配给他管理系统操作的一个方面所需的权力,分配的权力不要超出所需。在可靠的 UNIX 系统上,管理任务被分成几个逻辑角色。每个角色都负责维护系统的一个方面。专门管理角色的概念(以及相关的任务和责任)对于理解可靠的操作系统是十分关键的。每个逻辑角色可以分配给一个人也可以分配给几个管理人员。每个子角色都有一个专门的权限和一个 sysadmsh 选择。这种联系,再加上复杂的跟踪系统,就可以使管理员保持一个管理动作的清晰记录。这有助于防止问题的发生,并且也易于识别和解决已存在的问题。

为了完成与一个管理角色相关的任务,管理员必须拥有适当的子系统权限。表 6.1 列出了子系统、相关权限以及每个角色所维护系统的区域。

了解每个角色的职责以及他的动作对于系统安全性的影响,是十分重要的。应根据系统所保存信息的敏感程度、用户的协作和专业领悟程度、内在的和外在的非法侵入与误操作的威胁,来配置和运行系统。

表 6.1 受保护子系统以及管理角色

角 色	子系统权限	区 域
身份鉴别管理员	auth	系统帐号
打印机管理员	lp	行式打印机子系统
cron 管理员	cron	at 和 cron 子系统
审计管理员	audit	审计数据库和审计足迹
操作员	backup	文件系统备份
系统管理员	su * sysadmin	su 访问超级用户帐号 使用 integrity(ADM)程序
终端管理员*	terminal	终端设备允许权
内存管理员*	mem	访问系统内存



- \* 这些并不是真正的管理员，后面会进行解释。
- \* \* 这种权限可访问根帐号，但并不拥有指派给根的权限（也就是说，其权限并没有改变）。

只有保持警惕并正确使用系统才能够保持系统可靠，也才能保护系统的完整性。

为了分配子系统权限，可以做如下的 sysadmsh 选择：

Accounts > User > Examine: Privileges

注：你可能已经注意到了，每个主要的子系统权限看起来都与那个子系统的组名相同。这表示如果一个用户是一个子系统组的成员，就暗示他有访问这个子系统中文件的能力。然而，这并未授予所需的子系统权限。实际上，绝对不应该使一个用户成为一个子系统组的成员，因为这会对实际的数据文件造成威胁。最好使用正确的子系统权限来允许用户访问相应的子系统。

### 6.2.2 利用 sysadmsh 管理子系统

某些子系统只是逻辑区域而不是真正的系统管理区域。例如，内存子系统在本质上是不可管理的，但 men 权限的分配可以控制对核心内存结构的访问。其它子系统是需要管理的，也有相应的 sysadmsh (ADM) 选择。这些子系统可被分配给各个人，并且每个范围都必须提供文档。表 6.2 列出了必须管理的每个子系统、它们的 sysadmsh 选择以及有关的章节。

表 6.2 子系统和 sysadmsh 选择

受保护子系统	目的与角色	sysadmsh 选择	章节
行式打印机	打印机管理	Printers	“打印机”
备份	文件系统的副本	Backups	“文件系统的备份”
身份鉴别	帐号维护	Accounts	“用户帐号的管理”
Cron	作业调度	Jobs	“作业调度命令的授权使用”
审计	审计	System → Audit	“审计子系统”

本节将详细讲解各受保护子系统，在非可靠系统中许多由超级用户执行的功能在可靠系统中被分配给了这些受保护子系统。然而，有些功能依然需要由超级用户来完成。这包括安装和拆卸文件系统、遍历整棵文件树。只有超级用户可以做所有的操作。必须把根口令限制在很少的几个人范围内，把根帐号分配给一个负责任的用户。（详见本书中“用户帐号的管理”一章。）

### 6.2.3 分配核心权限

正如前面所讨论的，可靠的 UNIX 有两类权限：核心和子系统。表 6.3 是核心权限表。

表 6.3 核心权限

权 限	作 用
configaudit	修改审计参数
writeaudit	为审计结果写审计记录
execsuid	运行 SUID 程序
chmodsugid	给文件设置 SUID 和 SGID 位
chown	改变对象主人
suspendaudit	暂停一个进程的审计

大多数用户只需要 execsuid 核心权限就可以进行日常工作了。如果用户要建立带 SUID 或 SGID 位的文件，他们就必须有 chmodsugid 权限。要改变文件的所有权（丢弃它），chown 权限就是必需的。如果一个用户没有这个权限，那么文件的所有权就只能由根来改变。审计核心权限（configaudit, writeaudit, 和 suspendaudit）绝不应该分配给审计管理员以外的其它人。它们是给被作为可靠应用软件程序的设计和运行使用的。

注：NIST FIPS 151-1 标准需要严格的 chown。如果希望遵循 NIST FIPS 151-1 的要求，就不应该把 chown 权限分配给用户。

为了分配核心权限，可以进行如下 sysadmsh 选择：

Accounts→User→Examine:Privileges

被分配了管理角色的用户为了完成子系统所要求的工作还必须拥有某种核心权限。所需的核心权限列于表 6.4。如果在整个系统使用了“C2”或“放松的”缺省核心权限，就只需要给审计、cron 和系统管理子系统再分配另外的核心权限；chown 和 execsuid 已经由缺省值分配了。

表 6.4 子系统所需的核心权限

子系统	所需的核心权限
audit	configaudit,suspendaudit,execsuid
auth	chown,execsuid
backup	execsuid
lp	chown
cron	execsuid,chown,chmodsugid
sysadmin	execsuid,chown,chmodsugid

#### 6.2.4 系统访问的控制

在一个可靠系统上操作的一个重要方面是查找与安全性有关的问题。系统访问的限制

机制分为三类，对这三类限制都可以专门制定并提出报告：

- 口令限制
- 终端使用限制
- 注册限制

#### (1) 口令限制

身份鉴别管理员可以允许用户自己选取口令也可以由系统为用户生成口令。口令一旦选定了，还可能要接受简单的或深入的可见性检查，这是由身份鉴别管理员选择执行的。

口令的生命期定义如下：

- 口令有效
- 口令退役

用户仍然可以注册，然后改变这个口令（如果授权允许这么做）。

- 口令死亡

用户被锁在外面，管理员必须打开这个帐号。

如果不允许用户改变口令，那么新口令就只能由系统管理员分配。如果帐号被封闭了，用户必须通知管理员。

在一个要求定期改变口令的系统上，用户为了符合这个要求，经常先改变他们的口令，然后又立即改回他们以前使用的旧口令。为了防止用户这么做，身份识别管理员可以设定一个口令改变的最小时间，在这个时间之前，用户不能改变口令。所有这些参数可针对整个系统（系统缺省值数据库）或针对每个用户（受保护的口令数据库）进行修改。请看本书中“用户帐号的管理”一章中的“改变缺省口令限制”和“改变用户口令或口令参数”。

口令报告可以用来找到其口令将要退役的用户，以便管理员能够警告他们防止帐号被封闭。如果系统管理员不在的话，口令退役就会成为很麻烦的事。如果系统并不是每天都有人看护，就可能需要相应地延长口令生命期参数。

#### (2) 终端限制

终端是系统的门户。除了借助于帐号口令之外，也可以把终端保护起来，以防止打入系统的非法企图。可以定义注册失败的最大次数，以减少破译出帐号口令的可能性。那些超过了最大注册失败次数的终端会被封闭，而只有帐号管理员才能打开它。另外，还可以设定两次注册之间的最小时间间隔，这可以进一步阻止破坏口令的企图。（不仅对终端，对帐号也可以定义类似的限制。）要改变或检查终端限制，请参照本书中的“用户帐号的管理”一章中的“终端注册管理”。

#### (3) 注册限制

与终端类似，用户帐号也有与注册次数和作再一次努力的时间间隔有关的参数。要改变或检查注册限制，请参阅本书中“用户帐号的管理”一章中的“改变缺省注册限制”。

#### (4) 访问限制的状态报告

在本节中所讨论的每一种限制都有报告工具。例如，可以建立一个关于一个终端或一组终端的注册记录的报告，或建立一个其口令将要退役的用户帐号的报告。运行这些报告的过程可以在本书“用户帐号的管理”一章中的“活动报告的建立”中找到。

### 6.2.5 在系统上使用审计

审计保持了系统使用的详细记录，使管理员能够确定是否发生过对系统的干预事件（无论是成功的还是不成功的）。然而，审计需要更多的监督和磁盘空间，所以必须先判断在系统上审计是否有必要。审计在本书“审计子系统”一章有进一步的讨论。

### 6.3 保护系统上的数据

在本系统上，数据保护的第一道防线就是文件和目录的标准 UNIX 访问权限的使用。对于系统安全性来说，了解为保护文件和目录所设置的访问权限位是很重要的。系统上所建立的文件的缺省访问权限是由整个系统的 `umask (C)` 来管理的，也可为单个用户制定访问权限缺省值。

本系统还包括另外一些重要的文件系统特征，这些特征是对标准 UNIX 系统保护措施扩展。这些特征大大增强了系统的安全性。其中之一就是在对一个文件进行写操作时，SUID 和 SGID 位会被清除，并且在此过程中不需要系统管理员进行任何操作。另一个特征是主动的，意思就是管理员可以选择是否要使用它们。这些主动的特征会在下面讨论，包括目录的密封位、数据加密、从其它系统引进数据文件时要遵循的预防措施等特殊用法。

#### 6.3.1 写操作时 SUID / SGID 和密封位的清除

在程序执行时，文件的 SUID 和 SGID 访问权限位会改变进程的用户 ID 和组 ID。不应该允许普通用户使用这些位，它们的使用是由 `chmodsugid` 核心权限来限制的。（请看本章的“分配核心权限”。）可靠的 UNIX 系统保证在对文件进行写操作时，清除其 SUID 和 SGID。清除的目的就是为了防止用户利用程序的 SUID 或 SGID 位，因为用户无法以其它办法设置这些位。

在文件的访问权限中，SUID 位是以“S”来表示的。在图 6-1 中位清除演示了两次。

```
$ id
uid=76(blif) gid=11(guru)
$ ls -l myprogram
-rwsrwsrwt 1 root bin 1024 Jan 11 22:45 myprogram
$ cat sneakyprog > myprogram
$ ls -l myprogram
-rwxrwxrwx 1 root bin 10240 Mar 18 14:18 myprogram
$ ls -l anotherprog
-rws-----1 blif guru 83706 Dec 15 1987 anotherprog
$ strip anotherprog
$ ls -l anotherprog
-rwx-----1 blif guru 17500 Mar 18 14:19 anotherprog
```

图 6-1 位清除实例

例如，用户 blif (`id (C)` 已显示了用户身份) 首先使用 `cat` 来替换文件 `myprogram` 的内容。SUID 位在这个执行过程中被清除了。第 2 个例子是为了说明即使是对用户自己

的文件，这一位也要被清除。当 blf 对文件进行 strip 操作时（去掉一个编译过的二进制文件中的调试信息），SUID 位也被去掉。应该知道清除都发生在文件被替换时。可以通过调整安装程序，重新设置合适的模式。利用这个特征，管理员可以为用户程序设置这些位，这样就不必担心用户会把某一文件中的程序替换为另一个程序了。

对于目录来说，SUID 和 SGID 以及密封位是不能清除的。SUID 位对目录是无意义的，而 SGID 和密封位对目录是有意义的，它们确保目录保持不动。

### 6.3.2 密封位和目录

本系统的另一个重要改进是对目录使用了密封位。目录有了密封位就意味着只有文件的主人和超级用户能够把文件从目录中移去，其它用户没有移动该文件的权利。与文件不同，目录的密封位总是保持着，直到目录的主人或超级用户明确地删除该目录或对该目录执行 chmod (C) 或 chmod (S)。请注意，目录的主人只能删去密封位，但不能设置。

可以对所有公共目录设置密封位，从这个特征中可获得更多的安全性。任何非管理员都可对这些目录进行写操作。密封位与 umask 的默认值 077 解决了安全性较差的系统中的一大问题。同时，这两个特征也可防止其它用户修改或替换在一个公共目录中的任何文件。他们所能获得的信息只有文件名及其属性。

图 6-2 描述了这一体制的作用。在目录的访问权限中密封位是“t”。（在 UNIX 系统中，当前目录在文件表中表示为一个黑点（·），而两个黑点（..）代表这个目录的上一层目录）

```
$ id
uid = 76(slm) gid = 11(guru)
$ ls -al /tmp
total 64
drwxrwxrwt      2  bin   bin    1088   Mar   18   21:10  .
dr-xr-xr-x     19  bin   bn     608    Mar   18   11:50  ..
-rw-----     1  blf   guru   19456  Mar   18   21:18  Ex16566
-rw-----     1  blf   guru   10240  Mar   18   21:18  Rx16566
-rwxr-xr-x     1  slm   guru   19587  Mar   17   19:41  mine
-rw-----     1  slm   guru    279   Mar   17   19:41  mytemp
-rw-rw-rw-     1  root  sys    35     Mar   16   12:27  openfile
-rw-----     1  root  root    32     Mar   10   10:26  protfile
$ rm /tmp/Ex16566
rm: /tmp/Ex16566 not removed. Permission denied
$ rm /tmp/protfile
rm: /tmp/protfile not removed. Permission denied
$ cat /tmp/openfile
    Ha! Ha!
You can't remove me.
$ rm /tmp/openfile
rm: /tmp/openfile not removed. Permission denied
$ rm -f /tmp/openfile
$ rm /tmp/mine /tmp/mytemp
```

```

$ ls -l /tmp
drwxrwxrwt      2  bin    bin    1088   Mar   18   21:19  .
dr-xr-xr-x     19  bin    bin     608   Mar   18   11:50  ..
-rw-----     1  blf    guru   19456  Mar   18   21:18  Ex16566
-rw-----     1  blf    guru   10240  Mar   18   21:18  Rx16566
-rwxr-xr-x     1  root   sys     35     Mar   16   12:27  openfile
-rw-----     1  root   root    32     Mar   10   10:26  protfile

$ cp /dev/null /tmp /openfile
$ cat /tmp /openfile
$ cp /dev/null /tmp /protfile
cp:cannot create /tmp /protfile
$ ls -l /tmp
drwxrwxrwt      2  bin    bin    1088   Mar   18   21:19  .
dr-xr-xr-x     19  bin    bin     608   Mar   18   11:50  ..
-rw-----     1  blf    guru   19456  Mar   18   21:18  Ex16566
-rw-----     1  blf    guru   10240  Mar   18   21:18  Rx16566
-rw-rw-rw-     1  root   sys     0      Mar   18   21:19  openfile
-rw-----     1  root   root    32     Mar   10   10:26  protfile

$

```

图 6-2 密封位实例

被删去的文件只有用户 slm 拥有的文件，因为这个用户就是 slm。这个用户不能删除其它文件，即使是可访问的文件 /tmp /openfile。然而，这个文件本身所设置的访问方式却允许毁坏文件的内容；这就是要设置 umask 来保护数据的重要原因。/tmp /profile 的访问模式与 /tmp 上的密封位一起使用，就使得 /tmp /profile 不可侵入。

所有的公共目录都应该设置密封位。这包括如下目录，但不仅限于此：

- /tmp
- /usr /tmp
- /usr /spool /uucppublic

如果不敢保证，那么最好给每个目录都设置封位，任何目录都不要放弃。可以用如下命令给目录设置密封位，其中 directory 是目录名：

```
chmod u+t directory
```

### 6.3.3 数据加密的使用

在系统中，还可以利用 crypt (C) 命令来进行数据加密，以加强安全性。

---

注：数据加密软件不包括在所提供给你的软件中，它只能在美国境内根据需要给予提供。

---

### 6.3.4 引入数据

从别处引入系统的文件和文件系统如果不做适当的处理，就会对系统造成威胁。这部分讨论在向系统引入文件时所应进行的处理。

#### (1) 文件

不要对外界文件的访问权限想当然。不仅各系统的 /etc /passwd 和 /etc /group 文

件各不相同，而且不同系统上的安全策略也会规定不同的访问模式。当所引入的是系统文件时，这些考虑变得更加重要。

为了在引入文件后尽量减少管理员的干预和对它的清洗，应让系统上的每个人给归档程序使用不恢复其所有权的选项。这些文件应由其引入者所有。只有当 `cpio` (C) 程序的调用者是超级用户时，它才能改变这些文件的所有权。归档程序 (archive program) 一般要恢复文件原来的访问方式。对于这些文件除了设置超出所需的访问权限之外，还要设置 SUID、SGID 和密封位。上述所有这些情况都可能造成安全问题。

为了减少归档访问权限的影响，可以使用选项，使得不必抽取任何东西就可以检查其内容。例如，`tar` 的 `-tv` 选项和 `cpio` 的 `-tv` 选项都可以用来查看磁带上的文件的访问权限，为在恢复文件时可能造成的任何不良影响做准备。当拿到一个不熟悉的档案文件时，首先要把文件放入别人访问不到的目录。然后按照本系统的安全策略对其所有权、访问方式进行了调整，再把这些文件移到合适的目录中。

## (2) 文件系统

安装那些在别处建立或处理过的文件系统时应与引入文件做同样的考虑。文件系统也能随之带来两个主要问题。一是这个文件系统可能有讹误，二是文件系统中各文件的访问权限可能不为本系统所接受。无论是哪一种情况，安装一个不好的文件系统都可能造成系统性事故，在所引入的文件系统上的数据也会进一步受到破坏，或使其它文件系统受到破坏。这就是为什么 `mount` (ADM) 命令要保留给超级用户的原因。对于所有文件系统，在安装之前，都必须对它们执行 `fsck` (ADM) 程序。如果文件系统中包含系统文件，还应运行 `integrity` (ADM) 工具。

所引入的文件系统可能包含不适于本系统的文件访问权限。所引入的文件系统的原来的超级用户可能已经设定了所有权、密封位、特殊文件、SUID / SGID 位以及文件树等不适合于本系统的组合设置。还可能存在带有本系统所不允许的所有权和访问方式的特殊文件。

在安装之前，可以给 `ncheck` (ADM) 以 `-S` 选项来寻找存在潜在危险的 SUID 文件。文件系统与文件一样，在安装之前都应该先查看一遍。当文件系统第一次被安装时，最好把它安装在一个私有目录中，以便于在把它安装在它的最终位置上之前对它进行检查。要检查文件的组织、文件的所有者和访问方式、以及使用这个文件系统所期望达到的目的。

## 6.4 检查系统所受的损害

正如前面所讨论的，没有任何系统可以认为是绝对安全的。如果有些人使用了很显然的口令或整夜使其终端处于注册状态而自己离开，那么系统就会很容易遭到入侵，可见用户是这个体制的薄弱环节。系统在设计上可以正确地鉴别用户。另外对系统上要考虑其安全性的数据的访问也要根据子系统的授权权限。如果一个用户具备合适的权限，他才能够使用系统程序来修改安全性数据库（例如，审计管理员改变审计配置，或是帐号管理员改变用户口令）。

系统禁止非授权的用户对安全性数据库进行修改，在这种保护中，身份识别和口令鉴别是关键步骤。当有人要访问所需权限大于他所有的权限的帐号时，这个机制就要起作

用。在建立系统并尽量减小系统可能受到的损坏之后，其余的工作就是去检查系统是否受到破坏。破坏可能来自如下两方面：

- 有人掌握了其他用户的口令或访问了他们的帐号（譬如当有人离开时没有在终端上注销）。
- 有经验的用户对计算机系统本身进行了无监督访问。

#### 6.4.1 被盗口令

每次用户注册，系统就显示他们上一次注册的时间和日期。判别一个被盗口令的最明显的依据就是这个上一次注册的时间和日期与记忆中的不同。如果没注意到这一点，那么第二个依据就是用户文件被更改了。告诫用户要注意他们上一次注册的时间并且对任何异常现象都要立即报告，其中包括他们的文件所受的任何破坏事件。

管理员应该认真考虑对口令实施什么限制。一个普遍的（也是极其危险的）作法就是不给帐号加口令。虽然这也是系统所允许的，但我们极端不鼓励无口令的帐号。一旦有人注册到一个帐号当中，就很难防止他对那个帐号的危害以及对系统的进一步侵害。身份识别和口令鉴别过程是防止系统受到损害的第一道防线。

防止被盗口令的另一个武器是两次注册尝试之间的时间间隔以及对帐号和终端不成功注册的限制，这在“用户帐号的管理”一章中的“活动报告的生成”里有详细讲述。要注意如下事件：

- 在非注册时间的注册或在用户不可能在的时间的注册
- 有多次不成功注册尝试的帐号
- 有多次不成功注册尝试的终端

如果上述之一发生，就应该怀疑正有人企图非法侵入系统。应该保证口令按常规进行改变，并使其难以猜出，这才是口令安全性的最好保证。

#### 6.4.2 对计算机本身的无监督访问

最基本的安全性要求是防止对系统的无监督访问。虽然用户经常需要访问控制台来操作软盘或磁带驱动器，但是长时间使计算机硬件向无监督访问开放是危险的。一个有经验的用户可能会使系统失去作用并入侵到根帐号中。没有比这更危险的了。

当系统上无用户时，这种事件可能无人注意。这类破坏只能通过寻找如下记录才能被发现：

- 根帐号的注册报告
- 在审计档案中可疑的超级用户动作
- 在审计档案中无法解释的系统引导过程。由此可见，必须要把系统锁好，除了指定的系统管理员之外，任何人在非上机期间都不得使用系统。

### 6.5 处理文件系统讹误

把不可靠系统修复为可靠系统的代价要比维护可靠系统的代价大许多。一旦系统成为可靠系统，只要通过较少的工作来监视安全性参数的完整性就行了。文件系统讹误只是很偶然的事件，但却可能造成对系统十分重要文件的丢失。这个系统完整性的概念不同于对



系统破坏的处理，对系统的破坏是有恶意的用户蓄意更改或访问数据的举动。本节将介绍重要的安全性数据库文件以及在系统发生事故后如何恢复它们。

### 6.5.1 鉴定数据库文件

有几个数据库文件保存有关系统本身的数据，例如，它的用户、它的管理员以及它的子系统的有关参数，以便系统能控制其自身的安全性参数。这些数据库驻留在系统中并由一个管理员负责维护。

---

注：鉴定数据库文件绝不应手工编辑。包含在数据库中的信息可以利用可靠系统的工具和 `sysadmsh` 选择来维护和显示。不要试图用其它方法对它们进行修改。

---

审计和文件控制数据库是独立的数据库。这里所讲到的其它数据库（受保护口令数据库、终端控制数据库、子系统数据库、设备分配数据库）合起来被称作鉴定数据库。维护鉴定数据库是鉴定管理员的职责，鉴定管理员有 `auth` 授权。这里是各数据库的概要介绍：

#### 审计数据库

控制审计系统的行为。其中包括活动的种类、审计过程的系统报告、审计子系统的性能/可靠性特征和收集审计信息的文件系统设备。通过改变审计数据库中所存的参数，审计管理员能够调整审计子系统，以适应系统性能和安全性的要求。

#### 设备分配数据库

保存与同一个物理设备相关的设备路径名。例如，`/dev/tty1a` 和 `/dev/tty1A` 可以指同一个由调制解调器控制禁止和使能的串行端口。`init(M)` 和 `getty(M)` 利用这个数据库来制止一种注册欺骗形式，后面对此将有讲解。

#### 受保护口令数据库

保存每个用户的安全性信息。用户登记项包括加密了的口令（它永远不会在正常的口令数据库 `/etc/passwd` 中显示出来）和口令改变、用户权限、以及审计参数。通过正确地建立该数据库，鉴定管理员能够控制用户如何经身份识别和口令鉴定进入系统，和所允许的用户特权类型，以及在审计记录中要登记用户的哪些动作。系统缺省值数据库，包括整个系统的缺省安全性参数，它也被看作是受保护口令数据库的一部分。

#### 终端控制数据库

允许经过终端访问系统。它记录通过各连接终端所进行的注册活动（最近注册和注销的用户、时间标记等等）。终端控制数据库可允许鉴定管理员依据系统物理上和管理上的需要在不同终端上制定不同的策略。

#### 子系统数据库

实际上是一系列文件（每个子系统对应一个），在这些文件中保存着用户表，这些用户都是被授予了特权或是一个子系统的管理员或是有权在一个受保护子系统中执行特殊功能。这些文件是鉴定数据库的另一些组成部分。它增强了管理动作的可查性，因为它只允许特定的用户运行程序来对内部子系统进行维护。安全性也增强了，因为它能够控制谁有权执行程序来维护子系统，并且还能确定被赋予管理角色的真实用户。

## 文件控制数据库

协助维护可靠计算基础库 (the Trusted Computing Base) 的完整性。它是通过维护文件内容和保护属性的记录来完成这项工作的, 而这些文件对 TCB 的操作都是非常重要的。这个数据库提供了一个有效的工具来检查对 TCB 活动副本的修改。系统管理员可以运行 integrity (ADM) 来检查与这个数据库不符合的 TCB 文件访问权限。

### 6.5.2 发生故障后对系统进行检查

有好几个程序可以用来对鉴定数据库, 文件系统的系统区以及整个文件系统进行维护。维护工作的基本规则是从文件系统外表的最基本元素做起。否则在高层次上所进行的修正会被在低次上进行修复工作的程序给消除。知道了这一点, 在遇到系统故障或个别异常情况时, 就应按如下顺序使用这些程序:

1. 使用 fsck (ADM) 进行文件系统检查 (在重新引导时可自动执行)。
2. 使用 tcbck (ADM) 检查是否缺少关键文件 (在重新引导时可自动执行)。
3. 建立一个审计报告 (选做)。
4. 使用 authck (ADM) 检查鉴定数据库的一致性。
5. 使用 integrity (ADM) 检查系统文件的访问权限。这些程序都应在系统处于单用户 (系统维护) 模式下执行。

### 6.5.3 最优先终端 (the override terminal) 的使用

在安全性数据库发生错误且所有注册都被禁止的情况下, 对于根还存在一个最优先终端登记项。这是文件 / etc / default / login 中一个特殊的登记项。这个登记项标明着给根作最优先注册时用的 tty。这个登记项的默认值 (如下所示) 允许根在 / dev / tty01 上注册, 也就是控制台上的第一多屏幕页。也可以把这个默认值改为其它注册设备。

```
OVERRIDE=tty01
```

当这个数据库遭到破坏且根在最优先终端上注册时, 就会出现如下信息:

```
The security database are corrupt.
```

```
However,login at terminal tty is allowed.
```

当帐号被封了且根在最优先终端上注册时, 就会出现如下信息:

```
Account is disabled but console login is allowed.
```

所使用的 tty 在物理上应该是安全的; 请记住在这个 tty 上不会出现超级用户帐号被封锁现象。

### 6.5.4 文件系统检查: fsck(ADM)

包含有敏感文件的文件系统也必须被看作一个敏感的实体。因此, 由 fsck 所保证的文件系统的完整性增强了系统的整体安全性。

在系统产生任何故障或异常中断之后, 必须要运行 fsck 程序。就像常说的那样, 在运行 fsck 时必须使系统处于单用户模式。当系统发生故障时, 正在建立的进程中可能涉及用户、系统或是审计文件。虽然那些数据可能全丢失, 但 fsck 能够在文件系统的 lost+found 目录中恢复其中的一些文件, 并且最后修整文件系统的基本问题。

本书的“文件系统的管理”一章对 fsck 有深入的讨论。

### 6.5.5 数据库的自动检查和恢复: tcbck (ADM)

当由于掉电或硬件故障而使系统停机时,就可能破坏文件系统。这种破坏可能造成安全性数据库文件丢失,或者如果在系统出现故障时这些文件正被更新,它们就会停留在临时状态。在重新引导时,系统就会运行一系列程序,检查这些数据库文件的状态。当系统异常中止并重新引导时,这个检查将在 fsck (ADM) 在根文件系统上运行后执行,但在进入多用户模式之前。这个检查是这样进行的:

1. /etc/smmck (系统维护模式检查程序) 运行 tcbck (ADM) 程序清理那些正在更新而被滞留于临时状态的数据库文件。当安全性数据库文件被更新时,老文件 (file) 的内容被拷贝到或更新建立新的 -t 文件 (file-t)。然后,老文件 (file) 被移到 -o 文件 (file-o), 且新文件 (file-t) 被改回原文件名 (file)。当这个过程被打断, -o 和 -t 文件就被留下,并且在系统正常工作之前必须被删除。tcbck 首先处理留在 /etc/auth/system、/etc/auth/subsystem、和 /etc/files/auth/\* 目录中的 -t 和 -o 文件。如果一个文件有多个版本,那么附加文件就要被删除。具体措施如下:

A. 如果 file、file-o 和 file-t 都存在并且 file 的长度不为零,那么就删去 file-t 和 file-o。

B. 如果 file 和 file-t 存在,就删去 file-t。

C. 如果只有 file-t 存在,就把它改名为 file。

D. 如果只有 file-o 存在,就把它改名为 file。

如果上面的情况 C 发生,会显示如下信息:

```
/etc/tcbck: file file missing, saved file-t as file
```

这样做的原因是 -t 文件是原始文件的修改版本,并且可能被破坏;很有可能这个文件并不包括原始文件的全部登记项。在特定目录中,对于所找到的所有处于那个状态的文件都要重复这个信息。(-o 文件是不必怀疑的,因为它是在更新之前被改名文件的原始版本)

---

注: 如果没有备份并且文件 /etc/group 和 /etc/passwd 经过对 -t 文件改名已经重新建立了,就不要再从 UNIX 原盘中恢复。-t 文件中会包括原来的大多数 (如果不是全部) 登记项,而原盘版本中一个也没有。

---

2. 下面, tcbck 要检查关键的系统文件是否存在,其长度是否为零。如果有一个文件丢失 (或长度为零),就会显示如下信息:

```
/etc/tcbck: file file is missing or zero length
```

这个过程对下面的每一个文件重复进行:

```
/etc/auth/system/default
```

```
/etc/auth/system/files
```

```
/etc/auth/system/devassign
```

```
/etc/auth/system/ttys
```

```
/ etc / auth / system / authorize *
/ tcb / files / auth / r / root *
/ etc / group
/ etc / passwd *
```

执行完这个过程，如果还有文件丢失或有以-t 文件代替真实文件的，就会显示如下信息：

```
/ etc / smmck: restore missing files from backup or distribution.
```

---

注：tcbeck 不能报告出有错误的文件，但会有其它错误信息显示；这些信息在本章的“有关安全性的错误信息”中有讲解。如果有文件丢失，除了显示由 tcbeck 产生的信息外，还会产生其它信息；例如，当 /etc/passwd 丢了，会显示这个信息：

```
su:unknown id: root
* * * * PASSWORD FILE MISSING * * * *
--:mesg:cannot execute
```

---

3.如果关键数据库文件被删掉或发生错误（标有星号的文件），那么系统就会不问根口令而自动进入维护模式。（虽然这或许看起来是一个安全性缺口，但请记住系统本身必须要被锁起来，否则无论如何也不会安全的）。信息显示如下：

```
INIT:SINGLE USER MODE
Security databases are corrupt.
Starting root shell on console to allow repairs.
Entering System Maintenance Mode
```

如果没有关键数据库文件丢失，系统就会提示进入维护模式或进行正常操作。

4.如果发现文件必须重新恢复，那么第一个选择就是要从系统的备份中恢复文件。第二个选择才是从 UNIX 系统原盘中恢复文件。例如，如果系统报告文件 /etc/auth/system/ttys 丢失了，而有使用 sysadmsh (ADM) 建立的根文件系统的备份，就可以使用 Backups->Restore->Partial 选择来恢复文件。如果没有备份，就只好使用系统原盘上的文件了。

---

注：如果必须要从系统原盘上恢复文件 /tcb/files/r/root，就取 volume N02 并插入到软盘驱动器中，键入下面的命令：

```
cd / tcb / files / r
mount-r / dev / install / mnt
cp / mnt / tcb / files / r / root.
```

---

5.如果原系统是在磁带上，就把磁带插入磁带机中并键入下面的命令：

```
cd
tar xv8 files
```

其中 files 用所需要恢复的文件的完整路径名代替。

6.如果原系统在软盘上，键入下面的命令：

```
grep auth / system / etc / perms / * | sed " s / .* | <TAB> // "
```

可以看到输出如下：

```
./ etc / auth / system
./ etc / auth / system / authorize      B01
./ etc / auth / system / default       B01
./ etc / auth / system / default.c2    B01
./ etc / auth / system / default.unix  B01
./ etc / auth / system / devassign     B01
./ etc / auth / system / files         B01
./ etc / auth / system / ttys         B01
```

在上面的例子中，文件 / etc / auth / system 是放在 B01 软盘上。这个具体位置在原系统盘上可能不同。从 UNIX 安装软盘中取出相应的一张，插入驱动器中，并键入下面的命令：

```
cd
tar xv2 files
```

其中 files 是所需恢复的文件的完整路径名。当恢复了全部所需文件之后，按 <CTL>d；就会被提示进入系统维护模式或进行正常操作。再按一次 <CTL>d。

7. 系统进入多用户模式（显示“INIT New level: 2”）且被提示设置时钟后，系统就运行程序 authckrc (ADM)。这个程序首先再调用 tcbck 一次来确认以前曾被报告丢失的文件都已经恢复了：

```
Checking tcb...
```

如果又发现了丢失文件，就会显示前面所给出过的警告信息，后面还跟随如下信息：

```
/ etc / tcbck: file is missing or zero length
/ etc / authckrc: Log in on the OVERRIDE tty and restore
the missing files from a backup or the distribution disks.
```

这表示以前丢了一些文件。只有把这些文件都补充进来，系统才能进入多用户模式且才能允许注册。记下所丢失文件的名字。

8. 下面，authckrc 显示提示：

```
Check protected subsystem databases (y/n)?
```

如果回答 yes，authck 就开始运行，检查受保护子系统数据库文件，保证它们能正确反映受保护口令数据库中的子系统权限登记项。列在每个子系统文件中的每一个名字都要与受保护口令数据库中的同名登记项进行核对，以保证在这两个文件中不会有不一致的授权权限。另外，每个受保护口令登记项也要被扫描一遍，以保证其中所列的所有特权也都能在受保护子系统数据库中有所反映。如果发现不一致，就会询问是否要自动修复它们：

```
There are discrepancies between the databases.
Fix them(Y or N)?
```

---

注：如果系统自动引导（`/etc/default/boot` 中有 `AUTOBOOT=YES`），那么 `authck (ADM)` 就以非对话方式被调用，并且发现不一致时也会显示警告信息，但却不给 `authck` 机会来修复它们。转化到多用户模式的操作将正常进行下去。

---

9. 现在系统已经启动并准备接受注册。如果有文件被报告丢失（就像步骤 5 中所描述的），就必须在最优先终端上注册来恢复它们，按前面所描述的过程来做。按照默认设定，最优先终端被设定为 `tty01`，也就是第一多屏幕页。如果把这个默认登记项从 `/etc/default/login` 中删去了，就将不得不关闭系统、重新引导并进入单用户模式，然后按前述方法恢复文件。当在最优先 `tty` 上注册时，就会显示如下信息：

The security databases are corrupt.

However, root login at terminal tty01 is allowed.

10. 下一步就是检查受保护口令数据库。这是通过调用如下的 `sysadmsh` 选择来实现的：

Accounts→Check→Database

Accounts→Check→password

这些命令检查不同安全数据库间的一致性，包括 `/etc/group` 和 `/etc/passwd`。它们在本书的“系统安全性维护”一章中“鉴定数据库的检查”里有具体讲解。

### 6.5.6 数据库一致性检查: `authck (ADM)`

`authck (ADM)` 程序检查鉴定数据库的一致性。（正如下面所讨论的，`authck` 的功能也可由 `sysadmsh` 调用）。有几个选项可以控制检查的范围。

`sysadmsh` 的 `Accounts` 菜单中包括对鉴定数据库的检查和修理功能，前面讲过，鉴定数据库是一组包含有关可靠系统信息的文件。

#### (1) 检查鉴定数据库文件

为了检查口令、终端和子系统数据库的一致性，在 `sysadmsh` 中进行如下选择：

Accounts→Check→Databases

这个命令使用 `authck (ADM)` 命令检查每个数据库。当以命令行形式执行 `authck` 命令时，就可以使用 `authck` 的选项来修复不一致性错误。但是 `Accounts→Check→Databases` 选择不允许做这种修复。如果发现不一致性错误，可以从命令行执行 `authck` 或根据需要重新恢复文件。

#### (2) 检查 `/etc/passwd` 和 `/etc/group` 文件

为了检查 `/etc/passwd` 和 `/etc/group` 文件的一致性，可在 `sysadmsh` 中作如下选择：

Accounts→Check→Password

这个选择与创建新用户时作同样的检查。错误信息或警告信息可能产生；对任何错误信息都一定要进行处理。

### 6.5.7 系统文件的完整性检查: `integrity(ADM)`

`integrity(ADM)` 程序把文件控制数据库中的登记项与系统上的实际的文件访问权限进

行比较。从这一点来看，integrity 与 fixperm 命令有些相似，但它并不对访问权限进行修改。对这个程序的访问是由 sysadmin 子系统权限控制的，但作为超级用户运行它是最容易的。检查结果可能会报告某些文件比文件控制数据库中所述的有更多的访问权限。当发现错误时，应该对文件进行如下检查：

- 文件的主人 / 组 / 访问权限是什么？
- 文件的指定权限是什么？（使用 integrity 的 -e 选项。）
- 文件的最后一次修改和访问时间是什么？
- 那时有谁在系统上？
- 那时的审计情况如何？

一旦找到了差异的原因，就要重新给文件设置正确的访问权限。这件事做完之后，要运行 integrity 以使这些访问权限生效。

给 integrity 使用 -e 选项可以给出错误产生原因的简要说明。使用 -m 选项就只给出数据库中各文件的报告而不给出系统中文件的报告。有时在系统发生故障或受到入侵期间，很重要的文件会丢失。integrity 程序就是对此进行判断的一个手段。

integrity 的 -v 选项会打印出一些附加信息，包括通过完整性检查的文件的报告。这个选项会产生许多输出。

所有在完整性检查期间所发现的错误都会作为审计记录被保存，成为在审计过程中所发生的一件审计事件，通常被称为“数据库事件”（Database Event）。

## 6.6 有关安全性的错误信息

本节将列出管理员会遇到的问题和错误信息。下面所讨论的每个问题，都包括产生这种情况的原因、问题的解决，防止情况再次发生的方法。这里所讲的问题都要求作为根注册来解决。因为系统有可能会把所有用户（包括根）都锁在外面，所以必须知道给根使用的“最优先”终端的位置，这在本章前面“使用最优先终端”中已经讲了。如果管理员被锁在外面并且还取消了最优先 tty，就必须重新启动系统，并以单用户（维护）模式进入系统。由于关闭系统会引起文件系统破坏，因此保持一个最优先终端是非常重要的。

### （一）Account is disabled--see Authentication Administrator

下面三个原因之一会造成帐号被封闭：

- 利用 sysadmsh 的 Accounts → User → Examine:Logins 封闭了帐号。如果想重新建立这个帐号，可以使用同样的选择来开启这个帐号。

- 帐号的口令生命期终结了。在这个帐号的口令生命期结束之前没有改变口令。为了重新开启这个帐号，可以分配给它一个新口令或者使用 Accounts → User → Examine:Logins 选择来清除封锁（在这种情况下就要求用户必须在注册时设置口令）。建议用户注意在口令生命期结束之前更换口令。

- 对这个帐号曾作过一定次数的不成功注册操作，超过了所设置的次数的上限。这些次尝试并不一定都在同一台终端上进行的。在开启这个帐号之前，最好先判断出其封闭的原因。有可能因为用户的打字水平不高，或是有人想封闭这个帐号并知道这是实现其目的的一个办法，再或者就是确实有人想侵入这个帐号。可以根据系统用户的状况，数据的价值，以及对外界而言系统的可访问性来向上或向下调整这个不成功注册次数的限度。

(二) Account is disabled but console login is allowed.

或 Terminal is disabled but root login is allowed.

这些信息是有关超级用户在最优先终端上注册的（请看本章前面“使用最优先终端”）。如果控制台设备被作为特殊的设备并且被认为是一个值得保护的物理资源，那么对超级用户帐号的封锁就不能阻止超级用户从控制台注册。这就提供了一个进入系统的方法，即使当所有其它的帐号和终端都被封锁了。由于多次不成功注册而造成的系统控制台的封闭可被自动清除，但由于其它原因引起的封闭依然有效。实际上对超级用户而言，控制台是永远不会封闭的。

(三) Audit:file system is getting full

当审计文件系统达到某一限度，审计子系统可能会偶然显示这个警告信息。这个警告信息表明在某个设备上空间已经很少了。如果给这个子系统指定了一个附加目录，那么当这个文件系统已经达到所应保留空闲空间的最小限度时，子系统的附加目录就会自动切换。否则管理员就必须进行干预来获得更多空间。如果不这样做，当达到了界限值，审计就停止。如果因为空间不够或 I/O 出错而不能对一个紧致文件进行写操作，auditd 就会停止。可以使用 sysadmsh 的 System → Audit → Disable 选择来停止审计。分析问题的根源，并在重新开始审计之前解决。

(四) Authentication database contains an inconsistency

当运行一个与 TCB 或一个受保护子系统有关的程序时，可能显示这个信息。这说明鉴定数据库的完整性有问题。鉴定数据库由受保护口令数据库、命令控制数据库、受保护子系统数据库、系统默认值文件以及上述这些所有部分所提供的信息组成。错误原因是或者所要的登记项不存在或者登记项中的项目不正确。这个信息很含糊，用户被提醒出了问题，但又得不到出问题的足够信息来让他们发现安全性参数中的完整性问题。如果对于那个产生信息的用户有数据库事件 (Database Events) 记载，那么问题的真正原因可能会在审计记录中找到。Accounts → Check → Databases 选择能帮助对问题进行判断。

(五) Can't rewrite terminal control entry for tty

Authentication error;see Account Administrator

产生这个问题的大多数原因是设备 tty 在文件 / etc / auth / system / ttys 中的登记项出了错误。如果 / etc / auth / system / ttys 文件中的一个多屏幕登记项出了错，那么列在这个错误登记项下的多屏幕就不能访问了。例如，如果 tty03 的登记项错了，当按 <Alt> <F4> 和 <Alt> <F5> 时，系统就显示一个空白屏幕。

要解决这个问题：

1. 在最优先 tty 上注册。

2. 如果 / etc / auth / system / ttys 文件不包含 tty01 的登记项，那么就在 console 登记项下加入如下一行：

```
tty01:t__devname = tty01:chkent:
```

现在可以进入多用户模式并访问控制台的多屏幕页了。

3. 如果 / etc / auth / system / ttys 文件看起来并没受破坏，就应该检查一下 / etc / group 文件是否有错或被删除了。如果是，就从备份中恢复这个文件。

(六) Cannot obtain database information on this terminal



这个信息指出了 /etc/auth/system/ttys 文件的一个问题。这个信息可能偶然产生于有大量用户注册和注销的系统（譬如在商业批处理系统，BBS）。然而，如果这个信息在所有终端上都不断显示，那就需要对数据库文件进行检查。运行 tcbck (ADM) 工具，一般说来能够修复 ttys 文件的所有问题。正常条件下，当系统引导时，tcbck 自动运行。

#### (七) Login incorrect

用户输入了一个不正确的注册名或注册/拨号口令。如果这个情况重复发生，就需要更改这个口令以允许用户再次注册。

---

注：有几个信息与注册有关。在一些情况下，由 sysadmsh 也产生附加的解释信息。sysadmsh 信息以正常方式显示而不是以屏幕字形显示，以免与 login 信息混淆。

---

当用户（包括根）不能注册并且又不是由终端或帐号封锁引起时，就有可能是因为可靠数据库文件丢失或出现错误。如果用户见到“Login incorret”信息而他们又肯定其口令是正确的，就可能是 /etc/passwd 和 / 或受保护口令数据库发生错误。必须使用 Accounts→Examine 选择来查找真正的问题。当输入了用户名，Sysadmsh 就检查这个帐号的状态并给出更多的错误描述信息。这些 sysadmsh 信息和所需的处理如下：

- User exists in /etc/passwd but not in Protected Password Database

存在于 /tc/files/auth/[a-z]/username 中的数据库登记项被删除或出现错误。如果发现这对许多用户确实是个事实，就应该从备份中恢复这个文件。对于单独的用户，对此进行修复的最简单的方法是选择 Identity 信息并填入空白域，再分配一个新口令。

- User exists in Protected Password Database but not in /etc/passwd

由于某种原因 /etc/passwd 出现了错误或有人利用根的权力对这个文件进行了错误编辑。应该尽力选择 Identity（完整性）信息填入丢失的域。如果这样做了之后还不能正常工作，那么就必须在根中，从备份中恢复 /etc/passwd 文件。做完这之后，还必须删除下列文件以保证系统接受这个替换的文件 /etc/passwd：

/etc/auth/system/pw\_id\_map

/etc/auth/system/gr\_id\_map

#### (八) Login:resource Authorization name file could not be

allocated due to :cannot open

Bad login user id

/etc/auth/system/authorize 文件出现了错误或被删除了。在最优先 tty 上注册并从备份中恢复这个文件。

#### (九) Terminal is disabled--see Authentication Administrator

这个终端对所有用户都封锁。与帐号封锁类似，或是由于鉴定管理员使用 sysadmsh 封锁了这个终端或是对这个终端来说不成功注册次数超过了限度（对一个或多个帐号）。在这两种情况下，都要判断到底发生了什么，然后使用 sysadmsh 重新设置这个锁。

#### (十) You do not have authorization to run....

这个命令是一个受保护子系统的一部分。对于那个子系统，鉴定管理员并未提供给你

运行这个命令所需的核心权限或相关的命令。鉴定管理员可使用 Accounts→User ▶Examine 选择来赋予或剥夺这种权限。

## 6.7 可靠系统上所运行的守护程序

为了实现这个操作系统的可追究性的安全特点，就必须分别构造和维护若干不同的 UNIX 守护程序，并使它们作为后台进程运行。这一节将讲解影响系统守护程序的一些特征，并列出一一些必须有守护程序监督才能进行的操作和程序修改。这保证了守护程序以正确的标识符和优先级启动，并正确地处理边界条件和运行故障。

所谓操作系统责任性原则是指每个进程都必须被标记一个去不掉的标识符，叫做注册用户 ID (即 LUID)。这条原则仅有的例外是标记着标识符的进程，它们是 init (M)、login (M) 和 cron (M) 程序。所有的可靠实用程序或者标记自己的 LUID (例如 auditd (ADM)) 或者在运行前被标记 LUID (例如 lpsched (ADM))。如果没有设置 LUID, setuid (S) 和 setgid (S) 这两个系统调用就会失败。

作为系统管理员，必须保证每个新引入的守护程序，如果它是从系统启动文件 (/tcblfiles/rc?.d/\*) 开始执行的，就一定标记了一个 LUID。正确的过程是为该守护程序在 /etc/passwd 和 /etc/group 文件中建立适当的伪用户和用户组帐号及受保护口令登记项。如果要从 shell 启动描述程序启动守护程序，那么就在那个描述程序中加入一行，就像下面给出的那样，用 su (C) 命令运行这个程序，使得进程的标识符能够正确设置。这个过程与在某个帐号下使用传统的启动描述程序运行守护程序一样。例如，行式打印机守护程序 lpsched 用下面这样一行启动：

```
su lp-c /usr/lib/lpsched > /dev/null 2>&1
```

如果某进程的 LUID 还没有设置，su 的一个可靠版本就会为它设置 LUID。

请注意上面这条命令的标准输出和错误被重新定向到空设备 (null device) 上。这个操作系统有一个特点，就是它很难处理从守护程序获得的控制台输出，管理员必须相应地控制守护程序的输出。所有的终端设备都受控于系统调用 stopio (S)，这个系统调用是用来保证身份识别子系统能够有效地防止非法注册的。当用户注销时，为那条终端线服务的 getty 就会调用 stopio，并以该终端设备名作为 stopio 变元。任何一个进程，如果它还试图对那个终端执行写操作，就会被杀掉 (标记 SIGHUP)。如果在守护程序启动与守护程序输出之间用户从系统注销，守护程序对控制台的写操作就要经过这个标记。因为大多数守护程序都忽略 SIGHUP，所以它们输出的信息就会丢失。因此，如果这些信息需要保存，就应该把守护程序的输出重新定向到一个文件中，或向上面那样把输出重定向到一个空设备中。

## 第七章 审计子系统

审计子系统以审计行踪 (audit trail) 的形式对系统中所发生的有关安全性事件进行记录, 供以后查询。由这个子系统产生的审计行踪能够用来发现对系统的非法入侵和对系统资源的误操作。这个审计子系统在设计上符合美国国家计算机安全中心 (the U.S. National Computer Security Center) 所制定的审计标准。

审计可以通过复查该子系统所搜集的数据来检查对目标 (文件) 的访问方式, 还可以观察特定用户以及他们进程的活动。任何违犯保护和权限机制的企图都要受到审计。由于有关系统安全性的事件都要受到审计并且都可追究到一个特定的用户, 因此审计子系统对于那些企图对系统进行误操作的用户来说是一个制约。

审计子系统使用系统调用和各种实用工具把用户分成不同的事件类型。这可以用来选择增加或减少审计。有一种事件类型。叫做 Discretionary Access Control (DAC) Denial, 记录企图以某目标的访问权限所不允许的方式来使用该目标的事件。例如, 如果一个用户进程要对一个只读文件进行写操作, 就会产生一个 DAC Denial 事件审计, 以指出这个进程试图写一个没有资格写的文件。在检查审计行踪时, 会很容易注意到对权限所不允许的文件所进行的多次重复访问。这能够提醒管理员警惕外界对系统可能的破坏或入侵。

审计数据的基本作用就是能够唯一地识别所有的用户及其活动, 以使得审计行踪能够准确地反映每个用户的可审计活动。用户要在系统上注册, 他们在获准访问系统之前必须经过身份识别和口令鉴别。安全机制还要给用户建立的每个进程打上一个不可改变的识别标记: 注册 UID 或 LUID。LUID 将永远保持不变, 即使使用像 su (C) 这样的命令在用户帐号之间进行迁移。每个由子系统产生的审计记录都包含 LUID 以及该进程有效的用户 ID 和组 ID。这样, 就可以严格地使用户对其活动负责。

审计子系统由审计管理员进行管理。对于选择作为审计记录的事件, 对于控制子系统的参数值, 以及对于审计数据所做的后续简化和分析, 审计管理员都能完全控制。

审计子系统由五个主要部分组成:

- 核心审计机制
- 审计设备驱动程序 (/dev/auditd 和 /dev/auditw)
- 审计压缩程序 (auditd (ADM))
- sysadmsh 审计界面
- 数据精简与分析软件。

有些可靠的系统实用程序并不是审计子系统的真正部分, 但它们仍要负责向审计行踪中写审计记录 (例如 login (M))。

### 1) 核心审计机制

核心审计机制是审计子系统的中心。这一机制以用户通过核心系统调用而建立的进程活动为根据生成审计报告。每一个核心系统调用都在子系统表中有一个登记项, 来指示这个调用是否与安全相关, 如果是, 它又是与那一种事件类型相对应。另外, 错误代码表又进一步把系统调用划分成具体的安全事件。核心审计机制还要为设备驱动程序做一个内部

调用，向审计行踪中写一个记录。

例如，系统调用被划定为一个 Make object available 事件。如果用户 blf 对 /unix 执行了 open (S) 并且执行成功，那么就会产生一个审计记录记载这个事件。然而，如果 blf 在执行 open (S) 时要求对该文件进行写访问而他又对这个文件没有写的权限，那么这个活动就被划为一个 blf 对目标 /unix 进行的一个 DAC Denial 事件。可见，一个系统调用根据所访问的目标以及这次调用的结果可以对应几种事件类型。因此，根据所使能的事件类型，一个系统调用可以有选择地受到审计。

有些系统调用可以被认为是与安全性无关。例如，getpid (S) 能获得进程的 ID，但并不构成一个与安全性相关的事件。因此这个系统调用不会被审计。

## 2) 审计设备驱动程序

审计设备驱动程序作用如下：

- 从核心审计机制和可靠实用程序中接受审计记录
- 建立并撰写中间审计行踪文件
- 给审计守护程序提供审计行踪数据供其压缩使用。
- 根据事件类型、用户 ID 和组 ID 规定选择建立的审计记录。

设备驱动程序像其它许多字符设备一样提供了 open (S)、close (S)、read (S)、write (S) 和 ioctl (S) 界面。然而，审计设备只能由具有 contigaudit 或 writeaudit 核心权限的进程打开。这就限定了只有可靠的实用程序，譬如审计守护程序和审计管理员接口，才能够访问审计设备。一个审计设备能够在同一时间被许多进程进行写操作。这个设备把各记录合并为审计行踪。这个设备只能被一个进程读取，即审计守护程序。

审计设备驱动程序把审计行踪当作一组审计采集文件 (audit collection files) 进行维护。每当启动审计，就开始了一个新的审计阶段。当这个阶段开始后，子系统就建立一个新的采集文件，并向其中写入审计记录。当这个采集文件达到一定的大小 (由管理员设定)，子系统就再建立一个新的采集文件并开始对它进行写操作。因此，审计行踪可以被看作一个不断增长的顺序文件，即使是用到了许多采集文件。这正是审计行踪被审计守护程序查看的方式。当到达一个文件尾时，子系统就为守护程序切换到新的采集文件。所有这些过程对守护程序都是透明的。

## 3) 审计压缩守护程序

审计守护程序 auditd (ADM) 是一个可靠的实用程序，当启动审计后，它就作为后台守护进程运行起来。守护程序是审计设备的唯一读者，反过来，审计设备又从审计采集文件中为守护程序提供记录块。守护程序并不关心审计行踪如何分散于许多采集文件中。审计设备驱动程序满足守护程序所提出的读请求并按需要进行采集文件的切换和删除。

守护程序的主要目的是为审计阶段提供一个压缩和注册机制。守护程序也对各受保护子系统提供支持，使它们能够向审计子系统中写入审计记录。根据所选定的审计记录的生成准则，在系统中会产生大量的审计数据。对于一个典型的单用户系统来说，在一个小时中产生 200K 字节的审计数据是很正常的。因此，守护程序就提供了一个压缩机制，把审计数据压缩成一种紧缩记录形式，并贮存于一个审计压缩文件 (audit compaction file) 中。这种压缩算法使得文件空间平均压缩了 60%。这大大减少了用来存储审计记录的磁盘空间。

守护程序的另一个作用是提供了一个记录文件来描述当前的审计阶段。这个记录文件中包含着在压缩文件中有关此阶段的审计记录数、这个阶段的开始及停止时间、以及有关这个阶段状态的其它说明。就像审计设备驱动程序当采集文件达到一定大小之后就要对其进行切换一样，守护程序也会生成多个压缩文件，以防止一个单独的文件增长得过大而无法管理（这也要预先设置）。由守护程序所写的审计压缩文件可被放在不同的由管理员指定的目录中。因此，记录文件还能提供一个压缩文件的踪迹记录，以供将来做进一步的数据整理。

审计守护程序的第三个作用是作为审计设备驱动程序的接口程序，以使得那些没有 writeaudit 权限的受保护子系统能够写审计记录。因为这些子系统不能直接访问审计设备驱动程序，但能够以一种可靠的方式连接守护程序，所以必须由守护程序把它们所写的审计记录传递给审计子系统。

#### 4) 审计子系统接口

sysadmsh 提供了一些简单的选项来建立和维护审计子系统。这就允许系统管理员对子系统设置和初始化、修改子系统参数、维护子系统（备份、恢复等），全局性的或有选择地减少审计数据。

#### 5) 数据简化和分析工具

审计子系统还包含一个数据简化和分析工具，以检查以前的审计阶段或当前的审计阶段的审计行踪。利用由审计守护程序产生的记录文件，简化工具能够找出所有用以简化审计阶段的压缩文件。由于压缩文件中的数据是压缩格式的，因此这个简化程序还包含解压缩数据所需的例程。

为了更加有效地分析审计数据，这个简化工具还需要管理员指定某种事件类型、用户 ID、组 ID 和项目名称以便有选择地简化数据。另外，还能够在一个指定的时间区间里查找与指定的标准相匹配的记录。如果记录不在这个指定的时间区间中，那么在简化过程中就不予考虑。

作为一个例子，选取 DAC Denial 事件中用户 ID 为 blf 的用户寻找目标 / unix 这一事件，进行数据简化。只有反映 blf 由于不具备权限而对 / unix 的访问企图被禁止的记录被打印出来。这提供了一个很有力的机制，能够在不必分析整个审计行踪的情况下就立即找到有关的安全性事件。

### 审计方法

这部分介绍审计子系统如何工作，选择数据的标准是什么，以及审计要求如何影响系统性能。

#### 1) 审计权限

有四种权限与审计子系统有关：

- configaudit 权限允许为系统中的所有用户设置审计参数。
- writeaudit 权限允许向审计行踪中记录指定信息。
- audittrail 权限允许用户对他们自己的活动进行审计。当一个用户被赋予这个权限，他就能访问 System→Audit→Report 功能。
- suspendaudit 权限禁止所有审计。

#### 2) 审计记录的来源

审计行踪包含系统中与安全性有关的事件。有效的审计不仅关心用户进程所进行的系统调用，它还包含像注册、注销和注册失败这样的事件。这些事件的关键是要判断谁在什么时间从哪个终端访问了系统，并进行了什么活动。在核心级是不可能对注册失败进行审计的，这是因为核心不能知道一个应用程序做了些什么。因此，某些与安全性紧密相关的工具，例如 login，就必须被允许产生审计记录。

### 3) 核心审计机制

在审计行踪中所保存的一大部分审计记录是由核心审计机制产生的。审计子系统产生的记录与用户进程的系统调用相关，这种系统调用对应有关安全性事件。有些系统调用，例如 open (S)，依据用户的变元和所打开文件的状态，会对应多种安全性事件。如果 open (S) 的调用带有 O\_CREAT 标志，那么若这个文件不存在，它就会被建立。如果给定了 O\_TRUNC 标志，那么若它存在，其长度就被截为零。这说明了 open (S) 的调用是如何与三个独特的事件，Make Object Available、Object Creation 或 Object Modification 相对应的。

错误代码对于判断一个事件也起到重要的作用。有关系统调用的错误，包括访问或权限不允许以及资源耗尽问题，都与特定的事件类型对应。在系统调用的最后，核心审计机制要判断这一调用属于哪类事件，并且判断这个事件是否要像所指定的那样进行审计。另外，这个机制还可以使用其它的所选定的标准，例如用户 ID 或组 ID。在这种方式下，审计记录的产生可以被限制在一个选定的用户组中。

### 4) 可靠的应用程序

可靠计算基础库 (TCB) 包含一些为了得到一个可靠的环境所必不可少的可靠的应用程序。其中有 login、su 和许多审计子系统命令。为了减少写到审计行踪中的审计数据和使得审计更加有意义，这些可靠的应用程序被允许直接对审计设备进行写操作。举一个例子，login 可以往审计行踪中写一个注册审计记录，而不必使得在系统上的注册被当作一组系统调用而提出来去完成这个注册过程。

仅仅让可靠的应用程序来写审计设备是不够的。还必须有一个办法来压缩由核心审计机制通过系统调用而产生的审计记录，以防止出现审计行踪杂乱的问题。因此，正如前面所讨论过的，还存在 suseaudit 权限。可靠的应用程序都要在这种权限有效的情况下运行，暂停核心系统调用对那个进程进行的审计并打开审计设备进行写操作。只有一少部分可靠的应用程序是允许这么做的。用户进程的运行都不应带 suseaudit 权限。授权机制是由 login 管理的，使用严格的系统调用，并以受保护口令数据库 (Protected Password database) 的登记项为根据。

### 5) 被授权的子系统

审计记录产生的第三个方法是通过被授权的子系统，例如 lp、cron、terminal 和 mem 子系统。有时一个子系统会遇到矛盾或问题，这就需要它写一个审计记录来记载。然而，各子系统都不具备 writeaudit 权限而不能直接向审计子系统中写审计记录。

解决的办法是，各子系统会像可靠的应用程序那样对记录进行整理，然后通过一个可靠的接口把这些记录传送给审计守护程序进程。审计守护程序，作为一个可靠的应用程序，来继续完成把审计记录写入审计设备的任务。这就使得受保护子系统进程可以生成简明而有意义的审计记录，而又不必赋给它们 writeaudit 权限。

## 6) 责任性审计

审计子系统对有关系统安全性事件进行审计，并把各个事件与某一个特定的用户联系起来。用户通过 login 程序注册到系统中。这个程序对用户进行鉴定以判断是否允许其访问。注册过程对成功的和不成功的注册都提供了审计支持。当用户注册成功，login 就会在用户进程上标记注册用户 ID (LUID)。无论那个进程调用了多少次 setuid (S) 和 setgid (S)，LUID 都不会改变。这样，对于这个进程和这个用户就确定了严格的责任性。用户进程依然可以调用 setuid 和 setgid 这两个系统调用，这也要受到审计。审计记录把进程的 LUID 与该进程实际的用户 ID 和组 ID 记录在一起。

## 7) 审计事件类型

每一个审计记录，不管其来源是什么，都标记有一种事件类型。对于用户进程的系统调用，事件的类型根据这个系统调用及前面所讨论过的内容，被确定下来。对于应用程序或子系统审计，由写这个审计记录的进程来设置事件类型。这个事件类型不能被审计设备或审计守护程序改变。

事件类型是很重要的，因为它们是用来区分系统上安全性事件的。审计记录的产生和压缩都可以根据事件类型进行控制。例如，如果只关心用户在系统上的注册和注销，就可以指定对那种事件类型进行采集或归纳。

审计子系统提供了很全面的事件类型，对于各种特殊和常见的事件都能有正确的反映。这些事件都归纳在表 7.1 中，其中的字母是审计子系统用以查询事件的简单标志。

表 7.1 审计事件

A. Startup / Shutdown	B. Login / Logoff
C. Process Create / Delete	D. Make Object Available
E. Map Object to Subject	F. Object Modification
G. Make Object Unavailable	H. Object Creation
I. Object Deletion	J. DAC Changes
K. DAC Denials	L. Admin / Operator Actions
M. Insufficient Authorization	N. Resource Denials
O. IPC Functions	P. Process Modifications
Q. Audit Subsystem Events	R. Database Events
S. Subsystem Events	T. Use of Authorization

可以根据这些事件类型有选择地收集和归纳审计数据。审计子系统接口程序可以为审计子系统或数据归纳程序建立一张事件类型表。

审计子系统使用事件类型来确定一个审计记录是否应该写入审计行踪。审计管理员能够全面控制哪些事件需要审计。

为了控制审计事件类型，审计子系统中包含一个全局的“系统审计事件掩码 (system audit event mask)”，后面将有介绍。审计子系统还要对每一个系统上的进程维护一个事件类型的掩码。

## 8) 强制审计

为了得到用户进程的全部准确信息，而产生一个有意义的审计输出，核心审计机制必

须对某些系统调用进行审计。当审计使能之后，一些事件就会受到审计，即使审计管理员没有选择任何事件。这就是所谓的强制性系统调用。它们对于进程状态的维护十分关键。例如，open (S) 系统调用可以指定一个类似../newfile 这样的相对路径名。全路径名依据进程的当前目录而定，它是通过使用 chdir (S) 系统调用而设置的。如果没有当前目录值这一原来的信息，包含有路径名../newfile 的审计记录就没有意义。

对于 close (S) 系统调用也有此问题。这个系统调用只需要一个文件描述符作为变元来关闭以前打开的文件。除非所关闭的目标的名字被输出到记录中，否则 close 审计记录就无意义。然而，除非在文件打开时其路径名被保留，否则就无法找到路径名来关闭文件。

表 7.2 列出了有影响的事件类型

表 7.2 强制审计事件

Event type	Always audited	Optionally audited
Make object available Object creation	open, pipe creat	mount, opensem link, mkdir, mknod, creatsem, sdget
Map object to subject	dup, exec, exece	fstatfs, getdents, stat, statfs
Object modification	execseg, unexecseg	chsize, stime
Make object unavailable	close	sdfree, umount
Process create / delete	exit, fork	
Process modification	chdir, chroot, proctl, security, setgid, setpgrp, setuid	

强制审计并不仅限于表 7.2 中所列的那组系统调用。注册事件是所定义的唯一可靠应用程序强制审计记录。在一个用户注册之后，在注册记录中就包含一个在其上发生了注册活动的终端的指示符。如果同一个用户在系统上的多个终端注册，那么那个用户的活动都可以追溯到各特定的终端。

#### 9) 系统审计事件掩码

系统事件掩码是针对审计子系统全局的。如果想选择另一组不同的事件，可以在审计期间修改这个掩码。系统事件掩码对于每一个事件类型都有一位，当需要审计时这一位就置为“1”。这提供了一个快速检测手段（使用逐位操作）来判断一个新生成的记录是否要审计。当一个新的进程通过注册而建立时，审计子系统就要使用系统事件掩码来计算用户掩码。

#### 10) 特定用户和进程的事件掩码

可以通过建立一个“特定用户的事件掩码 (user-specific event mask)”给某用户一个掩码来替代系统掩码的缺省值。系统上的每个进程都有一个“进程事件掩码 (process event mask)”，它告诉系统要对那个进程进行什么审计。在用户注册之后，login 程序就要查找这个用户特定的事件掩码并像这里所讨论的那样为注册 shell 设置进程事件掩码。

对于每一种审计事件用户特定的事件掩码有下列三种值中的一种：

- 必须审计这一事件



- 禁止审计这一事件
- 使用系统审计事件掩码

对于每个审计事件类型，如果用户特定的掩码指示这个事件必须或禁止审计，进程审计掩码就按用户特定的掩码来设置。否则，进程审计掩码就按系统审计事件掩码来设置。在大多数情况下，用户特定的事件掩码为所有的审计事件设置第三种值，这使得系统把缺省值赋给用户。

使用审计子系统应该遵循某些指标。审计子系统在设计上具有灵活性和可靠性，可以采集希望得到的审计数据。审计记录的生成还支持审计事件、用户 ID 和组 ID 的预选择。如果因为某些原因（当某用户以某种方式企图访问没赋给其权限的文件时）只想注意一个特定的用户或用户组，那么预选择就很有用了。事件类型也可以被用来作预选择，例如只审计注册和注销事件。预选择对于节省磁盘空间是很有利的，这是因为由审计子系统写到采集文件中的记录减少了。然而使用预选择也有一个弊端。如果发生了一件侵害系统安全的事件，而这一事件或造成这一事件的用户又没选定被审计，那么这一活动的记录就要丢失。

因此，一定要十分谨慎，不要对审计事件用户或用户组作预选择，而要进行全面的审计。这样做的好处就是所发生的任何与安全性相关的事件都被记录到审计行踪中。全面审计的缺点是它消耗了大量的磁盘空间。

还可以把全面审计与后选择结合起来，只对感兴趣的记录进行检查。对于审计行踪的有选择的检查也是基于事件类型、用户 ID、组 ID，以及目标名称，还有记录产生的日期和时间。

对审计子系统的管理对于有效的审计是一个关键。通过谨慎地设置和使用审计子系统，就能够掌握一个强有力的工具来帮助管理员在问题产生时依然保持系统可靠并识别出所发生的问题。审计子系统设计上所包括的审计事件的范围是很全面的，既包括核心活动又包括系统应用程序的使用活动。它在设计上在注意了其可靠性的同时还注意尽量减少对系统整体性能的影响。

审计子系统能够符合要求的程度完全依赖于对系统的正确管理。可以使用审计参数在可靠性和性能之间进行权衡。不合适的设置可能造成很低的性能和审计数据丢失，或这两者同时产生。例如，设置审计事件掩码来管理审计子系统所审计的事件类型，这是十分关键的。假如事件的预选择不包括注册事件，那么通过拨号线路对系统的入侵就不会被察觉。

在对审计子系统对于系统性能的影响进行估计时，考虑子系统必须进行的活动是很重要的。审计子系统设备驱动程序是从所有来源采集审计记录的一个焦点，同时它还要负责把那些记录写到审计行踪中去。这个采集文件又为系统中所有被审计的进程所共享。这种形势很像一个飞机订票系统，许多办事员访问同一个公用数据库。必须有一个封锁机制来防止审计记录的混杂，而保证数据库的一致性。对于审计子系统的采集文件也是同样的。

有一个内部缓冲机制和一个写滞后（write-behind）策略被用来控制对一个采集文件的同时多重写入。这就可以使得审计子系统设备从各进程和应用程序获得审计记录，同时对于采集文件的写操作也在并行进行。可以按要求调整这一机制，譬如要用多大的缓冲区，要以何种频率把数据写入采集文件。

与系统性能同等重要的是审计行踪的可靠性。传统的 UNIX 系统在系统发生故障时缺少保护文件系统完整性的措施。这起因于 I/O 操作是通过异步写一个缓冲池来完成的。因此，对文件的修改在系统发生故障时可能实际上还没有记录到磁盘上。

这是很不幸的，因为那些引起系统故障的事件正是审计所最感兴趣的事件。因此，就非常希望尽量减少在系统发生故障时从审计子系统所得来的数据的丢失的可能性。为了达到这一目的，审计子系统运用了一种叫做同步 I/O 的工具，这可以使得当采集缓冲区和采集文件 i 节点一发生变化就立即对它们进行更新。这就最大限度地减少了由系统故障而引起的数据可能丢失的数量。

在数据的可靠性程度和审计子系统的性能之间有一种直接的相互联系。由核心审计机制、可靠的应用程序和受保护子系统所产生的审计记录长度一般为 40 到 60 个字节。如果每一个记录在它们被提供给审计子系统时被同步地写到磁盘上，那么就会带来较低的性能；I/O 系统也会由于这些记录产生的速度太高而无法承受。解决这一问题的方法是先把记录都暂存于一个缓冲区中，并间隔一定的时间把它们向审计行踪中一起写入一次。这个间隔可以由所经历的时间来定，也可以由所积累的数据量来定，这是可选择的。

对于审计子系统的管理来说，最后一个关键的问题是决定需要对什么进行审计。审计记录生成的预选择可被用来细致地调整审计行踪来注意一个事件或几个事件。例如，系统可能只允许一组人使用，但又不允许他们夜晚来上机。另外，对非工作时间的工也提供一些拨入线路。这时，-就可以只关心谁在何时使用了系统。在这种情况下，预选择可以只被用来审计注册和注销事件。未被授权用户入侵系统的企图就会被审计为不成功的注册。

审计也可以集中于某些特定的用户或用户组。这可以使注意力集中于被怀疑为违犯安全性策略的人的身上。审计子系统的要求越少，它对系统性能的影响就越少。完全的审计可以建立一个深入而详尽的系统事件的记录，但也需要非常多的资源完成。然而，先对各事件进行记录，而后再使用压缩工具把不想要的记录去掉，这要比没有某些记录而这些记录又对检查所出现的问题很有用要好。这一结论是依据所希望获得的安全性的程度得来的。

对于审计子系统来说，了解审计阶段这个概念是很重要的。一个阶段对应一段时间间隔，这个间隔从系统引导时开始到系统被关闭为止。为了减少写入审计行踪的数据量，审计子系统在设计上尽量减少每个审计记录的大小。因此，一个进程的状态只在有限的一串连续记录中记载，而不是在各记录中完完全全地对它进行说明。因此而节省的空间和时间都特别多，但这要求认真地进行管理，以免疏漏。

---

注：

如果在系统运行时审计子系统被关闭了，而过了一会儿又重新开始工作，那么就会建立一个新的阶段。一个阶段被定义为与一个特定的时间区间相关的包含有审计记录的一串采集和压缩文件。有些在第二个阶段或后续的阶段被审计的进程可能在第一个阶段就被建立了。因此，一个阶段可能并不包括某一个进程的所有的有关进程状态。反之，这又会造成记录整理的不完整。为了避免这一问题，应该只在关闭系统时关闭审计。更多的内容，请看本章后面的“保持审计行踪的连续性”。

---

下面将讨论审计管理员所关心的几个特殊问题。

### 1) 磁盘空间

审计子系统会生成大量的审计记录。即使记录都很小，保存它们所需的存储量也会增长得很大。因此，还必须注意系统的管理。审计应该直接被送到有大量空闲空间的磁盘中去。审计子系统中设置有一个保护机制，可以在审计设备的空闲空间很少时发出警告。如果这一情况不立即处理，并且所剩的磁盘空间减少到低于某一界限时，子系统就要切换到另一个新的审计目录中去。因此，另外的目录应被放在不同的文件系统中。当审计子系统遇到一个 I/O 错误时，它也要切换到一个新的目录中去继续审计。

### 2) 系统故障

大多数系统都会在某一时刻发生故障，尽管提供了很多容错的手段。如果发生了系统故障，就可能由于输出到缓冲区中的记录与 i 节点的不一致而造成数据丢失。审计子系统尽量使用同步 I/O 来对缓冲区、i 节点和目录的更新进行关键性的操作。然而，这并不能保证数据总在磁盘上。如果是由于磁盘的问题而造成系统发生故障，这种问题更可以产生。

在重新启动时，经常会发现文件系统中的审计行踪文件受到了破坏。这时没有别的选择，只有删除这些审计文件来消除问题。这在一定程度上解决了审计行踪的问题，但不应该影响对文件系统的恢复。

### 3) 子系统信息

审计子系统是有弹性的。I/O 错误可以由审计子系统把采集和压缩切换到新目录中去而解决。当文件系统的空闲空间变得很少时，也可以用同样的方法来解决。也有一些情况能使得审计子系统不能继续工作。如果磁盘介质受到破坏或者没有再多的文件系统了，那么审计子系统就要停止工作并在系统控制台上打出有关信息。任何异常停止条件都会产生一个控制台信息，来帮助对问题进行判断。

对于系统问题，其症状一般并不仅限于审计。有一种问题是由于删除随后又重建审计参数文件而引起的，这种问题关系到审计阶段的复制。每当审计生效，一个新的阶段就建立了。这个阶段由在这个审计期间所产生的记录文件和所有压缩文件定义。这些文件分别以阶段号来标记以易于识别，并被那些需要访问这些文件的子系统实用程序所使用；这些实用程序可能只关心阶段号而不是文件名。

如果各阶段都保留在机器上，并且参数文件被修改了，使得这个审计子系统阶段号被重新设置，那么就会造成所建立的一个审计文件使用的是以前某一个阶段所用过的一个名字。如果发生这一情况，就应该在再次使能审计之前使用 `sysadmsh` 的 `System → Audit → Files` 功能找到并删除那个旧的阶段。

## 7.1 数据采集

这一节将介绍如何建立、启动、管理和释放审计。这个系统是以默认采集参数设置属性的（请看本书“用户帐号的管理”一章中“缺省的帐号配置”中的一列缺省值）。可以修改这些缺省值以适合自己的需要。

正如前面所讨论的，审计子系统的使用分为两级，采集和报告。每级都涉及审计数据的选取。由审计子系统采集的数据由一组被称为掩码的参数文件来控制。之所以使用“掩

码”这个词，就是因为不想要的的数据被掩去了而不被采集。一旦启动，审计子系统就会按审计掩码所指定的那样采集数据，直到审计被停止或系统被关闭。系统支持两类掩码：控制对全体用户所做的缺省审计的系统全局性掩码，以及可以对每个用户分别定义的特定用户掩码。用户掩码比系统全局性掩码级别高。

有如下的数据采集功能：

目录：显示或修改审计采集和压缩文件目录表。

事件：显示或修改系统审计采集类型掩码。

ID：显示或修改被审计的用户或用户组表。

参数：审计子系统性能参数。

复位：把采集规则改回缺省值。

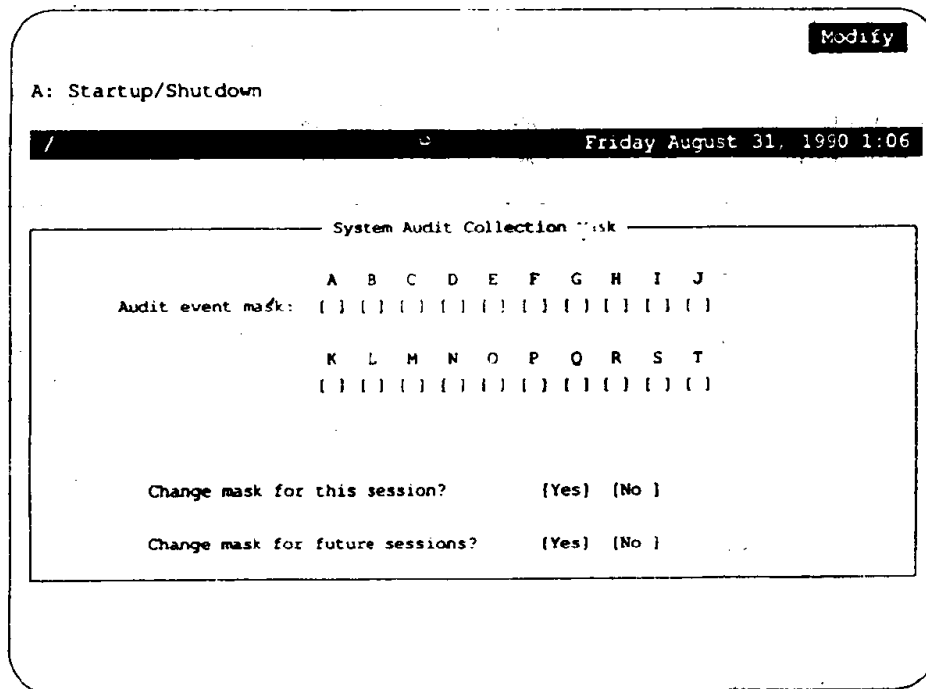
统计：显示对当前审计阶段的统计。

### 7.1.1 选取审计事件

为了针对于全系统选择要审计的事件，可在 Sysadmsh 中进行如下选择：

System → Audit → Collection → Events → Modify

会看到类似下面的屏幕：



每一种事件类型对应显示为一个字母。对于那些要被审计的事件，这种事件类型应被标上一个 Y。而那些不要审计的事件类型可使用 N 这个任选项来排除。使用空格键可把某一项的值从 Y 改为 N 或从 N 改为 Y。使用箭头键可从一项移至另一项。这个事件掩码可为当前的审计阶段作修改和动态的改变，也可被写入参数文件为以后的审计阶段服务。

正如前面在“审计事件类型”下所讨论的，有若干种审计事件可供选择，这些事件被归纳于表 7.3。

表 7.3 审计事件类型

事件类型	说明
A. Startup / Shutdown	系统启动（引导）和关闭。
B. Login / Logoff	成功的和不成功的注册。
C. Process Create / Delete	进程的创建和终止。
D. Make Object Available	文件、信息、信号量的打开和文件系统的安装。
E. Map Object to Subject	程序执行。
F. Object Modification	文件的写操作。
G. Make Object Unavailable	文件、信息、信号量的关闭和文件系统的拆卸。
H. Object Creation	文件 / 信息 / 信号量的建立。
I. Object Deletion	文件 / 信息 / 信号量的删除。
J. DAC Changes	文件、信息、信号量的访问权限或所有权的改变。
K. DAC Denials	不允许的访问权限。
L. Admin / Operator Actions	系统管理和操作工作。
M. Insufficient Authorization	由于权限不够而造成的工作失败。
N. Resource Denials	丢失文件和内存不够。
O. IPC Functions	送信号或信息给进程。
P. Process Modifications	有效身份或工作目录的变化。
Q. Audit Subsystem Events	系统审计的生效、禁止和修改。
R. Database Events	安全性数据的改变和完整性。
S. Subsystem Events	受保护子系统的使用。
T. use of Authorization	只限于超级用户的活动。

### 7.1.2 审计特定的用户和用户组

User 和 Group 这两个审计选择域可为当前的审计阶段做动态改变，或可以影响下一个审计阶段。在同一个阶段中，用户和用户组的选择可以进行很多次。如果没有任何用户和用户组被选中，那么所有的进程都要受系统全局性审计掩码作用。

要针对于特定的用户和用户组进行审计，可进行如下的 Sysadmsh 选择：

System > Audit > Collection > IDs > Modify

一个类似下页的屏幕就显示出来。

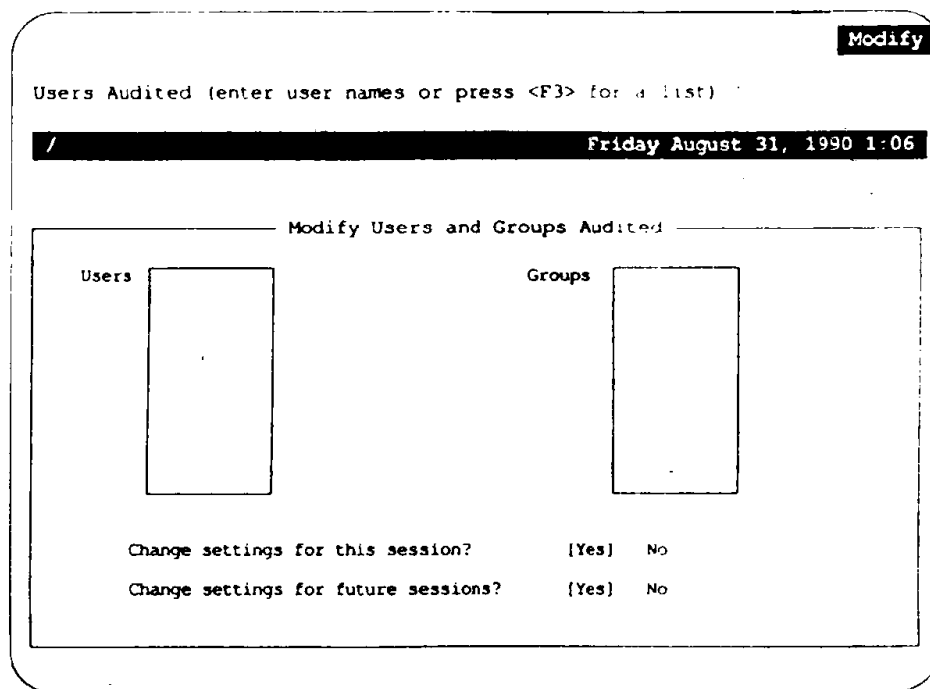
### 7.1.3 显示当前的审计统计

有一个选项用以给出对当前审计阶段的统计，其中提供如下信息：当前阶段号、采集和压缩文件的数目，核心审计机制所写的记录数和应用程序所写的记录数，以及其它信息。如果目前审计并不起作用，就不会显示各项统计。

要显示当前审计阶段的各项统计信息，可以进行如下的 Sysadmsh 选择：

System > Audit > Collection > Statistics

图 7-1 是统计选项的一个例子。



★★★ Audit Subsystem Statistics ★★★

Current Audit Session-6

Current Collection File Sequence Number-1488

Total count of audit data written:	7659433
Total count of audit records written:	156666
Audit records written by applications:	81
Audit records written by system calls:	155083
System calls not selected for audit:	751889
Total number of audit device reads:	2977
Total number of audit device writes:	324
Total number of collection files:	1489

图 7-1 审计采集统计信息范例

#### 7.1.4 启动与禁止审计

要打开或关闭审计，可使用如下的 Sysadmsh 选择：

System > Audit > Enable

System > Audit > Disable

启动功能使用当前的审计参数文件来进行审计子系统的初始化。禁止功能也在同一菜单中提供，并使系统正常退出审计（这时，守护程序读取所有的采集文件并对其进行压缩）。守护程序然后就结束工作，仅留下一个审计阶段的记录文件和这个阶段的压缩文件。

在审计进行期间，大多数子系统参数都可以修改，所以为了修改参数不必停止审计。启动和禁止功能都有一个确认屏幕，在 Sysadmsh 完成这一功能之前必须要对这些屏幕给予回答。在审计被启动或关闭时，就会显示一个信息来说明再引导时的审计状态。

### 7.1.5 维护审计行踪的连续性

有一个很重要的考虑是有关 LUID 和审计阶段的。图 7-2 是一个审计报告的一部分，它描述了一个遭到否决的文件访问，但所带有的用户 ID 是 root 而不是一个企图访问这个文件的未授权用户（这里要对文件 / a 做 touch 操作）。

```
Process ID: 227 (*INC*) Date / Time: Thu Dec 14 18:47:16 1989
Luid: root Euid: root Ruid: root Egid: root Rgid: root
Event type: Access denial
System call: Creat
Object: / a
Result: Failed-EACCES (Access denial)
Security policy: discretionary
```

图 7-2 审计报告的一部分

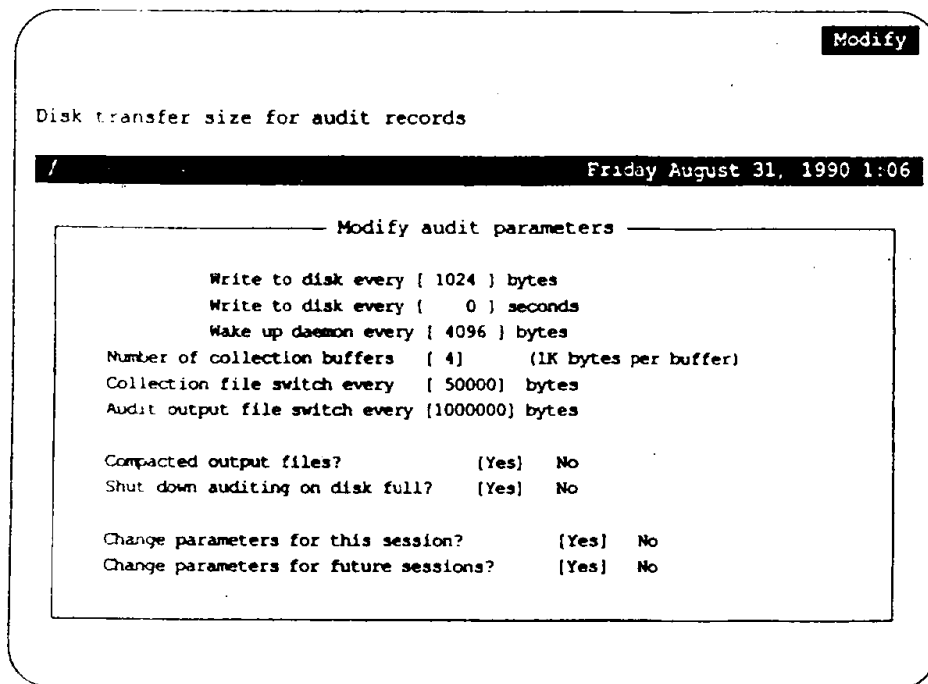
请注意进程 ID 旁边的 (\*INC\*)。这表示进程的审计行踪不完整。它说明审计是在这个用户注册之后才开始的，因此没有 LUID 被设置的记录，并且整理程序不认识它是什么。整理程序给那些不知道 LUID 的进程赋值为零 (root)。审计阶段必须包括被检查用户的注册号，否则审计子系统就没有这个用户 ID 的记录。保证这一点的唯一办法就是在允许用户注册之前开始审计。应该避免在系统已经开始活动后才开始一个审计阶段。

### 7.1.6 调整审计性能参数

可以通过改变一些审计参数来调整审计使其满足一个系统的需要。要检查当前审计参数的设置，可做如下的 Sysadmsh 选择：

System > Audit > Collection > Parameters > Modify

这时会显示如下的表格：



其中的一些参数与前面所讨论的权衡性能和可靠性有关。现在这应该变得更加明显了。这些参数如下：

Write to disk every [ ] bytes

Write to disk every [ ] seconds

这两个参数控制审计数据从内部审计缓冲区被同步写到审计采集文件中的频率。这种写盘操作可根据在写操作之前所积存的数据量来控制，也可由一个指定的时间间隔来控制。当所生成的数据量较小并且生成记录频率的时间分布性很广时，后者就是必要的。也可以同时定义字节数和时间来控制写盘。时间间隔总是以秒数来指定。

系统的性能可能会由于这两个参数的选择不当而受到不良影响。写操作过于频繁会使系统速度由于 I/O 负担过重而降低。另一方面，当这些值过大时，由系统故障而造成的数据丢失的可能性就会增大。一个好的经验是，当一个内部缓冲区被填满时就写一次盘。这样，把写盘的字节数定为 1024（一个内部缓冲区的大小）一般来说就足够了。

Wake up daemon every [ ] bytes

这个参数控制审计守护程序。这个守护程序不断地读审计设备并取得记录写到采集文件中。这些记录然后被压缩并写到压缩文件中供以后整理。为了最大限度地提高压缩算法的性能，守护程序需要读 4 至 5K 字节的数据块。这就要求审计子系统进行特殊的操作，因为一般的进程当有数据产生时，就进行读取，而不是等到数据积存到一定量时再读。为了获得最好的性能，这个参数应该被保持为缺省值 4K 字节。值大于 4K 字节也不会产生明显的更好效果。

Number of collection buffers

这用于指定被审计子系统用来采集数据的缓冲区的数目。审计子系统使用这些内部采集缓冲区来收集审计数据，为写采集文件作准备。使用多个缓冲区目的是为了系统的效率，因为所有的进程基本都要分享缓冲区空间来写记录。通过利用多个缓冲区，进程能够暂存审计记录，即使 I/O 正在对前面的缓冲区进行操作，进程也能继续执行而不会阻塞。至少需要两个缓冲区。大多数系统不能够有效地使用多于 4 至 6 个缓冲区来解决性能问题。没有一个确定的办法来计算缓冲区的最佳数目。一般情况下，根据希望系统所承担的进程来确定这个值。

Collection file switch every [ ] bytes

Audit output file switch every [ ] bytes

这两个参数指定采集和压缩文件最大可以长到多大时必须建立一个新文件。给各参数选取一个小值会造成过多的文件切换。因为压缩文件是永久性的，这就会引起系统上小文件的数量增多。选取一个很大的值则会造成这样一种形势，审计采集文件使用了大量的磁盘空间，即使它们只被审计守护程序读取一部分，而其余部分被丢弃。

审计压缩文件的大小可以被控制，因为这些文件一直保留在系统上直至被进一步整理或删除。希望这些文件工作在合理的大小上，包括能够方便地被保存和恢复。采集文件的大小缺省值为 50K 字节，而压缩文件为 1M 字节。一定要保证为压缩文件选取的最大大小不要超过为系统建立的 ulimit，它是用以控制一个用户文件的最大规模的。

Compacted output files

这个选项用以指定只需要非压缩文件。没有必要一定要这个选项，因为压缩文件并不



需要大量额外的处理时间同时又使磁盘的节省量大于百分之六十。压缩算法包含在用户所执行的审计守护程序中，而不是由审计子系统的核心部分完成的。

#### Enable audit on system startup

这个选项使得审计在每次系统重新引导时自动开始。这个域只有使用 View 选项时才能显示；它是依据审计是否起动而设置的。如果审计被禁止，那么在系统起动时，审计就被禁止。

#### Shut down auditing on disk full

这个选项允许系统在系统磁盘空间溢出的情况下自动关机，以防止数据发生错误。

#### Change parameters for this session

#### Change parameters for future session

屏幕上的最后两个选择可以允许动态地改变当前的阶段和/或为将来的阶段修改审计参数文件的永久部分。

## 7.2 文件管理

这部分将介绍如何选取审计记录文件所放置的位置，如何备份和删除旧的记录和如何维护被审计子系统所使用的磁盘空间。

有如下的文件功能：

List 列出系统上的审计阶段文件。

Backup 把一个审计阶段备份到后备介质上。

Delete 删除一个审计阶段。

Restore 从后备介质上恢复一个审计阶段。

一个审计阶段由一个阶段记录文件和审计子系统的一次启动和关闭之间所生成的一组压缩文件组成。在一个阶段中所创建的每一个采集文件和压缩文件都唯一地标有这个阶段号。当阶段结束，只保留记录文件和压缩文件。File Maintenance（文件维护）功能可以检查在系统上还有哪些阶段，并删去不想要的阶段。

### 7.2.1 改变审计目录的位置

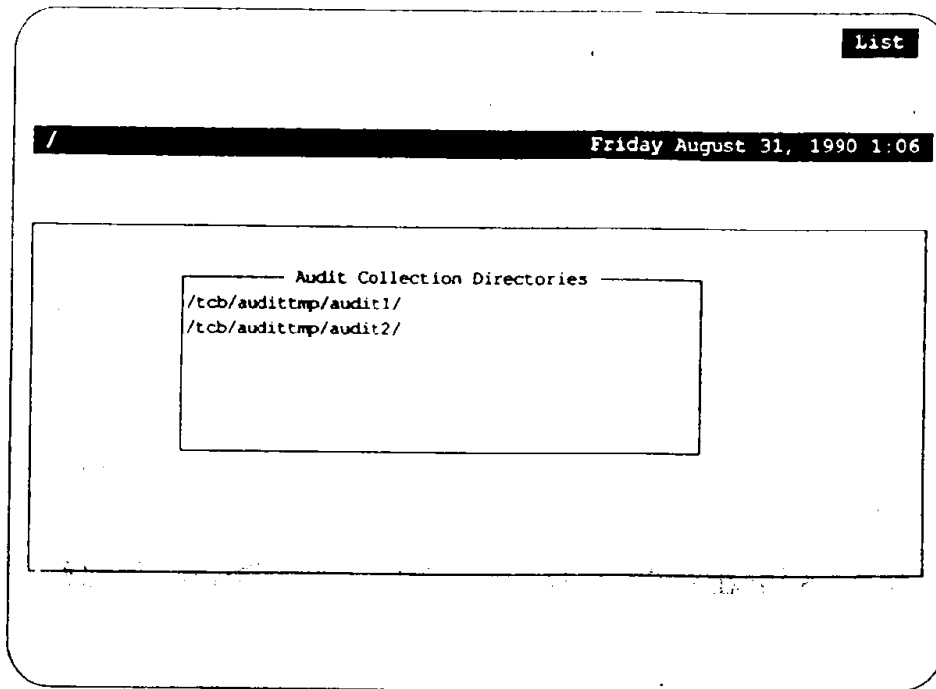
由审计子系统生成的采集文件和由审计守护程序生成的压缩文件都要写到指定的目录中。一个审计阶段可以包含被写到许多不同目录中的文件。在一个阶段的结尾，只有压缩文件保持着，这是因为当采集文件被审计守护程序读取之后它们就被审计子系统删去了。管理员不必记忆文件被写入的目录，因为一个阶段的记录文件已经保留了这个信息。

把审计目录放到与其它文件系统不在同一个物理设备上的文件系统中，会提高系统的性能。这可能减少对磁盘资源的竞争。另外，审计需要大量的空间，即使进行了压缩。当磁盘空间剩得很少时，审计子系统就会提出警告，如果文件系统的剩余空间变得过小时，甚至会停止审计。因此，审计子系统和守护程序支持多目录。如果在写一个目录时发生了错误或者空间耗尽，审计子系统和守护程序就会去使用另外的目录来继续工作。

要列出当前的审计目录，可作如下的 Sysadmsh 选择：

System > Audit > Collection > Directories > List

类似下面这样的屏幕会显示出来：

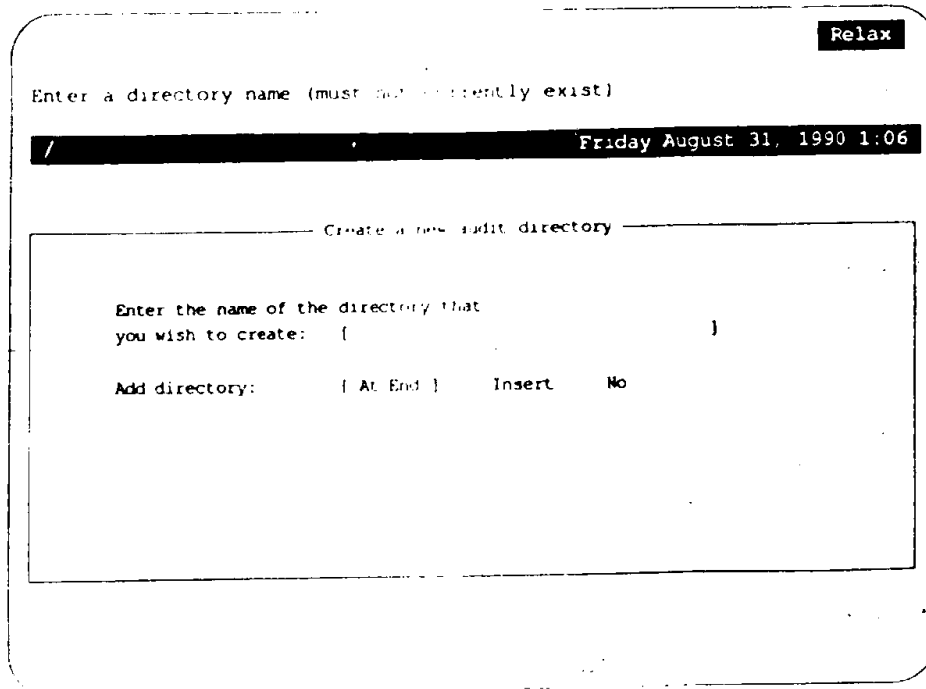


①增加一个采集目录

要给系统建立和增加一个新的采集目录，做如下的 Sysadmsh 选择：

System → Audit → Collection → Directories → Create

类似下面这样的屏幕会显示出来：



必须给出目录的完整路径名。增加目录的选项如下：

At End 在现存目录表的末尾增加新目录。

Insert 在现存的一个目录之前插入新目录。

No 不把这个目录加在这个采集目录表中。

要以绝对路径名的方式输入每个文件名。对于可以指定的目录没有人为的数目限制。

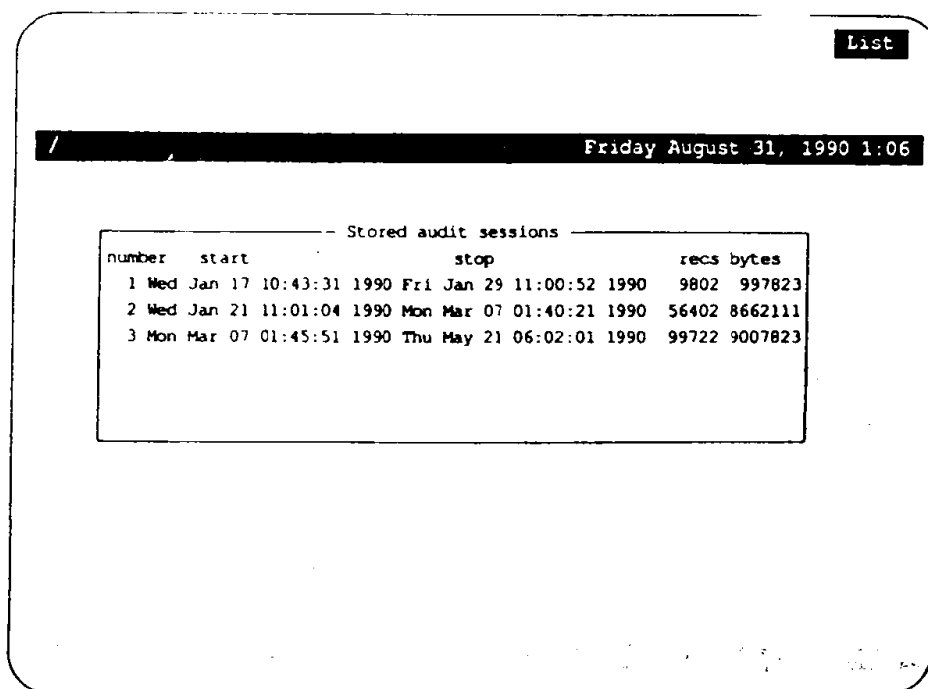
如果没有指定任何目录，审计子系统和守护程序就在根文件系统上使用保留的审计子系统目录 / tcb / audittmp 建立所有文件，这是缺省配置文件设定的。当向这些目录中填写数据时，它们就会被顺序使用；这就是必须指定它们的位置的原因。当阶段文件被备份并从审计目录中删去之后，系统就会把新的审计数据放到第一个目录中。

### 7.2.2 列出审计记录

要列出当前在系统上的审计阶段，做如下的 Sysadmsh 选择：

System > Audit > Files > List

一个类似下面的屏幕会显示出来：



在第一列上的数是阶段号。这张表是当从审计菜单的其它选项中要查询一个阶段号表时所显示的内容。

---

注：这里所显示的不是阶段数据文件表。它是那个阶段的记录数和字节数的总和；真正的数据被保存在一系列采集文件中。

---

### 7.2.3 备份审计文件

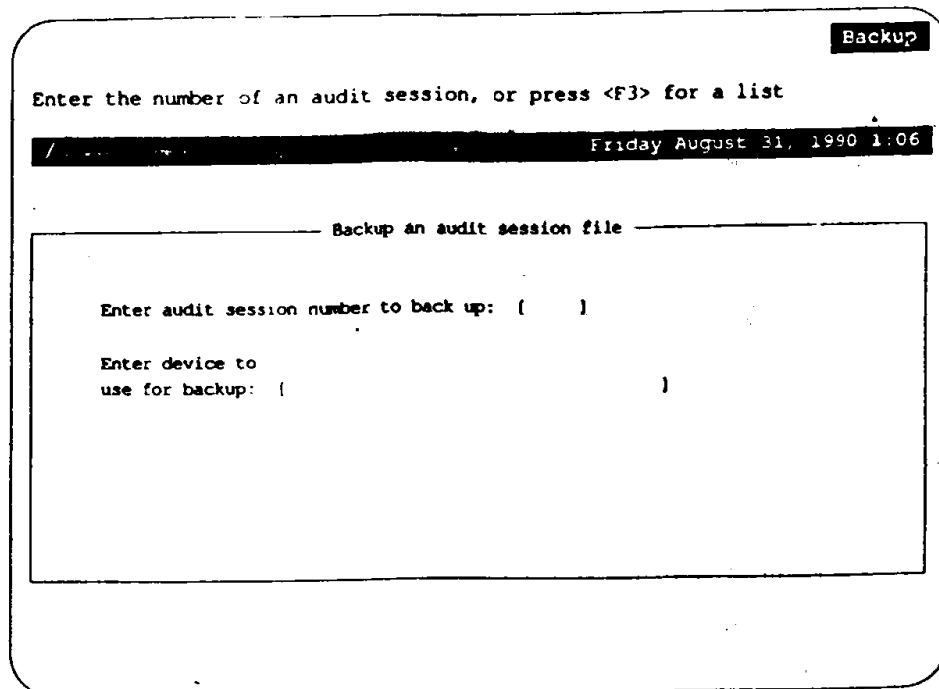
要备份审计文件，作如下的 Sysadmsh 选择：

System > Audit > Files > Backup

一个类似下页的屏幕显示出来。

因为审计阶段需要大量的磁盘空间，所以经常需要把审计数据归档备份，并待以后对其进行删减或保留一段时期，以备需要对那些没有立即发现的问题做进一步分析。备份和恢复接口程序提供了这一能力。Backup 选项需要一个阶段号作为输入。这可以通过按 <F3> 查询一个阶段表来取得。在选择了 Backup 之后，还必须为备份选一个输出设备。

这可以是系统上任何一个可移动的介质。



注：审计要消耗大量的磁盘空间。根据在系统上有多少用户和要对多少事件进行审计，可能需要几个星期就对阶段文件备份和删除一次。如果已经有了全系统的例行备份，就不必再进行审计备份了。另外，在文件被备份之后就应该把它们删除以释放磁盘空间。

#### 7.2.4 恢复审计文件

同样的，使用接口程序被备份到可移动的介质上的阶段可以使用 Restore 选项重新装回磁盘。要做这件事，先把包含有被存储的阶段文件的介质插入恢复设备，然后指定设备名。

要恢复审计文件，做如下的 Sysadmsh 选择：

System → Audit → Files → Restore

这时会被要求输入从哪个设备恢复，按 <F3> 会列出一张表。

#### 7.2.5 删除审计文件

要在文件备份之后删除它们，做如下的 Sysadmsh 选择：

System → Audit → Files → Delete

Delete 选项是用以删除审计阶段的。阶段可被归档到备份介质上并被删除来为更多的审计文件空出空间。阶段可以用阶段号来删除，阶段号可以按 <F3> 来得到。然后阶段号就被送给 Delete 选项，进而删除与此阶段相关的所有文件。

#### 7.2.6 监视磁盘空间的消耗

可以使用 System → Audit → Collection → Statistics 监视被审计记录消耗的磁盘空间。

所写的审计数据的总和就是当前存于系统上的字节数。

### 7.3 报告生成

整理功能使用一个叫“选择文件 (selection-file)”的文件来执行审计记录的后选择。这个文件由审计管理员接口程序依据管理员的输入而建立。可以建立并保存多个文件，而每一个都包括一组不同的选择标准。整理工作然后就可以针对同一个阶段的数据进行若干次，而每一次都可以使用一个不同的选择文件。因此，管理员可以建立并保存一些选择文件，每个文件都可以多次被用于数据的整理。当实际进行数据整理时，就可以使用已经建好的文件了。

Report 有如下选项：

List 列出所拥有的全部选择文件。

View 查看存于某个选择文件中的参数。

Creat 建立一个新的选择文件。

Modify 修改一个现存的选择文件。

Delete 删除一个现存的选择文件。

Generate 进行一次整理工作，指定审计阶段和选择文件。

正如前面所讨论的，审计采集标准给出了审计选择的第一层。在采集到数据之后，还可以进行进一步的处理或整理，而生成一个关于系统操作某一特定方面的有用的数据集。数据整理菜单可以对所需要的记录进行选择整理和判断。Generate 选项能支持一个很宽范围的后选择标准，可以帮助查找指定的事件、用户或目标。这需要给定阶段号和选择文件，这个选择文件可以使用选择文件建立或更新选项建立。

选项 List、View、Create、Modify 和 Delete 是用以选择文件的维护的。

#### 7.3.1 建立或修改选择文件

要建立或修改选择文件，可使用如下两个 Sysadmsh 选择之一：

System > Audit > Report > Create

System > Audit > Report > Modify

Create 和 Modify 选项的屏幕是完全相同的，Modify 允许对一个现存的选择文件做各种所需要的修改。

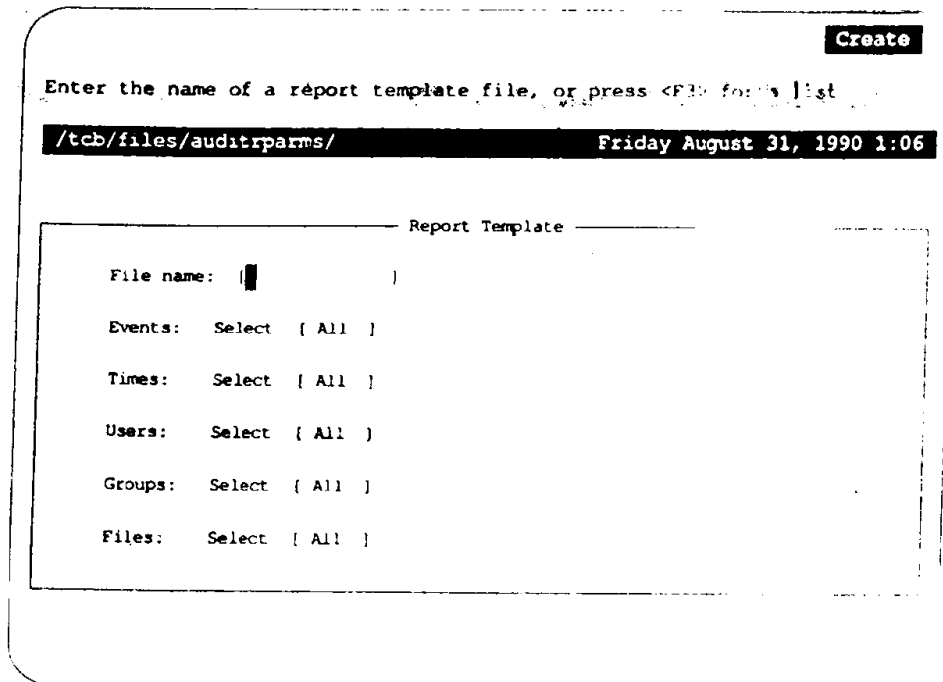
如下页的屏幕会显示出来。

输入要建立和修改的模板文件名。使用 <F3> 可得到一个模板文件表，其中有一些文件是系统提供的。

屏幕中有如下选择：

Events 对于审计事件，可以只选几类也可以选择全部的。对于那些没被选上的事件，相应的记录就会从输出中被除去。

Times 采集的起始和停止时间。如果在一天内的某一段时间内有一个值得怀疑的与安全性相关的事件，那么就可以利用这一特征来选择那些在那段时间里生成的记录。这就可以集中分析那些有可能揭示到底发生了什么的记录。



**Users / Groups** 用户和用户组都可以被单独提出来进行审计。如果某个用户帐号是一个被入侵的目标，就可以只选择那些由与那个用户相匹配的用户 ID 或用户组 ID 所生成的记录。这就使得所得到的记录都集中反映那个可疑帐号。

**Files** 文件（目标名）也可被用来从输出中选取审计记录。对于那些包含多个目标名的记录，如果这个指定的名字与记录中的任何一个目标相匹配，那么这个记录就要被选中。目标名必须以绝对路径名形式来指定，因为所有的目标名都被整理程序从相对路径名转化为绝对路径名。

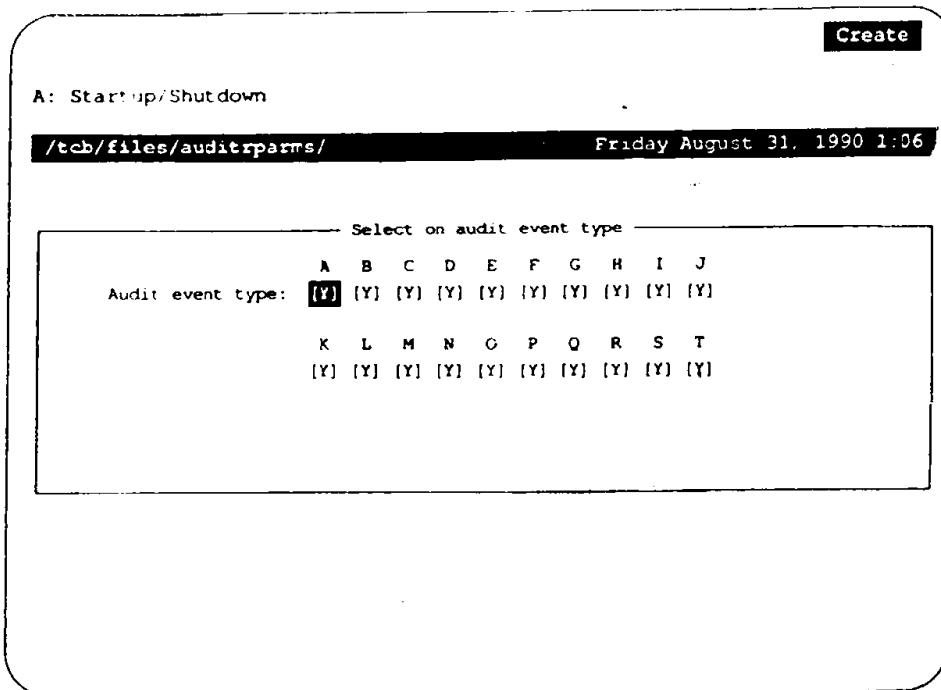
如果对某项选了 All，那么所有的事件、时间、等等都被选中了。如果选择 Select，就会显示一个窗口或表格，这样就可以分别选取所需要的标准了。

上面各标准的任何一个组合都可以使用。例如，时间区间，用户 ID 和目标名都可以在同一个阶段上组合起来。如果某个记录在这个指定的时间区间内由一个被选取的用户生成，并且在这个记录中还有一个被选取的目标，那么它就被选定输出。

对于记录的选取，有一个管理选择标准组合的优先顺序。如果某审计事件类型未指定，无论其它标准是什么，这个记录也不会被选中。同样的，如果指定了时间标记，但某记录并不符合这个标准，那么这个记录也不会被选中。如果某记录通过了事件类型和时间的选取标准，如果它还包含在选择文件中指定的一个用户 ID（注册的，有效的或真实的）、组 ID（有效的或真实的），或一个目标，那么这个记录就被选中。如果没指定用户、组和目标，那么就只根据事件类型和时间来进行选择。

#### 1) Events (事件)

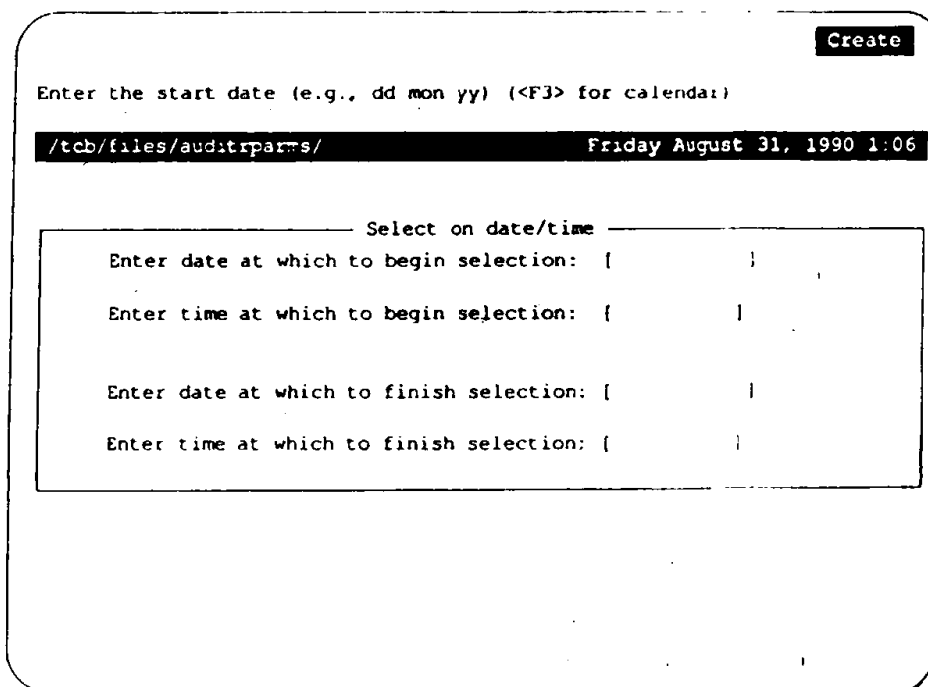
首先，Events 选择为高亮显示。如果选择 Select 并按了回车键，就会打开如下窗口，允许进行事件选择：



依照所选定的模板，各事件就会在括号中有一个 Y 或 N。要把一个事件从 yes 改换为 no，可使用空格键。当已经完全满意时，就按回车键来保存所做的修改。

### 2) Times (时间)

当 Times 选择为高亮度显示时，可使用箭头键来选取 All 或 Select，并按回车键。如果选了 Select，就会打开如下窗口选取开始和停止时间：



管理员会被要求依次输入开始和停止的时间。可以按 <F3> 获得一个日历。

### 3) Users (用户)

当 Users 选择为高亮度显示时，使用箭头键来选择 All 或 Select，并按回车键。如果选了 Select，就打开如下窗口来选择用户：

Create

Specify selection of users

/tc.../files/auditparms/ Friday August 31, 1990 1:06

Report Template

File name: [example ]

Events: Select [ All ]

Times: Select [ All ]

Users: Select [ All ]

Groups: Select [ All ]

Files: Select [ All ]

Users Selected

可以修改用户表和 / 或向窗口中的表里加入新的名字。写完这个表后，可按回车键。

#### 4) Groups (用户组)

当 Groups 选择为高亮度显示时，使用箭头键选择 All 或 Select，并按回车键。如果选择了 Select，就会打开如下窗口来选择用户组：

Create

Specify selection of group names

/tc.../files/auditparms/ Friday August 31, 1990 1:06

Report Template

File name: [example ]

Events: Select [ All ]

Times: Select [ All ]

Users: Select [ All ]

Groups: Select [ All ]

Files: Select [ All ]

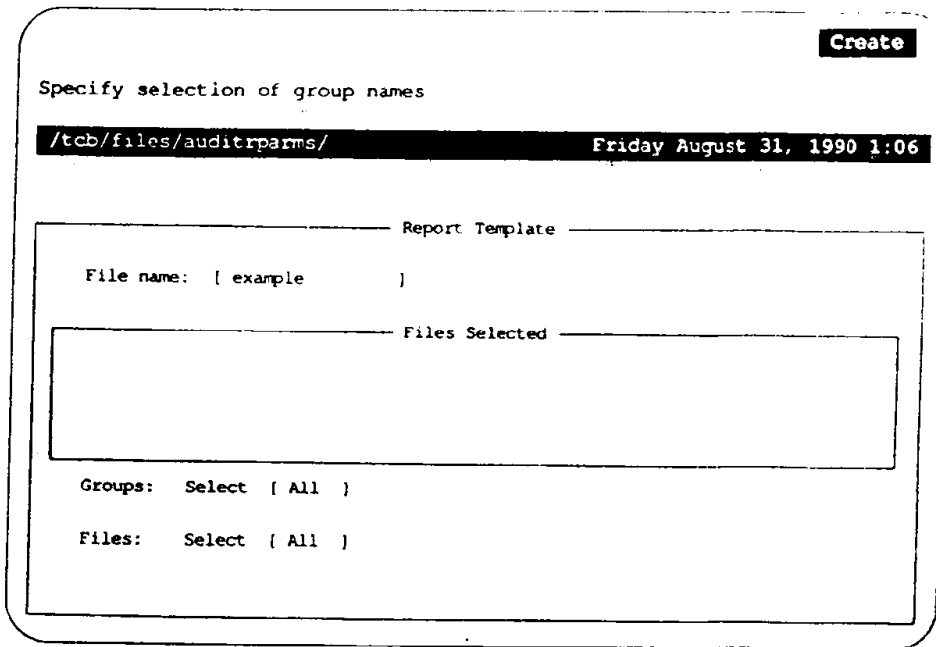
Groups Selected

可以修改用户组表和 / 或向窗口中的表中加入新名字。完成之后按回车键。

#### 5) Files (文件)

最后一项是 Files。如果在这里指定了文件（目标），它就要针对这里所指定的文件，找到对其删除，修改等的审计。使用箭头键来选择 All 或 Select，并按回车键。如果选择了 Select，就打开如下窗口选择文件：





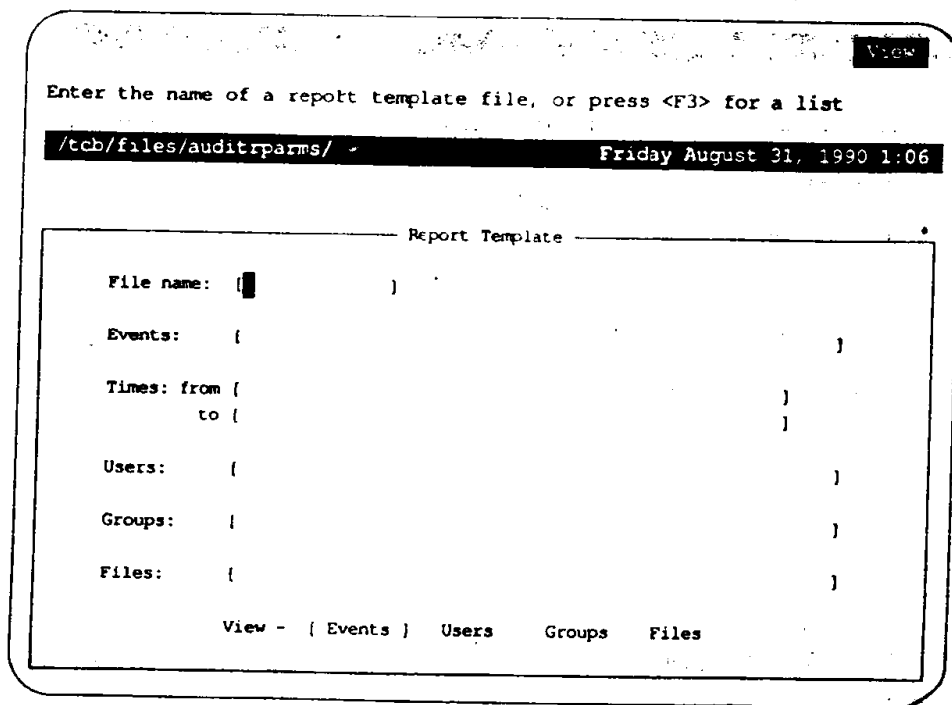
可以修改文件表和 / 或向窗口中的表里加入新名字。完成后按回车键。

### 7.3.2 查看选择文件

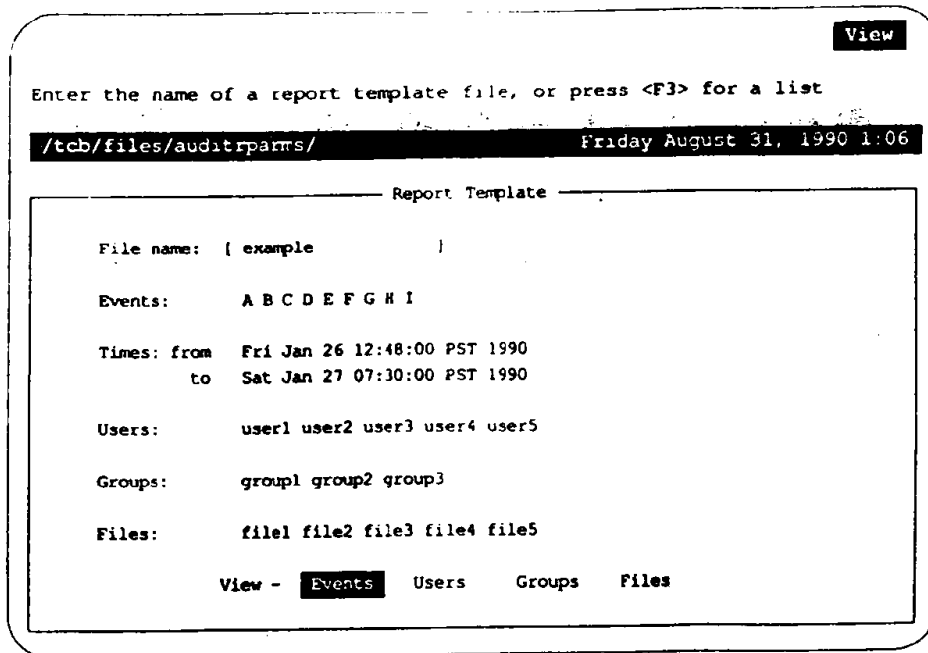
要查看一个选择文件，做如下 Sysadmsh 选择：

System → Audit → Report → View

如下屏幕会显示出来：



可使用 <F3> 键选取一个模板来看。不能使用这个功能进行修改；如果希望制做自己的一个模板，应该使用 Modify 选择。一旦选定一个模板，就可以像下面这个例子一样填各个域：



在屏幕最下端的各种选择是用以开辟窗口的，用来显示屏幕上不合适的各项数据。

### 7.3.3 列出选择文件

要列出所存在的选择文件，做如下的 Sysadmsh 选择：

System → Audit → Report → List

这时，一系列现有的报告模板就显示出来。下面是系统带来的文件：

- admin. actions      这些是管理员的活动；是事件类型L。
- all. objects        这些是有关建立、修改或删除目标的活动；是事件类型D-I。
- authorization      这是权限的使用；是事件类型M和T。
- dac. events         这些是DAC（Discretionary Access Control即分立访问控制）的改变或禁止；是事件类型J和K。
- denials             这些是DAC的禁止；是事件类型A-K和N。
- login. action       这是注册（成功和不成功的）和注销的一个记录；是事件类型B。

### 7.3.4 删除选择文件

要删除一个选择文件，做如下 Sysadmsh 选择：

System → Audit → Report → Remove

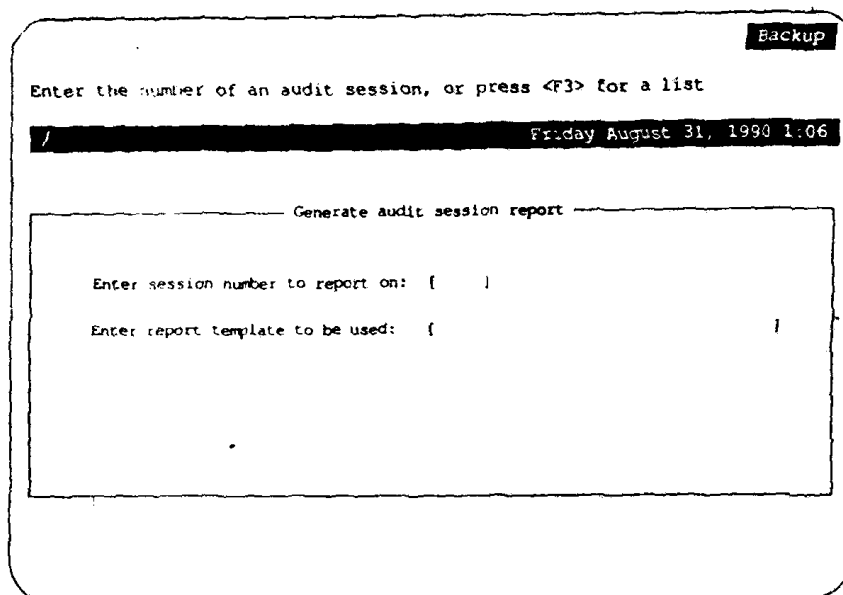
管理员会被要求提供要删去的选择文件（报告模板）的名字。按 <F3> 可得到一个文件表。

### 7.3.5 建立一个审计报告

要建立一个审计报告，做如下 Sysadmsh 选择：

System → Audit → Report → Generate

如下的一个屏幕将显示出来：



输入阶段号或按 <F3> 而获得一张表；对报告模板也同样。之后，就会被询问把输出送到哪里，送到终端、一个文件中或打印机。最好指示把输出送到一个文件中。当报告生成开始时，请注意，如果数据量很大就需要一定的时间。例如，如果选择文件没有指定开始和结束选择的日期和时间，那么就要对整个审计阶段进行整理归纳，这可能包含数十兆字节的数据。

报告的模板举例：

图 7-3 是根据带有如下特征的模板所建立的一个审计报告的实例：

```
Events: B K M T
Times: Start: Fri Feb 2 19:00 Stop: Fri Feb 2 21:00
Users: johnp
Groups: None
Files: All
```

这个报告模板把注意力置于不希望有的活动上，例如访问有限制的系统文件，运行有限制的管理程序等等。在这个简化了的例子中，用户 johnp 注册后想删除 /etc/passwd。在实际的情况中，应该有更多的检查记录。这个例子是用来展示审计数据的作用的。在下一部分“理解审计报告”中，有审计信息的详尽解释。

\*\*\*\*\*Audit Data Reduction Program\*\*\*\*\*

```
Audit session number: 2
Collection system name: unix
Collection file count:15
Compaction file count:1
Total audit records:11034
Total uncompactd size:696050
Total compacted size: 243262
Data compression rate: 65.05
Collection start time: Fri Feb 2 19:00:15 1990
```

Collection end time: Fri Feb 2 21:00:00 1990

\*\*\*\*\*Selection Criteria\*\*\*\*\*

Time Interval Selection:  
Start: Fri Feb 2 19:00:00 1990  
Stop: Fri Feb 2 21:00:00 1990  
Event Type Selection:  
Event type: Login / Logoff activity  
Event type: Access denial  
Event type: Insufficient privilege

UID selection in effect

johnp

\*\*\*\*\*Audit Records\*\*\*\*\*

Process ID: 235 Date / Time: Fri Feb 2 19:55:42 1990  
Event type: Login / Logoff activity  
Action: Successful login  
Username: johnp  
Login terminal: / dev / tty01

Process ID: 267 Date / Time: Fri Feb 2 19:56:22 1990  
Luid: johnp Euid: johnp Ruid:johnp Egid:group Rgid:group  
Event type: Access denial  
System call: Unlink  
Object: / etc / passwd  
Result: Failed-EACCES (Access denial)  
Security policy: discretionary

Process ID: 280 Date / Time: Fri Feb 2 19:58:14 1990  
Event type: Login / Logoff activity  
Action: Logoff  
Username: johnp  
Terminal: / dev / tty01

图 7-3 审计报告输出范例

### 7.3.6 理解审计报告

要解释审计行踪，需要先理解由程序所产生的记录及其含意。请记住审计记录有三个来源：系统调用、可靠的应用程序，以及受保护子系统。这三个来源所产生记录的格式有很大不同。另外，系统调用所产生的记录在内容上与其它记录还有很大差别，因为它执行的是特殊功能。例如，创建进程，fork (S)，只需要给出新创建进程的进程 ID 及其父进程的进程 ID。然而，对于 open (S) 这个系统调用，它必须作用于一个目标，这个目标的名字就必须被记录下来。对于像 mount (S) 和 link (S) 这样的系统调用，还必须记

录更多的信息；每一个都需要记录两个目标的名字。这个整理工具对提供给它的记录进行分类并以某种组织形式输出信息。

所输出的记录可被划分成两类：由核心审计机制所产生的系统调用记录以及应用程序审计记录。有些记录项对于所有输出记录都是共有的。例如，记录的日期和时间，与这个记录相关的进程 ID，它们对于每种类型的记录，都要显示出来。除此以外，记录的内容就要依据审计的内容了。

#### 1) 系统调用的记录格式：

在审计行踪中，系统调用记录占大多数。这个操作系统包含 60 多个系统调用。并不是所有这些系统调用都要审计，因为只有其中的一些是与安全相关的。只有一半多一点儿的系统调用有可能建立审计记录。有些系统调用支持多种功能（例如 `fcntl (S)`、`msgsys (S)`、`shmsys (S)` 和 `semsys (S)`），而其中可能只有某些功能会建立审计记录。另外，系统调用可能执行的是可审计事件的功能，但在此时这一事件又被系统事件掩码给屏蔽了。

就大部分而言，系统调用的输出对于所有记录都是相同的。不同点也存在，因为有些系统调用对目标进行操作（例如 `open (S)`），而目标名都要包含在记录中。每个记录都至少包含时间、日期、进程 ID、系统调用名、事件类型、注册用户 ID、实际用户和组 ID、有效用户 ID 和组 ID、以及调用成功或失败的指示。

每个输出的记录都包含三个基本信息域及其它依系统调用而定的域。基本记录请见图 7-4。这描述了系统调用的公共头和结果域。

```
Process ID: 68    Date / Time: Sat Mar 5 13:25:09 1988
Luid: root Euid: root Ruid: root Egid: root Rgid:root
Event type:
System call:
Result:
```

图 7 4 输出的公共记录头

每个系统调用都要根据所完成的动作被归为一个系统事件类型。这描述了系统调用的事件类型。实际的系统调用名称也被给出。在多数情况下，这唯一地标识了一个动作。不幸的是，有些 UNIX 系统调用是超负荷的，它使一个系统调用入口点被用来完成多个动作。例如，`msgsys()` 是对信息队列进行 IPC 操作的系统调用入口。这一个入口点调用了 `msgget (S)`、`msgop (S)`、和 `msgctl (S)` 来执行某个 IPC 功能。

像这样的系统调用是不能自我解释清楚的。审计子系统知道这些超负荷调用，并提供了另外的信息来说明每一种功能。对于成功的系统调用，结果被确定为成功。对于每个返回错误值的调用，错误还提供了另外的记录类型。例如，由于权限不足而造成的 `open (S)` 失败就被划归访问否决（`access denial`）。一个生成了审计记录的不成功的系统调用在结果域指明了出错情况。

系统调用的输出记录可被分为两组。第一组包含那些在审计记录中无需路径名的记录。例如，`fork (S)` 系统调用是用来在向系统中创建了新进程后而跟踪它们的，但审计记录并不需要路径名。而另一方面，`open (S)` 必须为指定的路径名返回一个文件描述符。后续的操作，如 `close (S)`，要使用这个文件描述符。为了提供一个有意义的审计记

录，第二类记录必须包含路径名。通过使用整理功能，这个路径名与作用于此文件上的进一步的动作联系起来，即使这一动作是利用文件描述符来完成的。

图 7-5 列出了不包含路径名信息的可审计的系统调用。

```
pipe fork kill setuid setgid exit
read setpgrp msg sem shm write
```

图 7-5 无路径名的系统调用

上述各系统调用所产生的输出记录使用了图 7-4 描述的一般的记录掩码。下图所示的例子描述了由一个成功的 setuid (S) 系统调用所产生的输出记录。

```
Process ID: 6381    Date / Time: Tue Mar 15 11:25:19 1988
Luid: blf Euid: blf Ruid: root Rgid: root Rgid: root
Event type: modify process
System call: Setuid
Result: Successful
```

图 7-6 Setuid (S) 系统调用记录

类似的，图 7-7 给出的也是 setuid (S) 系统调用所产生的输出记录，这个调用由于对此文件访问权限不够而失败。请注意错误反映于结果域。

在这一组中的很多系统调用在输出记录中还要产生另外的信息以协助说明审计行踪。信号量、共享存储器、信息队列和 security (S) 系统调用都是超负荷的。它们对应多种功能。这些审计记录要对所完成的特定功能和所影响的目标（例如共享内存）进行识别。close (S)、dup (S) 和 fcntl (S) 要对文件描述符进行操作，这些文件描述符是由路径名映射来的。一个记载对一个文件描述符所进行的 dup (S) 的输出记录是没什么用的，因为它不能唯一标识一个文件。因此，这个文件描述符还要联系一个路径名并在这个记录中显示这个路径名。

```
Process ID: 6381    Date / Time: Tue Mar 15 11:25:19 1988
Luid: blf Euid: blf Ruid: blf Egid: guru Rgid: guru
Event type: Modify process
System call: Setuid
Result: Failed (EPERM)-Not owner
```

图 7-7 访问否决的输出记录

虽然 read (S) 和 write (S) 这两个系统调用都被列在图 7-5 中，但它们只在某些情况下受到审计并且也没有一个专用的记录。这两个系统调用都只是在对某一个文件第一次发生作用时才被审计。而后续的读和写就不再需要审计了，因为它们不会再提供更多的信息。审计记录是用来跟踪一个文件的状态的。当文件由于被 exec (S)、exece (S)、close (S) 或 exit (S) 所作用而关闭时，在系统调用记录中就会包含这个目标的名字和文件读或写的指示符，来记录这个文件被关闭的动作。这在图 7-8 中有具体描述。

```
Process ID: 421    Date / Time: Sat Mar 5 17:15:09 1988
Luid: blf Euid: blf Ruid: blf Egid: guru Rgid: guru
Event type: Make object unavailable
```

```
System call: Close
File Access-Read: Yes Written: No
Object: /tmp/datafile
Result: Successful
```

图 7-8 close (S) 系统调用记录

第二组系统调用，见图 7-9，把路径名包含为输出记录的一部分。路径名代表了系统调用的作用对象。有两个系统调用记录实际上要包含两个路径名：Link (S) 和 mount (S)。

```
open unlink creat exec chdir mknod chown
chmod stat umount exece chroot link mount
```

图 7-9 带路径名的系统调用

图 7-9 中的每个系统调用都要带一个或更多的路径名作为这个系统的变元。这些路径名受到审计并且成为这个整理进程的重要部分。这些调用的输出记录指明了被作用的目标名。这个名字也要保留给整理程序，并且在适当时与系统调用所返回的文件描述符联系起来。这也为其它对文件进行操作但又不包含路径名的系统调用，譬如 dup (S)，提供了一个对应关系。图 7-10 给出了由 creat (S) 这个系统调用所产生的一个输出记录。这个记录格式就是带有路径名变元的基本格式。

```
Process ID: 64    Date / Time: Sat Mar 5 23:25:09 1988
Luid: root Euid: root Ruid: root Egid: root Rgid: root
Event type: Object creation
System call: Creat
Object: /tmp/daemon.out
Result: Successful
```

图 7-10 输出带路径名的记录

在这一组中的所有调用都使用同一种路径名的格式。有两个系统调用，link (S) 和 mount (S)，要对两个路径名进行操作：一个源和一个目的。这两个名字都要受到审计，并由一个整理程序把审计结果反映到输出记录中。图 7-11 给出了由 link (S) 系统调用产生的一个典型记录。

```
Process ID: 14231    Date / Time: Thu Mar 16 03:25:39 1988
Luid: lp Euid: lp Ruid: lp Egid: lp Rgid: lp
Event type: Object creation
System call: Link
Source: /tmp/printfile
Target: /usr/spool/lp/lp3014
Result: Successful
```

图 7-11 带两个路径名的输出记录

在这组中的另外两个记录会产生特殊的输出记录。它们是 chown (S) 和 chmod (S)，它们是用来改变分立访问权限和文件所有权的。由于其动作与安全性密切相关的特性，目标所有者及组和状态的原值与新值都要输出到记录中。图 7-12 描述了系统调用

chmod (S) 的输出记录。

```
Process ID: 6841    Date / Time: Sat Mar 5 13:25:09 1988
Luid: blf Euid: blf Ruid: blf Egid: blf Rgid: blf
Event type: Discretionary Access Change
System call: Chmod
Object: / tmp / demo / newfile
Old values: Owner-blf Group-guru Mode-100600
New values: Ower-blf Group-guru Mode-100666
Result: Successful
```

图 7-12 chmod (S) 系统调用记录

## 2) 应用程序的审计记录

由应用程序产生的审计记录有六种类型。它们的格式都是类似的。与系统调用不同，由这六类中的任何一类所产生的记录的格式都是完全相同的，只是信息不同。这些类型是：

- 注册和注销事件
- 用户口令事件
- 受保护数据库事件
- 审计子系统事件
- 被授权的子系统事件
- 终端和用户帐号封锁事件

每个记录都包含一些所有审计输出记录共用的信息。这包括进程 ID、时间和日期、审计事件类型。输出记录的其余部分依记录类型而定。各记录特定的域在下面各部分将分别进行介绍。

## 3) 注册 / 注销记录

所有要在系统上注册的企图都要受到 login 程序的审计。无论对于成功的和不成功的注册都是如此。这就建立了一个重要的用户访问系统的行踪，同时也建立了一个不成功访问的记录。可以使用注册和注销的审计记录来判断谁实际上使用了系统。对于判断是否曾有过对系统进行的重复的入侵企图，这种记录也是很有价值的。这个操作系统还支持在若干次不成功入侵企图之后就封锁终端这一特性，这种事件也要被审计。因此，管理员具备所有监视（和防止）对系统访问所需的工具。

每个审计记录都包含一个所审计的特定动作的指示符。三种可能性是：成功注册、不成功注册、或注销。所有成功的注册和注销都会产生一个审计记录，以指明这个注册阶段的用户帐号和终端。对于不成功的注册，用户名是没有意义的，因为这次尝试失败了。在这种情况下，只有这次尝试所在的终端与基本记录域一起被输出。图 7-13 描述的是由一次成功的注册所产生的输出。

```
Process ID: 2812    Date / Time: Fri Mar 4 10:31:14 1988
Event type: Login / Logoff Activity
Action: Successful login
Username: blf
Terminal: / dev / tty2
```

图 7-13 成功注册的审计输出记录



#### 4) 用户口令记录

所有要修改一个用户帐号口令的尝试，无论成功与否，都要被审计子系统认真地审计。出于安全的考虑，这些事件的审计记录都不包含口令正文，但指明被审计的帐号和动作。这些动作被划分为成功的口令改变，不成功的改变和口令改变权限不足。图 7-14 给出了一个不成功的口令改变的审计记录。

```
Process ID: 7314    Date / Time: Tue Mar 1 18:30:44 1988
Event type: Authentication database activity
Action: Unsuccessful password change
Username: blf
```

图 7-14 不成功口令改变审计记录

#### 5) 受保护数据库记录

对系统受保护数据库进行维护和修改的程序要对所有与这些数据库相关的访问和异常情况进行审计。其范围包括从一致性问题到与安全性相关的失败事件。除记录头和特定的审计动作之外，输出中还包括检测问题的程序名、此问题所影响的目标、期望值和实际值、以及动作和该事件的结果。请看图 7-15。

```
Process ID: 7314    Date / Time: Tue Mar 1 18:30:44 1988
Event type: Authentication database activity
Command: authck
Object: Protected password database
Value: Expected-O Actual-O
Security action: / tcb / files / auth / code
Result: extraneous file in protected password hierarchy
```

图 7-15 受保护数据库输出记录

#### 6) 审计子系统记录

影响审计子系统操作本身的事件也要受到非常严格的审计。sysadmsh 审计选择和审计守护程序 anditd 都要对它们所支持的功能生成审计记录。另外，审计设备驱动程序也要对某些功能需求写审计记录。受审计的功能如下：

- 子系统初始化
- 子系统终止
- 子系统参数修改
- 审计守护程序使能
- 审计守护程序禁止
- 子系统关闭
- 子系统出错

每个输出记录都包括公共的头信息以及被审计功能的描述。这就对所有影响审计子系统操作的企图提供了一个准确的说明。图 7-16 给出了一个所写的实际审计记录，它指明了审计子系统的启动和初始化。

```
Process ID: 517    Date / time: Wed Mar 2 8:30:04 1988
Event type: Audit Subsystem activity
Action: Audit enabled
```

图 7-16 审计子系统输出记录

#### 7) 受保护子系统记录

每个受保护子系统都可以通过审计守护程序生成审计记录。这些记录指明了被审计子系统发现的异常情况。例如，如果一个子系统遇到了关于一个文件的访问权限问题，或由于内存或其它资源不足而不能工作，那么审计子系统就会对这一结果产生一个错误信息。可以利用这些记录来协助维护系统的安全。

除了输出正常的记录头之外，审计子系统记录中还包含一个命令名、动作、及结果。这个命令名是那个检测出不一致错误和写审计记录的命令。这个动作和结果说明了子系统所执行的动作和所发现的问题。图 7-17 给出了一个子系统所产生的审计记录。

```
Process ID: 2812    Date / Time: Fri Mar 4 10:31:14 1988
Event type: Authorized Subsystem activity
Subsystem: System Administrator Subsystem
Security action: Update / etc / rc
Result: Cannot open for update
```

图 7-17 受保护子系统审计输出记录

#### 8) 终端和用户帐号记录

如果不成功注册的次数超过一定数目，这个数目存于授权数据库中，用户帐号或终端就会被封锁。例如，如果一个终端被用来进入系统，但结果却是一连串不成功的注册，那么在试了一定次数这后，login 程序就要封锁这个终端。同样，如果一个用户企图注册进入一个帐号，但又接连失败，那么那个用户帐号就要被封锁。封锁了帐号和终端就会防止后续访问，直到系统管理员清除了封锁。终端或用户帐号的封锁可以标志有人企图侵入系统。这些审计记录包含一般的头信息和用户帐号或终端的标识符。

```
Process ID: 517    Date / Time: Wed Mar 2 8:30:04 1988
Event type: System administrator activity
Action: User account locked ty system adminixtrator
Username: root
```

图 7-18 用户帐号封锁输出记录

### 7.4 授予用户审计权力

把一些审计能力分配给用户也是可能的。可以允许用户对他们自己的活动生成审计记录。第二级子系统权限 audittrail 允许在 System → Audit → Report 下访问审计功能的一个子集。记录的输出就限制在与用户的 LUID 相匹配的记录。用户可以使用所有的报告选择，包括报告模板的建立。它也与系统的模板存在一起。

audittrail 权限的分配在本书“用户帐号的管理”一章中的“改变或分配用户权限”中进行了讨论。

## 7.5 审计术语

审计采集文件 (audit collection file) 是由审计子系统设备驱动程序所写的文件, 其中包含从系统中所有审计资源, 包括系统调用、可靠的应用程序、和特许子系统, 得来的原始审计数据。

审计压缩文件 (audit compaction file) 是由审计守护程序所写的文件, 其中包含从审计设备驱动程序得来的并曾在其中缓冲暂存过的数据。这些数据可以是压缩的或不压缩的格式, 这由在这个审计阶段开始时所选择的选项而定。

审计守护程序 (audit daemon) 是一个守护进程, 它开始于系统转为多用户状态时。它从审计子系统设备读取审计记录, 并把这些记录压缩, 写到一个永久的压缩文件中, 留待以后进行整理。守护程序也起一个接口的作用, 它允许非受保护子系统向审计设备中写入审计记录。

审计阶段 (audit session) 是从审计生效到审计禁止之间的一段时间。在此期间, 审计数据被守护程序写到压缩文件中。每个阶段都被唯一地编号, 每一个属于这个阶段的文件都在文件名中包含这个唯一的 ID。对于每个阶段都有一个主文件来搜集阶段信息和阶段文件名, 供以后进行整理。

审计子系统 (audit subsystem) 由提供可靠的审计服务的各单元组成。其中包括审计设备驱动程序、核心审计机制、审计守护程序、审计管理员接口程序和审计整理工具。

审计行踪 (audit trail) 是从一个审计阶段得到的审计数据记录的集合, 它可以被整理为一份系统活动的报告。

审计整理 (audit reduction) 是从原始审计行踪数据到包含有日期、用户 ID、文件名和事件类型的输出记录的一个转化过程。输出记录以一种可读的文本形式描述了被审计事件。

审计配置 (configaudit) 是指一种核心授权, 它允许为系统上的所有用户设置一组审计参数。

事件控制掩码 (event control mask) 是一个保存在受保护口令数据库中的针对每个用户的用户专用掩码。这个掩码用来控制在审计启动时用户的事件掩码是否优先于系统的缺省事件掩码。在控制掩码中被置为“1”的每一位都会使事件处理掩码获得优先权。

事件处理掩码 (event disposition mask) 是一种用户专用掩码。它与事件控制掩码联合使用来进行用户审计事件的控制。如果用户事件控制掩码把某一位置为“1”, 那么在事件处理掩码中的相应的那一位就被用来决定这一事件是总要被审计还是绝不要审计。不管系统的缺省事件掩码是什么, 这一点都是正确的。

事件类型 (event type) 是每个审计记录的分类。系统中与安全性相关的事件可被划分为某些类型, 这可用以控制审计的生成或整理。每个系统动作, 无论是成功的还是失败的, 都可被归为一个事件类型。这个事件类型然后决定记录的处理。

目标 (object) 是被一个主体 (譬如文件、共享内存段、信号量、管道、或信息队列) 所作用的一个实体。

后选择 (post-selection) 是对所采集的审计数据的有选择的使用。后选择要涉及所采集到的全部事件和用户的审计数据, 所以审计行踪应尽可能的完整。在一个阶段的末

尾，任何与安全性相关的事件都被记到审计行踪压缩文件中。

预选择 (preselection) 有选择地控制审计记录的生成。这使得某些用户和事件生成审计记录而其它的则被丢掉。结果是得到一个更加紧缩审计行踪，但不如使用全审计全面。

选择文件 (selection file) 是由管理接口程序产生的用以控制审计阶段的有选择的归纳。选择所用的标准可以控制输出记录的用户、目标、和事件选择。

主体 (subject) 是一个对某一目标执行某一动作的活动实体，譬如系统中访问文件的一个进程。

审计挂起 (suspendaudit) 是一种挂起审计的核心权限。

系统审计掩码 (system audit mask) 是一个缺省的系统事件掩码，它用以决定当用户进程掩码无优先权时要审计什么事件。

用户审计掩码 (user audit mask) 与系统缺省掩码一起共同参与事件控制和事件处理的掩码，针对每个进程来控制审计记录的生成。

写审计 (writeaudit) 是一种核心权限，它允许审计行踪记录特定的信息。

## 第八章 UNIX 目录和特殊的设备文件

本章将列出在一些 UNIX 系统上最常用的文件和目录。其中许多文件和目录要求正确的操作，并且一定不能被删除或修改。下面各部分将简要地对各目录进行介绍。

本章还包含建立有关文件系统和终端的设备节点的内容。

### 8.1. UNIX 目录

本节将讨论操作系统的各主要目录。

#### 8.1.1. 根目录

根目录 (/) 包含如下系统目录:

/bin UNIX 的命令目录

/dev 设备的特殊目录

/etc 附加程序和数据文件目录

/lib C 程序库目录

/mnt 安装目录 (为所安装的文件系统保留)

/usr 用户服务例程 (可包含用户主目录)

/tcb 属于 TCB (可靠计算基础库) 的一部分系统文件

/tmp 临时目录 (为由程序产生的临时程序保留) 上面所有的目录都是系统工作所需的。根目录也包含一些普通文件。在这些文件中，最应注意的是文件 /unix，它包含 UNIX 核心映像。

#### 8.1.2. /bin 目录

/bin 目录包含最常用的 UNIX 命令，也就是说，这些命令可被系统上的任何用户使用。

#### 8.1.3 /dev 目录

/dev 目录中包含控制对外设进行访问的特殊设备文件。这个目录中的所有文件都是必需的，一定不能删除。在 /dev 目录下有几个子目录。每个子目录都保存着与某一类设备相关的特殊的设备文件。例如，/dev/dsk 目录就包含用于软盘和硬盘的设备文件。这个操作系统对 XENIX 和 UNIX 设备的命名规则都给予支持。在合适的地方，/dev/dsk 目录中的文件被连接到存在于 /dev 中的设备文件上。可以通过 /dev 中的文件来访问某一设备，也可以通过 /dev 的一个子目录中的文件来访问同一设备。

表 8.1 包含了部分设备。

表 8.1 /dev 设备节点

UNIX 设备	XENIX 设备	名字
/dev/console	同左	系统控制台
/dev/rdisk/*	/dev/r*	原始设备
/dev/dsk/0s0	/dev/hd00	驱动器 0 中的整个磁盘
/dev/dsk/0s1	/dev/hd01	驱动器 0 中的第一个磁盘分区
/dev/dsk/0s2	/dev/hd02	驱动器 0 中的第二个磁盘分区
/dev/dsk/1s0	/dev/hd10	驱动器 1 中的整个磁盘
/dev/dsk/1s1	/dev/hd11	驱动器 1 中的第一个磁盘分区
/dev/dsk/1s2	/dev/hd12	驱动器 1 中的第二个磁盘分区
/dev/dsk/f05d9	/dev/fd048ds9	360KB 的软盘驱动器 0
/dev/dsk/f05q	/dev/fd096ds9	720KB 的软盘驱动器 0
/dev/dsk/f05h	/dev/fd096ds15	1.2MB 的软盘驱动器 0
/dev/dsk/f03h	/dev/fd0135ds18	1.44MB 的软盘驱动器 0
/dev/lp	同左	行式打印机
/dev/kmem	同左	核心虚拟存储器
/dev/mem	同左	物理存储器
/dev/null	同左	空设备
/dev/rmt0	/dev/rct0	QIC 磁带设备
-	/dev/rft0	QIC-40 磁带设备
-	/dev/rctmini	小型盒式磁带设备
/dev/root	同左	根文件结构
/dev/swap	同左	交换区
/dev/tty	同左	终端

#### 8.1.4 /etc 目录

/etc 目录包含各种系统程序和数据文件。所有文件都是需要的，但大多数可以修改。在目录 /etc/rc.d 和 /etc/rc2.d 中的数据文件包括有一些当系统进入多用户模式时 /etc/rc2 所运行的初始化命令。

在目录 /etc/default 中的数据文件包含系统命令所用的缺省信息。下面的数据文件可以修改，但任何文件都不允许删除。请看表 8.2。

表 8.2 /etc/default 文件

文件	用处
/etc/default/archive	sysadmsh (ADM) 备份缺省信息
/etc/default/authsh	sysadmsh (ADM) 缺省帐号信息
/etc/default/backup	backup (ADM) 缺省信息
/etc/default/boot	boot (ADM) 信息
/etc/default/cleantmp	cleantmp (ADM) 缺省信息
/etc/default/cron	cron (C) 缺省记录信息
/etc/default/dumpdir	xdumpdir (ADM) 缺省信息
/etc/default/filesys	sysadmsh (ADM) 缺省文件系统数据
/etc/default/format	format (C) 缺省信息

文件	用处
/etc/default/goodpw	goodpw (ADM) 缺省口令检查数据
/etc/default/idleout	idleout (M) 缺省信息
/etc/default/lang	缺省地点信息
/etc/default/lock	lock (C) 缺省信息
/etc/default/login	login (M) 缺省信息
/etc/default/lpd	lp (C) 缺省信息
/etc/default/man	man (C) 在线人页缺省信息
/etc/default/mapchan	mapchan (M) 缺省信息
/etc/default/micnet	micnet (M) 缺省信息
/etc/default/msdos	DOS 磁盘的位置 (A, B, .....
/etc/default/passwd	passwd (C) 缺省信息
/etc/default/purge	purge (C) 缺省信息
/etc/default/restor	xrestore (ADM) 缺省信息
/etc/default/su	su (C) 缺省信息 (请注意, 管理员必须自己建立这个文件)
/etc/default/tape	tape (C) 缺省设备信息
/etc/default/tar	tar (C) 缺省设备信息
/etc/default/usemouse	usemouse (C) 缺省信息

### 8.1.5 /lib 目录

/lib 目录包含 C 和其它语言运行所需的库文件。这个目录是必需的。

### 8.1.6 /mnt

/mnt 目录是一个空目录, 它是为安装可移动文件系统而保留的。

### 8.1.7 /usr 目录

/usr 目录由一些子目录组成, 在这些子目录中包含另外一些 UNIX 命令和数据文件。它还是用户主目录的缺省位置。

/usr/bin 目录包含另一些 UNIX 命令。这些命令用得不多, 或被认为不是对 UNIX 系统操作所必需的。

/usr/include 目录包含 C 程序编译所需的头文件。

/usr/lib 目录包含不同的 UNIX 命令所用的另一些库和数据文件。

/usr/spool 目录包含不同的子目录, 各子目录中所存的文件是用以打印, 电子邮件或通过网络进行传递的。

/usr/adm 目录包含有与系统管理和记帐相关的数据文件。特别要提的是, /usr/adm/messages 文件包含有送给系统控制台的所有错误信息。这个文件对于诊断硬件问题特别有用。例如, 某一驱动器的一个异常磁盘错误号就说明这个驱动器有缺陷或没调准。由于这个文件中的信息会很快积累起来, 因此必须定期删除这个文件。

### 8.1.8 /tcb 目录

/tcb 目录包含组成 TCB（可靠计算基础库）的全部文件。这些文件增强了这个操作系统的的天性，使它比其它 UNIX 操作系统更加安全。安全性特征在本书“系统安全性维护”一章进行讨论。

### 8.1.9 /tmp 目录

/tmp 目录包含由 UNIX 程序所产生的临时文件。这些文件一般只在相应的程序运行时才存在，但如果程序过早地停止了，它们也会留在此目录中。可以删去不属于正在运行的程序的任何一个临时文件。

## 8.2 记录文件

许多目录都有一个记录文件，并且在系统正常运行过程中不断扩大。其中许多文件必须定期清理，以防它们占用过多宝贵的磁盘空间。（详见“文件系统的管理”一章中的“清理记录文件”）表 8.3 列出了这些文件及其内容。

表 8.3 系统的记录文件

文件名	说明
/etc/ddate	每个备份日期的记录
/usr/adm/pacct	记帐信息的记录；当进程记帐进行时，会增长得很快。
/usr/adm/messages	系统启动时所产生的错误信息的记录。
/etc/wtmp	用户注册和注销记录
/usr/adm/sulog	su 命令各次使用的记录；只有当在 /etc/default/su 文件中做了设置，它才增长。管理员必须建立 /etc/default/su。
/usr/lib/cron/cronlog	at(C)和 cron(C)命令各次使用的记录。
/usr/spool/micnet/remote/*/*LOG	在一个 Micnet 网络上各机器间所进行的传输的记录。（*）必须是与当前机器相连的一个远程机器的名字。
/usr/spool/uucp/.LOG/utility/sitename/*	在一个 UUCP 网络上使用的 UUCP 命令的记录。utility 和 sitename 分别是 uucp 实用程序和远程现场的名字。
/usr/spool/uucp/.LOG/.old/*	由 uudemmon.clean 保存的老记录文件的目录。

## 8.3 特殊设备文件

本书所讲的许多文件系统的维护工作都需要使用特殊的文件名、块容量、以及间隙和块数等概念。下面各部分将对每个概念作详细讲述。

### 8.3.1. 特殊的文件名

特殊的文件名是设备特殊块或字符 I/O 文件的名称，它对应与类似硬盘或软盘驱动器这样的外设。在像 mkfs (ADM)、mount (ADM) 和 df (C) 这样的命令中，需要用



这些名字来指定要建立、安装或搜索的包含有文件系统的设备。

表 8.4 列出了典型计算机上硬盘或软盘驱动器的 XENIX 和 UNIX 特殊文件名及相应的设备。

### 8.3.2 容量

一个磁盘的块容量是磁盘上所有的存储空间的块数，这里一块是 1024 字节的存储量。

表 8.4 磁盘设备文件名

文件名	磁盘驱动器
/dev/fd0	软盘驱动器 0
/dev/dsk/f0	软盘驱动器 0
/dev/fd1	软盘驱动器 1
/dev/dsk/f1	软盘驱动器 1
/dev/hd00	整个硬磁盘
/dev/dsk/0s0	整个硬磁盘
/dev/root	根文件系统
/dev/u	用户文件系统

大多数命令以 512 字节为一块来报告磁盘空间，特别是 df (C)、du (C)、ls (C)、lc (C) 和 find (C)。一个在以 1024 字节为一块的文件系统中的 500 字节的文件就要被这些实用程序报告使用了两个块，因为这个文件使用了一个系统块，一个系统块等于两个 512 字节的块。一个 40 兆字节的硬盘容量，以 1024 字节块来算，是 39168。要注意，磁盘上的一些块被保留给系统使用而不能被用户程序访问。一般软盘的块容量依磁盘全部的存储能力而定。

### 8.3.3 间隙和块的数目

间隙和块的数目是被 mkfs (ADM) 和 fsck (ADM) 命令用来描述如何安排磁盘上的块的。表 8.5 列出了一个典型计算机所用的软盘和硬盘的间隙和块的数目。

表 8.5 间隙和块的数目

磁盘	间隙	块
软盘, 48ds9	1	9
软盘, 96ds15	1	15
软盘, 135ds9	1	9
软盘, 135ds18	1	18
硬盘	1	34

块的数目也可以这样来定，以每一道的扇区数（一般为 17）乘以硬盘磁头的数目，再除以 2（因为每扇区有 2 块），然后再把这个值化为最近的整数。

#### 8.3.4 终端和网络的要求

enable (C) disable (C) 命令用来在系统上增加和去掉终端。前面的命令和选项要求给出串行线的名字，系统通过这些串行线与终端或网络连接。表 8.6 列出了这两条串行线（实际是两个带或不带调制解调控制的串行端口）的设备特殊文件名。与这些串行线相对应的字符 I/O 文件可在 /dev 目录中找到。请注意，文件 /dev/console 和从 /dev/tty01 到 /dev/tty12 代表硬连线设备，而不是为连接终端或硬件所用的。

表 8.6 串行设备

文件名	线
/dev/tty1a	主串行线（不带调制解调控制）
/dev/tty2a	备用串行线（不带调制解调控制）
/dev/tty1A	主串行线（带调制解调控制）
/dev/tty2A	备用串行线（带调制解调控制）

## 第九章 文件系统的备份

系统管理员的主要任务是保证系统中所存信息的连续完整性。文件和文件系统的损坏或数据丢失可能产生于如下情况：

- 电源中断（一定要有一个过压保护装置）
- 硬件故障（特别是硬盘）
- 用户错误（意外地删去了十分重要的文件）

拥有最新的备份是非常重要的。如果系统上有一些活动的帐号，必须注意每天备份。由于事故的发生而造成几个星期或几个月的工作毁于一旦，这种损失的严重程度在事前是很难意料的。

一个文件系统的备份就是一个存于介质（软盘或磁带）上的根文件系统和其它正常安装的文件系统中的文件的副本。（请看“文件系统的管理”一章中有关文件系统的讨论）备份就是要求系统管理员（或具备 backup 权限的用户）保存一个文件系统在一个特定时间的副本。

本章将介绍如何使用 sysadmsh (ADM) 来建立根目录和其它文件系统的备份，以及如何从备份中恢复文件。（还有一个用于简单备份的实用程序 tar (C)，这个程序还不够复杂，不能够进行例行的备份；它特别适于—组文件的归档）。

在本章中所讨论的工具提供了带有简单选项的菜单，而不像一些软件工具如 tar (C)、cpio (C)、backup (ADM) 和 restore (ADM) 那样需要使用复杂的命令行。有效备份的关键是要保存每天所修改的内容，一般还需要特别的书面记录（当使用了 backup 和 restore）。

### 9.1 使用 sysadmsh 进行备份的策略

作为系统管理员，应该熟悉这一章的内容并按所讲述的内容建立例行备份。当这一次例行备份完成之后，必须并且只要插上一个介质卷标号并归入一系列日常备份中。

sysadmsh 文件系统备份选项的主要目的是要给带有很多用户和大的文件系统的系统提供可靠的文件系统备份。这个程序能够自动定位于被修改的文件并把它们拷贝到备份介质上。如果系统上有很多用户和大量的在这一天中被修改了的文件，那么这个例行备份的选项就会使用一个预先定好的安排来进行有规律的备份。当这个 Backups 选项被调用时，程序就会把每项任务作为一个菜单选项给出来。要执行某个任务，只要从菜单中选取合适的选项并提供所需要的信息就可以了。

为了进行非正规备份，sysadmsh 还包括一个“非例行”备份的选项。这可允许系统管理员对一个文件系统进行单独的、完全的备份（请注意，这种类型的备份要覆盖整个文件系统，而不仅仅是被修改了的文件，并可能需要一定数量的存储介质卷）。如果打算依靠非例行的备份，至少一个星期进行一次。

#### 9.1.1 backup 权限的使用

必须给一个用户赋予 backup 权限，使其能够建立或恢复备份（要恢复一个完整的文件系统，管理员必须是根）。普通用户不能进行备份，因为他们没有访问所有文件的权限。如果作为根来进行备份，那么文件可能会被意外地毁坏，因为根对于系统上的每个文件都有无限的权力。backup 权限由于只有有限的根的权限而解决了这个问题。

### 9.1.2 软盘备份和大系统

如果系统上只有一个软盘驱动器，那么对于一个拥有若干用户的大系统的备份就会很费时，并需使用大量的软盘。对一个 20 兆字节的文件系统的完全备份需 15 张 1.2 兆字节 96tpi 的软盘，而一盘 450 英尺长的磁带可以存储的信息量是这个信息量的两倍多。更重要的是，软盘需要操作人员留在现场插入和取出磁盘，而一盘磁带只要被塞入，操作人员就不必留在系统旁。如果系统上有很多用户且只有一个软盘驱动器，那么必须安装一个磁带机，或每星期进行一次彻底的系统备份，并且提醒用户对他们自己的文件进行有规律的单独备份。

### 9.1.3 所用的实用工具

sysadmsh 在备份过程中需要用到几个实用工具。管理员不必熟悉它们。然而，如果想进一步使用这些工具，就必须知道如何去使用它们以及去哪读有关内容。sysadmsh 用到如下工具：

- fsphoto (ADM) 是用于控制自动备份实用程序的主要工具。
- fsave (ADM) 是与用户交互来进行备份的程序。
- schedule (ADM) 是一个备份数据库，其作用是指定要用的介质、要备份的文件系统以及何时做这件事。

- xbackup (ADM) xrestore (ADM) 是 XENIX 备份工具。当“xbackup”出现在安排表 (Schedule table) 的“Method”域时，这两个工具就要被取用。（这些工具是提供给那些希望继续使用他们在 XENIX 下所使用的同一种备份工具的人的。cpio 是一种更好的方法）。

- cpio (c) 是缺省的备份程序。它不指定文件系统。

sysadmsh 的 Backups 选项为用户提供了一个友好的界面，它把用户与上述程序复杂的句法分隔开来。

## 9.2 例行备份的准备

例行备份唯一的强制需求是建立一个备份安排表。另外，系统管理员最好能够遵循下面对备份的标记、存储、登记的过程。

### 9.2.1 建立一个备份安排表

第一步是要使用 Schedule 文件为备份建立一个时间表。这个文件位于 /usr/lib/sysadmin 目录中，它包括系统进行系统备份所需的所有数据，其中包括：

- 现场或机器的名称
- 介质的类型和要用的驱动器

- 要备份的文件系统的准确的安排表。

下面所要介绍的是要对 schedule 文件进行哪些修改。

### 1) 编辑 schedule 文件

可以用任何一种文本编辑器对 schedule 文件进行编辑；但必须要以根的身份登录。

图 9-1 给出了缺省的 schedule 文件。还可以使用下面的 schedule 选择来编辑 schedule 文件：

Backups → Schedule

schedule 以 Vi (C) 作为缺省编辑器，但也可以把环境参数 SA-EDITOR 设置为自己所喜欢的编辑器。

### 2) 写入现场或机器的名称

只要把文件顶部的 machinename 改为自己想要的名字就可以了。

### 3) 选择与配置相匹配的介质驱动器

根据所提供的介质，缺省的驱动器或者是 96tpi1.2M 字节或者是 135tpi1.4M 字节的软盘驱动器 0。96tpi 驱动器的描述请见图 9-2。

```
#SYSTEM BACKUP SCHEDULE
site machinename
#Media Entries
#
#96 tpi 1.2 MB floppy 0
media / dev / rfd096ds15 k 1200 format / dev / rfd096ds15
#96 tpi 1.2 MB floppy 1
#media / dev / rfd196ds15 k 1200 format / dev / rfd196ds15
#135 tpi 1.44 MB floppy 0
#media / dev / rfd0135ds18 k 1440 format / dev / rfd0135ds18
#135 tpi 1.44 MB floppy 1
#media / rfd135ds18 k 1440 format / dev / rfd135ds18
#Cartridge tape 1
#media / dev / rct0 k 60000 125000 150000 tape erase
#Mini cartridge drive (10MB)
#media / dev / rctmini k 8800 format / dev / rctmini
#Mini cartridge drive (20MB)
#media / dev / rctmini k 17200 format / dev / rctmini
#Mini cartridge drive (40MB)
#media / dev / rctmini k 37500 format / dev / rctmini
#9-track tape drive
#media / dev / rmt 0 d 1600 2400 1200 600
#backup Descriptor Table
#Backup Vol.      Save for      Vitality      Label
#level  size      how long      (importance)  marker
  0      -        "1 year"      critical       "a red sticker"
  1      -        "4 months"    necessary      "a yellow sticker"
  2      -        "3 weeks"     useful         "a blue sticker"
  3      -        "1 week"      precautionary  none
```

```

#Schedule Table
#          1 2 3 4 5      6 7 8 9 0      1 2 3 4 5      6 7 8 9 0
#Filesystem M T W T F      M T W T F      M T W T F      M T W T F Method
/dev/rroot  0 3 3 3 3      2 3 3 3 3      1 3 3 3 3      2 3 3 3 3 cpio

#Alternate schedule for systems with / u filesystems
# /dev/rroot 0 x 3 x 3      2 x 3 x 3      1 x 3 x 3      2 x 3 x 3 cpio
# /dev/ru    3 0 3 3 3      3 2 3 3 3      3 1 3 3 3      3 2 3 3 3 cpio

```

图 9-1 schedule 文件

```

#96 tpi 1.2MB floppy 0
media / dev / rfd096ds15 k 1200 format / dev / rfd096ds15
#96 tpi 1.2MB floppy 1
#media / dev / rfd196ds15 k 1200 format / dev / rfd196ds15

```

图 9-2 缺省的介质登记项

符号#是注释符，用以注释出正文，这样运行正文就会被程序忽略。请注意，缺省驱动器是唯一不带注释符的。如果打算使用非缺省的驱动器，那么就在 96tpi 驱动器前放一个注释符，并把要使用的驱动器前注释符去掉。其它的驱动器依然应该保持为被注释状态。

#### 4) 编辑备份描述表 (Backup Descriptor table)

紧跟在介质驱动器行下边的是备份描述表。这张表，其描述见图 9-3，描述了每个备份级的容量、保存的时间、重要程度，以及如何标记。这些缺省登记项将被证明是实用的，但卷的大小一定要按照所用介质的类型来进行编辑。

#Backup	vol.	Save for	vitality	Label
#level	size	how long	(importance)	marker
0	-	"1 year"	critical	"a red sticker"
1	-	"4 months"	necessary	"a yellow sticker"
2	-	"3 weeks"	useful	"a blue sticker"
3	-	"1 week"	precautionary	none

图 9-3 备份描述表

如果正使用的是软盘，那么就把“vol.size”改为其真正的值。这就会使得备份程序能够从那个设备的介质登记项中取到其容量。

如果使用的是磁带或盒式磁带，那么把磁带的容量（以 KB 为单位）填入“vol.size”这一列。如果所使用的磁带对于各备份级其容量都相同，那么就以所使用磁带的这一种容量来进行替换。

最后一列所包含的标记登记项将在后面的“给备份做标记”中讨论。

#### 5) 编辑备份安排表

缺省的安排表假定备份每天都要做。为了使备份更有效，它们被分为几个等级。0 级是最低备份级。0 级的备份把文件系统上的每件东西都保存起来，而 1、2 和 3 级只备份相对于前一级备份修改过的文件。这个概念在图 9-4 中给以描述，它用一堆玩具盘代表各等级。请注意，0 级的盘是最大的备份，并且其它的每一个都逐渐减小。这是因为 1 级仅包含 0 级备份之后所修改的文件，以此类推。这个图也说明了这些备份如何恢复：首先恢复 0 级，跟着恢复最近所做的 1、2 和 3 级备份。

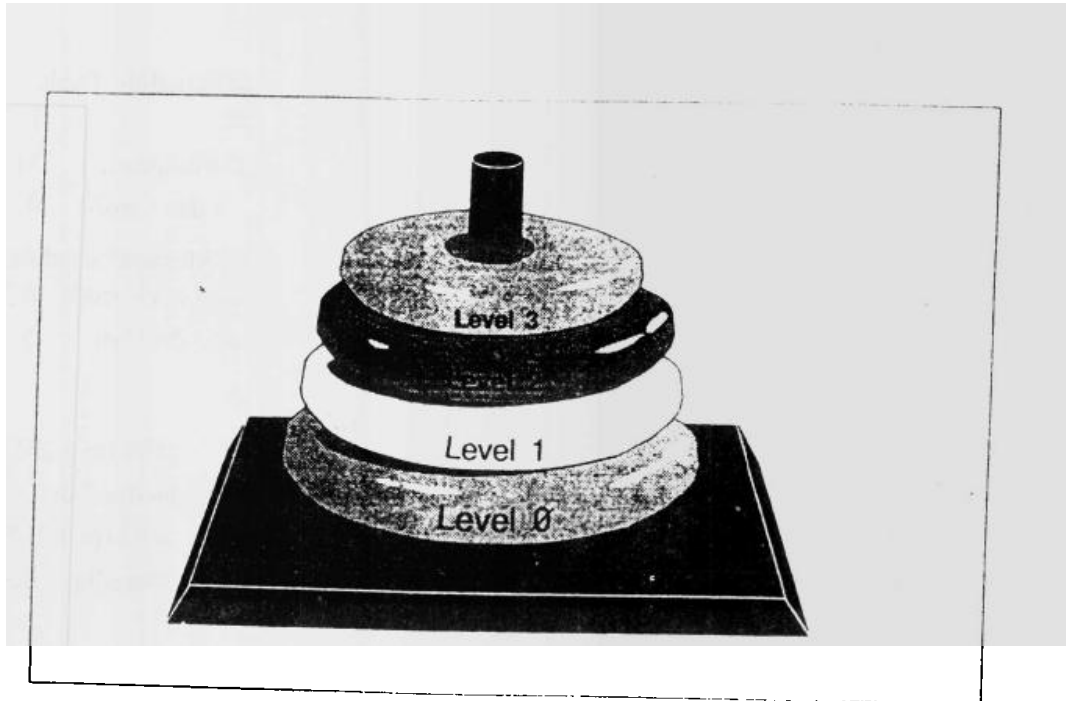


图 9 4 备份等级

这种分级的概念在一开始可能被看作是一种不必要的复杂作法，但应该考虑对于一些大的文件系统会发生什么情况。如果每天晚上都要对各个文件系统做一个完整的备份，那么这个过程会需要进行几个小时，系统也会在这个过程中陷入瘫痪。如果只保存最近修改过的文件，备份将会大大节省时间，还会依据介质的容量，少消耗一些介质卷。（有关备份级的更进一步的讨论请看本章末尾的“备份级的说明”）。

在本章中所给的 schedule 文件中还包括另一个安排表，是有关 /u 文件系统的，它也被变为注释了。注意，对于根文件系统隔一天做一次备份而对于 /u 文件系统每天都要做一次备份。这是因为 /u 文件系统（用户帐号）比根文件系统变化得更加频繁，根文件系统只包含系统文件。“x”表示对于那个文件系统在那一天不做备份。

如果没有 /u 文件系统，用户帐号就放在根文件系统中（在目录 /usr 中）。如果确实是这样，那么安排表就必须先规定为对根文件系统进行备份。然而，如果已经加了一个 /u 文件系统，那么就对安排表进行编辑并把登记项 /dev /ru 前面的 # 移去，如图 9-5 所示。这就保证了备份是对另外增加的文件系统做的。如果没有 /u 文件系统，但又想每天都做备份，这个登记项也可被改用到根文件系统上。

#	1 2 3 4 5	6 7 8 9 0	1 2 3 4 5	6 7 8 9 0	Method
#Filesystem	M T W T F	M T W T F	M T W T F	M T W T F	
/dev /rroot	0 x 3 x 3	2 x 3 x 3	1 x 3 x 3	2 x 3 x 3	cpio
/dev /ru	3 0 3 3 3	3 2 3 3 3	3 1 3 3 3	3 2 3 3 3	cpio

图 9 5 备份安排表

请注意，图 9-5 中星期一至星期五的标志可以是错误的；如果一个备份被延迟或没有成功（例如因为介质坏了），那么同一级的备份就会在下一次例行备份时再做一次。这样就会偏离原来的日期安排，但不必修改已经建成的备份序列。在 M-F 上面的 1-0 的数字标度是很准确的，但对以天和星期为工作单位的人来说是没什么用的。

另外，如果已经为其它文件系统而加了另外的行，那么应该注意不要在同一天安排做

两次大型文件系统的 0 级备份；因为这个备份过程太长并且会明显地降低机器的运行速度。

### 6) 备份方法域

缺省的备份形式类型是 cpio。如果希望使用用于 XENIX 文件系统的 xbackup 形式，那么就用“xbackup”代替“cpio”。请注意 xbackup (ADM) 只工作于 XENIX 文件系统，因为它使用了特定文件系统的信息。cpio 形式可以完美地作用于 UNIX、XENIX 和 AFS 文件系统，因为它并不使用这种信息。因此最好不要使用 xbackup 形式。

---

注：请记住 AFS 是本操作系统所使用的缺省文件系统类型。

---

## 9.2.2 给备份做标记

用有意义的和准确的信息对备份磁带进行标记是很重要的。如果备份是由一堆随意标记的磁带组成的，那么在以后的日子就很难找到所需的数据了。

图 9-6 是介质标签的推荐形式。

计算机名称	备份级	备份日期
	文件系统名	
	备份截止日期	
备份人员的名字		卷号、总卷数

图 9-6 介质标签样本

标签上的日期以及备份截止日期，应该是这个备份所覆盖的工作日的开始与结束日期。这样在需要从这盘磁带上恢复信息时就会避免混淆。

可能已经注意到在 schedule 文件中有一种推荐的颜色码体制以易于查询，请看图 9-7。

#Backup	vol.	Save for	vitality	Label
#level	size	how long	(importance)	marker
0	-	"1 year"	critical	"a red sticker"
1	-	"4 months"	necessary	"a yellow sticker"
2	-	"3 weeks"	useful	"a blue sticker"
3	-	"1 week"	precautionary	none

图 9-7 备份标志体制

如果一个备份有多于一盘磁带，就必须在各卷上标记日期的同时标记卷号及卷的数目，例如对于一个两卷的备份就要分别标记“1of 2”和“2of 2”。最后，把这个记有计算机名、文件系统和备份级的标签放到介质包装盒的侧面。

## 9.2.3 保存一个登记本

最好为每个机器都保存一个登记本。除了记载维护信息（例如何时系统产生了故障以及对此采取了什么措施），还应该记录如下信息：



日期：与磁带标签上的一样，这个日期应该是这个备份所覆盖的最后一天。

文件系统：这是在当前磁带上所备份的设备名称。

备份级：这是当前磁带的备份级。

卷数：这是磁带卷的数目。

开始/结束时间：(可选)从文件系统的一次备份开始到最后一个错误检查完成为止的一段时间。这个时间在这次备份完成后显示出来。结束时间通常是不准确的，因为当备份结束时管理员可能没在这个房间里，在管理员回来之前机器已经处于等待状态了。

如果备份产生了错误，也要把它记到登记本上，其中包括屏幕上出现的任何错误信息。

#### 9.2.4 备份介质的循环使用

在备份介质的使用中，至少应该有6到12个月的介质保存有文件。在缺省的 `schedule` 文件中包含有这样的循环使用建议：3级备份保存一个星期，2级备份保存三个星期，1级备份保存四个月，0级备份保存一年。这表示如果按这个缺省的安排表来执行，那么在一个星期之后再使用3级备份依然是安全的，依此类推。

#### 9.2.5 备份介质归档

所有的文件系统都应该周期性地备份并归档于另外的地方。万一遇到火灾或自然灾害，数据也能够恢复。

### 9.3 进行例行备份

这部分将介绍如何使用已定好的安排表进行备份。在对 `schedule` 文件进行编辑(或至少是检查了)并确认它完全适合需要之前千万不要开始进行备份。

系统管理员应该在系统上没有用户时进行备份，以确保正确地拷贝每个文件的最新版本。

正常的备份需要有很好的介质供给，并对介质进行合适的保存。0级备份应该至少保存一年，如果备份很重要，还要保存得更长些。其它级的备份都应至少保存两个星期。介质卷上应该正确地标记以该备份所进行的日期以及其中所包含的文件和目录名。在一个备份过期之后，这个介质就可用来建立新的备份了。

#### 9.3.1 使用格式化过的介质

如果使用需要格式化的介质，例如软盘或小型盒式磁带，就应该在进行备份之前对其进行格式化。介质卷的确切数目决定于要备份的文件的数量和大小。有关如何格式化介质的详细说明，请看“使用软盘和磁带驱动器”一章。从 `sysadmsh` 程序中也能找到进行格式化的选择，但不能在备份进行过程中格式化介质(请注意小型盒式磁带要花很长时间来格式化)。

#### 9.3.2 开始备份

进行例行备份，应遵循如下步骤：

1.调用 sysadmsh 并选择:

Backups→Create→Scheduled

2.这时会显示类似下面的一个菜单:

Level 0 backup of filesystem / dev / rroot, 31 Aug 1990

tape size: 1200kb

tape drive: / dev / rfd096ds15

This tape will be saved for 1 year, and is critical.

M) ounted volume, P) ostpone, C) heck or F) ormat volums, R) Retension of H) elp:

所显示的介质类型是那个在 schedule 文件中设置的类型。把一个介质卷, 磁带或磁盘放入所选的驱动器中, 键入 m 告诉这个程序这一卷已装入了, 并按回车键。

3.系统显示当前日期以及最后一次备份的日期:

Level 0 backup of filesystem: / dev / rroot

Backing up all files

Generating list of pathnames for backing up...

这个过程要持续几分钟。

4.系统然后就开始向驱动器中拷贝文件。如果一卷的空间用尽了, 程序就会显示如下信息:

Reached end of medium on output

Insert volume 2 and press <RETURN> to continue or 'q' to exit.

---

注: 如果所用的是 5.25 寸的软盘进行备份, 一定要在按回车键之前关上软盘驱动器的门。

---

把当前这一卷移走, 插入新的一卷, 然后按回车键。程序就会继续把文件拷贝到新卷上。重复这一步骤, 直至程序显示信息:

n blocks

check critical volumes for format errors

5.当备份完成之后, 将显示如下菜单:

M) ounted which volume, S) kip format check, or H) elp

0 级备份一定要检查格式错误。键入 m 检查介质。如果正在检查格式, 一定要按指示插入第一卷, 否则备份就会夭折。如果不想检查各介质卷, 就键入 s。

6.如果产生了错误, 这次备份就会被宣布为不成功并且必须从头再做一遍。介质也可能是坏的, 因此如果总出现错误就换掉它。下面的菜单对各介质卷的检查进行掌握:

M) ounted which volume, E) rror on previous volume, D) one, S) kip checks, or H) elp:

当完成了卷的检查, 就选择 d。

在备份成功地完成之后, 就会给出如何标记介质卷的指示。一定要对介质卷进行写保护。

#### 9.4 进行非例行备份

对于非正式的备份, sysadmsh 还包括一个“非例行”备份的选项。这就使得系统程序员能够执行一个文件系统的单独的、完整的备份, 而不必使用安排表。如果备份需求很简

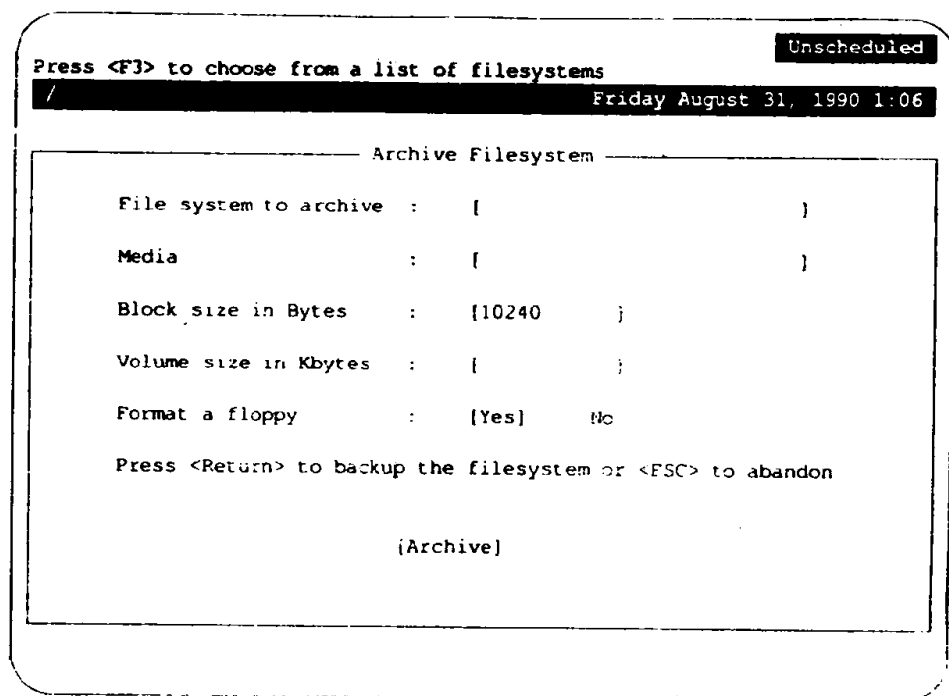
单，那么就可以按常规进行非例行备份。这种备份覆盖整个文件系统，而不仅是修改过的文件，并会需要大量的存储介质卷。如果打算依靠非例行备份，那么一定要至少一星期做一次。

要建立一个非例行备份，应遵循如下步骤：

1.调用 sysadmsh 并做如下选择：

Backups → Create → Unscheduled

2.如下的菜单会显示出来：



3.输入一个文件系统名或按 <F3> 得到一个文件系统表来选择一个要备份的文件系统。这个菜单会列出在文件 / etc / default / filesystems 中所能找到的所有文件系统。使用箭头键选择想要备份的文件系统并按回车键。

4.下面，输入介质设备名或按 <F3> 得到一张表来选择介质设备。块的大小会自动选择。

---

注：在选择介质设备号时一定要小心。例如，当想选择“软盘驱动器 0”（主软盘驱动器）时一定要选择“软盘驱动器 1”（第二软盘驱动器）。如果犯了这个错误，备份就会夭折，就必须从头开始了。

---

5.可以按需要格式化任意多的介质卷，只要每次插入一个介质卷，并在“Format floppy”一项选择“Yes”。（小型盒式磁带也可以被格式化，但这要花大量时间）。

6.把一个介质卷，磁带或磁盘，装入所选的驱动器中，并按回车键，系统就开始向驱动器中拷贝文件，在这些文件被拷贝的同时显示其文件名。如果一卷的空间用尽了，就会显示如下信息：

Reached end of medium on output

Insert volume 2 and press <RETURN> to continue or 'q' to exit.

7.把第一卷取出，插入新的一卷，并按回车键。程序就会继续把文件拷贝到新卷中。

重复这一步骤，直到程序显示信息：

DONE

如果使用的是软盘，那么可能在备份完成之前要多次重复最后这个步骤。应该在从驱动器中取出每一卷的同时对其进行标记。例如，把第一卷标记为“volume 1”、第二卷标记为“volume 2”，等等。

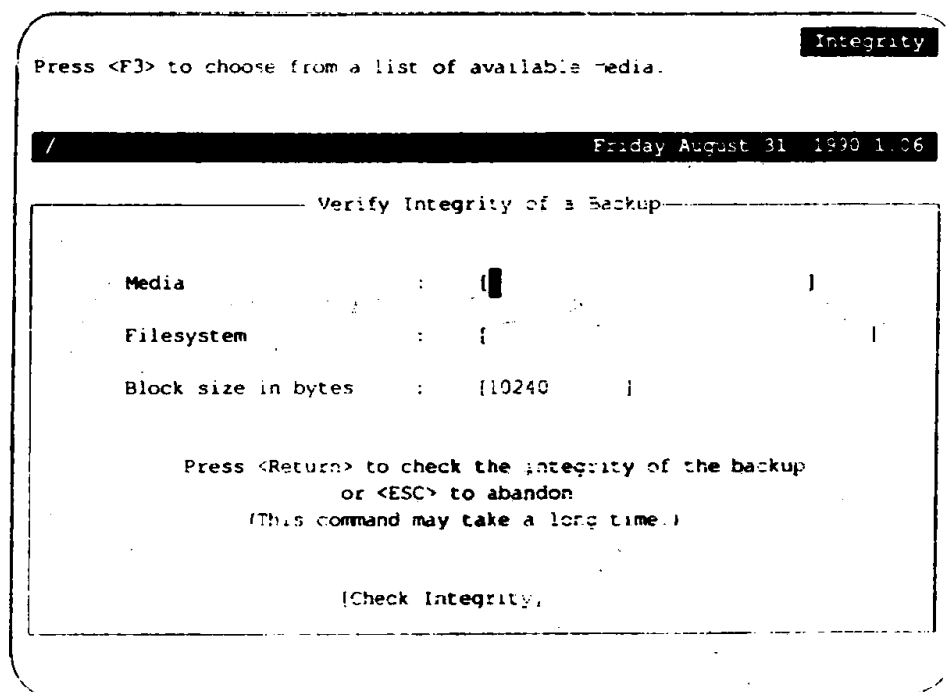
## 9.5 备份的核查

为了保证备份是正确无误，sysadmsh 备份菜单还包括一个 Integrity 选项。各卷都要被检查一下，看看它们是否可读，并且其内容都会列出来。

调用 sysadmsh 并做如下选择：

Backups→Integrity

此时会显示出如下表格：



输入或选取介质类型，并依次插入备份的每一卷。这是一个很长的过程；一个大的备份要花很长一段时间来进行核查。

## 9.6 获取备份表

可以通过在 sysadmsh 备份菜单中生成一张表来检查所备份的一系列文件。

要得到这张表，应遵循如下步骤：

1. 在 sysadmsh 中做如下选择：

Backup→view

2. 此时将显示如下的第一个表格。

3. 在第一个域按 <F3> 而得到一张介质设备表。块的大小 (block size) 会自动选取。

4. 程序提示插入第一个备份卷。装入第一卷，然后按回车键。

```

Press <F3> to choose from a list of available media
View
/ Friday August 31, 1990 1:06
----- View Contents of a Backup -----
Media : |
Block size in Bytes : 10240
Press <Return> to check the contents of the backup
or <ESC> to abandon
(This command may take a long time.)
[View]

```

5.当所有的备份卷都读了之后，屏幕上将有如下的显示：

```

<ESC> to exit, movement keys are active
View
cpio -itv /dev/rfd096ds15 -C 10240 08/31/90 11:03
----- cpio -----
100711 wadley 5678 Feb wadley/tell0
100711 wadley 6789 Feb wadley/tell1
100711 wadley 4112 Feb wadley/tell2
100711 wadley 9972 Feb wadley/tell3
100711 wadley 6689 Feb wadley/tell4
100711 wadley 1102 Feb wadley/tell5
100711 wadley 6602 Feb wadley/tell6
100711 wadley 5511 Feb wadley/tell7
100711 wadley 1111 Feb wadley/tell8
100711 wadley 3312 Feb wadley/tell9

```

### 9.7 从备份中恢复特定的文件或目录

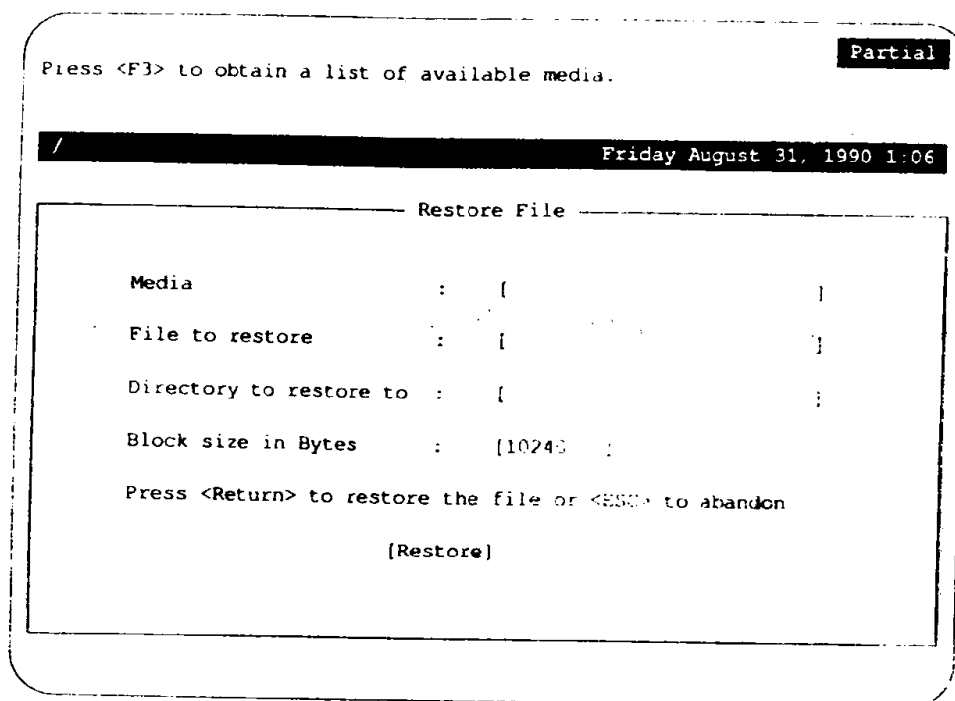
可以通过调用 sysadmsh 从文件系统备份卷中恢复特定的文件或子目录。管理员必须保存有包含要恢复文件最新版本的一组备份卷。如果要恢复的不是最近修改过的文件，那么就要使用最后一次的 0 级备份了。

要恢复一个文件，应遵循如下步骤：

1.调用 sysadmsh 并做如下选择：

Backups→Restore→Partial

2.会看到:



3.首先按<F3>从介质类型表中选取介质类型。当选中之后,就会弹出一个窗口来确认驱动器是否准备好:

Checking type of backup...

Make sure the media is in the drive and the drive is on line.

Press <Return> to continue or <ESC> to abandon

4.把备份集的第一卷装入驱动器,然后按回车键。当这一要求被满足之后,就回到“恢复文件(Restore File)”这个菜单下。下面就输入文件名,然后按回车键就到了“Directory”域,输入想把文件恢复到其中的目录。

---

注:重要的两点:

- 在指定路径名时,头一个斜线(/)一定要去掉。例如:如果正在恢复文件/bin/foo,必须这样指定它:

bin / foo

- 如果依然用其原来所在位置的路径名,那么所恢复的文件就会覆盖在那个同名文件的位置上。确认在备份卷中的文件正是所需要的版本是十分重要的。如果不能绝对肯定备份中包含有文件的更好版本,就应该先把它们恢复到一个临时的位置,例如/tmp,并使用diff(C)或cmp(C)把它们与磁盘上当前的文件进行比较。

---

5.到备份中去找所指定的文件,并且在文件被恢复到硬盘上的指定位置之后显示其文件名。在需要时还会得到更换介质卷的提示。如果获悉想要的文件都已经恢复了,那么可以使用<DEL>键退出恢复过程。(程序将继续找到备份的尾部)。

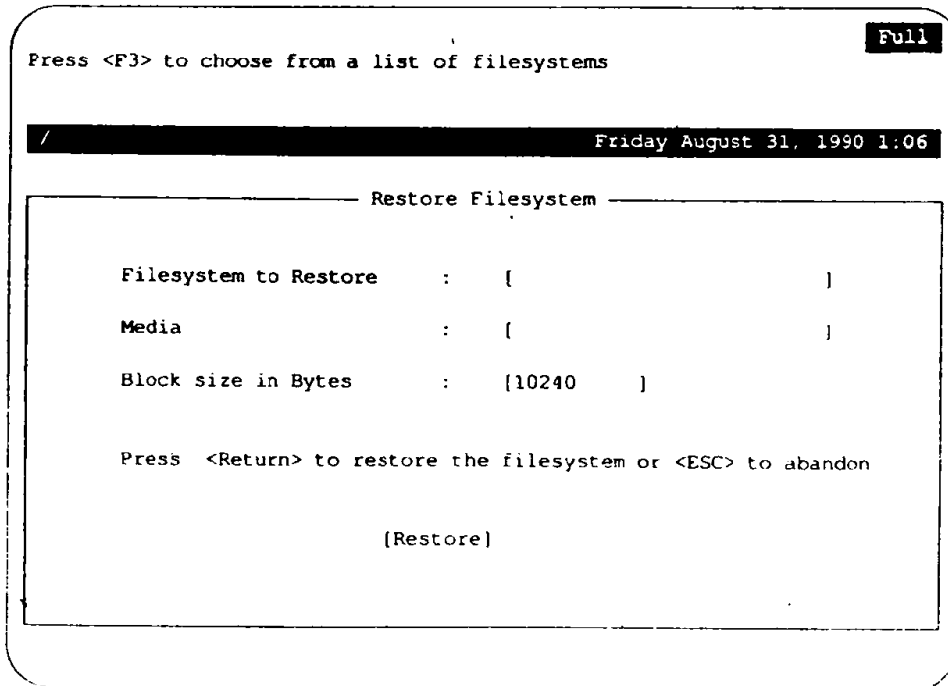
## 9.8 恢复完整的文件系统

请按如下步骤恢复文件系统的备份:

1. 插入第一卷, 并作如下 sysadmsh 选择:

Backups→Restore→Full

如下表格会显示出来:



2. 输入文件系统的名称, 或按 <F3> 得到一个点菜单。对于介质设备也是同样做法。要求对想要的这些作一次确认。

3. 在每个文件被恢复的同时, 其文件名也被显示到屏幕上。如果备份有多卷, 就会提示依次插入各卷:

Reached end of medium on input

Change to part n and press <RETURN> key.[q]

当恢复过程完成之后, 所恢复的块数会显示出来。

## 9.9 备份级的概念

保证数据安全的最直接并且最可靠的办法是一次对一个文件系统上的所有内容都进行备份。然而, 文件系统可能很大 (200 兆字节或更大), 并要花很长时间来备份。备份级 (或增值备份) 的概念正是针对这个问题提出来的。增值备份的总体概念是只对那些上一次备份之后修改过的文件进行备份。这可以明显地减小备份的大小和时间。请考虑如下方案:

- 每月 完全备份一次
- 每周 备份与上周不同的内容
- 每天 备份与前一天不同的内容

这表示在每月的月末, 对整个文件系统进行一次备份。每周, 要对自从上周以来所修改过的文件进行备份, 而每天要对这一天来所修改的文件进行备份。如果一个文件系统在某些方面受到了破坏, 那么只要把这次事故之前所作的最后一次完整 (月) 备份、最后一

次周备份和所有的日备份进行恢复就可以了。因此，总是能够从一系列备份中重新构建一个文件系统的。

这是一个很容易懂的方法，但使用增值备份级作为补充的方法却不很容易明白。

### 9.9.1 增值备份级的原理

为了使文件备份更加有效，备份实用工具使用了一系列等级，每级都建立在前一次较低等级的备份基础之上。

等级	所存文件
0	文件系统上的所有文件
1	上次 0 级备份之后所修改的文件
2	上次 1 级备份之后所修改的文件
3	上次 2 级备份之后所修改的文件

这些等级是为了把一个备份划分为一些可管理的单元。能够意识到每个备份级都是在前（低）一级备份基础上建立的这一点是很重要的。这意味着备份的顺序是没有意义的，而级数却是关键。

例如，让我们假设下面是为一个星期所做的备份：

日期	等级	所备份的文件
星期一	0	文件系统上所有的文件
星期二	2	星期一之后所修改的文件
星期三	1	星期一之后所修改的文件
星期四	3	星期二之后所修改的文件
星期五	2	星期三之后所修改的文件

这个例子是不符合逻辑的，但是说明了分级是怎么回事。请记住每个备份级都要保存前面低一级备份之后所修改的文件，而 0 级是最低级的。因此，星期五所做的 2 级备份要备份前面低一级即星期三的 1 级备份后所修改的所有文件。星期二 2 级备份只保存前一天之后的所修改的那些文件，因为前面较低等级的备份只有 0 级。如果除星期一以外的所有备份都是 2 级，那么每一次都要备份星期一 0 级备份之后所修改的所有文件。

### 9.9.2 缺省的和候补的安排表

系统原盘所提供的 schedule 文件对于中等使用规模（对于 200 至 400 兆字节硬盘的总存储量有 8 至 10 个用户）下的系统是最优的。给根文件系统用的缺省安排表与候补安排表中用于 /u 文件系统的类似。这样做是因为只有一个文件系统（根）的系统有活跃的用户帐号，并且应该每天都做备份。带有另一个用于用户帐号的文件系统（/u）的系统，只要每天对 /u 文件系统进行备份就可以了，而对于不太活跃的根本文件系统隔一天做一次备份。候补安排表请见图 9-8。



#	1 2 3 4 5	6 7 8 9 10	1 2 3 4 5	6 7 8 9 10
#Filesystem	M T W T F	M T W T F	M T W T F	M T W T F
/dev/rroot	0 × 3 × 3	2 × 3 × 3	1 × 3 × 3	2 × 3 × 3
/dev/ru	3 0 3 3 3	3 2 3 3 3	3 1 3 3 3	3 2 3 3 3

图 9-8 候补安排表

### ① /u 文件系统

文件系统 /dev/u 是一个用得很多的文件系统。有些等级的备份要每天做一次。这样做是为了增大安全性；如果某一个星期的一个或多个备份丢失或坏了，还可以有其余的备份来尽量减少数据的丢失。

按照缺省安排表，/dev/ru 的一个完全（0 级）备份应该在月初来做。（因为对于根文件系统的 0 级备份在星期一做，所以对于 /u 的 0 级备份应在星期二做）。星期三对 /dev/ru 所做的 3 级备份只保存自从 0 级备份之后所修改的文件。那么到了周末，所用的软盘或磁带总量就会比每天都做完全备份所用的要少得多。所用的时间也大大减少。如果需要把文件系统恢复为最后一次备份时的状态，那么就先恢复一次 0 级备份，然后再恢复从那以后所做的最新的较低级备份。

请注意在每个星期二，都要做一个较低级（0、1 或 2）备份来保存月初以来的每项内容，这就使得每一个在那之后所做的 3 级备份都是有关那一个星期的。这样，3 级备份就不会变得很大和多余。

### ② 根文件系统

根文件系统包含操作系统和其它系统文件。它改变得不很频繁，所以不必每天对它备份一次，除非用户帐号也放在那儿。每个星期一都要做一次较低级备份，并且每周还要做两次 3 级备份。与 /u 文件系统一样，3 级备份只限于保存那一周里所修改的内容。

## 9.9.3 如何用备份恢复一个文件系统

举一个例子，假设硬件发生了故障，而使得硬盘上的内容遭到了破坏。假设这发生在一个月的最后一个星期四，恰恰又在那天晚上进行备份之前。修好硬件故障并重新安装操作系统，但如何恢复备份呢？恢复每个备份级的最新备份，并按升序：

- 0 级（在本月的第一个星期二做的）
- 1 级（在第三个星期二做的）
- 2 级（在第四个星期二做的）
- 3 级（在星期三晚上做的）

不必恢复在第二个星期二做的 2 级备份，因为其后所做的 1 级备份包括了相同的文件。所丢失的信息只有在星期四那一天所修改的，恰在事故之前。这就是备份的主要原因；这样恢复应该是直接的并且所丢失的内容是最少的。

## 第十章 系统性能的调整

这种 UNIX 适用于很多种硬件配置并可作为很多应用程序的平台。操作系统的核心控制着许多资源，这些资源不断地被使用、释放与再循环。它们包括：

缓冲区：在内存中的缓冲存储器，其中保存有最近使用过的数据。（缓冲区通过暂存数据减少了读盘次数而提高了效率。）

表登记项：系统表中的一个空间，核心用它来跟踪当前的任务、资源和事件。

其它参数：这些是其它可定义的值，它们控制着特殊的资源（例如屏幕的数量以及信号量的数量）。

这些资源的使用是由某些限制参数，例如可调核心参数（tunable kernel parameters）来规定的。这些限制可以减少也可以增加，有时要以牺牲其它资源为代价。每一种资源或限制都由一个单独的核心参数来表示。确定如何最好地利用这些资源就被称作性能或核心的调整。这一章将介绍如何修改这些参数以适应系统的需要。另外，还包括了能够增强资源利用及系统性能的一般过程。

### 核心参数

核心参数是包含在 UNIX 系统核心的值，它们是操作系统的内核。每当这些可调参数被修改时，它们的新值就必须被重新连接（重新编译）到核心中，以使得这些新的限制能够起作用。configure (ADM) 这个实用程序就是用来修改核心参数值的。

当第一次建立 UNIX 系统时，性能的调整是一件值得注意的活动。当第一次启动系统时，系统将自动设置为最基本的配置，这种配置能够满足大多数工作环境的需要。但是，这种配置不能够考虑特殊应用需求。由于这个原因，系统的结构允许对其进行重新配置，针对特殊需要来增强其性能。

---

注：如果没有明显的必要，我们不主张对核心参数进行调整。

---

有如下几种理由来重新分配系统资源：

- 安装了另外的硬件存储器，这样就有更大的存储器资源要分配。
- 不断地显示错误信息，指出某种资源被用尽了，例如 i 节点或表登记项。
- 系统的反应总是很慢，这表明其它资源对于系统的高效运行制约太大（例如在只安装了很少一点内存时）。
- 资源的使用需要重新安排，以适应特殊应用的需求。

另外，判断哪些资源过于浪费或使用效率不高也是很重要的。当安装了另加的存储器而允许系统支持更多的用户时，某些可调参数一般要向上调整。然而，对于一台高效的个人计算机或专用处理器计算机来说，它可能不必增大核心可调参数。实际上，为了增加存储器以支持更多用户而对某些参数（NBUF、NCLIST、等等）进行调整都会降低系统的整体性能。因为这些参数增加了核心数据的空间需求，这样就减少了用户进程所能用的的存储器。简而言之，应根据计算机的使用目标和其性能状态来决定是否需要调整这些可调

参数。

特殊的应用经常需要对一些关键的系统资源进行重新配置以达到最佳性能。例如，有大型数据库的用户可能会需要同时封锁更多的文件。那些对于特殊特性（例如信息处理）没有需求的用户可能会感到，如果去掉那些特性将会获得在整体性能上的一定程度的提高。

## 10.1 利用 configure 重新分配核心资源

configure 是一个菜单驱动程序，它给出各个资源和提示以供修改。在修改核心参数之后，必须调用 link\_unix (ADM) 重新连接核心，把核心拷贝到根目录下，重新引导，并对新核心进行测试。

---

注：这一节只讲如何修改参数；还必须阅读本章的其余内容来弄明白为什么和什么时候必须进行修改。

---

要修改核心参数，应按如下步骤来做：

- 1.重新引导并进入单用户（维护）模式。
- 2.用 custom (ADM) 来判断一下用于连接的软件包 (LINK) 是否安装了。如果没有，用 custom (ADM) 来安装它。
- 3.在确定 Link 软件已安装了之后，输入如下命令：

```
cd / etc / conf / cf.d
```

```
./configure
```

△sysadmsh 用户可选择：

System → Configure → Kernel → Parameters

4.configure 的菜单显示出来：

- 1.Disks and Buffers
- 2.Character Buffers
- 3.Files, Inode, and Filesystem
- 4.Processes, Memory Management and Swapping
- 5.Clock
- 6.Multi Screens
- 7.Message Queues
- 8.Semaphores
- 9.Shared Data
- 10.System Name
- 11.Streams Data
- 12.Event Queues and Devices
- 13.Hardware Dependent Parameters
- 14.Remote file sharing Parameters

Select a parameter category to reconfigure by typing a number from 1 to 14, or type 'q' to quit:

参数被按类分组。要找到一个参数，请看本章末尾的“可调系统参数的介绍”。通过输入在某一类前面的数而选中这一类。在那一类中的所有资源显示出来，一个接一个，每一

个都带有当前值。对一个资源输入一个新值，或只按一下回车键来保留原来的值。当在这一类中的所有资源都显示出来之后，configure 就会回到分类菜单提示。选择另一类来进行重新配置或输入 q 退出 configure。

---

注：请注意管理员必须有与这个参数相关的软件包，这样才能使用与其相关的资源。例如，必须安装了 STREAMS 软件包，STREAMS 参数才能起作用。

---

5. 在完成修改参数之后，必须把它们连接为新的核心。输入如下命令：

```
./link_unix
```

△sysadmsh 用户选择：

System ▶ Configure ▶ Kernel ▶ Rebuild

这就把每个核心模块组装成一个新核心了，现在它一定安装好了。连接过程需要花一些时间。

6. 用如下命令引导新的核心：

```
/etc/shutdown
```

△sysadmsh 用户选择：

System ▶ Terminate

这时就出现一个引导提示符。当按回车键重新引导系统后，新核心就被装入并运行起来。

如果新核心中存在有问题，就在 Boot: 提示符下输入 unix.old 来重新引导 / unix.old。

### configure 命令行的使用

configure 也有适合于应用程序开发人员所使用的命令行方式。例如，有一个数据库开发人员，他需要同时封锁 70 个文件而不是 50 个文件，那么他就可以写一个 shell 程序来进行重新配置。要使用命令行方式来查看可配置资源的当前值，可输入：

```
./configure -y RESOURCE
```

这里 RESOURCE 是可调参数的名字（要大写）。要想以命令行来修改某种资源的值，可输入：

```
./configure RESOURCE = value
```

这种人机接口方式要除那种交互方式以外的另一种方式，各种资源都可以用这种方式进行配置。

下面将介绍配置核心资源的方案。

configure 给核心参数值强行规定了某些限制，这是因为当某些值过大或过小，系统性能就会因偶然的错误而遭受破坏。如果感到必须替换某个参数值，那么就要按 configure (ADM) 命令页上所介绍的使用 -O 选项。这个替换选项只能用于用命令行方式指定一个参数，在菜单方式下无法使用它。

## 10.2 修改系统配置

有时会出现这种情况，操作系统警告管理员系统的某些限制被超过了。这些信息都显示在控制台上。其中有些只是警告性的，而另一些还会造成系统恐慌，在这种情况下还会显示出另外一些诊断信息，然后系统“挂起”，需要重新引导。如果一种核心错误信息只出

现一次，或者甚至是两次，而不是同一个信息在系统对话区中不断出现，那么核心就不必要重新配置。

如果遇到列于表 10.1 中的某个信息，那么请参阅本章后面“可调系统参数的说明”一节中相应参数的说明。如果需要调整一个资源，那么要先试着少量增加。如果某个问题持续出现，那么就相对原值增加 100% 或更多。如果这个问题还没有解决，那么就要做更细致的检查来找到产出错误的确切程序和指令序列。

表 10-1 错误信息和相关可调参数

核心错误信息	参数
iget-inode table overflow	NINODE
Timeout table overflow	NCALL
File table overflow	NFILE
mfree map overflow (*)	SPTMAP
Region table overflow	NREGION
Configured value of NOFLLES n (*) is less than minimum (greater than maximum)	NOFILES
stropen: out of streams	NSTREAM
swapped-too few free pages	MINASMEM
stropen: out of queues	NQUEUE

(\*) 这个 n 值是指错误信息中所显示的真实值。

### 10.3 性能的重新配置

系统应该这样配置，绝大多数核心资源应即能适合于最普通的工作（例如磁盘的读写），同时又不忽略比较特殊的需要（例如进程间的通信）。使用下面所介绍的知识可以对这种平衡进行调整以符合特定的需求。

#### 10.3.1 核心调整的折衷

系统可以支持各资源的不同用途。一个支持几个用户编辑几个小文件的系统所消耗的资源比一个单用户系统运行一个大型数据库用得还要多。为了使核心运行得更加有效，就要对参数进行调整。这通常都会增大核心的数据结构。虽然这可能会使某一方面的系统操作效率提高，但是核心要花更长的时间对这个结构扫描一遍。这也意味着不必要地增大某些参数会降低系统速度。例如，增大参数 NPROC 可以使系统维护一个更大的活动进程表 (PROCess table)。但这对核心调度程序会有一个反作用，因为它必须重复地对这个更大的表进行扫描来检查下面该运行哪个进程了。另外，由于表增大时核心数据所需的空间也增多了，这样给用户进程用的内存空间就减少了，这也会降低系统的整体性能。

### 10.3.2 一般的资源需求

在系统的使用中通常需要调整某些参数以符合特定的环境要求。一个常见的需求是能够建立很大的文件。可以这样来做，首先成为超级用户，然后修改正在运行的这个特定的 shell 进程的 ulimit 参数。另一个解决办法是修改系统中所有用户的 ULIMIT。ULIMIT 参数和其它通常遇到的参数都归纳在表 10.2 中。在修改这些参数之前还请先看一下本章“可调系统参数的说明”一节中各参数的详细说明。

### 10.3.3 提高磁盘的利用率

表 10-2 特殊情况下的调整需求

需求	参数
安装了另外的存储器而提高系统性能	UBUF,NHBUF (请看“高速缓存的估算”)
其它与性能有关的系统参数	NAUTOUP,MAXSLICE,BDFLUSHR,AGEINTERVAL
安装了额外的存储器而增大了系统限制指标 (为了支持更多的用户和减少系统在重负载时发生问题的机会等)	NCALL,NINODE,NSINODE,NFILE,NPROC,NREGIONS,NCLLST (再看一下信息, 信号量及共享存储器的参数)
需要建立大文件的用户	ULIMIT
要打开更多文件的用户	NOFILES
要运行更多进程的用户	MAXUP
其它可能遇到的系统限制	SHLBMAX,FLCKREC,SPTMAP,NUMXT,NUMSXT,PRFMAX (再看看 STREAMS 参数)

磁盘的输入输出在系统性能中是一个瓶颈。为了得到更高的效率，对于磁盘子系统的调整可作三点考虑：

- 选取合适数量的缓冲区
- 增加更多的存储器
- 恰当地组织文件系统以减少磁盘的活动。

#### 1) 高速缓存的估算

系统把所有的存储器划分为两个结构：磁盘高速缓存和页缓存。高速缓存是一系列缓冲区，其中保存有最近使用的数据，以备再次需要。如果读写操作只使用高速缓存而不必用磁盘就能满足，那么系统的性能就会提高，这是因为存储器的操作要比磁盘的操作快得多。页缓存在概念上与高速缓存类似；磁盘缓冲区中包含的是数据而页中包含的是程序。

最理想的是，给在系统上的各项工作安装足够的存储器，但如果存储器是一种受限的资源，那么页缓存与高速缓存需求之间的平衡就会受到冲击。如果页缓存相对于系统能够承受的负担显得太小，那么系统就会不断地把各页换出换入以跟上当前的进程。如果页缓存只稍微小一点儿，那么其效果看起来就不是频繁地交换，而是缓存的性能降低。这是因为对于有效地处理当前的进程，这些页是足够的，但没有空闲的空间来保存最近使用过的页以备可能的访问。

NBUF 参数规定了在系统高速缓存中缓冲区的数目。NHBUF 规定了高速缓存中

hash 队列的数目。为了不必在整个缓冲池中进行搜索，高速缓存被分成一系列队列。缓冲区越多，在缓冲区中找到所要数据的机会就越多，系统也就不必去费时地读盘了。sar-b 和 sar-w 命令可以报告系统缓冲区的工作效率。（请看本章中“对低效系统进行诊断”。）NHBUF 的值必须是 2 的幂；另外，NBUF 除以 NHBUF 应该约等于 4。

如果在系统运行了一天左右之后想修改缓冲区数目，那么要先对系统性能进行检查，特别是对过多的交换活动。如果发现有这种活动，那么就要减少缓冲区的数目。使用 sar，可以判断在一个给定的时间区间中有多少程序被换入换出。如果频繁的交换很明显，就应该减少缓冲区的数目，这样可以通过给出更多的可用存储空间而增大页缓存。我们还建议增加尽量多的 RAM，这样交换就会减少，性能也会提高。

### 2) 增加存储器 (RAM)

过去 UNIX 系统管理员在小型机或超级小型机上安装了额外的存储器后就要增大所有的系统可调参数。这就可以使系统支持更多的用户而不会在系统重负载时遇到问题。对于单用户的 PC 环境，却一点儿也不必增大核心可调参数。并且，由于前面所提到的原因，即使在安装了更多的存储器时，依然把系统的限制保持为缺省值也会获得最佳性能。

正如表 10.3 所示，所规定的缺省值是针对一个 2 到 3 兆字节 RAM 的系统的，这是所推荐的最小内存量。如果系统工作于多用户配置，即 5 个或更多用户，那么就要增加更多的存储器并增大某几个参数以确保系统限制有充分的余量，并提高高速缓存的命中率。对于 4 至 6 和 6 至 15 兆字节的存储器配置，参数值在表 10.3 中也给出了。在进行这些修改之前，应该先建立一个性能的基线，然后再修改系统参数，并再次对系统性能进行判断。这是一个最好的判断参数的修改使系统性能是提高了还是降低了的方法。

表 10-3 关键的参数和存储器

参数	存储器的大小		
	2-3Mbytes	4-6Mbytes	6-15Mbytes
NHBUF	64	128	256
MAXBUF	250	400	600
NCALL	30	50	70
NINODE	150	300	400
NMPBUF	8	12	16
NFILE	150	300	400
NREGION	210	250	300
NCLIST	120	170	200
NPROC	100	150	200

### 3) 对文件系统的再组织

在文件系统的使用过程中，其各成员文件的各块就会逐渐分散到各磁盘上或磁盘的各处，这样 I/O 效率就会变得越来越低。这种分散会使得文件块的顺序变得很差，目录结构也不好。目录还会变大并增加搜索时间。这些问题增加了文件的访问次数，这在“文件

系统的管理”一章中的“维护高效的文件系统组织”一节中进行了讨论。

#### 10.4 规定有效的系统使用模式

在调整了核心和系统活动以及重新组织文件系统之后，提高系统性能的下一一些就是做一些“家务管理”工作。人的系统管理职责应该是对如下情况进行检查：

- 不太重要的（甚至是不必要的）工作对比较重要的工作的干扰。
- 用户定义的特性的效率，例如.Profile 和 \$PATH。

##### 10.4.1 用 PS 检查进程活动

PS (C) 命令包含活动进程的信息。这个命令给出了系统中正在执行的各进程，这对于确定在系统中装载了哪些进程是很有用的。情况基本上是随输出显示的时间而变化的；然而，管理员应该感兴趣的是 TIME（进程所用的 CPU 时间的分和秒）以及 STIME（进程第一次开始的时间）。图 10-1 包含了以 PS-afe 命令所得到的输出样本。

UID	PID	PPID	C	STIME	TTY	TIME	COMMAND
root	22247	21299	0	16: 54: 41	T1	0: 01	msscreen -n 2
johnson	22246	22079	0	16: 52: 53	T0	0: 04	vi file 2
root	22285	22247	0	16: 56: 04	T1	0: 06	msscreen -n 2
root	22284	22247	0	16: 56: 04	T1	0: 00	msscreen n 2
markham	22274	22271	0	16: 55: 09	P1	1: 05	rlogin colossus
markham	22271	22243	0	16: 55: 07	P1	1: 05	rlogin colossus
forbin	22304	21003	0	17: 02: 51	003	0: 00	/usr/bin/nail kuprin
fisher	22298	18505	2	17: 02: 29	011	3: 28	nethack
root	22305	1327	10	17: 02: 57	T2	0: 00	ps -ate

图 10-1 从 ps -ate 得到的输出样本

“解决系统问题”一章中介绍了如何使用 PS 命令来执行类似寻找“脱疆的”进程（在被监视期间使用的资源越来越多的进程）的工作。还可以用 PS 找到用了很长时间执行的进程；在非上机时间，可以考虑用 cron (C) 来执行这一任务。

##### 10.4.2 检查用户的 \$PATH 变量

\$PATH 环境变量列出了在执行一个命令时可以搜索的所有目录路径名。\$PATH 在用户的 .login.cshrc 或 .profile 文件中定义。在显示“not found”之前，系统必须搜索 \$PATH 中的所有目录。这些搜索既需要处理器时间又需要磁盘时间。如果存在磁盘或处理器瓶颈，那么修改这个变量会有助于性能的提高。

在用户 \$PATH 变量中所应检查的一些内容有：

- 路径的效率

\$PATH 从左向右读，所以最可能找到命令的地方应该放在路径头上（/bin 和 /usr/bin）。另外，一个目录一定不要在 \$PATH 中出现两次。



- 路径的长度

一般来说，\$PATH 中的项应尽可能地少。

- 大型目录的查寻

如果可能的话，应避免搜索大型目录。所以应把大型目录放在 \$PATH 的尾部。

## 10.5 对低效系统进行诊断

在操作系统中有一系列对性能进行评估的工具。这些工具可以供系统管理员查找问题的所在。在这一节里所介绍的对内部活动进行性能检查的工具具有：

sar (系统活动报告程序) 对系统状态进行采样并对各种全局活动提供报告。

timex 对在一个命令或程序的执行过程中的全局和每个进程的活动提出报告。

### 10.5.1 sar 命令

内部活动是由包含在核心中的一定数量的计数器来估量的。每进行一次操作，相应的计数器就要加 1。sar (ADM) 就是根据从这些计数器搜集来的原始数据而建立报告的。sar 报告可被用于诊断系统问题。所要监视的两个最严格的区域是内存和 CPU (中央处理器) 的使用。sar 所监视的活动在下面各小节中进行讨论，包括对一个 sar 输出样本的分析。sar 既可以搜集“活着的”系统活动数据又可以从 sadc 所产生的数据文件中提取有用的内容。在缺省设置下，装在 /usr/spool/cron/crontabs/sys 中的 crontab 登记项为：

```
0 * * * 0-6 /usr/lib/sa/sa1
```

```
20, 40 8-17 * * 1-5 /usr/lib/sa/sa1
```

```
5 18 * * 1-5 /usr/lib/sa/sa2-s 8: 00-e 18: 01-i 1200-A
```

sa1 登记项在工作时间里每 20 分钟产生一个记录而在非工作时间里每小时产生一个。sa2 登记项在工作时间中每小时向同一个文件中写入一个日平均报告。管理员可以按需要修改这些登记项。输出文件为二进制形式 (为了紧凑)，并存于 /usr/adm/sa 中。文件名的形式为 sadd，其中 dd 是当月的日期 (the day of the month)。(有关修改缺省 crontab 文件的更多的内容请看 sar (ADM) 页)

sar 命令的基本型式如下：

```
sar option t
```

其中，option 是一个选项，这将在后面介绍；t 是以秒为单位的采样间隔，它应该是 5 或更大一些。如果想看实时的数据，就必须给出这个采样间隔，否则就只显示在那一天里已经搜集的数据。在 /usr/adm/sa 目录中，管理员可以检查所有存在的记录。例如，要获得一个在当月 23 日所搜集的 (-A 选项) 所有 sar 数据的积累报告，就可以输入如下命令：

```
cd /usr/adm/sa
```

```
sar -A -f sa23|more
```

在本节中的例子没有给出基准点；这些例子只是用来说明如何使用这些输出结果。在调整系统时，建议使用基准点 (一个用于测算系统性能的程序) 并使系统在使用过程中处于正常负载状况。

### 10.5.2 缓冲区的活动: sar-b

-b 选项报告如下的缓冲区活动:

bread / s	每秒钟从磁盘 (或其它块设备) 读到系统缓冲区中的平均物理块数。
lread / s	每秒钟从系统缓冲区所读的平均逻辑块数。
%rcache	高速缓存的逻辑读比率 (100%减去 bread / s 与 lread / s 的比值)。
bwrit / s	每秒钟从系统缓冲区写入磁盘 (或其它块设备) 的物理块的平均数。
lwrit / s	每秒钟写入系统缓冲区中的逻辑块的平均数。
%wcache	高速缓存逻辑写的比率 (100%减去 bwrit / s 与 lwrit / s 的比值)。
pread / s	每秒钟物理读请求的平均数。
pwrit / s	每秒钟物理写请求的平均数。

最应感兴趣的是高速缓冲的命中比即 %rcache 和 %wcache, 它们是由于测算系统缓冲区的效率的。如果 %rcache 降到低于 90 或 %wcache 降到低于 65, 那么或许通过增加缓冲区数能够提高性能。

下面是 sar-b 输出的一个例子:

```
unix unix 3.2 2 386i 02 / 18 / 89
16: 32: 57 bread / s lread / s %rcache bwrit / s lwrit / s %wcache pread / s pwrit / s
16: 33: 07 3 39 93 1 16 91 0 0
16: 33: 17 4 40 90 2 16 87 0 0
16: 33: 27 4 41 90 3 7 64 0 0
Average 4 40 91 2 13 84 0 0
```

这个例子表明缓冲区没有产生瓶颈, 因为所有的数据都在可接受范围之内。

### 10.5.3 名缓存 (name cache) 的活动: sar-n

sar-n 选项对名缓存的统计提出报告。

c__hits	缓存命中数
cmisses	缓存未命中数
hit%	命中与未命中比值的百分比形式

下面是 sar-n 的一个例子

```
unix unix 3.2 2 386i 02 / 18 / 89
11: 26: 05 c__hits cmisses (hit %)
11: 26: 10 9 0 (100%)
```

名缓存在“调整磁盘效率”中有详细说明。

### 10.5.4 进程通过量: sar-q

sar-q 选项报告在队列被占用时间的平均队列长度及占用时间的百分比。

runq-sz 内存中的进程运行队列; 一般来说, 这个值应该小于 2。持续的高值意味着 CPU 处于负载上限。

%runocc	运行队列占用时间的百分比; 这个值越大越好。
swpq-sz	被换出的进程替换队列; 这个值越小越好。

%swpocc 替换队列占用时间的百分比；这个值越小越好。

下面是 sar-q 的一个例子：

```
unix unix 3.2.2 386i 02/18/89
11: 00: 56  rung-sz %runocc swpq-sz %swpocc
11: 01: 07      1.7   98      1.5     36
11: 01: 17      1.0   63      1.0     31
11: 01: 27      1.0   58      1.0     49
Average      1.3   74      1.2     39
```

在这个例子中，处理器的利用率（%runocc）变化于 58% 到 98% 之间，而替换队列所占时间比（%swpocc）却为 31% 至 49%。这说明内存不是系统中的主要瓶颈，但更多的内存有助于减少页替换活动。

### 10.5.5 CPU 的利用：sar-u

cpu 的使用情况是由 sar-u 给出的。在任何一个给定的时刻，处理器或者是忙的或者是闲的。忙时，处理器或者处于用户模式或者处于系统模式。闲时，处理器或者在等待输入/输出的完成或者没事要做。sar 的 -u 选项列出了系统处于以下各种情况所用时间的百分比：系统模式（%sys），用户模式（%user），等待输入输出的完成（%wio），以及空闲时间（%idle）。

在一般的分时使用中，%sys 和 %usr 的值基本相同。在特殊应用中，在没有什么不正常的情况下，其中的某一个可能要比另外一个大一些。过高的 %wio 一般意味着磁盘操作是个瓶颈。过高的 %idle，再加上过长的反应时，可能意味着内存紧张；内存的等待时间对 %idle 的增长有作用。

从“idle”（空闲百分比）这一列可以对系统性能有一定的洞察。这个数据正常的是在 40% 到 100% 之间，即使有很多活动的用户。当这个数据持续低于 30% 时，资源的主要竞争就一点儿也不涉及内存了，而关键的资源就是处理器的速度了。（运行 PS (C) 命令察看一下是否有“脱疆的”进程在偷用每一个空闲的 CPU 周期。）

如果系统带有大量用户，并且所使用的是非常普通的“笨”终端卡，那么把它换为智能串行卡可能会有帮助。智能卡能够承担一些 CPU 的负担而不会给 CPU 增加工作量。

另外，应该检查一下 /usr/spool/crontab，看看在高峰时间是否排队等候了很多作业，而这些作业最好是在系统空闲时运行的。使用 PS 命令来判断一下是什么进程使系统的负载这样重。应鼓励用户在非高峰时间运行大型的、没有交互的命令（例如 nroff (CT) 或 troff (CT)）。也可以使用 nice (C) 或 batch (C) 命令以低优先级来运行这些命令。下面是 sar-u 的一个例子：

```
unix unix 3.2 2 386i 02/18/89
09: 20: 08      %usr   %sys   %wio   %idle
09: 40: 12      6      7      2     86
10: 00: 03      7      9      3     80
10: 20: 07     14     16     10     61
Average      9     11      5     76
```

### 10.5.6 系统表: sar-v

-v 选项报告进程、i 节点、文件、共享存储器记录、以及共享存储器文件表的状态。从这个报告中能够知道系统表什么时候需要修改。

proc-sz 核心中现在正被使用 / 分配的进程表登记项的数目。  
inod-sz 核心中现在正被使用 / 分配的 i 节点表登记项的数目。  
file-sz 核心中现在正被使用 / 分配的文件表登记项的数目。  
ov 产生溢出的次数。(上面三项中的各项都有一列)。  
lock-sz 核心中现在正被使用 / 分配的共享内存记录表登记项的数目。  
fhdr-sz 不再可用的。

这些值是以级 / 表的大小的形式给出的, 表示所用的表登记项的当前数目和表的大小。下面是 sar-v 的一个例子:

```
unix unix 3.2 2 386i 02 / 18 / 89
17: 36: 05 proc-sz ov inod-sz ov file-sz ov lock-sz fhdr-sz
17: 36: 35 17 / 40 0 39 / 80 0 29 / 80 0 0 / 50 0 / 0
17: 37: 05 19 / 40 0 46 / 80 0 35 / 80 0 0 / 50 0 / 0
17: 37: 35 18 / 40 0 43 / 80 0 34 / 80 0 0 / 50 0 / 0
```

这个例子表示所有的表都是足够大而没有溢出。如果这些是曾记录的最高值, 那么各表的大小就可以减小以节省主要的内存空间。

### 10.5.7 替换活动: sar-w

-w 选项对替换和切换活动进行报告。下面是一些目标值和观察记录。

swpin / s 每秒钟向内存中进行传输的次数。  
bswin / s 每秒钟换入 (包括原来装入的一些程序) 的 512 字节块的块数。  
swpot / s 每秒钟从内存向磁盘交换区传送数据的次数。如果大于 1, 内存可能需要增大或者缓冲区减小。  
bswot / s 每秒钟换出的块数。  
pswch / s 每秒钟进程切换的次数。这在一个繁忙的 4 到 6 个用户的系统上应为 30 至 50。

下面是 sar-w 的一个例子:

```
unix unix 3.2 2 386i 02 / 18 / 89
19: 53: 44 swpin / s bswin / s swpot / s bswot / s pswch / s
19: 53: 58 0.0 0.0 0.0 0.0 37
19: 54: 14 0.0 0.0 0.0 0.0 39
19: 54: 24 0.0 0.0 0.0 0.0 39
Average 0.0 0.0 0.0 0.0 38
```

这个例子表示对于当前的活动用户有足够的内存, 因为没发生交换。

### 10.5.8 检查每个命令的系统活动: timex

timex 命令对一个命令进行计时并对在命令执行期间所发生的系统活动提出报告。如

果没有其它的程序在运行，那么 timex 能够报告在这个命令执行期间用到了哪些资源。可以用这个办法收集各应用程序消耗系统资源的情况并用来对负载过重的资源进行调整。timex 可以这样使用：

```
timex -s program
```

用户的应用程序可以正常操作。当结束并退出时，timex 的结果就显示在屏幕上。这样就得到程序所用资源的清晰描述了。

## 10.6 调整磁盘效率

本系统还包括两个可以提高系统效率的方案：

**名缓存**——最近使用的文件的名称被存在一个高速缓存中。这可以减少系统把一个文件名与其内容联系起来所花费的时间。（对可执行文件和数据文件同样对待。）如果文件被多次访问，这就会提高系统的效率。

**搜集分散的 I/O**——这个方案是把一组磁盘 I/O 请求搜集到一起而不是单独执行。这个过程能提高磁盘的整体效率。

这一节将介绍如何调整与这些方案有关的参数，以提高硬磁盘的 I/O 性能。

### 10.6.1 名缓存

名缓存涉及三个可调参数：

S5CACHEENTS	名缓存中文件名的数目
S5HASHQS	名缓存的 hash 队列数
S5OFBIAS	在高速缓存中保存已打开文件的文件名的倾向性

对这三个参数的设置应该能提供 95% 的高速缓冲命中率（这表示对一个给定文件的请求可以通过在名缓存中查找而不必到磁盘或高速缓存中去查找就能得到满足的可能性占 95%）。缺省值应满足大多数多用户任务，例如一组用户运行一小组应用程序。它们还应满足偶然的单用户任务，例如对一个大型原代码的编译。如果系统只获得了小于 90% 的高速缓存命中率，那么这些高速缓冲的参数就应该增大。（sar-n 给出了名缓存命中率的数据。请看本章中“对系统低效进行诊断”。）

为各工作小组使用的基本方案应是，使 S5CACHEENTS 很大（大约等于 NINODE 的 3 倍），使 S5HASHQS 大约为 S5CACHEENTS 的 1/4，而不管 S5OFBIAS，除非有关的几个应用软件包由一些要打开相同的几个文件的部分组成。例如，C 编译器会产生一些临时文件，这些文件由编译器的一遍来写而又由另一遍来读。在一个编译为主要活动的环境里，增大 S5OFBIAS 可能会有好处。

S5OFBIAS 的值为 0 表示所打开文件的文件名没有特殊的高速缓存优先级，并很可能与其它被缓存的项目一样从高速缓存中丢掉。如果 S5OFBIAS 等于 S5CACHEENTS，那么所打开的文件几乎都把其文件名保留在高速缓存中。S5OFBIAS 不应该大于 S5CACHEENTS / 10，除非 S5CACHEENTS 非常大（至少比 NINODE 大几倍）。如果 S5CACHEENTS 小于 NINODE 的两倍而 S5OFBIAS 大于 S5CACHEENTS / 10，那么这种组合会使得文件名查找的性能很差，甚至可能比在这里

一点儿名缓存也没有还要差。

### 10.6.2 搜集分散的 I/O

快速文件系统 Acer 使用了一个可调的参数 NMPBUF，它是控制缓冲区数目的，这种缓冲区用以在把真正的内容传送到用户空间或高速缓存之前先对大量的磁盘请求进行搜集。这种 NMPBUF 缓冲区在内存的消耗方面是很大的（每个约占 16K）。另一个参数，NMPHEADBUFs，可被用来分配独立簇缓冲区头，它们基本上是没有相应的 16K 缓冲区的 NMPBUF 缓冲区头。这些缓冲区头所消耗空间相对少一些，并能够对于某些配置提高磁盘性能。

配置高速文件系统缓冲区是对内存的使用与速度的一种折衷办法。下面推荐的方法可以降低对内存的需求而同时仍能够保持完整的性能。如果在系统上内存的大小不是主要问题，那么把 NMPBUF 设置为 16，把 NMPHEADBUF 设置为 NBUF / 3，这对大多数配置应该是足够的。NBUF 的值是在引导时确定的，并在控制台上作为“i/o bufs”显示出来。（这个内容也可以通过查看文件 /usr/adm/messages 而得到。）引导时的显示如下：

kernel: drivers=18K, 4 screens=68K, 600 i/o bufs=600k, msg bufs=8k

#### ESDI 磁盘和其它支持 1:1 隔道存储的磁盘

如果系统所使用的是 1:1 隔道存储的磁盘，那么就把 NMPBUF 设置为 3，并把 NMPHEADBUF 设置为磁盘缓冲区数目 (NBUF) 的 1/3 左右。

#### SCSI 磁盘和其它不能有效地使用 1:1 隔道存储的磁盘。

如果系统所使用的是一个 SCSI 磁盘，或是不能有效地使用 1:1 隔道存储的 MFM/RLL 磁盘，那么很小的一个 NMPHEADBUF 就应该够了（例如 16），这节省了一点儿内存。使 NMPBUF 为 0，这个值在引导时就会自动配置。

## 10.7 可调系统参数的说明

下面各小节提供了一个在文件 /etc/conf/cf.d/mtune 中定义的系统可调参数的分类。参数分类如下：

- 磁盘与缓冲区
- 字符缓冲区
- 文件、i 节点和文件系统
- 进程、内存管理和替换
- 时钟
- 多屏幕
- 信息队列
- 信号量
- 共享数据
- 系统名
- 流式数据
- 事件队列和设备

• 依赖于硬件的参数

远程文件共享参数 (Remote File Sharing parameters) 可由 configure 列出, 它们的值也可以改变, 但不支持 RFS。

---

注: 在各分类中列出了参数, 其顺序是按照 configure 所显示的顺序而不是字母顺序。

---

### 10.7.1 磁盘和缓冲区

NDISK	是连到系统上的磁盘驱动器的数目。它在引导时设置。
NBUF	在引导时所分配的1K系统缓冲区的数目。这些缓冲区在一个数据高速缓存中。这个数据高速缓存是一个含有磁盘文件信息的内存阵列。高速缓存的命中率随缓冲区的增加而提高。高速缓存的命中减少了对磁盘的访问, 并因此提高了系统的整体性能。这个参数值一般在 100 到 600 范围内。每个缓冲区包含 1076 个字节。hash 队列的数目 (NHBUF) 应该随系统缓冲区的增加而增加, 这样才能获得最优性能。
NPBUF	指定要分配多少个物理 I/O 缓冲区。每个读写活动都需要一个 I/O 缓冲区。每个入口包含 52 个字节。这个参数的缺省值是 20。
NHBUF	指定为 1K 缓冲区分配多少个 hash 队列。这是为了搜索给定设备号和块号的缓冲区的, 这样就不必在整个缓冲区队列中进行线性搜索了。这个值必须是 2 的幂。每个入口包含 12 个字节。NHBUF 的值必须这样选, NBUF 的值除以 NHBUF 的值约等于 4。(在文件 /usr/adm/messages 中查看 NBUF 的值, 它在引导时确定并显示。)
CTBUFSIZE	是以 Kbyte 为单位的磁带 (QIC-11、QIC-24) 缓冲区的大小。它的值应为 32 到 256。它是在初始化时分配的静态缓冲区的大小。下面是与相应环境对应的合理的值: 32K 勉强的最小值: 对数据流是不够的。 64K 允许数据流 (有益于内存小的系统) 或使用小型磁带 (性能并不关键) 的最小值。 96K 比缺省低一级的配置, 如果缺省值使用的内存太多, 就减于此。 128K 缺省值: 好的性能折衷配置值。 192K 比缺省值高一级的配置, 如果缺省值所提供的性能太差, 就增为这个值。 256K 最大值。
MAXBUF	允许高速缓冲中可有的缓冲区的最大数目。这是核心中所说明的缓冲区头的数目。如果所需的缓冲区数小于这个数值, 那么

在引导时就会依据内存的大小来自动进行配置。如果 NBUF 不为 0, 那么就会准确地配置 NBUF 个缓冲区, 而 MAXBUF 就没有必要大于 NBUF。如果 NBUF 为 0, 核心就会自动最多配置 MAXBUF 个缓冲区。

- OMAABLEBUF 每次传输大于16MB的DMA请求所需的传输缓冲区的数目。它一定是一个 4 到 128 的值。缺省值为 16。
- NAUTOUP 为文件系统的自动更新指定以秒为单位的缓冲区寿命。当一个系统缓冲区中的内容已经在内存中驻留了由 NAUTOUP 参数所指定的这么长时间时, 它就会被写入硬盘。如果指定一个较小的值, 那么系统的可靠性就会因为较频繁地把缓冲区中的内容写入磁盘而提高, 但系统性能会降低。指定一个较大的值则会提高系统性能, 但要以降低系统安全性为代价。这个参数控制着 bdflush 守护进程的活动。
- BDFLUSHR 为对把文件系统缓冲区中的内容写到磁盘上的需要进行检查指定以秒为单位的频率。范围是 1 至 300。缺省值为 30 秒。这个参数控制着 bdflush 守护进程的活动。
- PUTBUFSZ 指定环形缓冲区 putbuf 的大小, putbuf 中包含由操作系统写到控制台上的最后几个 PUTBUFSZ 字符。putbuf 的内容可使用 crash (ADM) 看到。
- PIOMAP 决定由核心可编程 I/O (PIO) 分解例程使用的映射入口阵列的大小。这个例程可允许设备驱动程序通过把大的数据块分解为小的数据单位而在中断级上完成对大型数据块的可编程 I/O 操作。用户不应该修改这个参数。
- PIOMAXSZ 可编程 I/O 一次所使用页面的最大数目。用户不应该修改这个参数。
- DO387CR3 当安装了一个 80387 浮点协处理器时, 用以控制对控制寄存器 3 (CR3) 高位字节的设置。
- NUMTRW 是在核心数据空间分配的“传输库接口 (TLI)”这个数据读/写结构的数目。用户不应该修改这个参数。

### 10.7.2 字符缓冲区

下面的参数控制不同数据结构的大小以及对底层系统设备驱动程序的其它限制。

- PRFMAX 是核心简要表 (/dev/prf, 在 profiler (ADM) 中有说明) 能够正确处理的文本符号的最大数目。
- NCLIST 指定要分配多少个字符显示缓冲区。每个缓冲区最多包含 64 个字节。这些缓冲区动态地连接于终端线或其它低速设备的输入/输出队列表。每个终端所需的平均缓冲区数目是 5 到 10 个。每个入口 (缓冲区空间加上头) 包含 72 个字节。满了以后, 与终端有关的输入/输出字符就会丢失, 虽然显示还在继续。



NEMAP	指定I/O传输映射的最大数目。
NUMXT	确定一个子设备能够配置支持的位映射显示设备（例如BLIT或AT&T5620终端）的数目。
NUMSXT	确定一个子设备能够配置的shell层的数目，其缺省值为6。
NKDVTTY	确定由控制台键盘驱动程序支持的虚终端（ttys）的数目，用户不应该修改这个参数。
NCPYRIGHT	定义用以打印控制台初始化信息的核心数据结构的大小。用户不应该修改这个参数。
KDBSYMSIZE	是符号表的大小。值一定要在10000至100000之间。缺省值为60000。

### 10.7.3 文件、i节点和文件系统

NINODE	<p>指定要分配多少i节点表的入口，每个表入口都代表一个活动文件的内存i节点。例如，一个活动文件可能是一个当前目录、一个打开的文件或是一个安装点。当修改了这个变量，文件控制结构就被修改了。所用的入口数目依赖于被打开文件的数目。入口数一般在100到400之间。缺省值为300。NINODE的值直接与NFILE的值有关。（NINODE等于或大于NFILE。）NINODE一定要小于或等于NS5INODE。NINODE若大于NS5INODE，就会使得系统无法工作。当i节点表溢出时，在系统控制台上会显示如下警告信息：</p> <p>WARNING: i-node table overflow</p>
NFILE	<p>指定要分配多少个打开文件表入口。每个入口表示一个打开的文件。入口数一般在100至600之间。缺省值为200。每个入口包含12个字节。NFILE入口直接与NINODE有关。（NFILE小于或等于NINODE。）NFILE以NINODE结构同样的方式控制其结构操作。当文件表溢出时，在系统控制台上会显示如下警告信息：</p> <p>NOTICE: file table overflow</p> <p>请注意，这个参数不影响每个进程打开文件的数目（请看NOFILES参数）。</p>
NMOUNT	<p>指定要分配多少个安装表入口。每个入口代表一个被安装的文件系统。根（/）文件系统总是第一项。当满了之后，mount系统调用就会返回一个错误EBUSY。由于安装表是线性搜索的，因此这个值应该尽可能的小。</p>
CMASK	<p>是用以建立文件的缺省掩码值（umask（S））。</p>
ETRUNC	<p>当被设置为1时，就会默默地把新建立的文件文件名截为14个字符长度。如果保持为缺省值0，要生成一个文件名长于14个字符的文件就会失败并返回错误ENAMETOOLONG，这</p>

	是由 POSIX FIPS 的要求而产生的结果。
NS5INODE	一定要等于或大于NINODE。
NOFILES	指定每个进程所打开文件的最大数目。缺省值为60。除非某个应用程序包要求修改 NOFILES, 其它情况下不要改变这个 60 的缺省设置。 / bin / sh 使用了三个文件表入口: 标准输入、标准输出和标准错误 (0, 1, 2 通常分别为 stdin、stdout stderr 保留)。这就只给每个进程留了 NOFILES 减 3 个其它可打开文件的入口。如果某个进程需要比这个数目多于三个的可打开文件, 那么标准文件就必须被关闭。这种做法是不提倡的, 一定要很小心地使用。如果所配置的 NOFILE 值大于最大值 (100) 或小于最小值 (60), 那个所配置的值就被置为缺省值 (60), 同时把一条信息送到控制台上。
SHLBMAX	指定共享库的最大数目, 这些共享库可一次被连接到一个进程上。
FLCKREC	指定可以系统锁定的记录的数目。缺省值为100。每个入口包含 28 个字节。
NMPBUF	是Acer快速文件系统核心阶段缓冲区的数目。它们在把其内容传送到用户空间或高速缓存之前先把大量的磁盘请求搜索在一起。mpbuf 都很大 (每个 16K)。NMPBUF 的值应该为 0 至 16, 后者只适用于带有 16M 字节或更多内存且有很多用户的机器。如果被设置为 0 (系统缺省设置), 这个参数的真正值就在引导时确定。
NMPHEADBUF	是Acer快速文件系统独立簇缓冲区头的数目。这个参数的值一定要在 0 至 600 之间。缺省值为 150。
BFREEMIN	是在等待一个缓冲区之间在空闲表中所必须有的缓冲区的数目。这个值应该通常为 0。主要用于介质拷贝、uucp 传输, 以及其它准单用户和访问大量文件的应用, 若把这个值设为 NBUF / 10 左右, 就会获得性能的提高。这个参数的最大值为 100。
S5CACHEENTS	是在文件名缓存中的各种名字的数目。其值在1到1024之间。缺省值为 256。为了适合各工作小组的需要, 这个参数的值最好大一些, 基本上为 NINODE 值的三倍。
S5HASHQS	是为名缓存服务的hash队列的数目 (最好为质数)。其值在1到 1021 之间。缺省值为 61。为了适合各工作小组的需要, 应该使这个参数的值为 A5CACHEENTS 的四分之一左右的一个质数。
S5OFBIAS	确定保存于高速缓存中打开文件的偏向性。其值在0到256之间。缺省值为 8。如果值为 0, 则表示被打开文件的文件名没有特殊的优先权。如果 S5OFBIAS 等于 S5CACHEENTS, 那

么所打开文件的文件名一定总保持在高速缓存中。

NGROUPS

是所能提供的小组的最大数目。其缺省值为8。

#### 10.7.4 进程、内存的管理和替换

页守护程序 vhand 负责在需要时释放内存。它使用“最近用得最少”的算法去找到进程工作区，并把那些在最近一段时间里没有修改的页写到磁盘上。页的大小为 4096 个字节。如果内存异常拥挤，那么整个进程工作区中的内容都可能被换出去。

下在的可调参数确定 vhand 和 bmapflush 的运行频率以及在什么条件下运行。对大多数应用程序各参数的缺省值都应该是合适的。

NPROC

指定要分配多少个进程表的入口。每个进程表入口都代表一个活动的进程，替换程序总在第一个入口中，而 /etc/init 总在第 2 个入口中。入口的数目依赖于终端线的数目和每个用户产生的进程的数目。每个用户进程的平均数在 2 至 5 之间（再请看 MAXUP，缺省值为 25）。如果表被填满，fork (S) 系统调用就会返回一个错误 EAGAIN。NPROC 的值可在 50 至 400 之间。缺省值为 100。

MAXUP

指定一个非超级用户可运行多少个共存的用户进程。这个值一般在 15 到 60 之间。这个值不应该大于 NPROC 的值（NPROC 应该至少大于 MAXUP 的 10%）。这个值是对每个用户而言的，不是对终端而言的。例如，如果 12 个人以同一个用户标识符登录，那么这个缺省限制很快就会达到。

NREGION

指定要分配多少个区域表入口。每个区域表入口包含 36 个字节。大多数进程有三个区：文本区、数据区和堆栈区。对于每个共享内存段和所连接的共享库（文本和数据）还需要额外的区域。然而，对于一个“共享文本”程序的文本所用的区域表入口是由所有执行那个程序的进程共享的。连接到一个或多个进程上的每个共享内存段都要使用另一个区域表入口。如果系统的区域表入口被用尽了，在系统控制台上就会显示如下信息：

Region table overflow

MAXPMEM

指定物理页的最大数目。缺省值 0 表示所有的物理内存都被用了。

ULIMIT

以 512 字节的块为单位指定普通用户可写的最大文件的大小。缺省值为 2097152；即一个普通用户可写的最大文件是一千兆字节。超级用户可以写文件系统所能装得下的任意大的文件。ULIMIT 对于读不起作用；任何用户都可以读任意大的文件。

SPTMAP

确定用于管理核心虚地址空间的映射入口阵列的大小。用户不应该改变这个参数。

VHNDFRAC

确定系统变量 VHNDL 的初始值。VHNDL 被设置为最大的用户可用内存除以 VHNDFRAC 或 GPGSHI 的值，选两者中较大的一个。VHNDL 值确定什么时候运行页守护程序 vhand。可

用的空闲内存的总量与 VHANDL 相比较。如果空闲内存小于 VHANDL, 那么就唤醒页守护程序 vhand。

VHNDFRAC的缺省值为16。减小这个值会使守护程序更加活跃; 增大这个值会使守护程序的活动少一些。(这个值必须小于可用内存的 25%。)

- AGEINTERVAL** 指定在一个进程的各页超时之前这个进程可运行的时间。
- GPGSLO** 指定vhand开始从各进程偷取内存页时所剩以页为单位的空闲内存的最低水准。缺省值为 25。增大这个值会使守护程序更加活跃; 减小这个值会使守护程序活动得少一些 (一定是一个大于 0 而小于 GPGSHI 的整数)。
- GPGSHI** 指定vhand停止从各进程偷取内存页时所有以页为单位计算的空闲内存的最高水准。缺省值为 40。增大这个值会使守护程序更加活跃; 减小这个值会使守护程序活动少一些。(这个值一定要是大于 0, 大于 GPGSLO 而小于可用内存页数目的 25%的整数。)
- GPGSMK** 是由页守护程序使用的标记。缺省值为0×00000420。这个值不要修改。
- MAXSC** 指定在一次操作中所换出的内存页的最大数目。缺省值为1。
- MAXFC** 指定在一次操作中被增加到空闲表中的内存页的最大数目。缺省值为 1。
- MAXUMEM** 指定以页为单位计算的一个用户虚地址空间的最大值。这个值不能大于 8192。缺省值为 2560。
- MINARMEM** 指定为用户进程文本和数据段保留的内存页的最小数目。
- MINASMEM** 是一个门槛值, 用以指定为系统保留的内存和替换页的数目 (对于用户进程的文本和数据段是不能用的)。
- MINHIDUSTK** 指定最小的数据重定位值, 以使用户堆栈和数据能够共享一个页表。这值不应该修改。
- MINUSTKGAP** 请看MINHIDUSTK。
- MAXSLICE** 指定用户进程的最大时间片。在一个进程运行完分配给它的时间片之后, 那个进程就会挂起。操作系统然后就调度最高优先级的进程并分配给它这个时间片。MAXSLICE 的值必须在 15 至 60 之间。缺省值为 60。

### 10.7.5 时钟

**NCALL** 指定要分配多少个调出表入口。每个入口代表一个将由核心时钟处理端口调用的功能。这个值一定要在 30 到 250 之间。缺省值为 30。每个入口包含 16 个字节。

软件驱动程序可以使用调用入口检查硬件设备的状态。当调出表溢出时, 系统就会瘫痪并在系统控制台上显示如下信息:

PANIC: Timeout table overflow

- TIMEZONE** 指定在 ctime (S) 中所涉及的 timezone 的设置。请注意这个 timezone 值是系统缺省时间区而不是 TZ 环境变量值。
- DSTFLAG** 指定为 ctime (S) 系统调用说明的 dstflag。

### 10.7.6 多屏幕

- TBLNK** 控制控制台屏幕的保存特性。它是在清屏而保存监视器上内容之前的秒数。TBLNK 的值应为 0 至 32767 之间的数。缺省值为 0。
- NSCRN** 指定可由视频驱动程序使用的多屏幕的最大数目。
- NSPTTYS** 是系统上伪tty的数目。缺省值为16，最大值为32。
- SCRNMEM** 是为控制台屏幕保存信息所使用的1024字节的块数。

### 10.7.7 信息队列

下面的可调参数是与进程间通信信息有关的：

- MSGMAP** 指定用以管理信息段的控制映射的大小。缺省值为100。每个入口包含 8 个字节。
- MXGMAX** 指定一条信息的最大长度。缺省值为2048。虽然核心能够处理的最大长度为 64KB-1，但限制为 8192。
- MSGMNB** 指定一个信息队列的最大长度。缺省值为40%。
- MSGMNI** 指定全系统 (id结构) 的信息队列的最大数目。缺省值为50。
- MSGTQL** 指定系统中信息头的数目，以及特殊信息的数目。缺省值为40。每个入口包含 12 个字节。
- MSGSSZ** 指定一个信息段的字节数。信息由相邻的一组信息段组成，以适应文本的需要。缺省值为 8。MSGSSZ 的值乘以 MSGSEG 的值一定要小于等于 131072 字节 (128KB)。
- MSGSEG** 指定系统中信息段的数目。缺省值为 1024。MSGSSZ 的值乘以 MSGSEG 的值一定要小于或等于 131072 字节 (128KB)。缺省值在引导时设定。

### 10.7.8 信号量

下面的可调参数与进程间通信的信号量有关：

- SEMMAP** 指定用于管理信号量设置的控制映射的大小。缺省值为0。每个入口包含 8 个字节。
- SEMMNI** 指定核心中信号量标识符的数目。这是在任何给定时间中能够活动的不同信号量设置的数目。缺省值为 10。每个入口包含 32 个字节。
- SEMMNU** 指定系统中取消结构的数目。缺省值为 30。其大小等于  $8 * (SEMUME+2)$  个字节。
- SEMMSL** 指定每个信号量标识符的信号量的最大数目。缺省值为25。
- SEMOPM** 指定每个semop系统调用能够执行的信号量操作的最大数目。缺省值为 10。每个入口包含 8 个字节。

- SEMUME 指定每个废除结构的废除入口的最大数目。缺省值为10。大小等于  $8 * SEMMU$  个字节。
- SEMVMX 指定一个信号量能够有的最大值。缺省值为32767，它也是这个参数的最大值。
- SEMAEM 指定最大值的调整出口，又叫semadj。当一个信号量变得大于或等于semop的绝对值时就使用这个值，除非程序设置了自己的值。缺省值为16384。缺省值就是这个参数的最大值。
- SEMMNS 指定系统中信号量的数目。缺省值为60。每个入口包含8个字节。
- XSEMMAX 指定全系统允许的XENIX特殊信号量的最大数目。XSEMMAX的最小值是20，最大值是60，缺省值也是60。

### 10.7.9 共享的数据

下面的参数与进程间通信的共享内存有关：

- SHMMAX 指定共享内存段的最大值。缺省值为524288。
- SHMMIN 指定共享内存段的最小值。缺省值为1。
- SHMMNI 指定全系统的共享内存标识符的最大数目。缺省值为100。每个入口包含52个字节。
- SHMSEG 指定每个进程所连接的共享内存段的数目。缺省值为6。最大值为15。
- SHMALL 指定正在使用的共享内存文本段的最大数目。缺省值为512。
- XSDSEGS 指定全系统所允许的XENIX特殊共享数据段的最大数目。XSDSEGS的最小值为1，最大值为150，缺省值为25。
- XSDSLOTS 指定每个XENIX共享数据段的页槽的数目。整个系统的XENIX特殊共享数据段附件的最大数目为  $XSDSEGS * XSDSLOTS$ 。XSDSLOTS的最小值为1，最大值为10，缺省值为3。

### 10.7.10 系统名称

NODE 指定系统名称。

### 10.7.11 流式数据

下面的可调数据与STREAMS处理相关：

- NQUEUE 是所配置的流队列的数目。队列总是成对地分配，所以这个数应该是偶数。最小的流包含四个队列（两个用于流头两个用于驱动程序）。每个被推到流上的模块还需要其它两个队列。典型的配置值为  $4 * NSTREAM$ 。
- NSTREAM 是要配置的流头（Stream-head）结构的数目。对于每一个打开的流都需要一个，包括从用户进程打开的流和连接在多路转换器下的流都是如此。所应配置的值对应用状况的依赖性很大，但对于只有一个传输数据提供者在中等传输数据量下的计算机，这个参数的值

为 32 至 40 通常是足够的。

- NBLK<sub>n</sub>** NBLK4到 NBLK4096控制着各规模级所要分配的流式数据块和缓冲区数目。信息块头也基于这些数目进行分配：信息块的数目是所分配的数据块总量的 1.25 倍。这就给每个数据块提供了一个信息块，另外还给复制信息（核心功能 dupb() .dupmsg()）提供了另外一些信息块。最佳配置决定于主存储器的容量和用途。
- NMUXLINK** 是要配置的多路转换器链路的最大数目。每个链路结构供一个活动的多路转换器链路（STREAMS L\_\_LINK ioctl）使用。这个数目依使用情况而定；87 这个缺省分配值保证了所具有链路的需要。
- NSTRPUSH** 是可以被压到流上的模块的最大数目。这能够防止一个走入歧途的用户进程在一个流上消耗所有可用的队列。缺省值为 9，但在实际应用中，现存的应用程序最多在一个流上压四个模块。
- NSTREVENT** 是要配置的流事件单元的初始值。流事件单元用来在 poll 系统调用中记录特定进程信息。它们也用于流 L\_\_SETSIGioctI 的补充和核心 bufcall() 机制。要配置的大致最小值为同时使用 poll 进程的数目乘以每个进程要查询的流的数目，再加上当时希望使用流的进程的数目。缺省值为 256。请注意这个值不必是系统上所具有的事件单元数目的硬上限（请看 MAXSEPGCNT）。
- MAXSEPGCNT** 是能够动态地分配给事件单元的另外的内存页的数目。如果这个值为 0，那么就只有由 NSTREVENT 所指定分配的事件单元可用了。如果这个值不是 0 并且如果核心运行超出了事件单元，那么就将在某些环境下分配额外的一个内存页以建立新的事件单元。MAXSEPGCNT 给为了这个目的所能够分配的页的数目设置了一个限制。一旦某一页分配给了事件单元，那么在这之后就不能再被覆盖而用于别处。缺省值为 1。

#### 10.7.12 事件队列与设备

- EVQUEUES** 是全系统所打开的事件队列的最大数目。
- EVDEVS** 是全系统连到事件队列上的设备数目。
- EVDEVSPERQ** 是每个事件队列的最大设备数目。

#### 10.7.13 依赖硬件的参数

- DMAEXCL** 指明是否允许同时的 DMA 请求。有些计算机有这样的 DMA 芯片，当同时使用它的多个通道时，它就会失灵。对于这种计算机，DMAEXCL 的缺省设置就是 1。对于没有这种问题的计算机，把 DMAEXCL 设置为 0 就允许同时使用 DMA 的多个通道。
- KBTYPE** 对于 XT-型的键盘这个参数就设为 0，对于 AT-型的就设为 1。

## 附录 A 系统启动的预设置

当系统上电并引导时，系统某些方面的操作特性就被设置好了。系统在启动时读初始化文件，当改变运行级，或有用户登录时，系统也要进行这种操作。通过修改这些文件，可以使其更符合需要。

系统初始化文件中包含一些命令或数据，用以设置初始运行级、设置系统时钟、使能终端、启动程序、检查和安装特殊的文件系统，清理临时目录，为用户设置主目录和终端类型并显示系统信息。这里所讨论的文件有 `/ect/inittab`、`/etc/rc2.d` 目录中的文件、`.profile`、`.login` 和 `/etc/motd`。

系统管理员可以修改启动文件来建立任何所需要的初始系统和用户环境。例如，通过增加或修改 `inittab` 文件中的登记项，在系统进入或退出某一个运行级时，特定的终端可被使能（或禁止）。通过修改 `/etc/rc2.d` 目录中的程序，就可以在系统启动时自动启动进程记帐。管理员还可以修改用户主目录中的 `.profile` 或 `.login` 文件而为他们设置特殊的环境。

初始化文件是普通的文本文件，并可使用象 `vi (C)` 这样的文本编辑器来修改。请注意，文件 `/etc/inittab` 中登记项的特殊格式必须按 `inittab (F)` 说明中描述的来写。

### A.1 修改 `/etc/inittab` 文件

`/etc/init (M)` 程序在核心初始化的最后阶级开始执行，并且进程 `id (PID)` 为 1。`init` 进程启动所有其它进程。`/etc/inittab` 文件包含 `init` 用的所有指令。`init` 在三种条件下读 `inittab` 文件：在引导系统时、当由 `init` 启动的一个进程结束时、当系统管理员带一个运行级参数运行 `/etc/init` 或 `/bin/telinit` 时。系统管理员可以通过给 `init` 一个参数表改变系统运行级或使 `init` 检查 `inittab` 文件而不改变运行级。当系统改变运行级时，`init` 扫描 `inittab` 文件而取得适用于新的运行状态的指令。

---

注：当修改 `inittab` 时，这种修改只是临时的，因为每次连接核心时，都要生成一个新的 `inittab` 文件。要想永久改变初始化过程，就必须修改建立 `inittab` 文件的源。要增加一个新的登记项，就要把它加到 `/etc/conf/cf.d/init.base` 文件的尾部。要想修改一个登记项，就要在 `/etc/conf/cf.d/init.base` 文件中或在 `/etc/conf/init.d` 目录的其它文件中找到要修改的登记项并对它进行编辑。

---

`inittab` 文件中的登记项由四个由冒号分开的域组成：

label: run\_level: action: process

表 A.1 对 `inittab` 中的各域给予了说明。

如果给一个 `inittab` 登记项指定了多个运行级，那么这些运行级都被放在第 2 个域中，并且各级之间没有分隔符。如果运行级这个域是空的，那么这个登记项可以在所有由数字表示的运行级（1-6）中执行。当运行级被修改时，所有没有这个新运行级登记项的进程就会得到一个警告信号（15）。如果在 20 秒钟之后这个进程还没有停止，那它就会得到一个杀死信号（9）。`init` 的当前状态决定它如何执行 `inittab` 登记项。



表 A.1 inittab 的域

域	说明
label	最多由四个字符组成的特殊的标识符。
run_level	登记项执行时所在的 init 级。
action	指明 init 进程动作的关键字。
process	在进入特定的运行级后 init 所执行的进程。

表 A.2 inittab 单用户关键字

关键字	说明
boot	启动这个进程并不等这个进程完成就继续找下一个登记项。当这个进程结束之后, init 不再重新启动它。
bootwait	一旦启动了这个进程, 就要等它结束之后再执行下一个 inittab 登记项。
initdefault	判断最开始要进入哪个 init 级, 使用 run_level 域中的最大的数字。如果 run_level 域为空, init 就会把运行级理解为 0123456 并进入运行级 6。如果在 inittab 中没有 initdefault 登记项, 那么在引导时 init 需要从用户那里取得原始运行级。
sysinit	在 init 第一次读 inittab 时就启动这个进程, 并等它停止之后再继续执行下一个登记项。带有关键字 sysinit 的登记项都在 init 访问控制台之前执行。

表 A.3 inittab 运行级域关键字

关键字	说明
off	发送一个警告信号, 等待 20 秒钟, 然后如果那个进程还在执行就发送一个杀死信号。如果那个进程不执行了, 就不再管这个 inittab 登记项了。
once	启动这个进程并不等这个进程完成就继续执行下一个登记项。当这个进程结束之后, init 不再启动这个进程。
ondemand	功能上与 respawn 相同;ondemand 只用于带有 a,b 或 c 运行级值的登记项。
respawn	如果这个进程没在运行, 就启动这个进程,并不等这个进程结束就进入下一个登记项;当它结束之后再重新启动这个进程。如果这个进程正在运行, init 就不管这个 inittab 登记项了。
wait	启动这个进程, 并等它结束之后再进入下一个 inittab 登记项。

当 init 程序从一开始就被调用时, 它会搜索 inittab, 寻找含有表 A.2 所描述的 action 关键字的登记项。这些登记项只在 init 引导时读 inittab 的过程中执行。

当运行级从单用户转为由数字表示的运行级 (1-6) 时, init 就要搜索带有表 A.3 中所描述动作 (action) 的登记项, 并只执行设置有合适的运行级的那些登记项。

如果系统检测电源故障, 并且 init 接收到一个电源故障信号, 那么 init 就要执行包含有表 A.4 中所给出的动作的登记项。

表 A.4 inittab 电源故障关键字

关键字	说明
powerfail	启动一次这个进程，并且不等这个进程结束，就继续执行下一个登记项。
powerwait	启动一次这个进程，在继续执行下一个登记项之前等待这个进程结束。

在下面的例子中，inittab 登记项的系统运行级被缺省设置为单用户模式，以供 init 初始调用。

```
is: S: initdefault;
```

注：initdefault 这个动作没有与任何进程相联系。第三个冒号(:) 是必需的，如果丢了，init 就会忽略这个登记项。

可以对这个登记项进行编辑，把系统配置为启动后进入多用户状态，只要把运行级域 (run\_level) 中的 S 改为 Z。

下面的 inittab 登记项在所有的 init 级都可运行 /etc/bcheckrc，并且要等这个进程完成之后再执行下一个登记项。(bcheckrc 对文件系统进行检查并设置日期。)

```
bchk::sysinit: /etc/bcheckrc < /dev/console > /dev/console 2 > &1
```

init 进程也按照 inittab 中的指令启动 /etc/getty (M) 程序。getty (get a tty) 程序在系统与终端之间建立通信。

下面这个 inittab 登记项让 init 在当前的运行级 2 上为 tty01 启动一个波特率为 9600 的 getty 进程 (如果还不存在这个进程)：

```
CO:2:respawn: /etc/getty tty01 m
```

respawn 这个动作指示 init 每当这个进程结束时都要重新启动它，并且不等当前进程结束就继续执行下一个登记项。如果把 respawn 改为 wait，那么就是告诉 init 等到当前进程结束后再读 inittab 中的下一个登记项。

为了使得对 inittab 的修改立即生效，应执行 telinit Q 命令。这个命令使 init 在不改变运行级的情况下重新检查这个被修改的 inittab 文件。

## A.2 修改 /etc/rc2.d 中的程序

在从较高级的 init 状态 (3-6) 或从单用户模式进入 init 状态 2 (多用户模式) 时，init 按照 /etc/inittab 中的指令执行 /etc/rc2 程序。rc2 设置某些环境参数并运行 /etc/rc2.d 中的程序。在 rc2.d 中的一些程序还要运行 rc.d 目录的子目录中的程序。rc2 所运行的程序以字母顺序执行以保证正确的初始化。

这一部分将介绍 /etc/rc2.d 目录中由 rc2 运行的程序并讲述向这个目录中加入自己程序的步骤。

表 A.5 以 rc2 执行的顺序给出了 /etc/rc2.d 目录中各程序的简要说明。

表 A.5 /etc/rc2.d 中的程序

程序	说明
S00SYSINIT	启动核心信息登录程序
S01MOUNTFSYS	安装在 /etc/default/filesys 中指定的文件系统
S03RECOVERY	在瘫痪后整理 vi 编辑对话区
S04CLEAN	删掉临时文件
S05RMTMPFILES	删掉临时文件
S15HWDNLOAD	硬件卸载
S16KERNINIT	启动 mapkey、进程记帐、网络、和其它核心初始化
S20syssetup	配置打印系统并生成 /etc/systemid
S21Perf	系统记帐
S70uucp	清理 uucp 封锁文件
S75cron	启动 cron 守护程序
S80lp	启动 lpsched 和网络实用程序
S87USRDAEMON	启动用户守护程序
S88USRDEFINE	在引导之后执行用户定义的命令
S90RESERVED	把自动引导期间所保存的 fsck 的输出邮送到根(root)。

在系统上的 /etc/rc2.d 目录中可能还包含某些在表 A.5 中没有列出的程序。这是因为，在安装期间很多附加程序把它们自己守护程序的初始化程序也插到了这个目录中。这个目录中可能还包括用以清除某附加程序的临时程序或用于封锁文件的程序。

在系统进入 init 状态 2 时，管理员也可以写自己的程序来运行。例如，可以写一个建立 RAM 磁盘或启动网络的程序并把它加到 /etc/rc2.d 中。

在写一个要放到 rc2.d 中去的系统启动程序时应考虑如下因素：

- 在从一个较高或较低的 init 状态进入到 init 状态 2 时，以 S 开头的文件要带 start 选项来执行。

- 在从一个较高的 init 状态（3-9）变换到 init 状态 2 时，在 rc2.d 中的以 K 开头的程序要带 stop 选项来执行。

- 以除 S 或 K 以外的字符开头的文件都被忽略。

- 文件以 ASCII 码升序顺序执行；在文件名中的数目决定其执行顺序。换句话说，这些程序以在 etc/rc2.d 中执行 ls 命令时各程序所出现的先后顺序来执行。

表 A.6 按顺序给出了能够用来命名文件的 ASCII 码字符。

表 A.6 有效的文件名字符

#	%	+	,	-	.	0	1	2	3
4	5	6	7	8	9	:	=	?	@
A	B	C	D	E	F	G	H	I	J
K	L	M	N	O	P	Q	R	S	T
U	V	W	X	Y	Z	^	_	a	b
c	d	e	f	g	h	i	j	k	l
m	n	o	p	q	r	s	t	u	v
w	x	y							

• 管理员自己的程序可能要依赖于其它程序启动的设备程序或守护程序（例如网络设置，cron 守护程序，或打印机调度程序）。例如，如果管理员自己的程序依赖于某些被安装的文件系统，那么就一定要这样程序命名，使其在 S01MOUNTFSYS 程序之后运行。（当用 mkdev fs 增加了一个新的文件系统，就应该用系统进入多用户模式时安装新文件系统所需要的信息来更新相应的文件。）

要想给初始化过程增加一个功能，应该遵循这些步骤：

1. 写一个能够实现所需功能的程序。

2. 给这个文件命名，文件名以大写字母 S（或 K）开头，后面跟着一个两位数，以指明它相对于目录中其它文件的执行顺序，并以一个说明这个程序功能的名字结尾。例如，S03RECOVERY 进行瘫痪后的恢复，它在以 S02 开头的程序之后并在以 S04 开头的程序之前执行。一定要遵循这个命令规则以确保自己的程序在正确的时间执行。

3. 把这个程序拷贝到 /etc/rc2.d 目录中，使其能够在系统进入（或退出）多用户模式时被执行。

如果想加的功能与已经处于 /etc/rc2.d 目录中的一个程序所实现的功能属于同一类，那么只要对这个已存在的程序进行编辑并加入新的功能就可以了。例如，可以向文件 S70uucp 中加入有关 uucp 的功能。也可以对任何程序进行编辑并对其裁剪以符合自己的需要。例如，在启动进程记帐时，从 S16KERNINIT 文件中删去适当的注释。请记住在对一个程序进行修改之前一定要对原始程序进行备份。

因为可靠的 UNIX 的安全性特性要求所有的进程都应标记一个 LUID（注册用户 ID），如果向 /etc/rc 结构中加入了一些调用 setuid 或 setgid（设置用户 ID 或组 ID）程序的初始化程序，那么一定要记住要设置 LUID。这样才能保证系统准确记载谁做了什么，即使用户的有效标识符改变了。如果想不设置 LUID 就运行 seruid 或 setgid 程序，那么就会得到一个错误信息：“cannot execute”。如果一个程序设有改变进程的 ID，那么也可以不设置 LUID。

为了设置 LUID，可以这种形式使用 su (C) 命令：

```
su username -C "command"
```

其中 username 是用户或帐号的名字，command 是管理员想执行的命令的全名。

例如，deliver 守护程序周期性地检查没有发送的邮件。这个程序在文件

/etc/rc2.d/s86mmdf 中。(反斜线是用来分割长命令行的。)

```
su mmdf -c "/usr/mmdf/bin/deliver -b -T3600\  
-cuucp -L /user/mmdf/log/uucp.log"
```

这个命令把 LUID 设置为 mmdf, 它是用来管理邮件系统的 LUID。当系统进入多用户模式时这个命令自动执行。

### A.3 修改 .profile 和 .login 文件

/etc/profile 和 /etc/eshrc 文件分别为所有的用户设置 Bourne 和 C shell 作为缺省环境。在用户主目录中的 .profile 和 .login (对于 C-shell 用户) 文件包含对用户的环境进行初始化的命令。当一个 Bourne shell 的用户登录之后, shell 首先执行在 /etc/profile 中的命令, 然后再执行用户主目录中 .profile 文件中的命令。当一个 C-shell 用户登录之后, shell 就执行 /etc/cshrc 中的命令, 然后执行用户主目录中 .login 文件中的命令。(根据所注册的 shell, 还可能用到其它文件。)

这些文件中包含设置不同的系统变量 (例如, TERM.PATH 和 MAIL) 的命令。这些变量给出系统信息, 例如用的是什么样的终端, 到哪去找用户所要运行的程序, 到哪去找用户信箱, 以及对于 "kill" 和 "backspace" 功能需要什么关键字。

在系统中每个用户有一个 .profile 或一个 .login 文件。当这个用户帐号建立时, 这个文件修改他们自己的 .profile 或 .login 文件, 或者让系统管理员进行修改。无论在何种情况下, 这些文件都是普通的文本文件, 并可以使用文本编辑器对它们进行修改; 可按需要加入或删除命令。

### A.4 修改 /etc/motd 文件

工作日信息文件 /etc/motd 包含当用户登录时所得的欢迎信息。最开始, 这个文件包含操作系统的名字和版本号。因为 /etc/motd 是一个普通的文本文件, 所以系统管理员可以用一个文本编辑器对这个文件进行编辑以修改其中的信息。一般情况下, 应该限制 /etc/motd 文件的大小, 使其包含的内容不超过一个屏幕的信息。

可以修改这个文件, 使其包含这样一些信息: 提示用户对目录进行清理以保留磁盘空间, 通知用户进行下一次的定期备份, 最新系统升级的说明, 或是系统关机的时间。当用 wall 宣布将要非例行关闭系统时, 可以编辑 motd 使其包括关机信息, 这样在运行了 wall 以后登录的用户就会被告知马上就要关机。

## 附录 B 作业调度命令的使用授权

这个附录将介绍如何允许或禁止用户访问或使用作业调度命令。sysadmsh 中的选项 Jobs → Authorization 包含授权功能。另外, 还可以通过建立一个模型文件指定 at 或 batch 命令的执行环境来控制 at 和 batch 命令。

---

注：除了给予使用这些命令的权限外，还必须确保用户有 `chmod sugid` 这个核心权限。如果原来没给用户这个核心权限，就必须按“用户帐号的管理”一章中所讲的那样把这个权限分配给用户。

---

### B.1 为作业调度修改缺省授权

系统最原先的配置是只允许根 (root) 使用作业调度命令，而不允许其它用户访问这些命令。

#### 1. 修改 cron 的缺省设置

为了修改 cron 的系统缺省设置，应该做如下的 `sysadmsh` 选择：

Jobs → Authorize → Scheduled → Default

这时会显示三个选择对象：

None 对任何用户都不允许执行

Allow 允许所有用户执行 cron 命令

Deny 禁止所有用户访问 cron 命令

当前的状态以高亮显示表示。使用箭头键可把高亮显示移到所需的选项上，或选第一个字母。请记住，用户也可以分别被允许或禁止（这将在后面说明）。对单独用户的设置要比系统的缺省设置优先级高。

#### 2. 修改 at 或 batch 的缺省设置

为了修改 at 和 batch 命令的系统缺省设置，可以选择如下的 `sysadmsh` 选项：

Jobs → Authorize → Delayed → Default

这时会显示三个选择对象：

None 对任何用户都不允许执行

Allow 允许所有用户执行 at 或 batch 命令

Deny 禁止所有用户访问 at 或 batch 命令

当前的设置状态以高亮度表示。使用箭键可把高亮度显示移到所需的选项上，或选第一个字母。请记住，用户也可以分别被允许或禁止（这将在后面说明）。对单独用户的设置要比系统的缺省设置优先级高。

### B.2 允许或禁止个别用户使用 cron

为了修改 cron 的系统缺省值，可进行如下 `sysadmsh` 选择：

Jobs → Authorize → Scheduled → User

光标被置于 User: 域。输入用户名或按 <F3> 得到一个用户表。当选取了某个用户名后，将显示如下的选择对象：

Allow 允许这个用户执行 cron 命令。

Deny 禁止这个用户访问 cron 命令。

使用箭头键把高亮显示移到所需要的设置上。这个设置将替代系统缺省设置。

### B.3 查看用户的 cron 权限

要想查看用户是被允许还是被禁止使用 cron, 可进行如下的 sysadmsh 选择:

Jobs→Authorize→scheduled→View

如果系统缺省值为 Allow, 那么就会显示一系列不允许访问 cron 的人。如果系统缺省值为 Deny, 那么就会显示一系列允许访问 cron 的用户。

#### B.4 允许或禁止个别用户使用 at 或 batch

要想修改 at 或 batch 的系统缺省设置, 可进行如下的 sysadmsh 选择:

Jobs→Authorize→Delayed→User

光标被置于 User: 域。输入用户名或按 <F3> 得到一个用户表。当选取了用户名之后, 将会显示如下选择:

Allow 允许用户执行 at 或 batch 命令。

Deny 禁止用户访问 at 或 batch 命令。

#### B.5 查看用户的 at 或 batch 权限

要想查看用户是被允许还是被禁止使用 at 或 batch, 可进行如下 sysadmsh 选择:

Jobs→Authorize→Delayed→View

象 cron 一样, 如果系统缺省设置为 Allow, 那么就会显示一系列允许访问 at 或 batch 的用户; 如果系统缺省设置为 Deny 那么就会显示一系列不允许访问 at 或 batch 的用户。

#### B.6 用于 at 或 batch 命令的环境文件

at 和 batch 命令的执行环境是可以定义的, 为了编辑 at 和 batch 的模型文件, 可以分别使用如下的两个 sysadmsh 选择:

Jobs→Authorize→Environment→At

Jobs→Authorize→Environment→Batch

这些选项可用于对文件 /usr/lib/cron/.proto (at) 或 /usr/lib/cron/.proto.b (batch) 进行编辑。这样文件都被放在为所有 at 和 batch 作业所做的 shell 程序的开始。这个 shell 程序必须符合通常的 /bin/sh 句法, 并包含用于这个模型文件的一些特定的变量:

\$d 这是提交作业时用户的当前目录。

\$l 这是提交作业时给用户的 ulimit。

\$m 这是提交作业时给用户的 umask。

\$t 这是这个 shell 程序运行的时间 (以秒为单位, 从 1970 年 1 月 1 日算起)。

\$< 这是由用户提交的整个 shell 程序替代程序。一般情况下, 这个变量出现在这个文件的最后, 也就是说处于管理员所建立的文件头之后。如果想在这个量之后包括一些信息, 那么这个 shell 程序会在到达这些信息之前就结束。

只有超级用户能够编辑这些文件。

模型文件有很多用处; 这里有两个模型文件的例子:

• 插入 nice (C) 命令, 使一个作业队列中的作业处于较低的优先级运行:

```
nice -5 / bin / sh << 'END_OF_FILE'
```

```
$ <
```

```
END_OF_FILE
```

- 使一个作业队列使用另一个 shell 执行命令:

```
/bin/csh << 'END_OF_FILE'
```

```
$ <
```

```
END_OF_FILE
```

对于大多数工作，随盘所带的模型文件应该是够用的。



# 第三部分 外设使用指南

---

## 第一章 增加多功能卡、内存卡及其它总线卡

计算机的总线（或母板）是系统的中心，系统管理人员必须与总线及与其相关的硬件打交道。要在系统中找到总线，从计算机的主体上移去外壳，会看到一个较大的电路板，其上有留给另外电路板使用的扩展插槽。这些电路板即所谓总线卡。

总线卡可以是系统的扩展内存、内部调制解调器，用于附加终端的多端口串行板、用于硬盘、磁带驱动器等外围设备的控制卡、用于带有彩色和图形功能的监视器的控制卡、鼠标控制器以及其它类似的扩展卡。本章中我们讲述一点关于总线卡的知识以及如何将它们安装于你的 UNIX 系统中。大多数带总线卡的设备的安装过程都在本书中另辟章节详细介绍。

### 1.1 安装总线卡

要安装总线卡，必须首先关闭操作系统并关掉系统的电源，保证计算机的插头已经拔下，以免出现触电事故。在开始在计算机上工作之前，用触摸计算机之外的一金属物体外壳的办法使自己接地，以免在接触计算机内部的硬件时静电通过手对计算机产生不利影响。

#### Dip 开关和跨接线（跳线）

在将电路板插入总线之前，保证板上已没有任何设置需要加以改变。随板而来的硬件文献中会给出缺省设置值，以及修改的办法。一般来讲，要改变板的设置，有 Dip 开关和跨接线两种形式。Dip 开关有“上”和“下”两种位置，如果板上有 Dip 开关，硬件文献中就应该有正确的设置状态。跨接线是跨接于伸出板上的小金属柱之间以实现连接的管形器件。将跨接线从一对金属柱之间移向另外一对金属柱之间实现不同的连接方式，可以改变板上的设置。

---

注：工作于 UNIX 系统的硬件一般都采用缺省设置，必须对板上的设置加以改变的情形是不多见的。

---

#### 硬件的安装

在打开计算机外壳裸露出扩展插槽时操作必须小心谨慎。新系统的可用插槽数应为 8 至 10 个，其中部分插槽要比其余插槽长些，而总线扩展卡则是可长可短，短卡的长度只有较长卡的一半。通常只有 2 到 3 个短插槽，其余的都是较长的。找一个适合你的的电路板的插槽，缓慢而有力地将板插入其中。注意扩展板的插入方向。如果插反了方向，会带来毁灭性的损失。当然，大多数扩展板都只能以一个方向，即正确的方向才能插入槽中。

这一步工作完成之后，盖上计算机外壳，开机启动。可能需要先按下面讲到的办法运

行厂商提供的 setup 程序，改变系统的配置之后才能使用新的硬件。

### 使用厂商的 setup 盘

随计算机一同到货的，应该还有一张由厂家提供的可启动软盘，上有 setup 程序。将此盘复制一张以供使用，然后将原盘放到一个安全的地方保存起来。此盘的作用是将描述系统硬件构成的信息配置于计算机的永久存储器中。无论何时硬件设备增加到系统中，都应运行 setup 程序，以告诉计算机关于硬件的变化。也有的计算机可自动识别硬件的存在。厂商提供的有关文献会让你知道是否有必要运行该软件。

## 1.2 增加存储器

可以用增加系统内存时的方法提高系统的性能，并运行更大的程序。

增加系统内存的步骤如下：

1. 关闭计算机。有关这一任务的步骤见本书中“启动和停止系统”一章。
2. 按照厂商的指令安装扩展内存。必须按照指令描述的方法设置所有开关。
3. 引导操作系统。此时屏幕信息详细描述了扩展内存对系统的影响。
4. 部分特性得到增强，比如：
  - 更多的多屏幕
  - 更多的缓冲区
  - 更大的用户进程最大容量。

其中多屏幕的数目也许没有改变。因为多屏幕的数目可由用户设置，而你可能已经对多屏幕的数目设置了某一上限。如果没有设置多屏幕数目的上限，则必然已经用着系统允许的最大数目的多屏幕。

缓冲区的数目也可能不发生改变，因为缓冲区的数目也可以由用户设置，而你可能已经设置了可用的缓冲区数的上限值。如果没有设置缓冲区数的上限，则所用的缓冲区数必然已到了系统允许的最大值。

如果用户进程的最大容量没有改变，则就是因为对换区文件系统大小不够而不是因为内存量不够。这时可采取以下措施：

- 重新安装操作系统，增加对换空间的大小。
- 对进程进行修改，使之可以不被对换出去而一直在内存中运行。

如果想从系统中移去部分内存，可按同样的过程执行。

如果内存硬件报告一个错误，则显示：

PANIC: memory parity error

接着就看到软件重新启动信息：

\* \* safe to power off

—or—

\* \* Press any key to reboot \* \*

如果系统重复地报告出现奇偶校验错，则应考虑更换存储器芯片。

---

注：有的机器对可以安装的最大内存量有硬件限制。查阅你的计算机硬件手册以确定可安装内存的最大量。

---

### 1.3 增加及配置串行端口

要增加一块多端口扩展卡，你必须首先确定该卡是“灵敏”串行卡还是标准串行卡。如果该卡是“灵敏”（smart）卡，则厂商已提供了安装软件及驱动程序，你所要作的工作就只是把它安装到你的 UNIX 系统上。

在安装卡之前，查阅本书中“兼容的硬件”一章，看看该卡是否与系统兼容。如果安装的是支持 4 至 8 个端口的扩展卡，则应检查引导信息，看系统在引导时是否已识别出该卡。如果引导过程没有报告该卡的存在，则定是卡上的开关设置不正确。在硬件手册中查出正确的开关设置状态并据此改正卡上的开关状态。

由厂商提供的驱动程序在引导时也许不会给出对扩展卡的确认信息。如果你的串行扩展卡是带有厂商提供的驱动程序的“灵敏”卡，则在安装该卡时不必运行 `mkdev serial` 程序。要使系统识别新卡，应运行厂商提供的安装软件。

应为两个标准 COM 端口配置中断：COM1 用中断 4，COM2 用中断 3。大多数串行卡每块卡用一个中断，所以可以有两块 4 端口卡可以使用 COM1 和 COM2。应注意其他产品及硬件的需求，以免发生中断冲突。关于 COM1 和 COM2 的内容请参考 `serial (HW)` 的解释。

如果你的卡是一个标准串行卡，则下面的指令给出了为新增端口建立新的设备文件的方法：

1. 引导系统并进入系统维护方式：
2. 在系统维护方式下，输入命令：

```
/etc/mkdev serial
```

Systemadmsh 的用户应选择：

```
System → Hardware → Card-Serial
```

3. 显示下面的信息：

```
Serial Board Initialization
```

```
You would like to install a:
```

1. 1port card
2. 2port card
3. 4port card
4. 5port card
5. 8port card

```
Select an option or enter 'q' to quit:
```

输入正确的数字并回车。

4. 程序响应以如下菜单：（在大多数系统中只显示 COM1 和 COM2，并只有两者可用）

```
The card is configured as:
```

1. COM1
2. COM2
3. COM3
4. COM4
5. COM5

```
Select an option or enter 'h' for help or 'q' to quit:
```

如果选择“h”，则看到一张表格，列出了端口、扩展卡类型，I/O 地址以及状态地址。

输入一个数字并按回车键。在 mkdev 程序接受了 COM 槽之后，即可看到一张清单，给出了新配置的端口以及相应的带调制解调器控制的端口。比如，tty2a 与 tty2A 是同一个串行端口，tty2A 带有调制解调器控制，tty2a 则是不带调制解调器控制的同一端口。每次访问端口可只有一个名字，用带调制解调器控制的和不带调制解调器控制的端口名均可。

5. 系统询问是否使用缺省波特率。

Would you like the modem devices set at the default speed of 1200 (Y/N)?

如果回答“y”则使用缺省波特率，否则会要求提供波特率值。

6. 这一步完成之后，显示类似如下的信息：

Modifying system files...

System files have been successfully updated.

The following standard serial devices have been installed:

tty2a tty2b tty2c tty2d

The following devices access the same physical port as their lower case counterparts ,but have modem control properties:

tty2A tty2B tty2C tty2D

这时串行端口配置完成。应保证它们在系统硬件配置中也有定义。

## 第二章 打印机

对任何计算机系统来说打印机都是很有用的。大多数系统都需要将数据打印到纸上。系统支持的打印硬件或行式打印机的范围很广。有的行式打印机是并行设备，但大多数还是以串行设备形式连接的。

要加入一台打印机，系统管理员必须：

- 将实际硬件连接到计算机上，然后
- 使用正确的系统命令使打印机处于有效状态。

请注意，打印机与系统之间的物理连接随硬件配置的不同而变化。

UNIX 行式打印机 spooling 系统是一组帮助系统管理人员有效地安装、监视和控制服务于自己系统的行式打印机的命令。打印文件的请求送到将送给打印机的打印作业队列中，等待执行。打印机依照顺序执行这些请求。

当用户需要用 LP (C) 命令打印文件时，行式打印机系统用一个“请求标识符” (request Identifier) 回答，其中包含了将用于打印的那个打印机的名字和一个用以识别该文件的唯一的号。使用请求标识符，用户可以查出打印请求的级别，或者删除打印请求。LP 的任选项使得用户可以很容易地控制打印机的输出。

打印服务完成以下任务：

- 处理接收用户所要打印文件的工作
- 如果必要就对文件进行处理，使之可以正确打印出来
- 调度一台或多台打印机的工作
- 启动与打印机接口的程序
- 跟踪作业的执行状态
- 提醒系统管理员注意打印机故障
- 在出故障时发布错误信息

下面几条术语用以描述打印设备的操作

**device:** LP 输出的目的地。它可以是硬线连接的打印机，也可以是有时被当作打印机用的终端，或者是常规文件。设备（打印机、终端或文件）是用完整的路径名来表示的。

**printer:** 系统管理人员指定来表示某一设备的名字。该名字最多可以有 14 个字符。在不同的时候，一个 printer 可以用于不同的设备。

**class:** 一组有序的打印机。发送到一组打印机的打印请求由该组中第一个可用的打印机执行。

**destination:** 指的是打印请求送往的地方。destination 可以是一组也可以是一台打印机。

### 2.1 安装打印机

必须将打印机连接到正确的端口上（串行打印机连到串行端口，并行打印机使用并行

端口), 保证它正常工作, 然后再使用系统管理 shell 命令 `sysadmsh` 的 `Print` 选择项安装 UNIX 打印机假脱机 (spooling) 软件。

---

注: 如果在连接打印机时遇到问题, 请查阅本书中“解决系统问题”一章的内容。

---

安装打印机应按以下步骤执行:

1. 给打印机找个地方安放起来, 应保证打印机装配正确, 通上电源。
2. 如果打印机是串行的: 将计算机串行端口的 RS-232 电缆与打印机的端口相连。串行打印机必须能够支持 XON/XOFF 或者 DTR (数据终端准备好) 协议, 并且按照这些协议进行配置。接着输入下面的命令, 其中用正确的端口号代替下面命令中的 `nn`:

```
disable / dev / ttynn
```

按回车键。该命令使连接到你的打印机的端口被置为无效。同时允许串行通信使用该端口。

3. 如果打印机是并行的: 打印机必须使用标准 Centronics 接口电缆。在单显示卡上的并行口应该配置以 7 号中断向量 (查阅“增加多端口卡, 内存卡和其他总线扩展卡”一章中配置中断向量部分的内容)。单色显示卡上的并行端口在系统引导时被自动识别为 LP1。主并行口 (相对单显卡上的并行口而言) 应配置以中断向量 7, 被识别为 LP0。必须使用主并行口或单显卡上的并行口之一——而不可两者同时使用——以避免发生硬件冲突。备用的或称“第二个”并行口应配置以 5 号中断向量, 被自动识别为 LP2。应该保证没有其他硬件使用这几个中断向量。

4. 直接把数据送给打印机以检验打印机是否安装好了。输入下面的命令之一:

对串行打印机来讲:

```
date > / dev / ttynn
```

这里 `nn` 是使用的串行端口号 (比如, `/ dev / tty1a`)。对并行打印机而言:

```
date > / dev / lpn
```

这里 `n` 是使用的并行端口号 (比如: `/ dev / lp0`)

5. 如果没有看到打印机上印出日期, 则极可能是发生了某种硬件故障, 应检验以下几项内容:

对串行打印机而言

- 确信使用的是非调制解调控制设备, 如 `/ dev / tty1a` 而不是 `/ dev / tty1A`。
- 用只将第 2, 3, 7 针相连电缆试一试。
- 重新检查打印机的配置情况, 方法是将打印机的开关设置与打印机硬件手册中内容相比较。

- 重新检查串行端口的开关。如果你正在用多端口卡, 应用该卡上的其他端口试一试, 不过要保证不与标准 COM 端口发生冲突。

- 将打印机连到标准串行端口 COM1 或 COM2 上试试, 看打印机及电缆连接是否正确。

对并行打印机而言

- 保证电缆的连接是正确的并且所有连线都是好的。在一个已知工作正常的系统上用

一下电缆，或者在 DOS 下做一做打印工作都是比较好的测试方法。

- 重新检查打印机的配置情况，方法是将打印机的开关设置状态与打印机硬件手册中描述的相关内容作一比较，看是否一致。

- 重新检查并行卡上的开关设置，而且并行卡必须在系统引导时能被识别，要检验这一点可以通过运行 hwconfig (C) 命令，或者在 /usr/adm/messages 目录中寻找类似下面的信息：

```
Parallel 0x378-0x37A 07-unit=0
```

验证并行卡被系统识别与否，可输入如下命令：

```
hwconfig name = parallel
```

如果系统已经识别了并行卡，则会打印出一条类似上面给出条目的信息。

6. 若已将打印机安装成功并且打印机已能正确响应，就可以准备运行系统管理 shell 程序 Sysadmsh。Sysadmsh 的 Printers 选择项会显示一个表格，其中有一系列域必须由你填入。如果在回答问题时出了错误，只需按 <ESC> 键即可退回。首先，应作如下的 Sysadmsh 选择：

Printers→Configure→Add

屏幕上显示以下内容：

Enter the name of the new printer

Friday August 31, 1990 1:06

Adding a Printer

Printer name	[	]
Comment	[	]
Class name	[	]
Use printer interface	[Existing]	Copy New
Name of interface	[	]
Connection	[Direct]	Call-up
Device name	[	]
Dial-up information	[	]
Device	[Hardwired]	Login
Require banner	Yes	[No]

下面对以上各域的含义加以解释：

Printer name: 新的打印机名。限制不超过 14 个字符。

Comment: 描述打印机的注释。

Class name: 本打印机所属的 class (组) 名。按 <F3> 键即可在屏幕上显示本 class 的所有打印机。

User printer interface: 使用打印机接口情况。

Name of interface: 接口程序名。按 <F3> 键可以得到现有的接口程序名清单。

Connection: 指示打印机是直接连接至计算机，还是必须通过调制解调器或网络开关

进行寻呼通信。

Device name: 连接打印机的设备名。(比如, 连接串行打印机的 /dev/tty01, 连接并行打印机的 /dev/lp0。)

Dial-up information: 调制解调器电话号码或者网络系统名称。

Device: 指示连接是供打印机专用, 还是也可由注册终端使用。

Require banner: 指示每次都要打印标题, 或者允许用户指定是否要打印标题。

在填定上述表格之后, 该选项命令即被执行, 新的配置信息就存放好了。要想使用打印机, 你还必须启动打印服务, 使打印机有效并允许打印机接收请求。这些工作都可以通过以下的 Sysadmsh 选项来完成:

Printers→Schedule→Begin

Printers→Schedule→Enable

Printers→Schedule→Accept

在做 Enable 和 Accept 选项时, 则必须根据提示信息的要求提供打印机名字。

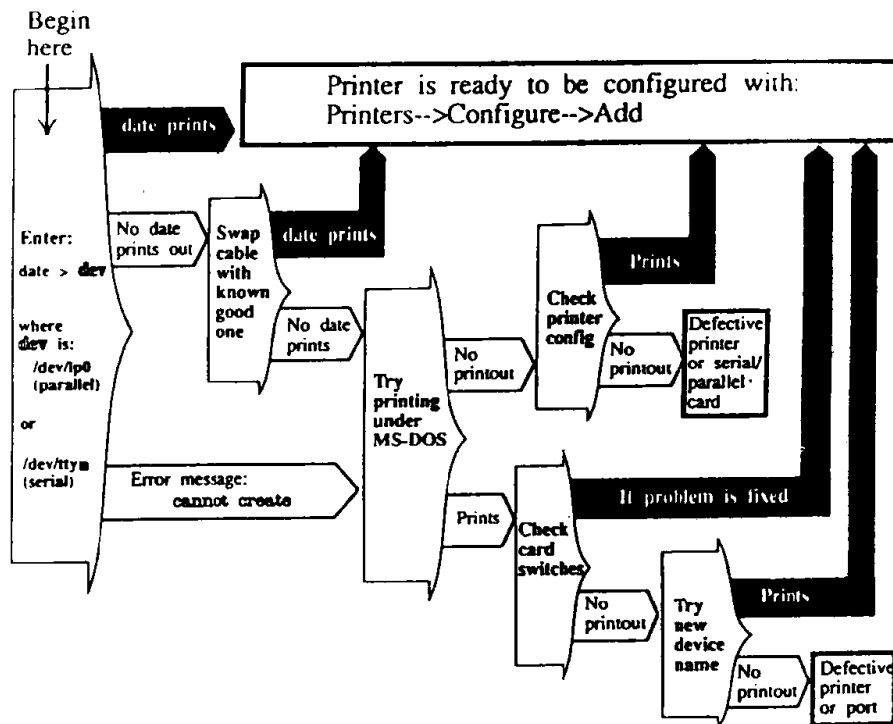


图 2.1 检测打印机连接的过程

## 2.2 打印机配置选项

对大多数的需要来讲, 使用打印机配置的缺省参数值就足够了。但也会用到部分选项来配置某些特殊的计算机操作特性, 比如错误提醒及错误恢复选项。下面是为定义所有打印机配置所给出的附加信息的清单:

- interface program (接口程序)
- printer type (打印机类型)
- Content type (内容类型)
- Connection method (连接方法)



- Character set or print wheels (字符集或打印字型轮)
- fault alerting (错误提醒)
- fault recovery (错误恢复)
- use restriction (使用限制)
- default printing attribute (缺省打印属性)

在打印设备上增加一台新的打印机时所必须给出的上述这类信息并不太多，但所提供的信息越多，打印机的管理就完成得越好，打印机为使用者提供的服务就越丰富。

### 2.2.1 接口程序

接口程序 (又称为接口控制程序) 用于对打印机进行初始化。这些程序是位于 /usr/spool/lp/model 目录中的字符文件 (文本文件)。标准接口程序定义了大多数打印机共同的最小能力。下面表中给出你的系统中已有的接口程序。当用 Sysadmsh 的 Printers >Configure->Add 安装打印机或者用 Sysadmsh 的 Printers >Configure->Modify 选项修改打印机配置时就应选择其中一个接口程序:

表 2.1 打印机接口程序表

---

1640:	DASI1640终端
5310:	AT&T5310 / 20矩阵打印机
TandyDMP:	TandyDMP打印机
dumb:	“哑”行式打印机
emulator:	用IBM仿真方式的Tandy打印机
epson:	Epson串行或并行打印机
f450:	DASI450终端
hp:	HP2631a行式打印机
lqp40:	LQP-40Letter Quality打印机
network:	通过UUCP,Micnet或者ethernet的远程打印
postscript:	postscript打印机
pplx:	带并行接口的Printronix行式打印机
proprinter:	IBM ProprinterXL
prx:	Printronix行式打印机
qume1155:	Qume Sprint 1155行式打印机
standard:	标准打印机接口程序
ti800:	Texas Instrument 855打印机

---

许多接口程序都有特殊的任选项，在 LP 命令中用 -o 格式即可指定。也可以自己建立接口控制程序，对现有的接口程序加以修改以满足需要。

### 2.2.2 打印机类型

打印机类型对打印机的正确使用是很重要的。打印设备利用打印机类型从 terminfo 数据库中分离出有关本打印机的信息。该信息描述了打印机的特性，这样当你提供的某些配置信息对本打印机不太合适时，系统会给出警告信息。有关打印机的信息还包含了在打

印文件时对打印机进行初始化所需控制数据。即使没有要求指定打印机的类型，最好也指定一个，这样打印服务会令人满意一些。

打印机类型是打印机的类别名，应像下面例中那样指示打印机类型：

```
/usr/lib/Lpadmin -P Printername -T Printer type
```

这里 printername 处应填打印机名，Printer type 处应填打印机类型。

Sysadmsh 的用户应选择：

Printers → Configure → Parameters

如果没有定义打印机类型，则使用缺省值“未知”，这样当打印服务查找打印机的信息时，就会产生“空”结果。使得打印服务不能确认某些请求或者不能对打印机进行初始化。

### 2.2.3 内容类型

打印机类型信息告知打印服务正在使用的打印机是什么类型的，而内容类型信息则是告知打印服务哪些类型的文件可以进行打印。大多数打印机只能打印一种类型的文件，对它们来讲，内容类型在打印机类型固定时是唯一的。但是对于有些能接受多种类型的文件并将它们正确打印出来的打印机来讲，情况就不一样了。在增加这样的打印机时，应该列举出所有它能接受的文件类型名的清单。

当一个文件交给打印服务打印时，打印服务即查找一个能执行这一作业的打印机。打印服务要么通过内容类型名要么通过打印机类型名都可找到合适的打印机。因此，在交付文件打印时可以指定名字，也可指定为 no-name。

内容类型名看起来与打印机类型名极为相似，但可随意选用一些对使用打印机的人来讲有一定意义的名字（simple, terminfo, any 等名字作为具有特定含义的名字被打印服务所接受，注意在使用这些名字时应保持一致性）。名字最多由 14 个字符组成，包括字母、数字和下划线。如果同一类型的内容在几种不同类型的打印机上都是可打印的，那么在增加这些打印机时应使用相同的内容类型名，这么做给使用计算机的人带来方便，因为他们可以用同样的名字来标识所打印文件的内容类型而不用考虑打印是在哪一台打印机上完成的。

例如，几个厂家生产能接受 Postscript 文件的打印机，这些打印机需要不同的打印机类型以便每台都可以正确地进行初始化（假设其初始化控制过程是不一样的），它们也许能处理同一类型的输入文件，在此为 postscript。

并不一定要为打印机开出可接受的内容类型清单。如果没有这么做，打印机类型会被用作该打印机所能处理的内容类型名。如果没有指定打印机类型，则打印服务假定这台打印机只能打印内容类型为 simple 的文件。如果能够要求人们选择自己该用的打印机实施打印，并且在文件交付打印之前保证文件已经准备好，那么这样假定也就足够了。

UNIX 系统上最常见的文件类型称为“Simple”，这类文件只包含可打印的 ASCII 字符及下面列出的控制字符：

backspace: 将打印机械回移一个位置，在每行开头的地方除外。

tab: 将打印机械移到下一个 tab 停顿处，在大多数打印机上两个 tab 停顿处相距 8 个字符。

linefeed: 将打印机械移到下一行的行首。对有些打印机而言也许需要特殊的端口设

置。

form feed: 将打印机械移到下一页的开头处。

carriage return: 将打印机械移至本行的行首。

先进的激光打印机已经没有一个“Carriage”，所以 Carriage Return 这一概念就已不是字面上意思，而是表示类似的操作。如果打印机能处理 simple 类型的文件，则在增加打印机并指定打印机能处理的文件类型时应把 simple 加上。如果你不希望打印机接受 simple 类型的文件，就必须另外给出一个打印机能接受的内容类型清单。

另外一个内容类型是 terminfo，这不是指某一类型文件，而是指在 terminfo 数据库中所有类型。这并不是说某一打印机能够处理在数据库中列举的所有类型文件，这一名字是用来描述可能有的滤通程序 (filter) 的能力的。同样，内容类型 any 也是用以描述某一 filter 所能接受或产生的文件类型的。这两个名字在增加打印机时不能用作内容类型。

应按如下命令指定内容类型：

```
/usr/lib/Lpadmin -P printername -I Content-type-list
```

这里应在 printername 和 Content-type-list 位置分别填入打印机名及内容类型表。内容类型表是一串以逗号或空格相隔的类型名。

Sysadmsh 的用户应选择：

Printers→Configure→Content

如果你没有定义打印机能接受的文件类型，则打印服务假定类型为 simple 以及与打印机类型有同一名字的文件类型，当然，后者是在打印机类型定义过之后才有。

#### 2.2.4 连接方法

打印服务允许你用不同的方法连接打印机。最简单的办法是将打印机直接连接计算机上，你也可能希望通过网络或拨号调制解调器连接打印机，这样可以与其他计算机或工作站共享使用。在你将打印机连接到计算机上，或者接到网络，再由网络连接到计算机上之后，你应该为打印服务描述连接方法。

打印机连接到计算机的缺省方法是直接连接方法。如果你已用这种方法将打印机与计算机相连，那么你需要做的就只有一件事：给出连接端口名字。

实现非直接连接有两种方法：通过拨号调制解调器或者通过其他类型的网络。打印服务使用 UUCP 系统来处理这两种非直接连接方法。使用拨号输出调制解调器时，必须满足三个先决条件：打印机必须通过拨号调制解调器连接；拨号调制解调器必须连接到计算机上；UUCP 必须能识别该调制解调器。

对每台通过其他类型网络相连的打印机都需要给出一个“系统名”，这是指在 Systems (F) 文件或相关文件中一个条目的名字。虽然打印机不是一个 UNIX 系统，但仍然可以用 Systems 文件来记录访问方式。(当然，不会给出注册信息)。

由于 cu 程序访问打印机的方法与打印服务访问打印机的方法完全一样，所以在建立文件时必须保证 cu 可以访问打印机。cu 命令不是用来访问打印机的，但它可以在建立文件时充当尺度：如果 cu 能访问打印机，那么打印服务也一定能访问。

要增加一台通过调制解调器或网络相连的打印机，应输入以下命令：

```
/usr/lib/LPadmin -P printer-name -U dial-info
```

其中 printer-name 处应填入打印机名字, dial-info 或者是可以拨号到达打印机的调制解调器的电话号码, 或者是在 UUCP Systems 文件中为打印机填入的系统名称。

Sysadmsh 的用户应选择:

```
Printers >Configure >Add
```

在输入 LPadmin 命令时可带有 -U 或 -V 选项, 并且, 除非用 -L 任选项, 打印服务将认定打印机不是用作注册终端的。

在使用拨号输出或网络打印机时应注意: 如果打印机或者端口处于“忙”状态, 则打印服务会在一会儿之后再次请求, 如果是打印机忙, 则 10 分钟后再试, 如果是端口忙, 则 20 分钟之后再试, 这个再试的频率是不可调整的。不过, 你可以给打印机发 enable 命令, 强迫立即再试, 如果端口或者打印机将在一段时间内保持忙状态, 你应该发 disable 命令, 不必强制立即再试了。

如果拨号不成功, 则 lpstat 命令会报告失败的原因。另外, 如果你被提醒发生拨号错, 则此提醒信息会指出错误的原因。

### 2.2.5 字符集或打印字型轮

不同的打印机打印的字体风格不同。有的打印机有可变化的打印字型轮, 有的有预编程控制的、可选择的字符集。系统管理员可以通过自己的工作, 使打印机之间的差别对打印服务用户的影响减到最低限度。

在增加打印机时, 你应指定这台打印机的打印字型轮和字符集。每台打印机的这两个值都应是唯一的。在列举打印字型轮或字符集时, 你应指定它们的名字, 起的名字应兼顾到你和用户都方便使用。因为不同的打印机也许有类似的打印字型轮或字符集, 所以你应该使用所有计算机共同的名字, 这么做使得用户可以将文件交付打印并要求用某种字体, 而不用考虑会用到哪一台打印机以及是否使用了打印字型轮或可选择字符集。

如果打印机配有可安装的打印字型轮, 你只需列出其名字即可; 如果打印机有可选择字符集, 就需要列出它们的名字, 并将每个字符集与 terminfo 数据库中唯一地标识它的名字或数字对应起来, 你可以用如下命令来确定在 terminfo 数据库中列举的字符集的名字:

```
TERM = printer-type tput csnm 0
```

printer-type 是打印机的类型名, 这时就打印出 0 号字符集 (在打印机初始化之后通过缺省设置得到的字符集) 的名字。重复上述命令, 用 1, 2, 3, 4, .....代替 0 的位置, 即可看到其余字符集的名字。一般来讲, terminfo 数据库中的名字应该与用户文献中用到的打印机名字相匹配。不过, 不是所有的制造厂商都使用同一个名字, 所以不同打印机类型的 terminfo 名字也可能不同。

---

注: 要使打印服务找到 terminfo 数据库中的名字, 你必须指定打印机类型。

---

在增加打印机时, 为指定打印字型轮名称表, 应输入如下命令:

```
/usr/lib/Lpadmin -P printername -S print-wheel-list
```

print-wheel-list 是由逗号和空格分隔开的名称表。如果你用空格来将名字隔开，就应当用引号将整个表（但不包括-S）括起来。

要指定字符集名字表并将每个名字与 terminfo 数据库中的名字或数字对应起来，应输入以下命令：

```
/usr/lib/Lpadmin -P printername -S character-set-list
```

character-set-list 也是用逗号或空格隔开的名字表，但表中的每一项类似：

```
csN = character-setname
```

```
character-setname1 = character-setname2
```

上面第一种情况中的 N 是从 0 至 63 的一个数，表示该字符集在 terminfo 数据库中的编号。第二种情况中的 character-setname<sub>1</sub> 以 terminfo 名字来标识该字符集。不管是哪一种形式，等号 (=) 右边的名字都是你选用的该字符集的别名。

Sysadmsh 用户应选择：

Printers → Comfigure → Parameters

来做上述两件工作。

---

注：如果对字符集来讲 terminfo 名字就够用了，你就不必另外指定一个别名。在指定了别名之后就可以用数字、terminfo 名以及指定的别名来表示字符集。

---

假定你的计算机有两个可选择字符集（#1 和 #2）以及标准字符集（#0），打印机类型是 5310，你应输入以下命令来决定可选择字符集的名字：

```
TERM = 5310 tput csnm1
```

```
english
```

```
TERM = 5310 tput csnm2
```

```
finnish
```

english 和 finnish 这两个词分别是两个命令的输出，也就是可选择字符集的名字。这时你感到表示字符集 #2 用名字“finnish”就足以说明了，但标准字符集及 1 号字符集（#1）需要用一个更贴切的名字，于是你输入以下命令来定义别名：

```
/usr/lib/Lpadmin -P printername -S“Cs0 = american, english = british”
```

如果你没有列举出打印机可用的打印字型轮或字符集，则打印服务假定：带打印字型轮的打印机只有一个单一的、固定的打印字型轮，人们在使用打印机时不能要求特殊的打印字型轮。同样，带可选择字符集的打印机认定任何 csN 名字或者 terminfo 名字能为计算机所识别。

如果你的打印机带有可变换的打印字型轮，而且在安装时已经列举出来，用户就能发使用其中之一请求。不过，在打印字型轮安装好之前，这些请求留在队列中并不执行。当然，你可以周期性地监视请求某一打印字型轮的队列中的打印请求数，看是否超过了某一限度，再决定是否安装打印字型轮，打印服务为此提供了一种较为简单的办法，你可以要求系统在请求一个打印字型轮的请求数目超过某一限度时提醒你。

可以选择其中一种接收提醒信息的方法：

- 可以通过电子邮件收到提醒信息

- 无论你在哪个终端上注册，提醒信息就写到该终端上，以此接收提醒信息
- 可以通过选择的程序接收提醒信息
- 可以要求不用提醒

如果选择不要提醒的方式，则自己应负责检查某一打印字型轮是否安装。

在选择提醒方式之后，还可以设置在提醒你之前在队列中等待的请求的数目，你可以安排在打印字型轮安装之前每隔几分钟提醒一次，也可以选择重复提醒的频度，或者选择每个打印字型轮只提醒一次。

在需要安装打印字型轮时安排提醒的方式，应输入下面命令中的一个：

```
/usr/lib/Lpadmin -S print-wheelname -A mail -Q integer -W minutes
/usr/lib/Lpadmin -S print-wheelname -A write -Q integer -W minutes
/usr/lib/Lpadmin -S print-wheelname -A 'command' -Q integer -W minutes
/usr/lib/Lpadmin -S print-wheelname -A none
```

头两个命令分别指示打印服务给你发邮件信息或直接将信息发到你注册的终端上。第三个命令指示打印服务在每次提醒时运行“command”命令，在输入第三个命令时当前发挥作用的 shell 环境必须存起来，在‘command’执行完之后加以恢复，shell 环境包括环境变量、用户和用户组标识符、以及当前目录。上面的第四个命令指示打印服务在打印字型轮需要安装时不要给你发提醒信息。

integer 是必须等待打印字型轮的请求数，minutes 是两次提醒中间的间隔分钟数。

注：如果你想把邮件或信息送到另一个人那里，你应选择上面所列的第三个命令。应选用 -A 'mail user-name' 或者 -A 'write user-name' 选项。

在收到多次连续的重复提醒信息后，你应指示打印服务不可再为当前情况向你发提醒信息，可输入下面的命令：

```
/usr/lib/Lpadmin -S print-wheelname -A quiet
```

为完成上面列举的各条命令，Sysadmsh 的用户应选择：Printers → Auxiliary → Alert

在打印字型轮安装之后又拆卸下来时，应重新开始请求数计数，若超过某一限度就再次发出提醒信息。如果等待某一个打印字型轮的请求数降到 -Q 限度之下又再次升上来超过 -Q 限度，提醒会再次发出，这种情况在某些等待请求被删除之后或提醒的类型变化时可能发生。

如果 print-wheelname 是“all”（在上述所有命令中），则提醒条件适用于所有定义了提醒的打印。

如果你没有对某一打印字型轮定义提醒方式，那你就不会收到它的提醒信息。

## 2.2.6 错误提醒

打印服务提供了检测打印机错误并提醒你注意的功能。错误从比较小的，比如缺纸，色带断，到较严重的错误，如电源不正常，或打印机故障，都可能发生。指示错误的方法也很多，从打印机联机信号掉，到发送 XOFF，或者错误信息。只有两类打印机错误指示方法能为打印服务识别，一种是联机信号掉，另一种是发出 XOFF 之后在指定的时间

内没有发 XON 信息。当然，你自己可以增加滤通程序，以检查出所有的打印机错误指示信息，并依靠打印服务自身给你发提醒信息。

你可以选择下列方法之一来接收打印机错误警告信息：

- 可以通过电子邮件接收警告信息；
- 可以通过你当前注册的终端收到提醒；
- 可以通过你选择的程序接收提醒信息；
- 可以指定不收提醒信息。

---

注：如果你选择不收提醒信息方式，自己必须要有办法发现错误并加以解决，打印机有错时，打印服务无法继续下去。

---

关于提醒的方法，你也可以安排在错误清除之前每隔几分钟重复警告（提醒）一次，频度可由你设定，也可以选择每个错误只提醒一次的方法。

---

注：如果没有能够提供较好的错误检测手段的滤通程序，打印服务只有再试着打印一次文件才能确定发现的错误是否已被改正，如果这一次打印成功了，即认为错误已被清除。在此之前，如果你选择了一个错误只提醒一次的方式，你便不会得到另一次提醒信息。这样，如果你修复了一个故障之后，但打印服务尚未作再次打印的尝试之前，打印机再次发生故障（也许是另一种故障），你就得不到任何提醒，如果你修复打印机故障的努力没有成功，你也无从知道，（没有提醒信息）。由此看来，每一个错误重复接收提醒信息的方法更为适用。另外，要求人工重新使能打印机的方法也能解决上述问题，这将在“错误恢复”部分讲到。

---

要安排接收打印机错误提醒信息的方法，应输入以下命令之一：

```
/usr/lib/Lpadmin -P printername -A mail -W minutes  
/usr/lib/Lpadmin -P printername -A write -W minutes  
/usr/lib/Lpadmin -P printername -A 'command' -W minutes  
/usr/lib/Lpadmin -P printername -A none
```

头两条命令分别指示打印服务为每次故障给你发一次邮件信息或者直接写信息到你的注册终端上。第三条命令指示打印服务为每次故障都运行'command'命令。为执行这一命令，当前的 shell 环境必须加以保存，在命令执行完毕之后再作恢复。该环境包括环境变量、用户和用户组标识符以及当前目录。minutes 是重复提醒时两次之间间隔的分钟数。第四个命令要求打印服务在检测出故障时不要提醒你。

---

注：如果希望在故障发生时给别人发邮件或给他的终端写信息，应用第三条命令，任选项如下：

```
-A 'mail username' 或者 -A 'write username'
```

---

如果错误发生后你不断收到关于此故障的重复性提醒信息，你可以指示打印服务不要再就此故障向你发提醒信息。方法是输入：

```
/usr/lib/Lpadmin -P printername -A quiet
```

如果在上述所有命令中的 printername 是“all”则发提醒条件适合所有打印机。

### 2.2.7 错误恢复

在发现打印机故障并提醒你之后，你可能会很快清除该故障，使打印机又可正常打印

了。这时，打印服务以下面三种方法中的一种进行恢复：

- 从打印停止那一页的首部开始继续打印
- 重新执行在故障出现时的“当前”打印请求
- 等候你告知打印服务重新启用打印机

---

注：从打印停止那一页的首部开始重新打印的实现方法对处理程序（滤通器）的要求非常高，这样一个处理程序要求对打印机的控制过程有非常具体的了解，才能将打印内容与每一页边缘对应起来，要求知道打印停止时文件停在哪一页边缘。打印服务缺省的处理程序不能做到这一点，如果没有使用这一类的处理滤通器，你会收到提醒信息，告诉你不能如你所愿。

---

要指示在故障排除之后打印服务应采取的恢复方式，输入下列命令之一：

```
/usr/lib/LPadmin -P printername -F Continue
```

```
/usr/lib/LPadmin -P printername -F beginning
```

```
/usr/lib/LPadmin -P printername -F wait
```

这些命令分别指示打印服务从打印停止那一页的首部开始继续打印、重新执行在故障发生时正在执行的打印请求及等候你输入 enable 命令重新启用打印机。

如果你没有指定在打印机错误之后打印服务的恢复方式，则打印服务试图从打印停止那一页的首部开始继续执行，如果没能做到，就重新执行打印请求。

如果恢复方式是 continue，但接口程序没有保持运行以致不能确定打印机故障是何时排除的，所以在打印机工作正常之前，打印服务每隔几分钟就试着打印一次，以判断故障是否已经排除，你可以发一条 enable 命令，强制使打印服务立即再试打印一次，以便判明自己所做的故障清理工作是否已成功。

### 2.2.8 限制用户使用打印机

你可以规定只有一部分人可以在你的计算机上使用打印机。比如，你的打印机是用来打印高精度信息的，而只有一部分人可以打印高精度信息，或者你的打印机特别昂贵，费用因素不允许所有人都使用这台打印机，在这样的情况下，打印服务就用允许使用打印机或不允许使用打印机的用户名单来限制打印机的使用。打印服务会拒绝用户在他或她未被授权使用的打印机上打印文件的请求。

列举允许或不允许使用某一打印机的用户名单的方法与列举允许或不允许访问 cron 和 at 设备的用户名单的方法很相似。简单说来，应遵循如下几条规则：

- 允许名单包含允许使用打印机的用户，而不允许名单包含不允许使用打印机的用户名。
- 如果允许名单不空，则不用不允许名单；反之则使用不允许名单。如果两个名单均为空，则对谁能使用打印机没有限制。
- 将 any 或 all 放入允许名单，则允许所有人用打印机。将 any 或 all 放入不允许名单，则禁止所有人用打印机，当然用户 Lp 和超级用户除外。

你可以用下面的命令对两个名单增加用户名：

```
/usr/lib/Lpadmin -p printername -u allow:user -list
```



```
/usr/lib/Lpadmin p printername -u deny:user-list
```

user-list 是用逗号或空格隔开的一个用户名单，如果你用空格将两个用户名分隔开，则应用引号将整个名单（包括 allow: 或 deny:, 但不包括 -u）包括起来。头一个命令将名字加到允许名单中并从不允许名单中移去，后一个命令将名字加入不允许名单并从允许名单中删掉。使用 allow: all 使所有人均可用打印机，deny: all 将拒绝所有用户使用打印机，用户 Lp 和超级用户例外。

如果你在两个名单中都不加用户名，打印服务就认为所有人均可使用打印机。

### 2.2.9 缺省打印属性

在用户提交一个打印文件请求时，页的大小、字符间距，行间距一般是由所打印的格式决定。如果用户不需要用打印格式，就应给出页的大小以及字符间距行间距参数。如果用户既不给出使用的格式，也不给出页面大小以及字符间距、行间距值，就使用缺省值。

你可以为每台打印机设置缺省值，最好给不同的打印机设置不同的缺省页面大小以及缺省字符间距、行间距值，这样需要不同打印效果的用户只用将请求发到缺省值满足要求的打印机上就可以了。比如，你可以将一台打印机设置为宽打印输出（132 列），另一台设置为正常打印输出（80 列，66 行），再将一台设置为每英寸 12 个字符，每英寸 8 行的输出方式。

你可以独立地指示四个缺省值设置：页宽、页长、字符间距和行间距。具体值可以通过调整以满足需要。前两个参数可以列数和行数，英寸或厘米为单位给出，后两个参数以每英寸或每厘米的字符数和行数的形式给出。另外，字符间距的表示法还有 pica（表示每英寸 10 个字符即 10 个 cpi），elite（12 个 cpi），或者 Compressed（表示打印机所能提供的最大的 cpi 值。）（最多为 30cpi）

用一条或多条下面的命令设置缺省参数：

```
/usr/lib/Lpadmin p printername -o width = scaled-number
```

```
/usr/lib/Lpadmin p printername -o width = scaled-number
```

```
/usr/lib/Lpadmin p printername -o cpi = scaled-number
```

```
/usr/lib/Lpadmin p printername -o lpi = scaled-number
```

给 scaled-number 值加上 i 表示英寸，或者加上字母 C 表示厘米，在 cpi 或行间距 Lpi 上加上字母“i”就多余了。在字符间距处你也可给出 pica,elite, 或 compressed 来代替数值。

如果你没有提供缺省值、页的大小，间距值就是打印机初始化时设定的值。你可以首先定义打印机的配置，却不提供自己的缺省值，然后用 Lpstat 程序来显示打印机配置情况，以此来查明缺省参数值。下面的命令

```
Lpstat -p printername -L
```

能给出缺省页面大小以及字符间距、行间距值。如果你没有提供缺省值，则报告的参数是根据打印机在 terminfo 数据库中的条目得出来的。很显然，这要求你已在打印机配置信息中给出了打印机类型信息。

## 2.3 用户命令综述

下面的表格给出打印服务的三条命令：

表 2.2 常用用户命令表

命令	解 释
cancel (C)	删除打印文件的请求
Lp (C)	将一个或多个文件送打印机
Lpstat (C)	报告打印服务的状态

在给打印服务系统发请求，检查请求的状态及删除请求等命令之外，用户还有使打印机有效和无效的能力，即当用户发现打印机有某种不正常时，立即请系统管理人员关闭打印机似乎没有必要，但如果允许普通用户在你个人的打印环境下使打印机无效又不太合理；这时你可以控制其余用户是否有权访问下表中的两个命令。

表 2.3 disable 与 enable 命令

命令	解 释
disable (C)	使指定的打印机无效
enable (C)	使能指定打印机（即使之有效）

## 2.4 管理命令综述

下面表中给出管理人员可用的命令集：

表 2.4 管理命令表

命令	解 释
/usr/lib/accept	允许作业请求进入队列等待送往指定目的地。
/usr/lib/reject	禁止作业进入等待送往指定目的地的队列。
/usr/lib/lpadmin	建立或改变打印机配置。
/usr/lib/lpfilter	建立或改变处理程序（滤通器）定义。
/usr/lib/lpforms	建立或改变预打印格式。应输入 /usr/lib/lpadmin 命令安装格式。
/usr/lib/lpmove	将输出请求从一个目的地移向另一个目的地。
/usr/lib/lpsched	启动打印服务。
/usr/lib/lpshut	停止打印服务。
/usr/lib/lpusers	设置或改变打印服务的用户的打印请求优先级以及优先级限制。

这些命令在 /usr/lib 目录中可以找到。如果你要频繁地用到它们，比较方便的办法是将 /usr/lib 目录包括进 PATH 变量中。为使用管理命令，你必须作为超级用户注册或者拥有 Lp 权限（关于权限的解释见“管理用户帐号”一章的内容）。

注意，除了上述命令之外，你可能还会用到使打印机有效/无效命令 enable (C) 和 disable (C) 以及前一节中讲到的用户命令。

上表中列出命令在 Sysadmsh 环境下都可实现：Sysadmsh → Printers  
这种方法以菜单实现，比上述打印服务命令的复杂语法要容易多了。

## 2.5 增加本地打印机

操作系统还支持使用连接在普通串行终端后部 AUX 或者 PRINT 端口上的本地打印机。这些打印机通过标准 RS-232 端口相连，能极大地减少共享系统资源的负载。lprint (C) 命令就是用来在本地打印机上打印文件的，但要使该命令能够工作，必须正确地配置终端。要在终端后部的 PRINT 端口连接打印机用作本地打印，应按下面的过程进行：

1. 用标准 RS-232 电缆将本地串行打印机与你终端上的 AUX 端口相连，至少应将第 2, 3, 7 针相连。保证打印连机的电源已经接好，并且处于联机状态。

2. 在终端上登录注册到 UNIX 系统，保证终端的工作状态正常。

3. 保证你的终端上的 AUX 端口配置与打印机的设置完全一样。（波特率，奇偶校验，数据位，xon/xoff 等等）

4. 要使 Lprint 命令工作，Lprint 需要知道对每个特定终端如何启动和停止本地打印。Lprint 在文件 /etc/termcap 中查找两个终端属性：PN（启动打印）和 PS（停止打印），它们是必须送往终端以控制本地打印的控制字符串。有些终端在它们的 termcap 条目中定义了这些属性。用一个文本编辑器（比如 Vi (C)）来检查 /etc/termcap 文件。查找你的终端对应的条目，例如，若你的终端是 Wyse，则你应查找‘wyse60’。

Wyse60 的 termcap 条目如下表

表 2.5 Wyse60 的 termcap 条目表

```
w71wy60lwyse60lWyse WY-60 with 80 column / 24 line screen in wy60 mode:\n:is = \\E\\072\\Ee (\\Eo\\Ee6\\Ec41\\E4\\Ec21\\Ed / : \n\n:if = /usr/lib/tabset/std:pt:\n\n:G1 = \\EH3:G2 = \\EH2:G3 = \\EH1:G4 = \\EH5:GD = \\EH0:GG#0:GH = \\EH\\072:\n\n:GU = \\EH=:GV = \\EH6:GR = \\EH4:GL = \\EH9:GC = \\EH8:GF = \\EH7:\n\n:PU = \\EJ:PD = \\EK:\n\n:al = \\EE:am:bs:bt = \\EI:cd = \\EY:ce = \\ET:cl = \\E+:\n\n:cm = \\Ea § i § dR § dC:co#80:dc = \\EW:d1 = \\ER:ei = \\Er:im = \\Eq:k0 = AI\\r:\n\n:k1 = A@\\r:k2 = AA\\r:k3 = AB\\r:k4 = AC\\r:k5 = AD\\r:k6 = \\AE\\r:k7 = AF\\r:\n\n:k8 = AG\\r:k9 = AH\\r:kd = J:kh = :kl = H:kr = L:ku = K:\n\n:li#24:mi:nd = L:se = = \\EG0:so = \\EG4:sg#0:ug#0:ue = \\EG0:ul:up = K:us = \\EG 8:\n\n:PN = \\Ed#:PS = ^ T:hs:ts = \\Ez(:fs = ^ M:\n\nw81wy60w1wyse60w1Wyse WY-60 with 132 colomn / 24 line screen in wy60 mode:\n:is = \\E\\073\\Ee (\\Eo\\Ee6\\Ec41\\E4\\Ec21: \n\n:if = /usr/lib/tabset/std:pt:\n\n:G1 = \\EH3:G2 = \\EH2:G3 = \\EH1:G4 = \\EH5:GD = \\EH0:GG#0:GH = \\EH\\072:\n\n:GU = \\EH=:GV = \\EH6:GR = \\EH4:GL = \\EH9:GC = \\EH8:GF = \\EH7:\n\n:PU = \\EJ:PD = \\EK:\n
```

Wyse60 定义了 PN 和 PS 属性。对于其它终端，你必须在 /etc/termcap 条目中为你的终端增加一行，该行包含 PS 和 PN 两个属性，具有如下形式：

```
:PN = start sequence:PS = stop sequence:\n
```

5. 查阅你的终端手册，找出用来打开和关闭辅助端口 AUX 的控制字符串。有时把这称为“Passthrough”或者“transparent”方式。举一个用 AUX 端口的控制字符串的例子。打开 Wyse60 终端上端口的代码为：

<ESC>d#

关闭该端口的代码为：<CTL>t

6. 某些特殊键在填入 termcap 文件之前必须转化为 termcat 格式。下表给出了 termcap 用来表示这些特殊键的符号

表 2.6 特殊键表

特殊键	termcap 代码
ESCAPE	\E
CTRL-X	\X(X 表示任意字符)
NEWLENE	\n
RETURN	\r
TAB	\t
BACKSPACE	\b
FORMFEED	\f

前面讲过，启动和停止打印的 termcap 属性是 PN 和 PS。用上面的表格可将 Wyse60 的控制代码串 <ESC>d# (PN) 和 <CTL>t (PS) 表示为：

:PN = \Ed#;PS = ^T:\

7. 对没有这样的条目的终端，你必须将上面的一行内容加以修改填入 termcap 条目。(注意不要加作第一行和最后一行)。修改时应查终端手册，找到打开和关闭辅助端口 AUX 的正确控制字符串，并用第 6 步中给出的表加以转化，得到的串填入对应你终端的条目。

有些终端（比如 Wyse60）包括一种“透明”方式，即数据在打印时不在屏幕上显示的方式。（这是由上例中 PN 串选择的方式）。

---

注：在修改 /etc/termcap 文件时必须作为超级用户注册。我们推荐你采用将本文件拷贝成其它名字再进行编辑的办法，这样可防止误操作破坏 /etc/termcap 文件。当然，如果该文件损坏了，你也可用 custom (ADM) 从存储卷中再次将该文件分离提取出来。

---

8. 在增加了 PN 和 PS 条目之后，退出注册，然后重新注册使新的 termcap 条目生效。

9. 用下面命令打印文件 filename:

lprint filename

在本地打印进行时不要敲键盘。在打印过程中你不能在终端上做任何其他事情。

10. 如果文件在屏幕上显示出来而不是在打印机上输出，则建立的 PS 和 PN 条目也许有问题。用正确的代码修改这两个条目，如果文件仍不能在打印机上打印出来，就试一试将连接终端 AUX 端口和打印机的电缆的发送数据针和接收数据针交叉互连的方法。

(这是被称为“nullmodem”的连接方法。)

## 2.6 启动和停止打印服务

在正常操作下，你当然不必手工启动或停止打印服务，每次系统进入多用户方式时打印服务会自动启动。如果你需要停止打印服务而不想同时停止操作系统，你可以按照下面讲述的方法做到这一点。

停止打印服务使得所有的打印工作在几秒钟之内完全停下来。任何没有执行完的打印请求在打印服务重新启动之后完全从头开始重新执行。打印机配置格式以及起作用的处理程序（滤通器）在打印服务停止后又重新启动时会自动恢复。

---

注意：要想手工停止和启动打印服务，必须作为超级用户或者具有 LP 权限的用户注册。

---

### 2.6.1 手工停止打印服务

要手工停止打印服务，应输入以下命令：

```
/usr/lib/lpshut
```

(Sysadmsh 用户选择，Printers→Schedule→Stop)

屏幕显示以下信息：

```
Print Services Stopped
```

所有的打印工作在几秒钟之内停止。如果你在打印服务没有运行时想停止它，你将看到：

```
Print services already stopped
```

### 2.6.2 手工启动打印服务

要手工启动打印服务，应输入如下命令：

```
/usr/lib/lpsched
```

(Sysadmsh 用户应选择：Printers→Schedule→Begin)

这时屏幕显示：

```
Print service started
```

在开始打印之前重新建立起停止打印服务时保存的打印请求所需的配置、格式以及处理程序要花去一到二分钟时间。如果你试图重新启动已经处于运行状态的打印服务，你会看到如下屏幕提示信息：

```
Print services already active
```

---

注意：如果你只想改变打印机配置或者增加格式或处理程序，则没有必要停止打印服务。

---

## 2.7 删除打印请求

要删除某一打印输出的请求，应用 cancel (C) 命令。当你请求打印时，系统会为你的作业 (job) 显示一个请求标识符 (ID)。比如，若你为一台名为“laser”的打印机发送

一个作业，则 UNIX 系统显示请求标识符形式如下：

```
request id is Laser-number
```

这里 number 是给你的作业安排的编号。在开始打印之前如想撤销这一作业，应用如下命令：

```
cancel laser-number
```

sysadmsh 的用户应选择：

```
Printers→Request→Cancel
```

这时打印请求就被撤消了。

大多数的系统打印非常快，所以 Cancel 命令必须尽快发出才能起到作用。

## 2.8 使打印机有效 / 无效

enable 命令允许 Lpsched 在打印机上打印文件。在接受 accept 命令之后打印机可以接受打印请求。但要想打印文件，还必须用到 enable 命令。

例如，要使一台名为“daisy”的打印机有效，应输入如下命令：

```
enable daisy
```

sysadmsh 的用户应选择：

```
Printers→Schedule→Enable
```

用 disable 命令可以使打印机无效。如果打印机处于无效的状态，则调度程序 Lpsched 不管它有没有得到 accept 命令，都不会给它发送打印请求。disable 命令的 -r 任选项允许你给用户发一条消息以解释使某一打印机无效的原因。

比如，要使一台走纸不正常的名为“laser”的打印机无效，应该输入：

```
disable -r "paper jam" laser
```

则用命令 Lpstat -P laser 请求使用打印机“laser”的用户会收到这样的消息：

```
printer laser disabled since Dec 5 10:15
```

```
paper jam
```

## 2.9 在一组打印机中增加一台

将若干台打印机视为一“组”的做法常常会带来许多方便，其好处在于：用户可以将待打印的文件交付给一组打印机，打印服务将作业交给该组中第一台处于空闲的打印机执行。这样做打印机周转较快，提高了每台打印机的使用效率。

如果只是想让用户通过 Lp-T type 命令交付一个文件并指定打印机类型，则不需要用到打印机组。让第一个满足条件的打印机执行该打印任务就可以了。

打印机组的一个用途是将应以某种顺序使用的打印机归入一个组中。如果你有一台高速打印机和一台慢速打印机，你当然希望让高速打印机来处理尽可能多的打印请求而将慢速打印机留作慢速打印机处于“忙”状态时使用。因为打印服务总是以打印机加到组里的顺序来查找可用的打印机并将任务交给它，所以你可以使高速打印机在慢速打印机之前加入到组中，则打印服务会以你希望的顺序来分配打印请求。

用以下命令将打印机加入到组中：

```
/usr/lib/Lpamin -P printername -C classname
```

Sysadmsh 的用户应选择:

Printers → Configure → Modify

如果打印机组 classname 尚不存在, 则会建立起来。使用 Lpstat-C 可以得到所有打印机组及其打印机的清单。

---

注: 组名和打印机名都必须唯一, 这样用户在根据名字为打印请求指定执行者时才不必分辨是一组打印机还是单一打印机。所以, 不能让打印机组和打印机有相同的名字。

---

如果你不把打印机加到某一组中, 它就不属于任何一个组。

## 2.10 设置系统缺省打印执行者

在用户没有明确指定某一打印机或打印机组来打印文件并且没有设置变量 LPDEST 时, 可以由你来加以定义。当然, 定义的打印机或打印机组首先必须已经存在。

要使某一打印机或打印机组成为缺省的打印执行者, 应输入如下命令:

```
/usr/lib/Lpadmin -d printername 或者 classname
```

如果在这之后不再想用缺省打印执行者定义了, 可用如下命令输入“空”打印机名或组名:

```
/usr/lib/Lpadmin -d
```

对以上两条命令, Sysadmsh 用户均可选择:

Pfprinters → Configure → Default

如果没有设置缺省打印执行者, 当然也就没有该缺省值, 用户就必须在打印请求中明确指定打印机名或组名, 或者用打印执行者的名字来设置 shell 变量 LPDEST。

对 C-shell 来讲, 命令为:

```
Setenv LPDEST printer
```

对 Bourne shell 而言; 命令为:

```
.....
```

```
LPDEST = printer; export LPDEST
```

用户还可以把这两个命令分别放入自己的 .login 和 .profile 文件中。

## 2.11 安装表格或打印字型轮

在打印服务开始打印需要预打印表格或打印字型轮的文件时, 你必须将其安装在打印机上。(打印字型轮用在老式的、所谓“菊花轮”撞击式打印机上, 是四周有打印字符的小轮)。如果对表格或打印字型轮设置了提醒方式, 则当有足够多的打印请求排在队列中要求用表格或打印字型轮时, 就必须将其安装在打印机上。

在安装表格时, 你也许希望看表格是否排列好了。如果表格设置了排列 (alignment) 模式, 你可以要求对表格进行重复的试打印, 直到你将打印调整好, 使排列方式符合要求为止。

安装表格或打印字型轮时首先要将其装载到打印机上, 然后告诉打印服务, 表格或打印字型轮已安装完毕。但是后一条对正在打印的打印机来说极难做到, 因为打印服务要继

续在该打印机上打印不需要用到表格的文件。你可能不得不首先用 `disable` 命令使打印机无效，然后才得进行安装。步骤如下：

1. 用 `disable` 命令使打印机无效。
2. 按本节后面讲到的方法安装新的表格或打印字型轮。
3. 用 `enable` 命令重新使打印机有效。

在将新的表格或者打印字型轮装载进打印机之后，应输入下面的命令告诉打印服务进行安装：（命令应在一行内输入）

```
/usr/lib/lpadmin -P printername -M -S print-wheelname  
-f formname -a -o filebreak
```

Sysadmsh 的用户应选择：

Printers > Auxiliary > PPforms > Configure

如果你只是要安装表格，则在命令行中不用加 `-S print-wheelname`，如果你只想安装打印字型轮，则在命令行中去掉 `-f formname -a -o filebreak` 部分。

如果你是在安装表格，则在表格排列模式的每一复制件打印之前都会要求你按回车键。每一模式打印完之后，你都应对打印机进行调整，然后再按回车键，直到你对打印的表格模式满意为止。如果没有安排排列模式，系统就不会要求你按键。如果你不想调整排列模式，那么在命令行中就可以不加 `-a` 和 `-o filebreak` 任选项。

`-o filebreak` 任选项指示打印服务在排列模式的每一复制件打印完之后增加一个表格更替控制符 `form feed`，该控制符实际用到的控制序列由具体的打印机决定，从 `terminfo` 数据库中获取。如果排列模式已经包含了表格更替符 `form feed`，则命令行中不用 `-o filebreak` 选项。

如果你想将表格或打印字型轮拆卸下来，应使用下面命令：

```
/usr/lib/lpadmin -P printername -M -S none -f none
```

Sysadmsh 的用户应选择：

Printers > Auxiliary > PPforms > Remove

如果只想卸下表格，则应在命令行中去掉 `-S none` 部分。若只想卸下打印字型轮，则在命令行中去掉 `-S none` 部分。

在打印机没有安装表格以前，只有不用到表格的打印请求才会送到该打印机。类似地，只有打印机上安装了打印字型轮之后才能将需要用到特定打印字型轮的打印请求送至打印机。

## 2.12 移去打印机或打印机组

如果某一打印机或打印机组没有挂起等候其服务的打印请求，则可将其移去。如果有，则必须先用 `lpmove` 命令将这些打印请求移到另外一台打印机或另一打印机组上去，然后才能将其移去。

如果移去的打印机是某一打印机组中仅有的一台，则打印机组在该打印机移走之后也自动地不复存在了。不过，将某一打印机组移去并不会导致其成员打印机移走。如果移走的打印机或打印机组同时也是系统缺省的打印执行者，则系统将不再有缺省打印执行者了，除非另外加以设置。



要将打印机或打印机组移去，应输入命令：

```
/usr/lib/lpadmin -x printername 或 classname
```

Sysadmsh 的用户应选择：

Printers→Configure→Remove

如果你希望将某一打印机从其所属的打印机组中移去，但并不想删除该打印机，则应使用如下的命令：

```
/usr/lib/lpadmin -P printername -r classname
```

Sysadmsh 的用户应选择：

Printers→Configure→Modify

## 2.13 管理打印负载

有时你也许希望让某一打印机停止接受打印请求，或者将打印请求从一台打印机移到另外一台上。这么做的原因可能有多种，比如：

- 打印机需要周期性维护
- 打印机坏了
- 打印机移去了
- 配置发生改变，打印机要以不同方式使用
- 某一打印机有太多大型打印请求排在队列中等待其服务，而别的打印机处于空闲状态，需要平均分担打印请求。

如果你打算对打印机的使用方式作大的改动，比如停止其处理某种表格的能力，改变其可用的打印字型轮或者禁止部分用户使用它，这时必须将排在队列中等待其服务的打印请求移走或者删掉。打印服务将力求寻找别的打印机来实施打印请求，但必须在用户不在乎用哪一台打印机打印的条件下才行。这样的请求不会自动移走。如果你不首先将它们移走，则打印服务会将其删除。

如果你决定让某一打印机停止使用并且将改变它在配置中的存在，或者本身负荷太重（等候其服务的打印请求太多），你可以将打印请求从它上面移走，并且拒绝让它接受新的打印请求，这可以用 `lpmove` 和 `reject` 命令来实现。如果你已拒绝让一台打印机接受打印请求，以后还可以用 `accept` 命令来再让它接受打印请求。

### 2.13.1 让打印机或打印机组拒绝打印请求

让某一打印机或打印机组拒绝接收新的打印请求，应输入以下命令：

```
/usr/lib/reject -r“reason”printername 或 classname
```

Sysadmsh 的用户应选择：

Printers→Schedule→Reject

在一条命令中你可以让多台打印机或打印机组拒绝打印请求，办法是在同一命令行中列出它们的名字，名字间以空格相隔。无论什么时候，只要有用户企图在该打印机上打印文件，命令行中所列出的拒绝打印请求的原因（reason）就会显示于用户屏幕上。如果你不想对用户解释原因，可以将命令行中的“reason”删除。

虽然 `reject` 命令使打印机拒绝接受新的打印请求，但并未将已经排在队列中等待打印

机服务的打印请求移或删除。如果打印机有效，这些请求就会继续得到执行。

### 2.13.2 让打印机或打印机组接受打印请求

accept 命令允许打印机组接受由 LP 命令发出的打印请求。在对打印机进行正确的配置之后就可允许打印机接受请求了。

在引起打印机拒绝请求的条件改变之后，可输入以下命令使打印机开始接受新的打印请求：

```
/usr/lib/accept printernamet 或者 classname
```

Sysadmsh 的用户应选择

Printers→Schedule→Accept

同样，你可以在一条命令中使多台打印机或打印机组接受打印请求，办法也是将其名字列在命令行中。由于打印服务不能自动地使新增打印机或打印机组接受打印请求，所以在每次增加打印机之后，都必须记住用 accept 命令使其能够接受打印请求。

### 2.13.3 将请求移至另一台打印机

如果你要将打印请求从一台打印机移到另一台上，应输入下列命令之一：

```
/usr/lib/lpmove request-id printername
```

```
/usr/lib/lpmove printername1printername2
```

在第一条命令中你可以在打印机名前给出多个打印请求标识符，该命令将列出的打印请求移到给出名字的打印机上。第二个命令将等待打印机 1 服务的队列中的所有打印请求全部送至打印机 2。使用了后一条命令之后，打印服务再也不为打印机 1 安排打印请求了。（与使用 reject 命令产生的效果完全一样）。

### 2.13.4 几个例子

下面给如何使用 lpmove, accept 以及 reject 命令的几个例子。

例 1：你决定更换打印机 LP1 上的色带并进行某些预防性维护，你就得把等待 LP1 服务的所有打印请求送至打印机 LP2。在这一工作完成之后，打印机 LP1 再也不能接受打印请求了。

```
/usr/lib/lpmove lp1 lp2
```

这时你应使打印机 LP1 无效，并开始做你想做的维护工作。完成之后，应使打印机重新开始服务：

```
/usr/lib/accept lp1
```

如果你已使打印机无效，应该重新使之有效。

例 2：你注意到有用户提交了好几个大型文件，将打印请求排入队列等候打印机 laser1 服务，而同时打印机 laser2 处于空闲状态，因为没有打印请求等候其服务。你可以将两个最大的文件的打印请求，laser1-23 和 laser1-46，移至 laser2 上，并且使 laser1 拒绝接受新的打印请求。

```
/usr/lib/lpmove laser1-23 laser1-46 laser2
```

```
/usr/lib/reject -r"too busy---will reopen laser"laser1
```

例 3: 你希望打印服务不要将打印请求分配给打印机, 因为它正在修理:

```
/usr/lib/reject -r "printer4 needs repair" printer4
```

请求在 printer4 上打印文件的用户收到如下信息:

```
UX:Lp:ERROR:Requests for destination"printer4"aren't being accepted
```

```
To fix:Use the"lpstat -a"command to see why this destination is not accepting requests
```

为查明指定打印机的接收状态, 应输入:

```
Lpstat -a
```

输出形式类似:

```
printername accepting requests since datetime
```

```
printer4 not accepting requests since datetime
```

```
printer4 needs repair
```

其中: date time 是当时的日期及时间。

## 2.14 管理队列优先级

打印服务提供了一种简单的优先级控制机制, 人们可以用它来判断打印请求在队列中的级别。每一打印请求都可由提交该请求的人给予一个优先级别。这是一个从 0 至 39 的数, 数值越小表明优先级越高。优先级较高 (即优先级别数值较小) 的请求在队列中排在优先级较低 (数值大) 的请求前面。

这样, 如果你觉得你的打印请求优先级太低, 那么在你将文件交付打印时可以设置一个较高的优先级 (优先级别数值较小), 同理, 如果你觉得你的打印请求优先级太高, 也可以设置一个较低的优先级。(数值较大)

如果对用户最高优先级不加控制, 这样一个简单的优先级方案是不可能正常工作的。对此方案你可以定义以下特性:

- 对每个用户安排优先级极限, 即用户提交的打印请求的优先级不应超过该极限, 也就是说用户打印请求的优先级别数值必须不小于某一个值。当然, 要大于该值是可以的。
- 对没有指定优先级极限的用户给予一个优先级极限缺省值。
- 可以设置优先级缺省值。即对用户没有指定优先级的打印请求给出一个优先级缺省值。

根据你的需要设置类似上面的一些特性, 你就可以防止低优先级打印任务 (比如大多数工作人员提交的日常打印任务) 干扰高优先级打印任务 (比如由会计人员提交的工资单打印任务) 的正常进行。

你也许会发现需要让某一关键的打印请求在别的请求之前进行, 有时甚至需要中断正在进行的打印而为该请求服务。这时你可以让打印服务对某一打印请求进行“立刻” (immediate) 的处理, 而让另一打印请求可以“保持”, 这可以让重要的打印服务得到立即执行, 而将正在进行的打印工作延迟到重要打印完成之后再继续进行。

Lpusers 命令让你可以为用户指定优先级极限, 以及设置优先级缺省值。此外, 你可以用命令 Lp-i request-id -H hold 和 lp -i request-id -H immediate 分别使请求处于“保持”状态和使请求处于立即打印的优先级别。

### 2.14.1 设置优先级极限

设置用户的优先级极限，应输入命令：

```
/usr/lib/lpusers -q priority-level -u username
```

用上述命令可以设置一组用户的优先级，方法是在-u 任选项之后列出用户的名字，两个名字之间用逗号或空格相隔。（如果用空格，则应将用户名表用引号括起来）。priority-level 是一个从 0 至 39 的数。如前所述：该数越小，则优先级越高，反之则优先级越低。

如果你想给所有其他用户设置缺省极限值，则应输入命令：

```
/usr/lib/lpusers -d priority-level
```

Sysadmsh 的用户选择：

Printers→priorities→Default

该命令设置了缺省的极限值，此缺省值适用于所有未用前面讲的 Lpusers 命令给出其极限值的用户。

如果在此之后你决定给某一用户一个不同于其他用户的优先级极限。可用前面讲过的方法给其设置一个单独的极限值。如果你决定让某个带有单独优先级极限值的用户将其极限值改为缺省优先级，可输入下面的命令：

```
/usr/lib/lpusers -u username
```

同样，这条命令也可使多个用户做同样的工作，只要在-u 之后列出用户名清单就可以了。

Sysadmsh 的用户应选择：

printers→priorities→Remove

如果你没有设置缺省的极限制，则不带单独的极限值的用户极限值范围从 20 到 39。

### 2.14.2 设置缺省优先级

缺省优先级是用于在没有给出优先级的打印请求上的。设置缺省优先级应用以下命令：

```
/usr/lib/lpusers -q priority-level
```

Sysadmsh 的用户应选择：

Printers→Priorities→Highest

注意不要将缺省优先级与优先级的缺省极限值混淆起来。前者适用于没有给出优先级的用户，而在你没有给用户指定优先级极限值时才会用到后者——后者是用来限制用户给出过高的优先级的。

---

注：如果给出的缺省优先级数值小于用户的优先级极限值即缺省优先级太高时，应以极限值代替缺省优先级数值提供给打印请求。

---

如果没有设置缺省优先级，则打印服务认为该值为 20。

### 2.14.3 检查优先级极限值和缺省值

输入下列命令即可以检查你设置的所有优先级极限值和缺省值：

/usr/lib/lpusers -L  
Sysadmsh 的用户应选择:  
Printers→Priorities→List

#### 2.14.4 在队列中移动请求

在用户提交打印请求之后,可以在一定限度内在队列中对它进行移动。比如可以:

- 将请求设置为“保持”状态,这样其他打印请求可以越过它进行打印。
- 将请求移至队更首部,进行“立刻”打印。

这些用普通的 LP 用户命令即可实现。分别描述如下:

改变请求的优先级:

还在队列中等待打印的打印请求可以被指定一个新的优先级,这会改变请求在队列中的位置,使之位于优先低于它的请求之前或在优先级不低于(高于或等于)它的请求之后。

输入下列命令可改变请求的优先级:

```
lp -i request-id -q new-priority-level
```

用此命令一次只能改变一个请求的优先级,如果某一请求已经在执行,你就不能改变它的优先级了。

使某一请求“保持”(hold)

任何还未完成打印的请求都可置为“保持”状态。如果请求正在执行,你可以停止正在进行的打印,并且使请求“挂起”,直到你恢复请求打印才会继续进行。不过其他用户不能将你置为“保持”状态的打印请求恢复。

输入下面命令可使请求“保持”:

```
lp -i request-id -H hold
```

而下面命令可以恢复请求:

```
lp -i request-id -H resume
```

请求恢复之后,又回到队列中,可以执行打印。如果某一请求正在打印时你将它置为“保持”状态,则恢复后成为下一个执行的请求,一般来讲打印从第一页开始进行,但你可以让它从稍后的页开始打印:

```
lp -i request-id -H resume -P starting-page-
```

如果要指定的是开始打印的页以及各页,则必须用到 starting-page 之后的短横线“-”。

---

注:要想对一部分页进行打印需要有能够处理这一任务的处理程序(滤通器)存在才行。打印服务所用的缺省滤通器不能完成这一任务,所以如果没有使用满足条件的处理程序,跳过某些页进行打印的请求就会遭到拒绝。

---

将打印请求移至队首:

你可以将某一打印请求移至队首,使之成为下一个执行的请求。如果该请求必须马上执行,而另一个打印请求又正在打印,你可以用前面讲到的办法使它处于“保持”状态。

输入下列命令可将打印请求移到队首:

```
lp -i request-id -H immediate
```

只有系统管理员可以这样移动请求, 普通用户不能使用-H immediate 任选项。

---

注: 如果你将不止一个的请求设置为“立刻”打印方式, 则这些请求逆序进行打印。即最晚移到队首的请求最早执行。

---

### 2.14.5 检查打印机配置

在定义完打印机配置之后, 你可能还想检查一下, 看配置是否正确。如果检查过程中发现了错误, 可运行相关的命令程序 (或 sysadmsh 选择) 进行修改。没有必要对整个配置都重新定义。

使用 Lpstat 命令可检查配置信息以及打印机当前状态。这条命令的短格式给出打印机状态, 你可用它来看打印机是否存在以及是否忙、空闲或者无效等。命令的-L 格式给出完整的配置信息。输入下列命令之一检查打印机:

```
lpstat -p pr entername
```

```
lpstat -p printername -l
```

后一条命令即是长格式。不管用哪条命令, 你都会看到类似下面的信息:

```
printer 打印机名 now printing 请求标识符 enabled
```

```
since 当前日期
```

```
printer 打印机名 is idle.enable since 日期
```

```
printer 打印机名 disabled since date.
```

```
原因.....
```

```
printer 打印机名 waiting for auto-retry.
```

```
原因.....
```

上面的 auto-retry 信息表示打印服务在使用该打印机时不成功 (信息中列出了原因), 打印服务以后将再次试用。

如果用长格式命令, 你可能看到输出还包括以下内容:

Form mounted:表格名, content types:内容类型名, printer type:打印机类型名, Description:注释名, Connection:连接情况,

Interface:路径名, On fault:提醒方式, After-default:错误恢复方式,

Users allowed:用户名单,

Forms allowed:表格清单, Banner required (标题信息);

Character sets:字符集清单, Default pitch:行间距、字符集清单,

Default page size:宽值、长值, Default port settings:串行端口清单,

### 2.15 表格与处理程序

预打印表格是你可以在打印机上打印的空白表格。典型的应用方式是, 应用程序产生文件, 在空表格上打印, 将空的表格填满。打印服务提供了建立和管理表格的设施。打印服

务还支持处理程序（滤通器）的使用。它用来完成以下三件工作：

- 将用户文件转化成可以在指定打印机上正确打印的数据流。
- 处理不同的打印方式，这些方式是用户以 Lp 命令的 -y 任选项的方式提出请求的。
- 检测打印机故障并通知打印服务，再由打印服务提醒用户或管理人员。

打印服务虽然支持表格和滤通处理程序，但实际上必须用到它们的情况是很少的。

## 2.16 使用请求登记中的信息

在目录 /usr/spool/lp/temp 和 /usr/spool/lp/requests 中包含着对提交给打印服务的请求进行描述的文件。每个请求有两个文件，各在一个目录中，每个文件都包含请求的有关信息。在 /usr/spool/lp/temp 目录中的请求文件只有提交该请求的用户才能进行修改，而在 /usr/spool/lp/requests 目录中的请求文件任何人——包括提交请求的用户——都无权修改。由此可以看出 /usr/spool/lp/requests 目录的请求文件更加安全，所以关于请求的重要信息大多放在该目录中。

只要请求还在队列中，请求文件就会一直留在这两个目录中。一旦请求执行完毕，这两个目录中请求文件的信息就会合并起来，一起加到文件 /usr/spool/lp/logs/requests 中。

请求登记的结构很简单，使用普通的 UNIX 命令即可很容易地将所需信息提取出来。打印请求在请求登记中的排列顺序与请求执行的顺序一致，请求之间由给出请求标识符的一行内容隔开，该行以下的每一行都以一个单字母为标志，表明本行所包含信息的种类。每个字母与本行数据之间以一个空格相隔，下面列举出登记文件代码：

表 2.7 行内容表

字母	本行内容
=	本行是间隔行,包含请求标识符、用户的用户标识符和组标识符,原始文件(即未经滤通程序处理的文件)的字节数,请求进入队列的时间。以上各项以逗号隔开,顺序如上所述。用户标识符,组标识符,以及文件大小分别以 uib,gib 和 size 为前导。
C	打印文件份数
D	实施请求的打印机或打印机组或者“any”
F	打印的文件名,对每个待打印的文件重复本行一次,文件以给定的顺序进行打印
f	所用表格名
H	写出用到的特殊操作种类,在该行中唯一有用的值是 immediate,表示使请求立即执行。
N	在打印请求成功地结束之后所用的提醒方式,如果以邮件通知用户则本行应为 M,如果以往用户终端写信息的方式通知用户,则本行值为 W,
O	-O 任选项
P	打印请求的优先级
p	打印页清单
r	如果用户要求对文件进行直接处理,则出现本行。(即 LP 命令的 -r 任选项)
S	用到的字符集或打印字型轮

s	以十六进制数形式给出的单独位的组合，用作请求执行的返回值。其中有些位由假脱机程序内部使用，最重要的几个值列举如下： 0×0004 慢速滤通（处理）成功完成 0×0010 打印成功完成 0×0040 请求被删除 0×0100 请求在滤通（处理）或打印时失败
T	放在标题处的内容
t	文件中发现的内容类型
U	提交打印请求的用户名
x	请求用的慢速滤通（处理）
Y	给予用于打印请求的滤通器（处理程序）的特殊方式清单。
y	请求用的快速滤通（处理）
z	请求所用的打印机，这与以D标志的行的内容含意不一样，因为请求可能在可以由任何打印机服务的请求队列中或者D行内容为一组打印机或者本打印请求已被打印服务管理人员移到另外一台打印机上，在以上几种情况下都有必要列出实际执行打印的打印机。

## 2.17 调整打印服务

虽然打印服务方式力求灵活，以处理大多数的打印机和打印需要，但实际上是不可能做到面面俱到的。本节讲述怎样才能：

- 调整打印机端口特性
- 调整 terminfo 数据库
- 写接口程序。

下图给出了打印请求执行的整个过程：

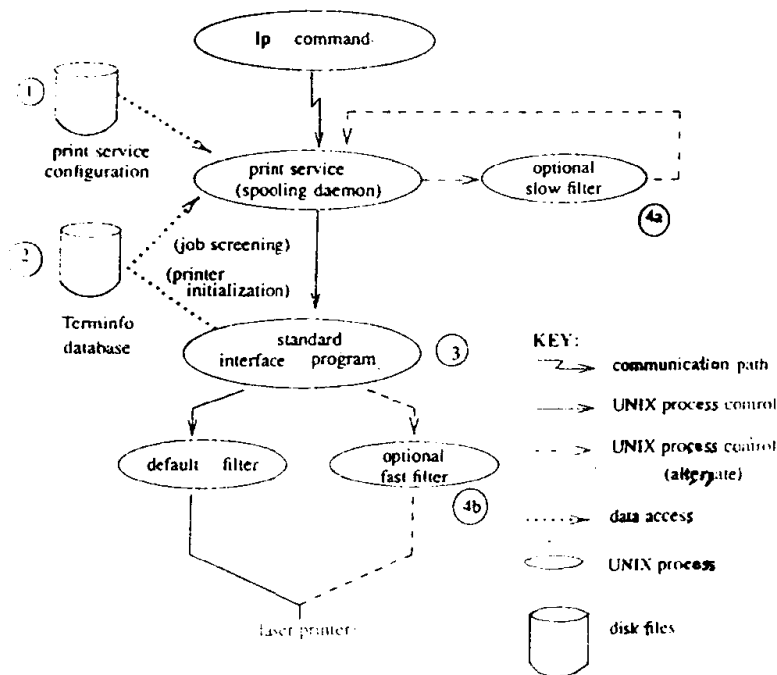


图 2.2 打印服务执行打印请求的过程



每个打印请求都被送往一个假脱机控制程序（后台程序），该程序跟踪所有的打印请求。在你启动打印服务时即建立该程序。UNIX 的系统进程同时负责跟踪打印机和慢速滤通器（处理程序）的状态，当打印机打印完一个用户文件之后，如果队列中还有等待服务的请求，则控制程序就使打印机开始执行其中一个请求。

你可以对图 2.2 中的某些部分（图中标注以圆圈者）进行调整或更换，以达到调整打印服务的目的。

1. 对大多数打印机而言，你只需要改变存于盘上的打印机配置信息就可以了。部分独立于打印机的配置数据为打印机端口特性：波特率，奇偶校验，等等。

2. 对 terminfo 数据库中没有表示出来的打印机，你可以在 terminfo 数据库中增加一个条目来描述该打印机的特性。该数据库用于两个平行的方面：审查打印请求，保证接受下的请求可以得到所要求的打印机的服务；设置打印机，使之准备好可以进行打印。

比如，如果 terminfo 数据库中不知道某台打印机可以按用户要求设置页长，假脱机控制程序就会拒绝这类请求，另一方面，如果 terminfo 数据库显示打印机有某种能力，则接口程序就会用该信息来对打印机进行初始化。

3. 如果你想对打印机增加某些打印服务的功能，可以改变标准接口程序。这个程序是负责管理打印机的。打印标题页，初始化打印机，启动滤通程序，将用户文件的拷贝送往打印机等工作都由该程序完成。

图中 4a、b 要建立你的系统上的应用程序与打印机之间的连接，可以增加慢速和快速滤通器。两种滤通器都将文件转化为另一种格式，比如将一种控制字符串的集合转化为另一种，并能通过对用户提出请求的打印方式的解释来提供特殊的启动动作。

### 2.17.1 调整打印机端口特性

你应该保证打印服务设置的打印机端口特性与打印机通信设置相匹配。标准的打印机端口设置可以用于典型的 UNIX 文件以及许多打印机上，但并不是适用于所有的文件和打印机。实际上我们所说的调整打印服务工作不应包括设置打印机端口特性，因为打印服务的标准特性本身就允许你对每台打印机的端口设置加以说明。

直接连接到计算机上的以及通过某种网络连接到计算机上的打印机需要由接口程序来为其设置打印机端口特性。这些特性定义了与打印机之间的低层通信。其中包括波特率，XON/XOFF 流程控制的使用，每个字节 7 位、8 位或者其它位数、奇偶校验方式，等。标准接口程序用 stty 命令来初始化打印机端口，将波特率设为最低，用缺省值设置其它特性。

标准接口程序所用缺省特性列举如下：

缺省值	含义
9600	9600波特率
CS8	8位字节
-cstopb	每个字节1位停止位
-parenb	不产生奇偶校验
ixon	使能XON/XOFF流程控制

-ixany	只允许XON来重新启动输出
opost	post-进程数据流列举如下
-oluc	不将小写转为大写
onlcr	将换行对应为回车 / 换行
-ocrnl	不要将回车对应为换行
-nocr	在包括第0列在内的各列上输出回车（如果有回车符的话）
nl0	换行之后不延迟
cr0	回车之后延迟
tab0	在tab之后不延迟
bs0	在backspaces之后不延迟
vt0	在纵向tabs之后不延迟
ffo	在formfeed之后不延迟

当增加新的打印机时，可以另外列出在打印用户文件时要用到的端口参数的清单。可以在缺省特性清单之后给出这张清单，这样对那些已在缺省特性清单中列出而又不想加以修改的条目就不必列在清单中了。可以如下的形式指示该清单：

```
/usr/lib/lpadmin -p printer-name -o "stty = 'stty-option-list'"
```

Sysadmsh 的用户应选择：

Printers > Configure > Parameters

注意：如果在 stty-option-list 中给出的条目不止一条，则必须同时用到双引号和单引号，如果你打算另外增加或改变打印机端口特性，就可使用上表中给出的缺省清单。

例如，假设你的打印机将用来打印图形数据，则换行字符 linefeed 即应单独输出而不用增加回车。你应输入命令：

```
/usr/lib/lpadmin -p printer-name -o "stty = -onlcr"
```

注意，因为例中只列出一个条目，所以单引号可以省去。

再举一例。假设你的打印机要求对发给它的数据进行奇校验，则应输入如下命令：

```
/usr/lib/lpadmin -p printer-name -o "stty = 'parenb parodd CST'"
```

### 2.17.2 调整 terminfo 数据库

打印服务依据标准接口和 terminfo 数据库对打印机进行初始化，设置页大小、字符间距、行间距以及字符集。所以，要在打印服务中增加一台新打印机，需要在 terminfo 数据库中有相应的正确的条目。在 terminfo 数据库中对每台打印机都用一简短名字来标识，这类名字与用来设置 TERM shell 变量的那一类名字是一致的。比如，AT&T455 打印机标识为 455。

如果你没能找到你打印机对应的 terminfo 条目，就应该加上一个。如果不这么做，可能也可以使用该打印机，但你就不能获得页大小、间距以及字符集的自动选择值，而且在根据不同的打印请求为打印机设置正确的方式时会遇到问题。如果不想变更 terminfo 条目，另有一个替代方法是调整打印机使用的接口程序。下一节描述如何做到这一点。

对每个终端或打印机，都有数以百计的项目可以在 terminfo 数据库中进行定义。但是打印服务只能用到不到五十条，而且大多数打印机用得还要少得多。下表列举出将新打

印机加到打印服务中时需要在 terminfo 数据库中加以定义的、与打印机有关的各条目：

表 2.8 条目含意表

terminfo 条目	含意
布尔量:	
daisy	打印机需要操作符来改变字符信 (Y/N)
数字量:	
bufsz:	打印前缓冲的字节数
cols	一行中的列数
lines	一页中的行数
orc	水平方向每个字符点单位数目
orhi	水平方向每英寸点的数目
orl	垂直方向每行点的数目
orvi	垂直方向每英寸点数目
cps	平均打印速率, 单位为字符/秒
字符串:	
cr	回车
cpi	改变每英寸的字符数
lpi	改变每英寸的行数
chr	改变水平分辨率
cvr	改变垂直分辨率
csnm	字符集名清单
mgc	清除所有边缘 (顶部、底部、边)
hpa	绝对水平位置
cudl	向下一行
cuf1	打印头向右移动1列
swidm	使能双面宽打印
rwidm	使双面宽打印无效
ff	退页
is1	打印机初始化串
is2	打印机初始化串
is3	打印机初始化串
if	初始化文件名
iprogr	初始化程序路径名
cud	使打印头下移若干行
cuf	使打印头右移若干列
rep	重复一个字符若干次
vpa	绝对垂直位置

scs	选择字符集
smgb	在当前行设置底部空白边
smgbp	设置底部空白边
smgl	在当前列设置左空白边
smglp	设置左空白边
smgr	在当前列设置右空白边
smgrp	设置右空白边
smgt	在当前行设置顶部空白边
smgtp	设置顶部空白边
scsd	开始字符集定义
ht	移到8位空格的tab停顿处

---

在完成了新的条目之后，应用 tic 程序将其编译到数据库中。只要输入以下命令即可：

```
tie filename
```

其中 filename 是包含你为新打印机建立的 terminfo 入口文件名。

---

注：在增加或删除了 terminfo 条目或者改变了控制字符间距、行间距、页大小以及字符集选择的值之后，都应停止打印服务，然后再重新启动，这样才能使新的 terminfo 有关信息生效。

---

### 2.17.3 如何写接口程序

如果你的打印机靠对 terminfo 数据库增加一个条目的办法已经不能支持，或者你的打印需求不能为标准接口程序所支持，这时你便可以设法构造你自己的接口程序。一般来讲从标准接口程序（或者在 /usr/spool/lp/model 目录中许多接口程序中选出一个）出发编制你的接口程序会容易一些。标准接口程序在 /usr/spool/lp/model/standard 目录中。

接口程序一般都完成以下几项任务：

- 如果需要，则初始化打印机端口。一般的接口程序使用 stty 命令来实现这一任务。
- 初始化打印机。一般的接口程序用 TERM shell 和 terminfo 变量获得控制字符序列来实现。
- 如有需求就打印标题页。
- 打印请求的份数。

接口程序不负责打开打印机端口，这是由打印服务完成的。打印机端口的连接方式作为标准输出送至接口程序。打印机被设置为接口程序的控制终端。这样，当端口挂起时就会将一个 SIGHUP 信号送至接口程序。

当打印服务发送一个输出请求到打印机时，该打印机的接口程序以如下方式启动：

```
/usr/spool/lp/admins/lp/interfaces/printer id user title copies options file1  
file2
```

接口程序的参数是:

printer: 打印机名。这是提供给接口程序的名字

id: 由 Lp 返回的请求标识符

user: 发出请求的用户的注册名

title: 由用户指定的可任选标题

copies: 用户请求的拷贝的数量

options: 一组以空格隔开的任选项, 由用户指定或者打印服务设置

file: 等待打印的文件的完整路径名

在接口程序启动后, 它的标准输入来自 /dev/null, 标准输出送往打印机端口, 其标准错误输出送往一个将要交给提交打印请求的用户文件。

标准接口识别下面各任选项:

nobanner: 该任选项用于跳过标题页的打印, 没有这一任选项, 就会打印出标题页。

nofilebreak: 该任选项用于跳过独立数据文件之间的空页。没有该任选项, 在打印请求的文件之间会有空页。

cpi = decimal-number<sub>1</sub>

lpi = decimal-number<sub>2</sub>

这两个任选项给出打印时每英寸的列数和行数的十进制表示。标准接口程序从 terminfo 数据库中分离出控制字符序列, 用来给出打印时字符间距与行间距的初始值。

pica、elite 和 compressed 等都可用在 decimal-number<sub>1</sub> 的位置, 分别表示 10 列/英寸, 12 列/英寸以及每英寸打印尽可能多的列。

length = decimal-number<sub>1</sub>

width = decimal-number<sub>2</sub>

这两个任选分别说明打印的长度和宽度。标准接口程序从 terminfo 数据库中分离出控制字符序列, 用来给出打印页的长度和宽度。

stty = 'stty-option-list' stty-option-list 用在 stty 命令参数的缺省值清单之后, 缺省值清单用来建立缺省的端口配置信息, 而本任选项清单 stty-option-list 提供给接口程序改变配置信息。

以上各任选项或者由用户在发出打印请求时指定, 或者由打印服务根据管理人员为打印机或者请求中用到的预打印表格指定的缺省值来确定。

附加的打印机配置信息以 shell 变量传送给接口程序:

TERM = printer-type

本 shell 变量指示打印机类型。该值被用作关键字来查找扩展 terminfo 数据库中的打印机能力方面的信息。

FILTER = 'pipeline'

本 shell 变量用来指示将请求内容送往打印机的滤通 (处理) 程序, 该程序控制打印机执行。

CHARSET = character-set

本 shell 变量指定在打印请求内容时所用到的字符集。标准接口程序从 terminfo 数据库中分离出控制序列，用于选择字符集。

经过调整的接口程序可以对这些任选项和 shell 变量不加使用，但如果使用了，就应该注意保持前后一致。

在调整接口程序时应注意，接口程序要设置正确的 stty 方式（终端特性，如波特率）。你可以仿照标准接口程序的实现方法来进行调整。参考以下面注释开始的接口程序中设置 stty 方式的部分：`###Initialize the printer port`

注意标准接口程序中使用的代码。它以类似下面一行的形式同时设置由打印服务或用户给出的缺省方式以及调整方式：

```
stty mode option 0 > &1
```

本行命令从打印机端口取得 stty 命令的输入。下面给出一个 stty 命令行的例子，该命令行设置波特率为 1200，并设置了一些任选方式：

```
stty parenb-parodd 1200 cs8 cread clocal ixon 0 < &1
```

有一个打印机端口特性接口程序没有设置，那就是硬件流程控制，该特性的设置取决于计算机的硬件，标准接口程序给出了用来设置硬件流控制及其他打印机端口特性代码的推荐使用值，在以下面注释行开头的部分：

```
##Here you may want to add other port initialization code
```

由于不同的打印机的字符列数不相同，应保证接口程序的相关内容具体的打印机相适应。标准接口程序以每页 80 列的形式打印标题，这一任务是由标准接口程序中以下面注释行开始的代码部分来实现的：

```
###Print the banner page
```

一般的接口程序应将用户有关的错误信息送到标准输出或者标准错误输出，送到标准错误输出的信息以邮件形式送给用户，而送至标准输出的信息用户在取输出时即可看到。

当打印结束时，你的接口程序应以一个代码退出，该代码说明打印作业的执行情况。对退出代码的含义解释说明如下：

0: 打印请求成功地完成

1 到 127: 在打印这个特定请求时遇到问题。（比如不可打印字符太多或者请求超越了打印机能力）这一问题不会影响以后的打印请求。打印服务通过 write (C) 或 mail (C) 通知提交该请求的用户。

128: 留作打印服务内部使用。接口程序不能以该代码退出。

129: 在执行打印请求时遇到打印机故障。这会影响到以后的打印。如果故障无法排除，则打印服务使打印机无效，等候管理人员修理，如果故障已排除，则不必使打印机无效，可以在几分钟皮后重新执行打印请求。

> 129: 129 以上代码留作打印服务内部使用，接口程序不能以此范围内的代码退出。

由上面的说明可以看出，在打印机故障发生时以代码 129 退出，打印服务在故障排除之后只能让请求重新从头打印，除此之外别无选择。还有一个办法可以提醒管理人员而

不用重新执行整个的请求，即让接口程序给打印服务发一条故障信息，然后等着故障排除。在故障排除之后，接口程序可恢复用户文件的打印。在打印完成时，可以 0 代码退出，就象没有出现过故障一样。另外，接口程序可以自动检测故障何时排除，这样就不必由管理人员来使能打印机使之有效了。

故障信息可用 Lp.tell 程序来送给打印服务。

有了特殊退出代码或者 lp.tell 程序，就不必于由接口程序来使打印机无效了。虽然你的接口程序可以直接使打印机无效，但这样做违反了故障提醒机制。只有在打印服务检测出故障之后才发出提醒，而主要的检测手段则是特殊退出代码和 lp.tell 程序。

如果打印服务在某一时刻必须中断文件的打印，可用信号“15”来“杀死”接口程序。如果接口程序被其他信号“杀死”，则打印服务就认为以后打印请求的执行不会受影响，从而继续使用打印机。打印服务通知提交打印请求的人：他或她的请求没能成功地执行完。

信号 SIGHUP, SIGINT, SIGQUIT, SIGPIPE (陷阱号 1, 2, 3, 13) 在接口启动时是被忽略的，标准接口改变这一点，认为收到这些信号即表明打印机出了问题，而在恰当的时候进入“陷阱”。这就是打印服务在每次打印文件时管理打印机的程序，它有四个主要任务：

- 初始化打印机端口
- 初始化打印机并设置字符间距，行间距，页的大小以及用户要求的字符集。
- 打印标题页
- 运行滤通程序打印文件

如果你没有选择接口程序，则使用标准接口程序。这对大多数打印请求来讲是足够了。如果你希望改变标准接口程序以满足你的要求或者重新编写自己的接口程序，必须在增加新的打印机时加以说明。

如果你打算用标准接口程序，则你就不必在增加打印机时另加说明了。如果你用了不同的接口程序，可以用其完整的路径名来指示它，还可以通过使用它作接口程序的另一台打印机来表示。

通过名字来标识经调整的接口程序，应以如下形式给出打印机名的接口程序路径名：

```
/usr/lib/lpadmin -p 打印机名-i 路径名
```

以使用某接口程序的打印机来标识该接口程序，可以用如下形式：

```
/usr/lib/lpadmin -p printername1 -e printername2
```

其中的 printername<sub>1</sub> 是增加的打印机名字，printername<sub>2</sub> 是已经存在的、使用经调整的接口程序的打印机的名字。

## 2.18 建立 RTS / CTS 协议的串行打印机

RS-232 串行接口的 RTS 线和 CTS 线本来是用作 DTE (数据终端设备) 设备与 DCE (数据通信设备) 设备之间的握手信号的。当 DTE 准备好向 DCE 发送数据时使 RTS 有效，当 DCE 准备好接收数据时，使 CTS 有效。如果 CTS 变成低电平，则 DTE 应停止发送数据，直到 CTS 变高为止。

UNIX 系统还将 RTS 线用作其他方面的握手之用。如果打印机发现自己的输入缓冲区快满了，它便使 CTS 线变成低电平，串行驱动程序即停止发送数据，等着打印机赶上

来。在打印机可以接收数据时，操作系统就使 CTS 线变成高电平。

许多打印机使用 DTR（数据终端准备好）线作握手信号而不用 RTS 或 CTS。

为建立 RTS / CTS 流程控制，应做以下工作：

1. 使用调制解调器控制端口（比如 / dev / tty1A）。若打算使用假脱机程序来访问打印机，则在要求输入设备名时应输入调制解调器控制端口而不是一个标准串行设备名。

2. 在用 RTS 和（或）CTS 线作握手线时，电缆的连接方式应如下图所示：

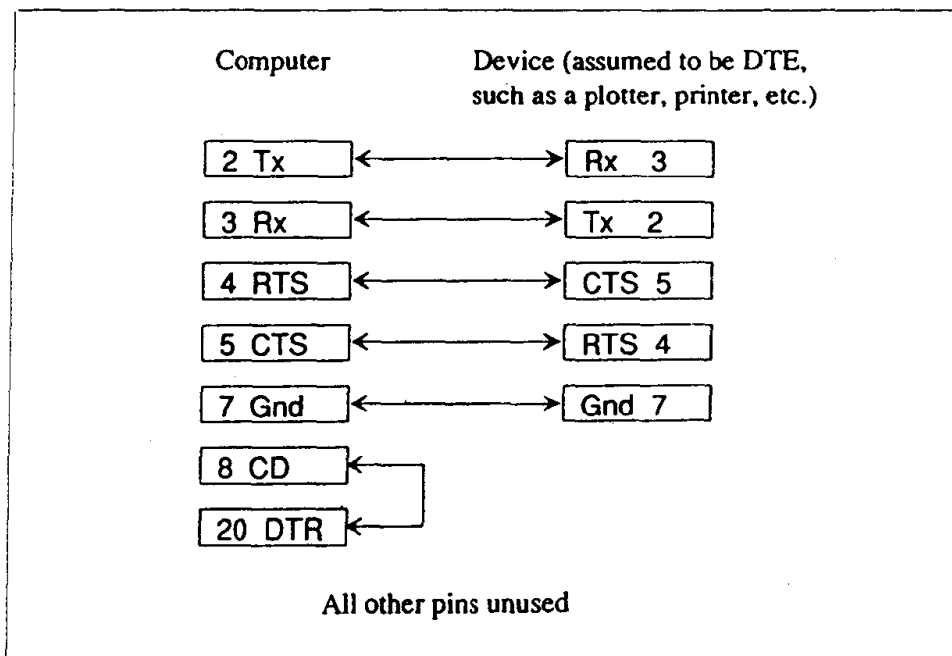


图 2.3 RTS / CTS 作握手线的连接方式

3. 如果设备使用 DTR 线作握手线，则电缆连接方式如下图所示：

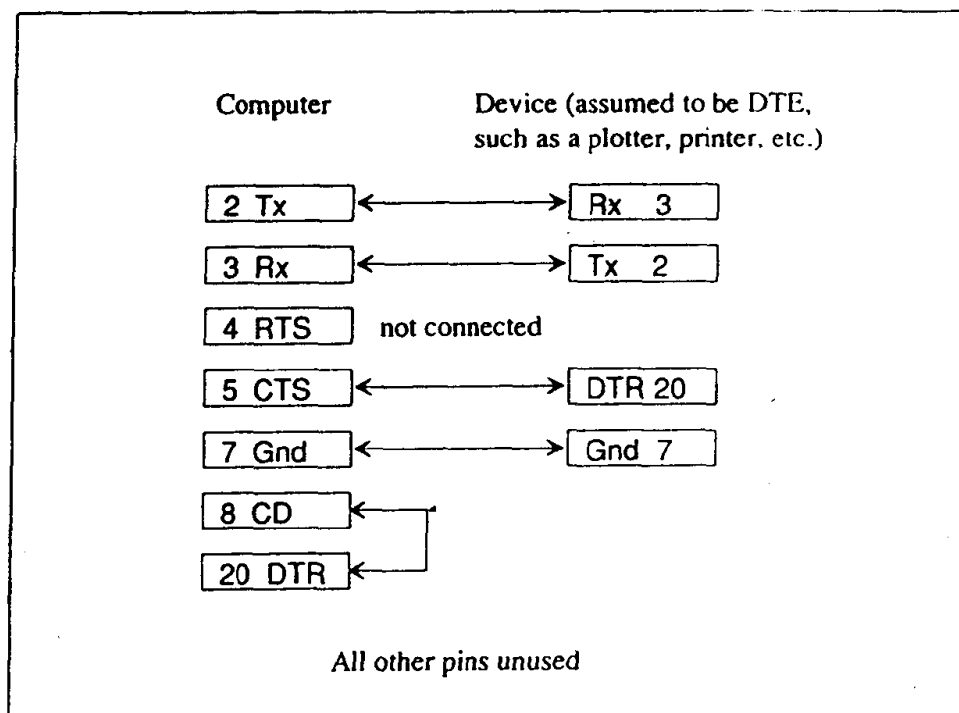


图 2.4 DTR 握手方式



## 2.19 使用不带假脱机程序的打印机

如果你使用不带假脱机程序的打印机，则指定给打印机使用的 stty 设置就无效了。应按本节中给出的指令执行。

首先作为超级用户登录注册，然后输入下面的命令或者把它们加入到初始化文件 /etc/rc2.d/s80lp 中。前一条命令用于串行打印机，后一条用于并行打印机。

```
(stty baud ixon ixoff -ixany; cat > /dev/null) < /dev/ttyn&
```

```
(stty baud onlcr; while; do sleep 3600; done) < /dev/lp1&
```

这里 baud 是打印机的波特率，tty 是串行设备名。本命令设置串行端口任选项，并保持端口的“打开”状态，用于没有假脱机程序的使用。

## 2.20 建立初始设备文件

标准并行打印设备 (/dev/lp, /dev/lp0, /dev/lp1 和 /dev/lp2) 在系统启动之后第一次打开设备时，要发出一个打印机初始化串 (init)。这只在第一次打开时才做，以便带有大的正文缓冲区的打印机在传送另一个文件时不致被刷空。

有些并行打印机每次接受一个文件，打印时都要求初始化，另外有一些打印机则是在打印机关闭后又重新打开（例如更换打印纸或色带的情况）时才要求执行初始化。在这种情况下，打印机关闭后再打开时打印机能够正常工作。

如果需要以比标准设备所能提供的次数更为频繁地对打印机进行初始化，则可以为用户的并行端口建立一个附加的设备文件，在需要初始化打印机时即可使用该设备文件。

其步骤如下：

1. 作为超级用户注册。

2. 确定正在使用的并行端口是哪一个设备，在本例中假定设备是主并行口 (/dev/lp0)。

3. 使用下面的 Sysadmsh 选择：

Printers → Configure → Modify

将一个并行初始化设备 (lp0i, lp1i, lp2i) 与打印机对应起来。

若打印机在关掉后又开时需要初始化，则在打印机打开之后使用下面的命令行，在第一个文件送到打印机之前初始化打印机（本例中假设是主并行口）：

```
> /dev/lp0i
```

如果打印机每传一个文件都要进行初始化且没有大的内部正文缓冲区，就可以始终使用设备 /dev/lp0i。

lp (C) 命令在每次向打印机传送一个文件时即发一次初始化字符串。

有的打印机没有换行 / 回车交换。如果打印机需要将换行对应为换车 / 回车，则应在用 Sysadmsh 设置打印机时，指定 Crnlmap 滤通程序。在提示输入打印机类型（哑打印机、Imagen 激光打印机 <并行接口>、Imagen 激光打印机 <串行接口> 或 other 时）应输入 other。接着会提示你输入接口程序的完整路径名。打印机滤通文件在目录 /usr/spool/Lp--/model/crnlmap 中。

## 第三章 使用软盘和磁带驱动器

计算机系统的重要功能之一是存储文件，并在需要时可以调入。可用的存储介质种类很多，软盘和磁带就是其中的两种。本章讲述如何在系统上安装和使用存储介质。你的系统上至少已安装好了一个软盘驱动器，本章描述在系统上增加磁带驱动器及使用软盘的指令。

### 3.1 使用磁带驱动器

卡式磁带驱动器是一种使用 1/4 英寸卡式磁带来存储数据的大容量存储设备。卡式磁带可以容纳好几倍于软盘的数据，因而对于大容量的备份操作非常有用。

#### 3.1.1 安装与配置

在设备连接到计算机上之后，以超级用户注册并输入如下命令：

```
mkdev tape
```

(Sysadmsh 的用户应选择：System→Hardware→Tape)

屏幕上显示出如下菜单：

```
Tape Drive Configuration Program
```

- 1.Install a Tape Drive
- 2.Remove a Tape Drive
- 3.Change default Tape Drive

Select an option or enter q to quit:

输入“1”选择“增加驱动器”一项。接着系统要求选择你所安装的磁带驱动器类型：

```
Tape Drive Installation Menu
```

- 1.Install Cartridge Tape Drive
- 2.Install Mini-Cartridge Tape Drive
- 3.Install QIC-40 Tape Drive
- 4.Install SCST Tape Drive

Select an option or enter q to return to the main menu:

下面讲述每种类型的驱动器的配置需求。注意在执行驱动器类型的有关指令之后应遵照“核心重连接”，“启动信息”两部分的要求执行。

卡式磁带：

选择“Cartridge Tape Driver”一项即是选择了 QZC-02-type 全尺寸卡式磁带驱动器。在安装 QIC 磁带驱动器之前应了解下面的技术性信息：

• 中断号。缺省中断设置在你的磁带驱动器控制卡上。如果该中断号与已经使用的某一中断号冲突，你必须改变卡上的设置。操作系统通常使用的中断号为 0，1，6，即使没有其他设备存在也是如此。如果你设置的中断号与缺省值不同，应记下你选择的设置，这

在运行 mkdev (ADM) 实用程序时会用到。

• DMA 通道与 I/O 基地址。在磁带驱动器控制卡上也有 DMA 通道和 I/O 基地址的缺省值设置。如果这些值与现有的硬件设备发生了冲突，就必须加以改变，记下你选择的设置，并在运行 mkdev 实用程序时加以说明。

如果你用了控制卡上的缺省设置，则应在第二张菜单输入“q”，使用缺省的磁带参数。

如果你改变了控制卡上的缺省设置，则在第二张菜单下选择“Modify Current Tape parameters”（修改当前磁带参数）一项。接着，会看到一张缺省磁带系数菜单，可对你想改变的参数加以修改。如果你改变 I/O 基地址，则应在地址值前面加上“0x”表示十六进制地址。如果未加“0x”，则系统认为地址值是十进制的。另外，如果在控制卡上选择 2 号中断，则在修改磁带参数时应指定中断 25，因为软件中断 25 对应于硬件的 2 号中断。其他的所有中断的软件中断号与硬件中断号都是完全一致的。

### Mini-Cartridge

小型磁带驱动器使用软磁盘驱动控制器，与标准的 QIC 磁带驱动器极不相同。这种部件即是所谓“floppy”或者“Irwin”磁带部件。它们是不可配置的，不需要输入参数。另外，小型磁带在使用之前必须进行格式化。不同类型的小型磁带的安装也略有不同。

应注意将驱动器的跳线设置正确。对于不同厂家的机器而言，正确的跳线设置也不相同。

### QIC-40

此类部件是特殊的小型卡式部件，其格式与上面讲的小型卡式磁带不同。本身也是不可配置的。

### SCSI

SCSI 磁带驱动器隶属于 SCSI 适配器。系统要求你提供设备 ID 号，驱动器附属的适配器号以及设备的逻辑部件号 (LUN)。如下图所示：

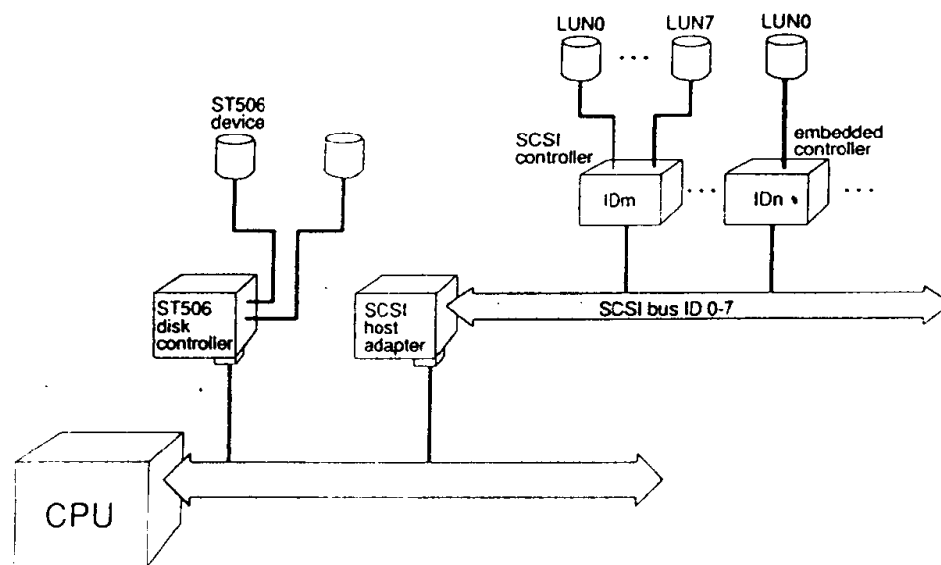


图 3.1 SCSI 配置图

由图示可以看出，SCSI 主适配器将信号从 CPU 总线转移到 SCSI 总线。SCSI 控制

器又称为 SCSI ID, SCSI 设备以逻辑部件号 (LUN) 来表示。

要配置 SCSI 磁带设备, 你必须知道:

- 主适配器上控制器的 ID 号 (0~7), 主适配器自己的 ID 号通常是 7, 这使得它在 SCSI 总线上具有最高的优先级。

- 主适配器号 (0 或 1)

因为磁带驱动器与其控制器是同一个部件 (即所谓嵌式), 所以逻辑部件号 (LUN) 是 0。mkdev tape 实用程序以下面的形式提示输入各值:

What is the ID of the controller for this device?

Select 0-7, or 'h' for help, or 'q' to quit:

Which SCSI host adapter supports this device?

Select 0 or 1, or 'h' for help, or 'q' to quit:

What is the LUN of this device?

Select 0-7, or 'h' for help, or 'q' to quit:

核心重连接:

在你选择了安装的驱动器并给出所需各参数值之后, 系统要求你允许进行操作系统核主的重连接。在重新启动之后, 即可使用磁带驱动部件。

启动信息:

在核心识别磁带驱动器且驱动程序已连接到操作系统核心之时, 在启动时就会显示信息以指示该设备的存在。使用 hwconfig (c) 命令也可使该信息显示出来。下表包含显示的信息, 指示出磁带的类型:

表 3.1 不同类型磁带的启动显示信息表

类型	启动显示信息				
QIC-02	%tape	—	—	—	type = w
mini	%Ctmini	—	—	—	type = ir
QZC-40	%Ctmini	—	—	—	type = qic 40
SCSI	%tape	—	—	—	type = S ha = 0 id = 2 lun = 0

### 3.1.2 编辑 / etc / default / tar 文件

在安装好磁带驱动器之后, 必须输入在文件 / etc / default / tar 中设置的正确的尺寸。在编辑此文件时, 对不同的设备会看到几种条目。下表给出了你的存储卷提供的 / etc / default / tar 文件:

表 3.2 / ect / default / tar 文件表

#device	block	Size	tape
archive 0 = / dev / rfd048d59	18	360	n
archive 1 = / dev / rfd148d59	18	360	n
archive 2 = / dev / rfd096ds15	10	1200	n
archive 3 = / dev / rfd196ds15	10	1200	n
archive 4 = / dev / rfd096ds9	18	720	n

(续表)

#device	block	Size	tape
archive 5 = / dev / rfd196ds9	18	720	n
archive 6 = / dev / rfd0135ds18	18	1440	n
archive 7 = / dev / rfd1135ds18	18	1440	n
archive 8 = / dev / rct0	20	0	y
archive 9 = / dev / rctmini	20	0	y
# The default device...			
archive = / dev / rfd096ds15	10	1200	n

### QIC 卡式驱动器

/ dev / rct0 条目访问 QIC 卡式磁带驱动器。磁带的尺寸如下表所示:

表 3.3 不同长度的磁带表

英 尺	条目中的“尺寸”域
300	30000
450	45000
600	60000

### 小型卡式磁带驱动器

在 / etc / default / tar 文件中找到 / dev / rctmini, 在上面给出的表中即 archive9。注意在上表中, size (大小) 域的值是“0”, 如果你打算使用缺省文件, 则在安装 rctmini 设备时必须改变该条目。对你的 rctmini 设备而言, 正确的大小值与所用磁带的尺寸有关。如下表所示:

表 3.4 不同尺寸的磁带表

磁带尺寸	实际容量	“size”域的值
10M 字节	8M 字节	8000
20M 字节	17M 字节	17000
40M 字节	35M 字节	35000
80M 字节	72M 字节	72000

实用程序 xbackup 和 xrestore 也有类似的文件和条目。

### 3.1.3 在磁带上保存文件

使用磁带驱动器的方法与软盘的使用方法极其相似, 只是前者的数据存储量要大得多。在存储或备份整个文件系统时使用磁带比较合适。推荐用户在存储程序时使用命令 tar (c)。该命令对文件的存放和传递非常有用。其它的程序, 比如 backup (ADM) 和 restore (ADM), 提供给系统管理人员在拷贝整个文件系统时使用。

cpio (c) 命令是一个通用的文档程序, 其使用格式与 tar 命令不一样。dd (c) 程序对非通用的文档进行传递和转换, 输入输出格式可以在命令行中加以指定。

## tar 命令

tar 命令对于建立整个目录的备份非常有用，命令句法为：

```
tar cvf devicefile files
```

其中 devicefile 是对应卡式驱动器的文件名。files 是要拷贝的文件或目录名。比如，为把目录 /u/bogart 中的所有文件都拷贝到卡式磁带驱动器 /dev/rct0 中，应打入：

```
tar cvf /dev/rct0 /u/bogart
```

(Sysadmsh 用户应选择：Media→Archive)

要恢复存储于磁带上的文件，应先插入含有要恢复的文件或目录的卡式磁带，然后输入以下命令：

```
tav xvf devicefile
```

(Sysadmsh 用户选择：Media→Archive)

即可将磁带上的所有文件恢复到原来的目录中。

### 3.1.4 磁带驱动器的维护

tape (c) 实用程序对所有的磁带驱动器执行多种不同的磁带维护操作。它向磁带驱动器发送命令并接收其状态。命令基本格式为：

```
tape command [device file]
```

例如，为了反绕一卡式磁带驱动器，应打入：

```
tape rewind
```

其它的命令包括：

earse: 抹除磁带内容，并固定好。

reset: 重置磁带控制器和磁带驱动器，清除错误环境，并将磁带返回至系统开启状态。

reten: 固定好卡式磁带。该命令应定期执行，以解决磁带过松的问题，否则会引起异乎寻常的大量磁带错，在极端情况下，会使磁带缠结到磁带驱动器中。

另外还有两条 Irwin 驱动器专用的命令：

info: 打印放入驱动器的磁带信息。包括是否格式化过，格式类型是什么以及写保护状态如何等。

capacity: 显示放入驱动器的磁带的容量，单位为块，每块 512 字节。

kapacity: 与 capacity 命令相似，不同点在于作为容量单位的块包含 1024 字节而不是 512 字节。

在执行了某些磁带操作之后，系统在磁带控制器完成其操作之前返回提示符状态，如果你在这时输入另外一条磁带命令，就会显示信息“device busy (设备忙)”，直到磁带设备完成上一次操作之后才能执行后一条命令。

应该清洗磁头，并固定好磁带，以免出现操作错误。

#### 磁带格式化

用于小型磁带驱动器 (ctmini) 的卡式磁带在使用之前必须先进行格式化。下面的命令可将一小型卡式磁带格式化：

```
tape format
```

关于该命令的更多情况，见 tape (HW) 和 tape (C) 的说明。

### 3.1.5 磁带驱动器错误校验码支持

系统还支持磁带错误校验码 (ECC)。ECC 磁带设备结点 `/dev/erct0` 是在运行 `mkdev tape` 命令时自动建立的。要使用 ECC，必须从设备本身，而不是通常的 `/dev/rct0` 进行读写。拥有支持容量大于 60M 字节的磁带驱动器的用户应该考虑对文件 `/etc/default/tar` 进行编辑，并用 `/dev/erct0` 取代通常的磁带设备。

错误恢复方案是 2/64，即对于每块 512 字节的数据，每 64 块中可以有 2 块发生错误，这时驱动程序可以将错误块校正。带有 ECC 的磁带驱动器出错的概率是  $10^{-14}$ ，而标准驱动器的出错概率就大多了，为  $10^{-9}$ 。

应注意对带有 ECC 的设备所建立的磁带作好标签，因为这样的磁带用标准设备不能读出。另外，如果在两台机器之间传送数据，则只有在目标机支持 ECC 方案时才建议使用 ECC 设备。

## 3.2 使用软磁盘

软磁盘是使用最方便的存储介质。在一张软盘上存储的数据量从 360K 字节到 1.4 兆字节不等，这决定于你的软盘驱动器。软磁盘可以用于以 tar, cpio, dd 或者 dump 格式存储数据，也可以在软盘上建立可安装的文件系统。下面的部分讲述如何用软盘来作数据存储以及如何用作额外文件系统的存储空间。

### 3.2.1 格式化软盘

软盘在使用之前必须先作格式化，UNIX 系统格式化软盘的命令为：

```
format /dev/floppy-device
```

(Sysadmsh 的用户选择: Mdeia→Format)

命令行中指示的 `floppy-device` (软盘设备) 与磁盘驱动器种类和使用的软盘有关。比如，若你的软驱是 5.25 英寸的高密度软盘驱动器，你可以将它用作高密度方式 (每英寸 96 磁道)，也可用作低密度方式 (每英寸 48 磁道)，如果要在该软驱中使用高密度软盘，则软盘设备可指定为：

```
/dev/rfd096
```

在本例中，`rfd` 表示原始软盘设备，`0` 表示这是主软盘驱动器，`96` 表示高密度方式。类似地，如果希望用低密度软盘和软盘驱动器的低密度方式，则设备名为：

```
/dev/rfd048
```

这里 `48` 表示 `0` 号软驱的低密度方式。

`/etc/default/format` 文件：

还可以在 `/etc/default/format` 文件中增加一个条目来定义缺省格式设备。比如：

```
DEVICE = /dev/rfd096ds15
```

在增加了上面的一行之后，就再也不必指定设备名了。另外，还可以定义使所有软盘必须经过检验，即确认软盘上的所有数据都是可读的。(这可以通过在命令行上增加 `-y` 任选项来指示)。可以用下面条目指定自动检验方式：

VERIFY=Y

如果在 /etc/default/format 文件中增加了以上条目，则所有用 format 命令格式化的软盘都要执行检验。（要跳过检验过程，可在命令行中使用 -n 任选项）

### 3.2.2 拷贝软盘

要保证软盘上存放的数据不致丢失，用户可以用 diskcp (c) 命令或 dd (c) 命令将软磁盘的内容复制到新的、经过格式化的软盘上。

diskcp 调用 dd 命令并提供一个简单的用户接口，dd 命令的功能非常强，你可以用它来执行多种不同类型的拷贝工作。

用于拷贝的软盘必须经过格式化。软盘一经格式化，即可反复使用而不必重新格式化。

如果你的软盘是在另一个操作系统下格式化的，那么在用作 UNIX 磁盘拷贝时必须先格式化。注意，在某些操作系统下格式化的软盘在其他操作系统下不能使用，即使重新格式化也不行。这样的软盘只能为特定操作系统专用。

格式化软盘可用 format 命令，这在本节的前一部分已经讲到。diskcp 命令也可用来格式化软盘。

(Sysadmsh 用户选择: Mdia >Duplicate)

用 diskcp 命令拷贝软盘，应按以下步骤：

1. 将你要复制的软盘即所谓“源”盘，插入你的主软盘驱动器——0 号驱动器中。

2. 在另一个驱动器中插入另一张软盘即所谓目标盘。应注意已经存在于目标盘上的信息在命令执行之后完全被破坏。

如果只有一个软盘驱动器，则只将源盘放入。diskcp 命令会在恰当的时候提示你放入目标盘。

3. 要在内容拷贝之前格式化软盘，应输入下面命令并按回车键：

```
diskcp -f
```

如果你的计算机有两个软盘驱动器，则输入下面的命令将源盘内容直接拷贝到目标盘上：

```
diskcp -d
```

如果你不必格式化目标盘，就输入以下命令并按回车键：

```
diskcp
```

4. 按出现在屏幕上的指令执行。在单驱动器系统中，会提示你移去源盘，插入目标盘。

用 dd 命令拷贝软盘，应按如下步骤：

1. 将待拷贝的软盘放入 0 号软盘驱动器

2. 将一张格式化过的软盘放入 1 号驱动器。如用必要，应先用前一部分讲述的 format 命令将软盘格式化。

3. 输入：

```
dd if= /dev /fd0 Of= /dev /fd1 Count= blk Count
```

按回车键。blkcount 是软盘上要拷贝的数据块的数目。如果你不知道该值是多少，



可在上述命令行上空出 Count = blkcount 部分。

该命令将第一张软盘的内容拷贝到第二张软盘上，然后显示拷贝的数据块的数目。

### 3.2.3 将软盘用于文件存储

将软盘用于单个文件的存储，首先应保证软盘已经格式化，然后把软盘片放入软盘驱动器中。所有标准 UNIX 系统中访问软盘的命令都可以使用，包括 tar, cpio, dd 或者 backup 等。

建议在可能的情况下尽可能用 tar 命令。比如，要以 tar 格式将文件的拷贝件放到一张高密度软盘中，应使用以下命令：

```
tar CV 文件名
```

关于 tar, cpio, dd 以及 backup 等的使用方法，请查阅本书中相应章节。

### 3.2.4 在软盘上建立文件系统

你可以在软盘上建立文件系统，方法与在硬盘上建立文件系统很相似。软盘上的文件系统可以携带，可以安装在任一 UNIX 系统上。在安装文件系统时如果没有指定安装点，则使用一个名为 /mnt 的特殊目录。出于系统安全性的考虑，在使用软盘文件系统时必须作为超级用户注册。

要在软盘上建立可移动的文件系统，应按下面给出的步骤进行：

1. 作为超级用户注册并输入命令：

```
mkdev fd
```

(Sysadmsh 的用户选择：Filesystem → Floppy)

2. 在屏幕上出现如下菜单：

```
Floppy Disk Filesystem Creation Program
```

```
Choices for type of floppy filesystem.
```

- 1.48tpi, double sided, 9 sectors per track
- 2.96tpi, double sided, 15 sectors per track
- 3.135tpi, double sided, 9 sectors per track
- 4.135tpi, double sided, 18 sectors per track

```
Enter an option or q to quit:
```

输入选用的磁盘类型号并按回车键。

3. 接着你看到如下菜单：

```
Choices for contents of floppy filesystem.
```

- 1. Filesystem
- 2. Bootable only (96ds15 and 135ds18 only)
- 3. Root filesystem only (96ds15 and 135ds18 only)

```
Enter an option or enter q to quit:
```

输入“1”并按回车键。

4. 系统提示：

Insert a type floppy into drive 0

Press Return to Continue or enter q to quit

其中“type”处为你选择的软盘类型。按回车键。

5. 屏幕显示提示信息:

would you like to format the floppy first? (Y/n)

如果你已经将软盘格式化, 则输入“n”, 文件系统会立即开始建立; 如果软盘尚未经过格式化, 则输入“y”, 你会看到:

formatting /dev/tape

track 00 head 0

随着格式化过程的进行, 磁道号和磁头号也随之跳变。如果 /etc/default/format 文件中包括 VERIFY=Y, 则在格式化之后还要进行检验。

6. 接着系统询问文件系统类型:

Do you want to use the default file system type AFS (Y/n)?

如果以“n”回答, 则要求你提供文件系统类型:

Please enter a file system type from (AFS, S51K, XENIX)

or enter q to quit

7. 输入文件系统类型并按回车键。

这一过程完成之后, 系统显示:

Filesystem creation complete.

8. 然后回到主菜单:

choices for contents of floppy filesystem.

1.filesystem

2.Bootable only (96ds15 and 135ds18 only)

3.Root filesystem only (96ds15 and 135ds18 only)

Enter an option or enter q to quit

这时输入 q 并按回车键。到此你的软盘上已经含有一个文件系统了。

### 3.2.5 安装软盘文件系统

要想使用软盘文件系统, 就必须将它安装到你的系统上。比如, 96tpi (每英寸 96 磁道) 的软盘应使用如下命令:

```
mount /dev/fd096 /mnt
```

(Sysadmsh 用户应选择: Filesystems→Mount)

注意, 所用的软盘设备是 fd096 而不是 rfd096。在安装软盘文件系统时, 必须使用不带前缀“r”的设备名。再举一例, 如果安装的是 48tpi 软盘上的文件系统, 则用下面命令:

```
mount /dev/fd048 /mnt
```

在发出 mount 命令之后, shell 应该返回提示符状态, 表明文件系统已经成功地安装完毕。现在就可以用 cd 命令进入文件系统, 并在其中正常地建立文件。当执行完毕, 你

希望移走软盘时应发如下命令：

```
umount /dev/fd096
```

(Sysadmsh 的用户应选择：Filesystem ▶ Unmount)

你的文件系统立即拆卸下来，文件包含在软盘上，可以很容易地恢复或传递。

## 第四章 使用鼠标

本章介绍在系统上安装不同厂家、不同类型的鼠标接口的基本知识。使用鼠标能给用户和开发人员带来极大的方便，因此，系统为串行鼠标和总线鼠标硬件都提供支持。

### 4.1 安装硬件

在你的硬件的文献手册中查明有关硬件配置的信息。注意鼠标的厂家和类型，以及是附加于串行口还是直接加到系统总线上的。同时可参考本书“增加多端口卡、内存卡以及其它总线接口卡”一章的内容。

在使用鼠标时应注意到以下几条限制：

- Microsoft 总线型鼠标不能用中断向量 7 来配置，应使用中断向量 2, 3 或 5。
- 在单用户（维护）方式下不要使用 usemouse 实用程序。
- 用 usemouse 实用程序不能启动 System >Terminate (shutdown) 的执行。

### 4.2 安装鼠标

在系统上安装鼠标，必须按照以下步骤进行：

1. 根据制造厂家的指令安装鼠标。
2. 保证 link kit 软件包已经安装好并能正常工作。如果没有 link kit 软件包，就不能安装鼠标驱动程序，这时应先用 custom (ADM) 命令将 link kit 安装到系统上。
3. 以超级用户身份注册，并输入命令：

```
mkdev mouse
```

(Sysadmsh 的用户应选择：System >Configure >Hardware >Mouse)

将看到鼠标初始化菜单：

```
Mouse Initialization Program
```

1. Display current configuration
2. Add a mouse to the system
3. Remove a mouse from the system
4. Associate a terminal with an existing mouse
5. Disassociate a terminal from an existing mouse
6. Remove the mouse drivers from the kernel

Select an option or enter q to quit:

要安装鼠标，应选择第 2 项并回车。其余各选项允许你在任一时刻改变你鼠标的配置。比如，在系统上增加或移去附加鼠标，或者改变可以从现有鼠标接受输入信息的终端。

4. 然后，指定使用的鼠标的类型。这时系统显示菜单如下：

The following mouse devices are supported:

1. Logitech Serial Mouse
2. Microsoft Serial Mouse
3. Mouse Systems PC II Serial Mouse
4. Mouse Systems PC Mouse
5. Microsoft Bus mouse
6. Olivetti Bus Mouse
7. Logitech Bus Mouse
8. Keyboard Mouse

Select an option or enter q to return to the previous menu:

5. 输入要安装的鼠标对应的数字并按回车键；你会看到：

```
mouse__type is currently configured
to attach to the system on / dev / tty
```

Do you want to install this mouse on a different port? (Y / n)

输入“n”表示不想改变缺省端口，而如果希望改变缺省端口的话，可输入“y”，并在系统提示时输入另选的端口。

6. 如果安装的是总线型鼠标，则要求选择总线型鼠标的总线卡配置情况。如果安装的是串行鼠标，可跳过这一步直接到下一步骤。如果选择的是总线型鼠标，则会看到：

Busmouse Configuration

1. Display current busmouse parameters
2. Modify current busmouse parameters
3. Select previous busmouse parameters
4. Select default busmouse parameters

Enter an option or q to quit:

如果希望使用总线型鼠标参数的缺省值，可选第4项，这时屏幕显示当前参数值，应按q键退出本菜单。这一选项对你的总线型鼠标进行自动配置。如果改变中断向量并选择中断向量5，则会与卡式磁带设备发生冲突，因为它也使用了中断向量5。当然，如果两者不同时使用，则不会发生问题。

7. 接着，系统要求指定可以从鼠标接受输入信息的终端和多屏幕。如果没有将鼠标物理地连接到系统上，则不要指定任何终端或多屏幕，否则会收到错误信息。你可以选择任何一个，甚至所有终端和控制台多屏幕使用鼠标。输入单词“multiscreen”表示允许所有控制台多屏幕使用鼠标。

注意：对任一指定的串行设备，只允许用一个鼠标作为输入。

关于几个终端或多屏幕共享鼠标的问题，将在下一节中加以讨论。

屏幕显示：

```
This mouse may be configured for use on any of the system's
terminals and miltiscreens. The multiscreens and terminals
that will be associated with this mouse need to be specified.
```

Specify them by entering, at the following prompt, all the ttys to be associated with this mouse. Entering the word

“multiscreen” will associate all of the console multiscreens.

Enter a list of terminals (e.g. tty1a tty2a multiscreen)

or enter q to quit. Press return when finished:

输入完所有需要以此鼠标为输入的终端和多屏幕之后，以回车结束。你会看到：

Do you want to use the <mouse\_\_type> on any other terminals? (Y / n)

在 mouse\_\_type 的位置上，是前面指定的鼠标的制造厂家或者类型。如果没有其他终端要接受鼠标输入就输入“n”。如果打入“y”，则回到提示输入终端清单的屏幕。

8.为使指定的驱动程序生效，需要将系统配置信息更新，并建立新的操作系统核心。此时屏幕显示以下信息，并将持续显示若干分钟：

Updating system configuration...

You must create a new kernel to effect the driver change you specified.

Do you wish to create a new kernel now? (Y / n)

打入“y”表示将鼠标驱动程序加入到操作系统核心当中。

9.之后，你看到：

The UNIX operating system will now be rebuilt.

This will take a few minutes. Please wait.

Root for this system build is /.

作为连接进程的一部分，屏幕显示：

The UNIX kernel has been rebuilt.

Do you want this kernel to boot by default? (Y / n)

如果希望以后每次启动系统时都使用这个已包含鼠标驱动程序的操作系统核心，则应打入“y”。

10.屏幕显示：

Backing up / unix to / unix.old.

Installing new / unix.

The kernel environment includes device node files and / etc / inittab.

The new kernel may require changes to / etc / inittab or device nodes.

Do You want the kernel environment rebuilt? (Y / n)

输入 y。

11.系统提示信息：

The new kernel has been successfully linked and installed.

To activate it, reboot your system.

Setting up new kernel environment.

至此已经将鼠标驱动程序安装到操作系统核心中。

12.最后，再次回到主菜单。如果此时你不想对鼠标配置信息作什么改动，可输入 q 并回车，退出主菜单。

在任何时候都可以执行 `ldev mouse`，以允许或防止某些终端使用鼠标输入，或者移去鼠标，或者检查当前配置信息。

移去鼠标:

从系统中移去鼠标或鼠标驱动程序的过程与安装鼠标的过程正好相反。应在 `mkdev mouse` 的主菜单下选择“移去”一项，而不是“增加”一项。

### 4.3 使用鼠标

鼠标的使用是自动的。如果一个应用程序或实用程序接受鼠标输入且允许终端使用鼠标，则你只要启动该程序，鼠标即开始工作。如果终端或多屏幕没有被允许使用鼠标（见前一节内容），或者程序的配置不接受鼠标输入，则使用鼠标没有任何用处。

#### 4.3.1 将鼠标用于多屏幕

总线视频卡上附加的监视器上的多屏幕（multiscreen）为使用鼠标提供了最方便的方法。如果鼠标与主系统控制台（通常是系统总线上的视频卡上附加的监视器）上的多屏幕相关联，则鼠标的输入即与当前处于激活的多屏幕相关联。比如，如果你的系统在主系统控制台上有多屏幕，而且所有的屏幕都被允许使用鼠标，则从鼠标输入的信息送到在当前处于激活状态的多屏幕上运行的程序中。

记住，如果程序不能接受鼠标输入，则即使在被允许使用鼠标的多屏幕上运行该程序，也不会受鼠标的移动所影响。

串行（终端）多屏幕和串行控制台也可以配置为能够使用鼠标的方式。

#### 4.3.2 将鼠标用于串行终端

在安装鼠标时，系统提示你列出允许使用鼠标输入的串行端口清单，你可以允许串行端口上的终端使用鼠标，如同允许多屏幕使用鼠标一样。再重复一遍：如果没有将鼠标物理连接到系统上，不可允许任何设备使用鼠标。

#### 4.3.3 几个终端共享鼠标

如果鼠标由几个终端共享，则确定鼠标与哪个串行端口上的终端相关联的原则是“先来先服务”。最早启动可以接受鼠标输入的程序的用户在整个程序运行过程中占有鼠标。另一个用户要想使用鼠标，必须等当前占有鼠标的用户退出执行的程序不再等待鼠标输入之后，才能从鼠标获取所需的输入信息。

注意：在可以使用鼠标的串行终端上工作的用户在鼠标忙时可以运行程序，只是不能从鼠标接受输入数据，但除这一点之外都能正常工作。

#### 4.3.4 将鼠标用于键盘有关的程序

`usemouse (C)` 实用程序将鼠标的移动及操作和与键盘有关的程序所用的键对应起来，以满足程序对键盘输入的要求。

## 第五章 增加硬盘

如果你的系统的硬盘空间经常不足，你可以通过增加一个硬盘的办法给系统补充额外的空间来存储用户的文件和目录。系统支持如下类型的第二硬盘和控制器：

- ST506 (IBM AT 标准)
- ESDI (SMS OMTI 8620 或 8627 控制器)
- SCSI

可能有的三种配置为：

- SCSI 主适配器上的根磁盘带有增加一个 SCSI 主适配器的任选项，每个 SCSI 主适配器支持控制器的数目最多为 7，而每个 SCSI 控制器支持的设备数可达 8 个。

- STD506 控制器上的根磁盘带有一个任选项，可增加一个 STD506 控制器，每个控制器可支持 2 个 ST506 磁盘以及两个 SCSI 主适配器。

- OMTI 控制器中的根磁盘支持另一个磁盘，这二者可以是 ESDI 或 ST506 这两种类型之一。

下图给出了第二类配置情况的示意图。

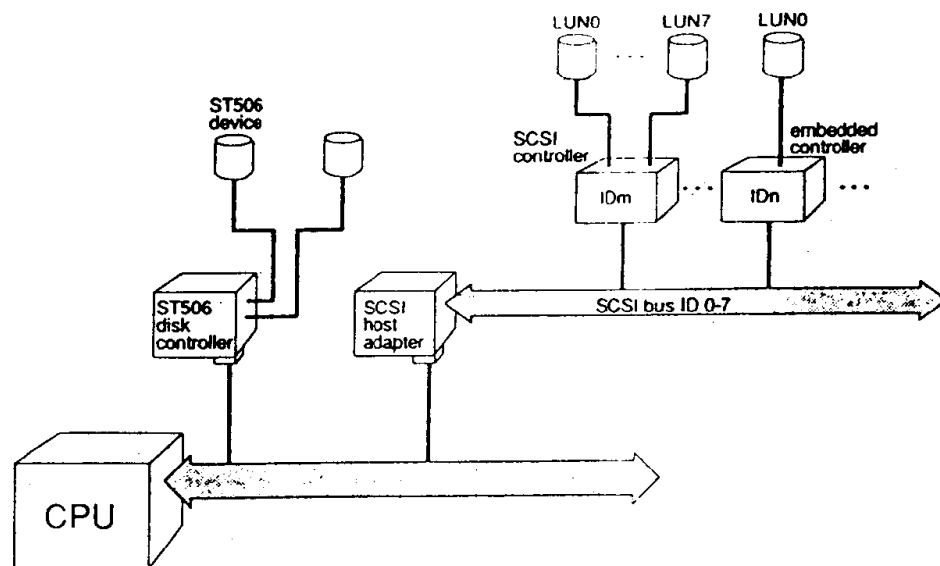


图 5.1 ST506 和 SCSI 配置示意图

SCSI 主适配器 (HA) 将信号从 CPU 总线传送到 SCSI 总线上。SCSI 控制器以 SCSI-ID 来标识，SCSI 设备由逻辑部件号 (LUN) 来指定。

当安装完操作系统，完成对根磁盘的初始化之后，根磁盘即被配置为第一个控制器上的第一个硬盘 (对 ST506 或 ESDI 磁盘而言) 或者第一个主适配器 (对 SCSI 磁盘而言)。

对各种类型的磁盘来讲，增加磁盘的基本过程都是相同的，但有时也需要根据所安装磁盘的类型采取某些不尽相同的步骤。下面将对这些不同的步骤加以说明。



## 5.1 在开始之前

在安装附加硬盘之前，首先必须决定如何配置硬盘，然后建立硬件并进行连接。本节讲述用 `mkdev hd` 命令配置和增加硬盘的实现方法。

### 5.1.1 配置硬盘

需要决定希望如何配置磁盘，以便将该信息提供给安装实用程序。

#### ST506 或 ESDI 磁盘

要用 `mkdev hd` 命令来配置 ST506 或 ESDZ 磁盘，必须知道新磁盘是由哪一个磁盘控制器支持的，以及是该控制器上的第一个还是第二个磁盘。

命令的句法为：

```
mkdev hd disk controller
```

磁盘和控制器编号从 0 开始。

ST506 和 ESDI 的相应命令见下表：

表 5.1 ST506 和 ESDI 的有关命令表

控制器	命令	增加的硬盘
ST506 及 ESDI	<code>mkdev hd 00</code>	第一个控制器上第一个硬盘
	<code>mkdev hd 10</code>	第一个控制器上第二个硬盘
只对 ST506	<code>mkdev hd 01</code>	第二个控制器上第一个硬盘
	<code>mkdev hd 11</code>	第二个控制器上第二个硬盘

### SCSI 磁盘

要用 `mkdev hd` 命令配置 SCSI 设备，必须知道：

- 主适配器上控制器的 ID 号 (0~7)，主适配器自身的 ID 号通常为 7，这使得它在 SCSI 总线上具有最高的优先级。
- 主适配器号 (SCSI-0 或者 SCSI-1)。
- 在 SCSI ID 上设备的逻辑部件号 (LUN)，在控制器与设备是同一个物理部件的情形下，LUN 值通常为 0。

以上各值的图示见图 5.1 所示。你可以选择由系统询问这些值的方式，也可以由你在命令行中给出。用以下的句法来指定配置：

```
mkdev hd id host lun
```

这里，`id` 是从 0 到 7 的数；`host` 是 SCSI-0 或 SCSI-1；`lun` 是从 0 到 7 的数。

以下的指令假定在执行 `mkdev hd` 命令时没有给出参数。如果你在命令行中给出了参数信息，即可直接转到以下过程的最后一步。在第二次运行 `mkdev hd` 命令时所给出的和参数值必须与第一次完全一致。

增加 SCSI 磁盘的步骤如下：

1) 输入如下命令：

```
mkdev hd
```

(Sysadmsh 用户选择：System > hardware > Hard Disk)

2) 如果你的根磁盘附加在 ST506 控制器上，你将看到屏幕上显示如下信息：

Your root hard disk is attached to an ST506 controller.

Pick one of the choices below or you may quit and invoke `mkdev hd-u` for a detailed usage message.

- 1) Add a hard disk to ST506 controller
- 2) Add a hard disk to SCSI controller

Enter 1, 2 or enter `q` to quit:

输入 2 并回车。

- 3) 如果根磁盘附加于 SCSI 控制器上, 则会看到:

Your root disk is attached to a SCSI controller.

The only available choice is to add another SCSI disk.

Do you want to add another SCSI disk?

(For detailed usage message, pick `'n'` to exit this script and invoke `mkdev hd-u`) (Y / n)

打入“y”并回车。

- 4) 接着, 你会看到:

What is the ID of the controller for this device?

Select 0-7, or `'h'` for help, or `'q'` to quit:

输入附加于适配器的控制器号。

- 5) 这时应输入磁盘附加于其上的适配器号:

Which SCSI host adapter supports this device?

Select 0 or 1, or `'h'` for help, or `'q'` to quit:

如果是第一主适配器, 则打入“0”, 如果是第二主适配器, 则打入“1”。

- 6) 系统给出提示信息:

What is the LUN of this device?

Select 0-7, or `'h'` for help, or `'q'` to quit:

输入附属控制器的设备号。对大多数磁盘而言, 控制器与设备是同一物理部件, 在这种情况下, 逻辑部件号为 0。

- 7) 数据输入之后, 程序接受了这些信息, 并询问是否对操作系统核心进行重连接:

Updating SCSI configuration file...

The SCSI configuration file has been updated.

A new kernel must be built, to reflect the changes

to the SCSI configuration. Do you want to do this now? (Y / n)

要能够识别出新的磁盘, 必须对操作系统核心进行重配置。这里给出“连接 / 不连接”的选择是在增加多个设备时, 这时核心需要作一次重连接, 而不必每增加一个设备就重连接一次。

在操作系统核心重新配置的工作完成之后, 屏幕显示下列信息:

After the system is rebooted with the new kernel,

reinvoke `mkdev hd` to initialize the new SCSI hard disk.

Use the same values for Host Adapter, ID, and LUN.  
到此你已为新磁盘配置了必要的软件支持。

### 5.1.2 硬件准备

在 ROM 表中没有匹配条目的硬盘是通过软件支持的。在增加第二硬盘时，必须改变某些关于主适配器、SCSI ID 以及磁盘的开关设置。

在改变带嵌式控制器的 SCSI 设备的设置时，应使用 SCSI ID 号，而不是逻辑部件号 LUN。在嵌式控制器（即控制器与设备是同一个物理部件）上的 LUN 值是 0，因为它是该控制器上第一个也是唯一的一个设备。

在增加新的磁盘之前，必须知道如何将它连接到计算机上。硬盘的连接方法在有关硬件手册中有详细说明，请查阅。

在安装系统之前应保证增加的驱动器经过格式化且能够通过厂家提供的诊断程序。如果不能通过，就不能把它用到系统中。

## 5.2 安装硬盘

下面是安装另一个硬盘的步骤，假定只有一个 UNIX 文件系统，没有 DOS 分区：

1、在连接好硬盘并启动系统之后，进入系统维护方式，使用某种格式的 mkdev 命令，在命令行中指定所需的配置信息：

```
mkdev hd disk controller
```

2、如果你的根磁盘附加于 ST 控制器，你会看到如下的屏幕信息：

```
Your root hard disk is attached to an ST506 controller.
```

```
Pick one of the choices below or you may quit and  
invoke mkdev hd -u for a detailed usage message.
```

- 1) Add a hard disk to ST506 controller
- 2) Add a hard disk to SCSI controller

```
Enter 1, 2 or enter q to quit:
```

输入一个数并回车，如果你想增加 ST506 磁盘，应转向第 6 步，如果增加的是 SCSI 磁盘，则转到第 5 步。

3、如果根磁盘附加于 OMTI 控制器，则会在屏幕上显示：

```
Your root hard disk is attached to an OMTI controller.
```

```
The only available choice is to add one other hard disk  
Enter 'y' to add another disk. If you enter 'n' you will exit  
this script. You may then invoke mkdev hd -u for a detailed  
usage message. (Y/n)
```

```
Enter y and press <Return> . Proceed to step 6.
```

打入“y”并回车。转到第 6 步。

4、如果根磁盘附于 SCSI 控制器，则屏幕显示：

```
Your root disk is attached to a SCSI controller.
```

The only available choice is to add another SCSI disk.

Do you want to add another SCSI disk?

(For detailed usage message, pick 'n' to  
exit this script and invoke mkdev hd -u) (y/n)

打入“y”并回车。

5、接着系统提示输入与前相同的控制器、主适配器及 LUN 信息:

What is the ID of the controller for this device?

Select 0-7, or 'h' for help, or 'q' to quit:

Which SCSI host adapter supports this device?

Select 0 or 1, or 'h' for help, or 'q' to quit:

What is the LUN of this device?

Select 0-7, or 'h' for help, or 'q' to quit:

按命令要求输入各值，并转第 7 步。

6、如果增加的是 ST506 或 ESDI 磁盘，则屏幕显示:

Will this disk be the first or second disk on this controller?"

Enter 1 (first) or 2 (second):

打入一个数并回车。

7、如果你只是增加 ST506 磁盘，将看到如下信息:

Will this disk attach to the first or second ST506 controller?

Enter 1 (first) or 2 (second):

输入一个数并回车。

8、系统显示提示信息:

During installation you may choose to overwrite all  
or part of the present contents of your hard disk.

Do you wish to continue? (y/n)

打入“y”并回车。

9、如果你用的是 SCSI 控制器，则会看到:

The hard disk installation program will now invoke /etc/fdisk.

Entering 'q' at the following menu will exit /etc/fdisk.

and the hard disk installation will continue.

If you wish to exit the entire installation at this menu,  
press the <DEL> key.

跳至第 15 步。

---

注: SCSI 的安装跳过第 10 步至第 14 步。

---

10、如果你用的是 ST506 (标准接口) 控制器，则会看到如下的信息及提示:

The hard disk installation will now invoke /etc/dkinit.

Entering 'q' at the following menu will exit /etc/dkinit,

and the hard disk installation will continue.

If you wish to exit the entire installation at this menu,  
press the <DEL> key.

#### Hard Disk Drive 1 Configuration

1. Display current disk parameters
2. Modify current disk parameters
3. Select default disk parameters

Enter an option or 'q' to quit:

11、如果用的是 OMTI 控制器，则会看到如下的附加信息。

Caution: Consult the ESDI installation Release Notes if  
you wish to modify the disk parameters the / etc / default  
will display.

如果打入“q”，则会看到下面的信息：

The hard disk installation program will now invoke  
two disk preparation utilities: fdisk and badtrk.  
Selecting 'q' at the main menu for each utility  
will exit that utility and continue with the hard  
disk installation.

跳到第 15 步。

12、dkinit 菜单用于为非标准磁盘设置参数。如果你的硬盘是标准的，即为你的计算机硬件或特殊母板 ROM 所支持，可打入“3”接着回车，继续安装过程。另外，如果你的硬盘是 SCSI 硬盘，则必须打入“q”，参数已经设置好。

在这里打入“q”表示为硬盘选择缺省参数。除非你知道硬盘是非标准的，否则假定它是标准的并继续安装操作。

如果磁盘是非标准的，你必须输入新的配置信息以覆盖原有的 ROM 磁盘配置信息。如果不能确定对非标准磁盘应输入什么样的参数，要向该磁盘的生产厂家索取该信息。

如果输入 1 或 2，则你看到如下显示：

Disk Parameters Values

1.Cylinders	value
2.Heads	value
3.Write Reduce	value
4.Write promp	value
5.Ecc	value
6.Control	value
7.Landing Zone	value
8.Sectors / track	value

在实际显示时，“value”被该变量的缺省值所取代。

如果打入“1”，再次显示第一个菜单。如果打入“2”，提示：

Enter a parameter to modify or 'q' to return to the main menu:

13、打入从 1 到 8 的任意一数以改变磁盘参数，如果打入“q”就会回到前一张菜单。

你会看到:

Enter the new value or <RETURN> to use the existing value:

如果你希望改变该值, 可输入新的值, 也可以按回车键保留原值。

14、在完成磁盘参数的修改之后, 打入“q”回到主菜单, 然后再次打入“q”将所作的修改存下来。用打入“q”的方法退出 dkinit 菜单, 即以输入值覆盖了原来的参数值。如果在作了修改之后希望恢复缺省参数, 可从第一张菜单打入“3”。

15、接着, 安装程序运行 fdisk 实用程序对硬盘进行分区。在分区时可以在同一硬盘上支持 DOS。(如果 DOS 已经安装好的话), 也可以将整个硬盘用作 UNIX 系统专用。

过一会儿之后, fdisk 菜单出现在屏幕上:

- 1.Display Partition Table
- 2.Use Entire Disk for UNIX
- 3.Use Rest of Disk for UNIX
- 4.Create UNIX Partition
- 5.Activate Partition
- 6.Delete Partition

Enter your choice or 'q' to quit:

选择第 1 项并回车。

如果你从未在硬盘上安装过操作系统, 则会看到类似的表格:

Current Hard Disk Drive: / dev / rhd10

Partition	Status	Type	Start	End	Size

Total disk size: 1220 tracks (5 reserved for masterboot and diagnostics)

Press <RETURN> to continue

如果以前在硬盘上安装过操作系统, 则 fdisk 菜单被填入内容。DOS 通常以分区号 4 加以显示。

16、再次按回车键回到主菜单。如果你希望由 UNIX 分区占用整个硬盘, 可选第 2 选项。在作完选择之后, 打入“q”退出 fdisk 菜单。如果以前在该硬盘上安装过其他操作系统, 则还会显示如下的警告信息:

Warning! All data on your disk will be lost!

Do you wish to continue? (y / n)

如果希望由 UNIX 系统占用整个硬盘, 则打“y”, 保证 fdisk 对整个硬盘进行分区。

---

注: 如果选了第 3 项, 即将硬盘的剩余部分分配给 UNIX 系统, 则必须选择第 5 选项使 UNIX 分区成为当前分区。如果不做这一步, 则当前分区是头一个分区。

许多计算机在硬盘的最后一个柱面存放着诊断程序, 这意味着不应把最后一个柱面分配给任一分区。如果你从 fdisk 菜单选择第 2 选项, 就没有分配最后一个柱面。如果选择了第 4 选项, 则应该注意不要将最后一个柱面分给 UNIX 系统。

---

17、按回车键回到主菜单。现在你已在硬盘上建立起分区, 为了继续安装过程, 应打入“q”并按回车键。

如果你的控制器是 ST506 的，继续执行第 18 步；

如果你的控制器是 SCSI 或 OMTI 控制器，则转到第 27 步，因为 SCSI 和 OMTI 控制器的硬盘安装不运行 badtrk 实用程序。

18、这时看到程序 badtrk (ADM) 的菜单。使用 badtrk 程序，可以在硬盘上扫描查找有问题的磁道。程序会将有缺陷的磁道位置对应到硬盘别处的正常磁道上。另外还建立一个坏道表，即硬盘上所有坏道的清单。

badtrk 的主菜单如下：

- 1.Print Current Bad Track Table
- 2.Scan Disk (You may choose Read-Only or Destructive later)
- 3.Add Entries to Current Bad Track Table by Cylinder / Head Number
- 4.Add Entries to Current Bad Track Table by Sector Number
- 5.Delete Entries Individually from Current Bad Track Table
- 6.Delete All Entries from Bad Track Table

Please enter your choice or 'q' to quit:

打入“2”并回车。

19、显示如下子菜单：

- 1.Scan entire UNIX partition
- 2.Scan a specified range of tracks
- 3.Scan a specified filesystem

Please enter your choice or 'q' to quit:

选择选项 1。

20、根据你所要求进行扫描的范围（整个 UNIX 分区，指定的磁道范围，指定的文件系统）打入相应的数字，系统显示如下提示：

- 1.Quick scan (approximately 7 megabytes / min)
- 2.Through scan (approximately 1 megabyte / min)

Please enter your choice or 'q' to quit:

如果选择第 1 选项“快速扫描”，则程序的执行会快些，但有可能不能将所有坏磁道都扫描出来。一般应打入“2”选择“彻底扫描”。

21、系统显示如下的提示信息，询问是否进行破坏性扫描：

Do you want this to be a destructive scan? (y / n)

打入“y”，则系统警告：

This will destroy the present contents of the region you are scanning.

Do you wish to continue? (y / n)

打入“y”并按回车键。你会在屏幕上看到如下信息，指示正在进行扫描：

Scanning in progress, press 'q' to interrupt at any time.

22、在回答了上述的系统询问之后，程序开始对新加硬盘的活动分区进行扫描以寻找坏的磁道。磁道空间越大，这一过程所花的就越长。所以一个非常大的硬盘的扫描要花去相当长一般时间。

badtrk 程序在扫描磁盘时，还在屏幕上显示检查的磁道号以及已经扫描过的磁道数

占总磁道数的百分比。在任一时刻按“q”键都可以中断扫描过程。在打入“q”中断扫描执行时，不必按回车键，系统会询问是继续扫描过程还是返回主菜单。

当 badtrk 程序扫描到有缺陷的磁道后，即以扇区号和柱面或磁头号的形式列出该坏道的位置。坏道信息填入表中并显示在屏幕上。以下是一个坏道的例子：

```
WARNING: wd: on fixed disk ctnr = 0 dev = 0 / 47block = 31434 cmd = 00000020
        status = 00005180, sector = 62899, cylinder / head = 483 / 4
```

23、扫描完成之后回到菜单，选择第 1 选项可查看扫描的结果。坏道表的格式类似：

Defective Tracks

	Cylinder	Head	Sector Number (s)
1.	190	3	12971-12987

Press <RETURN> to continue

按回车键回到主菜单。

注：如果 UNIX 分区的头几条磁道中有了坏道，则回到 fdisk 实用程序，对磁道进行重新分区，使 UNIX 分区不再包含有缺陷的磁道。必须进行试验才能确定应去掉多少磁道。不要将坏道分配给任何一个操作系统。退出 fdisk 之后，再次运行 badtrk 程序以扫描查找另外的坏道。

这一过程要不断地进行，一直到 badtrk 程序在头几条磁道中不再查出坏道为止。

24、如果硬盘本身有坏块图，应将其中的所有坏块都填入坏道表中。

由于大多数的磁盘坏块属于边界的或间歇性的，所以任一坏块图中含有的坏块必然多于扫描时所发现的，这时应把在坏道表中没有出现的坏道填入坏道表中。

25、如果你的硬盘没带有坏块图，或者你已完成对坏道表的修改，可打入 q 并按回车键。

26、接着提示你输入分配来替换坏磁道的磁道数，你输入的数不应小于推荐使用的值。如果直接按回车键，则表示使用显示在屏幕上的推荐数目的磁道：

```
Enter the number of bad tracks to allocate space for
```

```
(or press <RETURN> to use the recommended value of n)
```

推荐值的选择，是基于当前在坏道表中的坏道数，加上将来可能坏的磁道数。在选择时应小心，如果将来坏道数目超过了输入的数值或者推荐值，则必须重新安装系统。

接着出现 divvy 的提示。divvy 程序将分区划分为文件系统。可以在一个分区上建立的区段数可达到 7 个，并可以任意命名。

注：应尽量使文件系统的大小保持在 60 至 80 兆字节或再少些的范围内，因为在这样大小的文件系统上系统维护工具的使用较快、较有效。



27、接着显示 divvy 主菜单和一个关于盘是如何划分的表:

Name	Type	New FS	#	First Block	Last Block
	NOT USED	no	0	0	39011
	NOT USED	no	1	39012	41511
	NOT USED	no	2	-	-
	NOT USED	no	3	-	-
	NOT USED	no	4	-	-
	NOT USED	no	5	-	-
	NOT USED	no	6	41512	41521
hdla	WHOLE DISK	no	7	0	41980

41522 blocks for divisions, 459 blocks reserved for the system

- n [ame] Name or rename a division.
- c [reate] Create a new file system on this division.
- t [ype] Select or change filesystem type on new filesystems.
- p [revent] Prevent a new file system from being created on this
- s [tart] Start a division on a different block.
- e [nd] End a division on a different block.
- r [estore] Restore the original division table.

Please enter your choice or 'q' to quit:

在 divvy 表中的每一行对应一个文件系统。在你第一次看到该表时,也许已建立了一个或多个文件系统。用 start 命令和 end 命令可以改变这些文件系统大小的缺省值。注意,文件系统的边界不可重叠,比如,0 号文件系统不能结束于另一文件系统,如 1 号文件系统的开始块处。

在第一次看到 divvy 主菜单时,文件系统尚未命名。使用 name 命令可改变文件系统的名字了。文件系统的名字可随意给出。比如,你可将一个文件系统命名为 u,表示“user”。

不要改变 7 号文件系统的配置,它是为操作系统内部使用保留的。

一旦定义了文件系统的起始和结束点,应对每个文件系统使用 create 命令。

缺省的文件系统类型为 AFS。如果你想建立其他类型的文件系统,必须使用 type 命令。

要从 divvy 退出应打入“q”。程序询问是安装新的分区,还是退回主菜单,或者不安装分区表而直接退出程序。选择选项 r 即安装分区表。

28、到此,系统开始在硬盘上建立文件系统和对换区,这会花去几分钟。屏幕显示:

```
making filesystems
```

### 5.3 增加新的文件系统

在退出系统维护方式之前,必须将新的文件系统增加到系统中。要做到这一点,应按以下步骤执行:

1、输入以下命令：

```
mkdev fs
```

(Sysadmsh 的用户选择：Filesystem >Add)

2、会看到如下的文件系统初始化程序：

```
This program performs maintenance tasks required to add or delete
an existing filesystem.Would you like to:
```

- 1.Add a new filesystem to system.
- 2.Remove a filesystem.

```
Select an option or enter q to quit:
```

选择第 1 选项。

3、系统提示输入设备名：

```
Enter a device name and press <Return> or q to quit:
```

从 /dev 中输入设备的完整路径名。比如，要增加一个名为 u 的文件系统，应输入 /dev / u

4、此时系统要求提供安装点名称：

```
Enter a directory name and press <Return> or q to quit:
```

所谓安装点即是安装文件系统的目录。比如，一个名为 u 的文件系统安装于目录 / u 中。

5、屏幕显示：

```
Reserving slots in lost+found directory...
```

```
When entering multiuser mode:
```

- 1.Always mount filesystem
- 2.Never mount filesystem
- 3.Prompt before mounting filesystem

```
Select an option:
```

如果你希望文件系统在系统启动时自动进行安装，就打入“1”。如果指定文件系统只在系统管理员发请求时才进行安装，则打入“2”，如果打入“3”，则系统在启动时就会询问是否要安装文件系统。

6、接着系统询问是否允许用户安装文件系统：

```
Do you want to allow users to mount this file system? (y / n)
```

你必须回答“y”，这样系统备份程序才能根据需要安装和拆卸文件系统。

7、当这一过程完成之时显示下列信息：

```
Updating system files...
```

```
Filesystem has been successfully added.
```

8、下面，应用下面命令安装 / x 文件系统：

```
mount / dev / x / x
```

9、要保证系统能正确识别出新的文件系统（如果你在使用 sysadmsh，则在系统提示时按(!) 键)。应输入以下命令，“;”允许在同一行上输入两条命令：

```
Chmod 755 /dev/x; chgrp auth /dev/x
```

新的文件系统即准备好，可以使用了。

#### 5.4 操作系统核心的重连接

如果在安装 SCSI 磁盘后提示重新连接操作系统核心时回答了“no”，则必须手工地将新的配置信息重新建立于操作系统核心中。输入以下命令：

```
cd /etc/conf/cf.d
```

```
./link__unix
```

(Sysadmsh 的用户应选择：

```
System →Configure →Kernel →Rebuild)
```

#### 5.5 将用户帐号移出主硬盘

如果你想将用户帐号移到第二硬盘上的新目录（比如 /x）中，首先要保证新的帐号放入新的地址处。sysadmsh (ADM) Accounts (帐号) 选项（用来建立新的用户帐号）从 /etc/default/authsh 和 /usr/lib/mkuser/homepaths 文件中读出用户帐号的缺省位置。

对 /etc/default/authsh 和 /usr/lib/mkuser/homepaths 这两个文件进行编辑，/usr 条目改成 /x，这么做将 /x 作为新的用户帐号的位置。

这样，无论何时运行 Sysadmsh Account 任选项来增加新的用户，用户帐号都在 /x 中。应保证在建立新用户之前从文件系统已安装完毕，并且在 /x 安装时用户的主目录不可访问。

如果系统中已经有了用户，而现在希望将它们的帐号移至新的文件系统，则首先应该用软盘或磁带将打算移动的帐号备份下来，以防止可能出现的数据丢失，然后使用以下的 Sysadmsh (ADM) 选择访问它们的帐号信息，改变每个用户的主目录：

```
Accounts →User →Examine: Identity
```

移至“Home Divectory”域，选择“Move”选项，根据系统的提示进行操作。对每个主目录被修改的用户都要进行与此相同的操作。

## 第六章 终端的管理

多用户系统的最重要方面是终端设备的增加和维护。增加终端可以让更多的用户访问系统，并可增强整个系统的能力。

本章讲解如下内容：

- 使终端设备能够被操作，包括串行多屏幕。
- 维护终端。
- 配置和使用 `mscreen (M)`，与 `multiscreen (M)` 等价的一个串行终端。

给计算机增加终端的前提条件是能够给系统增加物理端口，并使用 `mkdev serial` 命令使端口做好准备。

### 6.1 增加终端

在给系统增加终端之前，要先在这个终端硬件手册中查找一下把终端连接到一个串行线下的指导说明。

要想给系统增加终端，应把它连到 RS-232 串行端口上，并以 `enable (C)` 命令使其能够工作。

系统可以支持很多种终端。对终端的支持都包含在 `/etc/termcap` 文件中，其中包含各种终端的击键和控制序列的定义与分类。

以下步骤给出了把终端安装到标准 COM 串行线或安装到串行扩展卡上的方法：

1. 如果要把终端直接加到一个 COM 端口上，就不必运行 `mkdev serial`。
2. 确认一下是否是在多用户模式下作为根 (root) 在系统上登录的。

插上终端的电源线并加电。把它设置为 9600 波特率、8 位数据、1 位停止位、无奇偶校验、全双工，以及 XON/XOFF 信号交换式。如果终端不是工作于这种模式，那么就请在本章“改变 `gettydefs` 文件”一节中查找配置终端的有关说明。

有些终端以一条直接电缆线直接连到计算机上，而另一些终端要连到调制解调器上。终端连到一个调制解调器上要使用一个“假调制解调器”或“调制解调器插头”，它是一个带有 2 针和 3 针交叉的电缆。把终端连上，使串行口的“数据发送” (Transmit Data) 连到终端的“数据接收 (Receive Data)”、终端的“数据发送”连到串行口的“数据接收”。“信号地 (Signal Ground)”应连到“信号地 (Signal Ground)”上。其它针就不必连了。操作系统只要求连接 2、3 和 7 针。

3. 如果端口已经可用，就按几次回车键，看看是否出现了“login”：这个提示符。如果有，就可以准备注册了。如果没有，就使用控制台或一台工作的终端以超级用户的身份注册，并以如下命令禁止这个端口：

```
disable ttyname
```

这里的 `ttyname` 是有问题的那个端口的设备特殊文件名称。要确认一下现在所使用的是一个非调制解调器控制的设备，例如，`/dev/ttyla`，而不是 `/dev/ttylA`。

4. 检查一下这个串行端口在 `/etc/inittab` 文件中的登记项是否像下面这样

(ttyname 是设备文件名, 例如: /dev/ttyla):

```
lla: 2: respawn: /etc/getty ttyla m
```

如果这个登记项不类似这个例子, 就要对这个文件编辑一下来修正它。

---

注: 如果对 /etc/inittab作了手工改动, 并希望使其永久化, 那么也必须对 /etc/conf/cf.d/init.base作相同的改动。这是因为每次对核心进行重新连接时(譬如在增加了一个驱动程序或改变了一个可调参数时), /etc/inittab就要根据在文件 /etc/conf/cf.d/init.base中所找到的登记项来进行重新建筑。

---

5. 从控制台, 作为根 (root), 看一看是否能够通过键入如下命令把输出重新定向到某终端上:

```
date > /dev/ttyname
```

如果没有在这个终端上看到有日期显示出来, 并且不能确定 ttyname 是正确的, 那么就在那个串行端口上试试其它 ttyname。如果还不能看到在这个终端有日期显示, 那么就做如下尝试:

- 确认这个终端的电源是接通的。
- 检查一下电缆线的配置是否正确。如果使用的串行端口有一个25针的接头 (DS-25), 就请通读一下前面的步骤2。针2、3、7连接得是否正确? (请注意2、3、7以外的其它针不必用)。

如果系统或扩展卡有一个9针接头 (DB-9), 那么必须使用一个9针转25针的接头。在硬件手册中找到9针转25针接头的有关内容。

- 检查一下终端的配置情况。看前面指示中的步骤2。试着改变一下波特率。
- 检查一下串行端口的设备开关。如果用的是一个多端口卡, 那么就试一试那个卡上的其它端口。
- 把终端连接到一个标准串行端口 (COM1或COM2), 看看终端和电缆线工作是否正常。如果所使用的已经是一个COM端口了, 那么把终端切换到另一个端口上试试。

如果能够成功地安装另一个终端, 那么就在能工作的和不能工作的两个终端之间交换一下某个部件。这样做可以帮助你隔离出一个硬件故障。请注意, 有些硬件可以在DOS下工作, 但不能在UNIX下工作。

6. 当日期在终端上显示出来之后, 可以用如下命令使端口生效:

```
enable ttyname
```

这个 enable 命令可以启动一个 getty 进程, 此进程显示如下注册提示符:

```
login:
```

如果没有看到“Login:”提示符, 那么就键入如下命令来核实一下 getty 是否正在这个端口上运行以及软件配置是否正确:

```
ps-t ttyname
```

屏幕上应显示出与下面的例子类似的一条信息, 在 COMMAND 这一列列出了“login”或“getty”:

```
PID  TTY  TIME  COMMAND
```

7. 如果已经多次键入 enable 和 disable 命令，那么很可能新的 getty 已不能再在那个端口上产生了。如果是这样，就关闭系统，再重新引导，在多用户模式下以根 (root) 身份登录进入系统，然后再试一次。

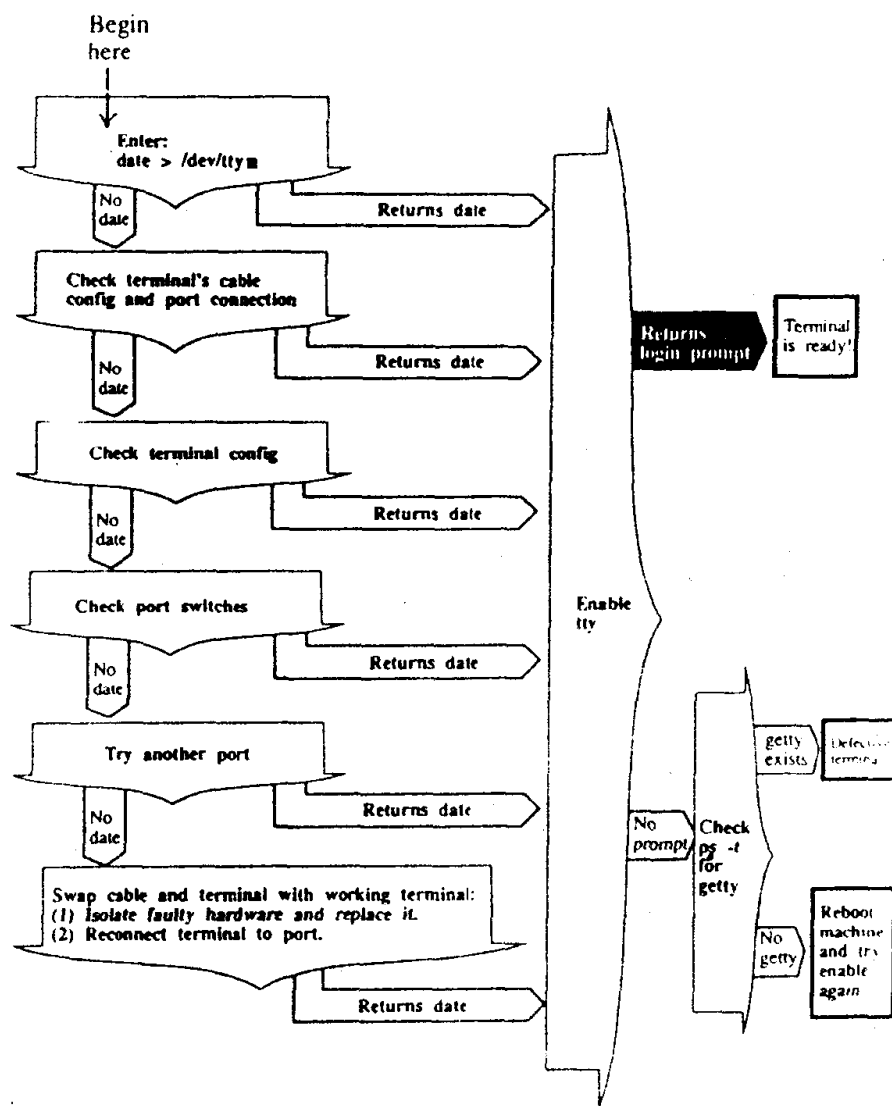


图 6-1 检查终端的连接

## 6.2 改变缺省的终端线特性

本系统可以自动适应若干种不同的终端波特率和设置。可以显示注册信息 geey (M) 的那个程序，从电缆线上读入这些终端线的值，并尝试各种设置，直至找到一个成功的设置，并能使用户登录到系统中。这张表为不同的几类终端线提供了几种缺省设置。

getty 作为注册进程的一部分能够自动执行。这张终端设置表可在一个叫做 /etc/gettydefs 的文件中找到。管理员可以编辑 gettydefs 来增加终端特性的设置或改变现存的终端特性。

### 6.2.1 文件 gettydefs

文件 /etc/gettydefs 中包含有 getty 用以设置波特率等终端线特性的信息。这个文件

是以表的形式存在的。每个表项都被分成五个域和一个任选域。这些域包括：

label#initial-flags#final-flags#login-prompt#next-label [#login program]

这些域的含义分别为：

- |                |   |
|----------------|---|
| label          | 给getty指定gettydefs登记项。这可以是一个数或是一个字母。这个标号与文件 / etc / inittab 中的线模式域相对应。init 把线模式作为一个变元传递给 getty。  |
| initial、 flags | 当getty第一次建立连接时，设置终端线特性。getty对于列于termio (M) 中的标志进行识别。在这个域中唯一的一个标志经常用来设置波特率。例如， B300 将把速度设置为 300。   |
| final-flags    | 设置终端线特性正是在getty执行login之前。这些终端描述了终端线的操作特性。波特率被再次设置。其它常用的标志包括 SANE (这个复合标志用于把一些终端特性设置为合理的值)。TAB3 (扩展制表键的区域)、IXANY (使某字符成为重启动输出)，以及 HUPCL (在最后关闭时挂起终端线)。这些标志可以以任意顺序输入。   |
| login-prompt   | 包含欢迎用户注册的信息。这个域将按其输入时的原样被显示出来，包括空格和制表符。<br>有几种字符串可被识别，包括：<br>\n 换行符           \t 制表符<br>\r 回车符           \f 换页符<br>\v 纵向制表符      \b 退格符<br>\nnn (3个八进制数) 指定的ASCII码字符。 |
| next-label     | 为getty指定在当前这个不成功时gettydefs文件中的下一个标号。如在登录进入系统时用户按了 BREAK 键， getty 就要试验下一个标号。各组登记项，譬如拨号线路或 TTY 线路，应该形式一个封闭设置，这样如果没有一个登记项成功， getty 就会转回最原先的登记项。                          |
| login-program  | 它是实际使用户登录进入系统的程序名称。缺省程序为 / etc / login。这个域是任选的。<br>如果加一个前缀AUTO， getty就不会提示输入用户名了，而是使用设备名 (例如 tty03) 作为用户名并且立即执行 login-program。  |

每个域之间都以一个#符号分隔，并且在 gettydefs 中的每个登记项之间都以一个空行分隔。

在 gettydefs 中的一个登记项类似：

4#B1200#B1200 SANE TAB3 HUPCL#login: #2#AUTO / etc / login.new

这里是这一行中每一部分的解释：

- 数字4供getty识别这一登记项。
- 下一个域设置波特率为1200。
- 第三个域给出了波特率 (B1200)， SANE (一些特性的组合标志) 和HUPCL (在最后关闭时挂起终端线)。

- login: 作为注册提示符出现。如果这个设置不成功, getty就会进入到gettydefs中的标号2。
- AUTO通过执行 / etc / login.new使用户登录。

如果最后一项也包含一个文件名, 那么那个注册程序就会执行。(请注意, 这个文件名和相应的注册程序是由用户建立的。) 例如, 包含一个象 / etc / dial\_login 这样的文件, 那么一个连接调制解调器的终端线就可使用。它将设置用户 ID, 获取一个口令, 使用户合法化, 然后成为一个用户。它可能除了一个帐号口令还需要一个系统口令, 甚至还需要一组包括在文件 / etc / default / dial\_login 中注册环境变量的特殊设置。

## 6.2.2 改变 gettydefs 文件

文件 / etc / gettydefs 有一组拨号线和终端线的登记项设置。这些不同的设置对应于 / etc / inittab 中的线模式的设置。init 程序把线模式作为一个变元传递给 getty。

可以编辑 gettydefs 来增加新的终端设置或改变已存在的设置。例如, 有如下的终端线的设置:

```
4#B2400 HUPCL#B2400 CS8 SANE HUPCL TAB3 ECHO IXANY # \r\n@!login: #5
5#B4800 HUPCL#B4800 CS8 SANE HUPCL TAB3 ECHO IXANY # \r\n@!login: #6
6#B9600 HUPCL#B9600 CS8 SANE HUPCL TAB3 ECHO IXANY # \r\n@!login: #4
```

要改变这个样本文件 gettydefs, 使其第一个波特率变为 1200, 就可以这样做:

1. 键入一个文本编辑器来编辑文件 gettydefs 的第一行。
2. 把第一和第三个域从 B2400 改为 B1200。
3. 把 gettydefs 存盘并退出编辑器。

这个样本文件应该像下面这样:

```
4#B1200 HUPCL#B1200 CS8 SANE HUPCL TAB3 ECHO IXANY # \r\n@!login: #5
5#B4800 HUPCL#B4800 CS8 SANE HUPCL TAB3 ECHO IXANY # \r\n@!login: #6
6#B9600 HUPCL#B9600 CS8 SANE HUPCL TAB3 ECHO IXANY # \r\n@!login: #4
```

还可以向 gettydefs 中加入另外的终端线设置。用于终端设置的标志和允许值被列于 stty (C) 中。

当增加一个新登记项时, 必须保证在文件 gettydefs 中的各组登记项能形成一个封闭的设置, 这样最后一个登记项的 next-label 就把 getty 指回到本组的第一个登记项。

要往前面的样本文件 gettydefs 中加入一个波特率为 300 的登记项, 应按如下步骤来做:

1. 键入编辑器来编辑文件 / etc / pettydefs。
2. 找到要插入的这个新设置在 gettydefs 中的位置。登记项的顺序没有关系, getty 只看标号。在这个例子中, 新登记项是文件中的最后一项。

3. 在文件的最后一行之后插入一个回车符, 并且输入下面这行:

```
7#B300 HUPCL#B300 CS8 SANE HUPCL TAB3 ECHOE IXANY # \r\n@!login: #4
```

4. 为了把标号 8 加到这组标号中, 就必须把登记项 6 的下个标号 (next label) 域改为 7。

```
6#B9600 HUPCL#B9600 CS8 SANE HUPCL TAB3 ECHOE IXANY # \r\n@!login: #7
```



现在, getty 就从标号 6 指向标号 7, 然后又回到 4。

5. 退出文本编辑器, 同时把这个修正过的文件 gettydefs 存盘。

新的 gettydefs 就变成这样了:

```
4# B1200 HUPCL #B1200 CS8 SANE HUPCL TAB3 ECHOE IXANY # \r\n@!login: #5
5# B4800 HUPCL # B4800 CS8 SANE HUPCL TAB3 ECHOE IXANY # \r\n@!login: #6
6# B9600 HUPCL # B9600 CS8 SANE HUPCL TAB3 ECHOE IXANY # \r\n@!login: #7
7# B300 HUPCL # B300 CS8 SANE HUPCL TAB3 ECHOE IXANY # \r\n@!login: #4
```

### 6.2.3 检查终端设置

每当在 gettydefs 中改变终端线设置或增加新的登记项时, 应该检查一下以确认这个新值对于 getty 有意义。要做这件事, 应使用带检查选项 -C 的 getty 命令以及这个文件名。

例如, 要检查 gettydefs, 输入:

```
getty-c /etc/gettydefs
```

这个文件被扫描一遍, 并显示出结果。如果在 gettydefs 中的某些值和设置是不允许的, getty-c 就会对它们进行报告。

### 6.2.4 改变串行线的操作

一旦用 enable 命令使一个终端能够工作, 系统就会自动给串行线的操作特性设置一组缺省值。有时, 这些值不能与这个终端所要用的值相匹配, 因此, 必须改变这些值, 使得系统和终端之间能够进行通信。可以用 stty 命令来显示一条串行线的操作特性。如果要改变一个已经过使能的端口的特性, 就应该使用 gettydefs 文件中的登记项, 而不是下面所给的 stty 命令。

---

注: 使用 stty 命令对一个端口所进行的任何设置只在这个端口打开的时间内有效。举一例子, 如果想改变 ttyza 的波特率, 并且 ttyza 还没有生效, 那么 stty 命令首先打开这个端口, 然后改变这个端口的设置, 并且最后关闭端口。当它在了后关闭这个端口时, 设置又回到原来的状态。在这部分后面的 stty 命令中, 使用 while 循环正是为了避免 stty 的这种做法。如果以 < /dev/ttyname 重新定向 stty 的输入, 那么它就会作用于当前的这个已经打开的串行线。在这种情况下, 因为在 stty 命令之后这条串行线保持打开状态, 因此所进行的设置也保持在那里。

---

还可以通过在与某串行线相连的终端上键入如下命令来显示这条串行线的当前的操作特性:

```
stty
```

如果不可能在那个终端上登录, 还可以使用其它终端来显示它的特性。在另外一个终端上以超级用户的身份登录, 并键入:

```
stty < ttyname
```

这里 ttyname 是与这条串行线对应的设备特殊文件的名称。例如, 这个命令显示出叫做 /dev/ttyla 的串行线的当前特性:

```
stty < /dev/ttyla
```

这个命令显示出关于这条串行线的波特率，奇偶性模式，以及其它信息。

对于一条串行线的最常见的改变是改变其波特率。这经常都是在与另外一条串行线相连的终端上来做的，因为改变波特率会破坏终端与系统之间的通信。在改变波特率之前，必须知道这个终端当前的波特率。一旦知道了这个波特率，就在另一个终端上以超级用户的身份登录，并键入：

```
(stty baud-rate; while:; do sleep 3600; done) < ttynome &
```

这里 `baud-rate` 是这个终端的当前的波特率，`ttynome` 是与想改变的串行线相对应的设备特殊文件的名称。波特率必须在以下范围内：50, 75, 110, 134, 150, 200, 300, 600, 1200, 2400, 4800 和 9600。例如：

```
(stty 9600; while:; do sleep 3600; done) < /dev/ttyla &
```

这个命令把串行线 `/dev/ttyla` 的波特率改为 9600。请注意，“小于”号 (<) 是用于从另一个终端对本串行线进行显示和设置的。

另一个常作的改动是系统通过这条串行线进行输入和输出的方式。这个改动经常是在与本串行线相连的终端上进行。例如：

```
stty tabs
```

这个命令使得系统以空格来扩展制表符（用在自己没有扩展制表符的终端上）。另一个例子是：

```
stty echoe
```

这个命令使得当使用 <BKSP> 键删除一个字符时，系统把这个字符从终端上抹去。

请注意，`stty` 命令还可以用来使一条串行线适应一个不常见的终端、需要奇偶生成和检测的另一类串行设备，或特殊的输入和输出过程。

### 6.3 设置终端类型

UNIX 系统要求在一个终端上进行任何工作之前，必须对这个终端进行明确定义。设置终端类型的最可取的方法是给 `TERM` 变量赋予一个类型，这个特殊的环境变量与你所使用的终端相关，这个终端的特性都在 `/etc/termcap` 文件中列出。这些特性告诉系统如何解释终端的键，以及如何在终端屏幕上显示数据。

如果使用的是 Bourne 或 Korn shell (`sh (c)` 或 `Ksh (C)`)，则 `TERM` 的赋值形式如下：

```
TERM = termtyp; export TERM
```

如果使用的是 C shell (`csh`)，则 `TERM` 的赋值形式如下：

```
setenv TERM termtyp
```

`termtyp` 必须是与 `/etc/termcap` 文件中所定义的终端之一相关的名称。赋值必须在正在设置其类型的那个终端上输入。

例如，要在 Bourne shell 中把终端类型设置为“ansi”，就到要设置的终端上，在 shell 提示符 (“\$”) 下输入：

```
TERM = ansi; export TERM
```

再按回车键。在 C shell 中，在 shell 提示符 (“%”) 下输入：

```
setenv TERM ansi
```

再按回车键。

如果不能肯定要给 `termttype` 用哪个名字，可以显示 `/etc/termcap` 文件来看这些名字。其中列出了 `/etc/termcap` 文件所支持的所有终端。要显示这个文件，键入：

```
more /etc/termcap
```

再按回车键。

可以让系统在登录时自动定义终端类型，只要在 `.profile` 文件中包括 `TERM` 的赋值就可以了。

如果让系统设置终端类型，在与普通终端不同的终端上登录时必须小心。系统无法检查对于这个给定的终端的终端赋值是否正确，只认为它与普通终端是相同的。如果不是这样，必须手工设置终端类型。

#### 6.4 自动设置终端类型

如果想在注册时自动设置终端类型，请按如下步骤来做：

1. 在这个终端上登录，并且通过键入如下命令来得到所使用的终端的名字：

```
tty
```

2. 以根 (`root`) 的身份登录，并且用一个文本编辑器来编辑文件 `/etc/ttytype`。把与这个终端相关的串行线的终端类型域改为所希望使用的终端类型。对于控制台也这样做。如果你想给 `/dev/ttyla` 的终端类型设置为 `'wy50'`，那么就那样编辑 `/etc/ttytype`：

```
wy50 ttyla
```

3. 然后，在用户的启动文件中必须写入一个合适的 `tset (C)` 命令行来自动设置终端类。每个 `c shell` 用户的 `.login` 文件中，加入下面这行：

```
tset-s-Q> /tmp/test $$; source /tmp/tset $$; /bin/rm /tmp/test $$
```

一定要从 `.login` 文件中去掉涉及 `TERM` 和 `TERMCAP` 的缺省 `setenv (C)` 命令。

在每个 `Bourne shell` 用户的 `.profile` 中，加入：

```
eval 'tset-s'
```

一定要从 `.profile` 文件中去掉已有的 `tset` 命令行。

4. 让全部用户都退出系统，然后再登录，以此来测试这个新的终端设置。在他们登录之后，让他们键入 `env (C)` 命令来核实一下新的终端类型：

```
env
```

#### 6.5 去掉一个终端

有时可能需要从系统上去掉一个终端，例如，如果想用其它一些设备替代它。在去掉一个终端之前，必须用 `disable (C)` 命令来禁止它。

要去掉一个终端，请按如下步骤执行：

1. 关掉这个终端的电源。
2. 在另一个终端上以超级用户的身份注册。
3. 使用 `disable` 命令禁止这个终端。这个命令形式如下：

```
disable ttyname
```

其中 `ttyname` 是这个终端所连接的串行线的名字。例如：

```
disable / dev / ttyla
```

这个命令使得连接于串行线 `/ dev / ttyla` 上的终端被禁止。

4. 把这个终端从系统上摘下来。

以前连着这个终端的这条串行线现在自由了，可以接受其它设备。

## 6.6 建立串行控制台

可以配置一个串行设备，而非显示适配器，来作为系统的控制台。`boot (HW)` 程序在引导时按如下过程来设备缺省的控制台：

1. `boot` 程序在 `/ etc / default / boot` 文件中找登记项 `SYSTTY = x` (`x` 是系统控制台设备的名字)。

2. 如果 `SYSTTY` 这个登记项没找到，或是文件 `/ etc / default / boot` 不可读，`boot` 就会在你的系统中查找显示适配器，并把它定为系统的控制台。

3. 如果没找到显示适配器，`boot` 就会找 `ttyla`，并把这个串行端口设置为 9600 波特率，8 个数据位，1 个停止位，无奇偶校验位，并且把它作为系统控制台。

要建立一个串行控制台，就要在文件 `/ etc / default / boot` 中建立如下登记项（其中 `x` 为“0”表示一个显示适配器，为“1”表示一个 COM1 串行端口）：

```
SYSTTY = x
```

要在命令行中改变这个系统控制台，可以在引导提示符下键入 `systty = x`（其中 `x` 为“0”表示显示适配器，为“1”表示 COM1 串行端口）。这不能在 `/ etc / default / boot` 文件中建立或改变 `SYSTTY` 登记项。

## 6.7 利用 `mscreen` 使用串行多屏幕

如果熟悉 `multiscreen (M)` 这个能够在控制台上提供多个分离的注册屏幕的特点，那么就能在一个终端上使用类似的特点。具有多屏幕存储页的终端可使用分离的屏幕，屏幕都有一个不同的注册对话区，就好像在所用的设备上有一个终端而不是一个。

在一个 `Wyse60` 终端上，两个完整的活动屏幕的内容都能够被很容易地保存。在 `Wyse60` 上的第三个屏幕的使用将在下面进行讨论。这就意味着使用两个屏幕就很像有多于一个终端。在各屏幕上都提供了注册对话区的全部功能特性，并且当它被使用时前面所执行过的命令（或它们的结果）也在这个屏幕显示出来。这一节中的所有例子都以 `Wyse60` 为基础，使用它的两个屏幕存储页。

还可以限制能够获得的多屏幕的数目。`mscreen` 实用程序所提供的对多个终端对话区的访问，非常像在一个终端上进行登录。这些对话区存在于“伪 `tty`”上，而不是由终端或调制解调器通常所使用的 `tty` 设备。一个 `tty` 是一个特殊的文件，它直接与一个用于与设备（如终端或打印机）进行通信的硬件设备相关。`tty` 可以在 `/ dev` 目录中看到，它们以文件的形式存在，名字为 `tty`，后面还跟有一个数字和一个字母。

一个伪 `tty` 是一个不与实际的硬件相关的设备，并被用来模拟一个实际 `tty` 的功能。网络产品的用户应该已经熟悉伪 `tty` 了，因为它们是用来在远程机器上进行登录的设备。

一个伪 tty 由两个软件设备来表示，它们都列在 /dev 中，表现形式划“pty”和“ttyp”，每个的后面都跟着一个数字。前一个叫“主”tty，而后一个叫“从”tty。在这两个之间，它们模拟了一个具备应有功能的 tty。

在安装时，系统允许最多有 16 个屏幕，而在安装期间，只建立了 8 个伪 tty。可以建立更多的伪 tty，这样就允许在系统上有更多的屏幕，程序见图 6-2。这个对可用的伪 tty 的限制可以通过一个可配置的参数 NSPTTYS 来控制，它决定了系统可识别的伪 tty 的数目（在安装时的缺省设置为 16）。最大值是 32。如果打算用多于 16 个伪 tty，除了建立更多的伪 tty 之外还必须增大这个参数。请注意在图 6-2 中的“for x in 8910...”这一行一直扩续到 23，这表示它另外又增加了 16 个伪 tty（0-7 是已经存在了的）。如果不需要这么多伪 tty，就把这一行缩短。如果想要 32 这个最大值，也可以把它扩展到 31。

```
:
# Type 'l /dev / ptyp *' You will get several lines similar to:
# crw----- 1 bin terminal 59, 0 may 26 08 : 21 /dev / ptyp0
#
## Set the 'major__master' variable (below) equal to the major number
## given.
major__master= 59
#Type 'l /dev / ttyp *' You will get several lines similar to:
### crw----- 1 bin terminal 58, 0 may 26 08 : 21 /dev / ttyp0
#
## set the 'major__slave' variable (below) equal to the major number
## given.
major__slave##58

for x in 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
do
    ## make master
    /etc / mknod / dev / ptypsx c major__master x
    chown bin / dev / ptypsx
    chgrp terminal / dev / ptypx
    chmod 600 / dev / ptypsx

    ##make slave
    /etc / mknod / dev / ttypx c major__slave x
    chown bin / dev / ttypsx
    chgrp terminal / dev / ttypx
    chmod 600 / dev / ttypx
done
```

图 6-2 makeptys 的 shell 描述

按如下步骤来配置更多的屏幕：

1. 对核心进行配置，使其支持更多的伪 ttys。阅读“调整系统性能”一章中“使用配置重新分配核心资源”，找到如何运行 configure (ADM) 的说明。选择第 3 项“Files,

Inodes, and Filesystem”，并把参数 NSPTTY 设置为所希望的数。按照这个说明，在继续向下进行之前重新连接核心。

2. 下面，把图 6-2 所描述的内容拷贝到一个叫做 makeptys 的文件中。要使这个 shell 程序可执行，要键入如下命令：

```
chmod 744 makeptys
```

3. 要运行这个 shell 程序，必须先确认自己是否是以根的身份登录的，并如下执行这个 shell 程序：

```
./makeptys
```

4. 在增加了这些设备之后，必须对系统进行配置，使得在每个伪 tty 上都启动一个 getty 进程。这个工作可通过向文件 /etc/inittab 和文件 /etc/conf/cf.d/init.base 中加入一些登记项来完成。表示前面的八个伪 tty 的登记项如下：

```
P0: 2: respawn: /etc/getty ttyp0 m
P1: 2: respawn: /etc/getty ttyp1 m
P2: 2: respawn: /etc/getty ttyp2 m
P3: 2: respawn: /etc/getty ttyp3 m
P4: 2: respawn: /etc/getty ttyp4 m
P5: 2: respawn: /etc/getty ttyp5 m
P6: 2: respawn: /etc/getty ttyp6 m
P7: 2: respawn: /etc/getty ttyp7 m
```

---

注：前面的8个伪tty可以用“off”来代替“respawn”。（鼠标器要求禁止伪tty。）可以通过使能各个设备把它们改为 respawn，例如：

```
enable p0
```

这就使伪tty /dev/ttyp0生效。（如果已经把它们用于鼠标器，就不要使它们生效了。）

---

还可以把它们拷贝并修改为其它设备。一定要两个文件都修改。例如，如果想加另外八个伪 tty，就必须把下面各行加入两个文件中：

```
P8: 2: respawn: /etc/getty ttyp8 m
P9: 2: respawn: /etc/getty ttyp9 m
P10: 2: respawn: /etc/getty ttyp10 m
P11: 2: respawn: /etc/getty ttyp11 m
P12: 2: respawn: /etc/getty ttyp12 m
P13: 2: respawn: /etc/getty ttyp13 m
P14: 2: respawn: /etc/getty ttyp14 m
P15: 2: respawn: /etc/getty ttyp15 m
```

5. 下面，键入如下命令：

```
/tcb/bin/ttys-update
```

这就保证了所加的能被系统的安全机制认可。

给系统增加更多的 mscreen 能力虽然能够提高用户的工作效率，但是，过多的好东西会降低系统的速度。如果一个系统有 10 个用户，每个用户都使用两个屏幕，那么这就使系统好像在为 20 个用户服务。在决定允许系统可允许多多少个多屏幕时，必须要考虑系统的性能以及谁应该能够使用它们。

### 6.7.1 问题的解决

不像许多其它的实用程序，mscreen 复杂的责任要求有一些使其能够正常发挥作用的`条件`。按照下面的建议来进行工作，能够避免新的 mscreen 用户通常遇到的一些错误。

在准备使用 mscreen 时，一定要使终端运行这个程序。mscreen 使用 `/etc/mscreencap` 文件来决定如何针对独特的终端来改变屏幕图像。在 UNIX 系统软件中 `/etc/mscreencap` 只提供了几种终端；这并不是说 mscreen 不能工作于其它终端，它是能的。只需要在使用终端之前对 `/etc/mscreencap` 进行配置就行了。如果在一个在 `/etc/mscreencap` 中没有登记项的终端上运行 mscreen，mscreen 就会失败。

如果肯定终端能够运行 mscreen，并且也有一个正在工作的 mscreencap，但 mscreen 依然失败了，那么就要检查一下如下常会发生的错误：

- 建立伪tty。

如果系统上的伪tty都正在被使用着，就必须建立更多的伪tty。可以使用图6-2中的 shell 程序来建立更多的伪 tty。

- 核实开关的设置。

确认 `/etc/mscreencap` 中有关所用终端的登记项是否正确。使用 `/etc/mscreencap` 中的一上例子来检查一下功能键输出序列与一个特定 mscreen 命令的映射情况。必须要在各个要用的屏幕中分别登录一次。

- 杀掉mscreen进程。

如果你正在测试一个mscreencap中的登记项，并且你在某些屏幕中遇到了问题，那么你应该按下述步骤来做：

1. 检查一下正在运行的进程：

```
ps-username
```

2. 杀死所有 mscreen 进程：

```
kill -9 process__numbers
```

### 6.7.2 高级的使用

很多用户对于所提供的 mscreen 已经很满意了。而对于高级的 mscreen 用户，或者对于对学习更多 mscreen 和操作系统知识有兴趣的人，这里还有一些使用和扩展 mscreen 的技巧。

在 mscreen (M) 的命令说明中，建议使用从 C shell 中自动调用 mscreen 这个方法，图 6-3 的内容中也允许在 Wyse 60 上有三个全功能的多屏幕，并且给 mscreen 用户增加了一定数量方便的特性。图 6-4 也给了 Bourne 和 korn shell 的 .profile 文件以同样的内容。请注意，这些例子是被设计并增加到你的 .login 或 .文件末尾的，并且代替某些已有的 tset 内容。

```
#  
#Example material for the end of a C-Shell .login file.  
#  
# If logging in via pseudo-tty, suppress terminal initialization.
```

```

set ttyname = 'tty'

#Set init to null, initially.
setinit = " "

set noglob

#Reset init to the value "-I" when logging in on a pseudo-tty to
#suppress the tset terminal initializations string.

if ('expr sttyname: " / dev / tty" > 0) set init = "-I"
set term = ('tset -m ansi: ansi -m wy60: wy60 -m: ? wy60 -r -S -o Sinit')
setenvTERM Sterm [1]
setenvTERMCAP Sterm [2]          #terminal data base
unset noglob term
#Put WYSE 60 in ECON-80 mode during initial log in process.
if ("Sinit" != "-I" && "STERM" = "wy60") / bin / echo "\033eG\c"

#Set the prompt to indicate the tty number of the current
#mcsreen and command.
set prompt = "'expr Sttyname: ' / dev / ) ' 1%"

#Release the local variables used.
unset ttyname init

#Run mscreen and logout if the 'stop'key (defined as S-F9 in
#the default / etc / mscreencap for wy60) is pressed. This string
mscreen -n 3
if (Sstatus == 0) logout

```

图 6-3 .login 程序

许多 termcap 登记项 (包括 wy60) 把清除屏幕缓冲区 (mscreen 用以存储多屏幕内容的) 作为初始化过程的一部分。图 6-3 和 6-4 中, tset (C) 只在第一个注册过程期间发送初始化指令串。当在伪 tty 上登录时, tset 在被调用时带有 -I 标志。这是通过给 tset 行增加 init 变量来实现的。在 tset 第一次运行时, init 的值等于 " ", 即不给 tset 命令增加任何东西。当以后再运行它时, init 的值就为 "-I" 了, 即给 tset 增加了一个选项。

第一次注册过程中, 紧跟着 tset 命令, 字符串 "\033eG\C" 被显示出来。这个逃逸序列把 wyse60 的 COLUMNS 设置改为 ECON-80 模式。这些设置的组合为使用三个屏幕释放了足够的屏幕缓冲区。更加方便的是, 把用户的提示符设置为能够显示当前的从伪 tty 序号, 这可以使用户很容易地掌握正在使用哪个屏幕。

如果所用的不是一个 Wyse60 终端, 依然可以设置提示符来标志当前的屏幕, 并在检查 shell 返回码时自动调用 mscreen, 正如图 6-3 和 6-4 所描述的那样。



```

#
#Example material for the end of a Bourne shell.profile file.
#
ttyname='tty'

init=""

if [ `expr Sttyname: " / dev / tty" i-gt"0` ]
then
    init=""-I"
fi
eval `tset -m ansi: ansi -m wy60: wy60 -m: ? wy60 -r -s -Q Sinit`
export PATH

if [ "Sinit"="-I"-a"STERM"="wy60" ]
then
    / bin / echo "\033eG\C"
fi
PS1=""`expr Sttyname: ' / dev / ) 'iS"
unset ttyname init
mscreen -n 3
if [ "$?" ] = "0"
then
    exit
fi

```

图 6-4 .profile 程序

## 第七章 硬件的兼容性

### 7.1 兼容的硬件

SCO UNIX System V / 386 可以用于许多种配置的 PC 系列机上, 包括工业标准总线 (ISA) 计算机, 扩展标准总线计算机 (EISA) 以及微通道结构的计算机。在开始安装操作系统之前, 必须先将最基本的硬件配置以及扩展板准备好。如果要增加另外的扩展板, 则应保证板上所有开关的设置状态与制造厂家硬件手册上的要求完全相符。

### 7.2 如何使用本章

要列出与你的机器相兼容的硬件清单, 你必须知道你机器的分类、机器使用的处理器以及机器的总线标准是 ISA、EISA, 还是微通道的。(微通道结构是指与 IBM PS / 2 兼容的计算机), 你还应该了解你的系统是否有升级。比如有没有非 ST506 的 AT 磁盘控制器。要找出机器的分类, 应阅读下一节“硬件支持”。在本章中有一节“一般兼容性标准”, 其中讨论了 SCO UNIX System V / 386 的一般硬件兼容性问题。在系统中安装任何附加的硬件之前都必须仔细阅读这一节。

### 7.3 硬件支持

SCO UNIX System V / 386 工作于标准的 386 为 CPU 的 PC 上, 包括那些支持 Adaptec AHA-154X, Western Digital WD7000 SCSI 主适配器和 SMS / OMTI 862X Direct ESDI 磁盘控制器的 386 计算机。有些计算机到货时, 其硬盘只部分地格式化了。若你的机器也是这样, 则在开始安装 SCO UNIX System V / 386 之前应对硬盘进行底层格式化。有些计算机要求在特殊的开关设置状态下才能运行 SCO UNIX System V / 386, 对此应向其厂家代表索取正确的开关设置信息。

### 7.4 标准结构的 386 类机器

我们在下列机器上安装并使用过 SCO UNIX System V / 386 操作系统:

ALR FLex Cache 20 / 386

ALR FLex Cache 25 / 386

ALR FLex Cache 33 / 386 Model 150

ALR Microflex 7000

ALR Power Cache 486e

AST Premium 386

AST Premium 386 / 33

AST Premium 486 / 257

Laguna Systems PDQ 386

Mitac 386

ACER 1100SX

ACER 1100 / 20  
ACER 1100 / 33  
ACER System 32 / 20  
Compaq 386  
Compaq 386 33Mhz  
Compaq Partable III 386  
Compaq DeskPRO 386 / 20  
Compaq DeskPRO 386 / 25  
Compuadd 386  
CSS 386  
DEC station 320  
EasyData 386 model 333  
Epson Equity 386 SX  
Everex 386 / 20  
Goupil uniprocessor 25MHZ Tawer  
GRiDCase 1530  
Hertz 386 / 25  
Hewlett-Packard Vectra QS / 165  
Hewlett-Packard Vectra QS / 20  
Hewlett-Packard Vectra RS / 16PC  
Hewlett-Packard Vectra RS / 20PC  
Hewlett-Packard Vectra RS / 20C  
Hewlett-Packard Vectra RS / 25C  
Intel 302  
ITT 386  
Legacy 386 / 33  
Mitsuba 386  
Mitsubishi PC-386  
NCR 3386  
NCR 316  
NCR 316SX  
NEC Business Mate 386  
NEC Business Mate 386 / 33  
NEC Power Mate 386  
NEC 386 20MHZ  
Noble 386 from PC Discount  
Nokia Alfaskop System 10 m52  
Nokia Alfaskop System 10 m54 / m55  
Olivetti M380  
Olivetti M380 XP5  
Olivetti M380 XP7<sup>[††]</sup>  
Olivetti M380 XP9<sup>[††]</sup>  
PC Craft PCC 2400 386  
Siemens Data Systems Model WX2000  
Tandy 4000

Tatung TCS-8000 386  
Tatung TCS-8600 386  
Televideo 386 / 25  
Texas Instruments System 1300  
Toshiba T3100SX  
Toshiba T3200SX  
Toshiba T5100 386  
UNISYS PW2 Series 800 / 20  
Victor 386 25MHZ  
Wang PC 386  
Wyse 386  
Zenith Z-386 / 16  
Zenith Z-386 / 25  
Zenith Z-386 / 33  
Zenith Turbos PORT  
Zenith 386 / 20 model ZVB 3524ED

---

[注]: 在 XP7 和 XP9 上安装 SCO UNIX System V / 386 时, 如果“软盘访问速度”选项被设置为“slow”方式, 则系统不能从软盘启动。因为该选择的“slow”方式是为 MS-DOS 环境提供的, 不能用于 SCO UNIX System V / 386 环境。要使你的系统从软盘启动, 应保证该选项的设置为“fast”。

---

另外 Intel 支持的 286AT 兼容机也支持 UNIX System V / 386, 下面的机型声称可以支持本操作系统:

keen 2500  
Cheetah CAT-386  
Corvus 331  
Arnet Multiuser 386  
NCR PC916

## 7.5 微通道结构的 386 机器

我们还在以下机型上安装过 SCO UNIX System V / 386:

Apricot Qi  
ALR 486 MC Model 150  
IBM PS / 2 Models  
55-061  
70-E61  
70-121  
80-041  
80-071  
80-111  
80-311  
Tandy 5000MC  
Olivetti P800

## 7.6 一般兼容性标准

这一节说明了哪些硬件可以与 SCO UNIX System V / 386 一起使用，它包含工业标准 (ISA) 总线结构及微通道结构的硬件信息。许多由独立厂家提供的设备也可以用，但要求提供这些设备的附加软件才行。如果你的机器是前面列出的一系列机型之一，则可以直接安装运行 SCO UNIX System V / 386。如果没有特别说明，就不必增加什么硬件设施，或者改变开关设置状态。

### 7.6.1 数学协处理器芯片

你的计算机中可能有 80387 数学协处理器，它是由 SCO UNIX System V / 386 自动识别并支持的，可以增强机器浮点运算能力。在 80486 CPU 中有一个片内协处理器，也是作为 80387 识别和使用的。

选用处理器芯片时应按照机器厂家的建议选择与你机器速度匹配的芯片。

系统在引导时如果检测到数学协处理器的存在，就会显示如下屏幕信息：

```
%fpu-35-TYPE = 80387
```

应该注意在系统主板上的开关应该设置为允许 80387 中断的状态，用厂家提供的 SETUP 软盘保证系统有协处理器的配置。此外还应保证系统的诊断程序能够检查出协处理器芯片的存在。主板上有关开关的设置状态在系统硬件手册中可以查到。

在一些母板上，操作系统会错误地认为 80387 协处理器芯片已经在板上，而实际上该芯片根本没有安装。在使用 Intel 母板的机器时这个问题尤为严重。如果你的机器有这种情况，应保证 E48 块与 E49 块没有通过跳线连在一起。

### 7.6.2 内存卡

一般来说，SCO UNIX System V / 386 支持大多数内存卡。如果在使用时出现“Panic: Parity”一类的错误，常常是由于内存芯片或内存卡的质量不合格所致。尤其是对早期 386 机器上所用的 32 位静态 RAM 芯片这个问题就更为突出。

在用内存卡时，应查阅计算机及内存卡的硬件手册，找到主板及内存卡的正确的开关设置状态。SCO UNIX System V / 386 支持的内存大小为 64 Mbytes。

我们郑重地推荐使用 32 位存储器。因为 16 位存储器的速度太慢，会降低机器的整体性能。有些厂家至今尚未解决 16 位存储器有关的 DMA 事宜，这样的机器不能识别 16 位存储器。（比如 Zenith 的机器）。

如果你见到这样一条信息：

```
Panic: memory failure --parity error
```

这表示你某一部分硬件正发出“不可屏蔽中断”，如有可能应运行系统的硬件诊断测试程序。你还可以把内存卡取出来再重新插上去，检查内存卡上的芯片，看有没有引脚弯曲等问题。如果这些措施都不解决问题，或者你自己不愿亲自动手检查硬件，则可以寻求硬件专家的帮助。另外，出现这一问题的原因还可能不是你选用的存储器芯片的速度低于厂家推荐芯片的速度使得内存卡不能正常工作。

### 7.6.3 多功能卡

对大多数多功能卡来说，只要 COM1 和 COM2 与卡上串行口的标准定义完全兼容，则串行口一般都能正常工作。主存、并行口及其他硬件一般都能正常工作。

### 7.6.4 串行 I/O 板

本部分说明各种串行 I/O 板与 SCO UNIX System V / 386 一起工作的条件和效果。标准的单端口串行 I/O 板在 COM1 和 COM2 与这些串行口的标准定义完全兼容时可以正常工作。

要配置系统以适应你安装的串行板，应运行 `mkdev serial` 命令。见本书“增加多功能卡，内存卡及其他总线扩展卡”一章中有关 `mkdev serial` 的内容。

每一种多端口 I/O 板都是特殊的，SCO UNIX System V / 386 对每一种板都配有专门的驱动程序。只有那些带有状态轮询寄存器的板才能与所选择的高效驱动机制一起工作。对新增加的板应配备新的驱动程序来支持。

某些计算机或附加串行 I/O 卡使用 8250 串行 I/O 芯片。（对 386 机器我们建议使用 16450 芯片）。这种芯片的某些修改型不能正确处理中断。MS-DOS 不使用中断，所以这种芯片与 MS-DOS 协同工作不会有任何问题。SCO UNIX System V / 386 是一个多任务操作系统，因而要使用中断。

在使用 UUCP (C) 或 `cu` (C) 时，这类修改型串行 I/O 芯片会出现问题。当 UUCP (C) 不断丢失字符并产生不可杀死的 `Uucico` 进程时，以及 `cu` (C) 在高波特率下停止执行且无法退出时，就表明你的计算机使用的修改型的 8250 芯片出现了问题。

在使用带终端的串行端口时就很少出现这类问题。问题常常与高速串行输入有关。如果想使用 UUCP (C) 或 `cu` (C) 而你的计算机上又有这么一个芯片，建议你用 8250b 串行 I/O 芯片代替 8250a 芯片，或者使用一个带有串行口的多功能卡，并将其配置为 COM1 或 COM2，然后关闭内部串行口，或者避免在该口上实施高速串行输入。

所有的 COM1 板都应使用中断向量 4，所有的 COM2 板都应使用中断向量 3。请查阅硬件手册找出实现这两个地址的开关位置信息。

### 7.6.5 磁带驱动器与控制器

注意：在这里讲的磁带驱动器有时会以其他形式的名字销售。每台计算机只能支持一个卡式磁带子系统。（系统上也有可能配置一个卡式磁带驱动器加上一个小型卡式驱动器或者 QIC40 驱动器。）这一限制不适用于 SCSI 接口的磁带驱动设备。（一条 SCSI 总线上可支持多达 4 个的 SCSI 磁带驱动器）。对全尺寸卡式磁带，系统支持 QIC-24 格式。带支持只对字符设备提供支持，而不支持块设备。系统提供的磁带实用程序 `tape` (C) 可用于倒带、抹带、格式化等。

要给系统配置某种磁带设备，应运行 `mkdev tape` 程序。若没有给出 DMA 通道、中断和基址值，系统会选用缺省值。注意大多数地址都使用十六进制表示。如果你不用缺省参数值，应注意不要与安装的其他设备发生冲突。下面的几节中将指出你的系统可能会用到哪些中断。不能使用中断 0, 1 或者 6，因为这些中断已经被使用，不再允许附加设备使用。在 `mkdev tape` 的菜单中将 Irwin 设备看成“小型卡式磁带”，而将所有其他设备

看成“卡式”，“QIC40”和“SCSI”设备。Irwin 设备是不可配置的。

### 7.6.6 显示适配器和监视器

所有使用与标准适配器等价的适配器的显示适配器 / 监视器组合都可以在 SCO UNIX System V / 386 下运行。

SCO UNIX System V / 386 只支持两类视频适配器及两种显示器，一种是单显，另一种为彩显。只支持单个 VGA 显示适配器。

### 7.6.7 附加硬盘

只要磁盘控制器支持其驱动器，SCO UNIX System V / 386 就可以使用多种硬盘，包括标准的和非标准的。硬盘必须在电气上与磁盘控制器接口。只要控制器与总线的接口是兼容的，那么支持其他驱动接口，比如 RLL 或 ESDI 的控制器也能正常工作。

一般来说，MS-DOS 不支持非标准硬盘（即那些未在 ROM 固化 BIOS 中定义的硬盘）。

某些硬盘在出厂时只作了部分格式化（比如，Maxtor 1140 140Mbyte），这个问题在安装时就会暴露出来，那时 badtrk (ADM) 会报告说在硬盘某个特定位置之后的所有扇区都是坏的。你可以询问厂家硬盘是否完全格式化了。格式化硬盘的工具具有许多种可供选用。

#### • 标准硬盘

主板 ROM 中必须有磁盘类型登记型。磁盘的类型由磁头数、柱面数，每柱面（头）的磁道数、每磁道上扇区数及其它特性信息所决定。

请按照厂家的要求来设置开关或配置系统。

#### • 非标准硬盘

用户在安装过程中可以键入信息以覆盖 ROM 中的磁盘配置信息。如果不能确定该为你的非标准硬盘输入什么参数，可以向硬盘厂家索取这一信息。dkinit 程序（在安装过程中调用）将提示输入这些参数。

#### 7.6.7.1 柱面数多于 1024 的硬盘

SCO UNIX System V / 386 支持柱面数大于 1024 的硬盘，但有以下几条限制：

- 硬盘控制器支持柱面数大于 1024 的硬盘
- 硬盘不允许有 MS-DOS 分区
- 如果硬盘被指定为主硬盘（即引导硬盘），则根文件系统必须驻存在前 1024 个柱面中，这是因为如果引导信息存放于第 1024 柱面之后的柱面中，ROM BIOS 就不能访问这些信息了。

你可以将硬盘的其余部分用作对换区，和（或者）附加文件系统（如 /u）。如果你只安装了一个 UNIX 分区，则应保证根文件系统存放于头 1024 个柱面上。

比如，对于 DPT PM3011 控制器，SCO UNIX System V / 386 可以支持 2048 柱面，16 个磁头，63 个扇区，总磁盘容量达 1 gigabyte 的硬盘。

如果引导信息位于前 1024 个柱面之外，则在启动过程中“Loading.text”信息出现过后

不久，系统会显示如下信息：

error in text

如果发生这种情况，你必须重新安排你的文件系统，使它的存放位置符合上面讲到的标准。

#### 7.6.7.2 磁盘控制器与主适配器

SCO UNIX System V / 386 支持使用下列的 ST506 / ESDI 硬盘控制器和 SCSI 主适配器的组合：

- 一个或两个 ST506 控制器（对微通道结构的机器而言，只支持一个）
- 一个 ESDI 控制器
- 一个或两个 SCSI 适配器
- 一个 ST 506 控制器带单 SCSI 适配器
- 一个 ESDI 控制器带单 SCSI 适配器

应该注意，对 ST506-SCSI 组合的机器来讲，SCO UNIX System V / 386 必须安装在 ST506 上。这意味着如果你启动的是安装在 SCSI 磁盘上的 SCO UNIX System V / 386，而现在想增加 ST506 磁盘，你就必须用 ST506 作引导磁盘来重新安装 SCO UNIX System V / 386。

如果你的系统上只安装了 SCSI 磁盘，那么在安装操作系统之前正常你必须运行计算机的 setup 程序并将计算机设置成“没有硬盘”的状态。这么做是为了迫使计算机识别 SCSI 适配器。

下面综述了每个控制器 / 适配器所能支持的设备的数目：

控制器或适配器	支持设备最大数
ST506	每个控制器 2 个磁盘
ESDI	每个控制器 2 个磁盘
SCSI	每个适配器 7 个设备 每个系统 4 个磁盘 每个适配器 4 个磁带驱动器 每个适配器 4 个 CD-ROM 驱动器

应该注意，不管安装了几个适配器，最多也只能支持 4 个 SCSI 磁盘，另外还有适配器只能支持 7 个设备的限制。比如，若在一个适配器上已经安装了 4 个 SCSI 磁盘，则只能再增加 3 个设备。（磁带或 CD-ROM。）

#### 7.6.7.3 ST506 和 ESDI 控制器

许多的 ST506 和 ESDI 硬盘都可与 SCO UNIX System V / 386 一起工作。磁盘能否工作取决于使用的磁盘控制器卡类型。要使磁盘正常工作，控制器必须满足以下两条件：

1. 磁盘控制器必须与当前配置下的标准控制器完全兼容。
2. 不需要使用由厂商特别提供的软件该控制器就能工作于 MS-DOS 下。



如果控制器达到这两个要求，就应该能够工作，否则必定不能工作。

#### 7.6.7.4 SCSI 主适配器

SCSI 主适配器接受带有 SCSI 驱动器，包括硬盘，磁带驱动器和 CD-ROM 驱动器的设备。SCSI 总线上的每个设备都必须有一个地址。这里说的地址包括两个部分：用于控制器的地址和用于设备本身的地址。设备地址也就是通常说的逻辑部件地址 (Logic Unit address)。对于 SCO UNIX System V / 386 来说，头一个 SCSI 磁盘必须设置成 ID: 0 LUN: 0 (即控制器地址为 0, 设备地址为 0)。ID 号对应于磁盘上的跳线状态或开关设置。所以必须小心检查以保证硬盘设置的正确性。注意，主适配器通常使用的 ID 号为 7, LUN 号必须总是 0。

#### 7.6.8 CD-ROM 驱动器

SCO UNIX System V / 386 还支持下列作为 SCSI 设备配置的 CD-ROM 驱动器:

Toshiba XM-3201B

Sony CDU-6110-01<sup>[注]</sup>

---

注: Sony 总线的 DIP 开关应该按厂商的配置来设定。

---

mkdev high-sierra 命令把对 CD-ROM 文件系统的支持加到 UNIX 核心中，在配置 CD-ROM 驱动器时必须使用 mkdev cdrom 命令。在用 CD-ROM 驱动器联机时一定要记住插入一张磁盘。否则屏幕上会显示错误信息：“cannot open”。

#### 7.6.9 鼠标和其他图形输入设备

SCO UNIX System V / 386 支持下列图形输入设备:

Logitech Serial Mouse

Microsoft Serial Mouse

Mouse Systems PC Mouse

Microsoft Busmouse 或 InPort Bus Mouse

Logitech Bus Mouse

Olivetti Bus Mouse

IBM Personal System / 2 Mouse

#### 7.6.10 调制解调器与自动拨号

使用 UUCP (C) 和 CU (C), 所有完全兼容于 Hayes 调制解调器的调制解调器都能工作。缺省拨号设备为 Hayes Smartmodem 1200。我们推荐使用外部调制解调器。另外系统还提供 Racal Vadec 3451, 212, Hayes Smartmodem 1200 和 2400, Telebit Trailblazer 以及 MuttiTech Muttimodem 224EM 的自动拨号程序。对其它的自动拨号调制解调器，只要编写相应的拨号程序，或者修改现存在 /usr/lib/uucp 目录中的原有程序即可支持。

另外，新的 Honey DanBer UVCP 软件包在 Dialers 文件中包含了大量的拨号程序。

## 7.7 标准结构机器的硬件兼容性

本节和下一节将分别对标准结构的硬件和微通道结构的硬件的兼容性进行阐述。

### 7.7.1 网络卡

SCO UNIX System V / 386 支持下列在标准结构计算机上的 Ethernet 板：

- 3Com Etherlink 3C501 卡
- 3Com Etherlink II 3C503 卡
- Western Digital WD8003E 卡

这些卡的安装软件允许你安装多达 3 个的同样类型的卡。在安装过程中每块卡的设备驱动程序被配置上不同的参数，比如中断向量和 I/O 基地址。在选择这些参数时必须格外小心，以免与系统中的其他硬件发生冲突。在卡上设置跳线之前必须警惕可能发生的冲突。注意，使用 IRQ2 中断的网络板应使用编号 9。

应该用中断向量 2 来安装你的网络卡，这是在 SCO UNIX System V / 386 安装过程中为安装网络卡配置的缺省值。503 卡的中断向量可由软件选择，如果你选择缺省值 2，则硬件的配置会自动进行。为 WD8003E 和 3C501 卡设定的中断向量缺省值为 3 (IRQ3)，因此在安装操作系统之前你必须查阅自己网络卡的文献，并通过跳线选择中断向量 2。如果没有做这一步工作，将会导致第二个串行端口 (tty2a 或 COM2) 与网络卡之间发生冲突。如果你确实需要用中断向量 3 (IRQ3)，就只好使第二个串行端口无效，这可以通过安装完成之后修改 / etc / conf / sdevice.d / sio 文件来实现。你应将第二行中第二域里的“Y”改成“N”。

另外某些图形适配器与 3C503 和 WD800 3E 卡使用的地址也有冲突，特别是 Orchid Designer 和 Genoa Super VGA 板与这些网络板的冲突更为严重。

### 7.7.2 内存卡

我们在标准结构计算机上使用过以下的内存卡：

AMI SMART PACK2

AST

Just RAM / AT 8Mbyte Card (Monolithic Systems)

Quadram

Tecmar

Talltree Systems

Silicon Valley Systems

STB Rio Grande

Micrcn Chessmate

### 7.7.3 串行卡

SCO UNIX System V / 386 支持下列工业标准总线结构计算机上的串行 I/O 板：

AMI lamb4 and 8 Part

Arnet Controls 2, 4 and 8 Port (不支持 clock 选项)  
Arnet Twin Port  
AST Four PORT / XN  
Control Systems Hostess 4 and 8 Port Versions  
CTC Versanet 4 AT (4 Port) and 8 AT (8Port)  
Digiboard 4 and 8 Port  
IBM standard COM1 and COM2  
Kimtron Quartet 4 Port  
Olivetti RS232C Multiport Board  
Quadram Quad Port™ 1 and 5 Port  
Stargate Technologies OC 4400 (4 Port) and OC 8000 (8 Port)  
Tandon Quad Serial Card  
UNI sys 4 Port

我们郑重建议，多端口卡应使用 16450 串行 I/O 芯片来代替较慢的 8250 芯片。如果你遇到“双响应”问题，就是由慢速串行 I/O 硬件所致。

#### 7.7.4 附加硬盘

许多硬盘都可工作于 SCO UNIX System V / 386 下。对 ST 506 / ESDI 驱动器来讲，必须要得到 ROM BIOS 支持，或者在安装时必须将 ROM 参数输入计算机。检查你计算机的硬件参考手册，找出相应的 ROM 表中对应你的计算机的条目。

经测试表明：下列硬盘驱动器可以正常工作于 SCO UNIX System V / 386 下：

CDC Wren IV  
CDC Wren V  
Conner CP-340  
Conner CP-3100  
Quantum Q250  
Quantum Q280  
Quantum P40S  
Quantum P80S  
Syquest SQ555

下面的硬盘驱动器声称可以工作于 SCO UNIX System V / 386 下但未经我们测试：

Maxtor XT-4380S  
Maxtor LXT-100S

对使用 Adaptec AHA-154 控制器的硬盘来说，头一个磁盘驱动器 (ID 0: LUN 0) 在  $A_0$ ,  $A_1$  和  $A_2$  上不应有跳线，第二个磁盘驱动器 (ID-1: LUN-0) 在  $A_0$  上应有跳线，而在  $A_1$  和  $A_2$  上则不应有。

#### 7.7.5 兼容的硬盘控制器

我们测试结果表明：下列控制器与 SCO UNIX System V / 386 是兼容的：

Adaptec ACB-2320  
Adaptec ACB-2322  
Adaptec ACB-2370 RLL

Adaptec ACB-2372 RLL  
 Adaptec AHA-154x SCSI Host Adopter  
 Adaptec 4525 SCSI ESDI DISK Controller  
 SMS OMTI 8620 (ESDI Controller)  
 SMS OMTI 8627 (ESDI Controller)  
 Western Digital WD 1010 及兼容者  
 Western Digital WD 1003  
 Western Digital WD 1003-WA2  
 Western Digital WD 1005  
 Western Digital WD 1007  
 Western Digital WD 7000  
 DPT PM3011  
 DPT MX3011  
 DTC WD1010 及其兼容者  
 DTC WD2010 及其兼容者

### 7.7.6 视频适配器

此处涉及 SCO UNIX System V / 386 支持的工业标准结构计算机上的视频适配器，  
 下面列出经过测试的适配器清单：

表 7.1 经过测试的适配器表

适配器	分辨率	类型
Compaq Plasma	640 × 400	2-color
Compaq VGC	640 × 350	16-color
Compaq VGC	640 × 480	16-color
Genoa Super VGA	640 × 350	16-color
Genoa Super VGA	640 × 480	16-color
Genoa Super VGA	800 × 600	16-color
Genoa Super VGA	1024 × 768	16-color
Hercules Monochrome	720 × 350	Monochrome
Graphics Card		
IBM EGA	640 × 350	16-color
IBM VGA	640 × 350	16-color
IBM VGA	640 × 480	16-color
Orchid Designer	640 × 350	16-color
Orchid Designer	640 × 480	16-color
Orchid Designer	800 × 600	16-color
Orchid Designer	1024 × 768	16-color
Paradise VGA Plus	640 × 350	16-color
Paradise VGA Plus	640 × 480	16-color

(续表)

适配器	分辨率	类型
Paradise VGA Plus	800 × 600	16-color
STB Extra-EM	640 × 350	16-color
STB Extra / EM	640 × 480	16-color
STB Extra / EM	800 × 600	16-color
STB Extra / EM	1024 × 768	16-color
Video 7 Fastwrite VGA	640 × 350	16-color
Video 7 Fastwrite VGA	640 × 480	16-color
Video 7 Fastwrite VGA	800 × 600	16-color
Video 7 VEGA (EGA)	640 × 380	16-color
Video 7 VRAM VGA	640 × 350	16-color
Video 7 VRAM VGA	640 × 480	16-color
Video 7 VRAM VGA	800 × 600	16-color
Video 7 VGA 1024i	640 × 350	16-color
Video 7 VGA 1024i	640 × 480	16-color
Video 7 VGA 1024i	800 × 600	16-color

注 1: Orchid 图形适配器不能工作于 20Mhz 或更快速总线的扩展方式。另外, Orchid 图形适配器在特定视频方式中执行刷新操作时使用的是中断向量 2。(IRQ2)

注 2: 如果你在用 STB VGA / EM 或者 Genoa Super VGA 图形板, 则在退出图形方式时屏幕上不能回到正常的字符方式, 在退出时出现许多竖杠, 后跟某种字符, 形成杂乱无章的图形。

注 3: Ordid Designer 和 Genoa Super VGA 板使用的地址与 3C503 和 WD8000 3E 网络卡所用地址发生冲突。

下面是声称可以工作于 SCO UNIX System V / 386 下但我们没有测试过的图形适配器清单:

表 7.2 未经测试的图形适配器表

适配器	分辨率	类型
EIZO MD-B07	640 × 350	16-color
EIZO MD-B07	640 × 480	16-color
EIZO MD-B07	800 × 600	16-color
EIZO MD-B10	640 × 350	16-color
EIZO Exrta / EM	640 × 480	16-color
EIZO Exrta / EM	800 × 600	16-color
EIZO Exrta / EM	1024 × 768	16-color
Quadram Quad VGA	640 × 350	16-color
Quadram Quad VGA	640 × 480	16-color

(续表)

适配器	分辨率	类型
Quadram Quad VGA	800 × 600	16-color
Quadram Quad VGA	1024 × 768	16-color
Tecmsar VGA / AD	640 × 350	16-color
Tecmsar VGA / AD	640 × 480	16-color
Tecmsar VGA / AD	800 × 600	16-color
Tecmsar VGA / AD	1024 × 768	16-color
Toshiba Grid 758 Display	640 × 400	Monochrome
Toshiba Grid Plasma Display	640 × 400	Monochrome

### 7.7.7 磁带驱动器控制器组

SCO UNIX System V / 386 下的磁带驱动是通过下面所列的驱动器 / 控制器组来实现的:

表 7.3 驱动器 / 控制器组

厂商	控制器	驱动器
Archive	SC400	Scorpion 5945
Archive	SC402	Viper QIC-02 / 60MB
Archive	SC402	Viper QIC-02 / 150MB
Archive	SC499r	Scorpion 5945
Bell Technologies	PC-36	XTC-60
Cipher	QIC-02	CP-60B
Cipher	QIC-02	CP-125B
Cipher	Cipher 811 / 817	5400 / plus
Computone	SC400	Scorpion 5945 / 60MB
COREtape	PC-36	5000 (E)
Emerald	xnx-50-2012	Cassette
Emerald	xnx-60-2002	Cartridge
Everex	PC-36	60MB Internal
ITT	PC-36	5000 (E)
Mountain	QIC-02	60MB Filesafe
Mountain	QIC-02	150MB Filesafe
Mountain	QIC-02	300MB Filesafe
Mountain	PC-36	60MB Internal
Mountain	PC-36	150MB Internal
Olivetti	PC-36	5000 (E)
Tecmar	PC-36	QIC-60AT

(续表)

厂商	控制器	驱动器
Tecmar	PC-36	QT-60I
Tecmar	PC-36	QT-90I
Tecmar	PC-36	QT-125I
Tecmar	PC-36	QT-150I
Tecmar	PC Bus Host Adapter	QT-60E
Tecmar	PC Bus Host Adapter	QT-90E
Tecmar	PC Bus Host Adapter	QT-122E
Tecmar	PC Bus Host Adapter	QT-150E

实际上大多数符合 QIC-02 标准的控制器都应该是可用的，上表所列只是经过测试的部分控制器。

### 7.7.8 典型设备中断

表 7.4 典型设备中断表

中断	设备
0	时钟
1	控制台
2	网络卡, 磁带及其它
3	串行口 COM2
4	串行口 COM1
5	备用并行端口 (LP2)
6	软盘
7	主并行口 (LP0 或 LP1)

注: 不要使用中断 0, 1, 6。

### 7.7.9 SCSI 标准

SCO UNIX System V / 386 支持 2 个 AHA-154x 或 WD7000 主适配器。下面分别予以说明。

#### 7.7.9.1 Adaptec AHA-154x 说明

并非所有厂家生产的计算机系统都与 Adaptec 主适配器兼容。如果你的主适配器不

能工作或者使用时文件遭到破坏，则应按本节描述的方法进行测试，看是否有不兼容问题存在。

Adaptec AHA-154x 主适配器支持第一类 DMA，异步和同步外设，跳线可选择的 DMA 和中断 I/O 端口地址，允许在 SCSI 总线上执行多线操作的可编程邮箱结构。AHA-154x 的 back-end 软件允许设置 154x 适配器配置参数。可能的配置有：

- on-time 总线
- off-time 总线
- 传输速率
- CCB 数目
- 邮箱数目

Adaptec SCSI 主适配器使用厂商提供的设置可以正常工作。它的配置应为：基地址 0×330，IRQ11，DMA 通道 5。不过在某些计算机上，比如 Tandy 4000，如果主适配器是 AHA-1540 或 AHA 154a，就应去掉跳线 J5，如果主适配器是 AHA-1542，则应去掉跳线 J8。

#### 7.7.9.2 对设备进行格式化和检验

Adaptec 系列 154x 的控制器有一组 BIOS 程序来维护和调试设备。

首先你需要知道 Adaptec 控制器的 BIOS 地址是多少，这一信息可以从 Adaptec 手册中查到。缺省的地址值是 0DC00H。有了这一数值之后你就可以做一系列的工作，其中很重要的一件工作就是对 SCSI 设备进行格式化及校验。在安装操作系统之前所有的 SCSI 设备（硬盘）都应该加以格式化。

要对设备进行格式化应按以下步骤执行：

1. 引导 DOS
2. 运行 debug 程序
3. 在出现提示符“-”之后输入：g=dc00:6（请注意 BIOS 地址的略写形式）。回车之后在屏幕上出现一个菜单。
4. 对指定的驱动器选择 format 任选项。（注意：这样一来磁盘上所有数据都遭到破坏）。
5. 在格式化完成之后，对指定驱动器选择任选项“verify”，进行校验。
6. 退出菜单。

#### 7.7.9.3 硬件不兼容性

下面提供一个对 Adaptec 控制器与主板之间接口进行测试的方法。按照以下步骤执行：

1. 引导 DOS
2. 运行 debug 程序
3. 在提示符“-”出现之后输入：g=dc00:9，回车之后开始测试。

在测试进行时应该在屏幕上看到诊断信息。如果出现失败信息或者屏幕上不显示任何诊断信息，则必然存在硬件不兼容问题。你的系统也许会因此而不能加载及运行 SCO



UNIX System V / 386。由于这一不兼容性是在硬件级别上，所以任何软件包均不能正常运行。

#### 7.7.9.4 Western Digital WD7000 说明

我们没有对 WD7000 适配器进行广泛的测试，但有些硬件配置确实不能工作。在此我们列出据称可以工作和不能工作的系统清单。应该说明的是，在清单中没有的系统完全可以工作。

下面的系统声称可以使用 WD7000 正常工作：

ACER 386 / 20  
ALR Flexcache 25386 (w / IOCR mod on 7000)  
AMI MARK II  
ARC Skyscraper  
AST Premium 386 / 16  
AST Premium 380 / 25 (w / IOCR nod on 7000)  
AT& T 6386 WGS  
Compaq Deskpro 386 / 2S  
CSS Mother Superior  
Everex Step 25  
Intel  
Micronics  
Mitac 25MHZ  
NCR PC-920  
NCR PC-925  
ROSE Hill 387  
Tandy 3000  
Tandon 386 / 20  
Unisys 386 / 25  
Unisys PC-6  
TI 386 / 20  
Wyse 16MHZ  
Wyse 3216-40

下列机器声称很难或者根本不能兼容于 WD7000：

Compaq 386 / 20E  
PC Kraft 386 / 25 with 387 / 20  
Unisys 386 / 20

## 7.8 微通道结构硬件的兼容性

下面说明微通道结构的计算机可以与哪些硬件相兼容。

### 7.8.1 网络卡

SCO UNIX System V / 386 支持以下 Ethernet 板：

- 3Com Etherlink / MC 3c523 卡

• Western Digital WD8003E 卡

这些卡的安装软件允许你安装 3 块同样类型的板。在设备驱动软件安装期间应给每块卡配置不同的参数，比如中断向量和 I/O 基地址。参数选择必须格外小心，以免与系统中的其他硬件发生冲突。在设置每块卡上的跳线（即跨接线）之前必须留神可能发生的冲突。注意：以中断向量 2（IRQ2）中断的网络板应使用编号 9。

应使用中断向量 2（IRQ2）来安装你的网络卡，这是在安装操作系统时为配置网络卡设定的缺省值。厂家为 WD8003E 和 3C523 卡设定的中断向量缺省值是 3（IRQ3）。在安装操作系统之前你必须查阅网络卡的文献，并据此改变板上的跨接线，以使用 IRQ2。如果没有作这一改变，会导致第二串行端口（tty 2a / COM2）与网络卡之间发生冲突。如果你确实需要使用 IRQ3，则只能使第二串行端口处于无效状态。这可以在安装完成之后通过修改 / etc / conf / sdevice.d / sio 文件来实现。具体办法是将第二行第二个域的“Y”改成“N”。

### 7.8.2 串行卡

微通道结构机器上的下列串行卡可在 SCO UNIX System V / 386 下使用：

IBM PS / 2 model 3033 dual async 2 Port Card

Stargate PLUS 8 MC

AST 4 Port or 8 Port Async Cluster Adapter

Digiboard PS-COM / 8 Port or 16 Port

Arnet Multiport / 2 Port or 8 Port-1 or 2 cards

Control Hostess / MC 8 Port

下面给出各串行卡的地址：

表 7.5 各串行卡地址

Physical Port	Board Type	Base Number of Ports	Address
COM1	Motherboard	1	0 × 3F8
COM2	Amet *	16	0 × 140
	IBM	2	0 × 2F8
	AST	4	0 × 2F8
	Stargate * *	8	0 × 400
	Digiboard	16	0 × 3000
	Digiboard	8	0 × DB80
	HostessMC	4	0 × 500
	HostessMC	8	0 × 500
	HostessMC	4	0 × 540
	HostessMC	8	0 × 540
HostessMC	4	0 × 580	
HostessMC	3	0 × 580	

COM2 上的“哑”板，如 Hostess, ASJ 4 Part, IBM Dual Async 占用地址方式如下表:

表 7.6 (OM) 上“哑”板占用地址方式

端口号	地 址
1	0×2F8
2	0×3220
3	0×3228
4	0×4220
5	0×4228
6	0×5220
7	0×5228
8	0×2F0

### 7.8.3 视频适配器与监视器

微通道结构计算机的母板上有 VGA 适配器，所以别无选择。凡是操作系统支持的计算机上的缺省适配器（单色或彩色监视器）都可兼容于 SCO UNIX System V / 386。

### 7.8.4 视频卡和监视器

SCO UNIX System V / 386 支持下面的视频卡:

IBM Personal System / 2 Integral VGA adapter

及 8503, 8512, 8513 和 8514 监视器

Olivetti P800, Integral VGA

Tandy 5000MC, Integral VGA

Apricot Qi, Integral VGA

### 7.8.5 兼容硬盘控制器

微通道结构机器上的下列控制器可用在 SCO UNIX System V / 386 上:

Adaptec AHA-1640 (SCSI 主适配器)

Adaptec 2610 (ESDI 磁盘控制器)

Adaptec 2620 (ST506 磁盘控制器)

IBM (ESDI 磁盘控制器)

IBM (ST506 磁盘控制器)

Western Digital 1006V-MCI (ESDI 磁盘控制器)

Western Digital 1007V-MCI (ST506 磁盘控制器)

### 7.8.6 磁带驱动器 / 控制器组

下面列出 SCO UNIX System V / 386 支持的磁带控制器及卡式磁带驱动器组的清单:

表 7.7 磁带驱动器 / 控制器组

厂 家	控制器	驱动器
Everex / Archive	QIC-02	2150L
Archive	SCSI	2150S
IBM	QIC-02	6157
IBM	QIC-02	6157-002
Irwin	floppy	245
Irwin	floppy	285
Mountain 7060	OIC-02	60MB Filesafe
Mountain 7120	OIC-02	150MB Filesafe
Tecmar / Wangtek	QIC-02	QT-60E
Tecmar / Wangtek	QIC-02	QT-150E

## 附录 A 使用系统控制台和彩色显示器

本附录主要讨论影响系统控制台和其它彩色显示器的实用程序和特性。控制台显示器与标准的彩色适配器相连接,而连接于系统上的彩色终端或显示器与特定的适配器相连接。

本附录将介绍如何进行下述工作:

- 使用 `kbmode (ADM)` 设置或改变控制台键盘类型。
- 在不用时清除控制台屏幕以防止控制台的过度磨损。
- 使用 `multiscreen (M)` 控制一个单独显示器的多屏幕。
- 使用 `vidi (C)` 改变用于屏幕显示的字形。
- 使用 `setcolor (C)` 设置彩色显示器上显示的颜色。

### A.1 控制台键盘类型的选择

操作系统支持两种键盘类型: AT 和 XT。系统的缺省配置是使用 XT 键盘。AT 类型支持只有 AT101 或 102 键的键盘所具有的扩展键的设置。AT 键盘可以正常地工作于 XT 模式,但这时不能使用扩展键。不能使 XT 或其它非 AT 键盘工作于 AT 模式。如果这样做了,系统就不会识别键盘的输入。`kbmode (ADM)` 就是用来测试和设置键盘类型的。

有一些键盘有 AT 键盘的布局,但又不能支持 AT 模式。要想对你的键盘进行测试判断它是否支持 AT 模式,可以如下方式调用带测试选项的 `kbmode`:

```
kbmode tset
```

下面是用 kbmode 进行测试的一个例子:

```
#kbmode tset
```

```
Current keyboard mode is XT
```

```
Do you want to determine if your keyboard supports AT mode? y
```

```
During the test the keyboard will be put into AT mode.
```

```
You should then press the space bar two or three times.
```

```
Are you ready to start? Y
```

```
Please hit the space bar now!
```

```
The key board has been returned to its default mode.
```

```
It Supports AT mode.
```

```
#
```

这时键盘就被临时初始化为 AT 模式。

kbmode 也可被用于设置模式。使用下面两个命令可以分别把键盘设置为 AT 和 PC / XT 模式:

```
kbmode at
```

```
kbmode xt
```

为了永久性地修改缺省模式, 核心参数 KBTYP E 就必须被设置为正确的键盘模式。要改变 KBTYP E, 可以运行 Configure (ADM) 并选择选项 13“Hardware Dependent Parameters”。然后按“系统性能的调整”一章中“使用 configure 重新分配核心资源”的方法修改参数值并重新连接核心。

## A.2 使用控制台屏幕的保护特性

可以把控制台设置为在若干秒之后就对控制台屏幕进行清除的方式, 以防止屏幕的过度磨损。(这与大多数终端所具有的特性类似。) 核心参数 TBLNK 控制着这个控制台屏幕的自我保护特性。其缺省设置是不对屏幕进行清除。为了使用这个特性, 就必须调用 sysadmsh 并选择 System → Configure → Kernel → Parameters 并选第 6 项“Multi Screens”, 然后把 TBLNK 的值改为在对屏幕进行清除之前系统应该等待的时间(以秒为单位)。之后必须重新连接核心并重新引导以使这一设置生效。使用 sysadmsh 的 System → Configure → kernel → Rebuild 选项来重新连接核心。

## A.3 使用多屏幕

利用多屏幕, 就能够像同时使用多个终端一样使用控制台。只要按一下某个组合键就可以从一个屏幕切换到另外一个屏幕, 每个屏幕都可以像一个独立的终端一样进行工作。

因为每个多屏幕都是独立的, 所以可以在各屏幕上注册并运行程序。因为程序的输出都被保存在一个屏幕缓冲区中, 所以不管看哪个屏幕, 都可以看到各自最近的输出结果。如果停止了某个屏幕的输出, 例如当按了 <CTL>s 这个组合键, 那么只会影响那个屏幕。

计算机中的内存量决定了系统所能提供的多屏幕数目。当引导系统时, 自动建立的多屏幕数就会显示出来。大多数机器可建立 2 至 6 个多屏幕, 但如果机器有足够的内存, 那

么系统就会最多达到 12 个多屏幕。要想增加系统的多屏幕数，就必须增加系统的内存；这样就可以自动建立更多的屏幕。

虽然可以同时打开并激活所有的多屏幕，但一次只能看到一个屏幕。这个被选中的多屏幕就像连接于这个键盘上的一个终端。在屏幕之间进行切换就像从一个终端移到另一个终端上，因为每个多屏幕都有它自己的设备文件。

多屏幕使用 `/dev/tty [01...12]` 设备文件。这些文件提供了系统、计算机屏幕和键盘之间的字符 I/O。

为了选取一个活动的屏幕，可按 `<Alt>-Fn`，其中 `Fn` 是你键盘上的一个功能键。功能键一般都位于键盘的上部或最左部。`tty01` 是 `<Alt><F1>` 终端，`tty02` 是 `<Alt><F2>` 终端，`tty03` 是 `<Alt><F3>`，等等。例如，`<Alt><F6>` 把管理员切换到屏幕 6 中，它对应于 `/dev/tty06`。

可以按组合键 `<CTL><PrtSc>` 循环进入各个屏幕。如果在键盘上没有功能键，就可以使用这个组合键访问各屏幕。例如，如果建立了 12 个多屏幕，但计算机键盘只有 10 个功能键，那么要显示屏幕 11，可以按 `<Alt><F10>` 得到屏幕 10，然后再按 `<CTL><PrtSc>` 进入屏幕 11。要访问屏幕 12，就再按一次 `<CTL><PrtSc>`。再按 `<CTL><PrtSc>` 就回到了第一个屏幕 `tty01`。

请注意，除了 `<Alt>Fn` 组合键可以改变屏幕之外，也可以使用 `<CTL><Alt>Fn`。在把 `<Alt><Fn` 保留给自己使用时，这种 `<CTL><Alt>Fn` 组合就显得特别有用。这可以使用 `mapkey (ADM)` 来进行配置。

视频适配器可以动态地分配给 `multiscreen`。所有的屏幕都开始于主适配器，但任何屏幕都可以使用 `ridi (C)` 命令移到其它视频卡上。

有效的适配器名为：“`mono`”、“`cga`”、“`ega`”和“`vga`”。

例如，如果有一个 EGA 适配器作为主适配器，一个 MONO 是第 2 适配器，那么任何一个想放到 MONO 卡的屏幕都可以用如下命令来移动：

```
vidi mono
```

#### A.4 改变视频字形

可以使用 `vidi (C)` 在一个显示适配器上显示所有字符。一般情况下，如果控制台有一个显示适配器，它的字符集被定义在一片 ROM 中，那么只能显示定义在 ROM 中的那些字符。另外，这个控制台的 `mapchan` 文件一定要对应于定义在显示适配器 ROM 中的字符集，这样才能显示整个字形集。

使用 `vidi` 除了能超越 ROM 之外，还可以使用它在一些显示适配器上定义某些显示字形。例如，VGA 适配器能允许你显示  $8 \times 8$ ， $8 \times 14$  和  $8 \times 16$  的字形。

`vidi (C)` 能定义如下六种字符集字形。下面的字形定义文件都在目录 `/usr/lib/vidi`：

Character Set	8 × 8font	8 × 14font	8 × 16font
PC standard	font8 × 8	font8 × 14	font8 × 16
ISO 8859 / 1	iso.8 × 8	iso.8 × 14	iso.8 × 16
PC Nordic	nor.8 × 8	nor.8 × 14	nor.8 × 16
PC Portuguese	por.8 × 8	por.8 × 14	por.8 × 16
PC Spanish	spa.8 × 8	spa.8 × 14	spa.8 × 16
PC Greek	grk.8 × 8	grk.8 × 14	grk.8 × 16

### A.5 用 setcolor 控制彩色显示器

setcolor 能控制用于显示屏幕上的颜色 (setcolor 命令一般对于单色显示器或终端是没有作用的)。前景和背景都可被独立地设置为 16 种颜色。setcolor 也可以设置反相的视频和图形字符颜色。

有如下几种颜色：兰色、洋红、棕色、黑色、淡兰色、淡洋红、黄色、灰色、青色、白色、绿色、红色、淡青色、强白色、淡绿色、淡红色。

要想显示这些颜色，只要调用不带任何选项的 setcolor 就可以。

在下面的变化中，“颜色”都从上面所列的颜色中取。

#### 1. 改变前景和背景的颜色

可以就像下面的例子那样用一个命令既设置前景又设置背景：

```
setcolor red white
```

其结果是在白色的背景上显示红色字符。如果只指定一种颜色，就只改变前景。要想只改变背景颜色，可像下面这样使用选项：

```
setcolor -b red
```

这只会把背景颜色改为红色。

#### 2. 改变反相显示的颜色

反相显示一般是颠倒前景和背景颜色。

setcolor 可以使管理员独立地对它们进行设置。例如：

```
Setcolor -r blue red
```

这条命令把反相显示的前景设置为兰色，把背景设置为红色。

#### 3. 改变屏幕边界的颜色

还可以改变定义文本显示区域的正方形边界的颜色：

```
setcolor -o green
```

上面的例子把边界改为绿色，但不影响其它部分的显示。

#### 4. 设置键盘响铃

setcolor 的一个不太引人注意的功能是控制响铃的声音，响铃一般配合显示器和键盘的使用。若想改变响铃的音调，必须给出音高和持续的时间。（音高用以微秒计算的时间表示，持续时间用以五十分之一秒为单位度量。）例如：

```
setcolor -p 500 2
```

这个命令使响铃为高音调而持续时间短。音高的数越大，所产生的音调就越低。例如，这个命令把响铃的持续时间加长，但音调变低：

```
setcolor -p 7000 8
```

请注意每当按一下 <CTL> g，响铃就按最近所设置的形式响一下。

#### 5. 使屏幕复位

-n 选项使屏幕恢复为原来的白字符黑背景的显示方式。



# 第四部分 网络通信指南

## 第一章 使用调制解调器

调制解调器对系统是一个重要补充，它允许你的系统通过电话线与远程系统通信。

本章讲述如下几个部分：

- 为计算机连接制解调器
- 维护调制解调器连接
- 为拨号输入线路定义口令

应该注意，设备与系统之间的物理连接随硬件配置的不同而不同。要连接你的串行设备，应查阅厂家随设备的计算机一道提供的手册以获取有关该设备的信息。在增加调制解调器之前，应保证有一个可用端口，或者直接来自端口 COM1 或 COM2，或者取自某一多端口卡。

### 1.1 选择串行端口

系统支持串行端口的调制解调器控制。表 1.1 给出了带调制解调器控制和不带调制解调器控制的串行端口的标准设备名。

表 1.1 串行端口

设备	功 能
/dev/ttyla	不带调制解调器控制的主串行适配器
/dev/tty1A	带调制解调器控制的主串行适配器
/dev/tty2a	不带调制解调器控制的备用串行适配器
/dev/tty2A	带调制解调器控制的备用串行适配器

/dev/ttyla/和/dev/tty1A 指的是同一串行端口（同样，/dev/tty2a 与/dev/tty2A 也一样），操作系统对这两者使用不同的设备驱动子程序。注意不要在同一时刻使用带调制解调器控制和不带调制解调器控制的同一端口，比如/dev/ttyla 与/dev/tty1A，否则你就会看到警告信息：

```
Cannot open: device busy
```

对包含多端口串行卡的系统来讲，设备/dev/tty[1, 2][a, m]指的是不带调制解调器控制的端口，而设备/dev/tty[1, 2][A, M]指的则是带调制解调器控制的端口。

### 1.2 配置调制解调器

在使用 CU(C)和 UUCP(C)时正确的调制解调器配置是必不可少的条件。每种调制解调器的配置都是不一样的。应查阅你的调制解调器手册以获取正确的开关设置状态。

### 1.2.1 Smartmodem 1200

如果你的调制解调器是 Hayes Smartmodem 1200 或其兼容者，开关 3 和 8 应处于向下的位置：

	1	2	3	4	5	6	7	8
上	·	·		·	·	·	·	
下			·					·

开关 3 向下时，产生的代码被送至终端或计算机的调制解调器；当开关 8 向下时，调制解调器能够解释当前发出的命令。这就使得 MS-DOS 和 UNIX 通信系统能够正常工作。

### 1.2.2 Smartmodem 2400

Hayes Smartmodem 或兼容的调制解调器如果作为拨号输入线路使用需要联机配置。要注意对于 Hayes 2400，如果没有用 2400 波特命令设置，就不能应答用 2400 波特传送的电话。必须通过 cu (C) 发出命令来配置调制解调器。cu 命令的格式为：

```
cu-S2400-l ttynn dir
```

其中 nn 是串行线路的“tty”号。为了在 tty1A 上配置调制解调器，需打入命令：

```
cu-S2400-l tty1A dir
```

回车之后接着输入下面的命令以配置调制解调器，它们将保存于调制解调器的非易失性存储器中。如果不想保存这些设置，则不要输入最后一个命令 (AT&W)。每个命令之后都应跟一个回车：

AT&f: 取出厂家做好的配置

ATT: 音调拨号

AT10: 低扬声器值

AT&d2: 设置 dtr“2”：当 dtr 失落时继续挂起

AT&C1: 设置 dcd“1”：dcd 跟踪远程载波

ATs0: 在 1 响后应答电话

ATe0: 无响应 (调制解调器将不回应它所送出的内容)。

ATq1: 安静方式 (调制解调器在收到此命令及随后的信号之后将不再响应“OK”信号)

AT&W: 将设置保存于非易失性存储器中。

要从 cu (C) 退出应输入一个波折线“~”和一个句点，后跟一个回车：

~.

现在调制解调器已经配置好可供使用。

## 1.3 安装拨号输出调制解调器

当你安装调制解调器或其他什么设备时，应保证连到计算机上的串行通信线不是悬挂

着的，因为这会极大地影响系统的性能。

经常用来连接终端和计算机的三线电缆如果用来连接调制解调器就显得不够用了。对一个 25 针串行端口上的调制解调器来讲，第 2，3，7，8，及 20 针必须直接相连。如果你不能确信哪些针是必须相连的，那么用一条电缆把所有针都连起来一定没有问题。带状电缆，或者被称为“straight-through（直通）”的电缆，即将各针直接交叉相连的电缆，都是可行的。

为了安装调制解调器，应按以下步骤操作：

1. 确认 UUCP 软件包已经安装完毕。否则应用 Custom (ADM) 实用程序加以安装。

2. 确认你所选来用在你的拨号输出系统的调制解调器上的串行端口在引导时已被系统识别，如果调制解调器用于内部，则应保证配置内部调制解调器的 COM 端口不与其他任何设备发生冲突。只支持附在 COM1 和 COM2 端口上的串行设备。

3. 保证端口已通过输 disable (C) 命令置为无效：

```
disable ttyname
```

4. 用一条“straight-through”电缆连接调制解调器与计算机（第 2 针与第 3 针不交叉）。电缆必须至少将第 2，7，8 针以及 20 针相连。

大多数标准 COM 端口使用“straight-through”电缆，但有些硬件要求使用 null-modem 电缆（第 2 针与第 3 针交叉）。标准 COM 端口又称为 DTE (Data Terminal Equipment, 数据终端设备)，需要使用 null-modem 电缆的端口称为 DCE (Data Communication Equipment, 数据通信设备)。如果你对自己的 COM 板是 DTE 还是 DCE 不是很有把握，可去查阅硬件文献。如果是 DCE，则需要使用 null-modem 电缆。

5. 在 /usr/lib/uucp/Devices 文件中增加适当条目。对用于调制解调器的每一串行端口，在 /usr/lib/uucp/devices 文件中都应该有两个条目。一个在你用调制解调器启动一次寻呼 (ACU 线路) 时使用。另一个则在用标准 Hayes 命令集来配置调制解调器时使用。下面的使用端口 COM1，为波特率 2400 的 Hayes 兼容的调制解调器建立的条目：

```
Direct tty1a-1200-2400 direct
```

```
ACU tty1A-1200-2400 /usr/lib/uucp/dialHA24
```

应注意在这样的条目之前不能有“#”号，因为若有“#”号在行头，则该行被认定是注释行而加以忽略。在 Devices 文件中有许多行因为有“#”号在行头而被忽略掉。

6. 输入以下命令，使 UUCP 成为你所选择端口的属主：

```
Chown uucp /dev/ttyname
```

现在可以测试这一拨号输出的调制解调器。

1. 输入以下命令，看调制解调器是否拨号成功，以此来测试该调制解调器的作用：

```
cu -ltty1a dir
```

你应该可以看到一条信息，指示你已经连接成功。如果你看到这么一条信息：

```
cu: dir permission denied
```

表示执行 cu 命令的用户不具有写 /usr/lib/uucp/Devices 文件的权限。

如果你没有看到这么一条信息，且无信息指示你已连接完毕，则必是 cu 命令不正

确，或者 Devices 文件不正确，或是串行端口工作不正常。

注：下面的命令假定使用 Hayes 兼容的调制解调器命令集和响应代码。其他的调制解调器也许有所不同。

2. 如果有信息指示连接成功，则应在键盘上输入命令：AT

回车之后屏幕应回打“OK”。如果已设定调制解调器以数字代码而不是文本形式返回结果代码，则你应看到“0”。

3. 如果不是这样，则输入一个键，观察调制解调器上的“接收”灯是否闪亮，该灯是用来指示调制解调器是否从键盘接收到信号的。如果灯没有闪亮，就该检查电缆和调制解调器的开关设置。如果“接收”灯闪亮了，而你还是没能从调制解调器获得响应“OK”，应输入以下命令，使调制解调器的回打 (echo) 功能有效：

ATE1

4. 如果你的终端还是不能显示“OK”或者“0”，应将调制解调器连接到另外一个端口 (COM1 或 COM2)，如果这时调制解调器工作正常，就应检查设备使用的中断向量是否正确。(串行端口 COM1 使用 4 号中断，COM2 使用 3 号中断)

5. 如果连接到另外一个端口之后终端依然不能回打“OK”信息，则该调制解调器是有问题的。

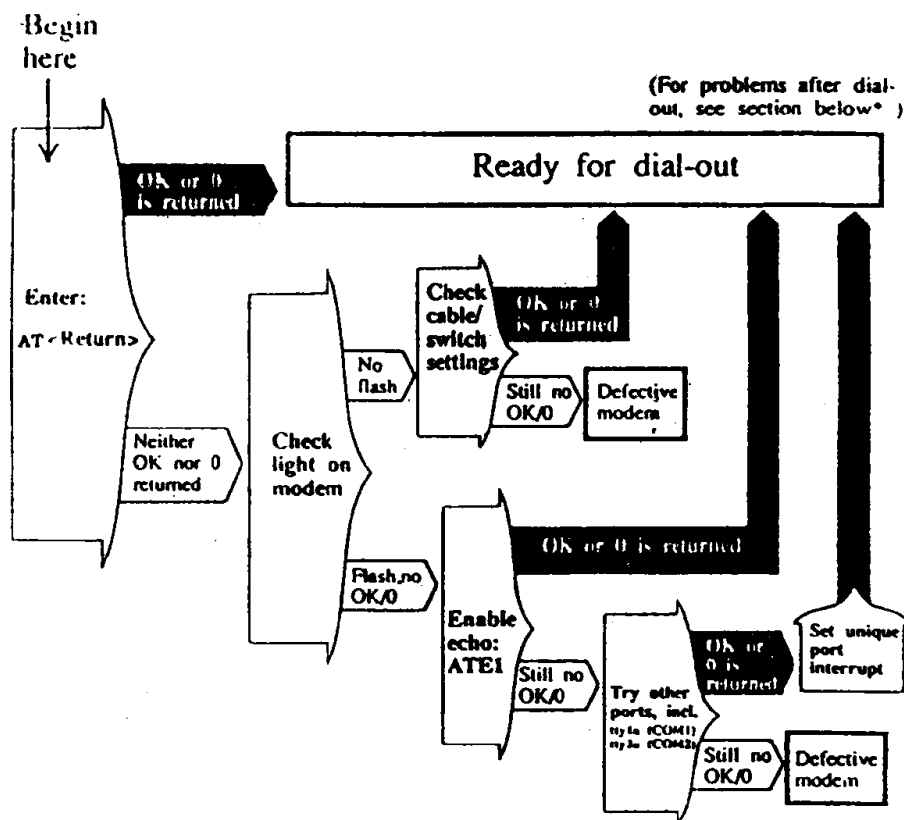


图 1.1 modem 测试流程图

在调制解调器配置正确而且终端显示了正确的回打信息 (OK 或 0) 之后，输入。

ATDT phonenumber

即可拨号输出。当你确信调制解调器可以拨号输出，可输入“~”并回车，退出 cu。现在你可以拨号进入另一个系统。使用以下命令来拨号输出：

```
cu -lty1A 555-1212
```

将以上命令中的“555-1212”换成你想要拨号进入的系统的电话号码。

---

注：在 cu 命令中加上一个电话号码来与某一调制解调器相连时，不应该使用逗号(,)，这在 Hayes 命令集中表示暂停。使代之以连字号 (-)。这一规定同样适用于 Systems 文件条目。

---

如果遇到问题，应查阅本书中“系统中可能出现的问题”一章中相应部分的内容。如果这条线路还用于拨号输入，则应按照下一节“建立拨号输入调制解调器”部分指定的步骤执行。

#### 1.4 建立拨号输入调制解调器

下面给出安装用于拨号输入操作的调制解调器的详细步骤。(对拨号输入线路建议使用口令，参考本章后面“为拨号输入线路增加口令”一节的内容)。

1. 按照安装拨号输出调制解调器的步骤执行。使硬件连接能够正常工作。
2. 有的调制解调器有用于设置调制解调器配置的开关或软件命令，若你的调制解调器有这样的设置，应按调制解调器手册中的要求对调制解调器进行配置。

---

注：如果你的调制解调器由拨号输入和拨号输出共享，则第 3 步可以省去。无论什么时候系统启动或者拨号输出完成，拨号输入的初始化工作都会自动执行。

---

3. 当有拨号输入时将调制解调器设置为自动响应电话方式。

大多数内部调制解调器没有自动响应方式，有的外部调制解调器也没有。如果是这样，就在初始化文件 /etc/rcd/8/userdef 中放置如下的一行内容：

```
(stty 1200; echo“ATS0=1\r”>/dev/tty1a) </dev/tty1a
```

tty1a 应该与调制解调器连接的不带调制解调器控制的设备相匹配。1200 应该是调制解调器所使用的最高波特率。ATS0=1 是将 Hayes 兼容的调制解调器设置为自动响应方式的命令。 \ r 给调制解调器发送一个回车信号以终止该命令行。

4. 设置你的调制解调器，使得在 DTR (Data Terminal Ready 数据终端准备好) 线路无效时不作响应，而在 DTR 从有效变为无效时脱开当前连接。

5. CD (Carrier Detect) 信号应设置为随外来载波的有无而变化。当有载波时，该信号为低电平，当没有出现载波时，该信号为高电平。

6. 设置你的调制解调器，使之不响应命令或显示响应。

7. 输入以下命令，保证端口已被置为无效：

```
disable tty name
```

这里 ttyname 是非调制解调器控制端口。

8. 在 /etc/inittab 文件中选择所需的 gettydefs 条目。输入“2”选择 1200-2400-300

周期。

9. 使用下面的 enable (C) 命令, 使你的调制解调器所有的端口有效:

```
enable ttyname
```

这里 ttyname 是调制解调器控制端口。

10. 从另一调制解调器对本调制解调器拨号。

11. 如果你不能成功地拨号进入, 请参考本书中“系统中可能出现的问题”一章中相关章节的内容。

### 1.5 安装拨号输入 / 输出共享的调制解调器

UNIX 系统支持在同一调制解调器线路上使用拨号输入和拨号输出功能, 而不必使 login 注册号无效。在拨号输出程序使用线路时, 就使本注册号暂时无效。如果有人正占用线路拨号输入, 而这时你的系统又企图使用这条线路拨号输出, 则拨号输出程序将不能打开此设备, 因为设备已被锁住。为了实现这一机制, 必须使用调制解调器控制设备, 调制解调器必须根据载波的出现与否将载波检测 CD 设置为高或者低电平。

按以下步骤执行即可安装共享拨号输入或拨号输出的调制解调器:

1. 按照安装拨号输出调制解调器以及拨号输入调制解调器的步骤执行。
2. 要拨号输出, 只要按正确的任选参数启动 cu 即可。线路上的 getty 会自动挂起, 等待拨号输出。输出完成之后应重新启动。

### 1.6 拨号输入你的计算机

要允许拨号输入你的计算机, 必须用命令 enable (c) 使一条串行线路有效, 并能识别调制解调器控制信号。

若使用主串行适配器 (COM1), 应输入:

```
disable tty1a
```

```
enable tty1A
```

若要使用备用串行适配器 (COM2), 输入:

```
disable tty2a
```

```
enable tty2A
```

---

注意: tty1A 与 tty1a 指的是同一 (主) 串行线路, tty2a 和 tty2A 指的是同一条 (备用) 串行线路。不要同时在调制解调状态和非调制解调状态下使用同一线路, 否则会产生错误。

---

#### 1.6.1 为拨号输入线路增加口令

如果需要, 你可以选择 tty 路线, 为其定义特殊的拨号输入口令, 要求经过授权的用户输入拨号输入口令。登记信息, 包括上一次连接的时间, 可以存放起来以备后用。

需要用口令的特殊的拨号输入线路在文件 /etc/dialups 中定义, 格式是每条线路一个 tty 设备, 例如:

```
/dev/tty1A
```

/dev/tty5C

实际的拨号口令保存于文件 /etc/d-passwd 中，其格式与 /etc/passwd 中口令完全一样。/etc/d-passwd 中的第一个域（用户名）中并不是用户名字，而是 shell 程序名（比如 /bin/sh）。如果企图在 /etc/dialups 中所列线路上注册的用户的注册 shell 程序名在 /etc/d-Passwd 中，则用户被提示输入存放于 /etc/d-passwd 中的拨号输入口令。

下面是建立拨号输入口令的命令格式：

```
passwd-m dialname
```

只有超级用户才有权力对拨号 shell 口令进行修改。

## 1.7 从你的计算机拨号输出

在 UNIX 系统下可以用 cu (C) 和 UUCP (C) 实用程序来寻呼 (call) 远程系统并进行数据传输。文件 /usr/lib/uucp/devices (称为 Devices 文件) 包含了用以确定一条特定串行线路特性的程序信息。

Devices 文件包含了一些行，它们定义了线路的设备、该条线路相关的呼叫部件以及 UUCP 所用的波特率。(调制解调器控制设备应用于与调制解调器相连的线路)

### 1.7.1 使用拨号程序

就拨号而言，cu 和 UUCP 使用同一拨号程序集，这可能是独立的二进制程序，如 /usr/lib/uucp/dialHA12，也可能是文件 /usr/lib/uucp/Dialers 中的条目。

拨号程序的源文件和用于重新编译源程序的 make 文件都包含在目录 /usr/lib/uucp 中。如果要用其它种类的调制解调器，可以修改源文件并建立自己的拨号程序。注意，必须安装了开发系统软件之后才能编译新的拨号程序。

要产生新的拨号程序，应按以下步骤执行：

1. 用下面命令将目录改换到 /usr/lib/uucp

```
cd /usr/lib/uucp
```

2. 在 /usr/lib/uucp 目录中编辑文件 makefile，找出下面的行：

```
EXES = dialHA12 dialHA24 dialMUL dialTBIT dialVA3450
```

增加你希望使用的拨号程序名。完成之后，退出文件，并将所作的改变存下来。

3. 然后在 shell 提示符下输入命令：

```
make
```

然后按回车键。

在 make 命令执行完之后，你就已经有了新的拨程序。这可以在 Devices 文件的某一条目的第五个域中使用。

## 第二章 建立 Micnet 局部网络

Micnet 网络允许在两个或多个独立的 UNIX 系统之间进行通信。该网络由通过串行通信线（即用电缆连接 RS-232 接口）连接的计算机组成。网络中的每台计算机仍作为独立系统运行，但允许用户使用 mail、rccp 和 remote 命令与网络中其他计算机进行通信。这些命令将信件、文件及命令等信息从一台计算机传送到另一台。

建立和维护 Micnet 网络是系统管理员的任务。系统管理员决定计算机之间如何连接，进行具体的硬件连接，然后使用 netutil 程序来定义和启动网络。

### 2.1 网络设计

为了建立一个 Micnet 网络，netutil 程序要求你提供网络中计算机的名字，计算机之间如何连接的说明以及使用的串行线路的表格等。

为了使这件工作尽可能简单，你应该花一些时间来设计网络，并且建立系统要求你提供的信息表格。下面几节将提出设计网络的方法。

#### 2.1.1 选择机器名

Micnet 网络要求网络中每台计算机都有唯一的“机器名”，以便将网络中的计算机区分开来。最好把选择机器名作为网络设计的第一步。这样能防止当你用 netutil 程序建立网络之后发生混乱。

机器名可以包含计算机所在的地名信息或者用户人名信息。当然，也可以用你想用的任何名字。名字必须是唯一的并且由字母数字组成。Micnet 程序仅用每个名字的前 8 个字符，所以必须保证这前 8 个字符是唯一的。

netutil 程序将计算机的机器名保存在文件 /etc/systemid 中。每台计算机建立一个文件。在建立并安装完网络之后，你可以在计算机上用显示该文件内容的方法查出机器名。

#### 2.1.2 选择网络拓扑结构

网络拓扑结构是网络中计算机如何连接的说明。在任何的 Micnet 网络中，都采用可以构成所有拓扑结构的两种通用拓扑结构，就是“星型结构”和“线性结构”。

在星型结构中，所有计算机都被连接到一台中央计算机上。所有的通信都经过中央计算机再到达所要求的目的地。

在线性结构中，计算机组成一个链，每台计算机都不直接与两台以上数目的计算机相连，所有的通信都沿着这条链到达所要求的目的地。

网络可以是严格的星型，严格的线性结构，也可以是星型和线性结构相结合的拓扑结构。唯一的限制条件是，不允许网络形成环。例如，不能将两台计算机首尾相连而使得线性结构的网络闭合。

你所选择的拓扑结构类型取决于需要连接的计算机的数目、希望进行通信处理的速度



以及希望如何分配网络上的通信任务。星型结构可以提供两台计算机的快速通信，但是它不仅要占用中央计算机大量的操作时间而且要求在中央计算机上有大量的串行通信线。线性拓扑结构平均分配通信负担，每条计算机上只要求有两条串行通信线，但是如果线性链比较长，那么通信速度就很慢了（两台计算机之间的通信可能要花费几分钟）。通常情况是，由星型结构和线性结构相结合的拓扑结构建立的网络最为理想。在任何情况下都应选择你认为最好的结构。如果你发现作出的选择不是最佳的，可以随时加以改正。

### 2.1.3 画网络拓扑图

网络拓扑图是网络中计算机间的相互连接关系的草图。你可以用这张图来确定建立网络所需要的串行通信线的编号和位置。

你可以在选定网络拓扑结构的同时画出这图。把网络中每台计算机的机器名直接排列在纸上，然后用线连接你希望用串行通信线相连的每对计算机。例如，三台计算机以线性结构相连的拓扑图可能形如：

a-----b-----c

在画图时，应保证网络中任意两台计算机之间没有多于一条的连线，并且要保证没有环形出现（环是指一系列形成闭合回路的连接）。多重连接和环形是不允许的。

### 2.1.4 网络连接策略

一旦画完拓扑图，你就可以确定用哪些串行端口来连接计算机。因为网络中每两台计算机之间连接要用到两个串行端口（每台计算机上要一个）和一条串行通信线。在分配时必须仔细以免出错。

为网络中每台计算机建立一个可用的串行端口（称为 TTY 端口）表。你可以检查文件 `/etc/inittab` 来显示计算机上的串行端口表，如果串行端口没有被用来连接象终端或调制解调器等设备，则该端口就是可用的。

例如，在我们上面所画的拓扑图中，计算机 b 有两个网络连接，一个与计算机 a 相连，一个与计算机 c 相连。所以应该给计算机 b 分配两个 TTY 端口，给计算机 a 和 c 各分配一个 TTY 端口。

对 Micnet 使用的 tty 名字有一个限制。网络中的 tty 端口号不允许重复。大多数计算机对 tty 端口进行编号的方法是类似的（tty1a, tty2a, tty3a, 以此类推），这意味着在网络的不同计算机上，串行端口的名字是相同的。比如，如果你把计算机 a 上的 tty1b 和计算机 b 上的 tty5d 相连，就不能将计算机 c 的 tty1b 端口与网络上的任何计算机相连。这一限制条件的原因是，Micnet 程序不能识别网络中的不同计算机，于是认定不同机器的同名端口为同一端口，所以在通信时必须清楚标明端口名。这样，如果程序发现在拓扑结构中有两个同名的 tty 端口，网络就不能正常工作，我们推荐一种方法，可以很容易地处理 Micnet 连接问题。

我们建议选择一组系统不用的数作为虚拟的 tty 端口连接到 Micnet 网上。建议使用从 40 或 50 开始的数作为 tty 编号，虚拟号没有上限，可以根据拓扑文件中需要的量选用。例如，因为没有两台计算机在拓扑结构中使用相同的 tty 名字，在拓扑文件中可以表示为计算机 a 用 tty50 与计算机 b 的 tty51 相连，因为不管是 tty50 还是 tty51 在系统中都

不存在（虚拟），你可以确信在网络中 tty 名字并不重复。

要使这些新的虚拟 tty 名字与计算机上实际的 tty 名字相对应，应用 ln (c) 命令将你在拓扑文件中给出的文件名与系统中真正的 tty 端口对应起来。在使用 ln (c) 将两个文件名相连时，你就告诉操作系统真正的 tty 端口也就是用现在赋予它的名字来表示的。当你列举 /dev 目录的内容时，看到两种名字（一种虚拟文件名，一种为实际端口名），两种名字都访问同一物理设备。

举一个 ln (c) 使用的例子。如果用线将计算机 a 的 tty1a 与计算机 b 的 tty5d 相连，你在计算机 a 上发出如下命令：

```
ln /dev/tty1a /dev/tty50
```

接着，当 Micnet 向 tty50 发送信息时，信息送到了 tty1a，类似地，把计算机 b 的 tty5d 与虚拟 tty51 联系起来：

```
ln /dev/tty5d /dev/tty51
```

这允许你可以将任意可用的端口用串行通信线相连而不用担心重复连接。例如，计算机 c 上唯一可用的端口是 /dev/tty1a，你就不必改变系统配置就可以直接在网络拓扑中使用该名字。

在某个端口因为某种原因不能工作时，这种策略可以给你带来方便。你只要将虚拟 tty 号（tty50, 51, 52, 之类）改连到其他正常的物理端口上，而不用重新建立新的网络拓扑结构，网络即可正常工作。

### 2.1.5 确定端口和速度

建立和分配 tty 和端口的步骤如下：

1. 用拓扑图和上面讲到的策略，为每台计算机的每一连接确定一个（仅一个）可用的串行线路。列出你计划使用的实际和虚拟串行端口。比如，如果计算机 a 只有一个可用的串行端口（tty1a），而你打算使用的虚拟名为 tty50，则在拓扑图中的入口应该是下面这个样子：

```
a-----b-----c
(tty1b)
(tty50)
```

2. 对拓扑图上的所有计算机重复第 1 步。确信每一个连接都分配到一个串行端口和一个虚拟端口并且没有任何连接使用同一个虚拟端口。完成之后，拓扑图的结构类似：

```
a-----b-----c
tty1b      tty2a      tty3a      tty1b
(tty50)    (tty51)    (tty52)    (tty53)
```

如果某一计算机没有足够的可用串行端口满足需要，你可以移去已经连接于其上的某些设备以空出串行端口。如果无法移去设备，你就只能够重新画拓扑图。

3. 使用拓扑图，确定每对计算机串行线的传输速度。这个速度必须在串行通信线所允许的传输速率范围内（从 110 至 9600）。传输速度取决于你的偏爱。一般说来，速度越高则完成一次传输所需时间越短，而对系统输入 / 输出能力的要求就越高。有的情况下，传输速率由硬件能力而定。一些硬件不能以高于 1200 波特率的速度工作。为此，当第一

次建立 Micnet 网络时，建议使用 1200 的波特率。如果发现硬件能够支持，还可以慢慢提高波特率。

4. 在拓扑图完全填满之后，列出所有的计算机对，标出它们的机器名，串行端口以及传输速率。在安装网络时，就可以使用这张表。下面是一张拓扑图，传输速率为缺省值。

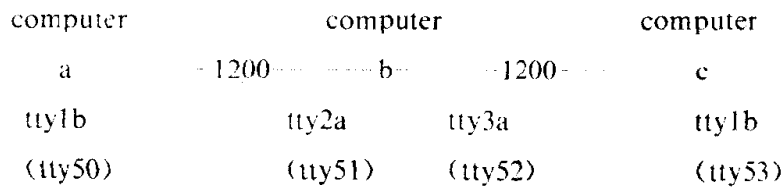


图 2.1 网络拓扑图例

这里是对上面拓扑结构列出的表：

表 2-1 图 2.1 对应的表

a (tty50) to b (tty51) at 1200 baud  
 b (tty52) to c (tty53) at 1200 baud

5. 到网络中的每台计算机上发命令，将每个虚拟串行端口和该机器上对应的物理设备联系起来。比如，命令应该具有如下形式：

```
In /dev / actual-tty /dev / virtual-tty
```

## 2.2 建立网络

建立网络应用 netutil 程序。该程序允许你定义构成网络的机器和串行端口。

为了建立网络，首先要创建定义网络的 Micnet 文件，然后把这些文件传输到网络中的每台计算机上。在所有的计算机都收到这些文件后，就可以启动网络并用它在两台计算机之间通信。

以下几节介绍如何建立网络。

### 2.2.1 创建 Micnet 拓扑文件

Micnet 文件是用 netutil 程序的 Install 任选项创建的。Install 任选项询问网络中每台计算机的名字和串行端口。当你提供这些信息之后，它自动地为每台计算机创建必需的文件。这些文件可以通过 netutil 程序的 Save 和 Restore 任选项传送给网络中的其他计算机。这意味着你可以在一台计算机上建立整个网络。

使用 install 任选项的步骤如下：

1. 作为超级用户注册
2. 输入：

```
netutil
```

按回车键。程序显示网络实用程序菜单。install 任选项是菜单的第一项。

3. 输入“1”，程序显示下面的信息：

```
Compiling new network topology
```

Overwrite existing network files (yes / no) ?

输入“Y”并回车，覆盖文件。要创建一个新网络就必须覆盖已有的网络文件。第一次安装网络时，这些文件包含不需要保存的隐含信息。如果是第二次安装系统或是扩充系统，那么比较明智的做法应该是在开始 install 任选项之前先保存这些文件的副本。这些文件可以用本章下一节将介绍的 save 任选项存储在软盘上。

在你输入“Y”之后，程序显示如下信息：

Enter the name of each machine (or press RETURN to continue installation).

Machine name:

4. 输入机器名并回车。在一行中可以输入多个机器名，中间用逗号或空格符隔开。在输入完所有名字后，按回车键继续执行下一步。程序显示你输入的所有机器名并询问是否要作什么改动。

5. 如果你想重新输入所有名字，则输入“Y”，否则输入“n”或者直接按回车键进行下一步。如果输入“n”，程序将显示：

For each machine, enter the names of the machines to be connected with it

Machine a:

Connected to:

6. 使用在设计网络时产生的计算机对的清单，输入连到指定计算机上的一组计算机的机器名。在一行中可以输入多个名字，中间用逗号或空格符隔开。当连到指定计算机的所有计算机的机器名都已输入完毕后，按回车键。程序询问连接到下一台计算机的计算机的名字。

7. 对所有剩下的计算机重复步骤 5。当程序询问每一组新的连接时，将显示它已知道与当前计算机相连的一组计算机的名字，你就不必再一次输入这些名字了。程序自动检查是否构成回路。如果发现，则拒绝接受构成回路的机器并询问另一个名字。

最后，当所有网络中的计算机的连接都已给出后，程序显示所有连接的清单，并询问是否希望作修改。

8. 如果希望重新输入连接关系则输入“Y”，否则，输入“n”进行下一步。如果你输入的是“n”，则程序显示如下信息：

For each machine pair, enter the tty name and tty speeds

For the a < = = > b machine pair.

Tty on a:

9. 使用在设计网络时产生的串行端口分配表，输入某一计算机对中的第一台计算机的串行端口名或编号（比如 tty14），按回车键，程序将显示以下信息：

Tty on b:

10. 输入该计算机对中另一台计算机的串行端口名并回车。程序将显示：

speed:

11. 输入速率（比如 1200）并回车。程序将询问下一对计算机的串行端口和传输速率。

12. 对所有剩下的计算机对重复步骤 8。当所有计算机对的串行端口和传输速率输入完毕之后，程序会显示这些信息并询问是否要作修改。

13. 如果希望重新输入串行端口和传输速率，应输入“Y”，否则，输入“n”完成安装。

14. 在用 netutil 完成安装时，程序将把你提供的信息复制到网络文件中，并且显示每一个被更新过的文件名字。

一旦文件被更新就可以用 save 任选项把 Micnet 文件复制到软盘上。

### 2.2.2 保存 Micnet 文件

使用 netutil 程序的 save 任选项可以在后援介质（如软盘）或硬盘上保存 Micnet 文件的副本。保存这些文件，能使你可以把它们传输到网络中的其他计算机上。在将文件保存到软盘片上之前应格式化好软盘片。将文件保存到硬盘上使你能够用 UUCP (C) 把它传输到其他机器上。

保存文件的步骤如下：

1. 作为超级用户注册

2. 输入：

```
netutil
```

并按回车键。程序显示网络实用程序菜单。

3. 输入数字 2 并回车，屏幕显示：

```
save to / dev / fdx (yes / no) ?
```

这里 x 是驱动器号。

4. 如果你准备使用指定的驱动器，则插入一张空白的且经过格式化的软盘片，等驱动器接受这张软盘后，接着输入“yes”并回车。如果不想使用这个驱动器，则输入“no”并按回车键。系统显示提示，询问你要用的软盘驱动器（或文件）的文件名。将一片空白的且经过格式化的软盘插入选择好的驱动器，等软盘驱动器接受这张软盘，然后输入该驱动器的设备文件名。缺省后援设备（磁盘驱动器）的名字在 / ctc / default / micnet 文件中进行说明。这个设备可以根据系统配置的改变而变化。

无论上述哪一种情况，程序都将把 Micnet 文件复制到软盘上。

5. 从驱动器中取出软盘。用软笔（不能用圆珠笔）标记这张软盘为“Micnet 盘”。

在复制完所有文件之后，就可以将它们传输到网络中的所有计算机上。

### 2.2.3 恢复 Micnet 文件

建立 Micnet 网络的最后一步是把 Micnet 文件从 Micnet 盘上复制到网络中的所有计算机上。用 netutil 程序的 restore 任选项可以完成这项工作。对网络中的每一台计算机安以下步骤执行：

1. 作为超级用户注册

2. 输入：

```
netutil
```

按回车键。程序显示网络实用程序菜单。

3. 输入数字 2 并回车，程序显示：

```
restore from / dev / fdx (yes / no) ?
```

这里 x 是驱动器号。

4. 如果你希望使用指定的驱动器，则把 Micnet 盘插入该驱动器。等待驱动器接受这张软盘，然后输入“yes”并回车。如果你不希望用该软盘驱动器，则输入“no”并回车。程序显示提示信息，询问你打算使用到磁盘驱动器的设备文件名。插入 Micnet 盘到选定的驱动器中，等驱动器接受了软盘之后，输入驱动器的设备文件名。

不论上述哪一种情况，程序都会把网络文件复制到适当的目录下，并且显示每个被复制的文件名。最后，程序显示：

```
Enter the name of this machine:
```

5. 输入你正在使用的计算机的机器名，并按回车键。程序将这个名称拷贝到该计算机的新的 /etc/systemid 文件中。如果需要，程序还要关闭用在该计算机上的串行端口，为在网络中使用作好准备。

在拷贝完这些文件之后，就可以用 start 任选项来启动网络。

## 2.3 启动网络

在 Micnet 文件传送到计算机上后，就可以用 netutil 程序的 start 任选项来启动网络。start 选择项启动 Micnet 程序，执行在网络中两台计算机之间通信所要完成的任务。

为了启动网络，对网络中的每一台计算机按以下几步执行：

1. 作为超级用户注册
2. 输入：

```
netutil
```

按回车键，系统显示网络实用程序菜单。

3. 输入数字 4 并按回车键，程序查找 /etc/systemid 文件。如果找到，就启动网络。否则，程序要求输入本计算机的机器名并建立这个文件。程序还询问是否要记录错误信息和传输过程。一般来说，除检查或测试网络之外并不需要这些记录。在第一次启动网络时，就每一个问题都回答“n”并按回车键。

网络一旦启动，就移到另外一台计算机处并在那儿启动网络。

值得一提的是，为了方便起见，可以让每台计算机在启动系统时自动启动系统，这只要在每台计算机的系统初始化文件 /etc/rc 中增加

```
netutil start
```

这一条命令就可以做到。要增加这条命令，需要用到正文编辑程序。如果你想要记录传输和错误情况，可以在这个命令中加入 -x 或者 -c 任选项。即使不用这些任选项，Micnet 在每次启动和停止网络时也要把注册和注销信息复制到系统 LOG 文件中。这意味着你必须定期清理这个文件。

## 2.4 测试 Micnet 网络

在第一次启动网络之后，应该测试一下网络，看是否完全装好。特别是必须检查是否每一台计算机都已连接到网络上。

要测试网络，需要知道如何使用 rcp (C) 命令。下面讲述如何测试网络以及在发现错误时应如何更正才能使网络工作正常。

### 2.4.1 检查网络连接

你可以通过用 rcp 命令给网络上的所有机器复制一个小文件的办法来检查是不是所有计算机都正确地连到网络上了。步骤如下：

1. 选择一台计算机
2. 作为超级用户注册
3. 使用 rcp 命令：

```
rcp / etc / motd machine: / tmp / test
```

这里 machine 是另一台机器的名字。对你的 Micnet 网络中的每一个节点重复这一命令。

4. 检查网络中所有机器的 / tmp 目录，看是否收到复制的小文件。

如果所有机器都收到该文件，则网络的安装是正确的。如果一台或多台计算机没有收到该文件，则这一台或多台计算机未被正确地连接到网络上，要解决这一问题，需要确定是哪一台计算机没有连接好引起的网络连接不正常。

### 2.4.2 使用 LOG 文件找出问题

可以通过对网络中每台计算机上的 LOG 文件进行检查来找出连接上的问题。LOG 文件记录每对计算机之间互相通信的情况。每对计算机有两个 LOG 文件（每台一个）。任一指定计算机的 LOG 文件保存在 / usr / spool / micnet 目录的子目录下。每个子目录以所在计算机对中的另一台计算机的机器名为目录名。可以打入：

```
cat /usr/spool/micnet/remote/machine-name/LOG
```

并回车，以此来检查 LOG 文件的内容。这里的 machine-name 必须是与当前正在使用的计算机相连的计算机名。

每个 LOG 文件应当包含系统的启动信息，它列出计算机对中各台计算机的名字以及相互连接所使用的串行端口。它同时还显示网络启的日期和时间。该信息显示如下：

```
daemon mn: running as MASTER
Local system: a
Remote system: b, / dev / tty52
Tue sep 24 22: 30: 35 1985
```

启动信息在每次网络启动成功后加到这个文件中。如果没有这个信息，则将会有一个或几个网络文件和目录不能找到。要保证使用 restore 任选项把所有网络文件传输到该计算机上。同时，要保证 / etc / systemid 文件中含有指定计算机的正确的机器名。

如果计算机对的连接已经开通，则每个 LOG 文件中都将含有“握手”信息。如果连接成功，则下面的信息将被写入 LOG 文件中：

```
first handshake complete
```

如果没有这个信息，则要弄清楚计算机中对另一台计算机上面的网络程序是否已经启动。在进行连接之前，必须启动该连接的两台计算机上的网络。如果连接的两端计算机上的网络都已启动而握手信息仍未出现，则可能是串行端口损坏或连接不正确所致。检查串行端口以保证电缆牢固地插在每台计算机正确的 RS-232 端口上。如果有必要的话，应用

一条已知是可用的电缆进行替换。

如果在 LOG 文件中已有启动信息和握手信息但是网络仍不能正常工作，那么问题是出在传输上。可以建立一个传输和发送过程中的错误信息记录。通过重新启动网络并请求：Micnet 记录整个传输过程和出错情况。这只需在 start 任选项询问是否要记录出错和传输过程时回答 y (yes) 即可。

出错记录中含有传输过程中产生的错误信息。每条错误信息列出错误原因和检查出有错的子程序。例如，下列信息表明 rsync 子程序从其他计算机上收到了一个错误响应（十六进制的 68）：

```
rsync: bad probe resp: 68
```

可以根据这一信息追踪出错的原因。一个常见的问题是由于电子干扰丢失了正在沿串行通信线传播的信息。要确保串行电缆不受干扰（例如，应保证串行电缆附近没有电动机、发电机或其他电磁辐射源）。还要保证电缆具有良好的导电性。

传输记录含有两台计算机之间正常传输信息的情况。每个记录列出传输方向、字节计数、所用时间以及日期时间。例如，下面的记录项显示在 16:22 时收到了 29349 个字节信息：

```
rx: 293496 2:22 @16:22
```

这次传输花费的时间是 2 分 22 秒。你可以使用这一记录看看信息是否真的传输了。

## 2.5 停止网络

可以用 netutil 程序的 stop 选项来停止网络。该任选项终止 Micnet 程序的执行，停止在网络中的计算机之间的信息传输。

要停止网络，应在网络中的每台计算机上按以下步骤进行操作：

1. 作为超级用户注册
2. 输入：

```
netutil
```

并回车，程序显示网络实用程序菜单。

3. 输入数字 5，并按回车键。程序停止在该台计算机上运行的网络程序。

## 2.6 修改 Micnet 网络

你可以在任何时候通过修改一个或几个 Micnet 文件的办法来修改 Micnet 网络，然后用 netutil 程序重新安装网络。对于较小的变动（比如修改机器名的拼写），可以直接用一个文本编辑程序来修改 Micnet 文件。

在对一个文件进行修改之前，应该做一个副本。可以用 CP 命令来做这个副本。使用 mv 命令把老文件用更新后的文件代替。在一台计算机上一次可修改一个或多个文件，这些文件必须用 netutil 程序的 save 和 restore 任选项传输到网络中的其他系统上。这项工作只能在终止网络运行后执行。



## 第三章 用 UUCP 建立远程网络

### 3.1 简介

本章讲述如何通过 UUCP，用普通电话线和调制解调器为你的计算机建立远程网络系统。

使用 UUCP，还可以用直接串行通信线将物理上不同类的 UNIX 系统或者 XENIX 系统连接在一起。另外，UUCP 系统还可以用来解决两个 Micnet 网络因距离问题或电缆问题而不能相连的问题。

---

注意：UUCP 不是一个终端仿真程序。如果你想用调制解调器拨号进入另一台计算机，应参考第四部分第一章“使用调制解调器”的内容。

---

如果你打算在 UNIX 和 Xenix 系统之间作大量的文件传输，就应该建立 UUCP 连接。

#### 3.1.1 什么是 UUCP?

UUCP 软件包允许 XENIX 和 UNIX 系统作为远程网络的一部分进行通信。UUCP 这个名字是“UNIX-to-UNIX COPY”的意思。UUCP 软件包包含了提供以下功能的一组程序：

- 远程文件传输 (UUCP)
- 远程命令执行 (UUX)
- 与远程点通邮 (通过 mail)

UUCP 通常是使用电话线，将指定的远程机器按照命令或调度方案连接起来，或者是拨号出去或者是允许其他机器拨号进来。

UUCP 在管理通信的交通量时是采用批处理的办法，在系统之间正在通信时如果有新的通信请求，则选择该请求存储起来，等当前通信完成之后再作处理。在 UUCP 命令执行时，所需的工作文件和数据文件都建立在 /usr/spool/uucp 目录下。uucico 程序在该目录中查找工作文件中包含的命令并加以执行。虽然命令有可能马上得到执行，但大多数系统都是根据白天的调度对其他系统进行访问（为了减少费用，对其他系统的访问常常安排在夜里进行）。

#### 3.1.2 如何使用本章

本章讲述如何建立 UUCP 系统，包括硬件安装和软件配置。还有几节谈到远程维护以及可能出现问题的解决办法。

下面是建立 UUCP 网络所必须遵循的过程：

- 连接和配置调制解调器或者直接线连
- 用 uuinstall 程序配置 UUCP 软件

- 为可能访问你的系统的所有系统建立注册帐号
- 对每一个远程点测试所作的连接是否正确

---

注意：如果打算通过 UUCP 系统进行远程邮件工作，应参见“建立电子邮件”一章，按照那里的命令通过 UUCP 配置通邮的交通量。

---

配置 UUCP 的最重要的任务是编辑几个充当 UUCP 数据库的控制文件。下面几节将描述这些文件的功能，在本章后面的“在你的系统上配置 UUCP”部分讲述这些文件包含的内容。uinstall 实用程序为你编辑这些文件并解释每一项含义，该实用程序还包括很广泛的帮助信息。在运行 uinstall 之前应仔细阅读“在你的系统上配置 UUCP”部分，以便理解 UUCP 数据库的含义。

### 3.1.3 你需要什么？

在你的计算机上安装一个直接线连的通信系统，你需要用到：

- 在你的计算机上至少有一个 RS-232 串行通信线（或串行端口）可用于 UUCP。
- 用 custom (ADM) 从你的 UNIX 系统中提取出来的 UUCP 和 MAIL 软件包。

如果希望计算机通过调制解调器成为拨号入 / 拨号出节点，则需要用到：

- 一个调制解调器。支持的调制解调器包括 Hayes, Penril, Ventel, Vadic, Rixon, AT&T 和 Telebit 模式。你可以为其他调制解调器提供拨号入口或拨号程序（最好是拨号程序）。Hayes Smartmodem 1200 或 2400 和其他兼容调制解调器的命令在本章稍后部分给出。

- 一个访问其他电话系统的标准电话插座。
- 一条将串行端口和调制解调器相连的电缆。

### 3.1.4 UUCP 命令

UUCP 被分成两类：用户程序和管理程序。下面将分别对这两类程序加以说明：

#### • 用户程序

**cu:** 将你的计算机和远程计算机连在一起，使你能够同时在两者上注册，你可以在自己的计算机或者远程计算机上传输文件或者执行命令而不必放弃初始连接。

**ct:** 将你的计算机和远程终端连在一起，以便远程终端的用户注册。远程终端的用户可以访问计算机并且要求计算机回访。接着计算机解除初始连接，使得回访终端时远程终端的调制解调器可用。

**UUCP:** 将文件从一台计算机上拷贝到另一台上。它建立工作文件和数据文件，将作业排队以便传输，并调用 uucico 控制程序与远程计算机联系。

**uupick:** 当文件用 uuto 传输时被放于 /usr/spool/uucppublic/receive 目录中，由 uupick 取出。

**uustat:** 显示要求的传输（UUCP, UUTO 或 UUX）的级别，它还提供了一种控制队列传输的方法。

**uuto:** 将文件从一台计算机拷贝到另一台计算机上的公共 spool 目录

/usr/spool/uucppublic/receive 中。它不象 UUCP 那样让你把文件拷贝到另一计算机上的任意可访问的目录上，而是将文件放到特定的 spool 目录中，并告诉远程用户用 uupick 来取。

**UUX:** 建立在远程计算机上执行命令所需的工作文件、数据文件和执行文件。工作文件的内容与 uucp 和 uuto 建立的工作文件完全一样。执行文件包含将在远程计算机上执行的命令字符串和数据文件表，数据文件是命令执行所需的那些文件。

#### • 管理程序

大多数管理程序和控制文件都是在 /usr/lib/uucp 目录中。uuinstall 和 uulog 是两上例外，它们分别在 /etc 和 /usr/bin 目录中。

**uuccheck:** 检查基本网络目录、程序和支持文件是否存在，它还检查 Permissions, Systems 和 Devices 文件中的语法错误。

**uuclean:** 清理 spool 目录。通常通过称为 uudemmon.clean 的外壳程序执行。

**uuinstall:** 是 UUCP 控制文件和端口的配置程序，还用于将 Version2 的 UUCP 控制文件转化为 Honey DanBer 格式。

**uulog:** 显示指定计算机的 log 文件。log 文件是为与你计算机通信的每个远程计算机建立的。log 文件包含 UUCP、UUTO 和 UUX 每次使用的记录。

**uutry:** 测试对访问的处理能力，并做相当一部分调试工作。它激活 uucico 以建立通信链接。

### 3.1.5 UUCP 的目录

与 UUCP 相关的有三个目录：

**/usr/spool/UUCP:** 这是 UUCP 的工作目录。工作文件，Log 文件和所有的 UUCP 通信情况存放于此。

**/usr/spool/uucppublic:** 这是大多数文件传输使用的公共可读或可写目标目录。

**/usr/lib/UUCP:** 大多数 UUCP 程序存放在这里，另外支撑数据库 (supporting database) 或控制文件也存放于此。主要的用户程序，包括 UUX 和 UUCP，在 /usr/bin 目录中。

/usr/lib/UUCP 目录还包含 UUCP 的配置文件，其中最重要的有：

**Systems:** 包含与一个远程计算机建立连接所需的信息，包含与远程计算机相连的连接设备名，计算机何时可以联系上，电话号码，注册顺序以及口令等。

**permissions:** 给要向你的计算机传输文件或从远程系统在你的计算机上执行命令的计算机定义访问级别 (权限)。

**Devices:** 包含关于端口名，速度，自动 Call 部件 (modems) 类型，直接线连和网络设备的信息。

### 3.1.6 UUCP 背景程序

UUCP 的通信是由三个守护程序，或称管理程序来处理的，这些程序在后台执行，处理文件传输和命令执行。(守护程序也可以作为普通命令执行)：

**uucico:** 选择连接所用的设备，建立到远程计算机的连接，执行要求的注册序列工

作，检查权限，传输数据，执行文件，记录结果，如有必要还要用邮件通知用户传输已完成。当地计算机的 uucico 控制程序访问远程计算机时，在对话时和远程计算机的 uucico 控制程序对话。

uuxqt: 实施远程程序的执行。它查找 spool 目录中由远程计算机送过来的可执行文件 (X. file)，当找到一个 X. file 时，uuxqt 将该文件打开，获取执行该文件所需的数据文件表，然后检查所需数据文件是否已准备好可以访问。uuxqt 还要检查是否有权利执行要求的命令。

uusched: 调度在 spool 目录中队列的工作。在启动 uucico 控制程序之前，uusched 打乱访问各远程计算机的顺序。

### 3.1.7 UUCP 是怎样工作的?

当你输入一个 UUCP 命令时，程序建立一个工作文件，通常还有一个用于传输的数据文件。工作文件包含传输文件所需的信息，数据文件是指定的源文件的副本。这些文件在 spool 目录中建立好之后，启动 uucico 控制程序。

uucico 控制程序要建立到远程计算机的连接。首先它从 Systems 文件中获取建立与远程计算机的连接所需要的信息。这就是 uucico 确定使用什么类型设备来建立连接的方法。接着，uucico 在 Devices 文件中查找与在 Systems 文件中所给设备类型相匹配的设备。当找到可用的设备后，它便开始建立连接并在远程计算机上登录。这一步工作完成之后，它启动远程计算机上的 uucico 控制程序，这两个 uucico 控制程序协商文件传输中将要用到的协议。接着，本地 uucico 控制程序将你想发送的文件传输到远程计算机上。远程计算机的 uucico 控制程序将文件放到远程计算机上的指定路径名下。在你本地计算机传输完成之后，远程计算机也可能有列好队的文件要发给你的本地计算机，但这一步传输可能因 Permissions 文件中的权限而被拒绝执行。如果这样，远程计算机必须建立到你本地计算机的连接才能执行这些传输工作。

如果远程计算机或者你选用来建立连接的设备没有准备好，则请求保留在 spool 目录的队列中。当 uusched 控制程序启动时，它查找 spool 目录中剩余的工作文件，产生一个随机顺序来处理这些请求，接着按前面描述的步骤启动传输过程 (uucico)。

### 3.1.8 一个 UUCP 执行范例

下面追踪 UUCP 命令的执行步骤:

1. 一个名为 kilgore 的系统上的用户希望将一个名为 minutes.01.10 的文件传送到一个名为 obie 的远程系统上。为了完成这件工作，该用户输入了以下命令:

```
UUCP minutes.01.10 obie \ !usr / spool / uucp pubic
```

---

注意: 在使用 csh 时应去掉上面命令行中的感叹号 (!)

2. 在 /usr / spool / uucp / obie 目录中建立工作文件 c。obienxxxx, 这里 xxxx 是作业号。

3. uusched 控制程序对请求加以调度，由 uucico 执行。

4. 到了执行时间里，uucico 首先检查 System 文件，确认 obie 是一个存在系统并且此时允许接收访问。

5. 接着，uucico 用在 Systems 文件中的信息，确定相应调制解调设备和串行端口在 Devices 文件中的位置。

6. 使用 Systems 文件中的电话号码和 Devices 文件中的调制解调器类型，uucico 在拨号文件中选用适当的调制解调命令（或者运行 /usr/lib/uucp 目录中的拨号程序）与远程系统相连。

```
Systems:      obie Any ACU 2400 14081234567 \  
              --ogin: -BREAK-ogin: nuucp ssword: mavra  
  
Devices:      ACU tty1A-2400 dialHA24  
  
Permissions: LOGNAME=ukilgore MACHINE=kilgore\  
              READ=/usr/spool/uucppublic:/usr/kilgore\  
              WRITE=/usr/spool/uucppublic:/usr/kilgore\  
              REQUEST=no SENDFILES=call \  
              COMMANDS=rmail:rnews:uucp
```

图 3.1 UUCP 控制文件例

7. uucico 建立一个锁文件（LCK..ttyxx）锁住串行端口，另一个锁文件（LCK..obie）锁住在 /usr/spool/uucp 中被访问的系统。

8. uucico 使用 Systems 文件中定义的注册顺序和口令登录 obie 系统。而在开始实际传输之前要由 obie 系统的 uucico 确认 kilgore 的存在。

9. 发出 call 的系统即 kilgore，称为客系统，接受 call 的系统即 obie，称为主系统。主系统的 uucico 检查本地 permissions 文件以确认客系统有权传输文件。

10. 客系统（kilgore）以包为单位传输文件，在包传输过程中要进行错误检查，如果出现误传，则需要重发此包。接收过程中，文件被放于 /usr/spool/uucp 目录的暂时文件（TM.xxxx）中。传输完成之后，该文件被移到指定目的地，在本例中是 /usr/spool/uucppublic/minutes.01.10。

11. 每个机器在 Log 文件中都有传输的另一方的记录，例如在本例中，obie 就把本次信息交换记录在名为 /usr/spool/uucp/.log/uucp/kilgore 的文件中。

12. 除非主系统 obie 提出特别请求，此时应发出挂起请求，终止连接，移去锁文件。

对于通过 uux 的远程命令执行，要在 /usr/spool/uucp 目录中建立 X.file 执行文件。uuxqt 控制程序在本目录中查找可执行文件 X.file，并检查 Permissions 文件以确认执行权限，然后加以执行。

### 3.2 将两个本地系统直接线连

用导线将两台计算机连接起来，应遵循以下步骤：

- 在每台机器上选择一个串行端口

- 用选定的串行端口通过串行线 (RS-232) 将两台计算机相连
- 确定哪台机器是拨号入者, 哪台是拨号出者, 后者发出寻呼 (call) 并登录到拨号入机器上去。

### 3.2.1 选择串行端口

在每一台机器上都必须选择想要使用的 RS-232 串行端口 (/dev/tty<sup>nn</sup>)。如果没有可用的端口, 你必须安装新的串行端口, 或者将已经连接上的设备移去以空出一个可用端口。如果你移去的是终端, 则应保证没有人注册。

找出与端口对应的设备特别文件名 (见本书“UNIX 目录与特别设备文件”一章)。设备名应有如下格式:

/dev/tty<sup>nn</sup>

这里 nn 是相应端口的编号。比如, /dev/tty<sup>la</sup> 通常对应于设备 COM1。实际端口名将在稍后的步骤中用到。

串行端口应由 UUCP 所有, 为了保证端口由 UUCP 所有, 输入如下命令:

```
chown UUCP /dev/ttynn
```

这里 nn 是相应端口的编号。

### 3.2.2 连接串行电缆

用一条 RS-232 电缆将两台计算机连在一起。实际的管腿配置有时随机器的不同而不同。

一般情况下, 电缆应该把一台计算机串行端口上的管腿 2, 3, 7 与另一台计算机串行端口上的同号管腿相连。有时电缆必须交叉连接即一台机器串行端口上的管腿 2 与另一台上的管腿 3 相连, 反之亦然。由于连接方式可能不同, 应检查每台计算机的硬件手册以确定正常的管腿连接。

下面对一个连接进行测试, 假设两台机器都用 tty<sup>2a</sup> 作串行端口。

要测试两台机器的线连, 应按如下步骤:

1. 使每台机器的串行端口无效。在每台计算机上都输入命令:

```
disable /dev/tty2a
```

同时使调制解调器控制端口也无效:

```
disable /dev/tty2A
```

2. 将串行线的一端连到一台机器上, 另一端连到终端的标准数据端口。

3. 在你的计算机上输入如下命令:

```
(stty 9600; date) < /dev/tty2a > /dev/tty2a
```

tty<sup>2a</sup> 是我们例中的串行端口由 date 命令提供标准输出。

你应该在终端的屏幕上看到 date 命令的输出。在另一台机器上重复这一过程。

如果工作不正常, 检查下列几个方面:

- 导线正确地连接到两端
- 导线的延展性好
- 终端配置正确 (波特率等)

- 串行端口被置无效
- 使用的管腿号正确

### 3.3 用调制解调器连接远程 UUCP 系统

使用调制解调器，你可以通过标准电话线实现计算机通信。安装调制解调器步骤如下：

1. 选择一个串行端口
2. 设置拨号配置
3. 连接调制解调器，设置开关
4. 检查连接

以下各部分将详述上列各步骤。

#### 3.3.1 选择串行端口

选择你要用的 RS-232 串行端口来连接调制解调器。如果没有可用的端口，必须安装一个新的端口，或者去掉一个连接于其上的设备以空出一个可用端口。如果去掉终端，则必须保证无人注册。

查阅本书“UNIX 目录和特殊设备文件”一章找出与选定端口应的设备特殊文件名。设备名应具有如下形式：

`/dev/ttyn`

这里 `nn` 是相应端口的编号。例如 `/dev/tty1A` 对应于 COM1。在以后步骤中你会用到实际端口名。

注意：`/dev/ttyla` 与 `/dev/tty1A` 是同一个物理端口，`ttyla` 用于终端和直接连接，`tty1A` 应用调制解调器连接。

串行端口应由 UUCP 所有。为了保证这一点应输入如下命令：

```
Chown uucp /dev/ttyn
```

这里 `nn` 是相应端口的编号。

#### 3.3.2 设置拨号配置

调制解调器可用来发送和接收寻呼。你必须在调制解调器上设置合适的开关。下面的命令专用于 Hayes 及兼容调制解调器，但也支持其他调制解调器。按照下列步骤设置 Hayes Smartmodem 1200 及其兼容的调制解调器：

1. 打开调制解调器前盖，找到 8 位配置开关的位置。（查阅调制解调器参考手册找出在你的调制解调器中确定开关位置的方法）

2. 开关的形式如下：

	1	2	3	4	5	6	7	8
上	•	•		•	•	•	•	
下			•					•

表 3.1 解释各位置上、下的含义:

表 3.1 开关位置的含意

开关	位置	含义
1	上*	Modem 响应来自计算机的 DTR
	下	Modem 强制 DTR 为高, 所以不需要从计算机来的信号
2	上*	结果代码为英文形式
	下	结果代码为数字形式
3	上	没有结果代码
	下*	发送结果代码以响应每个 modem 命令
4	上*	命令回应(echo)
	下	命令不回应(noecho)
5	上*	Modem 回答电话
	下	Modem 不回答电话
6	上*	载波出现时, CD 有效
	下	CD 和 DSR 强制为高
7	上*	Modem 连在单线电话上
	下	Modem 附于多线电话上
8	上	Modem 不识别拨号命令
	下*	Modem 识别拨号命令

星号 (\*) 表示要求的开关设置。如果你的调制解调器是另外一种, 则应该查阅相应的参考手册以确定正确的开关设置状态。

### 3.3.3 连接调制解调器

在调制解调器的拨号配置设置好之后, 就可以把调制解调器连到你的计算机上。为使调制解调器工作正常, RS-232 电缆应该提供如表 3.2 所示的管腿连接:

表 3.2 RS-232 管腿连接

名称	DTE	DCE
保护地	1	1
RD	2	2
TD	3	3
RTS	4	4
CTS	5	5
DSR	6	6
SG	7	7
CD	8	8
DTR	20	20



---

注意：计算机的串行连接器必须有一个 DTE 数据终端设备。假定调制解调器有一个 DCE（数据通信设备）。

---

熟悉调制解调器参考手册中给出的安装指令，然后按下面的步骤进行：

1. 将 RS-232 串行电缆连接到调制解调器上的串行端口连接器上，再连到你计算机的串行端口连接器上。
2. 将电话线电缆插入调制解调器上的“line”或者“wall”连接器中，再连到电话插座中。
3. 加上调制解调器电源。

### 3.3.4 配置 Hayes 2400 及兼容调制解调器

虽然大多数调制解调器的安装过程都很类似，但 Hayes 2400 Smartmodem 及其兼容调制解调器如果作为拨号输入线路使用需要联机配置。要注意如果 Hayes 2400 没有用 2400 波特命令设置，就不会应答用 2400 波特传送的电话。

1. 确认 Devices 文件包含线路的一个条目：

```
ACU ttynn - 300-2400 / usr / lib / uucp / dialHA24
```

2. 用下面的命令使串行端口暂时无效：

```
disable ttynn
```

3. 必须通过 CU (C) 发出命令来配置调制解调器。输入下面命令并回车：

```
CU -S2400 -1 / dev / ttynn dir
```

这里 nn 是串行线路的“tty”编号

4. 然后，输入以下命令来配置调制解调器，它们将保存在调制解调器的非易失性存储器中。如果不想保存这些设置，则不要打入最后一个命令 (AT&W)。命令列在左边，右是对它们所作的简要描述。每个命令之后都要跟“回车”：

AT&F: 取出厂家做好的配置

ATT: 音调拨号

ATLO: 低扬声器值

AT&D2: 设置DTR“2”：当DTR失落时继续挂起

AT&C1: 设置CD信号为“1”

ATS0=1: 回1响之后应答电话 (AA灯应亮)

ATE0: 不响应

ATQ1: 安静方式 (调制解调器在收到此命令及随后的信号后将不再响应“OK”信号)

AT&W: 在非易失性存储器中的保存设置

5. 从 CU 退出需要键入一个上波折线和一个句点，以及一个回车：

~.

有时在输入“~.”之前还需按一下回车。

6. 当你希望调制解调器接收访问时应重新使端口有效：

enable ttynd

现在调制解调器已经配置好可以使用。

### 3.3.5 可变速率调制解调器

有的调制解调器可以根据远程系统发来的信息确定连接的波特率。这些调制解调器在发出载波检测 (carrier Detect) 信号之前先将连接的波特率通知本地系统。Hayes 2400 调制解调器检测出连接的不同波特率, 并在连接成功建立后退出时通知 UUCP 和 CU 程序。

Devices 和 Systems 文件的速度域中指出了连接的波特率范围。如果一个拨号系统支持 300~2400 波特率, 则应在 Devices 文件的速度域中填入波特率范围如下:

300-2400

如果一个拨号系统或调制解调器不允许可变的波特率, 则在速度域内填入单一的波特率。如果某一远程系统支持几种不同的速率, 则在 Systems 文件的速度域中填入波特率范围。如果这个远程系统以单一波特率相连, 则在 Systems 中填入该数值。UUCP 在连接时将 Systems 和 Devices 的波特率范围都传递给拨号系统。如果拨号系统连接的波特率超过了范围, 则返回一个波特率越界错, 否则返回的是连接的波特率。

### 3.3.6 测试调制解调器

在安装调制解调器的最后一步, 应该测试调制解调器, 看它是否确实能够发送和接收寻呼, 一旦你确认了调制解调器能够正常工作, 这时就可以使用通信系统。

1.如果你使用的是 Hayes 1200 或者兼容的调制解调器, 则应该将调制解调器的音量开关调到一位合适的位置。要能听到调制解调器工作才算这一步测试工作成功。可查阅你调制解调器的参考手册以确定开关的位置。

2.应保证在 Systems 文件中有你打算访问的系统的条目, 并且在 Devices 文件中有与 ttynd 相匹配的条目。

3.通过输入下面命令启动 uucp 程序:

```
/usr/lib/uucp/uucp -x6 sitename
```

4.仔细听调制解调器的动静。在拨号时每一个数字的拨号声都应该可以听到。当别的调制解调器连接时可以听见一个尖锐的信号声, 接着什么声音也没有。

5.拨号程序自动地将它不能完成的寻呼所做连接释放。不要按 <DEL> 键进行中断, 否则会停止 uucp 的执行。可以让拨号程序挂起。

6.如果信号没有听到, 则检查如下几条是否做到:

- 调制解调器连接到了电话插座上。
- 插座连接到电话系统上
- 正确的电话号码是在 Systems 文件中

7.如果你没有听到调制解调器拨号, 则应检查以下几条:

- 音量开关处于 up 状态
- 调制解调器连接到正确的串行线路上, 电缆连接没有松动
- 正确的 tty 线路在 Devices 文件中

- 调制解高器的开关是打开的
- 在 /usr/spool/uucp 目录下没有 LCK.. 文件

8.uucico 只允许每隔 10 分钟对某个系统进行一次 call, 在再试一次之前应该等一会儿, 或者在 /usr/spool/uucp/status 目录中去掉与你正访问地点相对应的文件。(这个文件保持连接状态, 它的存在阻止下一次访问发生, 直到可以调度下一次访问为止)。

### 3.4 在系统中配置 UUCP

要配置 UUCP 系统, 必须编辑一系列文件, 包括存放 UUCP 程序信息的文件, 控制 UUCP 程序动作的文件, 等。UUCP 控制文件在 /usr/lib/uucp 目录中。可以用标准文本编辑程序来对这些文件进行修改, 或者使用这里讲述的 uuinstall (ADM) 程序。不管用哪一种方法, 你都应该运行一下 uuinstall 程序, 按照“用 uuinstall 建立控制文件”部分讲述的命令执行。

#### 3.4.1 一个重要考虑: 访问还是被访问

有三种办法来配置 UUCP 系统:

- 只允许拨号输入的系统
- 只允许拨号输出的系统
- 允许拨号输入/输出的系统

作为一个允许拨号输入的系统, 其他计算机可以加以访问并且在系统上登录, 可以传输文件, 执行某些命令。

作为允许拨号输出的系统, 你的计算机可以访问其他计算机并在上面登录, 可以向远程计算机传送文件, 以及进行本地及远程的命令执行。

#### 3.4.2 用 uuinstall 建立控制文件

下面将讲述作为 UUCP 数据库的控制文件的配置问题。uuinstall (ADM) 实用程序为配置这些文件提供了一种简单的方法。uuinstall 实用程序含有一系列的 help 文件 (在菜单中按“?”可以调出)。在对每个控制文件的用法有了一定的理解之后, 按以下步骤执行:

1. 作为超级用户注册, 输入以下命令以启动 uuinstall 实用程序:

```
/etc / uuinstall
```

Sysadmsh 的用户可选择:

```
System→Configure→Network→UUCP
```

uuinstall 主菜单显示如下:

```
UUCP Administration Utility
```

1. Display or update site name
2. Display or update list of remote sites (Systems)
3. Display or update direct- or dial-out lines (Devices)
4. Display or update direct- or dial-in lines (/etc/inittab)
5. Check consistency of UUCP files

6. Test Connection with remote sites
7. Convert old UUCP files to new format

Choose an option (1-7) , or enter "q" to quit

按下面的方法选择 `uinstall` 选项:

- 如果你在安装时没有设置系统名字, 或者希望修改系统名字, 应选用选项 1。
- 要选择拨号输入或输出所用的设备并把它们输入到 `Devices` 文件中, 应选任选项 2。
- 在 `Systems` 文件中建立条目, 标明你的系统将要与之通信的系统, 应选择 3。
- 增加在 `/etc/inittab` 文件中将要用到的 `tty` 线路应选择 4。

2. 如果别的系统想要访问你的系统, 应按本章“为拨号输入系统建立注册帐号”部分的要求建立注册帐号。另外, 你还应该定义一个安全方案, 即在 `Permissions` 文件中定义哪些命令和目录可以使用。

应该注意的是, 有些文件有多个选项, 有些用得不太多的选项和控制文件被提出来, 在本章后面“特殊的 UUCP 配置选择项”部分加以讨论。

### 3.4.3 选择和定义 UUCP 端口

我们前面讲过, 你必须选择一个串行端口, 如果是用于只允许拨号输出的系统应使之无效, 而如果用于拨输入的系统, 则应该使之有效。如有必要还应在 `/etc/inittab` 文件中定义串行通信线路的条目。

---

注意: 如果你是用手工编辑 `/etc/inittab` 文件而不是使用 `uinstall` 实用程序, 你就必须对文件 `/etc/conf/cf.d/init.base` 做同样的修改, 这是因为每次重新连接核心时 (如同增加某一驱动程序时那样), `/etc/inittab` 文件都要根据 `/etc/conf/cf.d/init.base` 文件中的条目重新加以构造。

---

1. 选择串行端口。对于拨号输入或拨号输出线路应选择带有调制解调器控制的端口, 而对于直接线连的系统则应选择不带调制解调器控制的端口 (前者如 `/dev/tty1A`, 后者如 `/dev/tty2a`)。

2. 使串行端口无效。如果你使用了调制解调器, 应保证经过测试工作正常。如果串行线路是拨号输入线路, 应用 `enable` 代替 `disable`, 使串行端口有效。输入命令:

```
disable /dev/ttynn
```

这里 `nn` 是你的串行端口的编号。如果端口已经处于无效或者有效 (在你用 `enable` 代替 `disable` 时) 状态, 这一命令执行后会在屏幕上出现一条错误信息, 但你可以忽略掉不加考虑。

3. 如有必要应编辑 `/etc/inittab` 文件。这个文件包含可能有的注册终端的清单。下面是一个拨号输出端口的条目的例子:

```
Se1A: 2: off: /usr/lib/uucp/uugetty -t 60 tty1A 2400
```

下面是拨号输入端口条目的例子:

```
Se1a: 23: off: /etc/getty tty1a m
```

在运行 `mkdev serial` 程序时就建立了 `/etc/inittab` 中的条目。如果你需要修改条

目，可以通过文本编辑程序来实现。注意，在用作拨号输出的端口上必须使用 `uugetty` (ADM) 程序。如果线路由拨号输入和拨号输出所共享，则必须保证它在 `/usr/lib/uucp/Devices` 和 `/etc/inittab` 中都有正确的条目。

#### 3.4.4 为拨号输入系统建立注册帐号

拨号输入系统必须为访问它的系统提供注册条目（帐号）。

UUCP 注册帐号跟普通用户帐号是完全一致的，但它有一个特殊的注册目录和注册程序，不同于普通的用户目录和 shell 程序。

值得一提的是，“UUCP”不应该被看作是 UUCP 用户或注册帐号的名字，它实际上是 UUCP 的所有者或者管理者的名称。

应按照下面步骤建立 UUCP 注册条目：

1. 为 UUCP 注册选择一个新的用户名和用户标识号。用户名可以是数字和字母的组合串，长度不超过 8 个字符。用户标识号必须是 50~65536 之间的整数。

应保证名字和标识号都是唯一的。UUCP 的注册条目的名字或标识号不能与其他任何注册条目的名字或标识号相同。

2. 要建立新帐号，应启动 `Sysadmsh` 作如下选择：

Accounts → User → Create

3. 使用下面信息来建立帐号：

注册 shell: `/usr/lib/uucp/uucico`

主目录: `/usr/spool/uucppublic`

口令可任选。

#### 3.4.5 在 Systems 文件中为远程系统增加条目

`Systems` 文件 (`/usr/lib/uucp/Systems`) 中包含着 `uucico` 控制程序建立到远程计算机的通信连接所需要的信息。该文件中的每一个条目代表你的计算机可以访问的一台计算机。

此外，通过配置 `Systems` 文件还可以防止本文件中没有出现的计算机登录到你的计算机上。对某一特定的计算机可以有多个条目与之相对应，多余的条目表示额外的通信路径。

---

注意：如果你把自己的系统建立成不发出 calls（即不访问其他系统）的、只允许拨号输入的被动的系统，那么你只需在 `Systems` 文件中加入将要访问你的系统的各系统的名字即可。

---

`Systems` 文件中的每个条目都具有如下的格式（每个域之间以空格隔开）

Sitename, Schedule, devices, speed, phone, login-script

这里：

sitename: 包含远程计算机的系统名。

schedule: 是表示远程计算机可以访问的时间，是由一周中哪一天，一天中什么时间等信息组成的字符串。

device: 是应该用来建立到远程计算机的通信连接的设备的类型。

speed: 表示用来建立通信连接的设备的传输速率。

phone: 提供远程计算机为自动拨号准备的电话号码。

login-script: 包含注册信息。

下面分别加以说明。

### 1. schedule 域

schedule 域包括 3 个子域, 第一个即 day 域是必需的, 另外两个, time 和 retry, 是任选项。语法如下:

```
day [time]; retry]
```

“day”子域可能包含以下内容之一:

Su,Mo,Tu,We,Th,Fr,Sa: 代表周日到周六的一天。

wk: 代表从周一到周五的任一天。

ANY: 代表所有日子。

Never: 如果 Schedule 域是 Never, 则你的计算机将不会访问其他计算机, 寻呼一定是由远程计算机发出的, 换句话说, 你的计算机总是处于等待远程计算机访问的被动状态。

任选项 time 子域应该是一天 24 个小时形式的时间范围, 如 0800-1230。如果没有标明 time 子域, 则表示一天中的任一时刻都可以进行访问。时间范围可以跨越 00:00 时刻, 例如 0800-0600 表示除了上午 6 点到上午 8 点之外的任何时刻都可以进行访问。

例如, 下面所列举的条目表示星期一、星期三和星期五中上午 9 点至正午 12 点的时间内可以进行访问。(为清楚起见, Schedule 域部分用黑体字标出)

```
grebe MoWeFr0900-1200 ACU D1200 14087672676 \
```

```
ogin: nuucp ssword: Crested
```

你还可以标明不同日子的不同时间, 只要中间用逗号隔开即可。这一方法可以用来说明更复杂的时间情况。下面的例子表示星期一到星期四中下午 5 点至早晨 8 点, 以及星期六和星期日全天可以进行访问。Schedule 域部分同样用黑体字标出:

```
gorgon Wk1700-0800, SaSu ACU D1200 14087672676 \
```

```
ogin: nuucp ssword: DontLook
```

任选项 retry 子域用于表示在一次访问失败之后再进行一次尝试之前必须等待的时间(时间单位为分钟)。该子域的分隔符是分号(;). 例如, 下面的形式表示“可以在任一时刻访问, 但在一次访问失败之后必须等待至少 9 分钟才能再作一次访问尝试”:

```
Any; 9
```

### 2. Device 域

device 域选择设备类型, 大多数情况下选择 ACU (Automatic Call Unit 自动寻呼部件)。比如, 下例域中用到的关键词 ACU 即与 Devices 文件条目的第一个域相匹配。

```
Systms: gorgon Any ACU D1200 14987672676 \
```

```
ogin: nuucp ssword: DontLook
```

```
Devices: ACU tty2A - D1200 hayes
```

### 3. Speed 域

本域包含一个字母和一个速度值（比如 C1200, D1200），字母用以区分拨号系统的级别。有的设备可以在任何速率下使用，这时可以用关键词 Any，不过我们仍建议指明可以选用的实际速率范围。（如果在 Systems 和 Devices 条目中的 speed 域都用了 Any，则认为速度为 1200）。例如，下例中的本域必须与相应设备文件入口中的速度域相匹配。

```
Systems:    gorgon Any ACU D2400-9600 14087672676 \  
           ogin: nuucp ssword: DontLook
```

```
Devices:    ACU tty1A - D2400-9600 hayes2400
```

如果不需要本域的信息，可以用一个“-”号为表明该域空。

#### 4. phone 域

本域为调制解调器拨号系统提供电话号码。电话号码由任选的字母缩写和一串数字组成。如果使用了字母缩写，则它必须是 Dialcodes 文件中列出的一个缩写。比如，

```
Systems:    gorgon Any D1200 CA2676 \  
           ogin: nuucp ssword: DontLook
```

```
Dialcodes:    CA 9=408767
```

在这个串中，等号（=）告诉 ACU 在开始拨数字码之前应等待，直到听到两下拨号声为止。

如果你的计算机连接在一个局域网（LAN）开关或者端口选择器上，你就可以访问连接到该开关的其他计算机。这些计算机 Systems 文件条目的 phone 域中并没有电话号码，相反该域内容是一个标记，该标记必须传递到局域网开关，使它知道你的计算机希望与哪一台计算机通信。（此标记其实就是系统名称）。相应的 Devices 文件条目的末尾应该有 /D 标，表示访问不是通过电话号码进行的。

#### 5. Login-script 域

login-script 的作用是开放调制解调器之间的通信，确认并发送一定的登录注册和口令信息。本域以用空格隔开的一系列的域和子域的形式给出，如下例：

```
expect send
```

这里“expect”是指收到的字符串，而“send”则是收到“expect”时发送的字符串。

“expect”域可以由下列格式的子域构成。

```
expect [-subsend-subexpect].
```

这里如果前一个“expect”的内容没能正确读到，则发送“subsend”，而“subsend”之后的“subexpect”是下一个希望收到的字符串。换言之，send-expect 序列如果收到希望收到的内容，则由它发出一个字符串，只有在 10 秒钟之内没有收到希望收到的内容才由 subsend-subexpect 序列发出字符串。

以“login-login”为例。UUCP 程序希望收到“login”，如果收到“login”，则继续处理下一个域。如果没有收到“login”，则仅发出一个回车，接着再次等待“login”。如果不希望从远程计算机收到字符，则应在头一个 expect 域中放“”表示空串。注意，每发出一个 send 域都应跟一个回车，除非 send 域本身以 \ C 结尾。

如果某一个 expect 字符串以虚线“-”打头，则应理解为一个空 expect 域后跟一个 subsend 串。比如，“-login:”表示发出一个回车并等待“login:”输入。

下面介绍如何建立 Login Script

考虑下例中的 Systems 文件入口:

```
terps Any ACU 1200 18005211980 "" \ r login: -BREAK-ogin: \  
uucpx word: ichore
```

连接过程中 login Script 工作情况如下:

- 1.最初不等待接收什么内容。
- 2.发出回车, login Script 域等待接收提示符“ogin:”(即 login)。
- 3.如果没有收到“ogin:”,则发出一次 break 信号。
- 4.当最终收到“ogin:”时,发出注册名 uucpx。
- 5.当收到提示符“word:”时,发出口令“ichore”

注册过程常常需要多次尝试才能成功。有时在出现“login:”提示符之前要发出一次 break 信号或多次 break 信号组(特别是对于变速率的调制解调器更是如此)。如果你不能从系统管理员那里获取必要的信息去登录某一个特定的系统,你也可以自己与该系统进行连接。你可用 cu 程序查明要得到从该系统发出的“login:”提示符自己必须发送什么内容。有些特殊字符在注册过程中发送时可产生特定的动作,这可能在你的 login Script 域中用得到:

- \ N: 发送一个 null 字符
- \ b: 发送或希望收到退格字符
- \ C: 如果在字符串的末尾,则一般情况下应跟在字符串后的回车可省掉
- \ d: 在发送或读取更多的字符之前等待 2 秒钟
- \ p: 暂停大约 1/4 到 1/2 秒
- \ E: 开始响应检查。(从这一点开始,无论何时发出一个字符,系统都必须等到这个字符收到之后才干其他事情)
- \ e: 关掉响应检查
- \ n: 发送或等待接收换行符
- \ r: 发送或等接收回车符
- \ s: 发送或等待接收空格符
- \ t: 发送或等待接收 tab 符
- \ \: 发送或等待接收 \ 符
- EOT: 发送 EOT
- BREAK: 发送一个 BREAK 信号
- \ K: 与 BREAK 相同
- \ ddd: 从八进制数字(ddd)生成单字符
- " ": 希望接收一个空串

### 3.4.6 用 permissions 文件限制访问

如果其他计算机要拨号进入你的系统,则由 Permission 文件(/usr/lib/uucp/Permissions)指出远程计算机在注册、文件访问及命令执行等方面的权限。有的任选项限制远程计算机请求及接收本地计算机文件的能力,有的则说明远程计算机可以在本地计算机上执行的命令的种类。



## 1. 构造 Permissions 文件条目

每个条目是一个逻辑行，如果到物理行尾，逻辑行还未结束，则用“\ ”连到下一个物理行。条目是由以空格相隔的任选项构成。每个任选项都是下列格式的名字-值组成的对：

```
name = value
```

---

注意，在任选项内不允许出现空格。

---

注释行以井号符（#）打头，占一个整行。忽略空行。

共有两种 Permissions 文件条目：

**LOGNAME:** 说明在远程计算机访问你的计算机时起作用的权限值。

**MACHINE:** 指出在你的计算机访问某一远程计算机时起作用的权限值。

## 2. Permissions 文件的限制

在使用 Permissions 文件限制远程计算机访问的级别时：

- 远程计算机在 UUCP 通信中登录注册所用到的所有注册 ID 必须只出现在一个 LOGNAME 条目中。

- 任何名字未出现在 MACHINE 条目中但实际上已被访问的系统，具有以下缺省的权限或限制：

- 只有本地的发送和接收请求可以执行；

- 远程计算机可以将文件发送到你的计算机的 /usr/spool/uucppublic 目录中；

- 远程计算机发送到你的计算机上执行的命令必须是一种缺省命令，通常是 rmail 命令

---

注意：如果某一远程计算机访问你的系统，除非你有那台计算机的唯一的注册号和口号，否则你不会知道访问者是否真是它自己声称的那个系统。

---

下面讲述每一个任选项，说明其用法，并列出它们的缺省值：

### REQUEST

指出远程计算机是否可以要求从你的计算机发送文件。当一个远程计算机访问你的计算机并且要求在此接收文件时，这一请求可以被接受，也可以被拒绝。下面的串表示远程计算机可以要求你的计算机传送文件：

```
REQUEST = Yes
```

下面的字符串表示远程计算机不能要求从你的计算机收到文件：

```
REQUEST = No
```

值“no”是缺省值，在没有注明 REQUEST 选项时即采用该值。REQUEST 任选项可以出现在 LOGNAME 条目中（远程计算机访问你的系统时），也可以出现在 MACHINE 入口中（当你的计算机访问远程计算机时）。

### SENDFILES

指出你的计算机是否可以发出为远程计算机准备的队列中的内容。若某一远程计算机

访问你的系统，在工作完成之时，它也许会试图取走你的计算机为它准备的队列的内容。

下面字符串表示：在某一计算机以 LOGNAME 选项中出现的一个名字登录时，你的计算机可以发送为它准备的内容：

```
SENDEILES = Yes
```

如果你的计算机对该远程计算机来讲是处于被动方式时，则上述字符串内容是强制的。

下面的字符串表示：你的计算机中队列中的文件只有在你的计算机访问远程计算机时才可以发送：

```
SENDFILE = Call
```

值“Call”是 SENDFILE 任选项的缺省值。这个任选项只对 LOGNAME 条目有用，因为 MACHINE 条目只适用于访问远程计算机的情形。当本选项用于 MACHINE 条目时，可以忽略。

**READ 和 WRITE**

指出 uucico 可以读出或写入的文件系统的不同部分。READ 和 WRITE 选项可以在 MACHINE 以及 LOGNAME 条目中。

READ 和 WRITE 选项的缺省值都是 uucppublic 目录：

```
READ = /usr/spool/uucppublic
```

```
WRITE = /usr/spool/uucppublic
```

下面的串的表示可以访问 UUCP 可以读出的任何文件：

```
READ = / WRITE = /
```

这些条目的值应该是彼此以冒号(:) 隔开的一组路径名。Read 任选项用于请求文件，而 WRITE 任选项用于存放文件。

需要指出的是，READ 和 WRITE 任选项的值不影响文件或目录的实际权限值。比如，一个权限值为 700 的目录只允许属主进行访问，不管 UUCP 在 Permissions 文件中定义了什么样的访问权限都不能对这个目录进行读写。

为获取在 /usr/tmp 目录以及公共目录中存放文件的许可权限，WRITE 选项必须使用下面的权限值：

```
WRITE = /usr/spool/uucppublic : /usr/tmp
```

应该说明，如果使用了 READ 和 WRITE 任选项，则所有的路径名都必须加以说明，因为在缺省表中不会增加未经说明的路径名。比如，如果在 WRITE 任选项中只说明了路径 /usr/news，那么就不允许在公共目录中存放文件。

对允许远程计算机读出和写入你的系统中哪一个目录这个问题必须小心。比如，你大概不希望远程计算机对你的 /etc/passwd 文件进行写操作，所以 /etc 目录就不应该允许远程计算机执行写操作。

**NOREAD 和 NOWRITE**

指出读选项和写选项或者缺省值的例外情况。下面的说明允许对 /etc 目录（以及它的子目录）中的文件之外的任何文件进行读操作，只允许写入缺省的 /usr/spool/uucppublic 目录：

```
READ = /
```

WRITE = /usr/spool/uucppublic

NOREAD = /etc

NOWRITE = /etc

NOWRITE 与 NOREAD 选项的工作方式完全一样。NOREAD 和 NOWRITE 选项都可以用于 LOGNAME 和 MACHINE 条目中。

### CALLBACK

在 LOGNAME 条目中指出，如果一个系统访问你的计算机，则必须要在回访该系统成功之后才开始实际动作。从安全的角度来讲，你可以通过这样做来确认访问你系统的计算机就是它自己声称的那一台，以防假冒。

下面的串表示在开始实际的文件传输之前你的计算机必须回访远程计算机：

CALLBACK = Yes

CALLBACK 选项的缺省值为：

CALLBACK = NO

本任选项极少使用。如果两个系统都为对方设置了这一选项，则在这两个系统就无法建立起对话。

### COMMANDS

在 MACHINE 条目中指示远程计算机可以在你的计算机上执行的各命令。由于这会影你系统的安全性，所以在使用时必须特别小心。

uux 程序产生远程执行请求，把请求放入队列中再传送到远程计算机上。文件和命令被送到目标机进行远程执行。注意 COMMANDS 选项不能用于 LOGNAME 条目中，MACHINE 条目中的 COMMANDS 选项定义的是你访问远程系统或者远程系统访问你的命令权限。

远程计算机可以在你的计算机上执行的缺省命令为：

COMMANDS = rmail

如果在 MACHINE 条目中使用了一串命令，则这一串命令对 MACHINE 条目中的各系统都有效。比如，下面一个串的含义是：计算机 owl, raven, hawk 和 dove 可以在你的计算机上执行 rmail, rnews 以及 lp 命令：

MACHINE = owl : raven : hawk : dove \

COMMANDS = rmail : rnews : lp

也可以使用命令的完整路径名。比如，下面的命令表示命令 rmail 使用缺省路径：

COMMANDS = rmail : /usr/lbin/rnews : /usr/local/cp

你的计算机的缺省路径是 /bin, /usr/bin 和 /usr/lbin。当远程计算机指明要执行的命令是 /usr/lbin/rnews 时，就执行 /usr/lbin/rnews，而不管缺省路径是什么。同样，/usr/local/cp 才是要执行的命令。

如果在列表中加入 All 值，就意味着远程计算机在条目中说明的所有命令都可以执行，这即是将你的计算机完全开放于远程计算机，所以必须谨慎使用 ALL 值，它给远程计算机的权限大大超过了一般用户。

下面的串说明两点：

COMMANDS = /usr/local/bin/lc : ALL : /usr/local/Lp

1. ALL 值可以出现于串中的任何位置；如果远程计算机发出的请求命令没有包含 lc 或 Lp 的完整路径名，则按串中给出的路径名执行（而不用缺省路径名）。

2. 任何时候当 COMMANDS 选项开放了像 cat 和 uucp 这样有潜在危险的命令时都应同时使用 VALLDATE 选项。任何读写文件的命令被 UUCP 远程执行控制程序 (uuxqt) 执行都对当地系统的安全有潜在危险

#### VALIDATE

在 LOGNAME 条目中的 COMMANDS 选项指定了对你计算机的安全有潜在危险的命令时应同时使用 VALEDATE 选项。它提供了对访问者身份的一种验证手段。该任选项的使用要求特权计算机对 UUCP 操作有唯一的注册号或口令，其特性是：与本条目相应的注册号或口令得到保护。如果未经授权者得到了该信息，则不再认为 VALIDATE 选项是安全的。VALIDATE 只是增加 COMMANDS 选项安全性的一种手段，但它在开放命令访问权限方面比用 ALL 值更加安全。

在将 UUCP 操作的特权注册号和口令提供给远程计算机时必须格外小心。将具有文件访问和远程命令执行能力的特殊的注册号和口令提供给远程计算机相当于给那台远程计算机上的用户以你计算机上的普通注册号和口令。所以，如果你不信任远程计算机上的某些用户，就不要把特权注册号以及口令提供给该计算机。

下面的 LOGNAME 条目指出，如果某一个自称是 eagle, owl 或 hawk 的远程计算机试图登录你的计算机，则它必须使用注册号 uucpfriend:

```
LOGNAME = uucpfriend VALIDATE = eagle : owl : hawk
```

可以想像，如果有未经授权者获取了 uucpfriend 注册号或者口令，那么这一措施就无效了。

但这与只出现于 MACHINE 条目中的 COMMANDS 任选项有何关系呢？它用与一个特权注册号相应的 LOGNAME 条目将 MACHINE 条目与 COMMANDS 选项联系起来。因为在远程机器登录之后没有运行执行守护程序，所以这一联系是必要的。因此，问题实际上是在于你的系统如何得知执行文件是从哪里来的。

每个远程计算机在你的计算机上都有自己的 spool 目录，只有 UUCP 程序才能在这些 spool 目录中执行写操作。来自远程计算机的执行文件在传送到你的计算机之后就存放于自己的 spool 目录之中。当 uuxqt 控制程序运行时，就使用 spool 目录名在 Permissions 文件中找到 MACHINE 条目，并得到 COMMANDS 表。如果计算机名字不出现在 Permissions 文件之中就使用缺省表。

下面的例子表明了 MACHINE 条目和 LOGNAME 条目之间的联系：

```
MACHINE = eagle : owl : hawk REQUEST = Yes\  
COMMANDS = rmail : /usr/local/bin/lc\  
READ = / WRITE = /  
LOGNAME = uucpz VALIDATE = eagle : owl : hawk\  
REQUEST = yes SENDFILES = yes\  
READ = / WRITE = /
```

例中 COMMANDS 选项行表明了远程邮件和 /usr/local/bin/lc 可以由远程用户执行。

在 MACHINE 条目中，你必须假设在你要访问该条目中列出的某一台计算机时实际上你访问的是 eagle, owl 或者 hawk。在其中之一 spool 目录中的任何文件都是由其中一台计算机送进去的。如果某一远程计算机机登录于你的计算机并自称是上列三者之一，且其执行文件确实存放在相应的 spool 目录中，这时你应确认计算机确实有特权注册号 uucpz。

将 MACHINE 条目和 LOGNAME 条目合二为一。

在 MACHINE 条目和 LOGNAME 条目的相同选项值一致时，有可能将此二者合二为一。比如，下列这两个条目的 REQUEST, READ, WRITE 选项的值完全一致：

```
MACHINE = eagle : owl : hawk REQUEST = yes \  
READ = / WRITE = /  
LOGNAME = uucpz REQUEST = yes SENDFILES = yes \  
READ = / WRITE = /
```

这两者可以合并如下：

```
MACHINE = eagle : owl : hawk REQUEST = yes \  
LOGNAME = uucpz SENDFILES = yes \  
READ = / WRITE = /
```

### 3.4.7 对设备文件增加拨号输出条目

Devices 文件（/usr/lib/uucp/Devices）包含所有可以用于建立到远程计算机的连接的设备信息。设备有自动寻呼部件 ACU，直接连接和网络连接几种。Devices 文件与 Dialers 文件，Systems 文件，Dialcodes 文件紧密相关，在对其中一个文件作修改之前，必须非常熟悉所有这些文件。对其中一个文件的某一条目的修改也许要求对另一个文件中的相关条目作相应的修改。

Devices 文件中的每一条目都应具有如下格式：

```
type,ttyline,dialerline,speed,dialer-token
```

这里：

- type: 包含两个关键词（direct或Accl）中的一个及局域网开关的名字，或系统名称。
- ttyline: 包含与该Devices条目相关的端口设备名称，比如，如果某一特定条目的自动拨号调制解调器加到 /dev/tty1A 端口，则该域中填入的名称应为：tty1A。
- dialerline: 只对本身不包含调制解调器，因而必须使用附加端口的801类型拨号系统有用。除非你用的就是 801 类型拨号系统，否则在这个位置上填入连字号“—”即可。
- speed: 是设备的速度或速度范围，也可以包含用以区分不同的拨号级别的标志。
- dialer-token: 即拨号者-标志对，每一对代表一个拨号者和传送给它的参数。dialer部分可以是自动拨号调制解调器的名字，也可以是 Direct 表示直接连接设备。

## TYPE 域

本域包含两个关键字 (Direct 或 ACU) 之一, 以及局域网络开关名, 或者系统名称。其中:

**Direct:** 指示到另一台计算机的直接连接, 或者用于 cu 连接的开关。

**ACU:** 表示到远程计算机的连接是通过自动寻呼部件 ACU 实现的。实现方式可以是直接连接到你的计算机上, 也可以通过局域网络 (LAN) 开关间接连接到你的计算机上。

**LANswitch:** 本位置也可以用局域网络 LAN 开关的名字来替代。

**Sysname:** 表示到某一特定计算机的直接连接。(sysname 用计算机名字来代替)。这意味着与本 Devices 条目相对应的端口是用于 Systems 文件中的某一特定计算机的。

比如, 在 Devices 文件的 TYPE 域中使用的关键字 gorgon 是与 Systems 文件入口的第三个域相匹配的。

```
Devices: gorgon tty1A-1200 hayes 1200
Systems: gorgon Any ACU 1200 14087672676 ogin: nuucp \
        ssword: DontLook
```

## Speed 域:

在大多数情况下, 如果在 TYPE 域中使用了关键词 ACU 或者 Direct, 则本域只是设备的速度。当然, speed 域可以包含一个字母和一个速度值 (比如 C1200, D1200) 以区分不同类别的拨号系统。这么规定的原因是许多比较大的办公室中有不止一种类型的电话网络。一种只用于服务内部通信, 另一种用于处理外部通信。这时就有必要区分一下哪一条线路用于内部通信, 哪一条用于外部通信。Devices 文件的 speed 域中所用关键字与 Systems 文件入口的第四个域相匹配。比如:

```
Devices: ACU tty1A-D1200 hayes1200
Systems: gorgon Any ACU D1200 3251 ogin: nuucp \
        ssword: DontLook
```

有些设备可以在任何速度下工作, 所以在 speed 域中可以用关键字“ANY”。如果使用了“ANY”作关键字, 则端口可以匹配 Systems 文件条目中要求的任何速度。若本域为“ANY”而 Systems 文件 speed 域也是“ANY”, 则速度为缺省值 1200bps。如果某一设备可以使用于一个速度范围内, 则本域应说明这一范围 (比如, 1200-9600 或者 D1200-9600)。人们往往愿意使用这种方式而不太愿意用“ANY”。

## Dialer-Token 域

本域具有以下格式:

```
dialer [token dialer token..]
```

对一个直接相连的线路来讲, 本域只包含关键词 Direct, 不需要任何标志。

对到一个拨号系统的简单连接而言, 本域包含拨号系统的名字, 标志被省略, 缺省值取自 Systems 文件条目的电话号码 phone number 域。

对拨号系统或者网络数据开关而言, 本域包含 Dialers 文件中的条目名称。(develcon 和 micom 是网络数据开关的例子)。

表 3.3 给出可用作 Dialers 条目的几种类型:

表 3.3 Dialer 类型

Dialer Type	Modem or Data Switch
Direct	direct line;no dialer
Penril	Penril modem
Hayes	Hayes modem(or compatible)
Ventel	Ventel 212+modem
Vadic	Racal Vadic 3451 modem
LANswitch	network switch described in type field
Hayes1200	Hayes Smartmodem 1200
Hayes2400	Hayes Smartmodem 2400
Develcon	Develcon network dataswitch
Micom	Micom network dataswitch
Rixon	Rixon Intelligent Modem
ATT4000	AT&T Programmable 300 / 1200 Modem Model 4000
ATT2212c	AT&T DATAPHONE II 2212C Modem
ATT2224	AT&T DTAPHONE II 2224 Modem
NLS	Network Listener Service

dialer-token 条目可用四种不同的方法加以构造。选用哪一种方法决定于与本条目对应的设备类型。

• 简单的调制解调器连接

如果自动拨号调制解调器直接连接到你计算机的端口上，则相应 Devices 文件条目的 dialer-token 域就只有一对，一般来讲是调制解调器的名字。该名字将特定的 Devices 文件条目与 Dialers 文件中的条目匹配起来。因此，dialer 域必须与下面 Dialers 文件条目的第一个域相匹配。

```
Devices:   ACU tty1A-1200 ventel
Dialers:   Ventel=&-%" "\ r\ p\ r\ c$ <k\ T%%\ r>\ c ONLINE
```

注意：只有 dialer 部分在 Devices 文件条目的 dialer-token 域中出现。这意味着要传送给拨号者的标志（在这种情况下是电话号码）是取自 Systems 文件条目的 phone 域。

• 直接连接

如果建立到某一特定计算机的直接连接，则相应条目的 dialer-token 域应包含关键字 Direct，对两种类型的直接连接条目，即 direct 和 sysname 都是如此。（可参考 type 域）

• 局域网络开关

如果你希望与之通信的计算机与你的计算机处于同一局域网络开关上，则你的计算机必须首先访问网络开关，再由该开关建立到另一计算机的连接。在这种类型的条目中只有一个对。dialer 部分与 Dialers 文件条目相匹配。如下面例子所示：

```
Devices:   develcon tty13-1200 develcon\ D
Dialers:   develcon" "" "\ pr\ ps\ c est: \ 007 \ E\ D\ e \ 007
```

可以看出，标志部分是 \ D，表示未经转化直接取自 Systems 文件。Systems 文件中对应本计算机的条目在 phone 域中含有这个标志。这一般用来保存计算机的电话号码

(参见 Systems 文件的 phone 域)。\**D** 标志保证 phone 域的内容不在 Dialcodes 文件中被解释为正当条目。

- 与局域网络开关一起使用的调制解调器

如果一个直接拨号调制解调器被连接到一个开关，那么你的计算机必须首先访问开关，再由开关建立到自动拨号调制解调器的连接。这种类型的条目需要两个 dialer-token 对。每一对的 dialer 部分都用于匹配 Dialers 文件中的条目，如下所示：(条目的第五和第七个域)

```
Devices:   ACU ttyl4-1200 develcon vent ventle
Dialers:   develcon" " " "\ pr\ ps\ c est: \ 007\ E\ D\ e 007
           Ventle=&-%" "\ r\ p\ r\ c\ $ <k\ T%%\ r>\ c
           ONLINE
```

在头一对中，**develcon** 是开关，**vent** 是传递给 **develcon** 开关并告诉它将哪一个设备连接到你的计算机上的标志。这种标志对每个局域网 LAN 开关都是唯一的，因为每个开关的建立方法都不同。一旦调制解调器 **ventel** 连接上了，就访问第二个对，其中 **ventel** 是拨号者，标志取自 Systems 文件。

下面是两个可能出现在 dialer-token 域中的组合字符：

\**T**: 指示 phone 域应该在这一级用 Dialcodes 文件进行转化。

\**D**: 指示 phone 域不应用 Dialcodes 文件进行转化。

### 3.5 特殊的 UUCP 配置选项

本节包含几个用于特殊环境的选项。这些选项在大多数情况下都可以不加考虑。

#### 3.5.1 对拨号系统文件增加拨号者 (Dialers)

Dialers 文件 (/usr/lib/uucp/Dialers) 指示在用端口传输数据之前必须先要在端口上进行初始对话。如如果你的调制解调器没有在 Dialers 文件中出现，则可以对文件增加新的条目。

#### 3.5.2 改变 uucico 包参数

有一项新增特性允许改变包含于 uucico 程序中的两个特定参数而不必重新对源文件进行编译。这两个参数是：

**Windows**: 指示移动窗口协议 (sliding-window protocol) 将要用到的窗口大小。

**pktime**: 是 uucico 在决定放弃本次包传送而重新启动一次包传送之前必须等待的秒数。

#### 3.5.3 防止未经授权的系统登录

如果某一未经授权的系统企图拨号进入你的系统，则执行 remote.unknown 文件，它记录下登录企图并拒绝建立连接。如果你允许这样的系统登录到你的系统中，你可以改变 remote.unknown 文件的权限，使之不能执行。这样你的系统就可以接受所有的通信请求。要做到这一点，应作为超级用户注册并输入以下命令：



```
cd /usr/lib/uucp
chmod 000 remote.unknown
```

### 3.5.4 连接 Micnet 系统与 UUCP 网络

要在你的 Micnet 系统上使用 UUCP 系统，应按以下步骤进行：

1. 在你的 UUCP 安装的计算机的 `maliases` 文件中增加下面的条目：

UUCP:

2. 对你系统中的其他所有计算机的 `maliases` 文件增加下面的条目：

UUCP: machine-name:

这里 `machine-name` 是安装 UUCP 系统的计算机的名字。这种格式的条目也可用在安装 UUCP 系统的计算机上。

你可以通过另一个系统给你自己发一短信，以此来测试 UUCP 系统。比如，如果你处于系统 `chicago`，而在系统中还有一个名为 `seattle` 的 Micnet 系统，则按下面命令执行可以发送邮件给系统 `seattle`，然后将邮件送回至你自己的 `chicago` 系统，最后再送到你的 Micnet 网络中的用户 `johnd` 处：

```
mail seattle! chicago! johnd
```

值得一提的是，UUCP 系统通常是按照固定好的调度方案来执行通信任务，所以送出的邮件也许不会立即送回。

## 3.6 管理 UUCP 系统

本节讨论管理和维护 UUCP 系统用到的各种 shell 程序。另外本节还包括对 `/usr/spool/uucp` 工作目录的讨论。

### 3.6.1 UUCP 维护 shell 程序

有几种作为守护程序运行以管理系统操作的 shell 程序：

- 多长时间检查一次 UUCP 目录是否工作正常（由 `uudemon.hour` 程序控制）。
- 对被动系统或者说不产生访问其他计算机的请求的系统进行登记。（由 `uudemon-poll (2)` 完成）
- 将状态信息传递给 UUCP 管理员。（由 `uudemon.admin` 执行）
- 对 UUCPspool 目录进行清理（由 `uudemon.clean` 完成）。

以上的控制程序都必须由系统管理员来建立。

### 3.6.2 产生 UUCP 使用登记报告: `uulog`

`uulog` 程序显示远程机器使用 UUCP 的记录信息。程序 UUCP, `uuto`, `uux` 的每次使用都记录在一个特殊的记录文件中。每台机器有这么一个记录文件。

`uulog` 命令有如下的任选项：

`-fsystem`：显示系统文件传输记录的最近一个或多个记录项。

`-ssystem`：显示系统文件传输信息。

`-x`：显示指定系统的 `uuxqt` 记录文件。

-unmber: 指示要求-f选项显示的记录条数。

比如, 要打印出系统chicago的最近10个文件传输记录项, 你应当输入:

```
uulog -fchicago -10
```

在uulog程序的执行过程中, 要检查下列目录中的文件:

```
/usr/spool/uucp/.Log/uucico/*: UUCP程序所用目录
```

```
/usr/spool/uucp/.Log/uuxqt/*: UUXQT程序所用目录
```

### 3.6.3 UUCP的spool目录

下面对UUCP spool目录中的所有文件和子目录加以讨论。这些文件在spool目录中创建, 用于锁定设备, 暂时保持数据, 或者保存远程传输或执行的信息。

#### TM. (临时数据文件)

这些数据文件是从另一台计算机收到一个文件时由UUCP进程在spool目录(即/usr/spool/uucp/system)下创建的。“system”目录与发送当前这个文件的远程计算机的目录名相同。这些临时数据文件的名字有如下形式:

```
TM. Pid. ddd
```

其中: Pid是进程标识符, ddd是从0开始的3位序数。

当整个文件收到之后, TM.Pid.ddd文件就移到由C.Sysnxxxx文件(下面将要讨论到)指示的路径名中。如果此时进程被非正常终止, TM.Pid.ddd文件会保留在“System”目录中。这些文件应该由uuclean自动删去。

#### LCK. (锁文件)

每个正在使用的设备在/usr/spool/uucp目录中都建立有一个锁文件。锁文件可以防止对同一设备在同一时间内的多次使用和请求。锁文件的名字具有以下形式:

```
LCK..str
```

其中str是一台设备或者一台计算机的名字。如果通信连接非正常断开(通常是由于计算机遭到破坏), 则这些文件保留在spool目录中。当父进程不再是当前进程之后锁文件被忽略(删去)。锁文件包含创立锁的进程标识符(ID)。锁文件名字通常以“a”结尾(表示非调制解调器控制), 这样做在同一端口同时以调制解调器控制和非调制解调器控制加以指定时可以避免冲突。比如, /dev/tty1A上的锁的名字为LCK..tty1a。

#### C. (工作文件)

工作文件是在远程计算机对作业(文件传输或者远程命令执行)进行排队时在spool目录中创建的。工作文件的名字形式如下:

```
C.sysnxxxx
```

其中sys是远程计算机的名字, n是代表作业优先级级别数的ASCII字符, xxxx是由UUCP指定的4位的作业顺序号。工作文件包含以下信息:

- 将要发送或请求的文件的完整路径名
- 目的地或者用户或者文件名的完整路径名
- 用户登录注册名
- 任选项表
- spool目录中相应的数据文件名。如果指定了UUCP-c任选项或者uuto -p任选

项, 则应使用 dummy name。

- 源文件的方式位
- 在传输完成之后要通知的远程用户的注册名。

#### D. (数据文件)

数据文件是在命令行中指定将源文件拷贝到 spool 目录时创建的。数据文件的名字应有如下形式:

D.systmxxxxyyy

其中 systm 是远程计算机名字的头五个字符, xxxx 是由 UUCP 指定的 4 位的作业顺序号, 这 4 位作业顺序号可能后跟一个数 yyy, 这是在有几个为工作文件 (C.) 建立的 D. 文件时使用的。

#### X. (执行文件)

执行文件是远程命令执行时在 spool 目录中创建的。执行文件名应有以下形式:

x.sysnxxxx

其中 sys 是远程计算机名字, n 是表示工作的优先级级别的 ASCII 字符, xxxx 是由 UUCP 指定的 4 位顺序号。执行文件包含以下信息:

- 请求者的注册号和计算机名字
- 要求执行的文件名字
- 对命令串的标准输入内容
- 接收命令执行标准输入的计算机和文件名。
- 命令串
- 用以返回状态请求的任选端口

### 3.7 完整的 UUCP 举例

本节将给出两个完整的 UUCP 系统及其数据库文件的例子。

#### 例 1. gomer 系统

gomer 系统具有:

- 在 tty4B 上有波特率为 1200 的调制解调器
- 在 tty4D 端口上与系统 poker 直接连接, 只允许向外访问:
- 三个合法的 UUCP 注册号:

nuucp: 用于 email 的公用注册号, 不需要口令

ubarn: 为系统 Poker 准备的注册号

upay4: 用于 email 和文件传输的私用注册号

所有以 '#' 号打头的行都是注释行, 并非必需的。大多数例子列出的都是部分清单, 也许还应包含其他条目。没有安装 Micnet。调制解调器首先以 1200 波特应答, 为输入/输出而建立。

---

注意: 这里包括了 /etc/passwd 文件中的内容。绝对不要用文本编辑程序来编辑 /etc/passwd 文件, 这样做会带来严重的后果。应使用 sysadmsh (ADM) 的 Accounts→User→Creat 或者 Accounts→User→Modify 选择来建立或变更 UUCP 注册帐号。

---

```

/etc/passwd
UUCP:*:5:5:Uucp • admin:/usr/lib/uucp:
nuucp::201:5:pubic:/usr/spool/uucplogins/nuucp:/usr/lib/uucp/uucico
upay4:*:202:5:private:/usr/spool/uucppublic:/usr/lib/uucp/uucico
ubarn:*:203:5:poker:/usr/spool/uucppublic:/usr/lib/uucp/uucico

/etc/group
uucp:x:5:uucp,nuucp,ubarn,upay4

/etc/systemid
gomer
gomer

/etc/inittab
Se4B:23:off:/usr/lib/uucp/uugetty -t60 tty4B 1200
Se4b:23:off:/etc/getty tty4b m
Se4D:23:off:/usr/lib/uucp/uugetty -t60 tty4D 1200
Se4d:23:off:/etc/getty tty4d m

/usr/lib/uucp/Devices
#300-1200 baud hayes 1200 baud modem
#Direct tty4b 用于使用 cu 进行向外访问之用
ACU tty4B -300-1200 dial HA12
Direct tty4b -300-1200 dialHA12
Poker tty4d -9600 direct

/usr/lib/uucp/Permissions
#Public uucp 注册号只用于 mail, 能够发送邮件
#传输文件至 uucppubic, 从 uucppubic 取文件
LOGNAME = nuucp MACHINE = OTHER \
COMMANDS = rmail:ls:uucp \
READ = /usr/spool/uucppublic:/usr/tmp \
WRITE = /usr/spool/uucppublic:/usr/tmp \
SENDFILES = yes REQUEST = yes
#私有的 uucp 注册号用于邮件和文件传输
#只有 digbat, ogre, grinch, ...可以使用本注册号
LOGNAME = upay4 VALIDATE = digbat:ogre:grinch:gomer:blitzen \
COMMANDS = rmail:ls:uucp:who:uux \
READ = / WRITE = / \
NOREAD = /etc \
SENDFILES = y REQUEST = yes
#只有 gomer 才能使用本注册帐号
LOGNAME = ubarn VALIDATE = gomer \
COMMANDS = ALL \
READ = / WRITE = / \
SENDFILES = yes REQUEST = yes

```

```

/ usr / lib / uucp / Systems
  #本地访问
    digbat Any Acll 1200 4444444 ogin:-BREAK-ogin:-BREAK-ogin:\
      uubig word:wetro
  #远距离（只在夜里才允许访问）
  grinch Any1800-0700 ACU 2400 18888888 " " \ r oginL-BREAK-ogin:\
    -BREAK-ogin:nuucp
  uunet Any1800-0700 ACU 2400 17031111111 ogin:-BREAK-ogin:\
    -BREAK-ogin:xytpq sword:grm5q
  #只对以 nuucp 访问输入（取邮件）但不访问输出的系统
    daboss Never
    sales Never
    guru2 Never

```

## 例 2. 系统 digbat

系统 digbat 具有：

- 在端口 tty1A 上有 2400 波特的调制解调器。
- 两个合法的 uucp 注册帐号：  
 nuucp: 用于email的公用注册帐号，不需要口令  
 uubig: 用于email和文件传输的私用注册号

以#号打头的行为注释行，并非必需。大多数例子都是列出部分清单，还可能包含其他条目。没有安装 Micnet。调制解调器以 2400 波特应答，为输入 / 输出访问而建立。

```

/ etc / passwd
  uucp:*:5:5:Uucpadmin: / usr / lib / uucp:
  nuucp:*:201:5:pubic: / usr / spool / uucplogins / nuucp: / usr / lib / uucp / uucico
  uubig:*:202:5:private: / usr / spool / uucppubic: / usr / lib / uucp / uucico
/ etc / group
  uucp:X:5:uucp,nuucp,uubig
/ etc / systemid
  digbat
  digbat
/ etc / inittab
  Se1A:2:off: / usr / lib / uucp / uugetty --t60 tty1A 2400
  Se1a:2:off: / etc / getty tty1a m

```

```

/ usr / lib / uucp / Devices
  # 300-2400baud hayes 2400 baud modem
  # Direct entry 用于使用 cu 时
  ACU tty1A --300-2400 dialHA24
  Direct tty1A --300-2400 dialHA24

```

```

/ usr / lib / uucp / Permissions

```

#公用的 uucp 注册号只用于邮件

#可以发送邮件，传输文件至 uucppublic，从 uucppublic 取文件，获取目录列表

(LS)

```
LGONAME = nuucp  MACHINE = OTHER  \  
COMMANDS = rmail:ls:uucp  \  
READ = /usr/spool/uucppublic:/usr/tmp  \  
WRITE = /usr/spool/uucppublic:/usr/tmp  \  
SENDFILES = yes  REQUEST = yes
```

#私用的 uucp 注册号用于邮件和文件传输

#只有 ogre, grinch, ... 可以使用本注册号

```
LOGNAME = uubig  VALIDATE = ogre:grinch:gomer:blitzen  \  
COMMANDS = rmail:ls:uucp:who:uux  \  
READ = /  WRITE = /  \  
NOREAD = /etc  \  
SENDFILES = yes  REQUEST = yes
```

/usr/lib/uucp/Systems

#本地访问

```
gomer Any ACU 1200 3333333 ogin:-BREAK-ogin:-BREAK ogin:\  
upay4 word:dryrot
```

#远距离（只在夜间访问）

```
grinch Any 1800-0700 ACU 1200 18888888 " " \ r ogin:  \  
-BREAK-ogin:-BREAK-ogin:nuucp
```

#对以 nuucp 输入访问（取邮件）但不输出访问的系统

```
daboss Never  
damgr Never  
guru2 Never
```

### 3.8 UUCP 的错误信息

本节列出与 UUCP 有关的错误信息清单。错误信息有两种类型。ASSERT 型的错误记录在文件 /usr/spool/uucp/Admin/errors 中，STATUS 型错误记录在 /usr/spool/uucp/Status 目录中的单个机器文件中。

#### 3.8.1 ASSERT 型错误信息

在一个进程失败时，ASSERT 型错误信息被记录在 /usr/spool/uucp/Admin/errors 文件里。这些信息包括文件名、SCCS 标识符、端口号以及正文信息。在大多数情况下，错误起因于文件系统故障。如果信息中给出了错误号，则以此为线索分析排除故障。如果错误号存在于信息中，则在下面清单中的( )中可见。

CAN'T OPEN: 表示 open( )或 fopen( )操作失败。应检查文件存在与否以及文件的权限是否正确。

CAN'T WRITE: write( ), fwrite( ), fprintf( )以及此类型的其他操作失败。应检查文件是否存在并检查文件的权限。

CAN'T READ: read( ), fgets( )等操作失败。检查文件是否存在并检查文件的权限。

CAN'T CREAT: creat( )调用失败。应检查文件权限。

CAN'T ALLOCATE: 动态分配失败。

CAN'T LOCK: 建立锁文件 LCK 的操作失败。在某些情况下, 这是一个致命的错误。

CAN'T STAT: stat( )调用失败。检查文件是否存在并检查文件的权限。

CAN'T CHMOD: chmod( )调用失败。应检查文件是否存在并检查文件的权限。

CAN'T LINK: link( )调用失败。应检查文件的存在与否并检查文件权限。

CAN'T CHDIR: chdir( )调用失败。应检查文件的存在与否并检查文件的权限。

CAN'T UNLINK: unlink( )调用失败。

WRONG ROLE: 这表示出现内部逻辑错误。

CAN'T MOVE TO CORRUPTDIR: 企图把一些坏了的 C 或者 X 文件移进 /usr/spool /uucp/Corrupt 目录的操作失败。该目录可能已丢失或者方式或属主不正确。

CAN'T CLOSE: close( )或 fclose( )操作失败。

FILE EXISTS: 企图建立某一 C 或 D 文件, 但该文件已经存在。常在执行顺序文件访问出现这种错误。通常由软件错误引起。

NO UUCP server: 企图执行 TCP/IP 调用, 但 UUCP 却没有服务器。

BAD UID: 在 /etc/passwd 文件中找不到 uid, 通常是因为文件系统出了问题或者 /etc/passwd 文件出现不一致性。

BAD LOGIN UID: 与前一条信息一样。

ULIMIT TOO SMALL: 当前用户进程的 ulimit 太小。文件传输可能失败, 所以未作传输尝试。

BAK LINE: 在 Devices 文件中有一个坏端口, 或者一个或多个端口没有足够的参数。

FSTAT FAILED IN EWRDATA: 以太网介质出了问题。

SYSLST ONERFLOW: gename.c 中的内部表溢出。

TOO MANY SAVED C FILES: 与前一条信息一样。

RETURN FROM fixline ioctl: ioctl 失败。系统驱动程序出现问题。

BAD SPEFD: 在 Devices 或者 Systems 文件中出现不正确的端口速度。

PERMISSION FILE: BAD OPTION: 在 Permissions 文件中端口坏或选项不正确。

PKCGET READ: 也许是远程计算机挂起, 可以不采取任何措施。

PKXSTART: 远程机器以不可恢复方式退出。通常可以忽略。

SYSTAT OPEN FAIL: /usr/lib/uucp/Status 的方式有错或者在目录中有一个文件方式错。

TOO NANY LOCKS: 有内部故障

XMV ERROR: 某些文件或目录错。很可能是 spool 目录。

CAN'T FORK: fork 与 exec 操作失败。当前作业不会丢失, 留待稍后再试 (uuxqt)。可以不采取任何措施。

### 3.8.2 UUCP 的 STATUS 错误信息

status 错误信息是存放在 /usr/spool/uucp/.status 目录中的信息。目录中对所有你的系统试图与之通信的远程系统都有一个独立的文件与之对应。这些对应单个机器的文件包含进行通信的状态信息，如通信是否成功等。下面是在这些文件中最常出现的错误信息清单：

OK：一切正常。

NO DEVICES AVAILABLE：当前没有可用于访问的设备。应检查在 Devices 文件中对应于特定系统是否有正常的设备，以及在 Systems 文件中是否有可用来访问系统的设备。

WRONG TIME TO CALL：对系统的访问是在 systems 文件规定的时间之外进行的。

LOGIN FAILED：对指定系统的注册失败。原因可能是注册号或者口令错等等。

CONVERSATION FAILED：在成功地启动之后会话失败。这通常意味着某一方出了故障，程序非正常中止或者连接脱开等。

DIAL FAILED：远程机器没有响应。这可能因为拨号系统坏或电话号码不对。

BAD LOGIN / MACHINE COMBINATION：有系统以与 Permissions 文件中的规定不一致的注册帐号或机器名来访问你的系系。

DEVLCE LOCKED：要用到的访问设备当前处于锁定状态，正为其他进程所用。

ASSERT ERROR：出现了 ASSERT 型错误。应检查 /usr/spool/uucp/Admin/errors 文件以找到错误信息。参考上一部分的内容。

SYSETM NOT IN systems：当前系统不在 Systems 文件中。

CAN'T ACCESS DEVICE：试图访问的设备不存在或者方式不对。应检查 Systems 和 Devices 文件中相应的条目。

DEVICE FAILED：打开设备失败。

WRONG MACHINE NAME：被访问机器报告的机器名与访问机器的名字不一样。

CALLBACK REQUIRED：被访问机器要求访问你的系统。

REMOTE HAS A LCK FILE FOR ME：远程系统有一个 LCK 文件给你的系统。它们可能正在试图访问你的机器。如果它们的 UUCP 版本较老，则正与你会话的进程可能已经失败，留下了 LCK 文件；如果它们的 UUCP 版本较新，且并没有和你的系统会话，则拥有 LCK 文件的进程被挂起。

REMOTE DOES NOT DNOW ME：远程机器的 Systems 文件中没有你系统的名字。

REMOTE REJECTAFTER LOGIN：你的系统登录远程系统所用的注册号与该系统期待的不一样的。

REMOTE REJECT,UNDNOW MESSAGE：远程机器拒绝与你的机器通信，原因不知。也许是因为远程机器运行的不是标准版本的 UUCP。

STARTU FAILED：注册成功，但初次握手即失败。应检查通信参数：数据字长，奇偶校验位，停止位等等。

CALL SCRIPT FAILED：通常与 DIAL FAILED 同义。



## 第四章 建立电子邮件

### 4.1 引言

UNIX 操作系统下的电子邮件是由 MMDF 处理的。MMDF 的全称为 Muttichannel Memorandum Distribution Facility (多通道消息传送设施)。

MMDF 是 mail (C) 等邮件程序的支撑程序, 它为在计算机网络上通过不同的邮件传输协议传送信息提供了一种有效的机制。这些网络可以通过局域网与你的机器相连, 网络形式包括 Micnet, UUCP, 以及被称为“internet”的网络互连组。(Internet 又称为 ArpaNet 或 DARPA internet; 其中 D 表示防务 (defense), Arpa 或 ARPA 表示高级研究项目协会)。

MMDF 在用户可以通过不同的协议对不同的网络进行透明访问的同时, 为系统管理人员提供了监视和调整 MMDF 执行的工具。即使在 MMDF 运行时也可以对其进行调整。

工作于 MMDF 的许多命令和系统调用均可独立地工作, 它们全都是操作系统的一部分。在系统安装时, 对 MMDF 进行自动配置, 可用于本地 (一个系统) 邮件。如果没有将整个操作系统都安装上, 则 MMDF 可以作 MAIL 软件包的一个部分用 custom 实用程序进行安装。

SCOMMDF 通道接口与标准 MMDF Iib 通道接口相兼容, 本章讲述的 MMDF 取自 BSD (Berkeley 软件卷) 版本 4.3 的磁带。唯一的例外是由 dbmbuild (ADM) 建立的哈希表 (hashed table) 数据库, 它是非标准的。

---

注: 应保证 mmdf 注册号有一个口令。因为在安装时没有给该帐号指定口令, 所以应用 sysadmsh (ADM) 来安排一个口令。在建立 MMDF 文件或对 MMDF 文件进行修改时应首先在 mmdf 帐号注册。

---

#### 4.1.1 支持的网络类型

MMDF 支持对以下网络类型的访问:

- Micnet
- UUCP
- 任何其它类型网络, 只要有附加软件包对其提供通道支持。

#### 4.1.2 章节安排

本章讲述以下内容:

- MMDF 是如何工作的
- 配置 MMDF
- 维护 MMDF 系统
- 改变机器名

• 关于 Internet 地址

其中“MMDF 是如何工作的”一节讲述了对理解 MMDF 的工作过程非常关键的概念，之后的各节则据此认为读者已经熟悉了这些概念。“配置 MMDF”一节列举了在建立 MMDF 时就遵循的一系列步骤。这些步骤看起来十分复杂，但大多数步骤只需要加以考虑而不必真的对文件进行编辑及修改。MMDF 提供了许多机会可以修改环境，但如果不愿意这么费事且无太多额外要求，则可以采用最小配置，使操作步骤变得非常简单。为了有助于决定该进行哪一步操作，本章中给出了大量的图表。这里的许多内容都是从 /usr/mmdf/mmdftailor 文件中摘录下来的。摘录命令如下：

```
nl /usr/mmdf/mmdftailor
```

### 4.1.3 获得更多信息

下面给出部分与邮件处理密切相关的命令解释，这对理解 MMDF 中不同组成成分之间的相互关系很是重要：

表 4.1 邮件处理命令

命 令	描 述
Checkaddr (ADM)	为保证正确，检查邮件地址。
Checkqueue (ADM)	报告 MMDF 队列状态。
Checkup (ADM)	检查 MMDF 系统配置。
Cleanqueue (ADM)	发送警告并返回终止邮件。
dbmbuild (ADM)	建立 MMDF 数据库
deliver (ADM)	发送邮件至 MMDF 通道
list (ADM)	处理邮件列表
mmdf (ADM)	在本地或通过网络发送邮件
mmdfalias (ADM)	将 XENIX 风格的文件转换为同名的 MMDF 文件
mnlist (ADM)	将 XENIX 风格的 Micnet 路径文件转化为 MMDF 文件
rmail (ADM)	提交通过 UUCP 收到的远程文件
submit (ADM)	接收将由 MMDF 进行处理的邮件
uucico (ADM)	传输 UUCP 文件
uulist (ADM)	将 UUCP 文件转化为 MMDF 格式
Checkmail (C)	检查已提交但未发送的邮件
mail (C)	在用户之间传递消息
uucp (C)	通过串行线路在 UNIX 机器之间传送信息
rcvtrip (C)	通知发送者：用户已离开
uuname (C)	指定新的机器名
uux (C)	传输 UUCP 文件并在远程系统上执行命令
logs (F)	登记 MMDF 信息
maildelivery (F)	描述用户对传送的指示文件
mmdftailor (F)	提供实时剪裁以满足具体需要
queue (F)	对等待传送的邮件文件进行排队操作
systems (F)	UUCP Systems 文件格式
tables (F)	描述别名、范围以及主表 (host table)

#### 4.1.4 系统管理人员的任务

在 MMDF 系统配置好后,所需的系统管理任务并不太重。但以下工作必须要做:

- 将过时的邮件从邮件队列中清除。
- Alias 文件需要进行维护。另外,绝不能将该文件的权限设为“other”。
- 监视 /usr/spool/mail 目录和 /usr/spool/mmdf 目录的空间,邮件接收在前中完成,而后者则存放队列中邮件。
- 定期从 /usr/mmdf/log 目录中删去登记文件。

#### 4.1.5 MMDF 的术语

在 MMDF 中,“Local machine”与“Local machines”这两个概念是有区别的,前者指的是你正在其上工作的机器,后者指的是你的组织中的机器。另外,术语“tables”意为“files”,所以,domaintable 即是指包含范围定义的文件。

### 4.2 MMDF 是如何工作的

本节讲述邮件穿越 MMDF 系统的过程。涉及的大多数概念都有“发出去”和“收进来”两方面的含义。本节讨论了以下两方面的内容:

- 发出的邮件——讨论发出的邮件、通道、范围、别名文件以及 mmdftailor 文件。
- 进来的邮件——讨论发至本系统的邮件。

本节讨论电子邮件的执行过程以及 MMDF 所独有的概念。

#### 4.2.1 发出的邮件

发出邮件的开始是由用户执行一个邮件命令,比如 mail (C)。在传送的消息 (message) 建立好之后,即以如下路径穿越 MMDF:

mail(C)命令让用户建立、编辑和启动文件传送。mail 程序将邮件交给 execmail,由它将邮件送至一个队列中,等待 submit(ADM)程序提供服务。submit 程序从“to:”行中取得路径,决定由何种类型的网络通道传递消息,并将消息排入队列,由 deliver(ADM)程序传递。

邮件头(mailheader)决定消息如何穿过系统进行传送,下面将说明邮件头的正确格式。

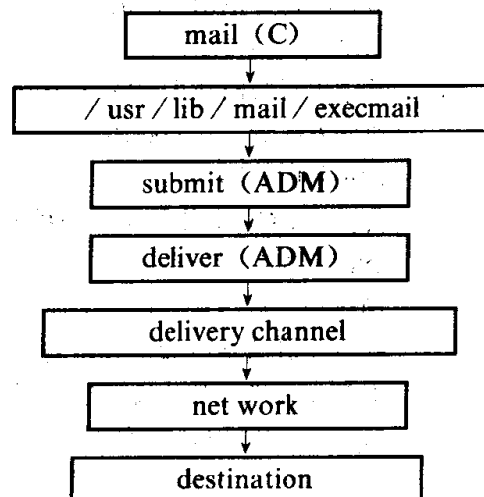


图 4.1 发出的邮件穿越 MMDF 的过程

### (1) 邮件头

邮件头至少包含如下各行，形式如下：

From: sendey

TO: recipient

Date: Weekday Momday hh: mm: ss: year

一个实例为：

From: george heSCO.COM (George Hoffer)

TO: market

Date: Fri Feb 2 14: 09: 47 1990

大多数的提交程序，如 mail (C) 和 exemail，还要在邮件头中另外增加若干行。增加新行是允许的，但对新行的解释应由接收邮件服务器及其用户接口来完成。

### (2) 传递程序

deliver (ADM) 程序从 submit 排好的邮件队列中取出邮件并将其交给合适的通道，传输通道提供给可以访问的网络（比如 UUCP 或 Micnet）。通道程序通过网络将邮件发送到一定的目的地。

### (3) 传输通道

通道是实现机器与某一种类型的网络通信协议进行对话的编译程序。有些简单的通道只是存放邮件以备进一步的处理。通道程序驻留在 /usr/mmdf/chans 目录中，可以由 deliver (ADM) 根据在 /usr/mmdf/mmdftailcr 文件中包含的信息进行访问。mmdftailcr 文件将在本节稍后部分加以讨论。

对每个通道都提供了两个程序：一个输入程序和一个输出程序。比如，UUCP 通道以 /usr/bin/rmail 作为输入程序，而以 /usr/mmdf/chans/uucp 作为自己的输出程序。

“channel”程序与发出去的邮件有关，而“server”程序则与进来的的邮件有关。在 MMDF 系统中，server（服务程序）是监视网络上进来的邮件的通道程序。在 MMDF 之外的其他系统中，邮件服务器的功能建立在邮件传输系统中。“channel”程序与“server”程序之间的联系见下图：

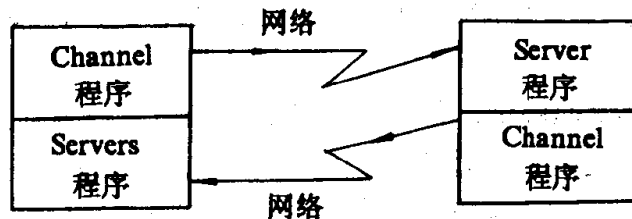


图 4.2 “channel”与“server”的联系

如果只在接收邮件的一方采用 MMDF 系统，另一方的处理程序虽然不是 channel 程序，但所作的传递邮件工作完全是模仿 channel 程序的。

由于有通道程序负责处理通信协议，所以无论是操作系统还是 MMDF 的其余部分都不必知道特定通信协议的具体情况。这种通信协议的处理方式是 MMDF 的优点之一，即只要有特定协议的程序模式，便可以将某一网络类型升级或改变为另一个，而不必重新编

写邮件传输系统。

submit (ADM) 命令指定要访问的通道, 由 deliver (ADM) 将邮件送至该通道。delivev 程序还可以将邮件放入一个通道, 并让另一个程序执行额外步骤确定该类型通道的情况。

不同的通道有:

badhosts: 当本地机器不能识别指定的机器时即访问本通道, 其实, 称该通道为“bad”hosts 是不妥当的, 因为所有发往未知机器的邮件都被送到本通道中。madftailcr 文件允许将 badhosts 指定为一种通道类型, 比如 uucp 或 smtp。

badusers: 当邮件到达本地机器, 而用户在这台机器上没有注册账号, 且在本地机器上也没有该用户别名时访问该通道。一般来说, 这时应将邮件放入队列, 等着提交给用户较多的另一机器。

list: 重发消息时访问。本通道简单地启动 submit, 将地址和正文反馈回 MMDF 邮件系统。这经常用于避免长地址确认或者在邮件清单非常大时强制出现地址确认。

local: 在将邮件传递给本地机器上的邮箱和进程时访问本通道。

micnet: 在通过 Micnet 网络连接传递或接收邮件时访问该通道。

uucp: 在通过 UUCP 连接向另一台机器发送邮件或从另一台机器上接收邮件时访问该通道。进来的邮件转化成由 DDN: NIC (网络信息中心) 指定的格式, 而出去的邮件加上一行形如“From <空格>”的内容, 邮件路径参数由(!)号隔开。

uux (C) 命令将邮件传输至另一台机器上。UUCP 以一台机器只知道其相邻机器地址的非正式网络形式工作。在一条或多条消息已存放起来准备发往另一台机器时, UUCP 在 UUCP 配置文件允许的前提下启动一次传输。

UUCP 网络的工作是“步进”的, 一台机器访问另一台机器并将邮件发送给它; 而它又顺次访问下一台机器并邮件发送给下一台, 这样, 经过若干步路径选择之后到达目的地。这种“步进”方式又称为“hops”。一个 hop 即是邮件在两台机器之间所走的一步, 这适合任何类型的通道。如果只有两台机器传递消息, 以其中一台作目的地, 则邮件称为“单 hop”的。如果邮件要到达最终的目的地之前要经一台或多台机器中转, 则称该邮件为“多 hop”的。

另外, 在加入了 TCP/IP 软件包之后, 通道文件可用于 SMTP。(TCP/IP 表示传输控制协议/接口程序。)

通道的作用不只是处理协议, 如有必要可实际启动与网络或另一台机器的通信。

SMTP 通道通过与远程机器建立 TCP/IP 连接的方法传消息, 并用简单邮件传输协议 (simple Mail Transfer Protocol) 发送一条或多条消息。Internet 协议 (IP) 允许多个局域网或广域网相互透明地互连。这允许 MMDF SMTP 通道通过网络与任何机器交换信息。比如, 如果你的机器与 Internet 相连, 则机器可以与世界上所有的机器直接交换信息, 条件是它们也连到 Internet 了。

**域: (Domains)**

信息被传递到指定的机器上, 该机器的位置由接收者地址信息中的系统名部分给出。这些系统名即称为域。域以组织、部门或地点所在的组织或国家的形式指定机器。域名唯一地标识出机器, 但并未给出到达该机器的路径。“From:”行以及其他可能的行都包含域

名信息。

域描述了邮件发自或发往的组织。在描述 MMDf 时，用到以下类型的域名：

- 最高级别域——最高级别域是描述一个组织的性质或与国家有关的代码的正式登记名。比如，以 com 表示商业组织，或以 fi 表示芬兰。

在 Internet 上定义了如下的最高级别域：

COM: 商业机构

CODE: 国家代码

EDU: 教育和研究机构

GOV: 政府机关

MIL: 军事机构

NET: 网络

ORG: 组织 (类别)

UUCP: 一个未登记的域名，其中用户通过 UUCP 在相邻的机器之间交换信息；

国家代码是由国际标准组织 (ISO) 标准 3166 定义的，域名可以小写也可大写，MMDf 在识别域名时不加区分。

子域——子域也是正式登记名，给出的是公司、部门或者在一最高级别域之下的子组 (subgroup)，比如子域“SCO”。

本地域——本地域是只在本组织内部识别的域，只在组织内有意义。比如，某一个部门名为“engr”。

- 全称域——全称域名是机器名以及其他所有域名，其顺序如下：

机器名.部门名.公司名.最高级别域

或者: 机器名.公司名.最高级别域。

下面给出一个例子：

mama.comp.tortilla.com

含义为：在地点 Tortilla 的 comp 机器中心的机器 mama，该组织是一个商业组织 (Com)。

下面给出全称域名的用法：

ucsec.ucsc.edu 在 UC Santa Cruz 的机器

seismo.css.gov 在地震仪研究中心的机器

最高级别域名和子域名在以下标准组织中登记：

DDN 网络信息中心 (NIC)

SRZ International

333 Ravenswood Avenue, Room EJ291

Menlo Park, CA 94025 USA

以后我们将把该组织简称为 NIC。如果你或者你所在地的某位用户需要往所在组织外发送邮件或从组织之外接收邮件，应该在 NIC 登记最高级别域或子域。即使只想使用 UUCP，也只有在 NIC 登记才能保证你的系统名不被重复使用 (唯一)。越早登记域名并获取一个地址，日后被迫更换机器名或其他系统标识符的可能性就越小。

域文件搜索算法

submit (ADM) 命令使用域表的意图有两种：一是指定目标机器（作为邮件目的地的机器），二是有选择地指定到达目的地的路由，办法是列举出邮件必须经过的一台或多台机器的全称域表。邮件要传送到每一个地址都要在域表中进行查找，直到找到匹配者为止。MMDf 要用到如下的查找算法：

1. 把在地址中指定的机器名和域表左边栏目中指定的所有机器相比较。

2. 将地址中指定的机器名，附加上域表名（即在 mmdftailcr 文件的 MDMN 条目中的第一个参数），与域表左边栏中的域名进行比较。

3. 将地址中指定的机器名，附加上域表名，与域表左边栏中的未修改的名字相比。

4. 将地址中指定的机器名，附加上域表名，与地址及域表左面栏目中的条目进行比较。

比如，对 Tortilla 公司来讲，如果域文件的左边栏中含有“finance”，则根据上述第 2 步的规则，与下面的地址匹配：

jane@finance.tortilla.com

此外，根据第 4 步的规则，下面的地址也是匹配的：

jane@finance

如果对地址中的完整域名没有搜索到匹配者，则搜索过程便将域中最左边的部分去掉，然后再次搜索，直到找到匹配者，或者地址在去掉若干部分之后只剩下最高级别域了才停止搜索。所以地址“perry@blue.sco.com”的搜索过程为：首先搜索“blue.sco.com”，若不成功，就搜索“sco.com”，如果还不成功，则搜索“com”，若到了最高级别域搜索仍不成功，就认为地址非法，如果 badhosts 通道存在则通过它进行传送。如果在 mmdftailor 文件中将 badhosts 通道的信息指定为注释而无意义，则会产生错误信息。

如果搜集时找到了完整域名（即没有从域名的最左边去掉任何一部分地址信息）的匹配项，则用匹配项所在行的域表的右边栏内的内容取代域名。例如，Tortilla 公司的域文件包含有下面信息的域文件：

finance finance.tortilla.com

则地址“jane@finance”即会被重写为：

jane@finance.tortilla.com

如果为域的一部分找到了匹配项，则该匹配项所在行的域表条目中右边栏的信息被当作邮件必须经过的机器名，原来的域名被保留下来。比如，如果根域表包含以下条目：

sco.com gateway.tortilla.com

则地址“perry@blue.sco.com”将会被重写为：

@gateway, tortilla.com: perry@blue.sco.com

这使得邮件到达目的地址“blue.sco.com 之前必须经过下面的路径：

gateway.tortilla.com

不管发生上述哪一种匹配情况，如果有域表的右边不止一个域，则第一个域作正常处理（或者替代目的域或者用作邮件到达目的地之前必须经过和路径，这取决于匹配的类型），但之后的各域也被当作附加的中间节点，邮件必须首先经过这些路径。比如，如果根域表包含：

sco.com gateway.tortilla.com localgate.tortilla.com

则地址被重写为：

@localgate.tortilla.com, @gateway.tortilla.com: perry@blue.sco.com

在结果地址形成时，使邮件首先经过路径：

localgate.tortilla.com

然后经过路由：

gateway.tortilla.com

最终邮件到达：

blue.sco.com

在匹配项所在行中下面域之后的附加中间机器

local gate.tortilla.com

应该被插入：

gateway.tortilla.com

之前，也就是说，下面的域必须是邮件在到目的机器之前所经过的最后一台机器：

gateway.tortilla.com

别名 (Alias) 文件。

MMDF 提供了一系列文件，你可以在其中指定用户名或者程序的别名。别名是缩写或简写的名字，由 MMDF 转换成较为复杂的字符串。这一特性很有用，可以指定一个简单名字来表示一组用户。比如，某公司用“sales”来作为别名，表示公司销售部的所有成员。于是当用户想给销售部的每个成员发送邮件时，只需打入：

mail sales

就可以了。这个别名文件，可以像这样定义条目：

sales: joe, jane, bob, mike, karen, ann

当给别名“sales”发送邮件时，别名中的每个人都会收到邮件的拷贝。

对使用别名文件的数目没有限制，其名字在 /usr/mmdf/mmdftailor 文件中说明。系统为你提供了三个文件可输入别名，这些文件是：

- alias.ali——用于本地机器系统的别名。本文件至少应包含 mmdf 的别名。
- alias.list——用于输入用户名单的别名。本文件至少应标识出 postmaster 的别名。
- alias.user——用于输入属于用户以及用户“专用”机（即用户接收邮件时所在机器）

的别名。

alias.ali 文件

alias.ali 用于不属于 alias.list 或 alias.user 文件的别名。该文件通常包含说明性的别名。如：

writers hanna, dianna, karen

sysmoms angela, john, eric

alias.list 文件

alias.list 文件是为多用户别名提供的。使用 list 别名，可以表示出邮件是从发送者而不是从“TO:”一行中的第一个人那里发出的。下例中给出的 3 行都是访问 list 通道所需要的。使用 list 通道要求有 3 行：list 名，包含组成该 list 的各注册号的 name-outbcund 行，以及给出能够对 list 进行增删的 list 维护人员注册号的 name-request 行。

sales sales-outbcund @list-processor



```
scdes-outbound ripleymama.com.comp.tortilla.com, george. hanna
```

```
sales-request hanna
```

关键字“list-processor”是 MMDF 的保留字，指示使用 list 通道处理邮件 list 的路径，sales-request 别名提供了请求对 sales 别名进行增加或删除的办法。

alias.user 文件

下面给出一个 alias.user 文件的例子，把用户与机器一一对应起来，这用在你部门里每个人都在自己专用的机器上接收邮件时：

```
hanna hanna@ guardian
```

```
karen karen@ writer
```

```
george george@dera.EDU
```

### 邮件路由选择的实现

当邮件发往 postmaster 时，MMDF 先从 alias.list 文件中搜索无序的别名表，将 postmaster 别名扩展为相应的用户名。在 alias.list 文件中可能有如下这样一个条目：

```
postmaster: perry
```

接着 MMDF 搜索 alias.user 文件，寻找与用户名相应的本地机器名。alias.user 文件可能包含：

```
perry: perry@blue
```

MMDF 搜索将本地机器名与完整的域名对应起来的不同的 .dom 文件。在这里，机器 blue 在本地域中，所以 MMDF 在 local.dom 文件中找到如下条目：

```
blue blue.sco.COM
```

接着 MMDF 搜索将完整域名与寻址数据对应起来的不同的 .chn 文件。在此，域 blue.sco.COM 是由 Local 通道服务的，所以 MMDF 在 local.chn 中找到如下条目：

```
blue.sco.COM SCO
```

根据 mmdftailr 中对 MCHN 的定义，Local 通道将邮件放入 /usr/spool/mmdf/lock/home/q.Local 文件中的队列。Local 通道程序（在 /usr/mmdf/chans 目录中）将邮件发往 perry 的邮箱。

mmdftailr 文件

/usr/mmdf/mmdftailr 文件定义了本地机器的所有邮件属性，比如名字，用的是哪一个域，哪一个通道，以及哪一个名文件，每个通道是如何建立的，怎样执行登记等。行号由下列命令产生：

```
nl /usr/mmdf/mmdftailr
```

缺省的 mmdftailr 文件如下：

```
1.;
```

```
2. @ (#) mmdftailor1.3 89/05/26
```

```
3.;
```

```
4. First, We define the local domain and system name.The default
```

```
5. domain is UUCP.If our organization only has one machine, We
```

```
6. may just name it as in madine.UUCP, for instance.otherwise,
```

```
7. We might have a number of machines, and have the machine name
```

8.: be below an organization name as in machine.company.UUCP.of  
9.: course, if we have a registered domain name, we use that instead  
10.: of the default (pseudo) domain UUCP. UUnane is used only for  
11.: the UUCP channel, and is the short UUCP name that we use  
12.MLDOMAIN UUCP

第 12 行描述了本地机器所用的最高级别域。UUCP 是类别名，可以是 UUCP 连接，也可表示 Micnet 连接。如果本地机器有正式登记的最高级别域或子域，则就使用本组织的最高级别域，比如 COM，或者 EDU。

13.MLNAME scosysv

第 13 行给出了本地机器的名字。如果第 14 行未被作为注释而去掉，第 13 行就包含子域名，比如“SCO”。

14.; MLOCMACHINE systemid

第 14 行是本地机器名。“systemid 指的是存放于 / etc / systemid 文件中的名字。只有在组织中的机器名是“隐藏”的情况下才使用本行，例如，机器以地址形式出现，比如“blue”。

15.UUname scosysV

给予 UUCP 的机器名。这一名字通常与第 13 行或 14 行中列出的名字是一致的，但如果需要的话，尽可不相同。

16.; MSUPPORT is the address to which problem notifications concerning

17.; the delivery of mail are sent.It must be a valid address

18.; MSUPPORT postmaster

第 18 行是本地机器上有问题的邮件送往的地址。本地址由 deliver 和 submit 在处理从本地机器发出的邮件时进行访问。“postmaster”一般被认为是 Internet 上的查询地址。

19; Lets define all the tables we will use...

20 MTBL auser, file = “alias.user”, show = “User alias”

21 MTBL lalias, file = “alias.list”, show = “List channel Aliases”

22 MTBL alias, file = “alias.ali”, show = “Local Name Aliases”

23 MTBL local, file = “local.chn”, show = “Local Host Aliases”

24 MTBL list, file = “list.chn”, show = “List channel”

25 MTBL mnchn, file = “micnet.chn”, show = “SCO Micnet Channel”

26 MTBL mndom, file = “micnet.dom”, show = “SCO Micnet Domain”

27 MTBL uuchn, file = “uucp.chn”, show = “SCO UUCP Channel”

28 MTBL uudom, file = “uucp.dom”, show = “SCO UUCP Domain”

29 MTBL rrootdom, file = “root.dom”, show = “Root Domain”

30 MTBL Locdom, file = “Local.dom”, show = “Local Domain”

第 20 行至 30 行给出了缺省别名，通道和域文件（也称作域表）的文件名。（MTBL 意思是 M MDF table）在 MTBL 之后的第一个参数是标识对文件的描述的关键字。参数“show”用于显示，由程序如 checkup (ADM) 显示有关信息。

第 20 行给出有关用户名之别名的别名文件，第 21 行描述用于邮件 list 的别名文件，第 22 行描述了不能归类为另两种别名文件的一种别名文件。在第 18 行的地址“postmast-

er”在第 21 行的 `aliast.list` 文件中有别名。关于别名问题的进一步的信息将在第 32 行至 34 行加以说明。

第 23 行描述了用于本地机器的 `local` 通道文件，第 24 行说明用于解决邮件 `list` 问题的 `list` 通道。第 25 行描述了 `Micnet` 通道，而第 27 行则给出了 `UUCP` 通道的说明。通道信息将在第 36 行至 45 行作进一步的说明。

第 26、28、29、30 行分别对 `Micnet` 域，`UUCP` 域，根域及 `Local` 域作了说明。域文件一般以通道类型来命名，但 `root.dom` 是一个例外。它还包含了如何访问最高级别域的信息。（“`root`”的意思是层状结构的最高一级。）在第 45 至 50 行对域的情况作了进一步说明。

```
31.; let's set up the aliases
32  ALIAS table=alias, trusted, nobypass
33  ALIAS table=lalias, trusted, nobypass
34.  ALIAS table= auser
```

第 32 至 34 行定义了关于别名文件的附加信息。“`table`”参数说明本行中描述的是哪个文件。“`trusted`”参数表明在所描述文件中的每个别名都可以用系统中任意一个用户（包括 `root`）的权限给文件或管道（`pipe`）发邮件。由于这给予用户极大的权限，所以一般在超级用户编辑文件时才设置“`trusted`”参数。“`nobypass`”数用于防止地址 `address` 别名穿越机制（`address alias bypass mechanism`）工作于该文件的别名上。

```
35; let's set up the channels...
36  MCHN local, show="local Delivery", que=local, tbl=local,
37.  ap=same, pgm=local, mod=imm, host="scosysv.UUCP"
38.  MCHN list, show="list processing", que=list, tbl=list, ap=same,
39.  pgm=list, mod=imm, host="scosysv.UUCP,"confstr=sender
40.  MCHN micnet, show="SCO Micnet Delivery", que=micnet,
    tbl=mnchn,
41.  ap=same, pgm=micnet, mod=imm
42.  MCHN UUCP, show="SCO UUCP Delivery", que=UUCP, tbl=uuchm,
43.  ap=same, pgm=UUCP, mod=imm
44; MCHN badhosts, show="last-chance Routing", que=badhosts,
    tbl=mchm;
45;  ap=samw, pgm=micnet, mod=imm, host=smart machine.UUCP
```

第 36 行至 45 行提供了每个通道的附加信息。（`MCHN` 表示 `MMDF` 通道），`MCHN` 之后的第一个参数是描述通道的专用名称。“`show`”参数用于显示。其余参数描述如下：

`ap`: 指定发出去的邮件的邮件头地址类型。“`ap=same`”表示邮件头的格式没有改变。

`badhosts`: 指定在本地机器不能识别目的机器时邮件送往的地方。`badhosts` 的“`host`”参数指定了邮件将送至的机器。一般认为邮件送往的机器可以与外界相连，并将把邮件送至某一网络中。

`confstr`: 指明显示的消息发送者是发送邮件的人而不是邮件 `list` 上的第一个人。“`confstr=sender`”通知 `list` 通道将错误信息送至消息的原始发送者而不是送给在“`-request`”别名中给出的 `list` 维护者。

host: 指示通道转发所有邮件的智能机器的完整域名。local 和 list 通道都使用本地机器。

mod: “mod = imm”表示邮件马上发送。

pgm: 指示通道的传输方式。指出通道程序的文件名。通道程序存放于 /usr/mmdf/chans 目录中。

que: 指示 /usr/spool/mmdf/lock/hom 目录的目的子目录, 该通道的消息队列将放于其中。这里给出的名字以 q. 作为前缀以形成子目录名。

tbl: 指示作为 MTBL 行的头一个参数显示的名字。tbl 参数将在此处提供的信息与以前的通道 MTBL 条目连接在一起。

46; Bring on the domains...

47; MDMN “scosysv.UUCP”, show = “local domain”, table = locdom

48; MDMN “Micnet”, show = “Micnet Domain”, table = mndom

49; MDMN “UUCP”, show = “UUCP.Domain”, table = UUDOM

50; MDMN “”, show, “Root Domain”, table = rootdom

第 47 至 50 行描述了域文件 (又称为域表)。第一个参数指出该表列举的域的名字。“table”参数转回前一个 MTBL 条目去指示读文件。“show”参数用于显示之用。

51.; logging levels

52.MMSGLOG level = FAT, size = 20

53.MCHANLOG level = FAT, size = 20

54.AUTHLOG level = FAT, size = 20

第 52 行说明信息是如何从 deliver (ADM) 和 submit (ADM) 进行登记的, 第 53 行说明如何登记 MMDF 所支持的、除授权登记信息之外的所有程序的信息, 而授权登记信息则在第 54 行说明。

“level = FAT”指示只登记致命错误。“size”参数现在没有采用。

#### 4.2.2 进来的邮件

进来的邮件的处理情况如图 4.3 所示:

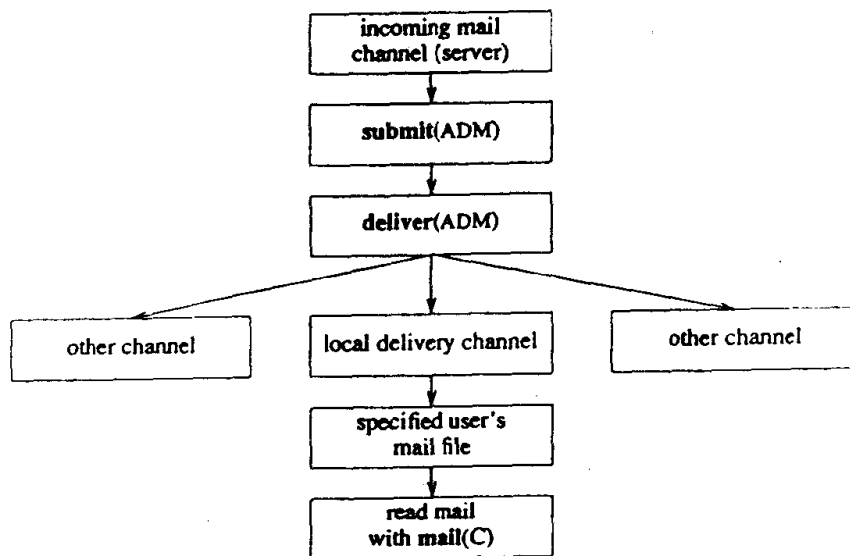


图 4.3 进来邮件的的处理过程

发进来的邮件是由通道的输入程序（称为服务器）来接收的，如 UUCP 程序的 rmail (C) 程序。submit (ADM) 程序确定邮件的发送方向，并分配接收邮件的通道。此时所选择的通道可能为：对本地机器为 local 通道；如果机器是本地的，但用户却不是，则应为 badusers 通道；当需要作进一步的处理时为 badhosts 通道；如果本地机器只是充当中间机器，邮件还需要转发时，也可能是其它网络通道。发送程序将邮件放入通道，最终将其发送至用户。

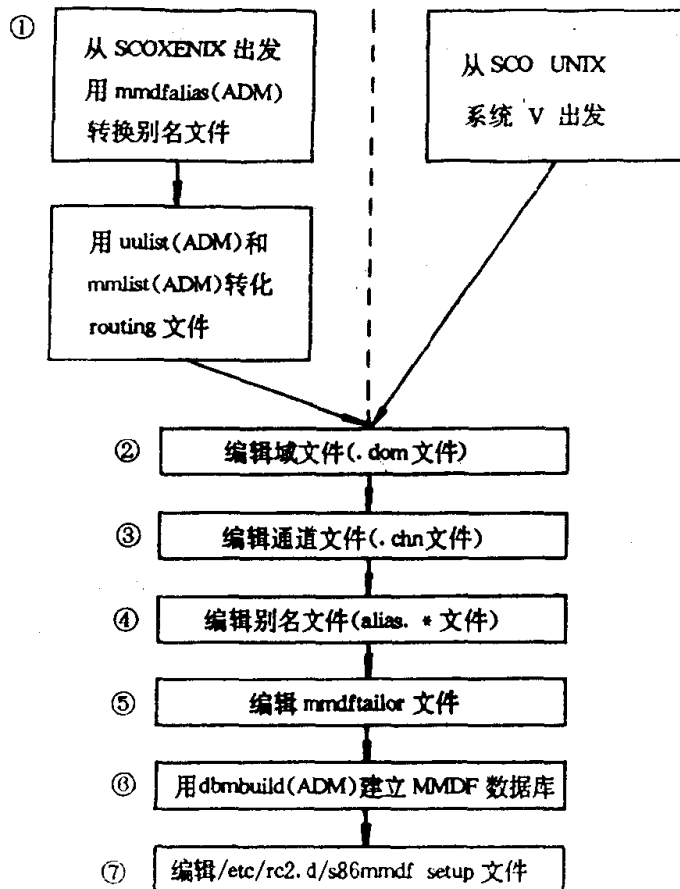
### 4.3 配置 MMDF

在开始之前，先看表 4-2，确定该如何按以下给出的步骤执行（可能不必执行所有步骤，这决定于你的系统）。在表格之后，有一张流程图列出了所有的步骤。对照表格与流程图，看看你的系统需执行哪些步骤。

表 4-2 配置 MMDF 的执行步骤

系统建立情况	需要执行的步骤
从 XENIX 转化成 UNIX	1 (其余步骤根据需要选用)
单独的标准系统，没有通道	只执行 5.1
连接到 UUCP 的系统	2-7 步
连接到 Micnet 的系统	2-7 步
连接到 Internet 的系统，或者连接到与外部有集中式连接的其它机器上的系统，包括连接到 LAN/WAN，到 Ethernet，或到 SLIP 线路的工作站	2-7 步

配置 MMDF 的步骤如下：



### 第一步：转化 XENIX 文件（如有必要的话）

在转化 XENIX 文件时，应在 mmdf 帐号注册，所有的转化程序，mmdfalias, mnlst 以及 uulist 都在 /usr/mmdf/table/tools 目录中。

在执行所有命令时都应输入完整的路径名，因为 /usr/mmdf/table/tools 目录并未包含在 mmdf 注册号的 PATH 中。

在将文件从 XENIX 操作系统转化成 UNIX 操作系统的文件时，应首先建立 Micnet 和 UUCP。这可以确保建立 MMDF 别名文件、通道文件和域文件所要用的文件包含正确的信息。请参考本书关于 UUCP 和 Micnet 的章节。在安装好 UUCP 和 Micnet 之后，将使用以下文件来转化 XENIX 文件：

表 4.3 用于转化 XENIX 文件的文件

文 件	用 于
/usr/lib/mail/aliases	mmdfalias
/usr/lib/mail/top	mnlst
/usr/lib/uucp/systems	uulist

在开始转化文件之前，执行 Ls-l，保证 aliases 和 systems 文件是 mmdf 可读的。转化完 MMDF 使用的文件之后，将 systems 文件中的权限方式恢复到原来的情况。

执行转化的过程如下：

1. 作为超级用户注册并输入如下命令：

```
chmod +r /usr/lib/uucp/systems
```

2. 以 mmdf 注册，并输入下面一组命令：

对 UUCP 文件而言：

```
cd /usr/mmdf/table
```

```
tools/uulist
```

对 Micnet 文件而言：

```
cd /usr/mmdf/table
```

```
tools/mmdfalias
```

```
tools/mnlst
```

3. 再次作为超级用户注册，并输入：

```
chmod -r /usr/lib/uucp/Systems
```

mmdfalias (ADM) 命令用于转化 XENIX 别名文件。该命令没有任何参数，mmdfalias 转化 /usr/lib/mail/aliases 文件，在当前目录中建立了 alias.list 和 alias.user 文件。

mnlst (ADM) 命令转化 Micnet routing 文件，uulist (ADM) 命令转化 UUCP routing 文件。这两个命令都不接受参数。mnlst 对 /usr/lib/mail/top 中的系统名进行归类，并用此信息在当前目录中建立 micnet.chn 和 micnet.dom。uulist 在当前目录下建立 uucp.chn 和 uucp.dom 文件。uulist 在建立新的通道文件和域文件时不需原来的

XENIX 文件。

如果某一输出文件已经存在，则在文件名尾部加上后缀“-”。在所有文件建立之后，保证权限值为 644，文件所有者和小组所有者（group owner）都是 mmdf。所有这些文件都应该放在 /usr/mmdf/table 目录中。

### 第二步：编辑域文件

如果你配置的系统使用 UUCP，则只要将机器名的信息加到 local.dom 和 uucp.dom 文件中就可以了。同样，如果你的系统只与 Micnet 相连接，只要编辑 local.com 和 micnet 文件即可。

域文件将机器名与一个全称域名相匹配。域文件有两种用途：将机器连接方式的信息传送给 MMDF；指定子域或最高级别域的特殊注意事项，域文件以通道类型来命名，而 root.dom 是个例外，它还包含有关访问最高级别域的信息。

通常的做法是对本地连接有一个域文件，即连接到域文件存放的机器上。

域文件名都有 .dom 格式。在操作系统存储卷中，/usr/mmdf/table 目录下提供了四个域文件，如有必要，可对每个文件进行编辑以加入关于你的系统的信息。操作系统提供的四个域文件为：local.dom, micnet.dom, uucp.dom 和 root.dom。

下面表格列出了随操作系统及附加软件包提供的几个域文件：

表 4.4 域和域文件

域	域文件	作用
local	local.dom	用于本地机器的域名
micnet	micnet.dom	Micnet 域名
root	root.dom	在其它域文件中不取名的域名
uucp	uucp.dom	uucp 域名

文件的权限应该都为 644，属主（owner）和小组（group）属主都应是 mmdf。

通常每个文件都应包含两部分信息。例如：

root.dom:

```
com  uunet.UU.NET
edu  ucsc.UU.EDU
mil  uunet.UU.NET
gov  uunet.UU.NET
net  uunet.UU.NET
```

“UU.NET”表示 UUnet 网络系统。域名可以用小写字母，也可用大写字母。在每行的两部分之间应用 tab 键、空格，冒号或者以上符号的组合来分隔开。头一部分指示名字，同行的另一部分内容则作进一步描述。从前面给出的表中可以看出，所有发往“com”的邮件都被送至 UUnet 上。

如果你的机器靠另一台机器来实现与外部环境联系，如靠 UUCP 或 TCP/IP 网络，则应在域文件中以通道类型的形式指出另一台机器的名字。比如，有一台远程机器“cac-

tus”连接到 UUCP，则 uucp.dom 文件应为：

```
uucp.dom:
```

```
cactus    cactus.UUCP
```

这样所有需要通过访问 UUCP 送至“cactus”机器的邮件都被送往 UUCP 网络。

注意域文件的名称确定了搜索域的名称。比如，UUCP 的域文件（MDMN “UUCP”）应该只包含下列形式名称的条目：

```
machine.UUCP
```

本地域文件，比如 MDMN“SCO.COM”，应该为下列形式的本地机器所用：

```
machine.SCO.COM
```

对每种通道类型，相应的域文件都应遵循这些格式。

关于 Micnet 的情况，在第 5.4 步有相应说明。

考虑 LAN 的情况

如果你配置的 MMDF 用于局域网（LAN），则可以使用域文件在机器上分配处理负载，也可以指定一特殊的邮件处理服务器来处理所有消息。

你可以有以下几种选择：

- MMDF 服务器（server）——指定一台机器作网络服务器，该服务器也可以对外部进行访问。每台机器的域文件只需要加进含有该服务器全称域名的 badhosts 和 badusers 通道。网络中每台机器的域文件都可以相同。网络可以扩大，可以增加或减少机器，而不会对其他机器的域文件产生影响。这种办法的缺点是如果服务器出了问题，所有机器之间的邮件传输就只能停止。改进的办法是使系统中的所有机器围绕处理器进行分组，系统中有多个处理器，它们之间可以通信。

- 分布式处理——每台机器的域文件中都有网络中其他所有机器的信息。这种办法的优点是网络中的机器可以独立于其它机器工作。其中一台出了问题不致影响到其它机器的邮件收发能力（除非这台机器通过 UUCP 连接到另一个网络中）。缺点是当网络中增加机器或者从网络上移出机器时系统管理需要做的工作要困难许多。当机器或者用户在网络中增加或移去时，所有的域文件或别名文件都必须加以更新以适应新的要求。因为域文件中可能包含关于本地机器的冗余信息，所以有可能在每台机器上使用一个域文件。分布处理的办法一般只用于小型网络。

- 门通路——这种结构源于服务器结构。门通路用于将一个网络连到另一个局域网或广域网上，而服务器机器一般是连接多个网络。门通路机器除了包含 LAN 上本地机器的名字之外，还包含着通过其它网络可以到达的别的机器的名字。这些信息保存在各网络的域文件和通道文件中，同时也保存在 root.dom 文件里。LAN 上的其它机器可以使用 bad hosts 通道将非本地邮件发送到门通路机器上。

### 第三步：编辑通道文件

如果你配置的是使用 UUCP 的系统，则你只需要把关于你的机器的信息加到 local.chn 和 UUCP.chn 文件中就可以了。类似地，如果你的系统只通过 micnet 连网，则只编辑 local.chn 和 micnet.chn 文件。

/usr/mmdf/mmdftailor 文件中的 MCHN 定义指示 MMDF 在



/usr/mmdf/table 目录中查找指定的.chn 文件, 以获取通道定义信息。

local.chn 文件中包含着类似以下各种形式的本地机器名:

sco.COM	sco
sco	sco
blue.sco.COM	sco
blue	sco

下面给出在缺省 mmdftailor 文件中与本部分讨论有关的内容。每个关键字左边的数字是行号。完整的 mmdftailor 文件在前面已经列出:

12.MLDOMAIN UUCP

13.MLNAME scosysv

14.; MLOCMACHINE systemid

上面的两个条目将全称域名和机器名对应起来。如果想将本地机器名隐藏起来, 则全称域应包括 MLOCMACHINE 中的值。对给定的缺省 mmdftailor 文件而言, 条目应该形如:

scosysv.UUCP	scosysv
scosysv	scosysv

从概念上来讲, 这与下列形式是一致的:

MLNAME.MLDOMAIN	MLNAME
MLNAME	MLNAME

如果第 14 行未作为注释行处理, 条目应该形如:

systemid.scosysv.UUCP	scosysv
scosysv	scosysv

从概念上来讲, 这与如下形式一致:

MLOCMACHINE.MLNAME.MLDOMAIN	MLNAME
MLNAME	MLNAME

list.chn 文件包含着:

list-processor	list-processor
list-proc	list-processor

其左边一栏是在邮件 list 别名中定义的伪机器名。这两个条目指示 MMDF 将指定地址为一个邮件 list 的邮件送给 list-processor 程序。

UUCP.chn 文件也包含类似的条目, 用于指示本地机器与外部是直接相连的:

mcvax.UUCP	uunet! mcvax! %S
sri-nic.ARPA	uunet! sri-nic.orpa! %s
uunet.uu.NET	uunet! %s

%表示使用此处地址的其余部分。左边一栏是 UUCP 机器名，右边一栏是 MMDF 在启动 UUCP 软件时所用的 UUCP 地址。在本例的第一个条目中，当邮件通过 mcvox.uucp 发往用户 hillis 时，UUCP 通道根据 UUCP 地址的剩余部分 (mcvox!hillis) 将邮件发至 uunet。第二条目表明了 UUCP 通路中是如何使用域名 (sri-nic.ARPA) 的。

如果目的机要经过几个 hops 才能到达，则 UUCP 通道的通道表条目应为如下形式：

```
uucp.chn:stooges.uucp moe! curly! larry! stooges! %s
```

在右边指定机器的“地址”。在此“地址”是一个 UUCP 通路。

Micnet 通道文件包含类似下面的条目：

```
ivy.sco.COM          ivy:%s
warwick.sco.COM     ivy:warwick:%s
```

左边一栏是取自 local.dom 文件的机器名，右边一栏是 MMDF 启动 Micnet 软件时使用的 Micnet 地址。在本例中，当邮件发往用户 ross (ross 在机器 warwick 上接收邮件) 时，Micnet 通道根据 Micnet 地址的其余部分 (warwick:ross) 将邮件传送到 ivy 上。

根据机器名在左边一栏，地址信息在右边一栏的模式，可以对 mmdftailor 中的每一个 MCHN 定义都建立一个通道文件。两栏之间可以用空格或 (和) 冒号隔开。

#### 第四步：编辑别名文件

对于机器名不“隐藏”的所有系统配置而言，本部分内容是任选性。关于“隐藏”的概念将在第 5.2 步“指定机器名和域名”中讲述。“隐藏”机器名要求在本地域中的任何一台机器都应知道如何将邮件发送给每一用户的“主”机器 (home machine)。这就需要有一个别名文件将用户及其主机器名对应起来。

你可用本部分介绍的内容建立别名文件或对其进行编辑修改。无论何时以何种方式改变 alias.list 或 alias.user 文件，都必须重新建立哈希数据库 (这在第六步讲述)。

alias.list 文件

alias.list 文件包含着 list 类型 (list-type) 别名，即指定一个名字来表示：

- 一个或多个用户名或其它别名
- 将一条消息重定向至一个文件
- 将一条消息重定向至一个管道
- 一个邮件 list。

比如：

```
postmaster:          admin,perry.LogUUCP
Loguucp:             "network // usr / spool / log / UUCP"
Loglog:              "network|cat V>> // usr / spool / log / mlog"
printer2:            "network| / usr / bin / lp d printer2"
staff:               staff outbound@list processor
staff outbound:     " !include: | / etc / alias / staff"
staff request:      ross
```

应该为自己的系统指定一个 postmaster，并定义 postmaster 别名。在本例中，发往 postmaster 的邮件被送至用户 admin 和 perry，并记录在 UUCP 登记（LOG）文件中。在重定向时用“/”语法是很方便的，可以将有关内容直接记录在登记文件中。“postmaster”地址通常被看作是 Internet 上的一类查询地址。

也可以使用通常的重定向符号（>）和管道（|）重定向之后做更为复杂的处理。发往 Logmlog 的邮件通过管道进入 cat（c）命令，然后登记在 mlog 文件中。发往 printer2 的邮件通过管道送至 LP（c）命令实施打印。以上这些重定向别名的例子都使用用户“network”的用户标识符和小组标识符，你可以指定 /usr/passwd 文件中的任何用户名。别名文件在 /usr/mmdf/mmdftailor 文件中的定义必须指定为“trusted”，这样才能在其中使用重定向别名。这样的文件别人（外界）是不能写的。

上例中的最后三行执行对“staff”邮件清单（list）的处理，表示出如何用“:include”格式通过在指定文件中列出的名字定义别名。你也可以用普通的重定向符号（<）从一个文件中读出别名信息。

在 alias.list 文件中，别名和其定义之间可以用空格、冒号或二者组合来分隔开。在定义包含许多用户名的别名时，可以使用反斜线（\）作为续行符。在用一个别名来定义另一个别名时，应注意不要出现“循环”定义的情况。

alias.user 文件

alias.user 文件包含着将用户和他们的“主”机器对应起来的别名。例如：

```
admin:          admin@blue
carmen:        carmen@ivy
perry:         perry@blue
ross:         ross@warwick
```

### 第五步：编辑 mmdftailor 文件

/usr/mmdf/mmdftailor 文件是 MMDF 的最高级别的配置文件。该文件包含着一定量的配置信息，并指示 MMDF 处理所有其它的 MMDF 配置文件。在编辑该文件时，应以 mmdf 帐号登记注册，并使用 UNIX 的文本编辑程序，如 Vi（c）。

在开始编辑该文件之前，应使用下面给出的一个表格来确定该如何根据系统的建立方式来进行 mmdftailor 文件的编辑。在表格之后，提供了一个流程图并列出了所有需要的步骤。根据该流程图，可以确定在各种可能的情况下应如何编辑该文件。

表 4.5 系统建立方式

系统建立方式:	应执行如下步骤:
只带电子邮件的独立系统	5.1
连接到 UUCP 的系统	5.1,5.2,5.3
连接到 Micnet 的系统	5.1,5.2,5.4
连接到 Internet 的系统，或者连接到其它与外界有集中式连接的机器的系统，包括连接到 LAN/WAN 的工作站	5.1,5.2,5.5,5.6

第 5.7 至 5.14 步是任选的，只在必要时才使用。注意，所有的参数名字都可以大写或小写形式出现，但是参数的值使用大写还是小写则有区别。

mmdftailor 文件编辑过程：

下页给出编辑 mmdftailor 文件的流程图。在左边的圆圈内是步骤序号：（如图 4.4）

#### 步骤 5.1 MSUPPORT

在本地机器上的邮件不可传输时，由关键字 MSUPPORT 给 deliver (ADM) 和 submit (ADM) 指定访问地址。邮件不可传输的原因可能是消息头的格式不正确，或者含有 MMDF 所不能识别的字符。MSUPPORT 在安装好的 mmdftailor 文件中出现形式如下：

##### 18.MSUPPORT Postmaster

将地址从“Postmaster”改成指定来处理邮件故障的注册帐号。注意，地址“Postmaster”被看作是 Internet 上的通用查询地址。有两种方法可以改变“Postmaster”：

- 将 Postmaster 的一个别名放入 /usr/mmdf/alias.list 文件中，使 Postmaster 成为一个别名，例如：

Postmaster mmdf

- 建立“Postmaster”帐号，保证有人每天检查该帐号，处理有问题的邮件。帐号建立可以从根号下以 sysadmsh (ADM) 命令来完成。

指定作为 MSUPPORT 别名的地址必须是合法的。如果不是合法地址，MMDF 就不会将原来不可传送的邮件送至该地址，因而又出现一件不可传送的邮件，如此循环下去，直至机器溢出为止。

可以指定任何用户来接收那些不可传送的邮件，但最好指定一个本地用户，因为其地址比较简单，不太容易出错。另外，如果指定非本地用户来接收邮件，则消息的传送就会依赖于网络连接，如果网络连接出现故障，就会导致邮件不可传递。

#### 步骤 5.2 指定域名和机器名

mmdftailor 文件的第 12 至 14 行定义了本地机器的全称域名。当你用 custom 命令安装好 MMDF 时，这些行的初始设置为：

12.MLDOMAIN UUCP

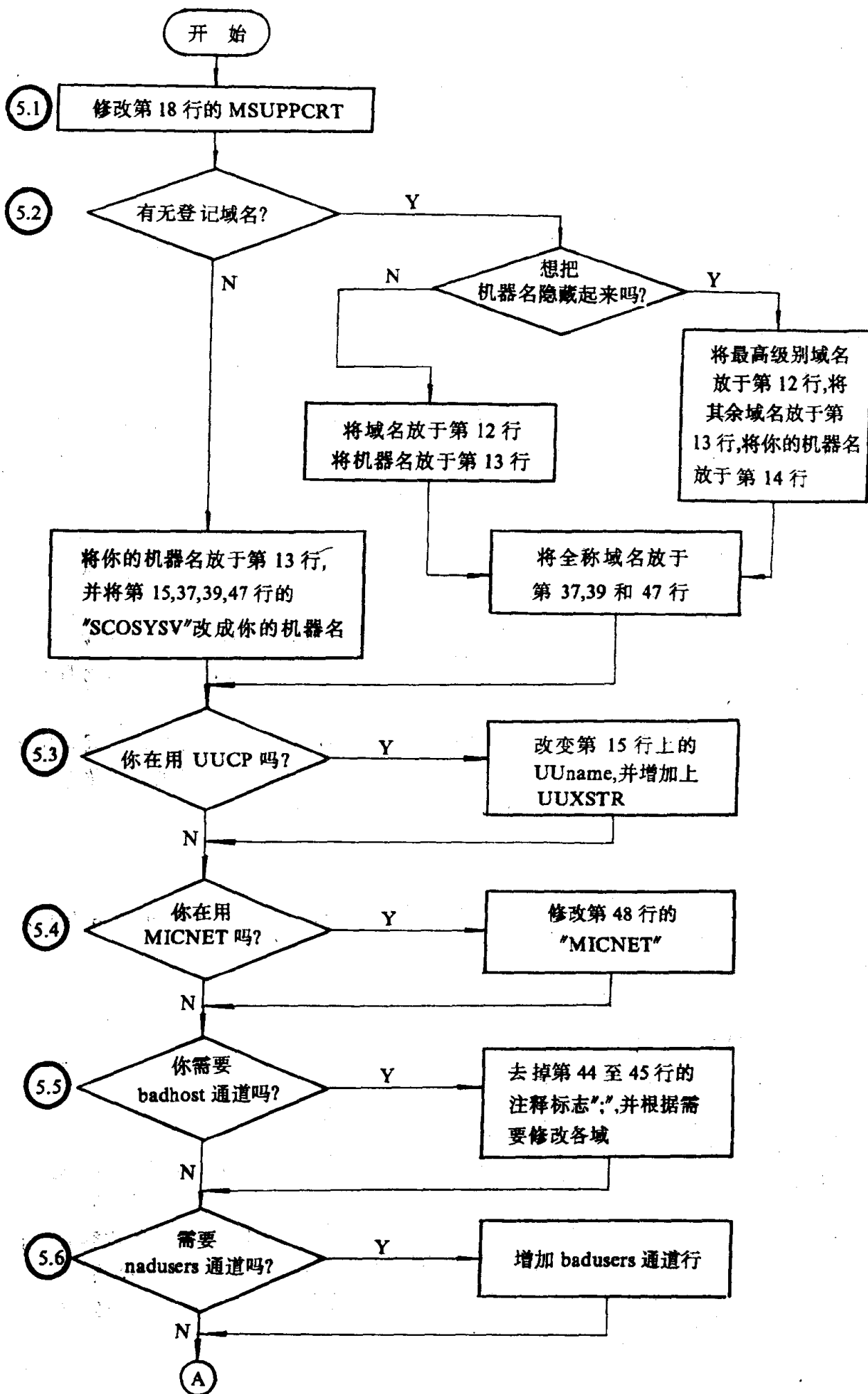
13.MLNAME scosysv

14.; MLOCMACHINE systemid

UUCP 是域名，scosysv 是机器名。

对于独立系统而言，可将安装时指定的名字作为机器名。

对连接到其它机器上的机器来讲，每个名字都必须是唯一的。如果机器连接到 UUCP，Internet 或者其它公共网络上，则机器名在该网络的地理范围内必须是唯一的。要保证机器名唯一，NIC 提供了登记功能（见“MMDF 是如何工作的”一节的内容）。当登记了机器名，公司名或者别的什么名字之后，就可确信系统在 NIC 服务的网络上不会发生机器名冲突。登记的名一个唯一的机器名。如果没有在 NIC 登记，则必须使用 UUCP 作为最高级别域名。



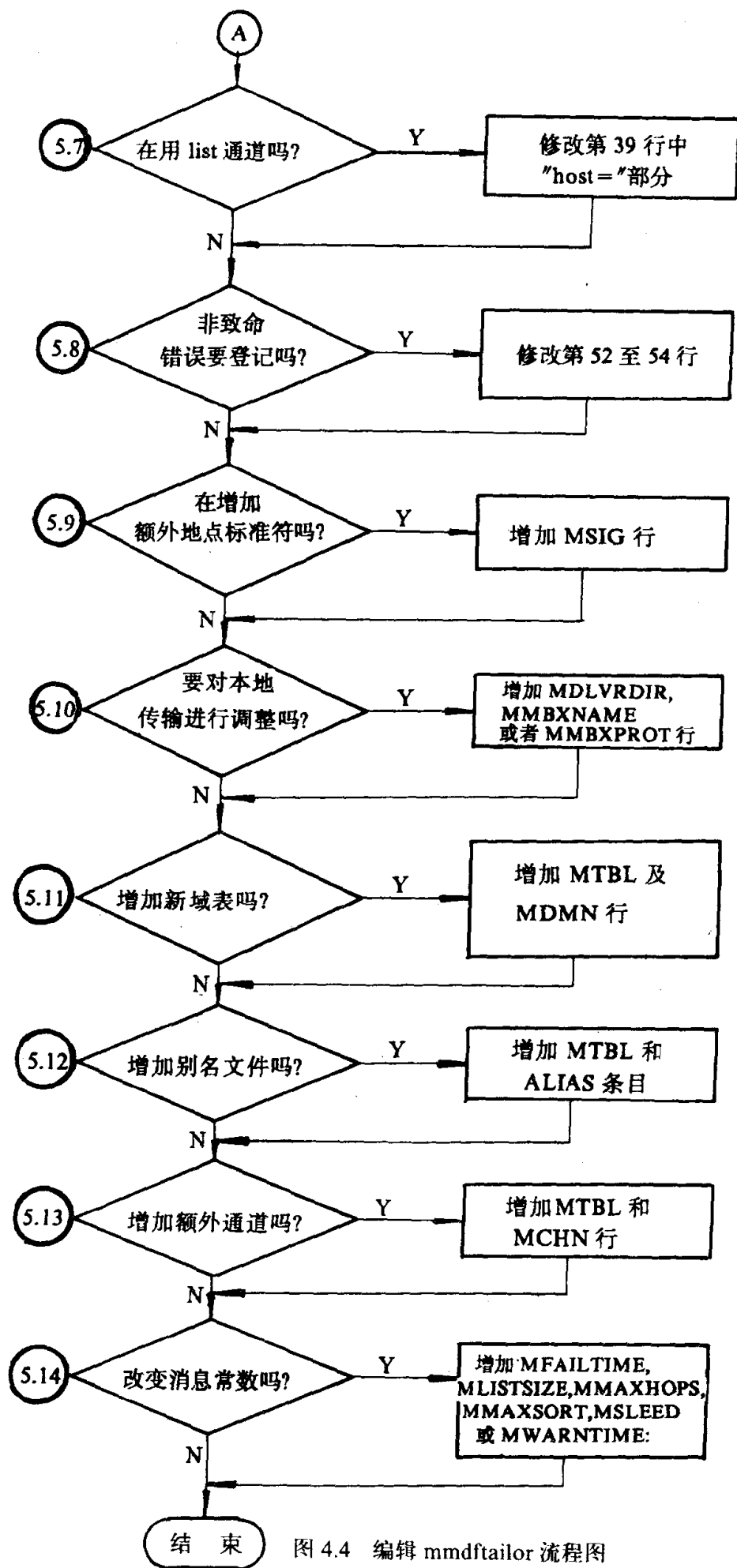


图 4.4 编辑 mmdftailor 流程图

如果使用已经登记的域名，则也许希望你的组织之外的用户可以向你组织内部的用户发送邮件，而不必知道组织内的用户将在哪一台机器上接收邮件。如果确实有此要求，则可指示 MMDf 将邮件看作来自同一个“源”的，而 MMDf 能够将这个来自同一“源”的邮件分送给正确的用户。这即是所谓“隐藏机器名”的含义。比如在 Tortilla 公司中，所有用户对外都标识为 @tortilla.COM。在你加入同一个域名下的本地机器组时，就建立了一个管理域。在该域中，所有的用户名必须唯一，这样邮件在域中传递时就不必在邮件地址中指明本地机器名而能够到达任一想到的用户处。

如果使用已登记的域名，而且打算：

- 隐藏机器名，则将 12 行中的“UUCP”改变为已登记的最高级别域名（比如 COM）；将第 13 行的“scosysv”改成已登记的域名，比如 tortilla；将第 14 行的“systemid”改成你的机器名。去掉 MLOC MACHINE 行的头一个分号“；”使第 14 行不再是注释行。

- 不隐藏机器名，则将第 12 行的“UUCP”改变为你的域名，比如 tortilla.COM，并将第 13 行的“scosysv”改变为你的机器名。如果你没有在 NIC 登记你的公司名，则必须使用第 12 行的 UUCP 最高级别域：

```
12.MLDDOMAIN COM
```

然后将全称机器名填入第 37、39 及 47 行。转而执行下一步。

如果没有登记域名，则应将第 13 行中的“scosysv”改为你的机器名。不要改变第 12 行。然后将第 15、37、39 和 47 行中的“scosysv”改变为你的机器名。

### 步骤 5.3 UUCP 的建立

如果机器使用 UUCP 和其它机器通信，则已经安装好的缺省 /usr/mmdf/mmdftailor 文件已经作了使用 UUCP 的相应配置。在整个文件中表示机器名的“scosysv”都应改变实际的机器名字（尤其是在第 15 行）。第 15 行，UUsername 行，给出用于 UUCP 通信的唯一的机器名。保证第 12 行含有下列内容：

```
MLDDOMAIN UUCP
```

另外，可在文件的任何一个位置上增加一行含 UUXSTR 参数的内容，指示在发送文件及在远程机器上执行命令时 UUX (C) 的有关资料并确定什么样的任选项能满足你的需要。

推荐使用下列三种设置之一：

```
UUXSTR "uux-"
```

```
UUXSTR "uux--r"
```

```
UUXSTR "uux--r-gA"
```

### 步骤 5.4 Micnet 的建立

这一步骤假定你的机器已经（或将要）与另一机器通过 Micnet 相连。这里唯一需要做的是将第 48 行 MDMN 条目中的字“Micnet”改成“UUCP”。如果有一个已登记的域名，则使用该域名来取代“UUCP”（第 47 行中指定了什么域名，这里就用什么名字）。这么做的原因是，“Micnet”不是一个合法的网络域。

### 步骤 5.5 badhosts 通道的建立

如果你的机器直接或间接地连接至一个网络，希望在邮件发送的目的机器未知的情况下将邮件通过某种连接送至网络中的另一台机器时，应执行本步骤。要使用这一特性，应从缺省的 mmdftailor 文件的第 44 和 45 行中去掉第一列的分号“;”，这两行的形式为：

```
44.;MCHN badhosts,show = "last-Chance Routing",que = badhosts, tbl = mnchn,  
45;ap = same,Pgm = micnet,mod = imm,host = smartmachine.UUCP
```

然后将“host = ”的值改为目的机器的地址；将“tbl = ”和“Pgm = ”参数值改为传送消息至指定的 badhosts 机器的通道名。

### 步骤 5.6 badusers 通道的建立

badusers 通道与 badhosts 通道非常类似，用于用户名与本地机器上的用户对应不上，且在别名文件中也找不到该用户时。复制 badusers 通道的 MCHN 条目，但要将字符串“badhosts”改为“badusers”。将“host = ”参数设置为一个别名文件集更为丰富的机器的名字。

### 步骤 5.7 使用 list 通道

这一步骤假设是在 MMDF 负担很重的条件下，想通过处理较长的别名的方法来减轻 MMDF 负担。本通道一般用来处理带有许多用户的别名清单。比如邮件清单。要使用 list 通道，可指定另一台机器来处理地址 lists。如果希望使用本地机器，则名字已在第 2 步时设置好，不需要其它步骤。如果希望指定另一机器，则应将“host = ”域设置为该机器名。应该在第 7 步建立 / etc / rc2.d / s86mmdf 文件时增加 list 通道。

对 list 通道的访问是通过缺省的 list.chh 文件使用关键字 list-processor 来实现的。

在 list-processor 以形如“listname-outbound”的别名对应起来的时候可启动 list 通道。

别名应为如下格式：

```
name:name-outbound@list_processor  
name-outbound: <list-of-recipients>  
name-request: <login of list maintainer>
```

下面给出一个实例：

```
sales          sales-outbound @ list-processor  
sales-outbound ripley @mama.comp.tortilla.COM,geore,hanna  
sales-request hanna
```

### 步骤 5.8 改变错误登记级别

mmdftailor 文件的最后部分设置了保存信息的级别和 MMDF 登记文件的最大尺寸，这些文件在 / usr / mmdf / log 目录中。比如：

```
52.MMSGLOG level = FAT,size = 20
```

MMSGLOG 控制登记文件 msg.log，这是由 deliver 和 submit 程序产生的；AUTHLOG 控制存于 auth.log 文件中的授权信息；MCHANLOG 则控制在文件



chan.log 中的大多数 MMDF 程序的登记。

如果采用登记范围最广的方式，就会产生大量的数据，因而降低了处理速度。最常用的设置的级别参数如下表：

表 4.4 级别参数

级别参数	描述
BST	登记基本统计数字
FAT	只登记致命错误
FST	登记所有统计数字
GEN	登记一般的诊断信息

尺寸参数对文件大小没有任何影响，因为没有对这一条特性的支持。如果登记文件不是周期性删除的，则登记文件会一直增大，直到磁盘空间满为止。

#### 步骤 5.9 增加额外的系统标识符

MSIG 行是 MMDF 用来把邮件传送故障通知邮件发送者的内容。这条消息应该指出是哪一个邮件传送系统对邮件故障负责。

MSIG “MMDF Mail System”

#### 步骤 5.10 本地传送的调整

如果希望 MMDF 不要将邮件发至以 /usr/spool/mail 目录中，而发至另外的目录或文件中，可以用以下关键字在 mmdftailor 文件的任何位置增加几行内容：

MDLVRDIR ""

MMBXNAME ".mailbox"

MMBXPROT 0600

如果 MDLVRDIR 为空，则 MMDF 将邮件发至用户的主目录下；如果 MMBXNAME 为空，则 MMDF 将用户的注册名作为邮箱文件的名称；MMBXPROT 设置邮箱文件的保护方式，使用的 4 位八进制数的约定与 chmod (C) 命令用来改变存取权限的 4 位八进制数完全一致。在上例中，MMDF 将邮件发送至用户的主目录下，邮箱文名为 .mailbox，该文件的保护方式确保只有该文件属主 (owner) 可以对其进行读/写操作。

#### 步骤 5.11 增加新的域表 (域文件)

在按前面讲的方法建立域表之后，在 MTBL 条目中标识出文件名，然后给此别名文件增加适合的 MDMN 条目。

MDMN 条目定义了 MMDF 的域，域是以某种方式相联系的机器的集合，联系方式可能是同一地理位置 (如 CAMFORD.AC.UK)，同一组织 (如 SCO.COM)，或同一性质 (如 OXBRIDGE.EDU)。域定义形式类似：

MDMN "SCO.COM",show = "Local Domain",table = locdom

```
MDMN "UUCP".show = "UUCP Domain".table = uudom
```

```
MDMN "" .show = "Root Domain".table = rootdom
```

第一个参数是域的名字。root 域定义中没有名字 ("")，因为 root 域表中可以包含许多不同域的条目。

“show”参数给了域一个更形象的名字，可作为某些程序的显示行，但这一特性没有被采用，所以“show”参数只是充当注释之用。“table”参数从 MTBL 定义中取出一个缩写名来指示域表。

MMDF 为每一个域查找最长的匹配项，例如，对发往 CAMFORD.AC.UK 的邮件，域表 AC.UK 在域表 UK 之前得到匹配，如果 MMDF 没有匹配成功，则查找部分匹配，并以此作为传送邮件的方向。比如，若邮件是发往 CAMFORD 的，但 MMDF 没有找到 CAMFORD 域表，则依照列举的顺序在域表中查找 CAMFORD 条目。MMDF 将邮件发往找到部分匹配的域。

如果 MMDF 在前面的域表中连部分匹配也找不到，则在 root 域中寻找匹配，以求将邮件发往较智能的机器。如果确实找不到任何匹配，则作为最后的措施，将邮件送至 badhosts 通道（如果该通道存在的话）。如果在前面的域中找到了合适的匹配，就选定该域作为邮件的目的机器对应域，而不再排在后面的域表中找匹配，所以排列 MDMN 定义的顺序十分重要。应保证本地域排在前头，root 域在最后。

#### 步骤 5.12 增加别名文件

本步骤假设你已照第 4 步的方法创建了一个新的别名文件。应该注意的是，在已有的别名文件中增加条目的办法比另外建立一个别名文件要容易得多。如果是对现存别名文件增加信息，即可跳过本步骤。也许希望研究是否有可能对 ALIAS 参数进行修改或替换，实际上缺省参数值对大多数需要来讲都已足够满足要求了。如果想改动别名文件，则需要重新建立 MMDF 数据库，这将在第 6 步“建立 MMDF 数据库”部分进行讨论。

如果建立了新的别名文件，则应将该文件放入 /usr/mmdf/table 目录中，保证它的属主和小组属主都是 mmdf，且权限方式值是 644。建议在设置权限方式时不要让别人有权对文件进行写操作。给予任何人以写别名文件的权利都会使邮件的传送产生混乱。

在 mmdftailor 文件中，每个 MTBL 行对应一个缩写名字和一个更能对含有新的别名表的文件进行描述的名字。缩写名字在 mmdftailor 文件的后面用来作为表示域文件的标记。

例如，将用户和机器映射起来的别名表在 mmdftailor 中定义为：

```
20 MTBL auser, file = "alias.user".show = "User Alias"
```

至此，在 mmdftailor 文件中即以 auser 来表示 /usr/mmdf/table/alias.user 文件。

ALIAS 条目定义了别名信息的不同“源”，用到了 MTBL 定义中指定的缩写名字。每个别名表均可用以下特性进行定义：

nobypass: 这一任选项确保对于发出去或收进来的邮件的所有地址都首先查找别名文件。在 MMF 中，在地址前面加上符号 (~) 表示该地址被解释为其字面上的意思。“nobypass”参数表示不能越过别名文件进行处理。比如，别名文件含有这样一个条目且没

有设置“nobypass”参数:

```
sam      sam@astoria
```

如果打入:

```
mail sam
```

则邮件被送至 astoria 机器上的 sam 帐号, 如果接着输入:

```
mail    “~sam”
```

则别名转换为无效, 邮件被送至本地机器的用户 Sam。如果设置了“nobypass”参数, 则在地址前面加上“~”起不了任何作用, 每个邮件请求都需要经别名文件进行转化。

public: 设置了“trusted”的别名文件可将邮件送往系统中任何用户(包括超级用户)上具有任何权限的文件或进程; 只有超级用户才能够对“trusted”别名文件进行修改。

这里给出几个标准的别名定义:

```
33.ALIAS table=lalias, trusted, nobypss
```

```
34.ALIAS table=auser
```

MMDF 查找别名表的顺序与列举顺序是一致的, 并使用首先匹配上的域表, 不再在以后的别名表中寻找其它的匹配。

### 步骤 5.13 增加新通道文件

本步骤假定的已从 SCO 附加软件包增加了一个新的通道程序。

MCHN 条目定义了可以被 MMDF 用于邮件传输的通道。通道是将邮件送至本地机器上的邮箱或者通过网络传至远程机器的一种机制。

至少有两种通道是必需的: 一个用于传送本地邮件, 另一个用于处理较大的邮件 Lists。你应根据配置在系统中的网络协议定义别的通道。通道的定义形式类似:

```
36 MCHN local, show““Local Delivery”, que=local,
```

```
37     tbl=local, ap=same, pgm=local, mod=imm
```

```
38 MCHN list, show“=List Processing”, que=List,
```

```
39     tbl=List, ap=same, pgm=list, mod=imm
```

```
     host=“SCO.COM”, Confstr=sender
```

```
40 MCHN uucp, show=“UUCP.Delivery”, que=uucp
```

```
41     tbl=uucpn, ap=822, pgm=uucp, mod=imm
```

```
42 MCHN micnet, show=“Micnet Delivery”, que=micnet
```

```
43     tbl=mmchn, ap=same, pgm=micnet, mod=imm
```

```
44 MCHN badhosts, show“=last-chance routing”.
```

```
45     que=badhosts, tbl=mnchn, ap=same, pgm=micnet,
```

```
     mod=imm, host=“sco.sco.COM”
```

MCHN 定义的顺序很重要, 因为 MMDF 就是以此顺序查找通道表的。

本例中的最后一个通道(badhosts)用在 submit (ADM) 程序不能正确识别出邮件发往的机器之时。这时, 由该通道将邮件向前传递至一个机器数据库更完整的机器上。badhosts 实际上并不是一个通道, 因为它没有自己的传输程序。在本例中这个“伪通道”使用 Micnet 通道将邮件送往一个具有更高智能的机器。如果 badhosts 通道不存在, 则发

往未知机器的邮件返回至发送者。

在通道定义中，头一个参数是通道的名字。用于定义这些通道的参数为（以字母顺序描述）：

ap: 选择发出邮件的邮件头地址解释类型。在下面的各类型中，“RFC”表示“Request for Comments”（请求注释），733 和 822 指的是两种对邮件地址进行破译解释的标准。这两种标准由 ARPnet 出版，可以从 NIC 获得。这些类型是：

733: converts to RFC733-style address（转换成 RFC733 型地址）

822: converts to RFC822 style address（转换成 RFC-822 型地址）

Same: doesnot reformat headers（不改变邮件头地址类型）

Confstr: 将一个指示通道的标志送至由通道运行的程序。List 通道用一个配置字符串使得 sender 方式有效，这样没有为邮件 list 定义 list-request 别名，消息的发送者被记录为给 list 通道发邮件的人。

host: 给出通道转发所有邮件至其上的智能机器名字。

mode: 设置通道的传输方式：

host 和“reg”参数相似，但要根据不同机器对队列进行归类（见下面对“reg”的说明）。SMTP 通道据此一直保持与远程机器的连接，直到所有要发往该机器的消息都已处理完毕为止。

imm 立即发送邮件

reg 将邮件放入队列，但不发送；必须运行 deliver 程序（或者手工做这一工作，或者作为后台程序做）实际地通过某一规定通道发送邮件（缺省通道）

pgm 指示在 /usr/mmdf/chans 目录中的、从 deliver (ADM) 程序中取出邮件并送至本地机器上的目的地或通过网络送往远程机器的程序。

que 指定在其中对本通道的消息进行排队的 /usr/spool/mmdf/lock/home 的子目录。

show 给出某些程序用来对通道作进一步说明的内容，通常指示通道的功能。

tbl 使用 MTBL 定义中的缩写名字来指定通道文件。

#### 步骤 5.14 改变消息参数

本步骤通过对消息参数的改动使 MMDF 系统变成可调整的系统。

MLISTSIZ 指示消息中地址的最大数目。如果地址数目超过该数，则 MMDF 不发“等待邮件”的警告信息，而只返回一条错误信息指明邮件发送失败。错误信息包含整个邮件头以及消息体的头几行信息。

MLISTSIZ 20

如果一条消息经过了某一数目的“hops”还未到达目的机器，则 MMDF 认为该邮件已处于“循环”状态并不再继续进行。MMAXHOPS 指示了这一数目的最大值。

MMAXHOPS 20

如果一条消息留在队列中的时间太长而没有发送，MMDF 则发一条警告信息给消息发送者表示“传输延迟”，MWARNTIME 参数即指示这一时间值，单位是小时。

## MWARNTIME 72

如果消息在队列中停留的时间到了某一数值而还没有得到发送，则 MMDF 给邮件发送者发一条消息指示邮件发送失败，消息从队列中清除。这一时间数值即为 MFAILTIME，单位为小时。

## MFAILTIME 144

如果队列中消息条数少于 MMAXSORT，则消息以进队列的时间排序，并依该顺序发送；如果消息条数多于 MMAXSORT，则消息在目录搜索中找到时才发送。

## MMAXSORT 150

MSLEEP 是后台传输程序在两次队列扫描之间 sleep 的时间长度，单位是秒。下面的设置使程序在两轮扫描之间 sleep 5 分钟：

## MSLEEP 300

### 第六步：建立 MMDF 数据库

哈希 (hashed) 数据库使 MMDF 可以很快地访问别名和邮件路由 (routing) 信息。无论何时改变别名文件或路由文件，都应改变哈希数据库。要重建该数据库，应作为 mmdf 注册，并从 /usr/mmdf/table 目录执行 dbmbuild (ADM) 程序：

```
cd /usr/mmdf/table
./dbmbuild
```

dbmbuild 程序使用在 /usr/mmdf/table 文件中的定义来建立哈希数据库，如果有文件丢失，该程序会报告。关于 dbmbuild (ADM) 程序的细节请参考该程序的使用说明

### 第七步：编辑 S86mmdf 文件

如果要使用 Local 通道之外的通道，这一步工作就是必需的。/etc/rc2.d/S86mmdf 文件确保 deliver (ADM) 一直都在运行。在已经安装好的配置中，此文件只保证监视 Local 通道。注意，在每次发送或接收邮件时都自动调用 deliver (ADM) 程序；S86mmdf 文件只是确保由于通道冲突使得邮件不能立即传送时 Local 通道的邮件不致被忽略。通过编辑 S86mmdf 文件，可以使 deliver 程序也监视其它通道，或者为每个通道建立一个独立的 deliver 命令。也可以为同一通道准备多个 deliver 调用。但对启动的每一个进程而言，系统开销增加了，并且 root 可以启动的进程总数受到了限制。

如果为每个通道建立独立的 deliver 命令，则可以用的一个选项设置不同的轮询次数，一次轮询次数为若干秒。这样你就可以每小时查询一次 UUCP (-T3600) 或每分钟轮询一次 SMTP，选择权在你。这有利于你的系统选择合适的通信速率，以及限制网络上交通量等。

另上，如果用一个 deliver 程序为多个通道服务，则 deliver 分别处理各个通道，并力图在转到下一个通道之前将当前通道队列中的所有消息全部传送完。

如果有多个 deliver 程序，则可以让每个程序服务于一个通道。这样各通道可并行工作，这常常是最理想的，至少对门通路机器来讲，增加了机器的总体邮件带宽。为此，对繁忙的通道，可以提供多个 deliver 程序给一个通道。（系统应提供锁定机制以防止多个程

序同时传输同一消息，因为这会导致同一信息的多个复制件得到传输)。

已安装好的 S86mmdf 文件包含如下一行：

```
/ bin / su mmdf -c "/usr/mmdf/bin/deliver -b -clocal"
```

为监视其它通道，应在 -clocal 之后增加其它通道的名字。比如，若想监视 UUCP 和 Micnet 通道，应使用如下命令：

```
/ bin / su / mmdf -c " /usr/mmdf/bin/deliver -b -clocal, UUCP, micnet"
```

类似地，要为每个通道建立独立的 deliver 进程，则为每个通道另外增加若干行。要使装有 UUCP 和 Micnet 通道的系统使用独立的 deliver 调用，应使用形如下列的命令：

```
/ bin / su mmdf / -c "/usr/mmdf/bin/deliver -b -clocal"
```

```
/ bin / su mmdf -c "/usr/mmdf/bin/deliver -b -cuucp -T3600"
```

```
/ bin / su mmdf -c "/usr/mmdf/bin/deliver -b -cmicnet"
```

#### 4.4 MMDF 系统的维护

cleanque (ADM) 命令从邮件队列中清除过期的文件。可以安排 cron (C) 程序执行每天 (也可更频繁，这决定于邮件量) 运行 cleanque (ADM) 的任务，也可以在怀疑邮件传递有问题的时候手工运行 cleanque 命令。

checkaddr (ADM) 程序用于对单个地址进行测试以保证正确性。本命令在建立新的别名文件之后测试别名时特别有用。

checkup (ADM) 命令检查本地机器上完整的 MMDF 系统。报告的信息列出了 mmdftailor 文件的许多设置情况。

checkque (ADM) 程序检查邮件队列的状况并表示有多少消息在等待传输。如果一个邮件队列溢出，应试着自己使用 deliver (ADM) 程序传送邮件：

```
deliver -w -clist, UUCP
```

-c 任选项指定要处理的通道，-w 任选项使得 deliver 程序和通道程序在想发送邮件时输出指示性消息。(deliver (ADM) 随后讨论)。在发生异常情况，比如发送拒绝或者接收拒绝时，即可看看这类输出。

deliver (ADM) 命令在每个通道的基础上控制消息的传输。其任选项为：

-b 在后台运行 deliver (ADM) 程序

-Tn 每隔几秒钟作一次传输尝试

-cchan 在指定通道上传输信息。如有多于一个的通道，则应以如下形式使用该任选项：

```
-c chan1, chan2 chan3.
```

-w 产生调试信息

-L file 使用指定文件而不是 /usr/mmdf/log/cha.log 文件作为登记文件。

deliver 在每次系统从 /etc/rc2.d/s86nndf 文件进入多用户时自动启动。

应周期性地检查 /usr/mmdf/log 中登记文件的大小。如果为了限制累计登记数据的数量而将一个登记文件移开时，应保证新的登记文件 (初始时空文件) 的属主是 mmdf。比如，可以执行以下几条命令来达到这一要求：

```
cd / usr / mmdf / log
mv chan.log chan.log~
> chan.log
```

#### 4.5 改变机器名

在系统的安装过程中，会为机器提供一个名字。在安装之后若想修改系统名称，应按如下步骤执行：

1.作为超级用户注册，用 `uname (c)` 的 `-S` 命令改变机器名。比如，要将机器名字改为“elgrande”，应打入如下命令：

```
uname -S elgrande
```

2.然后以 `mmdf` 注册，移至 `mmdf` 目录：

```
cd / usr / mmdf
```

3.编辑 `mmdftailor` 文件，并将所有原机器名出现的地方取代以新的名字。

4.编辑 `/usr/mmdf/table` 目录中的所有 `.dom` 和 `.chn` 文件，并将所有系统名出现的地方改成新的系统名。

5.输入下列命令：

```
cd / usr / mmdf / table
./dbmbuild
```

6.退出 `mmdf` 的注册，以超级用户身份注册，关闭系统进入单用户方式，然后启动系统至多用户方式。

7.如果你在使用 `TCP/IP`，应保证 `/etc/hosts` 文件中包含新的系统名。

#### 4.6 Internet 地址

如果在使用基于 `TCP/IP` 的通信网络，则连接到网络上的每台机器都需要一个唯一的 `Internet` 地址。即使你没有将 `TCP/IP` 用于网络通信，只要你想使用 `UUCP` 之外的最高级别域，你就必须至少有一个由 `NIC` 发布的 `Internet` 地址。

`Internet` 地址为 32 位长（4 字节），指定某一特定网络及该网络上的一台特定机器。`Internet` 地址以标准格式给出，分成 4 个域，域间以圆点（.）相隔，比如 `12.345.67.890`。4 个域中的每一个都表示完整 `Internet` 地址的一个字节。

`Internet` 地址可以以十进制，八进制或十六进制给出。八进制值以 0 开头，而十六进制以 `0x` 或 `0X` 在前。为了灵活地表示互连在一起的多个网络和一个网络中的多台机器，`Internet` 地址使用 4 个域中的一个，二个或三个域来指示网络。`Internet` 地址中没有用来指示网络的部分用来指定该网络中的特定机器。

根据连接到基于 `TCP/IP` 的网络上的机器的数目，有三种 `Internet` 地址形式可供选择。如果你要将最多 254 台机器（可以少些）连至网络上，则用 C 类地址；如果连接到网络上的机器数目多于 255 而为超过 64000 台，则可选 B 类 `Internet`；如果数目多于 64000 台，但不超过六百万，可用 A 类地址。`Internet` 地址的 4 个字节的分配与类别的关系见下表：

表 4.7 类别与字节分配表

地址类别	网络部分	机器部分	头 1 字节值
A	字节 1	字节 2-4	0--127
B	字节 1 和 2	字节 3 和 4	128--191
C	字节 1-3	字节 4	191--223

Internet 地址的“网络部分”唯一地标识出你的网络位置,“机器部分”唯一地标识出一个特定机器,地址的头一个字节值即能确定地址的类别。



# 第五部分 实用参考

## 第一章 解决系统问题

本章将全面讨论在 UNIX 系统管理过程可能遇到的常见问题。包括：

- 系统不能引导
- 控制台和控制台键盘不能工作
- 一般问题，包括对文件系统的修理和恢复
- 用户不能登录进入系统
- 在安装过程中产生的问题
- 运行mail时常见问题的解决
- 为拨出和拨入调制解调器提供的问题解决的信息
- 打印机和打印系统的常见问题
- 关于停止失控和杀不掉的进程
- 启动停止的调度程序
- 恢复其它系统故障，例如电源故障，系统紧急情况或硬盘上的坏道。
- 修理磁带机
- 解决终端问题，例如修理杂乱显示或解开被封锁的终端
- 解决UUCP系统的问题

### 1.1 解决引导问题

系统不能引导的情况可被分为两类：

- 系统在安装期间不能引导
- 系统不能引导，虽然以前曾引导成功过

这一节将简要地讲解系统在安装期间不能引导的一般原因。而对于在曾成功地引导过以后又不能引导将深入讨论。其中包括对关键的系统文件的恢复和特殊问题的解决，例如当系统在注册提示符状态下挂起该怎么办，以及系统为什么不能进入多用户状态。

#### 1.1.1 系统在安装期间不能引导

如果系统在安装期间第一次从软盘启动失败，那么某个东西可能坏了。

1. 如果在驱动器中的软盘不是 N01（引导）盘，那么系统就不会显示引导提示符。把软盘换为 N01 软盘，再给机器重新加电。

2. 如果在驱动器中的软盘是 N01，那么就要确认一下软盘插得是否正确，软盘驱动器的门是否关了。然后再给机器重新加电。

3. 如果对 N01 盘的读操作总是出错，那么就要找一张新的 N01 软盘。

如果系统还是不能引导，那么很可能是机器的硬件有问题。

1. 查看一下随机资料，看看系统是否有足够的 RAM（随机访问存储器）。计算机中必须要有“系统的安装”一章中所要求的内存容量。如果没有足够的 RAM，系统就不能引导。有关内存要求的内容请看“安装问题的解决”一节。

2. 如果系统有足够的 RAM，那么就检查一下母板上所插的其它板（总线卡）是否正确。

3. 如果系统还不能引导，那么可能有硬件故障。查看机器的随机资料并进行硬件测试。

### 1.1.2 系统在曾成功引导之后又不能引导

如果系统以前能够成功引导，而现在又不能引导，那么问题可能是由如下一些情况引起的：

- 软盘驱动器中有一张非引导盘。
- 关键的系统文件，例如 /boot 或 /unix，被损坏或丢失。有关如何从软盘启动并恢复这些文件的内容请看本章中“恢复丢失的或被损坏的系统文件”一节。
- 硬盘可能产生了一个坏磁道，破坏了系统引导所需的系统文件。有关如何恢复的内容请看本章中“对其它系统故障的修复”一节。
- 如果系统中没有 387 浮点协处理器，并且 /etc/emulator 文件又丢失了，系统也不能引导。如果确实是这个问题，请看本章中“不能装入浮点模拟器”一节。

### 1.1.3 恢复丢失或损坏的系统文件

在极少的情况下，一个或多个关键的系统文件被偶然修改或删除，至使系统不能正确引导和操作。在系统不能引导的情况下，为了访问系统就必须从软盘引导，这样才能从备份中恢复这些关键的文件。

为了引导和访问一个不能从硬盘引导的系统，必须按“系统的安装”一节中所说明的那样做一组“应急引导软盘”。这一组软盘包含引导软盘和根文件系统软盘。引导盘中包含三个引导和装入 unix 系统核心所需的文件：/boot、/etc/default/boot 和 /unix。根文件系统软盘中包含一组用以恢复系统的 unix 工具。

---

注：每个系统必须分别有各自的“应急引导软盘”，否则会对系统造成更大的破坏。

---

如果没有做这些软盘，就必须重新安装操作系统。在某些情况下，如果没有根文件系统的备份，也必须重新安装操作系统。

### 1.1.4 /boot 找不到

当给计算机加电后系统显示如下信息时，/boot 就是丢失了：

```
/boot not found
```

```
Stage 1 boot failure : error loading /boot
```

/boot 文件包含 boot (HW) 程序，它用于在机器上电之后装入和执行核心。

如果 /boot 丢失了，就需要使用如下过程从“应急引导软盘”引导系统，这样才能恢复文件：

1. 在驱动器中插入引导盘，并对机器重新加电。这样就从引导盘执行了初始引导。

2. 在引导提示符下输入

```
hd (40) unix
```

在从软盘引导了系统之后，这个命令从硬盘把核心装入。

3. 在提示符下输入根口令把系统引入单用户模式。

4. 使用如下命令安装软盘文件系统：

```
mount /dev/fd0/mnt
```

5. 当软盘在驱动器中时，在系统提示符下输入如下命令来恢复 /boot：

```
cp /mnt/boot /
```

这样就可以访问硬盘上的 /boot 文件了。

6. 在把软盘从驱动器中取出之前，要输入如下命令把软盘文件系统拆卸下来：

```
umount /mnt
```

7. 从驱动器中取出软盘并使用 haltsys (ADM) 关闭系统。

8. 在引导提示符下按回车键从硬盘重新引导系统。

### 1.1.5 unix 没有找到

如果开机后系统显示如下信息，那么就是 /unix 文件丢失了：

```
unix not found
```

/unix 文件包含 UNIX 核心。如果 /unix 丢了，可以从另外一个核心文件引导，例如 /unix.old、/etc/conf/cf.d/unix 或 /etc/conf/cf.d/unix.old，只要在引导提示符下指定这个文件的完整的路径名就可以。

如果在系统上没有其它核心文件，那么就使用如下过程从“应急引导盘”引导系统，这样就能够恢复 /unix：

1. 在驱动器中插入引导软盘并重新给机器加电。

2. 在引导提示符下输入

```
fd (64) unix root=hd (40) swap=hd (41) pipe=hd (40)
```

这就执行了初始引导，从引导软盘装入核心并从硬盘安装了根文件系统。

3. 使系统进入单用户模式。

4. 把软盘文件系统安装到 /mnt。（请看前面“/boot 找不到”中的第 4 步。）

5. 当软盘在驱动器中时，用如下命令恢复 /unix：

```
cp /mnt/unix /
```

这个命令把 /unix 核心文件从引导软盘拷贝到硬盘上。

6. 拆卸软盘文件系统。（请看前面“/boot 找不到”的第 6 步。）

7. 从驱动器中取出软盘并用 haltsys (ADM) 关闭系统。

8. 在引导提示符下按回车键从硬盘重新引导系统。

### 1.1.6 不能装入浮点模拟器

如果计算机没有 387 浮点协处理器芯片并且 /etc/emulator 文件也丢失或损坏，那么引导就会失败并显示如下信息：

WARNING: cannot load floating point emulator (如果 387 芯片存在, 核心就会在硬件识别引导信息中对它进行识别。)

如果引导失败并给出这条信息, 那么就使用如下过程来引导系统并恢复 /etc/emulator:

1. 把引导软盘插入驱动器并重新开机。
2. 在引导提示符下, 按回车键并在得到指示时插入根软盘。这就引导了系统并从软盘上安装了根文件系统。
3. 在软盘上工作, 使用如下的 mount (ADM) 命令把硬盘的根文件系统安装到 /mnt:

```
mount /dev/hd0root /mnt
```

如果 mount 失败, 请看本章中“解决文件系统问题”一节下的“文件系统安装失败”, 可找到有关使用 fsck (ADM) 检查硬盘的内容。

4. 把 /etc/emulator 从软盘的根文件系统拷贝到所安装的硬盘上:

```
cp /etc/emulator /mnt/etc/emulator
```

5. 卸下硬盘:

```
umount /mnt
```

6. 确认软盘还在驱动器中并用 haltsys (ADM) 重新引导系统。
7. 从驱动器中取出软盘并在引导提示符下按回车键从硬盘引导系统。

### 1.1.7 系统在引导时挂起

如果引导进程在显示“Kernel: i/o bufs”这条信息之后挂起, 那么就说明 /etc/init 文件从系统中丢失。/etc/init 文件包含 init (M) 程序。一旦开始执行, init 过程就会在系统上产生所有其它进程。没有 /etc/init 文件, 新的进程就不能开始执行。

使用如下过程来恢复 /etc/init:

1. 把引导盘插入软盘驱动器并重新开机。
2. 在引导提示符下按回车键并在得到指示时插入根软盘。
3. 安装硬盘根文件系统:

```
mount /dev/hd0root /mnt
```

如果 mount 失败, 请看本章中“解决文件系统问题”一节下的“文件系统安装失败”, 可找到有关使用 fsck (ADM) 检查硬盘的内容。

4. 把 /etc/init 文件从软盘上的根文件系统拷贝到所安装的硬盘上:

```
cp /etc/init /mnt/etc/init
```

5. 输入 umount /mnt 卸下硬盘。
6. 使软盘留在驱动器中, 用 haltsys (ADM) 重新引导系统。
7. 从驱动器中取出软盘并在引导提示符下按回车键从硬盘引导系统。

### 1.1.8 系统不能进入多用户模式

如果在按 Ctrl-D 之后系统不能进入多用户模式或在引导时显示如下信息, 就说明 /etc/inittab 文件丢失:

INIT: Cannot open /etc/inittab errno: 2

INIT: SINGLE USER MODE

/etc/inittab 中包含给 init 用的指令。当 inittab 丢失，init 就不能执行系统启动指令且系统不能进入多用户模式。当按了 <CTL>d，系统依然保持为单用户模式并显示上面的引导错误信息。

因为 /etc/inittab 中包含供系统其它部分参考的指令，所以在建立“应急引导软盘”组时就要往根文件系统软盘中写一个特殊的 /etc/inittab。因此，如果出现 /etc/inittab 丢失的问题对系统的恢复就比较复杂，而不是简单地把 /etc/inittab 从软盘文件系统拷贝到硬盘中去就可以。

要恢复 inittab，就必须先把 /etc/conf/cf.d/init.base 文件拷贝到 /etc/inittab，然后再重新连接核心。做这件事，使用如下过程：

1. 输入如下命令：

```
haltsys
```

2. 把引导软盘插入驱动器中。

3. 从“应急引导软盘”引导系统并安装根文件系统。（请看“系统在引导时挂起”的第 2 步。）

4. 把硬盘安装于 /mnt。（请看“系统在引导时挂起”的第 3 步。）

5. 在所安装的硬盘上，将 init.base 拷贝到 /etc/inittab：

```
cp /mnt/etc/conf/cf.d/init.base /mnt/etc/inittab
```

6. 卸下硬盘。（请看“系统在引导时挂起”的第 5 步。）

7. 使用 haltsys (ADM) 重新引导系统并在引导提示符下按回车键从硬盘引导。

8. 在提示符下输入根 (root) 的口令把系统引入单用户模式。

9. 调用如下命令重新连接核心：

```
cd /etc/conf/cf.d
```

```
./link__unix
```

这就建立了一个新的 /etc/inittab 文件。

10. 用 haltsys 重新引导系统使用新的核心和 /etc/inittab 文件。

### 1.1.9 /etc/bcheckrc 找不到

当在引导时系统显示如下信息，说明文件 /etc/bcheckrc 丢失：

```
INITSH: /etc/bcheckrc: not found
```

当系统引导时，init 就会按照 /etc/inittab 的指示执行 bcheckrc。如果需要，bcheckrc 将对根文件系统进行检查并修理。在引导系统时，文件 /etc/bcheckrc 应该在硬盘上。如果 /etc/bcheckrc 丢失了，就使用如下过程恢复它：

1. 把系统引入单用户模式。

2. 在系统上做任何事情之前使用 fsck 手工对根文件系统进行清理：

3. 一旦根文件系统被清理干净，就从备份中恢复 /etc/bcheckrc。

4. 在提示符下输入 haltsys 并重新引导系统。

### 1.1.10 /bin/sulogin 的 execlp 失败

如果在启动时系统显示如下信息并直接进入多用户模式（运行级 2），那么就表明 /bin/sulogin 文件从硬盘丢失：

```
INIT: execlp of /bin/sulogin failed; errno = 2
```

sulogin (ADM) 实用程序必须在系统中，用来访问单用户模式。如果这个文件丢失，那么就以根的身份登录并从根文件系统备份（而不是从“应急引导软盘”的根文件系统软盘）恢复文件 /bin/sulogin。

### 1.1.11 系统在注册提示符下挂起

如果系统正确地引导，但当进入多用户模式时在注册提示符下挂起，这表明文件 /bin/login 丢失了。/bin/login 文件包含 login (M) 程序。这个命令在每个终端对话区开始时运行，以允许用户访问系统。按如下步骤恢复 /bin/login：

1. 给机器重新上电并在引导提示符下按回车键。
2. 在提示符下，输入根口令而进入单用户模式。
3. 从根文件系统备份（而不是从“应急引导软盘”的根文件系统软盘）中恢复 /bin/sulogin 文件。

## 1.2 解决控制台问题

这一节将涉及系统的控制台可能出现的常见问题：

- 控制台键盘被封锁。
- 如何通过提供一个特殊的“patch”来防止键盘被封锁。
- 由于选择了错误的控制台键盘类型而使得控制台不认识键盘输入。
- 当系统在多用户模式下时，用户不能在控制台的多屏幕中登录。

### 1.2.1 控制台键盘被封锁

当系统不能对控制台键盘的输入产生反应时，这种情形就是所谓的“键盘被封锁”。控制台键盘被封锁只影响与视频显示适配器相连的键盘，而不影响连接于串行线上的标准终端。

如果出现了下面所讲的情况，那么可能是遇到了控制台键盘被封锁：

- 系统控制台键盘不能用来输入数据或执行任何任务。
- 不能使用 <Alt>-<F1> 到 <F12> 键来切换多屏幕，并且 <CapsLock> 键不能把 CapsLock 灯点亮或熄灭。
- 系统上的其它终端在继续工作。
- 打印机或其它设备在继续工作，并且系统依然运行。

在试图对被封锁的键盘进行修复之前，要确认：

- 未曾偶然按了 Ctrl-S（它使屏幕停止滚动）。要对此进行检查，就按一下 Ctrl-Q 并再看看是否能从键盘输入字符了。
- 如果计算机有一个 <Keyboard Lock> 键，而它不处于封锁位置。
- 键盘插的是正确的槽座。

· 系统本身仍然在运行。

查看一下终端，看它是否正在工作，是否能够执行系统任务，例如登录和检查日期。如果没有终端，那么就看看硬盘访问指示灯。如果它定时地闪烁，至少是每 30 秒钟闪一次，那么就说明系统仍在工作并且正在使用硬盘。

---

注：如果处于单用户模式，那么就不能使用其它终端并且硬盘访问指示灯也不会闪。

---

如果对上述所建议的都进行了检查而控制台键盘依然被锁着，那么就把它拔下来然后再重新插上。如果问题解决了，那么所遇到的情况就是一种有限的键盘锁定。如果这最后一步没有解决问题，那么键盘仍然是锁着的。

#### 防止控制台键盘被封锁

可以提供一个特殊的“补钉”来修改操作系统的核心而防止控制台键盘被封锁。（核心是操作系统的主要程序，它一直在内存中运行。）

---

注：因为这个补钉会使键盘指示灯失效，所以只有在试了其它方法而全无效之后再使用它。

---

使用这个补钉的过程如下：

1. 如果系统控制台没工作，就使其工作。必要时重新引导系统并使其进入单用户模式。

如果没有重新引导，那么就在系统控制台上以根的身份登录并使用 shutdown 命令关闭系统使其进入单用户模式：

```
/etc/shutdown su
```

(有关更多的内容，请看 shutdown (ADM)。)

2. 在系统处于单用户状态后，用如下命令备份核心：

```
cd /
```

```
mv unix unix.00
```

```
cp unix.00 unix
```

3. 现在，用如下命令给核心打补钉：

```
/etc/_fst-w/unix
```

```
ledspresent/w 0
```

```
$q
```

4. 使用 /etc/shutdown 关闭系统。

5. 当看到“Normal System Shutdown”信息之后，按任意键重新引导系统。现在就解决了键盘被封锁的问题了。

6. 最后，与维修部联系并报告问题。

### 1.2.2 错误的控制台类型

如果控制台键盘是一个 XT 或其它非 AT 键盘而操作系统又被配置为使用一个 AT 键盘，那么系统就无法识别从键盘来的输入数据。有关检测和切换键盘模式的内容请看附录“使用系统控制台和彩色显示”中的控制台键盘类型的选择一节。

### 1.2.3 不能在控制台上登录

如果在多用户模式下试图在控制台上登录，而系统又显示了如下错误信息：

cannot obtain database information on this terminal 那么就请参阅“系统安全性维护”一章中“与安全性有关的错误信息”一节。

## 1.3 解决文件系统的问题

这一节讲述如何释放 i 节点并当文件系统被用尽之后如何释放空间；当系统被非正常关闭而数据被破坏之后如何修理文件系统；当 mount (ADM) 失败时如何检查并修理文件系统；并将解释为什么在目录的 sticky bit (密封位) 被设置之后就不能从这个目录删除文件了。

### 1.3.1 文件系统的 i 节点被用尽

当一个文件系统的 i 节点被用尽时，系统将显示如下错误信息：

kernel: WARNING: Out of inode on dev nn / mm

其中 nn / mm 是用尽了 i 节点的文件系统。要解决这个问题：

1. 从文件系统中删去了不必要的 (老的、临时的、core 或记录文件) 的文件。
2. 使用带 -size 参数的 find (C) 命令来查看一下在文件系统中是否有大量的小文件。

一个文件系统中可用的 i 节点的数目是在文件系统建立时 (使用 mkfs (ADM)) 确定的。如果文件系统的 i 节点总是被用尽，那么就on应该重新配置这个文件系统并增加 i 节点的数目。要做这件事：

1. 使用如下的 sysadmsh 命令备份文件系统：

Backups → Create → Unscheduled

2. 使用 sysadmsh 选项核实备份的完整性：

Backups → Integrity

3. 以命令行方式运行 mkfs 并为这个文件系统指定更多的 i 节点。例如，要把文件系统 /dev / u 的 i 节点数重新配置为 6400，就使用如下的 mkfs 命令：

mkfs /dev / u 20000: 6400 9 400

更多的内容，请看 mkfs (ADM)。

4. 用如下 sysadmsh 选项从备份中恢复这个文件系统：

Backups → Restore → Full

### 1.3.2 文件系统空间被用尽

当文件系统只有很少或根本没有空间用来工作时，系统会显示如下信息：

No space left on device: nn / mm

当文件系统的空间被用尽时，系统就会停止所有对这个文件系统的写操作。恢复系统操作的唯一办法就是从这个文件系统中删去或减少文件。

使用如下办法来恢复文件系统的空间：

1. 使用 wall (ADM) 命令向全系统发布消息，让用户删去不必要的文件。



2. 用 du (C) 命令检查这个文件系统中各个文件和目录所用的块数。例如, 要显示 /u 文件系统中的文件的块数, 就在 /u 中使用如下命令:

```
du |sort -nr
```

这个命令把文件按大小排序并先显示大文件。然后可以发送一个邮件, 让这些大文件的用户把它们删除。

3. 使用 find (C) 命令查找特别大的或老的目录和文件, 并给其主人发送一个邮件, 让他们删除不必要的文件。

4. 用 find 查找和删除临时文件和 core 文件。另外, 还可以修改 cleantmp (ADM) 来定义多长时间对关键目录 (缺省设置为 /tmp) 中的文件清除一次。

5. 清除系统记录文件中的内容, 例如 /usr/spool/netmail/smtpc.log, /usr/spool/lp/lgo, /usr/adm, 特别是 /usr/adm/messages 和 /usr/adm/sulog (如果有), /usr/preserve, 任何不用的 a.out 文件, 在 lost+found 目录中的文件, 以及附加程序的记录文件。

要做这件事, 用如下结构:

```
> filename
```

6. 如果使用了安全性审计, 那么检查一下目录 /tcb/audittmp/audit\* 中的磁盘使用状况。使用 sysadmsh 选项备份并删除老的审计文件:

```
System → Audit → Files → Backup
```

```
System → Audit → Files → Delete
```

管理员应该定期检查并删除这些文件。

7. 按本章中“减少磁盘碎片空间”一节中的步骤来减少磁盘碎片空间。

8. 如果系统经常缺少空间, 那么就建立并安装另外一个文件系统

这里所讲的每个步骤都在“文件系统的维护”一章中维护空闲空间一节中有详细说明。

### 1.3.3 检查文件系统的自由空间

要想防止文件系统的空间被用尽, 就应该定期使用 sysadmsh 选项来对各文件系统的自由空间进行检查:

```
System → Report → Disk
```

这个命令以 512 字节的块为单位显示出所剩空间的大小。

### 1.3.4 减少磁盘碎片空间

如果磁盘已经用了一段时间, 多次建立和删除文件就会产生一种情况, 叫做“磁盘碎片空间”。这表示文件系统中的文件被写到分布于硬盘各处的小空间片中。这就会使得磁盘的 I/O 性能很差。要想减少磁盘的碎片空间, 首先要对文件系统中的所有文件进行备份:

1. 调用 sysadmsh 并选择:

```
Backups → Create → Unscheduled
```

2. 选择文件系统和介质设备 (块的大小自动选定) 以及格式化过的软磁盘, 如果需要。

3. 在选定的驱动设备中装入一卷，磁带或磁盘，并按回车键。系统就把文件拷贝到驱动设备中。

4. 然后，对备份的完整性进行核实。调用 `sysadmsh` 并选择：

Backups → Integrity

输入介质类型并依次插入备份的各卷。

5. 现在，从文件系统中删去所有的文件。例如，要从 `/u` 文件系统中删去文件，就要在提示符下输入如下命令：

```
cd /u
```

```
rm -rf *
```

6. 最后，从备份中恢复文件。调用 `sysadmsh` 并选择：

Backups → Restore → Full

插入第一卷并输入要恢复的文件系统名和介质驱动设备名。这时就显示出真正的 `cpio` 命令行。

因为文件又被完全重新写到了磁盘上，每个文件被写到一块里，这样碎片空间就减少了。这样就恢复了少量空间（一般为 5-10%）。对于一个常用的系统应该一年做一次减少碎片空间的工作，而对于不太常用的系统，做这个工作的周期可以长一些。在做这个工作之前，必须确认一下自己是否有完整、准确并且可读的备份，否则会丢失文件的。

### 1.3.5 恢复被破坏的根文件系统

如果根文件系统被严重破坏了，以至于在引导系统时，`fsck (ADM)` 不能运行，那么就使用如下过程来恢复系统：

1. 在驱动器中插入引导软盘并重新开机。

2. 在引导提示符下按回车键。在提示符下插入根文件系统软盘。这样就从软盘引导了系统并安装了根文件系统。

3. 在系统提示符下输入：

```
/etc/fsck -y /dev/hd0root
```

这时会看到一些信息，指示 `fsck` 正在进行系统清理的五或六个阶段。如果 `fsck` 在几秒钟内就退出了或者系统显示出没有意义的信息，那么必须从备份恢复整个根文件系统，这里有两个无意义信息的例子：

```
UNKNOWN FILE SYSTEM VERSION 65535
```

```
CLEANING NON SYSTEM 3 FILESYSTEM
```

（再请看下一节 `fsck` 在检查文件系统大小时停止）

如果 `fsck` 成功，就用 `haltsys (ADM)` 关闭系统并按回车键从硬盘引导。如果 `fsck` 不成功，继续按如下步骤恢复根文件系统：

4. 安装硬盘根文件系统：

```
/etc/mount /dev/hd0root /mnt
```

如果系统显示如下信息，就必须重新安装操作系统：

```
mount: cannot access special file
```

5. 如果成功地安装了硬盘，那么现在就可以从备份恢复根文件系统。

在系统提示符下输入如下命令：

```
cd /mnt
```

```
cpio -iudv -1 /dev /devicename C 512
```

devicename 是读备份所用设备的设备名。例如，某一个盒式磁带机的设备名为 /dev/rct0，小型磁带机的设备名为 /dev/rctmini。96tpi 软盘驱动器的设备名为 /dev/fd096

6. 恢复完成之后，使用 umount 卸下这个文件系统，并用 haltsys (ADM) 关闭系统。

7. 在引导提示符下按回车键从硬盘引导系统。现在应该能够用如下 sysadmsh 命令来恢复其它文件系统了：

```
Backup->Restore
```

如果这时还不能从硬盘引导系统，那么可能必须从头安装操作系统。

### 1.3.6 fsck 在检查文件系统大小时停止

如果一个文件系统的超级块在非正常关机或发生硬件故障之后被严重损坏，以至于 fsck (ADM) 在大小检查时就退出或报告了很多错误信息，那么可以使用 fsdb (ADM) 用手工来修补它的超级块。

fsck 从超级块中读出这个文件系统的大小。如果保存在超级块中的 FSIZE (这是文件系统的总块数) 和 ISIZE (分配于这个文件系统中的 i 节点数) 的值对于这个文件系统来说不是正常值，那么 fsck 就会显示出大小检查的结果并退出，下面是一个例子：

```
#fsck /dev /root
/dev /root
/dev /root File System: /volume: root
size check: FSIZE 0 ISIZE 0
#
```

当使用 fsdb 时，只要修改超级块中的这两个值中的一个就行了，FSIZE 或 ISIZE，把它们改回文件系统的正常值。

---

注：如果不能引导系统，那么就不能在根文件系统中运行 fsdb。

---

因为 fsdb 是一个允许直接修改超级块 (其中包含有关文件系统的重要信息) 的功能很强的工具，所以在使用这个工具时必须非常小心。如果用 fsdb 输入了不正确的值，那么可能会完全破坏系统并丢失所有的数据。在 fsdb (ADM) 的说明中讲述了多种用 fsdb 观察和操作超级块的办法。这种高级功能的使用只能提供给有经验的系统管理员。

---

注：在使用 fsdb 修复一个文件系统时必须先把这个文件系统卸下。

---

要想进行上面所讲的修复工作，必须按照如下过程：

1. 输入如下命令：

```
divvy -b l -c l
```

因为 divvy 文件系统表不位于超级块中，所以表中的信息可能是正确的。从 divvy 表中取得想恢复的文件系统的第一个块号和最后一个块号。

2. 要获得 FSIZE 的正确值, 应利用如下公式:

$FSIZE = \text{“最后一个块号”} - \text{“第一个块号”} + 1$

3. 要获得正确的 ISIZE, 就运行 bc (C), 并输入如下命令:

$((FSIZE * .25) - ((FSIZE * .25) \% 16))$

其中 FSIZE 是第 2 步计算所得的值。把小数部分去掉并记录这个值作为 ISIZE 供以后使用。输入 quit 退出 bc (C)。

4. 要想把所给的 i 节点的数目转换为分配给这个节点的第一个存储块的实际地址, 还必须做另外的计算。把这个数记为 isize 以区别于在 fsdb 开始时所显示的 ISIZE 值。要做这件事, 应做如下计算:

$isize = (ISIZE) / 16 + 2$

其中 ISIZE 是在第三步中计算获得的数。

5. 使用 FSIZE、ISIZE 和 isize 值, 现在可以利用 fsdb (ADM) 来修正超级块中的值了。输入:

fsdb / dev / filesystem

其中 filesystem 是要修复的文件系统的名字。

6. 如果通过计算所得到的值与真正的值很相近, fsdb 将有如下显示:

FSIZE = 52985, ISIZE = 13232

如果匹配得很好, 那么就不必再修改那个参数了。如果两个值看起来都正确, 但还有别的东西不对, 那么就必须从备份重新恢复数据。

---

注: 可以按 INTERRUPT、<DEL> 或 <CTL>c 在任何时候停止地址的显示。

---

7. 确定必须修改哪个参数 (或两个) ISIZE 或 FSIZE。

8. 输入如下命令 (对于 XENIX 或 UNIX 和 AFS 文件系统), 用计算得到的值替换 FSIZE 和 isize。如果 isize 正确, 那么只要按回车键就可以了。

---

注: 每次按回车键, fsdb 就会显示这个文件系统所认为的 FSIZE 和 isize 的正确值。

---

要修复 XENIX 文件系统, 应输入如下命令:

1024 < Return >

002000: 000000 (0)

= isize < Return >

002000: 000xxx

< Return >

002002: 000000 (0)

= FSIZE < Return >

002002: 000yyy

q

要修复 UNIX 或 AFS 文件系统:

512 < Return >

001000: 000000 (0)

= isize < Return >

001000: 000xxx

```
< Return >
001002:    000000 (0)
< Return >
001004:    000000 (0)
= FSIZE < Return >
001004:    000yyy
```

q

到此，就可以在这个文件系统上运行 fsck 并能够正确恢复系统。

---

注：目前，还不能用 fsdb 来修复 DOS 文件系统。

---

### 1.3.7 文件系统安装失败

如果 mount (ADM) 命令失败，那么要在再次运行 mount 之前检查文件系统：

```
/ etc / fsck -y / dev / filesystem
```

其中 filesystem 是要检查和修复的文件系统的名字。

### 1.3.8 不能删除文件

如果用户不能从目录中删除文件，那么一定是那个目录设置了密封位 (sticky bit)。密封位是目录保护设置，它只允许那个文件的所有者 (或根) 从那个目录中删去文件，无论那些文件的权限如何设置。

---

注：如果某文件的访问权限为可写，那么在这个目录中的文件内容可以被修改或删除。

---

只有根 (root) 可以设置密封位；只有根 (root) 和这个目录的所有者可以取消密封位。

要确定某个目录是否设置了密封位，可以以长格式列出其父目录 (密封位在权限表的最后一个域显示为“t”)。下面是一个设置了密封位的目录。

```
drwxrwxrwt 5 sys sys 2432 Jan 17 16:58 tmp
```

要想从目录中取消密封位，可输入如下命令：

```
strip directory-name
```

这时用户应该能够从目录中删除文件。

## 1.4 解决安装中的问题

这一节将讨论一些在操作系统的初始安装或后续安装期间遇到的问题，还将讨论最小的内存需求和如何重新启动安装。

### 1.4.1 divvy: 在 /mnt 上安装失败

如果安装失败并且系统显示如下信息，那么可能是系统内存不够：

```
divvy: mount on /mnt failed
```

```
invalid argument
```

```
cannot set up /dev directory on new device
```

UNIX 需要的最小内存已在“系统的安装”一章中加以说明。另外，如果使用的是最

小内存配置，那么内存必须能够完全被识别。在安装期间，这些内存中的大部分被用作 RAM 替换设备。如果内存不够，那么在文件系统生成期间系统的替换空间就会溢出，以至于安装失败。

如果机器为最小内存配置，那么就要核实一下操作系统是否能够识别所安装的所有内存。要做这件事，应检查引导屏幕信息。在硬件识别信息之后，系统会显示如下：

```
mem: total = 7808k, kernel = 2468k, user = 5340k
```

在某些 386 机器上，UNIX 核心不能识别在 640K 与 1024K 之间的内存，因为硬件制造厂家把这 384K 内存映射给他们的固件（firmware）使用。因为位置不同，UNIX 内核就不知道去哪找它。

要安装系统并避免 divvy 错误，就必须加上足够的内存以使得 UNIX 核心能够识别所需的最少内存。我们特别建议使用 32 位内存机器。

可使用如下两种方法增加内存：

- 如果主板上只有 640K 内存，那么就买一个扩展板并使其容量大于 1024K。这是最好的解决办法，因为没有一点内存浪费。
- 如果主板上已经有两兆字节的内存，并且还有另外的空插槽，那么至少再加 384K RAM 到主板上。如果主板没有空的插槽，那么就必须买一个扩展板装上。

#### 1.4.2 从 N01 引导而使用硬盘

如果试图重新安装并从 N01 软盘引导但又要使用硬盘的根，那么可以在引导提示符下通过输入 restart 而重新启动安装进程。

### 1.5 解决注册问题

这一节将讨论当遇到如下情况应该如何处理：

- 在系统进入多用户模式后管理员或用户无法在控制台上登录于多屏幕
- 用户不能在系统上登录
- 当用户想登录时系统显示“Login Incorrect”
- 用户帐号被封锁
- 用户忘记了口令

#### 1.5.1 在进入多用户模式后不能登录

如果在进入多用户模式后，不能够在控制台上登录，即使作为根（root）也不行，并且系统显示如下错误信息：

```
Cannot obtain database information on this terminal
```

就请参阅“系统安全性维护”一章中的“与安全性相关的错误信息”一节，找到这条错误信息的说明。

#### 1.5.2 用户不能在系统上登录

当 /etc/group 文件丢失时，用户就无法登录并且系统在用户终端上显示如下信息：

Can't rewrite terminal control entry for tty01.

Authentication error; See Account Administrator

如下信息也在引导时显示:

/bin/su: cannot setgid to auth, no auth entry

请参阅“系统安全性维护”一章中的“与安全性相关的错误信息”一节，找到这条错误信息的说明。

### 1.5.3 注册不正确

如果当用户试图登录时系统显示错误信息“Login Incorrect”，那么可能存在某种错误。请参阅“系统安全性维护”一章中的“与安全性相关的错误信息”一节，找到这条错误信息的说明。

### 1.5.4 打开被封锁的用户帐号

如果用户帐号被系统管理员有意封闭，那么系统就封锁了这个帐号，原因可能是用户超过了不成功注册次数，或是用户口令已经过期，这时当用户想在这个帐号上登录时系统会显示如下信息:

Account is disabled -- see Account Administrator

要想打开这个帐号，应调用 sysadmsh 并选择:

Accounts → User → Examine: Logins

把“Lock status”域改为 Clear all locks。更多的内容，请看“用户帐号的管理”一章的“封锁或打开用户帐号”一节。

### 1.5.5 替换被忘记的用户口令

这个系统设有提供查询一个所存在的口令的办法。如果用户忘记了口令，系统管理员就必须把口令改为另一新口令。

要做这件事，应按如下步骤:

1. 调用 sysadmsh 并做如下选择:

Accounts → User → Examin: Password

2. 从当前口令状态的选择中选取 Change。

3. 从确认改变表格中选取 Yes 来选择一个新口令。这就调用了口令修改过程。

4. 选 1 (选取一个口令) 或 2 (让系统生成一个口令) 并且输入或选择一个新口令。

5. 按回车键返回到修改用户帐号表格中。

## 1.6 解决邮件故障问题

因为 MMDF 系统是为本地系统而配置的，所以用户在处理本地邮件传送时应该不会遇到什么问题。一般来讲，如果邮件传送不正常，则要么是网络出了问题，要么是 MMDF 配置不正常。

邮件传送发生故障之时，应首先检验网络是否工作正常。为此，对相应网络（例如 UUCP 或者 TCP/IP）进行测试。

如果 UUCP 工作正常，而在使用 mail 命令时仍遇到问题，则可能是 MMDF 配置或 deliver 命令不正常。下面分几个部分讨论可能出现的问题及解决办法。

### 1.6.1 “邮件发送失败”错

如果系统马上将发出的邮件返回，并在提示行出现指示信息“Failed Mail”，则一定是 /usr/mmdf/table 目录中的通道文件（\*.chn）或者域文件（\*.cbm）出了问题。用下面的过程进行检查：

1. 以 mmdf 用户身份注册
2. 改变目录进入 /usr/mmdf/bin
3. 以使用的邮件地址为参数启动 checkaddr (ADM) 命令。比如，若命令 mail sylvia@seatt'e 失败，就应打入：

```
./checkaddr sylvia@seattle
```

4. 如果 checkaddr 命令显示如下的消息，则必然是机器或域不在通道文件或域文件中，或者是 /usr/mmdf/mmdftailor 文件包含一个错误：

```
Unknown Host / Domain
```

如果 checkaddr 显示“OK”，则机器和域都在通道文件和域文件中，但配置不正确。

5. 用“建立电子邮件”一章的内容，检验域、通道以及 /usr/mmdf/mmdftailor 文件的建立是否正确。

### 1.6.2 Mail 不能工作，但没有返回邮件

如果邮件没有到达目的地，但 MMDF 没有将未送出的邮件返回，则或者是 deliver 运行不正确，或者是 /usr/mmdf/mmdftailor 文件包含错误。deliver 控制程序是在系统启动时由 /etc/rc2.d 目录中的 script 启动的，这一进程为 mmdf 所有。用 ps-ef 命令检查 deliver 是否可以运行。如果 deliver 没有运行，则应执行下列过程：

1. 作为 mmdf 用户注册
2. 用下列命令手工启动 deliver 控制程序：

```
cd /usr/mmdf/bin
```

```
./deliver -c channelname (通道名) -b -T60
```

其中 channelname 是想要检查的通道（比如 UUCP 或者 TCP/IP）。这一命令指定通道启动 deliver 的执行，每 60 秒钟在后台对邮件队列进行一次扫描。

3. 如果手工运行 deliver 不成功，则可用下列命令检查邮件队列：

```
./checkque -c channelname
```

这一命令显示在本地通道的邮件队列中尚未发送的消息。

4. 如果队列中有消息，则再次运行 deliver。
5. 如果手工运行 deliver 还是不成功，则应检查 /usr/mmdf/table 中的文件和通道文件的配置情况以及 /usr/mmdf/mmdftailor 文件。

### 1.6.3 Mail 命令挂起

如果在用户想读邮件时，mail 命令挂起，则应作以下的检查：



1. 确定该用户的 mailbox 文件在 /usr/spool/mail 目录中存在。除非 postmaster 对系统进行了特别的配置，每个使用 mail 命令的用户都对应有一个文件存在于 /usr/spool/mail 目录中充当该用户的邮箱。

2. 确保 /usr/spool/mail 目录的属主和小组标识符 ID 都是 mmdf。

#### 1.6.4 MMDF 系统名的不一致性

如果在安装过程中没有改变系统名，则 MMDF 邮件系统识别的系统名与 uname (C) 命令报告的系统名及 /etc/systemid 文件包含的系统名都不相同。MMDF 邮件系统使邮件地址为 user@unix.UUCP，而 uname (C)，/etc/systemid，以及注册提示报告的系统名均为 scosysv，这在建立如 UUCP，Micnet 和 TCP/IP 的网络时即带来不一致性。

要解决这一问题，应改变系统名：

1. 选择一个新的系统名

2. 作为 root 注册

3. 在提示符下输入下列命令：

```
uname -s newname (新名字)
```

4. 作为 mmdf 用户注册

5. 编辑 /usr/mmdf/mmdftailor 文件，将出现“unix”字样的地方全改为新名字

6. 编辑 /usr/mmdf/table/\*.dom 和 /usr/mmdf/table/\*.chn 文件，将所有出现“unix”字样的地方全代以新的系统名。

7. 进入 /usr/mmdf/table 目录并打入：

```
./dbmbuild
```

8. 重新引导系统。

至此系统名已设置正确。

## 1.7 使用调制解调器可能出现的问题

虽然本节中所给出的例子都假定调制解调器直接连至 COM1，但实际上除 COM1 之外的串行端口也会经常用到。如果出现问题，首先应检查电话插座是否插好以及电话线路中是否有拨号声。

### 1.7.1 在拨号输出时出错

本部分内容讨论在调制解调器上拨号输出时可能出现的错误及解决办法。

没有 OK 信息：

在用下列命令测试调制解调器的连接时，得到“连接好”的信息：

```
cu -s 1200 -l ttyla dir
```

可是，当打入 AT 时，系统不显示“OK”信息。用以下步骤可解决这一问题：

1. 确信调制解调器开关和软件设置是正确的。

2. 检查调制解调器电缆：

• 如果所用的是“直通” (straight-through) 电缆，则用 null 调制解调器电缆试一试，

至少使用引腿 2, 3, 7, 8 和 20。

- 在发出cu命令之后, 观察调制解调器上的灯并回车若干次。在按回车键时“接收”灯应闪亮。如果灯不闪, 则检查电缆的引腿 2 是否连接正确(引腿 2 是从串行端口至调制解调器的数据传输线)。
  - 如果调制解调器上的“发送”灯闪亮, 则可能是本地响应被关掉了。应用ATE1命令将调制解调器的响应功能打开。
3. 检查计算机上的串行端口是否正常:
    - 将调制解调器连至计算机的另一个串行端口, 或者将终端或一台串行打印机连到本端口, 看能否正常工作。
    - 如果是端口不能工作了, 则查阅硬件文献以便修理。

4. 如果上述步骤不能解决问题则这个调制解调器也许不能用了, 应考虑进行修理或更换。

#### **调制解调器可以拨号, 但不能连接:**

如果调制解调器拨号正常, 但访问不能连接上, 则应做如下检查:

1. 检查电话号码是否正确以及调制解调器所附着的电话线是否正常。为此, 从电话线上取下调制解调器并插入一个普通电话中。手动拨号, 看在线路另一端的调制解调器是否能正确响应。

2. 在拨号访问时仔细听听调制解调器的动静。有些商用电话系统要求在某些数字之间有一段暂停时间。在 cu 命令中用一个 (-) 号即表示 2 秒的暂停。如在下面的例子中, 调制解调器在拨完第一个数字之后要暂停 8 秒:

```
9----458--1234
```

拨号程序将传给 cu 命令的 (-) 号转换成一定的调制解调器代码。比如, 拨号程序在将 (-) 号发往 Hayes 兼容的调制解调器之前要将其转换成逗号(,)。

#### **连接失败; 没有可用设备:**

当想从调制解调器拨号输出时, 显示下列信息:

```
Connect failed: No DEVICES AVAILABLE
```

要解决这一问题, 应按如下步骤执行:

1. 检查调制解调器口在 /usr/lib/uucp/Devices 文件中是否有一条目。下面给出一个在 /dev/tty1A 上波特率为 2400 的 Hayes 兼容的调制解调器条目的例子:

```
Direct tty1A -2400 direct
```

```
ACU tty1A -300 -2400 /usr/lib/uucp/dialHA24
```

应注意这两行内容在 Devices 文件中不能以“#”号开头, 否则即被理解为注释行。

2. 检查调制解调器口在 Devices 文件中是否有正确的波特率与之对应。如果用 -s 选项对 cu 指定波特率, 则应保证在 Devices 文件中有一条目与该波特率相对应。

#### **调制解调器应答, 但终端显示不正常:**

如果调制解调器能应答, 但终端显示字符不正常, 则应使用以下步骤:

1. 检查呼叫对象的数据位数和奇偶校验情况设置与本系统是否一致。cu 的缺省情况用了 8 个数据位, 没有校验位。要将此设置改为 7 位数据位以及偶校验, 应打入命令 cu-e, 若要使用 7 位数据位带奇校验的设置, 应输入命令 cu-o。

2. 检查远程计算机的波特率设置与本系统是否一致。

如果正在拨号进入另一 UNIX，可以迫使远程机器将波特率降低一级，办法是发出一条“break”信号。一般来说都应从最高波特率出发并根据需要逐渐降低。在注册序列时若想发送“break”信号，应打入：

```
~%b
```

3. 检查电话线上的声音。声音在波特率达 2400 或更高时变得很尖锐。正常情况下，如果线路的声音有问题，那么在屏幕上出现的字符也不会正常，即使线路另一端正在发送正确的数据也是如此。

#### 调制解调器不挂断

如果调制解调器在呼叫结束之后也不能挂断，则应做以下检查：

1. 应确保使用的是在 /usr/lib/uucp/Devices 文件中配置的调制解调器控制端口。如果用的是非调制解调器控制端口，应将其改为相应的调制解调器控制端口。比如，对应 tty1a 的调制解调器控制端口是 tty1A。

---

注：非调制解调器控制端口只应该用在终端上。

---

2. 如果呼叫的连接解除之后调制解调器上的 CD（载波检测）灯还在闪亮，则应检查调制解调器开关，看调制解调器是否设置为检测进来的载波的状态。如果调制解调器是 Hayes 2400 或其兼容者，可使用 AT&C1 命令，使载波检测线路根据载波线路的情况改变状态。

3. 检测调制解调器开关，看调制解调器是否设置为检测 DTR（数据终端准备好）的状态。如果调制解调器是 Hayes 2400 或与其兼容者，可以使用命令 AT&D2。

4. 有的调制解调器有一开关，如果设置了即会忽略 DTR 信号，应保证该开关是关闭的。

#### 双重回应

在从调制解调器上拨号输出时，如果得到双重回应，则应检查本地回应的设置状态，如果该设置有效，应将其置为无效。

### 1.7.2 在改号输入时出错

本部分内容为在从调制解调器上拨号输入时可能出现的错误提供解决办法。

调制解调器不能回答电话

如果调制解调器不能回答电话，则应作如下检查：

1. 确保调制解调器控制端口有效。要使调制解调器端口有效，应输入以下命令：

```
disable /dev/tty1a
```

```
enable /dev/tty1A
```

2. 确保调制解调器配置为自动回答方式。检查调制解调器的开关。如果调制解调器是 Hayes 2400 型，则可打入：

```
cu -ltty1a dir
```

然后，使用 ATSO=1 命令告诉调制解调器在第一次铃响时即回答电话。

3. 检查 DTR（数据终端准备好）线是否从计算机连接到了调制解调器。特别是保证

第 20 腿连接好。调制解调器通信时会用到第 2, 3, 7, 8 及 20 腿。

#### 调制解调器响应, 但又挂断

如果调制解调器回答了, 但在连接上之后立即挂断, 这时应做如下检查:

1. 如果调制解调器设置为自动响应和检测 DTR 的方式, 则应检查 DTR 线是否连接好。

2. 确保调制解调器控制端口有效:

```
disable / dev / tty1a
```

```
enable / / dev / tty1A
```

3. 检查电缆是否正常。如果使用的是“直通”(straight-through) 电缆, 至少应连接第 2, 3, 7, 8 及 20 腿, 特别是注意把第 20 腿连接好。

#### Garbage 或电缆松

如果在一次呼叫进入到调制解调器时控制台显示类似下面的提示:

```
Garbage or Loose cable on / dev / tty1A, Port shut down
```

则应做如下检查:

1. 确保调制解调器没有设置为回应数据或者发送命令响应的方式。如果调制解调器没有以此种方式建立, 则也许会发送“RING”信号来指示你正在呼叫的电话响铃了。由于 CD 信号未被激活, getty 将此信号解释为串行线路上的随机数据。对 Hayes 2400 调制解调器而言, 正确的命令应为 ATE0Q1。

2. 如果使用的是内部调制解调器, 而且以上的办法不能消除错误提示, 则可能是因为调制解调器不兼容的缘故, 应将其替换为一个标准的 Hayes 兼容型号。

#### 调制解调器回答了, 但没有注释提示符

如果调制解调器回答了, 但没有显示注册提示符, 则应作以下检查:

1. 确保在调制解调器回答电话之后, CD 线为调制解调器所确认。检查调制解调器上的开关。如果所用的是 Hayes 2400 或与其兼容的调制解调器, 则可用 AT&C1 命令。

2. 确保端口是有效的。输入以下命令即可使端口有效:

```
enable / dev / tty1A
```

3. 确保调制解调器用的是正确的 / etc / gettydefs 条目并且选择的波特率也很合适。/ etc / inittab 文件中的调制解调器端口设备行应该具有如下形式:

```
tlA: 2: respawn: / etc / getty tty1A m
```

此行中的最后字符是 / etc / gettydefs 文件中条目的指针。确保 / etc / gettydefs 文件中的该条目是正确的。

#### 屏幕显示一串注册提示符

如果在注册时屏幕不可控制地滚动, 通常显示一系列注册提示符, 则应确保只有调制解调器设备有效。如果非调制解调器设备有效, 应将其置为无效:

```
disable / dev / tty1a
```

#### 系统显示无意义的字符

如果系统显示了注册提示符, 但没有口令提示符, 或者在注册提示符之后显示没有意义的字符, 则应检查线路设置是否正确:

1. 确定正呼叫的系统上的串行线路设置。cu 使用的标准设置是 8 位数据位, 一位停

止位，没有校验位：

- 如果远程系统使用的是偶校验，则应执行命令 `cu-e`。
- 如果远程系统使用的是奇校验，则应执行命令 `cu-o`。

2. 如果正拨号进入 UNIX 系统，则应检查远程系统上的 `/etc/inittab` 文件，以确保 `/etc/gettydefs` 文件中的指针是正确的。在选定的 `gettydefs` 条目中第三个域所定义的串行端口设置值也许会不匹配，可改变端口的配置为 8 位数据位，1 位停止位，没有校验位。

条目应该形如：

```
4#B1200HUPCL#B1200CS8SANE HUPCL TAB3ECHOE IXANY#\ \ --&Y\ \ &n@llogin:#5
```

## 1.8 打印机系统故障排除

本节讨论在使用打印机系统时可能会遇到的一些共同的问题：

- 打印机不能打印
- 系统在启动时不能识别打印机端口
- 打印机端口的重定向功能不正常
- 打印的输出非法
- 打印机输出格式不对
- 并行打印机异常的慢
- 拨号输出打印机报告 UUCP 错误

### 1.8.1 打印机不能打印

如果在打印命令发出之后打印机不动作，没有打印输出，则应作如下检查：

1. 确保打印机电源接好。
2. 检查打印机的硬件是否正常。进行自检的方法应查阅打印机的有关手册。
3. 检查打印机电缆，保证它正常地连接到端口和计算机上。在计算机的使用手册中查阅安装命令。

4. 确保打印机配置正确。要建立并行或串行打印机使之能正确接收数据。

如果打印机是串行的，应保证计算机向打印机发送数据的波特率与打印机的波特率相匹配。关于如何重新设置打印机的波特率，请参考本章“打印机的输出非法”一节的内容。

5. 应保证打印机处于有效状态。为此，应启动 `sysadmsh` 并选择：

```
Printers >Schedule >Accept
```

```
Printers >Schedule >Enable
```

6. 确保系统在启动时能够识别打印机端口。在版权信息之后，系统对每个端口都应显示类似下面的信息：

```
Parallel 0x378-0x37A 07 - Unit = 0
```

如果系统没有显示对应打印机的此类信息，则应按照本章后面“在启动时不能识别端口”一节中的有关命令执行。

7. 确保端口配置的中断向量正确，并且没有其它硬件在使用同一中断向量。关于可用的中断向量的情况，见本章中“解决磁带驱动器故障”一节中“在启动时不能识别磁带”部分

的讲述。关于如何配置端口的内容，请查阅硬件文献。

8. 测试打印机端口的连接，办法是将命令的输出直接重定向至设备。

- 对于并行打印机 LP0，打入：

```
date > /dev/lp0
```

- 对于串行打印机 ttyla，打入：

```
(stty options; date) > /dev/ttyla
```

这里 options 是波特率、校验码以及别的你要传给串行打印机的设置。

如果重定向命令的输出不能打印，应按照本章中“不能重定向输出至打印机”一节的命令执行。

如果重定向命令的输出能打印，则提交一个标准文件（如 /etc/motd），启动 sysadmsh 进行打印，选择：

```
Dirs / Files → Print
```

如果硬件连接正常，打印机有效且配置也正确，但打印请求仍得不到执行，则应作以下检查：

1. 检查 lpsched 进程是否在运行：

```
PS -U IP
```

2. 如果 lpsched 进程不在运行，则重新启动：

```
/usr/lib/lpshut
```

```
/usr/lib/lpsched
```

3. 检查打印请求是否经过筛选：

```
lpstat -o -l
```

该命令显示有关输出请求的状态，打印机名以及相关设备的详细情况。

4. 如果打印机检测出错误，不会马上继续自动打印过程。通过使打印机有效强制继续该过程。

5. 检查是否因为拨号输出打印机忙或者没有回答，或者所有的拨号输出端口都忙。打印服务在再次尝试访问拨号输出打印机之前要等待 5 分钟。在那之后可使能（enable）打印机，再作一次访问尝试。

### 不能重定向输出至打印机

如果将输出直接重定向至并行或串行端口，结果毫无动静，或者系统显示“不能建立”的消息，则应作如下检查：

1. 检查该端口对应的设备文件在 /dev 目录中是否存在。确保此文件是一个设备文件而不是文本文件。

2. 从一个正常的系统上取下电缆用在本系统中以测试电缆的连接情况。

3. 从 DOS 操作系统打印一个文件。如果在 DOS 下能够正常打印文件而在 UNIX 系统下却不能，则应作如下检查：

- 确保端口在系统启动时已被识别，并且端口配置正确。

- 如果端口配置正确，而还是不能将输出重定向至端口，则应换一个设备名试试。比如，对并行端口来讲，用 LP1 来替换 LP0；对串行端口而言，用 tty2a 替换 ttyla。

- 如果换了一个设备名还是不能打印，则可能是打印机有缺陷，应检查随打印机一同到货的硬件文献以寻求解决办法。

4. 如果在 DOS 下也不能打印，则应检查打印机配置。

如果打印机的配置正确，而还是不能将输出重定向至打印机，则问题很可能出在硬件上。这时应再次检查电缆和端口。如果问题还是不能解决，可与打印机供应商联系。

系统启动时没能识别端口：

如果系统在启动时未能识别端口（即在版权信息之后未显示对应于端口的“parallel”或“serial”内容的一行信息），应作如下检查：

1. 确认并行或串行卡的放置正确。

- 关掉电源，打开机器。

- 从总线上拔出打印卡，再重新插入。

2. 如果并行卡多于一块，则可能是其中一块与另一块发生了冲突，应移去第一块。

3. 检查打印卡的配置是否正确，这可以通过对照随卡一同到货的手册来实现。如有可能，可将卡设置为另一种配置试一试。

4. 打印卡可能坏了，换一块试试。

#### 打印机的输出非法

如果打印机打印出非法输出，则应对其作出如下检查：

1. 确定串行打印机的波特率，看它与计算机的波特率是否匹配。（如果打印机是通过并行端口连接的，则与波特率没有关系。）为达到最佳性能可将波特率设置为 9600。

要将打印机服务所用的波特率作一改变，应使用如下的 sysadmah 选择：

Printers >Configure >Parameters

在“Default initial settings（缺省初始设置）”部分的“stty”域内填入波特率数值。然后将一个标准样本文件提交打印。

2. 确定打印机的奇偶校验设置情况，看它与计算机的校验设置是否匹配。（如果打印机通过导线与计算机直接相连，且相间距离小于 50 英尺，则没有必要使用奇偶校验位。）

设置奇偶校验后，应使用如下的 sysadmsh 选择：

Printers >Configure >Parameters

在表格的“Default initial settings”部分，对“stty”域增加下列三项之一：

oddp 设置产生奇校验

evenp 设置产生偶校验

-parity 设置不用校验位（缺省值）

根据打印机的需要设置匹配的选项。

3. 检查“tab”的设置是否正确。

如果这些设置及波特率都正确，而输出还是不正常，则应检查打印机类型正确与否。如果在用打印服务建立打印机时选择的打印机类型不对，就会将不正确的控制字符传送给打印机，这会引起打印机的输出消失或者非法。另外，还可能由设置了错误的字符集所致。

如果不知道打印机的类型，则应使用以下步骤查出打印机类型。

1. 如果认为打印机是某一类型的而又不能确信，可在系统提示符下输入如下命令：

TERM = printer-type tput longname

一段对由 printer-type 指定的打印机的简单描述会出现在终端上。如果以为该描述与你的打印机不相符，可以另换一个 printer-type 试试，重复这一过程直到描述与实际打印机相符为止。

2. 如果不知道用什么名字 (printer-type) 来试，可用以下命令来检查 /usr/lib/terminfo 目录，以得到打印机类型名清单：

```
ls -R /usr/lib/terminfo |more
```

从所得清单中选出一个词或数字来代表打印机类型。比如，名字 495 即代表 AT&T495 打印机。用清单中的每一个名字来替换第 1 步的命令中的 printer-type，直到满意为止。

当获得自己认为是正确的打印机类型名之后，可改变打印机类型设置。为此：

1. 启动 sysadmsh 并选择：

Printers→Configure→Parameters

2. 在打印机名 (terminfo 数据库) 域中，打入打印机的类型名。

#### 打印机输出格式不对

如果打印机的输出是合法的，但格式不对。则应启动 sysadmsh 并选择：

Printers→Configure→Parameters

针对下面的每一种情况，调整：“Default initial settings”部分“stty”域的设置：

- 如果打印机的输出的间距是正常情况下的两倍，则使用 -onlcr 或者 -tabs 选项。
- 如果输出的正文挤在一起，应输入 -tabs 选项。
- 如果打印机的输出呈锯齿状，则应输入 -onlcr 选择项。(该选项是缺省设置的，但管理员可能由于误操作将其去掉)。

#### 并行打印机慢

如果打印机速度异常地慢，则应按本书“打印机”一章中“安装打印机”部分讲述的方法检查配置是否设置好。

如果在确认并行端口配置正确之后打印机速度仍然很慢，则问题可能在于并行端口产生中断的能力不够。

要加快并行打印机的打印速度，可以改变硬件和打印驱动程序通信的方式。可以使并行打印机的驱动程序“假脱机”于并行端口，这样驱动程序即可不依赖于从并行端口产生的中断。

---

注：当打印机驱动程序“假脱机”并行端口时，可能会遇到系统资源用完的问题。

---

要对并行端口或并行打印机建立假脱机机制，应建立所谓的“特殊设备节点 (special device node)”。

执行步骤如下：

1. 注意一下在系统启动时识别的是哪一个并行打印机端口。(可以在 /usr/adm/messages 文件中查看最后的启动信息。) 比如：

```
parallel 0x378 0x374 07 unit=0
```

2. 作为超级用户注册。



3.输入下列命令中的一个以建立对应打印机的特殊设备文件:

- 对 lp0, 打入: `mknod /dev/lp0p c 6 64`
- 对 lp1, 打入: `mknod /dev/lp1p c 6 65`
- 对 lp2, 打入: `mknod /dev/lp2p c 6 66`

4.对所有想要配置的打印机都输入以下命令。比如,对 /dev/lp0, 应输入:

```
chown bin /dev/lp0p
chgrp bin /dev/lp0p
chmod 222 /dev/lp0p
```

5.如果正在使用打印的假脱机程序,现在就必须将新的并行假脱机设备通知假脱机程序。

启动 `sysadmsh` 并选择:

Printers >Configure >Modify

6.在“Device name (设备名)”域中,不要用标准并行设备名,应该用 /dev/lp0p, dev/lp1p, 或者 dev/lp2p。

### 打印机报告 UUCP 错

如果配置了 UUCP, 打印机服务就用 UUCP 软件来处理拨号输出的打印。如果出现拨号失败, 接收到打印机故障提醒, 打印服务也会报告同样的错误, 这是由 UUCP 软件报告的。(如果没有作接收故障提醒的安排, 则该信息邮往用户 LP, 这是缺省规定。)

## 1.9 解决进程管理中的问题

这一节将讨论如何解决系统进程被用尽的问题(进程表满了), 如何处理键盘被失控的进程锁死或进程杀不掉的情况。这一节还要讨论如何为整个系统或每个用户配置最大数目的进程。

### 1.9.1 再没有进程了

一个用户(不包括超级用户)能够同时拥有的进程数目是有限的。每一个在系统上运行的进程都在核心进程表中使用一个登记项。如果在没有可用的进程时(当进程表已经满了), 或在给一个 UID 所限定的进程数目已经被超过时, 还要创建一个新的进程, 那么 `fork` 就会失败同时系统显示如下信息:

```
No more processes:
```

这个错误不会在控制台上显示什么信息。

当系统显示了这个错误信息, 可以用 `ps (C)` 命令查找失控的进程。要想显示系统上的所有进程, 可输入:

```
ps -ef | more
```

有关这个命令的更多的内容, 请看 `ps (C)` 的说明。

当一个程序进入了无限循环并不断产生新进程时, 系统可能用尽所有进程。如果确实发生了这个问题, 就请参阅本章后面的:“停止失控的进程”一节。

如果系统不断地显示:“No more processes”这个错误信息, 那么可使用如下过程来看一下系统是否需要调整:

1.使用如下命令判断进程表目前的大小:

```
cd / etc / conf / cf.d
```

```
./configure y NPROC
```

2.用如下命令来判断当系统处于很重负载的情况时所运行的进程的数目:

```
pstat -p | head -1
```

NPROC 的值应该比在使用高峰时所运行的进程的数目大 10%。

如果 NPROC 的值太小,就应该给核心配置更多的进程。要做这件事:

1.调用 sysadmsh 并选择:

```
system >Configure >Kernel >Parameters
```

2.选择 4 (Processes, Memory Management, and Swapping) 并按回车键。

3.在 NPROC 提示下,指定一个新值给(全系统)进程的最大数目。

4.要想增加每个用户进程的最大数目,可遵循同样的过程,只是要在 MAXUP 提示下指定一个新值并按回车键。

5.按 e 并按回车键而回到 Parameters 菜单。

6.返回 sysadmsh 主菜单并选择 System >Configure >Kernel >Rebuild 来重新连接核心。

7.选择 System >Terminate 重新引导系统。

系统启动后,新的进程表的大小就生效了。

### 1.9.2 停止失控的进程

失控的进程能够造成封锁键盘的错误,使得用户所打入的任何内容都无法进入系统。因此,无法在调用这个失控进程的终端上停止这个进程。

为了继续工作,必须找到并停止这个失控的进程:

1.在没有被锁死的终端(或控制台屏幕)上以根(root)的身份登录。

2.输入如下命令并按回车键:

```
ps -ef
```

系统将显示当前所有进程的所有信息和它们的进程标识号(PID)。找到失控进程的 PID。

3.输入 Kill PID。这个程序几秒钟后就停止。如果这个进程没有停止,就再输入 Kill -9 PID。

最后这个命令可能会留下几个在程序正常结束时通常应被删掉的临时文件,也可能造成终端不能回显。为了恢复终端的正常操作,请遵循本章后面“解决终端问题”中恢复不回显终端中的指令。

### 1.9.3 删去杀不掉的进程

如果当使用带选项-9的 Kill 后失控的进程仍然不停止,那么这个进程就被认为是一个杀不掉的进程。如果这个杀不掉的进程是一个用户 shell,那么必须停止这个进程用户才能继续工作。这了停止一个杀不掉的进程,必须重新引导系统。使用如下过程:

1.以根(root)的身份登录并用 wall (ADM) 给其它系统用户发送一条信息通知他

们就要关闭系统了。

2.当所有用户都退出系统后,输入 /dec/shutdown 而关闭系统。

3.在引导提示符下按回车键重新引导系统。

当运行 ps 时那个杀不掉的进程就不会再出现了。

## 1.10 解决调度问题

这一节将介绍如何启动 cron 和 lpsched 守护程序以及如何赋给用户 chmodsuid 核心权限来使用 at (C)。

### 1.10.1 cron 守护程序没在运行

cron 守护程序在特定的日期和时间执行随 at (C)、batch (C) 和 crontab (C) 命令提交的命令。(cron 守护程序在系统启动时由 /etc/rc2.d 目录中的一个程序自动启动。)如果 cron 没在运行,而你又想随这些程序提交一个作业,那么系统就会显示如下信息:

```
cron may not be running -call your system administrator
```

如果先前调度的一个作业没有执行,这也说明 cron 没有运行。可以使用如下命令来看 cron 守护程序是否在运行:

```
ps -ef | grep cron
```

如果没有 cron 进程,那么这个守护程序就没有运行。要想启动 cron,就必须以超级用户根 (root) 的身份登录,并在系统提示符下输入 cron。

系统将显示如下信息:

```
! * * * cron started * * * pid=4612 Mon Dec 18 17: 44: 19 PST 1989
```

随 at、batch 和 crontab 所调度的作业现在应该正确执行了。如果 at 和 batch 继续报告错误信息,那么请看本章后面的“at 命令失败:不能改变作业模式”。

### 1.10.2 lpsched 打印调度程序没在运行

如果打印设备在打印请求的中间停止,或是不能启动新的打印作业,那么就要判断一下 /usr/lib/lpsched 守护程序是否正在运行。要做这件事,请输入如下命令:

```
lpstat -r
```

(lpsched 守护程序在系统启动时由 /etc/rc2.d 目录中的一个程序自动启动。)如果这个调度程序没有执行,那么系统将显示如下信息:

```
scheduler is not running
```

如果确实是这样,就按如下步骤启动 lpsched 守护程序:

1.以超级用户根 (root) 的身份登录。(一个有 lp 权限的用户也可以启动这个打印守护程序。)

2.调用 sysadmsh 并选择:

```
Printers→Schedule→Begin
```

系统会显示如下信息:

```
Print services started
```

在所保存的打印请求被打印以前可能用一至两分钟来重新建立打印机参数配置,格式

和筛选程序。在这个调度程序停止时还没打印完的任何打印请求都会在打印设备启动之后完整地打印出来。在打印设备停止时仍然有效的打印机的参数配置，格式和筛选程序都会在它重新启动时被恢复。

### 1.10.3 at 命令失败：不能改变作业模式

如果不具备核心权限 `chmodsugid`，那么在想调用 `at (C)` 和 `batch (C)` 时系统就会显示如下信息：

```
can't change mode of job
```

为赋予 `chmodsugid` 权限而使用户能够使用 `at`，可用如下过程：

- 1.以根 (`root`) 的身份注册。

- 2.调用 `sysadmsh` 并选择：

Account→User→Examine: Privilege

- 3.把 `chmodsugid` 加到 Kernel Authorization 域

为了使这个修改起作用，这个用户必须退出系统然后再登录。

这与用户没被授权使用 `at` 的情形不同。如果一个没被授权的用户想用 `at`，那么系统就会显示：

```
at: you are not authorized to use at.Sorry.
```

要想让用户使用 `at`，请参阅附录“授权使用作业调度命令”中的“允许和禁止各用户使用 `at` 和 `batch`”。

## 1.11 解决磁带驱动器的问题

本节讨论当在系统中使用卡式磁带、floppy 或 SCSI 磁带驱动器时可能会遇到的一些带有普遍性的问题。一般来讲，这类问题是在用命令 `mkdev tape` 安装磁带驱动器之后随即出现的。

### 1.11.1 地址格式错

如果在运行 `mkdev tape` 命令安装卡式磁带驱动器之后，系统显示如下信息：

```
/etc/conf/pack.d/ct/space.c line 46 bad octal digit
```

则表示地址格式不对。要解决这一问题，应检查磁带控制器的基地址在输入时是否用了前缀“0x”而不是在地址后面加上“H”，如果是，则应改过来。再次运行 `mkdev tape` 命令以修改卡式磁带的参数。

### 1.11.2 启动时不能识别磁带

如果系统在启动时没有显示类似下面的信息，则是因为系统没有识别磁带控制卡：

```
%tape 0x338-0x33C 05 1 tape=W
```

这时系统会显示类似信息：

```
ct: ERROR: Tape controller (tape=W) not found
```

对于卡式磁带驱动器，应作如下检查：

- 1.确保磁带控制卡物理上配置到了在运行 `mkdev tape` 时给予的基地址上。为做此检

查，应运行 `hwconfig (C)`。如果该值不对，则要么在物理上对磁带控制卡上的开关进行重配置，使之与在运行 `mkdev tape` 时所指定的基地址值相一致，或者再一次运行 `mkdev tape` 以改变此参数。

2.使用 `hwconfig (C)` 确保 DMA（直接存储器访问）通道，基地址及中断向量与系统中的其它任何硬件设备都不冲突。如果有一个或多个值发生冲突，则应运行 `mkdev tape` 以修改设备。

下表给出了系统硬件通常使用的中断向量：

表 1.1 中断向量

中断	设备
0	时钟
1	控制台
2	可用（经常以 25 输入）
3	COM2
4	COM1
5	LP2（替用并行端口）
6	软磁盘控制器
7	LP0 或 LPI（主并行端口）

注：不要用中断 0, 1, 6。

3.检查磁带控制器是否正确地安放到母板上，如有必要，可换一个槽插入控制器。

4.有些磁带驱动器（尤其是外部驱动器），要求驱动器附于磁带控制器，并且在启动时加上电源。

5.检查卡式磁带驱动器是否为系统所支持。

6.如果系统在启动时还是不能识别磁带控制卡，则该磁带控制器已坏。

对于 SCSI 磁带驱动器，应运行 `mkdev tape` 作如下检查：

1.检查设备控制器的 ID 号是否正确。ID 号由控制器上的跳线（跨接线）设置确定。正常值的范围是 0-7。

2.确保主机适配器号正确。第 1 个 SCSI 驱动器为 0 号，第 2 个为 1 号。

3.检查逻辑部件号 LUN 是否正确。在大多数情况下，控制器与设备在同一物理部件中，支持一个设备，LUN 为 0。如果控制器不与设备共部件，则能支持 8 个设备。如果是后一种情况，则 LUN 由每一个设备上的跨接线设置决定，其值的正常范围为 0-7。

4.检查磁带驱动器是否为系统所支持。

### 1.11.3 Tape 命令挂起

如果在用下列命令进行测试时磁带驱动器挂起：

```
tape reset
```

则应作如下检查：

1. 确保卡式磁带控制卡的物理配置与运行 `mkdev tape` 时给予的 DMA 值相一致. 如果不一致, 要么重新设置磁带控制器 DMA 的物理配置使之与运行 `mkdev tape` 时指定的 DMA 值一致, 要么再次运行 `mkdev tape`, 根据磁带控制器上 DMA 物理设置情况来指定 DMA 值。

2. 检查在磁带控制器和磁带驱动器之间的电缆连接情况是否正常。

3. 如果经过上述检查之后 `tape reset` 命令仍然挂起, 则磁带控制器, 磁带驱动器或者电缆, 三者中至少有一个坏了。

#### 1.11.4 不能打开 `/dev/rct0`

如果 `tar (C)` 命令不能将某一目录的内容写到磁带上, 并且系统显示如下信息:

```
Cannot open /dev/rct0
```

则作如下检查:

1. 使用 `hwconfig` 命令确保卡式磁带驱动器所用的中断向量不与系统上任一别的设备发生冲突。参阅本节中给出的可用中断向量表进行判断。为改变中断向量, 应运行 `mkdev tape` 并指示新的中断向量值。

2. 确保在卡式磁带控制卡上实际物理设置的中断 (IRQ) 与用 `mkdev tape` 指定的中断向量相一致。

3. 保证磁带驱动器对应的设备在 `/dev` 目录中。打入:

```
l /dev/rct0
```

输出应具有以下形式:

```
crw-rw-rw 1 root root 10,0 Feb 14 12:00 rct0
```

如果 `/dev` 的列表不包括这么一行内容, 则应再次运行 `mkdev tape` 以建立该设备。

4. 确保磁带控制器和磁带驱动器之间的电缆连接正确。

5. 如果 `tar` 命令还是不能正常工作, 则磁带控制器, 磁带驱动器及电缆三者中必有一个坏了。

## 1.12 解决终端问题

这一节将介绍如何解决如下有关终端的问题:

- 终端完全不起作用
- 终端被挂起
- 终端显示混乱
- 终端被封锁
- 终端不回显

### 1.12.1 恢复一个不起作用的终端

不起作用的终端不显示注册提示符, 对键盘输入也没有反应。这种情况通常是由硬件故障或配置有问题引起的。按如下步骤对不起作用的终端进行检查:

1. 检查终端上的亮度控制。
2. 检查终端和计算机的电源和通信连接。

3.如果终端提供了模式设置功能,就进入终端的设置模式并核查终端的配置设置。在这些设置中应该包括 9600 波特、8 位数据位、1 位停止位。无奇偶校验。

4.使能这个终端所连的端口。例如,要使能 tty007,就使用如下命令:

```
enable tty007
```

5.证实一下确实有一个与这个终端相关的 getty 进程。例如,输入:

```
ps -t tty007
```

6.通过禁止这个端口并且重新给这个不起作用的终端指定一个输出端口来测试硬件的通信情况:例如,要测试 tty007,就使用如下命令:

```
disable /dev/tty007
```

```
echo hello > /dev/tty007
```

如果这些步骤还是不能使这个终端恢复作用,那么就要看看随终端一起来的资料,看看如何解决其硬件上的问题。

### 1.12.2 修复挂起的终端

如果前面的工作对话区一直显示着,但是又不对键盘输入产生反应,那么这个终端就被认为是挂起了。使用如下步骤来修复这个挂起的终端:

1.要试图使这个终端复活之前至少等 60 秒钟(如果系统很忙,终端可能不会立即对击键产生反应,因为系统的反应时间加长了)。

2.按〈CTL〉q 来重新使能数据传输,以防 Ctr-S (传输暂停)被无意中按过。

3.检查一下所有的电源线,键盘线和通信电缆是否都连着。

4.给这个终端重新加电启动终端,然后运行不带参数变元的 test (C) 对它重新初始化。

5.检查一下终端设置模式下的配置情况。

6.像前面所说的那样从一个可操作的终端把输出重定向到这个被封锁的终端,以检查一下硬件通信。

7.使用如下命令检查一下运行于被封锁的终端端口上的进程:

```
ps -t tty007
```

使用 Kill (C) 停止在终端挂起时用户正在运行的进程。如果这个程序没有死,就必须重新引导系统来停止这个进程。

8.判断一下当前串行线的特性参数是否正确。例如,可使用如下命令显示 tty007 的这些值:

```
stty -a < /dev/tty007
```

9.用如下命令重新设置串行线特性:

```
〈CTL〉j stty sane 〈CTL〉j
```

如果 ps-t 命令只显示了 getty 程序,那么终端应该显示一个注册提示符。如果没显示,就应该再检查一下终端的硬件。

### 1.12.3 修复显示混乱的终端

混乱的终端对键盘的输入产生反应,但显示不正确。按如下各步修复混乱的终端:

1.用 env 命令检查该用户的终端类型 (TERM)。如果终端类型不对, 就对它进行重新设置。例如: 要把终端类型设置为 wyse60, 则输入:

```
TERM = wy60
```

设置终端类型之后, 输入不带参数的 tset 对终端重新初始化。

2.用如下命令重新设置串行线特性:

```
<CTL> j stty sane <CTL> j
```

#### 1.12.4 打开被封锁的终端

如果某终端被系统管理员封锁以防止在该终端上的注册, 或者如果因为用户超过了所限定的不成功注册的次数而使得系统封锁了终端, 那么在那个终端上将显示如下信息:

```
Terminal is disabled --see Authentication Administrator
```

要想打开这个终端, 可调用 sysadmsh 并选择:

```
Accounts->Terminals >Unlock
```

#### 1.12.5 恢复无回显的终端

无回显终端是对键盘输入发生反应但又不显示从键盘输入的字符的终端。(这区别于根本不对输入发生反应的被封锁的或不起作用的终端; 请看本章前面的“打开被封锁的终端”或“恢复不起作用的终端”, 以得到这些问题的解决办法。)

有时, 当一个程序由于出错而中止, 或者当用户按了 BREAK 键时, 终端就会停止回显。要恢复这个终端的正常操作, 就应输入:

```
<CTL> j stty sane <CTL> j
```

一定要准确地输入这个命令, 因为终端并不回显在键盘上输入的内容。

这个终端现在应该显示键盘的输入内容。如果它还不显示, 那么就要按着本章前面“恢复不起作用的终端”中所讲的步骤执行了。

### 1.13 解决 UUCP 连接的问题

本节讨论在使用 UUCP 时可能会遇到的一些带有普遍性的问题, 主要包括:

- 如何检测不正常的 ACU / modem
- CU 程序显示的错误信息
- 运行 UUCP 时遇到的共同性的错误
- 如何检测 UUCP 请求的状态
- 使用 uutry 来调试 UUCP 的通信
- 在 UUCP 登记文件和状态文件中出现的普遍性的信息
- 如何检查 UUCP 文件上的权限设置
- 检查系统名 (sitename) 是否唯一
- 如果 UUCP 速率非常慢该检查哪些方面
- 如果 UUCP 程序能正常工作, 而 uux 不能, 该怎么办。
- 对解决 UUCP 的问题的实用程序, uucheck, uulog, uuname, uustat, 和 uutry 的描述。



在开始解决 UUCP 系统中出现的问题之前，首先应保证物理连接是正确的。如果是通过调制解调器连接至 UUCP，则应确保调制解调器已安装好，并且配置正确；如果用的是导线直线连接，则应保证计算机连接正确。

### 1.13.1 检测不正常的 ACU / Modem

下面给出检查 ACU（自动寻呼部件）或者调制解调器（Modem）工作是否正常的两种方法：

- 为显示排入队列中请求的清单，最后请求时间，以及相关状态，应输入：

```
uustat -q
```

- 使用特定行打印接触过程中的调试信息，应打入如下命令：

```
cu -x9 -lline
```

---

注：为确保调制器免受未经授权的用户的干扰，本命令允许能对 Devices 文件进行写操作的用户使用。

---

### 1.13.2 用 cu 测定连接时的错误

本部分讲解在用 cu 命令测试连接失败时系统所显示的信息。

```
Connect Failed: CANNOT ACCESS DEVICE
```

如果连接失败，系统显示“CANNOT ACCESS DEVICE”信息时，应检查设备文件的权限。比如，要检查 tty1a 的设备文件，应打入：

```
l /dev/tty1a
```

属主信息和允许值设置应该具有如下形式：

```
crw--w--w- 1 uucp uucp 5,0 Feb 14 12:00 /dev/tty1a
```

如果该设备文件的属主信息及允许值设置形如：

```
crw----- 1 bin terminal 5,0 Feb 14 12:00 /dev/tty1a
```

检查在 /etc/inittab 文件中对应 tty1a 端口的一行上是否用了 /etc/getty 而不是正确的 /usr/lib/uucp/uugetty，正确的内容应该类似：

```
t1a:23:tty1a:off:/usr/lib/uucp/uugetty -t60 tty1a 2400
```

如果此行中误用了 /etc/getty，则应编辑 /etc/inittab 及 /etc/conf/init.d/sio 文件，并在 tty1a 端口对应的各行中将 /etc/getty 改成 /usr/lib/uucp/uugetty。

---

注：因为每次重新连接核心时都要创建一个新的 inittab 文件，为了保留 inittab 文件中的经改变的内容，必须同时对 /etc/conf/init.d/sio 文件进行修改。

---

要使对 inittab 文件所作的修改立即生效，可打入命令：

```
telinit Q
```

```
Connect Failed: SYSTEM NOT IN Systems FILE
```

如果计算机上的 /usr/lib/uucp/systems 文件中没有包含想要访问的系统对应的条目，则 cu 程序会显示 SYSTEM NOT IN Systems FILE（系统不在 Systems 文件

中) 错误信息。

要显示与系统相连的所有系统的清单, 应执行命令 `uname`。

如果使用 `-L` 任选项来指定一个串行端口时打入了错误的行号, 则系统也会显示这一错误信息。这时应检查端口配置是否正确。

### Connect Failed, NO DEVICES AVAILABLE

如果 `cu` 程序连接失败, 并且显示错误信息 `NO DEVICES AVAILABLE` (没有可用设备), 则应检查一下 `/usr/lib/uucp/Devices` 文件的建立是否正确。

还应确保对应正在使用的设备的行未被作为注释处理。(行首加上分号;) 该行内容即成为注释)。比如, 在 `Devices` 文件中, 使用 `ttyla` 且波特率为 9600 的直接端口对应的条目形式为:

```
Direct ttyla --9600 direct
```

如果在 `Devices` 文件中没有发现错误, 则可能因为远程线路忙的缘故。过一会儿再试试。

### Connected, but No Login Prompt

如果 `cu` 显示信息“Connected” (连接好), 但没有出现注册提示符, 则可能因为远程系统所用的线路忙, 可输入 `~` 并回车, 退出 `cu` 程序。

如果系统上一切正确, 但远程机器的注册提示符没有出现, 则应与远程机器的系统管理员交涉, 看远程机器上建立串行端口所用的通信参数与自己系统所用参数是否完全一致。**Connected, but Screen Displays Garbage**

如果连接上了远程系统, 但自己屏幕上显示内容杂乱 (所谓 `Garbage`), 则可能是连接坏了。打入 `~` 退出 `cu`。过一会儿再试。

出现这种情况的另外一种可能性是: 自己系统上的通信设置与远程系统上的设置不同。如果是这种情况, 则应与远程系统的管理员联系, 将两者统一起来。

### 1.13.3 UUCP 失败信息

如果打入 `uucp` 命令之后, `UUCP` 立即显示错误, 则此类错通常是语法错误或者权限错误, 错误信息有以下两类:

```
uucp failed completely (uucp 完全失败)
```

```
uucp failed partially (uucp 部分失败)
```

下面是一些带有共同性的问题以及系统显示的错误信息:

- 如果权限的设置使 `uucp` 命令不能访问源文件, `UUCP` 就显示以下信息:

```
can't read file (文件名)
```

```
uucp failed Partially (UUCP 部分失败)
```

应改变源文件上的权限值, 使所有用户可对其进行读操作。

- 如果在使用 `uucp` 命令时没有同时打入“源”和“目的地”, 则显示如下信息:

```
usage uucp from ... to (uucp 用法: from...to)
```

```
uucp failed completely (uucp 完全失败)
```

- 如果输入的系统名不在 `/usr/lib/uucp/systems` 文件中, 则 `UUCP` 显示:

```
bad system name: sitename (系统名错)
```

uucp failed completely (uucp 完全失败)

可用 uuname 来列出与自己的系统连接的所有系统的清单。

#### 1.13.4 检查 uucp 请求的状态

如果在输入“uucp”命令之后，系统没有立即显示错误信息，并且出现系统提示符，则 uucp 请求已进入队列。可用 uustat 和实用程序来检查 uucp 作业的状态。

用 uustat 可显示当前队列中 uucp 请求的状态，或者显示与其它系统连接的状态。比如，要为用户 robertm 显示所有排在队列中的作业的清单，可打入：

```
uustat -u robertm
```

uustat 实用程序以如下格式显示用户状态信息：

```
couscousN266 01 / 26-15:43 S couscous robertm rmail edwarda
```

若要列出所有远程机器的可访问性 (accessibility) 状态，应使用如下命令：

```
uustat -m
```

下面给出 uustat 命令输出形式的例子：

```
scooter 01 / 26-15 : 43 CAN'T ACCESS DEVICE
```

```
disco 01 / 27-12 : 01 SUCCESSFUL
```

```
obie 01 / 25-15 : 12 CALL FAILED, JOB DELETED
```

uulog 命令显示 UUCP 程序写入 /usr/spool/uucp/.Log 目录中的 uucp/sys, uux/sys, uucico/sys 以及 uuxqt/sys 文件里的状态信息。(sys 是 uucp 想要访问的系统名)

比如，要显示有关系统 scooter 的文件传输的状态信息，应打入如下命令：

```
uulog -s scooter
```

uulog 实用程序以如下格式显示关于系统的信息：

```
uucp scooter (01 / 26-15 : 43 : 00, 304, 5) COPY (SUCCEEDED)
```

---

注：uulog 命令显示 uucp 在 .Log 文件中记录的所有状态信息，但不一定包含错误信息。

---

#### 1.13.5 调试 UUCP 通信

如果不能与某一特定机器联系，可按照如下过程，使用 uucp 和 uutry 命令，调试与该机器的通信：

1. 作为超级用户注册。
2. 输入下列命令，在每一行之后打入回车：

```
uucp -r testfile machine\! /usr/tmp/testfile  
/usr/lib/uucp/uutry -r machine
```

这里 testfile 是所要传输的文件，machine 是想要联系的机器的系统名。-r 任选项的使用，使 /usr/lib/uucp/systems 文件中指定的重试 (retry) 时间不再有效。

---

注：如果用的是 csh，则必须去掉“!(\!)”符号。

---

3. 当看到下列信息之一时，按<DEL>键：

Conversation Complete: status SUCCEEDED

Conversation Complete: status FAILED

如果系统显示了头一条信息，则可在系统“machine”中查找文件 /usr/tmp/testfile。如果文件存在且不空，则 UUCP 的连接成功。如果系统显示的是第二条信息，则连接的文件传输失败。

如果 UUCP 系统的建立是正确的，则系统试图将“testfile”文件传输至远程系统“machine”。uucp 的 -r 任选项使文件进入队列而不开始传输，这样 uutry 即可启动 uucico 进行调试。

---

注：如果作为超级用户 root 运行 uutry 命令，则会显示出更多的调试信息。

---

uutry 命令将调试结果存入本地机器上的文件 /tmp/machine 中，并用 tail -f 命令将该结果打印到终端上。如果希望保存该结果，可从 /tmp/machine 文件中拷贝出来。可以用 -x 任选项改变调试级别。（缺省级别为 5。）

如果系统显示“Status FAILED”信息，则可以在该文件中查看状态和错误信息。

如果还是不能解决问题，也许就该与维修人员联系。可将调试结果保存下来以备诊断。

### 1.13.6 UUCP 登记和状态文件信息

UUCP 跟踪 /usr/spool/uucp 目录中登记和状态文件的活动。UUCP 程序 uucp, uux, uucico 和 uuxqt 将每个系统的状态信息写入 /usr/spool/uucp/.Log 目录中的文件里。uucico 实用程序将信息写到.Status 目录中的文件里。这些信息描述了传输请求的状态，比如 uucico 失败、完成、以及与自己的系统通信的每个远程系统允许进行下一次访问之前的时间间隔等。

下面讲述 UUCP 写到这些文件中时出现的带有普遍性的错误信息。

#### DEVICE LOCKED (设备锁)

uucico 在 /usr/spool/lock 目录中为远程系统建立一个名为 LCK..sitename 的锁文件。如果对应自己想要访问的系统的锁文件存在，则 uucico 认为设备正在使用。

要使用该设备，可在远程系统上输入下面的命令：

```
rm /usr/spool/locks/LCK *
```

#### LOGIN FAILED

如果出现此错误信息，则：

1. 检查本地机器上的 /usr/lib/uucp/Systems 文件，以验证该文件上的信息是当前信息。可能出现过期情况的是电话号码，注册号及口令。

2. 检查在通信双方的调制解调器配置是否正确。

3. 检查电话的连接，或者换一条电话线。

在每一步之后，都应启动 uutry 命令。

## NO DEVICES AVAILABLE

如果系统显示这一信息，则可能是由于以下情形之一：

1. 系统认为调制解调器正在使用，因为对应它有一个锁文件。可从本地机器打入下列命令：

```
rm /usr/spool/locks/LCK *
```

2. 寻呼系统没有正常的设备可用。检查在文件 /usr/lib/uucp/Systems 中命名的设备是否与 /usr/lib/uucp/Devices 中的条目相对应。

3. 如果错误信息依旧，则重新启动系统。再次执行 rm 命令，然后再启动 uucp 命令。

4. 如果重新启动系统之后错误信息依然存在，则可能由于调制解调器的配置不正确。试试是否能够用 cu 向外寻呼 (call out)。

## REMOTE DOES NOT KNOW ME

如果系统显示错误信息 REMOTE DOES NOT KNOW ME

(远程系统不认识我的系统)，则是因为想要访问的机器上的 /usr/lib/uucp/Systems 文件中不包含你的系统对应的条目。应检查远程机器的 Systems 文件。

## REMOTE HAS A LOCK FILE FOR ME

如果系统显示信息 REMOTE HAS LCK FILE FOR ME，(远程系统对我的系统有一个锁文件)，则要么是远程系统企图寻呼你的系统，要么是在远程系统上对你的系统有一个锁文件。

1. 输入下面的命令，在远程系统上删除锁文件：

```
rm /usr/spool/locks/LCK *
```

2. 再次启动 uucp 命令。

## SYSTEM NOT IN Systems FILE

如果在自己系统上的 /usr/lib/uucp/Systems 文件中不包含想要访问的系统所对应的条目，则显示错误信息“SYSTEM NOT IN Systems FILE” (系统不在 systems 文件中)。可用 uname 检查在本地机器的 Systems 文件中是否该有要访问的系统对应的条目。

## RETRY TIME NOT REACHED

当 uucp 请求失败之后，并不马上执行重试。在企图访问远程系统之后，有一个状态文件留在 /usr/spool/uucp/.Status/nodename 中。(nodename 是想要访问的远程系统名。) 该文件含有最近请求的相关信息，至少要在一定重试间隔时间之后才允许进行下一次请求。(重试间隔时间在文件 /usr/lib/uucp/Systems 中指定。) 如果想再次使用 UUCP，则系统会显示出类似信息：

```
RETRY TIME NOT REACHED (重试时间未到)
```

若想马上进行重试，可从 /usr/spool/uucp/.status 目录中删除远程系统对应的状态文件。

## CANNOT ACCESS FILE

如果系统显示信息“CANNOT ACCESS FILE”，则它不能访问寻呼设备端口。此时应检查设备文件上的权限值和属主信息：

```
l /dev/ttyxx
```

属主信息和权限设置的显示应该类似：

WRONG TIME TO CALL

如果系统显示信息“WRONG TIME TO CALL”（寻呼时间不对），则 `/usr/lib/uucp/Systems` 文件不允许在这个时候向外寻呼。这个时候再重发请求也不会有任何作用。

### 1.13.7 检查 UUCP 文件的权限设置

如果 UUCP 不能读文件或执行文件，就不会正常工作。由于从本质上来讲所有的 UUCP 文件都是只有超级用户可写的，其中有相当一部分是只有超级用户和 uucp 可读、可执行的，所以，应作为超级用户注册，修改 UUCP 系统。进一步工作完成之后，应确保所有 UUCP 文件的属主都是 UUCP 而不是 root。

用下列命令检查 UUCP 文件的权限：

```
cd /
```

```
fixperm -n -v -dUUCP /etc/perms/ *
```

fixperm 命令显示带有权限制的 UUCP 文件清单。

---

注：由于 Systems 文件和 Permissions 文件中含有未加密的口令，所以应该是只有 uucp（和超级用户）可读的。同时还应注意，`/usr/bin/ct` 程序的属主必须是超级用户而不能是 uucp，否则工作不能正常进行。

---

### 1.13.8 验证系统名是唯一的

应确保系统名的头 7 个字符是唯一的。如果自己机器与一台具有同样名字的机器直接相连，则 UUCP 不允许与那台机器通信。如果自己机器与另一台机器名相同的机器没有直接相连而是处于同一 UUCP 网络上，则电子邮件可能会发错地方。

要检查系统名，可打入：

```
cat /etc/systemid
```

想要修改系统名，要么手工编辑 `/etc/systemid` 文件，要么启动 `uuinstall` 程序，显示或修改自己的系统名。

### 1.13.9 UUCP 异常慢

因为 UUCP 是批处理网络，请求不是立即执行的。因此，当用户报告 UUCP 不能工作时，问题可能在于系统太慢。如果系统超乎寻常的慢，则应检查以下几方面：

1. 检查假脱机目录，`/usr/spool/uucppublic` 和 `/usr/spool/uucp`，看是否老的工作文件使系统超负荷，如果是，则删除这些文件。

2. 用 PS (C) 命令确保一次没有进行太多的 uucico 进程。通信设备数目的极值在 `/usr/lib/uucp/Maxuuscheds` 和 `/usr/lib/uucp/Maxuuxqts` 中指定。检查一下该极值是否太大。缺省值是 1。

3. 确保包含假脱机 (spool) 目录的文件系统没有把所有空间都用尽。  
(out of space)

4. /usr/lib/uucp/uudemon.clean 定期执行删除未能成功执行的老文件的操作。  
要修改这一操作的频率, 可修改 uudemon.clean 程序的 find 语句。

### 1.13.10 uucp 能工作, 但 uux 不能

如果能够用 uucp 在两个系统之间传输文件, 而不能用 uux, 则 /usr/lib/uucp/Permissions 文件有问题。因为在使用 uucp 实用程序时, 远程系统只需要 Permissions 文件的 LOGNAME 条目, 而 uux 还需要 MACHINENAME 条目。

要解决这一问题, 可用远程系统的名字在 Permissions 文件中增加 MACHINENAME 条目。比如, 若本地机器, couscous, 要访问 obie, 则在 obie 上的 Permissions 文件中对应 couscous 的条目应该具有如下形式:

```
LOGNAME=uucouscous MACHINE=couscous\  
COMMANDS=ALL\  
READ= /\  
WRITE= /\  
SENDFILES=yes REQUEST=yes
```

---

注: 上例中给出的权限没有考虑实际的安全性要求, 所以只能用于紧密耦合系统中。

---

### 1.13.11 解决 UUCP 问题的实用程序

下表给出几条可用于检查基本通信信息的命令。

表 1.2 通信信息的检查命令

命令	描述
uucheck	检查 UUCP 所需的文件和目录是否存在。本命令还能检查 Permissions 文件中的明显的错误。
uulog	显示指定机器的登记目录的内容
uuname	列出想要与之联系的机器的名字。
uustat	显示当前处于队列中的 uucp 请求或者与其它系统连接的状态。
uutry	启动 uucico 执行调试任务, 将信息存入文件 /tmp/machine 中, 并将调试结果的最近 10 行的内容定向至终端。-x 任选项可改变调试级别。(缺省值是 5)

### 1.14 排除其它系统故障

这一节将解决由系统故障引起的问题, 例如如何映射硬盘上的坏道、如何复位 HZ 值, 如何在 i 节点表溢出时调整系统、当系统发生紧急情况时应如何处理, 以及如何处理由于电源故障引起的非正常停机。

### 1.14.1 映射坏道

在系统安装过程中，系统先对硬盘扫描查找坏道并根据这些信息建立一个坏道表。列于表中的坏道被替换到好道中，这样操作系统就可以避开不能被读写的磁盘区域。

如果硬盘在安装系统之后产生了坏道，运行时就会在控制台上显示如下错误信息：

```
wd: ERROR: on fixed disk ctrlr=0 dev=0/47 block=31434 cmd=00000020
status=00005180, sector=62899, cylinder/head=483/4
```

如果发生这个错误，就要使用 badtrk (ADM) 建立一个新的坏道表，这样系统才能避开新的坏道。badtrk 这个实用程序是一个用来在坏道表中查看，增加，或删除登记项的菜单驱动程序。要使用 badtrk：

1. 以根 (root) 的身份登录并进入单用户模式：  
/etc/shutdown su
2. 在系统提示符下输入 badtrk。
3. 选择选项 2 来扫描磁盘，然后选择选项 1 来扫描整个 UNIX 分区。
4. 指明是想做快速的还是做彻底的扫描，并在破坏的 (destructive) 扫描提示下输入 n。

---

注：在非破坏模式下运行 badtrk 以保存硬盘上的数据。

---

5. 当扫描完成后，又会重新出现主菜单。选择 q 返回系统。  
badtrk 自动把它发现的坏道写到坏道表中。

### 1.14.2 修复坏的 HZ 值

如果 /etc/initscript 系统文件不正确或在系统操作的某一个时间被破坏，那么系统就会显示如下信息：

```
bad HZ Value
```

操作系统使用 HZ 这个变量代表系统中断时钟频率。HZ 的值必须在 /etc/initscript 文件中设置为每秒钟 60 个周期：

```
HZ=60
```

如果 HZ 在 /etc/initscript 中设置得不正确，或者这个文件被损坏或丢失，那么系统就会显示 HZ 错误信息。

如果已经证实 /etc/initscript 存在并且正确，那么这个错误信息可能表明系统核心串联得不对。利用如下步骤重新串联核心：

1. 以根 (root) 的身份登录。
2. 备份 /unix 文件：  
cp /unix unix.bkp
3. 输入：  
/etc/brand <serid#> <activationkey> /unix
4. 利用 /etc/shutdown 命令重新引导系统。



5. 在引导提示符下按回车键。  
系统应该不再显示 HZ 错误信息。

### 1.14.3 处理 i 节点表的溢出

每个打开的文件都需要在 i 节点表中有一个 i 节点登记项。如果 i 节点表太小，系统就会显示如下错误信息：

```
kernel: WARNING: Inode table overflow
```

在 i 节点表溢出时，个别的申请就会被拒绝。虽然对系统来说并不致命，但是 i 节点表的溢出可能会破坏各种缓冲池、守护程序、邮件以及其它重要实用程序的操作。异常的结果和数据文件的丢失就会经常产生。

如果系统接连显示这个错误信息，那么就使用如下步骤估计一下系统是否需要调整：

1. 输入如下命令判断当前 i 节点表的大小：

```
cd /etc/conf/cf.d
```

```
./configure -y NINODE
```

2. 通过在系统处于重负载的情况上运行如下命令判断活动 i 节点的数目：

```
pstat -i | head -1
```

3. NINODE 的值应该比在使用高峰时间的活动 i 节点的数目大 10%。

如果 NINODE 的值太小，那么就应重新配置核心以重新分配 i 节点表登记项入口。

要做这件事。

1. 调用 sysadmsh 并选择：

System→Configure→kernel→Parameters

2. 选择 3 (files, Inodes, and Filesystems)。

3. 在 NINODE 提示下，指定一个新值作为所打开 i 节点的最大数目。

4. 输入 e 并按回车键返回 Parameters 菜单。

5. 返回 sysadmsh 主菜单并通过选择 System→Configure→Kernel→Rebuild 来重新连接核心。

6. 选择 System→Terminate 重新引导系统。

当系统启动后，这个新的 i 节点最大数目开始生效。

### 1.14.4 处理系统的紧急事件

当系统遇到硬件问题或核心不一致现象时，这就是系统的“紧急状态”，这种情况非常严重，以至于使得系统不能继续运行。当这种情况发生时，系统就会在控制台上显示一条信息，并且停止所有系统活动。

系统的紧急状态信息包含在 severity 域中的单词“PANIC”，后面跟着一条诊断信息。系统也可能不显示 PANIC 信息就瘫痪了。当发生这种情况时，系统就会拒绝处理从系统控制台和所有其它终端得到的任何输入（包括 BREAK、INTERRUPT，和 CANCEL 键）。

为了解决系统的紧急状态：

1. 如果有 PANIC 信息，就把它从控制台屏幕上抄到系统记录本上。

2. 重新给机器加电，并在引导提示符下按回车键来重新引导系统。
3. 在根文件系统的检查提示下，回答 y。
4. 使系统进入单用户模式并对那些当系统发生紧急故障时正安装于系统上的文件系统运行 fsck (ADM)。例如，使用如下命令检查 /u 文件系统：

```
fsck /dev/u
```

5. 修复造成系统紧急故障的问题。

如果系统没能重新启动，或者每次一启动它就立即瘫痪，那么就是操作系统被破坏，并且必须重新恢复或重新安装。要做这件事，应按照本章前面的“恢复被破坏的根文件系统”中的过程来做。

---

注：如果用所提供的安装原盘中的引导软盘也不能启动系统，那么计算机一定有严重的硬件故障。应该请硬件维修人员来修理。

---

在大多数情况下，只要重新引导系统就能解决问题。然而，如果系统接连以同一条 PANIC 信息而停机，那么就应该找到引起系统发生这种情况的原因。

例如，如果向外寻呼 (call-out) 表包含的入口太少，那么控制台就会接连显示如下 PANIC 信息并瘫痪：

```
PANIC: Timeout table overflow
```

因此必须通过增加向外寻呼表的入口数来解决这个问题：

1. 调用 sysadmsh 来选择：

```
System→Configure→Kernel→Parameters
```

2. 选择 5 (clock) 并按回车键。

3. 在 NCALL 的提示下，为向外寻呼表的入口数指定一个新值。

4. 输入 e 并按回车键而返回 Parameter 菜单。

5. 返回 sysadmsh 主菜单并选择 System→Configure→Kernel→Rebuild 来重新连接核心。

6. 选择 System→Terminate 重新引导系统。

当系统启动之后，新的向外寻呼表的大小就生效了。并且系统不应该再以“timeout table overflow”这条错误信息而发生紧急事件。

#### 1.14.5 处理掉电故障

当系统由于掉电而关机时，这种关机被认为是异常的。当系统正常关机时，shutdown (ADM) 程序会停止所有守护程序，杀掉活动的进程、卸下所有文件系统、让 init 进入单用户模式，并运行 sync 命令。如果系统在这个关机过程完成之前就关掉了，那么就会发生如下情况：

- 文件系统可能会不一致或被破坏，造成数据丢失。
- 由于高速缓存中的数据没有被写到磁盘上而使得用户所进行的工作和其它数据可能丢失。

因为 UNIX 每 30 秒钟写盘一次，所以由于没执行 sync 命令就关掉了系统而丢失的

数据量很小。然而，因为在系统被关掉之前没有正确地卸掉文件系统，所以文件系统的破坏可能更严重。

如果根文件系统被破坏，系统就不能正常工作。

当掉电时，应立即关掉机器电源。这样所造成的损失比让电源继续波动要小。

一旦重新加电，应使用如下步骤恢复系统：

1. 开启计算机电源，并在引导提示符下按回车键。

2. 如果根文件系统被破坏，系统就会提示是否要检查 `/dev/root` 文件系统；输入 `y` 来调用 `fsck (ADM)`。系统会显示如下信息：

```
FREE INODE COUNT WRONG IN SUPERBLK (FIX?)
```

当系统没有正常关闭，就会例行产生这个信息。输入 `y`，`fsck` 就会解决这个问题。

当自动引导时，`fsck` 会自动解决根文件的任何问题，只要在 `/etc/default/boot` 文件中包含下面这一行就可以：

```
FSCKFIX=YES
```

如果 `/etc/default/boot` 不包含这一行，`fsck` 就会向管理员询问如何解决它所遇到的问题。

3. 如果当系统进入多用户模式时有某一个非根文件系统被破坏，系统就会提示管理员是否要清理这个文件系统；输入 `y`。

如果在 `/etc/default/filesys` 文件中有关那个文件系统的登记项为：`fsckflags=y` 那么这时 `fsck` 命令就会自动修复这个文件的任何问题。

如果在 `/etc/default/filesys` 中有关这个文件的行中不包含这一行，`fsck` 就会提示管理员对这个被破坏的文件系统中的各种问题进行修复。

## 第二章 常用命令解释

### at, batch

在以后的某个时间执行命令

#### 语法:

```
at time [date] [increment]
at -r job-id...
at -l [job-id...]
at -qletter time [date] [increment]
batch
```

#### 说明:

at 和 batch 都从标准输入设备接受一个或多个命令供以后某个时间执行。at 和 batch 在命令或作业调度的方法上不同: at 可以让你指定这个作业应该执行的时间, 而 batch 是在系统负载水平允许时执行这个作业。

at 有如下变元:

- time            time可以是1位, 2位或4位整数。一位或两位整数表示小时, 四位整数表示小时和分钟。time 可以被表示为由一个分号分开的两个数, 表示小时: 分钟。可以填上后缀 am 或 pm; 否则就当作 24 小时的时钟时间来理解。后缀 zulu 可以用来指示格林威治标准时间 (GMT)。一些特定的名字, 如 noon、midnight 和 now, 也是可认的。
- date            选项date既可以被指定为月份后跟着日期的形式, 也可以用星期几表示 (全拼形式或简拼为三个字符)。另外两个特殊的日期, today 和 tomorrow 也是可认的。如果没给出日期, 那么如果所给的小时大于当前的小时, 日期就假设为 Today, 如果所给的小时小于当前的小时, 日期就假设有 tomorrow。如果所给的月份小于当前的月份, 就假设是下一年。
- increment       time和date这两个变元可以被“+n units”这种形式的增益变元来修改, 其中 n 是一个整数, units 是 minutes、hours、days、weeks、months 或 years 中的一个。单数形式也可以被接受, 并且“+1unit”也可以被写成“next unit”。
- r job-id...    删去以前被at或batch命令调度过的特定的作业。job-id是由at或batch返回的作业标识符。除非你是超级用户, 否则你只能删除你自己的作业。
- l [job-id...]  列出指定作业的调度时间。如果没有指定任何job-ids, 就会给用户列出所有当前正被调度的作业。除非你是超级用户, 否则你只能列出你自己的作业。
- qletter        把指定的作业放入由letter注明的队列, 其中letter是任何一个从“a”到“z”的小写字母。队列字母被加在作业标识符的后面。下面的字母有特殊意义:

- a at 队列
- b batch 队列
- c cron 队列

batch 没有变元；它提交一个作业立即执行，但这个作业的优先级比普通 at 作业的要低。

## backup

执行 UNIX 备份功能

语法：

```
backup [-t] [-p] [-c] [-ffiles] [-u“user1] [user2]”] -d device
```

```
backup -h
```

说明：

UNIX 备份程序是 cpio (C) 的前端。要使用 restore (ADM) 来恢复由这个程序所做的备份。建议不要用这个程序进行系统备份，最好使用 sysadmsh (ADM) 接口程序来进行系统备份。

- h 产生一个备份的历史。告诉用户最后一次完全的和增量 / 部分备份是什么时候做的。
- c 完全备份。自从系统安装之后所有修改过的文件都备份。
- p 增量 / 部分地备份。这个选项只备份自从最后一次备份之后所修改的文件。在部分备份之前一定要做一次完全备份。
- f 由 <files> 变元指定备份文件。文件名中可以包含扩展字符 (例如 \*)。这个变元一定要在引号中。
- u 备份用户的主目录。在用户主目录中的所有文件都将被备份。至少要指定一个用户，也可以指定多个。如果指定了多于一个用户，那么这个变元必须放在引号中。如果用户名为“all”，那么所有用户的主目录都被备份。
- d 用于指定所用的设备。缺省设备为 /dev/rdisk/f0q15d (1.2M 软盘)。
- t 当设备为磁带时使用这个选项。当所指定的设备为磁带时，这个选项一定与 -d 选项一起使用。

在部分备份之前一定要做过一次完全备份。这个程序也可以处理多卷备份。当程序需要使用下一卷介质时，它会提示用户。这个程序还会给出一个备份所需的软盘 / 磁带数目的估计值。软盘一定要在备份之前进行格式化。磁带不必要格式化，除非使用的是小型盒式磁带。如果备份被做到磁带上，磁带必须要倒回。

## badtrk (ADM)

在固定盘上扫描坏块并建立坏道表

语法：

```
badtrk [-e[-m max]][-s qtdn][-f device]
```

说明：

badtrk (ADM) 主要在系统安装时使用。它在磁介质表面查找有缺陷的磁道，建立

新的坏道表，打印当前坏道表，并可增删表的各条目。列在表中的坏道都有相对应的好磁道，当处理器要对列在坏道表中的磁道进行读写时，即用该磁道对应的好磁道来代替。替换用的好磁道是在安装系统的过程中运行 badtrk 程序时分配的。若要改变分配作替换用的磁道数目，需要重新安装操作系统，所以替换用的磁道数目应相当大。

要使用 badtrk (ADM) 命令，必须处于单用户方式。(即系统维护方式)。

任选项:

-f device: 打开分区 device，并读该分区相应的坏道表。device 必须是一个固定盘的活动 UNIX 分区: / dev / rhd0a 表示第一驱动器， / dev / rhd1a 表示第二驱动器，以此类推。缺省值为 / dev / rhd0a。

-e: 在安装过程中使用，-e 标志使得 badtrk 程序改变坏道表的大小。

---

注: -e 标志不可由用户来使用。因为这会破坏硬盘，使存于其上的大量信息遭到破坏。

---

-m max: 只在非交互方式中与 -e 标志联合使用，-m 选项将坏道的最大数设置为 max。

-S argument: 以非交互方式启动 badtrk，使之在磁盘上扫描坏道，并将发现的坏道填入坏道表中。argument 指示是快速扫描还是彻底扫描，以及是破坏性扫描还是非破坏性扫描:

[q] 表示快速扫描 (quick)

[t] 表示彻底扫描 (thorough)

[d] 表示破坏性扫描 (destructive scan)

[n] 表示非破坏性扫描 (non-destructive scan)

用户应指定 q 或 t，以及 d 或 n。

用法:

在交互地执行 badtrk 时，程序首先显示主菜单:

1.Print Current Bad Track Table

2.Scan Disk (you can choose Read-only or Destructive later)

3.Add Entries to Current Bad Track Table by Cylinder / Head number

4.Add Entries to current Bad Track Table By Sector Number

5.Delete Entries Individually From Current Bad Track Table

6.Delete all Entries From Bad Track Table

Enter your choice or 'q' to quit:

系统提示输入选项数字，根据所作的选择，后面将询问更多的信息。

坏道表形式为: (选任选项 1 可得)

---

Defective Tracks

---

Cylinder	Head	Sector	Number (S)
1. 190	3		12971-12987

---

若选 2，则会在磁盘上扫描坏道。如果在最后一次更新磁盘上的坏道表之后表的内容有所改变，系统就会询问是否愿意在开始扫描之前将新的坏道表的内容写到磁盘上。应打入“Y”将所作的改动存下来。如果打入“N”，则表示不愿保存到现在为止对坏道表所作的

改动。接着系统要求指定扫描的类型：是快速扫描还是彻底扫描，是破坏性扫描还是非破坏性扫描。若选择破坏性扫描，则在扫描过的区域上的所有数据都会丢失。在回答了以上的询问之后，badtrk 程序开始执行扫描。管理员可以随时中断扫描过程，只要打入“q”就行了，这时就会询问是继续扫描还是返回主菜单。

当程序发现有缺陷的磁道时，即显示每一坏道的位置。下面给出一条错误信息的例子：

```
wd: ERROR: on fixed disk ctrl=0 dev=0 / 47 block=31434 cmd=00000020
```

```
status=00005180, sector=62899, cyclindev/head=483/4
```

(如果在扫描过程中出现读或写错误，也会出现类似的一条信息。)

扫描完成之后，重新出现主菜单。程序将自动地把发现的坏道填入坏道表中。

若磁盘配有坏块图 (flaw map) 则应该将这些坏块填入坏道表中。根据磁盘所配坏块图的格式选择任选项“3”或“4”。在一行上只应填入一条坏道。

当认为 badtrk 已经将所有坏块列到坏道表中，即可在主菜单下打入“q”退出 badtrk 程序。

如果启动 badtrk 程序时带有 -e 标志 (这种情况只应出现在安装过程中 hdinit 程序调用 badtrk 时)，而且磁盘中含有正确的分区表，则在出现 badtrk 菜单之前会显示下列信息：

```
This device contains a valid division table. Additional (non-root) filesystems can be preserved across this reinstallation. If you wish to be able to preserve these filesystem later, you must not change the current limit of the bad track table, which is n bad tracks. Do you wish to leave it unchanged? <y/n>:
```

如果以“y”回答，则以后系统不会提示给输入新的坏道表最大容量值。你可以增加或删除坏道条目，但不允许增加分配的坏磁道数的最大值。如果回答“n”，则坏道表的大小改变，磁盘分区表遭到破坏。

如果磁盘分配表不正确，或者在系统询问时打入了“n”，则系统要求你给出分配作替换磁道的数目。对该数目给出了一个推荐使用值，它是基于已知的坏道数，加上为将来可能出问题的磁道留的裕量而得。在分配替换磁道数目时，至少不应低于该推荐值。在作此分配时必须格外，因为在选定该值之后若又想改变，则只能重新安装系统。

badtrk 程序在退出之前，会询问是否希望以新的坏道表更新旧表。如果想使所作的变动生效，应回答“y”。如果希望坏道表保持运行 badtrk 程序之前的状态，则可打入“n”。

---

注意：本实用程序只能在单用户方式下使用。

---

## boot (HW) ——UNIX 引导程序

boot 是用于装载并执行独立的 UNIX 程序的交互式程序，主要用于装载和执行 UNIX 操作系统核心，但也可用于装载和执行其它的程序。boot 程序是 UNIX 操作系统所必需的部分，而且必须放于根文件系统的根目录下，才能保证 UNIX 系统核心的成功装入。

计算机每次启动，系统都要调用 boot 程序。若想不经过冗长的关闭系统 (shutdown) 过程而直接重新启动系统，可以使用 reboot 命令。这使得在停机后不等待键盘输入就直接启动系统。

从软盘引导的步骤如下：

- 1.ROM 从软盘的 0 扇区装入引导块。
- 2.引导块从软盘文件系统中装入 boot 程序。
- 3.执行 boot 程序，出现提示符。

从硬盘启动的步骤为：

- 1.ROM 从软盘的 0 扇区装放主引导块。
- 2.主引导块从活动分区的 0 扇区装入分区引导块。
- 3.然后，假设 UNIX 分区是活动分区，则将 boot1 装入活动分区。boot1 在硬盘上占据 20 个连续块（每块 1K 字节）。
- 4.boot1 从 UNIX 文件系统装入 boot 程序。
- 5.执行 boot 程序并出现提示符。

由于主引导块，boot0 和 boot1 不能放于坏道上，所以有可能被放在经过 badtrk (ADM) 映像好的“好”磁道上。

如果软盘驱动器空，则执行硬盘引导。

在首次调用时，boot 程序显示以下的信息：

UNIX System V

Boot

:

要求输入要装载的程序的位置。为此，应给出设备名和文件名。文件名必须是存放独立程序的文件的完整路径名。打一个(?)号，即可显示当前可用的设备名清单。

设备和文件名的格式如下：

xx (m, o) filename

或

xx (m) filename

其中，xx 为设备名（硬盘用 hd，软盘用 fd）

m 为次设备号（硬盘上的根文件系统用 40）

o 为分区内的偏移量，该值是任选的。

filename 为标准的 UNIX 路径名。

所有的数值都是十进制的。

若希望 boot 程序暂停一下，等待打入回车键之后再执行装入的程序，则应输入关键字“prompt”。比如，如果输入“prompt”并回车，则 boot 程序再显示如下信息，等待再次打入回车：

Loaded, press <RETURN>。

打入回车之后，开始执行 UNIX。“prompt”可以在命令行或 /etc/default/boot 文件中设置。如果不指定“prompt”，则 boot 程序不经暂停，直接执行所装入的程序。

如果只从 UNIX 软盘装入 boot 程序，则只要打入回车，boot 程序就会选取正确的



缺省值。

若从硬盘上装入 UNIX, 需装入:

```
hd (40, 0) unix
```

要使用在 /etc/default/boot 文件中指定的缺省引导字符串, 则在系统显示引导提示符时只用按回车键, boot 程序将使用在 /etc/default/boot 文件中以 DEFBOOTSTR 指定的字符串。

如果希望在硬盘上安装 DOS, 则建议在安装 UNIX 之前先安装 DOS。之后可在系统出现“boot:”信息时打入 dos, 从 DOS 启动。

#### 核心的配置

boot 程序把“prompt”串之外的所有输入串传递给操作系统核心。

核心读入引导串来确定根、管道和对换区分别是哪些设备。如果在 /etc/default/boot 文件和命令行中都未对设备加以说明, 则使用编译到核心中的缺省设备。

#### 选择系统控制台

在引导时选择系统控制台有两种方法: 在引导提示符下打入命令 systty = x, 或有 /etc/default/boot 文件中设置关键字 SYSTTY = X。字母 X 表示一个数或字符串参数。

如果在引导时用了 systty = x 命令, 则 boot 程序用字符串参数 x 把所选的控制台参数传递给核心。引导串参数 systty 的值为:

sio: 串行端口 COM1

scrn: 显示适配器

例如, 将系统控制台指定为串行端口 COM1, 应在引导提示符下打入下列命令:

```
systty = sio
```

如果在引导时不特别设置系统控制台, 则 boot 程序将按下面步骤来确定:

- boot 程序读 /etc/default/boot 文件, 查找关键字 SYSTTY = X, 这里 X 是表示系统控制台设备的数字:

- 1 表示串行适配器 COM1

- 0 表示显示适配器

- 如果没有查到关键字 SYSTTY, 或者 /etc/default/boot 程序不可读, 则 boot 程序查找显示适配器, 并将其指定为系统控制台。

- 如果没能找到显示适配器, 则 boot 程序查找串行端口 COM1, 将其设置为 9600 波特, 8 位数据位, 1 位停止位, 不用校验位的方式, 并用作系统控制台。

这样, 为使 boot 程序自动地把串行端口 COM1 设置为系统控制台, 应在 /etc/default/boot 文件中加入一行:

```
SYSTTY = 1
```

#### Boot 任选项

引导任选项可以通过 /etc/default/boot 文件中的关键字来改变。下列关键字可以为 boot 程序识别: AUTOBOOT = yes 如果是 yes, 则 boot 在经过一段时间的延迟之后自动装载 UNIX。

DEFBOOTSTR = string 指定缺省引导字符串。

SYSTTY = X 如果 X 是 1, 则系统控制台被设置为 COM1 上的串行适配器; 如果 X 是 0, 则系统控制台设置为主显示适配器。

RONLYROOT = NO 指示根文件系统是否被安装为“只读”方式。

TIMOUT = n n 是在 boot 程序引导操作系统之前 prompt 必须等待的秒数, 只在 AUTOBOOT 设置为 YES 时有意义, 缺省值为 60。

文件:

/ boot  
/ etc / default / boot  
/ etc / masterboot  
/ etc / hdbboot0  
/ etc / hdbboot1

### cleantmp

删去指定目录中的临时文件

语法:

/ usr / lib / cleantmp

说明:

cleantmp 删去指定目录中的临时文件, 这些目录由 / etc / default / cleantmp 文件中的 TMPDIRS 指定。按照 TMPDIRS 的缺省设置, 本命令要对 / tmp 和 / usr / tmp 两个目录进行检查。用户可以自己向 TMPDIRS 中加入目录名, 各目录名之间用空格来分隔。在这些目录中的文件, 如果在最近 n 天之内没有被访问过, 就会被删除, 其中 n 是文件 / etc / default / cleantmp 中的 FILEAGING 变量所指定的天数。按照缺省设置, FILEAGING 为 7。用户可以修改 FILEAGING 的值。 / usr / lib / cleantmp 在每天 3:00a.m. 作为核心作业运行。请参阅 / usr / spool / cron / crontabs / root, 超级用户可以修改这个文件, 以改变 / usr / lib / cleantmp 运行的频率和时间。如果所指定的目录不存在或者它们只是一个安装点但并没有安装文件系统, 那么 cleantmp 就会给根发送一个邮件, 告诉根 (root) 这个目录不存在。

/ etc / default / cleantmp 的格式如下:

FILEAGING = 7

TMPDIRS = / tmp / usr / tmp

### configure

核心配置程序

语法:

/ etc / conf / cf.d / configure [options] [resource = value...]

说明:

configure 用于对各种核心资源进行确定和修改。对于最终用户, 使用 configure 比直接修改系统配置文件更加容易。对于设备驱动程序的设计者来说, configure 避免了对配

置文件的困难的编辑。

必须进入 `/etc/conf/cf.d` 才能执行 `configure`。

资源可以交互地或使用命令行变元来修改。加入可删除设备驱动程序由命令行中的 `options` 来指定。

下面的各段将讨论如何交互地使用 `configure`。

在使用 `configure` 修改系统配置文件之前，要先使用如下命令对核心备份一个副本。

```
cp / unix / unix.old
```

交互式用法：

当在命令行中没有给出任何选项（包括 `configure = value`）或者只给了 `-f` 选项，`configure` 就以交互方式工作。

当以交互方式调用 `configure` 时，首先会看到如下的分类菜单：

- 1.Disk and Buffers
- 2.Character Buffers
- 3.Files,Inodes,and Filesystems
- 4.Processes,Memory Management and Swapping
- 5.Clock
- 6.MultiScreens
- 7.Message Queues
- 8.Semaphores
- 9.Shared Data
- 10.System Name
- 11.Streams Data
- 12.Event Queues and Devices
- 13.Hardware Dependent Parameters
- 14.Remote File sharing Parameters

Select a parameter category to reconfigure

by typing a number from 1 to 14,or type 'q' to quit:

要想选其中的某一项，只要输入它的编号就可以（例如对于“Disk buffers”，只要输入“1”），然后再按回车键。

每一项都包含一些可配置的资源。对于每一个资源都会显示其实际的名称，简短的说明及其当前值。例如，对于“Disk Buffers”这项，你会看到：

NBUF:total disk buffers.

Currently determined at system start up:

NSABUF: system-addressable (near) disk buffers.

Currently 10:

NHBUF: hash buffers (for disk block sorting) .

Currently 128:

要想保持当前值，只要按回车键就可以。否则，就要对该资源输入一个合适的值，然后按回车键。`configure` 对每个值进行检查，以保证各值处于合适的范围。如果不是这样，`configure` 就会警告你这个值不合适，并认定你要修改这个值。

要想退出 `configure`，只要在分类菜单提示符下输入 `q`。如果曾做过修改，`configure` 就会问是否要以所修改的内容对配置文件进行更新。要想保持原来的配置值，就在提示符

下输入 n, 这就不会做任何修改。否则, 输入 y, configure 就会更新所需的系统配置文件。在 configure 完成之后, 就准备连接核心。

连接核心, 应输入:

```
/etc/conf/cf.d/link__unix
```

连接可能需要几分钟。在核心连接之后, 输入如下命令重新引导系统来运行新的核心:

```
/etc/shutdown
```

选项:

命令行选项是给写安装设备驱动 shell 程序的人设计的。你可以配置驱动程序, 删除配置文件中的驱动程序定义, 以及修改一些驱动程序的属性, 所有这一切都可以由命令来完成。还有一些对当前驱动程序配置进行查询的选项。

configure 可使用如下选项:

```
-a [func1 func2 ...]
-b
-c
-d [func1 func2 ...]
-f master file [dfile]
-g dev__name handler | dev__name
-h dev__name
-j [prefix][NEXTMAJOR]
-l priority level
-m major__dev__number
-o
-s
-t
-v interrupt__vector [interrupt__vector2...]
-w
-x
-y resource
-A address address
-C channel
-D
-G
-H
-I address address
-J address address
-M maximum minimum
-O
-P
-R
-S
-T interrupt__scheme
-U number__of__subdevices
```

-V interrupt\_\_vector  
-Y  
-Z

由于这些选项很烦琐，因此建议最好使用交互方式来执行 configure。

## cpio

文件归档与恢复

### 语法:

```
cpio-o [ac0BvV][--C bufsize][--O file][--k volumesize][--M message]
cpio-i [BcdmrtTuvVfsSb6k][--C bufsize][--I file][--k volumesize][--M message]
      [pattern...]
```

cpio-p [adlmuvV] directory

### 说明:

cpio-o(copy out)从标准输入设备读取一路径名，并把那些文件一起拷贝到标准输出设备上，同时给它分配一个路径名和状态信息。按照缺省设置，所输出的内容被送到一个 512 个字节的区域中。

cpio-i(copy in)从标准输入设备取得文件,这些文件可以假设是从前面的 cpio-o 获得的。只有那些与 patterns 匹配的文件名所对应的文件才能被选取。patterns 是在 sh(C)的文件名生成符号中给出的正则表达式。在 patterns 中，元字符?, \*, 和 [...] 与斜线(/)字符相匹配，反斜线(\)是逃逸字符。元字符!“!”表示“非”。(例如,! abc \* 表示包括所有不以 abc 开头的文件。)可以指定多个 patterns，如果不指定 patterns，patterns 的缺省值就是 \* (即选取所有的文件)。每个 patterns 都必须放在双引号中，否则就当作当前目录中的一个文件名使用。所选取的文件都根据下面所说明的各选项被有条件地建立并被拷贝到当前的目录树中。文件的权限与前面 cpio-o 的相同。文件的所有者和用户组与当前用户相同，除非当前用户是超级用户，如果当前用户是超级用户，则取前面 cpio-o 文件的所有者和用户组。请注意，如果 cpio-i 企图建立一个已经存在的文件，并且这个现存文件与所要建立的文件年龄相同或更小，cpio 就会输出一条警告信息，同时也不会替换这个文件。(-u 选项可以无条件地对已存在的文件进行重写。)

cpio-p (pass)从标准输入设备读取文件的路径名，这些文件都是根据下面所说明的选项有条件地建立并被拷贝到指定 directory 中的。由 cpio 建立的文本文件的档案文件可在各种 UNIX 系统 V 之间移值。

下面是一些常用的选项:

- B 输入 / 输出以 5120 字节为一个记录。当不用这个选项和 -C 选项时，缺省设置为 512 字节。  
(-B 不能用于 pass 选项; -B 只对字符设备有意义。)
- c 以 ASCII 字符形式写头信息，为移植服务。(当源和目的机器不属于同一类型时，总使用这个选项。)
- C bufsize  
输入 / 输出以 bufsize 字节为一个记录，其中 bufsize 是一个正整数。当这个选项和

- B 选项都没用时，缺省设置为 512 字节。（-C 不能用于 pass 选项；-C 只对字符设备有意义。）当它与 -K 选项一起使用时，bufsize 被强制设定为 1K 的倍数。
- d 根据需要建立目录。
  - f 不仅拷贝符合 patterns 的文件，还拷贝其它的所有文件。
  - I file  
读 file 的内容作为输入。如果 file 是字符设备，当第一个存储介质满了，换一个介质并敲入回车，继续拷贝。只用于 -i 选项。
  - k 越过可能遇到的坏文件头或 I/O 错误。当想从一个被损坏的或无序的介质中拷贝文件时，这个选项使你只读那些有好的文件头的文件。只与 -i 选项一起用。
  - K volumesize  
指定介质卷的大小。一定以 1K 的块为单位。例如，一个 1.2MB 的软盘的 volumesize 为 1200。一定要包括带 1K 的 bufsize 倍的 -C 选项。
  - M message  
定义当换介质时用的信息。当使用 -O 或 -I 选项并指定字符设备时，可以用这个选项指定当到达介质末尾时应显示的信息。
  - O file  
指示 cpio 的输出文件 (file)。如果 file 是个字符设备，当第一个介质满了，换一介质并敲入回车，继续拷贝。只与 -o 选项一起使用。
  - r 交互式地从文件命名。如果用户敲入一个空行，该文件就被越过。如果敲入“.”，就还用原来的路径名。（不能用于 cpio-p）。
  - t 显示输入的内容表。不建立文件。
  - u 无条件拷贝（一般情况下，一个老文件不能替换同名的新文件）。
  - v 显示文件名。当与 -t 选项一起用时，所显示内容表与 Ls-L 命令的输出类似。
  - V 为每个可见文件显示一个点。用于使用户在使用 cpio 时不必显示所有的文件名
- 

**注意：**

当使用 cpio 读或写字符设备时，若没有用 -O 和 -I 选项，如果 cpio 到达介质末尾，就会显示：

If you want to go on ,type device / file name when ready.要想继续，必须换介质，敲入这个字符设备名（例如 / dev / rdisk / f0f09q15dt）并按回车键。

---

**custom**

安装 UNIX 系统的特定部分

**语法：**

```
custom [-od] [-irla[package]] [-m device][ -f [file]]
```

**说明：**

可以用 custom 按不同的要求在 UNIX 系统上有选择地安装或删除特定的部分。custom 只能由超级用户执行，可以由接口程序调用也可以以命令行中带几个选项的形式调用。

文件可以被取到软件包 (packges) 中，也可以从中删除。每个软件包都是文件个体

的集合。可以按下面所说明的那样使用 custom 命令列出软件包中的文件。

使用:

要想以交互方式使用 custom, 可输入:

custom

custom 的主菜单显示出如下选项:

#### Install

允许加入一个程序或系统。

首先打开一个窗口, 用来选择系统集或软件。当选了系统或软件之后, 你就被要求选择加入“Entire Product”、“Packages”或“Files”。当选了“Entire Product”之后, custom 就会计算需要哪一个安装卷 (原盘提供的), 然后提示以正确的卷号。

如果选了“packages”, 就会显示所有软件包。每行描述一个软件包的名称, 这个包是完全安装、部分安装还是没安装, 这个包的大小 (以 512 字节块为单位), 以及这个软件包中内容的一行描述。

所指定的多个软件包可以用空格键对它们标记。所选中的软件包都带有星号 (\*) 显示。执行时, custom 将提示插入所需的介质卷。(如果 RTS 软件包已经部分安装了, 就不能再用 custom 安装整个的 RTS 软件包了。如果遇到这种情况就必须用 fixperm (ADM) 来确定一下丢失了哪些文件, 然后再用 custom 分别安装各文件。)

如果选了“Files”, 就会被提示选择软件包和文件名。custom 然后就会提示卷号。

#### Remove

删除指定软件包 / 软件中的适当文件。选择要删除的软件包或软件的方法与选择要安装的软件或软件包的方法相同。

#### List

列出指定软件包中的所有文件或一个软件产品集中的所有软件包。

#### Quit

退出 custom

选项:

完整的非交互式地使用 custom 需要三个变元:

- 一个系统集标识符 (-o 或 -d)
- 一个命令 (-i, -r, -l, -f, 或 -a)
- 以及一个或多个软件包名, 或一个文件名。

如果在命令行中缺少某信息, custom 就会提示给出这个没给的参数。

只能指定 -o 或 -d 中的一个。它们分别代表:

-o: 操作系统。

-d: 开发系统。

只能指定 -i, -r, -l, -f 或 -a 中的一个, 在其后跟一个适当类型的变元 (一个或多个软件包名, 或一个文件名)。这些选项执行如下功能:

- i: 安装指定的软件包
- r: 删去指定的软件包
- l: 列出指定软件包中的文件

-f: 安装指定文件

-a: 增加一个新的软件产品

-m 标志允许指定介质设备。缺省设备为 /dev/install (总为 0 设备, 例如 /dev/fd0)。如果系统在 /dev/fd0 上有一个 5.25 英寸软盘驱动器, 在 /dev/fd1 上有一个 3.5 英寸软盘驱动器, 当需要安装 3.5 英寸盘时, 这个标志就非常有用。例如:

```
custom -m /dev/rfd196ds9
```

这就忽略缺省设备而使用由 -m 标志所指定的设备。

---

注意:

如果想对系统的某一部分进行升级, custom 就会检查一下你是否有一个不同的版本, 并提示你插入适当的软盘。同样, 如果你插入的软盘不对, custom 会提示你重新插入正确的软盘。在安装操作系统的过程中, 一般总是完整地安装 RTS 软件包。

---

### dbmbuild(ADM):

建立 MMDF 哈希数据库(hashed database)

语法:

```
/usr/mmdf/table/dbmbuild[-nvdk][database[table...]]
```

说明:

dbmbuild 程序将在 MMDF 的剪裁文件中指定的表项 (table) 读入到哈希数据库 (hashed database) 中, 在对提交的消息进行快速地址检验以及有效地指定通道时会用得着。无论何时以何种方式改变了 MMDF 别名或路由 (routing) 信息, 都必须重建哈希数据库。具体作法是: 首先以 mmdf 身份注册, 然后从 /usr/mmdf/table 目录运行 dbmbuild 程序。

如果没有指定 database 文件, 则用缺省文件 mmdfdbm。如果没有给出 table 文件, 则使用剪裁文件中列出的所有文件。特别地, 对每个通道定义都应读入三个表项: 经授权的“源”清单, 经授权的“目的”清单, 以及该通道的名字/别名表。另外, 保留表项 (MTBL 和 MDMN) 也应读入。

任选项有:

n: 建立新的 database。如果省去了该选项, 则 dbmbuild 程序将现有的 database 更新。如果没有给出任何任选项, 则假定有 -n 选项。但如果给出了某一选项 (如 -v, 则必须指定 -n 选项才能建立新的 database。

v: 在冗长方式下运行, 即表处理过程中将信息显示到屏幕上。

d: 在调试 (debug) 方式下运行, 报告所发生的一切。

k: 保持运行。如果用到根本不存在的文件, 可加以忽略。在有的系统中该任选项是缺省设置值。

在 database 上加了适当的锁机制, 这样当 MMDF 操作时 dbmbuild 也可安全地运行。

文件:

```
/usr/mmdf/mmdftailov
```



```

/usr / mmdf / table / alias.list
/usr / mmdf / table / alias.user
/usr / mmdf / table / * .chn
/usr / mmdf / table / * .dom
$(tblldb).div      database 目录
$(tblldb).pag      database 页
$(tblldb).lck      database 锁文件
$(tblfldiv) / *    形成 database 的不同表项

```

## df

报告空闲磁盘块的数目。

### 语法:

```
df[-t][-f][-v][filesystems]
```

### 说明:

df 通过检查在线文件系统超级块中的计数器，显示出空闲块和空闲 i 节点的数目。filesystems 可以由设备名（例如 / dev / root）来指定。如果 filesystems 这个变元没有指定，所有被安装的文件系统的空闲空间都会被送到标准输出。被安装的文件系统由 / etc / mnttab 给出。

### 选项包括:

- t 报告所分配的总块数和空闲块数。
- f 只报告空闲表中的实际块数（不报告空闲 i 节点）。带有这个选项，就只对原始设备进行报告。
- v 报告所使用块的百分比以及使用了的和空闲的块数。  
-v 不能与 df 的其它选项一起使用。

注意：这个程序所报告的块是 512 字节，但文件所使用的系统块是 1024 字节。

## deliver(ADM)

MMDF 邮件传递命令

### 语法:

```
deliver [-bdpsw] [-cchan,chan] [-lmins] [-thrs] [-mmaxsort] [-Llogfile]
[-Tsecs] [-Vloglevel][message1...messageN]
```

### 说明:

在 MMDF 邮件系统中由 deliver 程序处理所有邮件的传递工作。deliver 程序并不直接传递邮件，而是访问 MMDF 通道来执行实际的传递工作。deliver 的操作由 MMDF 剪裁文件 / usr / mmdf / mmdftailor 和命令行任选项来控制。该程序可以作为守护程序 (daemon) 运行，也可由用户启动。可以调用该程序来处理整个邮件队列，或者只处理一些指定名字的消息。在可能的情况下，deliver 总是试图以接收到的顺序来处理消息。deliver 程序还在每个通道上保存着各机器的有关信息，使得邮件传递时跳过不可访问的机

器。

deliver 首先通过命令行，或者根据系统中的所有非被动 (non-passive) 通道建立要处理的通道清单。然后收集需要处理的消息清单，这可以通过命令行得到，或者扫描每个通道的邮件队列亦可得到消息清单。如果某一通道的邮件队列中消息数超过了 maxsort (该值在命令行或剪裁文件中给出)，则在处理该通道中队列目录时以读入的顺序进行，而不按邮件提交的时间分类。如果在命令行中给出了消息清单，则不作任何分类，而将邮件以命令行中给出的顺序加以传递分类的关键字 (keys) 依次是：通道、提交时间和机器。这么做使消息有多条访问途径，但使通道程序的的开销增加。

使用下面的任选项可改变 deliver 程序动作：

-b: 后台方式。使 deliver 以后台控制程序运行，周期性地扫描邮件队列，找到尚未传递的邮件加以传递。只有 MMDF 用户和超级用户有权使用该任选项。deliver 程序首先传递邮件队列中某一消息，然后 sleep，接着又重复这一过程。缺省的 sleep 时间是 10 分钟。但这也是可以改变的。-c 通道 1, 通道 2, ...: 通道选择。要处理的通道清单。各通道之间以逗号相隔。

-d: 本选择项使 deliver 假定自己已处于邮件队列中，因而不再发外部 chdir() 命令。这主要用于测试。

-lminutes: 设置“已死机器” (dead-host) 存储器的条目的“生存时间”，已死机器存储器是用来防止邮件传递到已不工作的机器上的。“生存时间”以分钟为单位给出。如果该值为负，则使已死机器存储器无效。缺省值为 2 小时。

-mmaxsort: 设置分类阈值。如果某一通道的队列中消息数超过 maxsort 值，则在处理消息时以目录中的顺序进行，而不按提交消息的时间早晚排序。如果 -m 没有指定，则 maxsort 的值在剪裁文件中由 MMAXSORT 给出。

-S: 强制对邮件队列进行线性搜索。在正常情况下 deliver 程序以收到消息的顺序传递消息，这一顺序与目录中的顺序一般是不一样的。如果邮件队列非常大，deliver 不能在可以接受的时间内对队列进行排序，此时即可用 -S 任选项。

-thrs: 时间限制。本任选项不让 deliver 程序传递在队列中停留时间超过 hrs 的邮件。考虑到效率，此选项仅用于经排序过的队列。如果在命令行上直接给出了要传输消息的清单，或者 -S 任选项已设置，或者队列中消息数据超过 maxsort 值 (见 -S 任选项)，则不应用本任选项。

-W: 观察传递过程。使 deliver 程序在传递的同时将指示性信息打印到标准输出上。

-Llogfile: 将传递过程的登记文件指定为某一文件。缺省值是 MMDF 登记目录中的文件 msg.log。本任选项只有超级用户和 MMDF 可用。

-Tseconds: 设置后台对邮件队列进行扫描的时间间隔。缺省值是 10 分钟。

-Vloglevel: 将本次传递的登记级别设置为指定值。loglevel 应该是一个正确的 mmdf 登记级别串，比如 FTR。本选项仅超级用户和 MMDF 可用。

## divvy(ADM)

磁盘划分实用程序

语法：

```
divvy -b block__device -C character-device [-v virtual__drive]
[-p physical__drive][-i][-m][-n]
```

说明:

divvy 将一个 fdisk (ADM) 分区划分成一系列独立的区域称为“division”。“division”由各不相同的主设备号和次设备号进行标识, 可用于文件系统, 对换空间, 或用来将设备上的坏块隔离开。

使用 divvy, 可以:

- 将磁盘或 fdisk 分区划分成独立的设备
- 建立新的文件系统
- 改变文件系统大小
- 删除文件系统

任选项:

divvy 的任选项有:

-b 块设备

块接口的主设备号。

-c 字符设备

字符接口的主设备号。

-v 虚拟设备

用于划分虚拟磁盘

-p 物理设备

用于划分共享同一控制器的几个物理磁盘之一。

-i 只用于安装。被划分的磁盘在 0 号 division 包含根文件系统。

-m 被划分的磁盘应包含若干个可安装文件系统。

-n 只用于安装。非交互式任选项。被划分的磁盘应包含以下内容:

在 0 号 division 的根文件系统

在 1 号 division 的对换空间

在 2 号 division 的 U 文件系统

在 3 号 division 的 scratch 文件系统

用法:

被划分的设备必须是带字符接口的块设备。比如, 要用 divvy 来划分一个块接口号为 1 且字符接口号也为 1 的设备, 应打入:

```
divvy -b 1 -c 1
```

-v 任选项指示划分的是哪一个虚拟盘, 缺省值为当前活动盘。这里, “虚拟盘”指的是 MS-DOS 分区。虚拟盘号由 fdisk (ADM) 实用程序指定。

-p 选项允许对共享一个控制器的几个物理磁盘之一进行划分。divvy 认为第一个物理设备号为“0”。若要访问第 2 个物理磁盘, 应用 -p1 选择项。

-i 任选项用于安装。它指示被划分的设备将包含根文件系统。用此任选项, 可相对于新的根——通常是硬盘, 而不是当前根, 建立设备节点。建立根文件系统, 对换区域和恢复区域 (recover area)。divvy 提示询问对换区的大小。如果磁盘足够大, 则 divvy 提

示建立独立的 /u (用户) 文件系统。如果根文件系统很大, 需要使用 scratch 文件系统, 则 divvy 会询问是否建立一个 scratch 文件系统。

-m 选择项是在初次安装时不会用作 root 的设备上使用的。

当 divvy 通过命令行启动之后, 会看到如下的主菜单:

```
n [ame] (对 division 命名或者重新命名。)
c [reate] (在本 division 上建立一个新的文件系统。)
t [ype] (对新文件系统选择或修改文件系统的类型。)
p [revent] (禁止在本 division 上建立新的文件系统。)
s [tart] (不同块上开始一个 division。)
e [nd] (在不同块上结束一个 division。)
r [store] (恢复原始 division 表)
```

please enter your choice or 'q' to quit:

要选择某一命令, 应打入该命令的第一个字母, 并按下回车键。

divvy 的 division 表类似:

Name	Type	NewFS	#	First Block	Last Block
root	XENIX	no	0	0	47402
swap	NONFS	no	1	47403	50368
u	XENIX	no	2	50369	70368
	NOT USED	no	3	—	—
	NOT USED	no	4	—	—
	NOT USED	no	5	—	—
recover	NON FS	no	6	70369	70378
hd0a	Whole DISK	no	7	0	70676

divvy 还显示系统表格的块分配情况及坏道信息。

可以用 'n' 命令改变设备名。divvy 会提示输入 division 号 (从上面显示的 divvy 表中可得), 然后要求输入新的名字。

使用 'C' 命令使得指定 division 在你从 divvy 退出时变成新的空文件系统。在用 'C' 命令之后, 可在 'New File System?' 一栏中看见 'yes' 字样。这时如果用 'P' 命令, 这一栏中的 'yes' 即变成 'no', division 的内容不会发生变化。'C' 命令必须用于改变文件系统大小时。

用 'S' 或 'start' 命令, 可从指定块号处开始一个 division, 用 'e' 或 'end' 命令, 则从指定块号处结束一个 division。

可以用这些命令来改变分区的大小。比如, 若你的磁盘与前面 divvy 表所表示的磁盘类似, 你希望让 u 文件系统再大些, 对换区域小些, 则可以按以下步骤进行:

- 用 'e' 命令使对换区域变小。
- 用 's' 命令使 u division 变大。
- 用 'c' 命令重建 u 文件系统。

如果出现 division 重叠的情况, 则在你试图退出 divvy 时会自动返回 divvy 主菜单并报告所出现的错误, 这时你可加以改正, 然后再退出。

'Y'或'restore'命令可恢复原始的分区表。当你在使用 divvy 时犯了严重的错误, 希望返回到初始状态时即可使用该命令。

在退出 divvy 时, 系统会询问是保存所作的修改还是不用修改直接退出, 这时你也可以返回 divvy 菜单。如果你选择保存所作的修改, 则新的分区表将写至硬盘上, 由'C'命令指定的新文件系统就建立了。

注意: 如果 divvy 显示磁盘大小为'0'块, 或者显示:

```
cannot read division table
```

或

```
cannot get drive parameters
```

则可能因为在 divvy 之前运行的 fdisk,badtrk 等不正确所致。

### **haltsys, reboot**

关闭文件系统并停止系统

语法:

```
/etc/haltsys [-d]
```

```
/etc/reboot
```

说明:

haltsys 实用程序的作用是关闭挂起的磁盘 I/O, 将文件系统标记为“干净的”, 最后停止处理器执行。haltsys 命令是立即生效的, 所以用户进程必须提前“杀死”。考虑到这一点, 对于一般情况下的系统关闭, 推荐使用 shutdown (ADM) 命令。只有在由于某种原因 (比如出现系统故障时) 才使用 haltsys 命令。

haltsys 显示提示信息指出系统已经关闭, 可以重新启动或者关掉电源。如果使用了 -d 任选项, 则系统保持关闭状态, 不能重新启动。(当然, 关掉电源然后再打开是可以启动的。)

reboot 命令与 haltsys 命令的功能完全一样, 只是系统随后自动地重新启动。

只有超级用户有权使用 haltsys 和 reboot 命令。

### **uuxqt(ADM)**

执行远程命令请求

语法:

```
/usr/lib/uucp/uuxqt [-s system][[-x debug_level]
```

说明:

uuxqt 是执行来自远程系统的作业请求的程序。远程作业请求是由 uux 命令产生的。uuxqt 程序在 spool 目录中搜索 X 文件。对于每一个 X 文件, uuxqt 都检查是否所有需要的数据文件都是可以访问的, 以及提出请求的远程系统是否有权执行所提请求。Permissions 文件就是用来检验文件的可访问性和命令执行权限的。

在执行 uuxqt 命令之前有两个环境变量需要加以设置:

UU\_MACHINE 是发送作业的机器。

UU\_USER 是发送作业的用户。

-x debug\_level 是 0 至 9 的一位数字，其值越大表明提供的调试信息越详细。

文件：

/usr/lib/uucp/Permissions

/usr/lib/uucp/Maxuuxqts

/usr/spool/uucp/\*.

## hd

内部硬磁盘的驱动

说明：

对主硬盘的块缓冲访问是通过如下块特殊文件进行的：hd00, hd01 到 hd04, hd0a 和 hd0d, root 和 swap。对第二硬盘的块缓冲访问是通过如下块特殊文件进行的：hd10, hd11 到 hd14, hd1a。

hd00 代表整个物理磁盘，hd01 到 hd04 代表 fdisk 分区，root 代表根文件系统，swap 代表对换区。块特殊文件通过系统的正常缓冲机制访问磁盘，并不受物理磁盘记录大小的限制对磁盘进行读写。

字符特殊文件的命名规则与块特殊文件的相同，只是字符特殊文件以“r”开头。例如，代表整个物理磁盘的字符特殊文件是 /dev/rhd00。

下面是固定磁盘分区的名字。每个分区都可以通过一个块接口进行访问，例如 /dev/hd01 或通过一个字符（原始）接口进行访问，例如 /dev/rhd01。

上面的设备遵循 XENIX 的命名规则。相应的 UNIX 设备可在 /dev/dsk（块）和 /dev/rdisk（字符）目录中找到。在下面的表中，XENIX 和 UNIX 设备都列出来了。

Device File Names for Fixed disks

First Controller		Second Controller		Partition
Disk1	Disk2	Disk1	Disk2	
/dev/hd00	/dev/hd10	/dev/dsk/4s0	/dev/dsk/5s0	entire disk
/dev/rhd00	/dev/rhd10	/dev/rdsk/4s0	/dev/rdsk/5s0	
/dev/hd01	/dev/hd11	/dev/dsk/4s1	/dev/dsk/5s1	first partition
/dev/rhd01	/dev/rhd11	/dev/rdsk/4s1	/dev/rdsk/5s1	
/dev/hd02	/dev/hd12	/dev/dsk/4s2	/dev/dsk/5s2	second partition
/dev/rhd02	/dev/rhd12	/dev/rdsk/4s2	/dev/rdsk/5s2	
/dev/hd03	/dev/hd13	/dev/dsk/4s3	/dev/dsk/5s3	third partition
/dev/rhd03	/dev/rhd13	/dev/rdsk/4s3	/dev/rdsk/5s3	
/dev/hd04	/dev/hd14	/dev/dsk/4s4	/dev/dsk/5s4	fourth partition
/dev/rhd04	/dev/rhd14	/dev/rdsk/4s4	/dev/rdsk/5s4	

Device File Names for Fixed disks

First Controller		Second Controller		Partition
Disk1	Disk2	Disk1	Disk2	
/dev/hd0a /dev/rhd0a	/dev/hd1a /dev/rhd1a	/dev/dsk/4sa /dev/rdisk/4sa	/dev/dsk/5sa /dev/rdisk/5sa	active partition
/dev/hd0d /dev/rhd0d	/dev/hd1d /dev/rhd1d	/dev/dsk/4sd /dev/rdisk/4sd	/dev/dsk/5sd /dev/rdisk/5sd	DOS partition
/dev/root /dev/rroot				root file system
/dev/swap /dev/rswap				swap area

请注意，root 和 swap 文件只在根磁盘上存在。

要访问 DOS 分区，就要用像“C:”或“D:”这样的字母来表示第一或第二分区。在文件 /etc/default/msdos 中包含为 DOS 设备名分配的简写字母。

下面的表列出了硬盘特殊文件次设备号定义。请注意块特殊设备与字符特殊设备共用同一个次设备定义。次设备号定义如下：第 7 和第 6 位表示物理驱动器，第 5 至第 3 位表示虚 (fdisk) 分区，而第 2 至第 0 位表示 divvy 分区。

Minor Device Bits

Phys.	Virtual	divvy	Device special	Description
7 6	5 4 3	2 1 0	file name	
0 0	0 0 0	0 0 0	/dev/hd00	whole PD0
0 1	0 0 0	0 0 0	/DEV/HD10	Whole PD1
1 0	0 0 0	0 0 0	/dev/dsk/4s0	whole PD2
1 1	0 0 0	0 0 0	/dev/dsk/5s0	whole PD3
0 0	0 0 1	1 1 1	/dev/hd01	PD 0,whole VD1
0 0	0 1 0	1 1 1	/dev/hd02	PD 0,whole VD2
0 0	0 1 1	1 1 1	/dev/hd03	PD 0,whole VD3
0 0	1 0 0	1 1 1	/dev/hd04	PD 0,whole VD4
0 0	1 0 1	1 1 1	/dev/hd0a	PD 0,whole active VD
0 0	1 1 0	1 1 1	/dev/hd0d	PD 0,whole DOS VD
0 0	1 0 1	0 0 0	/dev/root	PD 0,active virtual,DP 0
0 0	1 0 1	0 0 1	/dev/swap	PD 0,active virtual,DP 1
0 0	1 0 1	0 1 0	/dev/usr	PD 0,active virtual,DP 2
0 1	1 0 1	1 1 0	/dev/recover	PD 0,active virtual,DP 6
0 1	0 0 1	1 1 1	/dev/hd11	PD 1,whole VD 1
0 1	0 1 0	1 1 1	/dev/hd12	PD 1,whole VD 2
0 1	0 1 1	1 1 1	/dev/hd13	PD 1,whole VD 3
0 1	1 0 0	1 1 1	/dev/hd14	PD 1,whole VD 4

Minor Device Bits				
Phys.	Virtual	divvy	Device special	Description
7 6	5 4 3	2 1 0	file name	
0 1	1 0 1	1 1 1	/dev/hd1a	PD 1,whole active VD
0 1	1 1 0	1 1 1	/dev/hd1d	PD 1,whole DOS VD
0 1	1 0 1	0 0 0	/dev/u0	PD 1,active virtual,DP 0+
0 1	1 0 1	0 0 1	/dev/u1	PD 1,active virtual,DP 1+
0 1	1 0 1	0 1 0	/dev/u2	PD 1,active virtual,DP 2+
1 1	0 0 1	1 1 1	/dev/dsk/4s1	PD 2,whole VD 1
1 1	0 1 0	1 1 1	/dev/dsk/4s2	PD 2,whole VD 2
1 1	0 1 1	1 1 1	/dev/dsk/4s3	PD 2,whole VD 3
1 1	1 0 0	1 1 1	/dev/dsk/4s4	PD 2,whole VD 4
1 1	1 0 1	1 1 1	/dev/dsk/4sa	PD 2,whole active VD
1 1	1 1 0	1 1 1	/dev/dsk/4sd	PD 2,whole DOS VD
1 1	0 0 1	1 1 1	/dev/dsk/5s1	PD 3,whole VD 1
1 1	0 1 0	1 1 1	/dev/dsk/5s2	PD 3,whole VD 2
1 1	0 1 1	1 1 1	/dev/dsk/5s3	PD 3,whole VD 3
1 1	1 0 0	1 1 1	/dev/dsk/5s4	PD 3,whole VD 4
1 1	1 0 1	1 1 1	/dev/dsk/5sa	PD 3,whole active VD
1 1	1 1 0	1 1 1	/dev/dsk/5sd	PD 3,whole DOS VD
<b>KEY</b>	VD = virtual drive DP = divvy partition		PD = physical drive + = user-defined name	

设备文件 `usr` 和 `u[0-2]` 是可选文件系统名；这些文件系统一般不存在，除非由系统管理员建立。

注：在第一个磁盘上，`hd00` 代表整个磁盘，用于访问主引导块，其中包括 `fdisk` 分区表。对于第 2 个磁盘，`hd10` 代表整个磁盘，并被用于访问其 `fdisk` 分区表。不要对 `hd10` 和 `hd00` 进行写操作。

### integrity

依据鉴定数据库检查系统文件。

语法：

```
integrity[-v][-e][-m]
```

说明：

`integrity` 遍历文件控制数据库 (File Control database)，依次把其中的每一个记录与文件系统中的实际文件进行比较。如果发现两者的所有者，小组或访问权限有差异，就会输出一个错误信息。

-v 选项会列出所考虑的所有文件。

-e 选项解释检查失败的原因，以及差异到底在哪里。



一般情况下，对于在文件控制数据库中有，而在文件系统中没有的文件不做报告。但 `-m` 选择会对此进行报告。

### **link, unlink**

对文件和目录进行连接和解除连接

语法:

```
/etc/link file1 file2
```

```
/etc/unlink file
```

说明:

`link` 命令可用来建立一个指向另一个文件的文件名。所连接的文件和目录可以由 `unlink` 命令解除连接；然而，最好使用 `rm (C)` 和 `rmdir (C)` 命令来代替 `unlink` 命令。

`ln (C)` 和 `link/unlink` 之间的唯一区别在于后者只做它们被要求做的事，而不做任何错误检查，这是因为它们直接调用 `link (S)` 和 `unlink (S)` 系统调用。

### **link\_\_unix**

建一个新的 UNIX 系统核心

语法:

```
/etc/conf/cf.d/link__unix
```

说明:

在安装一个设备驱动程序之后，使用 `link unix` 建新的 UNIX 核心。这个程序使用 `/etc/conf` 中的当前系统配置建立 `/etc/conf/cf.d/link__unix`。

### **Lpsched(ADM),Lpshut(ADM),Lpmove(ADM)**

启动/停止打印服务及移动打印请求

语法:

```
/usr/lib/lpsched
```

```
/usr/lib/lpsched -q integer
```

```
/usr/lib/lpsched -a integer
```

```
/usr/lib/lpsched -p integer
```

```
/usr/lib/lpsched -s integer
```

```
/usr/lib/lpshut
```

```
/usr/lib/lpmove requests dest
```

```
/usr/lib/lpmove dest1 dest2
```

说明:

`lpsched` 程序启动 `lp` 打印服务，这只能由超级用户和 `lp` 来执行。

`lpshut` 关闭打印服务。在启动 `lpshut` 时正在打印的所有打印机都将停止打印。在 `lpsched` 再次启动时，在打印机关闭时正在执行的打印请求将从头开始重新执行。

`lpmove` 移动排在队列中等待 `lp` 服务的打印请求。该命令的第一种形式是将指定名字的请求移至目的的打印机 `dest` 上。请求是以 `lp` 返回的 `request-ids` 来标识的，第二种形式

将打印机 dest1 的所有请求移至打印机 dest2；然后打印机 dest1 拒绝接受任何新的打印机请求。

在移动打印请求时，lpmove 程序不检查指定的目的打印机的接收状态，同时也不改变被移动请求的请求 ID 值。

Lpmove 所用到的任选项如下：

- q integer: 指示所分配的请求结构数目。
- a integer: 指示想要分配的提醒结构 (alert structure) 数目。在给系统中的每一台打印机分配一个提醒结构之外，所分配的空提醒结构数的缺省值为 40。用 -a 任选项可改变该数值。
- P integer: 指示想要分配的打印状态结构数目。在给系统中每一台打印机分配一个打印机状态结构之外，所分配的空状态结构数目的缺省值为 25。可用 -p 任选项修改该值。
- S integer: 指示每台打印机的可同时运行的慢速滤通程序数目。

---

注意：/usr/spool/lp 被当作存放 lp 打印服务所用到的所有文件的缺省目录。在运行 lpsched 之前修改 Spool\_DIR 环境变量可将其改变为另一个目录。

---

文件：

/usr/spool/lp/\*

## Lpsh

菜单驱动的 lp 打印服务管理实用程序

语法：

/usr/lib/sysadm/lpsh

说明：

lpsh 是由 sysadmsh(ADM)选择 Printers 一项所启动、用以管理打印服务的屏幕接口。该程序允许系统管理人员执行下面的任务：

- 配置 LP 打印服务以描述打印机和设备。
- 管理 LP 打印服务使用的滤通程序。
- 管理 LP 打印服务使用的表格 (form)
- 启动 LP 打印服务。
- 关闭 LP 打印服务。
- 在打印机之间移动打印请求。
- 删除打印请求。
- 允许打印机接受或拒绝打印请求。
- 设置分配给 LP 打印服务用户所提交作业的打印队列优先级。
- 使打印机有效 / 无效。

---

注意：直接启动 lpsh 方法不值得提倡，应使用 sysadmsh 的 Printers 选项。

---

## Lpusers(ADM)

设置打印队列优先级

### 语法:

```
/usr/lib/lpusers -d priority-level(优先级别)
/usr/lib/lpusers -q priority-level(优先级别)-u user-list(用户表)
/usr/lib/lpusers -u user-list(用户表)
/usr/lib/lpusers -q priority-level(优先级别)
/usr/lib/lpusers -l
```

### 说明:

lpusers 命令用于限制分配给用户所提交作业的优先级别。

命令的第一种形式（带-d 任选项）将系统范围之内作业的优先级设置为 priority-level，这里 priority-level 是从 0 至 39 的数值，0 表示最高优先级。若用户没有指定某一打印请求的优先级，则使用缺省值 20。

命令的第二种形式（带任选项-q 和-u）设置在用户表中的用户在提交作业时所能请求的最高优先级。若用户请求的优先级高于此设置值，则实际请求得到的优先级与本设置值相等。

命令的第三种形式（带任选项-u）使额外为用户指定的优先级无效，仍然用缺省优先级。

命令的第四种形式为在用命令第二种形式没有涉及的用户指定最高优先级。

命令的最后一种形式（带任选项-l）列出分配给用户的优先级缺省值及最高优先级。

## Lp(HW)

行式打印机设备接口

### 说明:

Lp0, Lp1, Lp2 提供对打印机任选的并行口的访问。Lp0 和 Lp2 文件分别提供对并行端口 1 和 2 的访问，Lp1 提供对单色适配器上的并行口的访问。

在给定的系统上只能用 Lp0 和 Lp1 之中的一个。要访问一个系统上的两台打印机，应使用 Lp0 或 Lp1 及 Lp2。

文件:

```
/dev/Lp0
/dev/Lp1
/dev/Lp2
```

## mkdev(ADM)

增加外围设备

### 语法:

```
/etc/mkdev dos
/etc/mkdev fs
```

```
/etc/mkdev fs [device file(设备文件)]
/etc/mkdev hd [[disk][controller 1 adapter]][lun]
/etc/mkdev mouse
/etc/mkdev serial
/etc/mkdev shl
/etc/mkdev streams
/etc/mkdev tape
```

#### 说明:

mkdev 建立与外围设备有关的设备文件。根据所提供的参数，mkdev 调用 /usr/lib/mkdev 目录中的一个控制程序。如果没有给出参数，则 mkdev 印出命令的用法说明。/etc/mkdev dos 初始化必要的设备，并对系统进行配置，使之能够支持安装的 DOS 文件系统。

/etc/mkdev hd 建立用于外围硬盘的设备文件。内部硬盘的设备文件已经存在，不必再次建立。ST506 磁盘用下列命令之一进行安装：

```
mkdev hd 0 0    第一个控制器上的第一个磁盘
mkdev hd 1 0    第一个控制器上的第二个磁盘
mkdev hd 0 1    第二个控制器上的第一个磁盘
mkdev hd 1 1    第二个控制器上的第二个磁盘
```

ESDI 磁盘则用下列命令之一安装：

```
mkdev hd 0 0    控制器上的引导磁盘
mkdev hd 1 0    控制器上的第二磁盘
```

SCSI 磁盘需要用到 3 条信息：驱动器号，适配器号以及逻辑部件号 (LUN)。SCSI 硬盘的安装将用到下面的命令之一：

```
mkdev hd [0~7][0-1][0-7]
mkdev hd [0-7][SCSI-0 或 SCSI-1][0-7]
```

在安装 SCSI 磁盘时必须两次用到 mkdev hd 命令。第一次，用来重新配置操作系统核心以支持新的磁盘。第二次则实现磁盘初始化。两次调用 mkdev hd 所用的参数是相同的。

/etc/mkdev serial 建立串行卡使用的设备文件。第一个和第二个端口对应的设备文件已经存在，若要对系统增加扩展卡则必须另外建立相应端口的设备文件。

/etc/mkdev fd 命令用来建立可启动的、带有根和文件系统的磁盘。执行一次 mkdev fd 命令可建立多张软盘，但命令本身并不提示你取出第一张，再放入下一张。可在 mkdev 命令显示“Would you like to format the floppy first”时插入下一张软盘。

/etc/mkdev tape 配置磁带驱动器，为重新连接操作系统核心使之支持磁盘作准备。

SCSI 磁带也可以用 mkdev hd 命令进行安装。在安装时必须指定适配器和逻辑部件号。

/etc/mkdev mouse 对必要的设备进行初始化，并配置系统使之支持鼠标。  
文件：

/usr/lib/mkdev/\*

## mkfs

构造文件系统

### 语法:

```
/etc/mkfs [-yl-n][ -f fstype] special blocks [:inodes][gap inblocks]
```

```
/etc/mkfs special proto [gap inblocks]
```

XENIX 文件系统选项

```
[-s blocks [:inodes]]
```

UNIX 文件系统选项

```
[-b blocksize]
```

AFS 文件系统选项

```
[-c clustersize]
```

### 说明:

mkfs 依据命令行中的指示构造一个文件系统，文件系统名写于命令行中的 special 域。mkfs 实际上只是一个前端，它将根据该文件系统的类型调用相应的 mkfs 版本。-f 选项指定文件系统类型，它可以是如下几类：

AFS (Acer Fast Filesystem)

S51K(UNIX)

XENIX

DOS

其中 AFS 为缺省类型。

#### 1. 标准选项

如果所指定的 special 是一个已经存在的文件系统，在重写数据时要求操作者给以确认。-y (yes) 选项可以省去这个确认，并无条件重写现存数据。-n 选项使得在 special 为一个已经存在的文件系统时，不向用户做任何提问就中止这个命令。检查是这样做的，从目标设备读第一个数据块（第一个数据块是超级块），然后看这些字节是否相同。如果这些字节不相同，那么这些就是有意义的数据并要求给以进行重写的确认。

如果 mkfs 的第二个变元是一个数字串，它就被解释为一个十进制整数作为该文件系统大小的 blocks 值。这是该文件系统将要占的物理盘数据块（512 字节）的数目。如果没有 i 节点的数目，那么缺省值就约等于逻辑数据块的数目除以 4。mkfs 在建文件系统的同时还在其上建一个空目录。但不对引导程序块（第零块）进行初始化。

如果第二个变元是一个可打开的文件名，mkfs 就把它当作原型文件 proto，并从那个文件里取得命令执行的指示。原型文件中包含由空格或换行符分隔的若干标志。下面是原型文件定义的一个例子（行号用于帮助解释）：

```
1 /stand/diskboot
2 4872 10
3 d 777 3 1
4 usr d 777 3 1
```

```

5          sh      -----755 3 1 / bin / sh
6          ken      d-----755 6 1
7          $
8          b0      b-----644 3 1 0 0
9          c0      c-----644 3 1 0 0
10         $
11         $

```

第 1 行是要作为引导程序拷贝到第零块中的文件名。第 2 行指定了该文件系统要占的物理块 (512 字节) 数和文件系统中的 i 节点数。第 3 行至第 9 行告诉 mkfs 包括在该文件系统中的有关文件和目录。第 3 行指定根目录。第 4 至第 6 行和第 8 至第 9 行指定其它目录和文件。在第 7 行中的 \$ 告诉 mkfs 结束它所在的文件系统的这个分枝上的活动, 并在上一层目录上继续工作。第 10 和第 11 行上的 \$ 结束这个进程, 因为后面没有另外的定义了。

文件的定义说明给出了文件的模式、用户 ID、组 ID 和初始内容。内容域的有效说明依模式的第一个字符而定。

文件的模式由一个由 6 个字符组成的字符串而定。第一个字符指定文件的类型。字符 -bcd 分别表示普通文件、块文件、字符文件和目录文件。模式的第二个字符或-, 分别定义设置用户 id 模式或不设置。第三个字符 9 或-, 用来设置或不设置组 id 模式。模式中的其它字符是一个 3 位八进制数, 它给出文件的所有者, 所在小组和其它用户的读、写和执行权限。

在模式之后的两个十进制数标志指出该文件的所有者的用户 ID 和组 ID。

如果该文件是普通文件, 那么定义说明的下一个标志就可能是一个路径名, 这个路径名指出该文件内容和大小的拷贝来源。如果该文件是块文件或字符文件, 后面就会跟着两个十进制数, 分别指定主设备和次设备号。如果该文件是一个目录, mkfs 就会建两个登记项和., 然后向目录中读入文件名和相应文件的定义说明, 作为目录登记项。正如上面提过的这次搜索以标志 \$ 结束。

在此命令两种形式中的 gap inblocks 变元所指定的都是循环间隙和块 / 柱面数。

## 2.XENIX 文件系统选项

-S 选项所带的大小和 i 节点数用来代替原型文件中的大小和 i 节点数。

## 3.UNIX 文件系统选项

-b blocksize 选项指定该文件系统的逻辑块的大小。逻辑块的大小是指在一次 I/O 操作中操作系统读或写的字节数。blocksize 的有效值为 512、1024 和 2048。缺省值是 1024。只有当安装 2K 的文件系统软件包时, 才选择 2048。如果使用了 -b 选项, 它必须放在命令行的末尾。

## 4.AFS 文件系统选项

-C clustersize 选项定义该文件系统簇的大小。这只提供给 AFS; 如果这个选项包含在命令行中, 不管使用了其它什么选项, 所生成的文件系统都是 AFS。

## **mmdf(ADM)**

在本地或通过网络发送邮件

### **说明:**

操作系统使用 MMDF 在本地或通过 Micnet、uucp 或其它提供 MMDF 支持的网络发送邮件。custom 实用程序在本地机器上安装 MMDF，并配置一个发送邮件的基本系统。

MMDF 是一个功能非常强的、可配置的邮件发送系统。MMDF 的配置从 /usr/mmdf/mmdftailor 文件开始，该文件定义了机器名，域名，不同的文件（别名文件，域文件，通道文件），以及其他配置信息。要改变系统上 MMDF 的配置，应作为 mmdf 用户注册，修改配置文件。无论何时以何种方式修改了 MMDF 别名或其它信息，都应重新建立哈希数据库。（hashed database）

### **文件:**

- /usr/mmdf/mmdftailor
- /usr/mmdf/table/alias.list
- /usr/mmdf/table/alias.user
- /usr/mmdf/table/\*.chn
- /usr/mmdf/table/\*.dom
- /usr/spool/mail\*
- /usr/spool/mmdf/...

## **mnt,umnt**

安装文件系统

### **语法:**

- /usr/bin/mnt[-urat][directory]
- /usr/bin/umnt directory

### **说明:**

mnt 除了超级用户之外还允许其它用户使用 mount (ADM) 命令的功能来安装所选取的文件系统。超级用户可以通过 /etc/default/filesys 文件中的特定的登记项来指定允许什么时间、以何种方式安装文件系统。

文件系统的要求与 mount (ADM) 中定义的相同。

umnt 用于把目录 directory 中的以前安装的文件系统拆卸下来。

当系统进入多用户状态时，mnt 就由 /etc/rcshell 程序调用，并带有 -r 选项，也许还有 -a 选项，来安装文件系统。-a 标志用于系统自动引导时。在平时使用时，不要使用这两个选项。

-t 选项可显示 /etc/default/filesys 的内容。

-u 选项强制 mnt 实现 umnt 的功能。

## **mount**

## 安装和拆卸文件结构

### 语法:

```
/etc/mount[-r][-fstyp] special directory
/etc/umount special-device
```

### 说明:

mount 告知系统, 有一个可移动的文件结构在设备 special-device 上。同时把这个文件安装到目录 directory 上。directory 目录必须是一个存在的目录; 它将成为这个新安装的文件结构的根。目录 directory 应该是空的。如果目录 directory 中有文件, 在设备 special-device 被安装时它们就好像被删掉了, 而当 special-device 被拆卸之后它们又会重新出现。

mount 和 umount 命令还要维护一张安装设备表。如果不带任何选项地调用 mount, 就会显示出每个所安装设备的名称、该设备所在的目录、该文件结构是否为只读、以及它安装日期。

-fstyp 这个选项表示所安装的文件系统的类型是 fstyp。如果省去这个选项, fstyp 的缺省设置为 root。

选项 -r 表示被安装的文件为只读文件。对于带有物理写保护的文件结构, 例如带写保护的软磁盘, 都必须以这种办法安装, 否则在更新访问时, 无论是否要对其进行写操作, 都会出现错误。

umount 用于把设备 special-device 上的文件结构卸掉。对于这个文件系统所做的 I/O 都结束, 并且这个文件结构被标志为清除。

---

注: 只有超级用户可以使用 mount 命令。

在安装一个软盘上的文件系统时, 不必每次都使用同一个 directory 目录。然而, 如果每次都使用同一个目录, 那么在每次使用时这些文件的全路径名就都是一致的。

在从软盘驱动器中取出软盘之前必须先对该软盘上的文件系统做 umount 操作。如果不这样做, 在下次安装这个磁盘时就要运行 fsck。

---

## mouse (HW)

### 系统鼠标器

#### 说明:

UNIX 支持直接连在总线控制卡上的鼠标以及连接到标准串行端口的鼠标。命令:

```
mkdev mouse
```

用于配置新的鼠标器或重新配置已有的鼠标器。

文件:

/dev/mouse	与鼠标有关的特殊设备文件目录。
/dev/mouse/bus[0-1]	总线鼠标设备文件。
/dev/mouse/Vpix[0-1]	Vpix-鼠标设备文件。
/dev/mouse/microsoft_ser	Microsoft 串口鼠标设备文件。



## netutil (ADM)

管理 micnet 网络

语法:

```
netutil [option] [-x] [-e]
```

说明:

netutil 命令允许用户建立和维护 UNIX 机器的网络。Micnet 网络是两个或多个系统通过串行线路的连接,用于在系统之间通过 mail (C) 命令传送邮件、使用 rcp (C) 命令传输文件,以及用 remote (C) 命令在远程系统上执行命令。

netutil 命令用来建立及传送实现 Micnet 网络所需的数据文件,还用来启动和停止网络。option 参数是 install, safe, restore, start, stop 之一,或者分别用从 1 到 5 的数字表示。-X 选项登记传输过程,-e 选项登记错误。-X 和 -e 只有在与 start, stop 或者表示这两者的数字 4, 5 联合使用时才有意义。

install 任选项以交互方式建立运行网络所需的数据文件。save 选项将这些文件存于软盘或硬盘上,并可在网络中的其他系统上使用。如果将 micnet 文件存在硬盘上,可以用 uucp (C) 将文件传输到其他机器上。restore 任选项将数据文件从软盘拷贝到系统中。start 任选项启动网络,stop 任选项停止网络。如果 netutil 程序在启动时没有带任选项,则命令会显示一个菜单,你可从中选择一项。在任选项选择好之后,如果还需要用其他信息,则命令会提出要求。

网络在能够启动之前必须进行安装。安装过程包括用 install 任选项建立适当的配置文件。install 任选项要用到网络中每台机器的名字,用于连接机器的串行线路,每条线路的传输速率,以及所有机器上用户的名字。配置文件在建立之后必须用 save 和 instore 任选项分配到网络中的所有计算机上。在网络中的每一台机器上都是用 start 任选项来启动网络的。在启动之后,邮件和远程命令即可通过网络进行传递。网络中计算机之间的传输记录保存在网络登记文件中。

文件:

/ bin / netutil

/ etc / default / micnet

## parallel (HW)

并行接口设备

说明:

并行设备有以下几类:

/ dev / lp0           主并行适配器。

/ dev / lp1           在单色视频卡上的适配器。

/ dev / lp2           备用并行适配器 (在某些机器上)。

在一个系统上同时有这三种并行设备是不可能的。有些 AT 机允许使用两个并行设备, / dev / lp2 及 / dev / lp0 或 / dev / lp1。

如果并行设备不能正常中断,则并行驱动程序进入“轮询”方式。一旦收到从设备发来

的中断，驱动程序就返回其初始方式。

当并行设备关闭时，并行设备驱动程序会延迟一定的时间。这一延迟无疑会影响打印机性能，但考虑到打印机缓冲区大小及打印速度的差异，这一延迟还是必要的。比如，下面的命令使设备关闭后的延迟时间为1秒（以1/10秒为单位给出）：

```
stty time 10 < / dev / lp0
```

---

注：附加卡上的并行适配器会起作用，但开关必须设置正确。有些兼容机的 lp0 端口与 lp1 端口颠倒了。

---

文件：

/ dev / lp0

/ dev / lp1

/ dev / lp2

## restore

UNIX 增量文件系统备份的恢复

语法：

```
restore [-c][-i][-o][-t][-d device][pattern[pattern]...]
```

说明：

这个实用程序作为 cpio(C)的前端，并因此用于读 cpio 格式的磁带和软盘。这个实用程序只能用来恢复由 AT&T backup (ADM) 所作的备份，而不能恢复 xbackup (ADM) 的。

- c 完全恢复。恢复磁带上的所有文件。
- i 取得介质的索引文件。这个选项只对于使用 backup 建立的备份有作用。输出是介质上的所有文件的目录。但没有真正恢复任何文件。
- o 重写现存文件。如果所恢复的文件已经有了，那么除非有这个选项，否则是不会再恢复的。
- t 表示所用的是磁带设备。当从磁带上恢复备份时，一定要与 -d 选项一起使用。
- d device 是所用的原始设备。其缺省值为 / dev / rdsk / f0q15d(1.2M的软盘)。

在恢复备份时，可以指定一个或多个 pattern。这些 pattern 与磁带上的文件进行匹配。当匹配成功，就恢复该文件。因为备份是使用完整的路径名进行的，所以文件被恢复到它原来的目录中。元字符可用于匹配多个文件。pattern 一定要放在引号中。如果没有指定 pattern，其缺省值表示恢复所有文件。如果某个 pattern 与磁带上的任何文件都不能匹配，就会显示一条信息。

当到达介质的最后时，用户就会被提示换下一卷介质。这时用户还可以输入“q”退出这个命令。（如果某文件正好跨越两个介质，这样做就会破坏这个文件。）总之，在中间退出不是好主意。

如果某个文件已经存在，你又在带 -o 选项的情况下对其进行恢复，这个文件名就

会在屏幕上显示出来，其后还跟着一个问号。这个文件将不会恢复。

为了正确地进行多卷恢复，一定要使用字符设备。

## **sar,sa1,sa2,sadc**

系统活动报告包

语法:

```
sar [-ubdycwaqvmnpr DSAC][[-o file] t [n]
sar [-ubdycwaqvmnpr DSAC][[-s time][[-e time][[-i sec]
[-f file]
/ usr / lib / sa / sadc [t n][ofile]
/ usr / lib / sa / sa1 [t n]
/ usr / lib / sa / sa2 [-ubdycwaqvmnpr DSAC][[-s time]
[-e time][[-i sec]
```

说明:

sar 在第一种用法中对 n 个 t 秒钟的操作系统活动计数器进行采样，其中 t 应该是 5 或更大。如果指定了 -o 选项，就会把样本以二进制形式存于文件 file 中。n 的缺省值是 1。在第二种用法中，没有指定采样区间，sar 从以前的记录文件 file 中取数据，这个文件可以是由 -f 选项指定的，或是标准系统活动每日数据文件 /usr/adm/sa/sadd，dd 指明当前日期。报告的开始和结束时间可以由 -s 和 -e time 变元以 hh[: mm[: ss]] 形式来限定。-i 选项选取在 sec 秒区间的记录。否则，在数据文件中所有的区间都被报告出来。

在任何一种形式中，数据的子集都根据选项的指示被显示出来:

- u 报告CPU的使用情况
- b 报告缓冲区活动
- d 报告各块设备的活动
- n 名缓存的统计报告
- y TTY设备活动报告
- c 系统调用的报告
- w 报告系统替换和切换活动
- a 报告文件访问的系统例程的使用
- q 报告所占用队列的平均长度和占用时间的百分比
- v 报告进程、节点i、文件表的状态
- m 报告信息和信号量的活动
- p 报告分页活动
- r 报告没用的内存页和磁盘块
- D 报告远程文件共享活动
- S 报告服务和请求队列状态
- A 报告所有的数据。与-udqbwcaqvmpr SDC等价。
- C 报告远程文件共享高速缓存的开销。

## screen

tty[01-n], 彩色、单色、EGA、VGA 显示适配器和视频监视器

### 说明:

tty[01-n]设备文件在系统和视频显示器或键盘之间提供字符 I/O。每个文件都对应一个单独的电转设备。虽然最多有 12 个屏幕, 但能够获得的确切数目 (n) 却依赖于计算机内存的容量。屏幕都被设置为 25 行、80 列的 ASCII 终端, 除非有特殊设置。

从核心来的系统错误信息都被写到 /dev/console, 它一般都是当前的多屏幕。如果 /dev/console 是系统错误信息的缺省输出设备, 而所使用的显示器却被切换为图形模式, 那么就不会显示控制台 (console) 信息。当视频设备返回文本模式, 就能显示各种注释信息, 并且核心错误文本也能从 /usr/adm/messages 取出显示。

虽然所有的 tty[01-n]设备当前都可能打开, 但是在任何给定时间中只能有一个设备是活动的。这个活动的设备显示其自己的屏幕, 并只对单一的键盘提供服务。在没有相应的适配器存在或没有与其相关的多屏幕时, 这个设备就访问某个彩色、单色或 ega 文件, 就会产生错误。

要想进入下一个多屏幕, 可输入 Ctrl-PrtSc。任何一个屏幕都可通过输入 alt-Fn 来选取, 其中 Fn 是一个功能键。例如, F1 对应 tty01 设备。

## serial(HW)

串行端口的接口

### 说明:

ttyl[a-h], ttyl[A-H], tty2[a-h]和 tty2[A-H]文件提供对计算机的标准串行端口和任选串行端口的访问。每一个文件对应一个串行端口 (包括带 modem 控制的和不带 modem 控制的)。文件按如下规则进行命名:

- 文件名中的第一数字对应于 COM 扩展槽。
- 小写字母表示不带 modem 控制。
- 大写字母表示带有 modem 控制。

tty1a 和 tty1A 同指 COM1, 而 tty2a 与 tty2A 都表示 COM2。

比如, 若有四个扩展板安装于 COM1, 一个单端口板安装 COM2, 则应访问:

ttty1a	tty1A
ttty1b	tty1B
ttty1c	tty1C
ttty1d	tty1D
ttty2a	tty2A

每个串行端口都有带 modem 控制和不带 modem 控制这两种可能。下面给出带 modem 控制和不带 modem 控制的串行端口。表的第一部分给出安装在 COM1 的板, 第二部分给出安装于 COM2 的板。“Minor”表示该端口的次设备号:

Serial Lines							
Board Type	Non-Modem Control		Modem Control				
	Minor	Name	Minor	Name			
1 Port 4 Port 8 Port	0	tty1a	128	tty1A			
	1	tty1b	129	tty1B			
	2	tty1c	130	tty1C			
	3	tty1d	131	tty1D			
	4	tty1e	132	tty1E			
	5	tty1f	133	tty1F			
	6	tty1g	134	tty1G			
	7	tty1h	135	tty1H			
	1 Port 4 Port 8 Port	8	tty2a	136	tty2A		
		9	tty2b	137	tty2B		
		10	tty2c	138	tty2C		
		11	tty2d	139	tty2D		
12		tty2e	140	tty2E			
13		tty2f	141	tty2F			
14		tty2g	142	tty2G			
15		tty2h	143	tty2H			

中断向量:

安装在 COM1 的所有板——4

安装在 COM2 的所有板——3

访问

只有当相应的串行接口卡已安装好,而且 I/O 地址的跳线开关也设置正确时,才能对这些文件进行访问。同样,对于多端口卡,必须用 mkdev (ADM) 程序建立多于文件缺省数目的文件,才能进行访问。如果在有关硬件文件中没有特别说明其它的 COM 端口,则只能用 COM1 和 COM2。

串行口也必须在系统配置中定义。参考硬件手册,看看系统是如何配置的,是通过 CMOS 数据库还是由系统板上的开关设置的。若是前者,则端口在 CMOS 中定义,若是后者,则应将系统板上的有关开关设置正确。

如要访问一个未安装或未定义的串行端口,则会出现错误。

串行端口可以用来实现不同的串行通信,比如将注册终端连到计算机上,或连接打印机,或与其它计算机组成串行网络等。注意,串行端口可在大多数标准波特率下运行,并且在大多数计算机的串行端口上都有 DTE (数据终端设备) 配置。下面的表定义了 25 腿和 9 腿连接中每个腿的使用情况:

25 腿	9 腿	说 明
2	2	传输数据
3	3	接收数据
4	7	请求发送
5	8	清除发送
7	5	信号地
8	1	载波检测
20	4	数据终端准备好。

对于直接连接而言，只有 2、3、7（对 9 腿连接则是 2、3、5）腿是必须用的。

文件：

```
/dev/tty1[a-h]
/dev/tty1[A-H]
/dev/tty2[a-h]
/dev/tty2[A-H]
```

---

注：如果你通过 modem 控制的串行线路注册，则挂断线路后会“杀死”你的后台进程。

不能同时使用带 modem 控制和不带 modem 控制的同一端口，比如，不能同时使用 tty1a 和 tty1A。

可以用一条调制解调器电缆将你的 modem 连接到一台计算机上。

---

## stty

为终端设置选项

语法：

```
stty[-a][-g][options]
```

说明：

stty 为作为当前标准输入的设备设置某些终端 I/O 选项；如果不带任何变元，它就会报告某些选项设置。当带有 -a 选项时，stty 就会报告所有的选项设置。-g 选项使 stty 输出终端的当前 stty 设置，输出形式为 12 个由冒号分隔的 16 进制数。这个输出还可以作为 stty 命令行的变元，以供以后在该终端上恢复这个设置使用。这个输出比 stty-a 的压缩了许多。例如，下面的 shell 描述程序使用 stty-g 保存当前的 stty 设置，然后关闭读输入行时的反显。所保存的 stty 值最后再恢复到该终端上：

```
:
echo "Enter your secret code:\c"
old = 'stty-g'
stty -echo intr ^ a
read code
stty $ old
```

各种模式将在下面分几组进行讨论。

一般控制模式:

parenb(-parenb) 使能(禁止)奇偶校验  
parodd(-parodd) 选择奇(偶)校验  
cs5 cs6 cs7 cs8 选择字符的大小  
0 立即挂起电话线  
50 75 110 134 150 200 300 600 1200 1800 2400 4800 9600 19200  
把终端波特率设置为所给的数

ispeed 50 75 110 134 150 1200 1800 3400 4800 9600 19200  
单独设置终端输入波特率

ospeed 50 75 110 134 150 1200 1800 3400 4800 9600 19200  
单独设置终端输出波特率

hupcl(-hupcl) 在最后关闭时挂起(不挂起)电话连接  
hup(-hup) 同hupcl(-hupcl)  
cstopb(-cstopb) 每个字符使用两个(一个)停止位  
cread(-cread) 接收使能(禁止)  
clocal(-clocal) 假设一个终端线不带(带)调制解调控制  
ctsflow(-ctsflow) 为一条调制解调或非调制调线使能CTS协议  
rtsflow(-rtsflow) 为一条调制解调或非调制解调线使能RTS信号

输入模式:

ignbrk(-ignbrk) 忽略(不忽略)输入中的中断  
brkint(-brkint) 在中断时给(不给)INTERRUPT信号  
ignpar(-ignpar) 忽略(不忽略)奇偶校验错  
parmrk(-parmrk) 标记(不标记)奇偶校验错  
inpck(-inpck) 使能(禁止)输入奇偶校验检查  
istrip(-istrip) 截取(不截取)输入字符至7位  
inlcr(-inlcr) 在输入上把NL映射(不映射)为CR  
igncr(-igncr) 忽略(不忽略)输入上的CR  
icrnl(-icrnl) 在输入上把CR映(不映射)为NL  
iuclc(-iuclc) 在输入上把大写字母映射(不映射)为小写字母  
ixon(-ixon) 使能(禁止)START / STOP输出控制。通过发送一个  
ASCII DC3 停止输出,发送一个 ASCII DC1 启动输出。  
ixany(-ixany) 允许任意字符(只允许DC1)启动输出  
ixoff(-ixoff) 当输入队列将要空 / 满时, 请求系统发送 (不发送)  
START / STOP 字符

输出模式

opost(-opost) 后处理输出 (不要后处理输出; 忽略所有其它输出模式)  
olcuc(-olcuc) 在输出中把小写字母映射 (不映射) 为大写字母  
onlcr(-onlcr) 在输出中把NL映射 (不映射) 为CR-NL

ocrnl(-ocrnl)	在输出中把CR映射（不映射）为NL
onocr(-onocr)	在第0列不输出（输出）CR
onlret(-onlret)	在终端上NL执行（不执行）CR功能
ofill(-ofill)	使用填充字符（使用计时）作为延迟
ofdel(-ofdel)	填充字符为DELETE（NUL）
cr0 cr1 cr2 cr3	为RETURN选取延迟形式
nl0 nl1	为LINEFEED选取延迟形式
tab0 tab1 tab2 tab3	为水平TAB选取延迟形式
bs0 bs1	为BACKSPACE选取延迟形式
ff0 ff1	为FORMFEED选取延迟形式
vt0 vt1	为纵向TAB选取延迟形式

本地模式:

isig(-isig)	使能（禁止）与特殊控制字符INTERRUPT和QUIT的对照检查
icanon(-icanon)	使能（禁止）规范输入（ERASE和KILL处理）
xcase(-xcase)	规范处理（不处理）大/小写
echo(-echo)	反显（不反显）所键入的字符
echoe(-echoe)	把ERASE反显（不反显）为一个空格。
echok(-echok)	在KILL字符之后反显（不反显）NL
lfkc(-lfkc)	与echok（-echok）相同；已不用。
echonl(-echonl)	反显（不反显）NL
noflsh(-noflsh)	在INTERRUPT或QUIT之后禁止（使能）清除
iexten(-iexten)	使能扩展的补充（补充定义）功能
tostop(-tostop)	禁止/使能后台进程组对控制终端的写操作，这只有当系统支持作业控制时有效。

控制定义:

control -character C	把 control -character 设置为 C，其中 control -character 是 erase、kill、interrupt、quit、eof、eol、switch 或 susp。如果 C 前面加一个 (^)，那么所用的值就是相应的 CTRL 字符（例如，“^ D”就是 CTRL-D）；“^ ?”被解释为 DELETE，而“^ -”被解释为无定义。
----------------------	--

组合模式:

evenp或parity	使能parenb和cs7
oddp	使能parenb、cs7和parodd
-parity,-evenp或-oddp	禁止parenb,并设置cs8
raw(-row或cooked)	使能（禁止）原始输入和输出（无ERASE、KILL、INTERRUPT、QUIT、EOT）
nl(-nl)	不设置（设置）icrnl,onlcr。另外，-nl不设置inlcr,igncr,ocrnl



	和 onlret
lcase(-lcase)	设置 (不设置) xcase,iuclc和olcuc。
LCASE(-LCASE)	与lcase(lcase)相同
tabs(-tabs或tab3)	在显示时保持制表符 (把制表符扩展为空格)
ek	把ERASE和KILL字符恢复回正常的CTRL-H和CTRL-U

## su

使用户成为超级用户或另外一个用户

语法:

```
su[-][name[arg...]]
```

说明:

su 使得某被授权用户不必先注销就变成另一个用户。缺省用户是 root (即超级用户)。

su 在这种 UNIX 中不能用来简单地接受其它用户的注册。su 只能在如下四种情况下使用:

- 超级用户可以“su”到任何帐号。
- 有 su 权限的管理员用户可以“su”到超级用户帐号。
- 用户可以“su”到他们自己的帐号 (虽然这样做很傻, 但是可能的)。
- 系统的守护程序可以“su”到一个帐号。

要使用 su, 必须提供正确的口令 (除非已经是超级用户了)。如果口令正确, su 将执行一个新的 shell, 这个 shell 带有给指定用户设置的有效用户 ID。(LUID 并不改变。) 这个新的 shell 是在指定用户口令文件中 shell 域命名的程序, (如果没有特殊设定, 就是 /bin/sh)。为了恢复原来的用户 ID, 特权, 就在新 shell 中按 EOF (Ctrl-D)。

在命令行中给的其它变元都被送到被作为 shell 调用的程序中。当使用像 sh (C) 这样的程序时, 格式为 -C string 这样一个 arg 会通过这个 shell 执行这个 string, 而 -r 这样的 arg 会给用户一个有限的 shell。

## fsname

显示或修改文件系统的名字

语法:

```
fsname[-p][-s name] / dev / device
```

说明:

/etc/fsname 用于显示或修改文件系统的名字。

其选用项如下:

- P 选取“pack”名字域而不是文件系统名域。
- S name  
修改超级块中指定的域。  
该命令的缺省动作是显示文件系统名。

## tar

文件归档

语法:

```
tar [key][files]
```

说明:

tar 向档案介质存入或从中取出文件，这种档案介质一般为像软盘或磁带这样的存储设备，或是一个文件。其动作由变元 key 控制。key 是一个字符串，其中最多包含一个功能字符，同时还可能包含一个或多个功能修正字符。有效的功能字符有 c, t, x 和 e。此命令的另外的变元是 files，这些文件（或目录名）都是要备份和恢复的。在各种情况下，目录名的出现就代表着这个目录中的文件和子目录。r 和 u 选项不能用于磁带设备。

key 的功能字符由下列字母之一指定:

- r: 所指定的files写到现存档案介质的末尾。
- x: 从档案介质读取所指定的files。如果没有给定files，就要读取档案介质中的所有内容。
- t: 列出档案介质中的被指定的文件名。如果没有给files变元，就会列出档案介质中的所有文件名。
- u: 如果所指定的files不在档案介质上，或者自从上一次写档案介质之后该文件被修改过，就把它写到档案介质上。
- c: 建立一个新档案；从档案的开始处写起，而不是在档案的末尾写起。

下面是修正字符:

0,...,9999

这个修正字符用于选择档案介质安装的驱动器。缺省设置可在 /etc/default/tar 中找到。

- v: 一般情况下，tar 执行时不带任何显示。v 选项使得在处理每个文件的同时显示该文件名，在该文件名之前还带有一个功能字符。当与 t 选项合用时，v 还会给出关于档案介质中各文件或目录的更多的信息，而不仅仅是名字。
- w: 使 tar 显示要执行的动作，后面跟有文件名，然后再等待用户的确认。如果用户给了一个以 y 打头的单词，这个动作就被执行，所有其它的输入都表示“no”
- f: 使 tar 使用下一个变元作为档案介质的名字，而不用列于 /etc/default/tar 中的缺省设备。如果文件名是一个破折号 (—)，tar 就向标准输出设备中写或从标准输入设备中读。这样 tar 就可以被用作一个管道的头或尾。tar 也可以被用在如下命令中来移动分层目录结构:  

```
cd fromdir;tar cf-.1(cd todir;tar xf-)
```
- k: 让 tar 把下一个变元作为档案介质的容量，用 KB 为单位。最小值可以是 250。非常大的文件这样就可被分开放到几个介质卷中。
- e: 防止把一个文件分开放到几个介质卷中。如果对于这个文件所给的介质卷容量不够，tar 就会提示换一个新介质卷。这个 e 字符只有当与 k 一起用时才有效。

- n: 指明档案设备不是磁带。k选项是这样暗示的。
- p: 指明获取文件后，依然使其带有原来的读写权限。

## tset

设置终端模式

### 语法:

tset [options][type]

### 说明:

tset 可以使用户设置终端的 ERASE 和 KILL 字符，并通过给 TERM 这个环境变量设定不同的值来定义终端的类型和能力。tset 使用 tput (C) 对终端进行初始化或复位。如果使 type 带一个 -s 选项，tset 就会给出所指定类型终端的信息。这些类型可以是 terminfo 数据库中所给的任何类型。如果 type 带 -s 选项，tset 就会给出由环境变量 TERM 的值所定义的终端类型的信息，除非给了 -h 或 -m 选项。如果定义了 TERM 变量，tset 就会使用相应的 terminfo 数据库的记录。如果使用了这些选项，tset 就会在 /etc/ttytype 文件中查找与当前串行口对应的终端类型；然后它就会根据这个类型为终端建立信息。如果在 /etc/ttytype 文件中没找到这个串行口，终端类型就被设置为 unknown。

tset 在标准输出设备上显示所建立的信息。信息的形式能被用来设置当前的环境变量。确切的格式依赖于调用 tset 的注册 shell。

有如下选项:

- e [c] 在所有终端上把 ERASE 设置为 c。缺省设置是 BACKSPACE 或 CTRL-H。
- E [C] 与 -e 命令大体相同，区别在于它只能用于能够用 BACKSPACE 的终端。
- k [c] 把 KILL 字符设置为 C，缺省值为 CTRL-U。
- 在标准输出设备上显示终端类型。
- s 输出“setenv”命令[对于 csh (C)]，或“export”命令[对于 sh(1)]。命令的类型由用户的注册 shell 决定。
- h 使 tset 在 /etc/ttytype 中找信息，而不管环境变量 TERM。
- S 只显示被置于环境变量中的字符串，而不显示 -S 的 shell 命令。
- r 在诊断输出上显示终端类型。
- Q 隐藏“Erase set to”和“Kill set to”信息的显示。
- I 隐藏终端初始化字符串的显示。

-m[ident][test baudrate]:type

使用户指定一个给定的串行口如何映射到一个实际的终端类型。这个选项用于 /etc/ttytype 中类型没确定的任意串行口。type 指定所用的终端类型，ident 表示要匹配的未确定类型名称。如果没给 ident，就会匹配所有的未确定类型。test baudrate 定义一个在分配类型之前在该串行端口上所进行的测试。baudrate 一定要与 stty (C) 中定义的不同。test 可以是 >, =, <, @ 和 ! 的任意组合。如果 type 以问号开头，就是询问用户是否确实想要那种类型。应答为空表示就是要用那种类型，否则就必须输入另一种类型

来代替原来的类型。如果所给的-m 选项多于一个，那么第一个正确的映射是有效的。

tset 的最大用处在于放在 .login 或 .profile 文件中，在用户注册时自动执行，用其-m 选项指定终端类型。

### uucico(ADM)

UUCP 系统的文件传输程序

语法:

```
/usr/lib/uucp/uucico [-r role__number][ -X debug__level]
[-i interface][ -d spool__directory][ -s][ -S]system__name
```

说明:

uucico 是用于 uucp 工作文件传送的文件传输程序。-r 所跟的 role\_\_number 为 1, 则表示主方式, 为 0, 则表示从方式, 缺省值为 0。当 uucico 是由一个程序所启动时, 或者 cron.uux 和 uucp 都将作业放入队列等待 uucico 传递时, 应将-r 选择项值置 1 表示主方式 (master mode)。uucico 通常是由调度程序 uusched 启动的, 但也可手工进行启动, 这常常用于调试目的。比如, shell 程序 uutry 启动 uucico 时就设置了调试状态。-x 选项的意义前已讲过, 此处不再重复。

-i 任选项定义 uucico 所用的接口, 该接口只对从方式起作用。缺省接口是 UNIX。

-d 任选项用于指定 spool 目录。缺省值为 /usr/spool/uucp。

如果指定了-s 任选项, 则即使什么工作也不做, 对指定系统的寻呼 (call) 也要执行, 在寻呼时间必须在 System 文件允许的范围内。这一选择项常用于系统查询。

-S 选项用于指示系统名, 替代 Systems 文件中给出的系统。

文件:

```
/usr/lib/uucp/systems
/usr/lib/uucp/Permissions
/usr/lib/uucp/Devices
/usr/lib/uucp/Maxuuscheds
/usr/lib/uucp/Maxuuxqts
/usr/spool/uucp/*
/usr/spool/uucppublic/*
```

### uucheck(ADM)

检查 uucp 目录和 Permissions 文件

语法:

```
/usr/lib/uucp/uucheck[-V][ -X debug__level]
```

说明:

uucheck 检查 uucp 系统所需的文件和目录是否存在, 也对 Permissions 文件 (/usr/lib/uucp/Permissions) 中明显的错误进行检查。当带着-V 任选项执行时, 给出 uucp 程序如何解释 Permissions 文件的详细说明。-X 选择项前已说明, 此处不再解释。

uuclean 程序只有超级用户和 uucp 有权使用。

文件:

/usr/lib/uucp/Systems  
/usr/lib/uucp/Permissions  
/usr/lib/uucp/Devices  
/usr/lib/uucp/Maxuuscheds  
/usr/lib/uucp/Maxuuxqts  
/usr/spool/uucp/\*  
/usr/spool/Uucppublic/\*

## uuclean (ADM)

对 uucp 的 spool 目录进行清理

语法:

```
/usr/lib/uucp/uuclean [-Ctime][-Dtime][-Wtime][-Xtime]  
[-mstring][-otime][-ssystem][-Xdebug_level]
```

说明:

uuclean 在 spool 目录中扫描“老”文件，并采取适当措施将其删除:

- 通知发送/接收请求的发出者: 系统不能访问。
- 将不能传递的邮件返回发送者。
- 删除所有其它文件。

另外，还会给请求已经等待了一段时间（缺省值是 1 天）而还未得到执行的用户送警告信息。值得一提的是，如果所有任选项中的 time 值没有设定，uuclean 就使用各自的缺省值。

uuclean 的任选项有:

- Ctime: 所有等待时间长于或等于 time 天的 C 文件都将被删除，同时给各自请求者发相关信息。time 的缺省值为 7。
- Dtime: 所有等待时间长于或等于 time 天的 D 文件都将被删除。time 缺省值为 7。
- Wtime: 在 C 文件等待时间等于 time 天时，就给请求者发送消息，告知未能顺利与远程系统联系上。time 缺省值为 1。（单位为天）
- Xtime: 所有等待时间大于或等于 time 天的 X 文件都将被删除。time 缺省值为 2。
- mstring: 本行被加在由 -W 产生的警告消息中。
- otime: 所有等待时间大于 time 天的其它类型文件将被删除。
- ssystem: 只在 system 的 spool 目录中执行。

本程序通常是由 shell 程序 uudemond.clean 启动的。uuclean (ADM) 只能由超级用户和 uucp 执行。

文件:

/usr/lib/uucp 带有 uuclean 内部使用命令的目录。  
/usr/spool/uucp spool 目录。

## **uuinstall(ADM)**

管理 UUCP 控制文件

### **语法:**

`/ etc / uuinstall[-r]`

### **说明:**

uuinstall 程序用于管理 uucp 通信系统所用的控制文件，它允许用户修改这些文件的内容而不必使用文本编辑程序。用户不必知道每个控制文件的具体格式，但必须熟悉这些文件中不同的域的功能。

uuinstall 程序只有超级用户有权执行。在使用 -r 任选项时，不允许修改任何一个文件，而不管用户是否对文件作了改动。

如果 uuinstall 发现某一需要使用的 uucp 控制文件从系统中丢失，它就会重新建立一个权限和属主都与以前一样的文件。

文件:

`/ etc / Systemid`

`/ usr / lib / uucp / Systems`

`/ usr / lib / uucp / Permissions`

`/ usr / lib / uucp / Devices`

## **uusched (ADM)**

UUCP 文件传输程序的调度程序

### **语法:**

`/ usr / lib / uucp / uusched[-x debug__level][-u debug__level]`

### **说明:**

uusched 是 uucp 文件传输调度程序。它通常是由守护程序 uudemon.hour 启动的。

uusched 的两个任选项都是供调试用的。-x debug\_\_level 输出 uusched 的调试信息，-U debug\_\_level 将作为 -x debug\_\_level 传递给 uucico。关于 debug\_\_level，前面已多次提及，这里不再重复。

文件:

`/ usr / lib / uucp / Systems`

`/ usr / lib / uucp / Permissions`

`/ usr / lib / uucp / Devices`

`/ usr / lib / uucp / Maxuuscheds`

`/ usr / spool / uucp / *`

`/ usr / spool / Uucppublic / *`

## **uutry (ADM)**

尝试与远程系统联系的实用程序

### **语法:**

`/ usr / lib / uucp / uutry [-x debug__level][-r]System__name`

### 说明:

uutry 是启动 uucico 对远程系统进行访问的 shell 程序。执行时自动设置了调试 (debug) 状态, 级别缺省值是 5, -x 任选项可对级别进行修改。如果 uutry 成功地实现了与远程系统的连接, 则将调试结果存放于文件 /tmp/system 中, 其中 system 是远程系统名。另外, uutry 还可以用 tail -f 将调试结果的最后 10 行送至标准输出。

要退出 uutry 建立 shell 程序的执行, 打入 DELETE 和 BREAK 均可。这使得 uucico 继续运行的同时返回对终端的控制, 同时将结果送至文件 /tmp/system\_\_name, 其中 system\_\_name 是远程系统名。

-r 任选项可覆盖在 /usr/spool/uucp/.status 中指定的重试 (retry) 时间。

文件:

```
/usr/lib/uucp/Systems
/usr/lib/uucp/Permissions
/usr/lib/uucp/Devices
/usr/lib/uucp/Maxuuscheds
/usr/lib/uucp/Maxuuxqts
/usr/spool/uucp/*
/usr/spool/uucppublic/*
/tmp/system__name
```

### xbackup

执行 XENIX 增量文件系统备份

语法:

```
xbackup [key [arguments]filesystem]
```

说明:

xbackup 用于拷贝 filesystem 中在某一个日期之后修改过的所有文件。xbackup 用于 XENIX 文件系统; 对 UNIX 文件系统要使用 backup (ADM)。key 指定日期和有关 xbackup 的其它选项, 其中 key 由 0123456789kfusd 中的字符组成。这些字符的意义说明如下:

- f 在后面的 arguments 域中放置一个设备文件名来代替缺省文件。
- u 如果 xbackup 成功完成, 就向文件 /etc/ddate 中写入 xbackup 开始的日期。这个文件分别记录了每个文件系统和每个备份级的日期。
- 0-9 这个数是“xbackup level” (x 备份级)。在低一级上备份同一个文件系统中上一次备份之后修改过的所有文件。(最后一次备份的日期一定要记在 /etc/ddate 文件中。) 如果没有由备份级确定上一次的备份日期, 那么这个日期就确定为系统开始时间; 这样选项 0 就会进行整个文件系统的备份。
- s 用于向磁带中做 xbackup, 指定以英尺为单位的磁带长度。英尺数从后面的 arguments 获得。当到了指定的长度, xbackup 就等待换下一盘磁带。缺省值为 2300 英尺。
- d 用于向磁带中做 xbackup, 指定磁带的密度, 以 BPI 表示, 这个密度可从后面

的 arguments 获得。可用于计算每一次写操作所用的磁带容量。缺省值为 1600。

- k 这个选项用于向块结构设备中做备份，例如软盘。所写的介质卷的大小（以 KB 为单位）从 arguments 得到。如果指定了 k 变元，就会忽略 s 和 d 变元。缺省时，就使用 s 和 d。如果没给出任何变元，key 就被假定为 9u，即把缺省的文件系统备份到缺省设备上。

第一次 xbackup 应该是完全的 0 级 xbackup:

```
xbackup 0u
```

下面，定期的 9 级备份应该以指数过程做到磁带或磁盘上:

```
xbackup 9u
```

这个过程表示如下:

```
12131214...
```

其中 xbackup 1 隔一次使用一次，xbackup 2 每四次做一次，xbackup 3 每 8 次做一次，以此类推。当磁带已经满了或所用的软盘太多了，就应该做一次 1 级 xbackup:

```
xbackup 1u
```

然后，再继续进行上述指数备份过程，就好像没中断过一样。这些 9 级备份都以这个 1 级备份为基础，而这个 1 级备份又是建立在 0 级完全备份基础之上的。这个分级备份过程可以按需要进行下去。

所备份的文件系统和 xbackup 设备的缺省设置分别依赖于变量 DISK 和 TAPE 的设置，这两个变量都在 /etc/default/backup 文件中。

## xrestore,xrestor

调用 XENIX 增量文件系统的恢复程序

语法:

```
xrestore key[arguments]
```

```
xrestor key[arguments]
```

说明:

xrestore 被用来读由 xbackup(ADM)命令备份的档案介质。

key 指定要做什么。key 是字符 cC, rR, tT 或 xX 中的一个与 k 和 f 或 F 的组合。

xrestor 是同一个命令的另一种拼法。

c,C 核查磁带。在这个磁带被确认没有任何 I/O 错误和其它问题之后就可以使用了。C 与 c 相同，只是 C 提供的是更高一级的检查。

f 把一个 argument 作为档案介质名（备份设备 / dev / \*）使用，来代替缺省设备。

F F 指定在要从磁带上读的的第一个文件号。所有在其之前的文件都被掠过。

k 在这个选项之后跟着每个备份卷的容量。这可以用来读多卷的信息转储介质，例如软盘。

r,R 读取档案介质中的内容并装入由 argument 指定的文件系统中。如果关键字为 R，xrestore 就会问从多介质卷档案中的哪一卷开始。这可以允许 xrestore 被



中断，然后再重新启动（在重新启动之前必须做 fsck）。

t 显示档案所写的日期和文件系统的备份日期。

T 显示磁带中内容的完整目录。与t类似。

x 取由argument指定名字的每个文件。请注意，在档案介质中所有文件的安装前缀都被去掉了；例如，如果 /usr 是一个被安装的文件系统，那么 /usr / bin / lpr 在档案上的名字就是 / bin / lpr。

X 与x相同，只是文件被放在备份之前的位置。当使用这个选项时，在命令行中要去掉文件名中原来的斜线（/）。

r 选项只用于把一个完全的备份恢复到一个空的文件系统中，或把一个增量备份恢复到这样建立的文件系统中。它不应该用于把备份恢复到根文件系统。因此，

```
/etc / mkfs / dev / hd1 10000
```

```
xrestore r / dev / hd1
```

是恢复完整备份的一个典型顺序。在这个文件系统中，还可以用另一个 xrestore 恢复一个增量备份。

在 xbackup 命令之后使用 mkfs，然后再使用 xrestore 可改变文件系统的大小。

---

注：

用 xrestore 不可能成功地恢复整个活动的根文件系统。

xrestore 可能不会从磁带设备 / dev / nrct0 和 / dev / nrct2 中恢复一个以上的文件系统。

---

[ G e n e r a l   I n f o r m a t i o n  
]

书名 = S C O   U N I X 系统管理员实用指南  
1 9 9 4

作者 =

页数 = 4 9 8

S S 号 =

D X 号 =

出版日期 =

出版社 =

书名  
前言  
目录  
目录

概述

第一部分安装指南

第一章安装之前

- 1.1 关于安装介质
- 1.2 安装要求
- 1.3 必须知道的内容
- 1.4 建立和格式化DOS

分区

- 1.5 响应提示信息
- 1.6 计划磁盘布局

第二章安装过程

- 2.1 安装指令

附录A 安装中可能出现的问题

附录B 安装和移去附加软件

- B.1 安装附加产品
- B.2 安装UNIX系统

软件包

- B.3 从系统中移去软件

附录C 建立紧急启动软盘组

- C.1 建立软盘

附录D 硬盘配置

附录E 设置北美之外的时区

附录F 建立附加文件系统

附录G 配置非缺省参数的磁带机

- G.1 开始磁带配置
- G.2 标准卡式磁带
- G.3 SCSI卡式磁带

驱动器

G . 4 回到 U N I X 系统安装

附录 H 安装词汇解释

第二部分 日常管理指南

第一章启动和停止系统

1 . 1 启动系统

1 . 2 作为超级用户注册

1 . 3 停止系统

1 . 4 关于启动显示信息

1 . 5 改变引导进程

第二章系统管理工具—— s y s a d

m s h

2 . 1 s y s a d m s

h 的屏幕组织 & ( 5

2 . 2 s y s a d m s h

的操作

2 . 3 s y s a d m s h

的其它功能

第三章用户帐号的管理

3 . 1 帐号管理

3 . 2 帐号的缺省配置

3 . 3 终端注册管理

3 . 4 活动报告的生成

第四章文件系统和管理的

4 . 1 文件系统的安装和拆卸

4 . 2 文件系统的类型

4 . 3 文件系统的可配置特

征

4 . 4 文件系统中空闲空间的

维护

4 . 5 维持高效的文件系统组

织	
系统	4 . 6 增加磁盘空间的文件
	4 . 7 文件系统的完整性
OS	第五章在同一硬盘上使用UN I X和D
硬盘分区	5 . 1 用 f d i s k 进行
UN I X	5 . 2 在D O S系统上安装
使用UN I X和D O S	5 . 3 在两个硬盘的环境下
系统	5 . 4 从硬盘上删除一个操作
序	5 . 5 访问D O S的实用程
D O S文件系统	5 . 6 在UN I X系统上安装
	第六章系统安全性维护
	6 . 1 什么是可靠系统
	6 . 2 运行一个可靠系统
	6 . 3 保护系统上的数据
	6 . 4 检查系统所受的损害
	6 . 5 处理文件系统错误
息	6 . 6 有关安全性的错误信
护程序	6 . 7 可靠系统上所运行的守
	第七章审计子系统
	7 . 1 数据采集
	7 . 2 文件管理

	7 . 3	报告生成
	7 . 4	授予用户审计权力
	7 . 5	审计术语
	第八章	UNIX目录和特殊设备文件
	8 . 1	UNIX目录
	8 . 2	记录文件
	8 . 3	特殊设备文件
	第九章	文件系统和备份
s h	9 . 1	使用 s y s a d m s
进行备份的策略	9 . 2	例行备份的准备
	9 . 3	进行例行备份
	9 . 4	进行非例行备份
	9 . 5	备份的核查
	9 . 6	获取备份表
	9 . 7	从备份中恢复特定的
文件或目录	9 . 8	恢复完整的文件系统
	9 . 9	备份级的概念
	第十章	系统性能的调整
i g u r e	1 0 . 1	利用 c o n f
重新分配核心资源	1 0 . 2	修改系统配置
	1 0 . 3	性能的重新配置
	1 0 . 4	规定有效的系统使
用模式	1 0 . 5	对低效系统进行诊断
	1 0 . 6	调整磁盘效率
	1 0 . 7	可调系统参数的说
明		
	附录 A	系统启动的预设置

附录 B 作业调度命令的使用授权  
第三部分 外设使用指南  
第一章 增加多功能卡、内存卡及其它总

线卡

- 1 . 1 安装总线卡
- 1 . 2 增加存储器
- 1 . 3 增加及配置串行

端口

第二章 打印机

- 2 . 1 安装打印机
- 2 . 2 打印机配置选项
- 2 . 3 用户命令综述
- 2 . 4 管理命令综述
- 2 . 5 增加本地打印机
- 2 . 6 启动和停止打印服务
- 2 . 7 删除打印请求
- 2 . 8 使打印机有效 / 无效
- 2 . 9 在一组打印机中增加

一台

- 2 . 1 0 设置系统缺省打印

执行者

- 2 . 1 1 安装表格或打

印字型轮

- 2 . 1 2 移去打印机或打印

机组

- 2 . 1 3 管理打印负载
- 2 . 1 4 管理队列优先级
- 2 . 1 5 表格与处理程序
- 2 . 1 6 使用请求登记中的信

息

- 2 . 1 7 调整打印服务

	2 . 1 8 建立R T S / C T
S 协议的串行打印机	2 . 1 9 使用不带假脱机程
序的打印机	2 . 2 0 建立初始设备文件
	第三章使用软盘和磁带驱动器
	3 . 1 使用磁带驱动器
	3 . 2 使用软磁盘
	第四章使用鼠标
	4 . 1 安装硬件
	4 . 2 安装鼠标
	4 . 3 使用鼠标
	第五章增加硬盘 & ( 2 8
	5 . 1 在开始之前
	5 . 2 安装硬盘
	5 . 3 增加新的文件系统
	5 . 4 操作系统核心的重连
接	5 . 5 将用户帐号移出主硬
盘	
	第六章终端的管理
	6 . 1 增加终端
	6 . 2 改变缺省的终端线特
性	6 . 3 设置终端类型
	6 . 4 自动设置终端类
型	6 . 5 去掉一个终端
	6 . 6 建立串行控制台
	6 . 7 利用m s c r e e n
使用串行多屏幕	



	第七章硬件的兼容性
	7 . 1 兼容的硬件
	7 . 2 如何使用本章
	7 . 3 硬件支持
机器	7 . 4 标准结构的 3 8 6 类
机器	7 . 5 微通道结构的 3 8 6
容性	7 . 6 一般兼容性标准
容性	7 . 7 标准结构机器的硬件兼
	7 . 8 微通道结构硬件的兼
	附录 A 使用系统控制台和彩色显示器
	第四部分网络通信指南
	第一章使用调制解调器
器	1 . 1 选择串行端口
器	1 . 2 配置调制解调器
	1 . 3 安装拨号输出调制解调
	1 . 4 建立拨号输入调制解调
共享的调制解调器	1 . 5 安装拨号输入 / 输出
	1 . 6 拨号输入你的计算机
	1 . 7 从你的计算机拨号输出
	第二章建立 M i c n e t 局部网络
	2 . 1 网络设计
	2 . 2 建立网络
	2 . 3 启动网络
	2 . 4 测试 M i c n e t 网

络		2 . 5 停止网络
		2 . 6 修改M i c n e t 网
络	第三章	用U U C P 建立远程网络
		3 . 1 简介
		3 . 2 将两个本地系统直接
线连		3 . 3 用调制解调器连接
远程U U C P 系统		3 . 4 在系统中配置U U C
P		3 . 5 特殊的U U C P
配置选项		3 . 6 管理U U C P 系统
		3 . 7 完整的U U C P 举例
		3 . 8 U U C P 的错误信
息	第四章	建立电子邮件
		4 . 1 引言
		4 . 2 M M D F 是如何
工作的		4 . 3 配置M M D F
		4 . 4 M M D F 系统的维
护		4 . 5 改变机器名
		4 . 6 I n t e r n e t
地址		
		第五部分实用参考
	第一章	解决系统问题
		1 . 1 解决引导问题

- 1 . 2 解决控制台问题
- 1 . 3 解决文件系统的问题
- 1 . 4 解决安装中的问题
- 1 . 5 解决注册问题
- 1 . 6 解决邮件故障问题
- 1 . 7 使用调制解调器可能
- 出现的问题
- 1 . 8 打印机系统故障排除
- 1 . 9 解决进程管理中的问
- 题
- 1 . 1 0 解决调度问题
- 1 . 1 1 解决磁带驱动器
- 的问题
- 1 . 1 2 解决终端问题
- 1 . 1 3 解决UUCP连接
- 的问题
- 1 . 1 4 排除其它系统故障

## 第二章常用命令解释