



Solaris 8:
The Complete Reference

Sun 公司核心技术书库

Solaris 8

技术大全



Paul A. Watters
(美) Sriranga Veeraraghavan 著

李明之 裘晓峰 等译



机械工业出版社
China Machine Press

OSBORNE

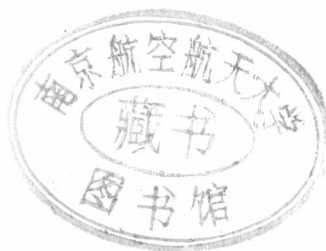
McGraw Hill Education

TP316
1838-3

Sun公司核心技术书库

Solaris 8 技术大全

(美) Paul A. Watters 著
Sriranga Veeraraghavan
李明之 裘晓峰 等译



200233411



机械工业出版社
China Machine Press

200233411

本书使用完全真实的例子和技巧帮助读者全面了解Solaris系统丰富的系统功能, 充分发掘Solaris系统的能力。本书不仅介绍了Solaris的安装、分析了Solaris操作系统的优势, 还介绍了进程管理、用户和软件管理、设备管理、高级网络服务的配置等内容。本书实用性极强, 有大量背景知识和实例, 为Solaris用户及管理员更好地使用Solaris操作环境提供了有益的参考。

Paul A. Watters and Sriranga Veeraraghavan: Solaris 8: The Complete Reference (ISBN: 0-07-212143-2).

Copyright © 2000 by the McGraw-Hill Companies, Inc.

Authorized translation from the English language edition published by McGraw-Hill, Inc.

All rights reserved. For sale in the People's Republic of China.

本书中文简体字版由机械工业出版社和美国麦格劳-希尔国际公司合作出版。未经出版者书面许可, 不得以任何方式复制或抄袭本书内容。

版权所有, 侵权必究。

本书版权登记号: 图字: 01-2001-0480

图书在版编目(CIP)数据

Solaris 8技术大全/(美)瓦特斯(Watters, P.A.), (美)维拉拉格哈温(Veeraraghavan, S.)著; 李明之等译.- 北京: 机械工业出版社, 2002.1

(Sun公司核心技术书库)

书名原文: Solaris 8: The Complete Reference

ISBN 7-111-09399-2

I.S... II.①瓦... ②维... ③李... III.操作系统(软件), Solaris 8 IV.TP316.89

中国版本图书馆CIP数据核字(2001)第072670号

机械工业出版社(北京市西城区百万庄大街22号 邮政编码 100037)

责任编辑: 高智勇 张鸿斌

北京牛山世兴印刷厂印刷·新华书店北京发行所发行

2002年1月第1版第1次印刷

787mm×1092mm 1/16·29印张

印数: 0 001-5 000册

定价: 48.00元

凡购本书, 如有倒页、脱页、缺页, 由本社发行部调换

前 言

客户经常会问我们为什么要选择Solaris作为操作环境。是因为价格吗？是因为最新功能吗？还是因为我们与Sun公司达成某种协议来吹嘘它们的操作系统？这些原因都不是。操作系统是任何计算机系统的核心的核心。它是建立应用的基石，它实现了关键的软件到硬件的接口，这样才使得计算成为可能。这样，如果操作系统失败（不是因为硬件质量或应用软件质量引起的），整个系统就都不能运转了。这就是为什么我们坚信可靠性是评价任何操作系统的关键条件。

可靠性不是由几个特定变量来决定的，它是一种信心，来源于知道只有自己的应用出现错误才会导致失败的信心。Solaris并不完美——远远不是——但是，不像当前其他流行的操作系统那样，Solaris有能力将所有用户或应用的潜在灾难风险隔离开来。UNIX系统早期就使用了逻辑隔离方法，正是这种逻辑上的隔离给了我们信心。

Solaris操作环境由所有的工具、应用和利用SunOS的服务组成。SunOS现在是版本5，10年前的SunOS才只是版本4！这说明虽然操作环境的数字看起来高速膨胀，但是核心操作系统只是在增强核心功能方面做了改变。因此，许多10年前为SunOS写的软件可能现在还能够移植运行。在迅速变化的因特网经济里，还存在着一块世外桃源——可靠的、稳定的解决方案使系统管理员晚上能安稳地睡觉。

当然，在最近几年中SunOS也开始出现一些变化来跟上硬件技术的进步。现在Solaris出售版本中同时包含32位和64位内核。尤其是对于使用数据库和科学搜索这些计算密集的任务，64位能够可以大幅度地提升Solaris系统的性能。在很多情况下，这种性能的提升甚至不需要修改多少源代码：应用可以“继承”操作系统的特性。

本书可以作为Solaris发行包中随带的CD-ROM中已经提供的出色的参考资料的伴侣。读者也可以自己下载或在Solaris文档站点(<http://docs.sun.com/>)搜索。可能读者会问，既然我们已经有了Sun的参考指南，那么为什么还要买这本书呢？原因有三。首先，作者不是Sun公司的员工。我们是从经常使用其他操作系统的经验丰富的专家角度来写本书的，所以不会被市场宣传所迷惑。所以本书能够在该表扬的地方表扬，但是也明确指明了该操作环境中存在的某些不足。我们以安全远程登录为例：Solaris 8 包括了最新的Kerberos 5认证系统，但是却没有提供一个远程登录用的安全shell。我们意识到许多系统管理员和UNIX用户都经常使用Telnet，但是说实话，哪个组织不想保护自己的用户名和口令呢？幸运的是，我们看到第三方的远程登录软件可以提供很好的安全保护，带来心理上的踏实。

其次，本书不是简单的命令解释（就像Solaris中的手册页一样），而是希望能够介绍系统操作中的某些特定领域，并详细说明Solaris发行包中随带的软件和其他第三方软件的优缺点。此外，我们还尝试着解释为什么要进行某些操作（例如备份），而不仅仅是机械地告诉读者怎样做。我们还排除了某些纯粹的命令参考资料。如果你想查找某个特殊命令的某个选项，请使用手册页，这也是设计手册页的原因。然而，如果你想查阅实际命令的具体实例，使用最常用的选项，

本书最适合你。

第三,我们还努力将Solaris资料和其他操作系统的实现进行了有益的对照。对于新用户来说,最困难的事情就是要记住不同操作系统的所有新命令和例程。我们希望这些用户可以更有效地重新利用原来的知识。例如,我们还包含了与微软公司的Windows和苹果公司的Macintosh系统之间进行打印和文件共享的资料,以便系统管理员能够在这些异构环境中可以充分利用系统硬件和打印技术方面的已有投资。我们希望这些处理操作系统和普通组织背景下的实用知识对所有的读者都有价值。

本书对于Solaris的初级、中级和高级用户来说都是个综合的指南。初级用户会发现讲述Solaris优势和怎样安装Solaris系统的最初的几章有用。例如,你可能是个Solaris x86的新用户,想多了解一下选择硬件的知识以及可用的安装选项。中级用户可能已经知道了Solaris的优势,但可能会对学习Solaris中某个特殊方面感兴趣。例如,我们覆盖了高级网络服务的配置,例如邮件传输代理sendmail和Apache网页服务器(包括Java servlets)。高级用户最有可能在某个特殊场合使用本书,例如,设置NIS+域、利用联合域名系统(Federated Naming System)。所有这些内容中,本书都提供了背景知识和实际的例子,以方便Solaris的用户和管理员为了特定的需求最好地利用操作环境。

我们希望多年后你仍然认为本书是本有益的参考书。

本书英文原书书名: Solaris 8 : The Complete Reference

英文原书书号: ISBN 0-07-212143-2

原书出版社网址: www.osborne.com

参加本书翻译工作的人员有:李明之、裴晓峰、张磊、汤钊、钱海春、吴晖、赵粮、张侃等。

目 录

前言

第一部分 Solaris基础知识

第1章 Solaris 8简介	1
1.1 什么是UNIX	5
1.2 UNIX的历史	6
1.2.1 UNIX的起源	6
1.2.2 BSD的特征	8
1.2.3 System V版本4的特征	10
1.3 Solaris系统的改进	11
1.3.1 硬件支持	13
1.3.2 跨平台内部合作	13
1.4 额外信息的出处	14
1.4.1 Sun文档	14
1.4.2 Web站点	15
1.4.3 USENET	15
1.4.4 邮件列表	16
1.5 在Solaris 8中有什么新东西	16
1.5.1 Star Office	18
1.5.2 移动计算	20
1.5.3 PC支持	20
1.6 服务器工具	22
1.6.1 群集技术	22
1.6.2 lxrun	23
1.7 改进的安全性	24
1.7.1 Kerberos 版本5	24
1.7.2 IPv6	24
1.7.3 IPSec	25
1.8 本书的总览	25
1.9 小结	27
1.10 如何得到更多信息	27
第2章 Solaris的安装	28
2.1 如何得到Solaris	28
2.2 预安装	29
2.2.1 决定硬盘分区	29
2.2.2 确定网络参数	30
2.2.3 x86系统所需的额外预安装	31
2.3 安装Solaris	33
2.3.1 安装到x86系统上	33
2.3.2 通用安装过程	35
2.3.3 安装步骤	37
2.4 小结	46
2.5 如何得到更多信息	47
第3章 起步	48
3.1 CDE环境	48
3.1.1 登录CDE	49
3.1.2 dtterm	53
3.1.3 定制CDE	57
3.2 Solaris文件系统结构	59
3.2.1 设备目录	60
3.2.2 程序目录	61
3.2.3 配置目录	61
3.2.4 存放临时文件的目录	62
3.3 小结	63
3.4 如何得到更多信息	63
第4章 系统启动和关机	65
4.1 System V启动过程	66
4.2 各个启动阶段	69
4.2.1 init	71
4.2.2 inittab	72
4.2.3 控制脚本/目录	74
4.2.4 /etc/init.d	75
4.2.5 改变运行等级	76
4.3 系统关机	76
4.3.1 使用 shutdown 或 init	78

4.3.2 Kill脚本	78
4.4 小结	79
4.5 如何得到更多信息	80
第5章 进程管理	81
5.1 进程	81
5.2 得到进程的详细信息	89
5.2.1 proc工具	90
5.2.2 lsof	93
5.3 发送信号	95
5.4 自动运行任务	97
5.4.1 使用at	97
5.4.2 使用cron	98
5.5 小结	99
5.6 如何得到更多信息	100

第二部分 系统维护

第6章 用户和软件管理	101
6.1 用户管理	101
6.1.1 用户数据库	101
6.1.2 管理用户数据库	108
6.2 软件管理	114
6.2.1 管理软件包	114
6.2.2 管理补丁	120
6.3 小结	125
6.4 如何得到更多信息	125
第7章 设备管理	126
7.1 设备文件	126
7.1.1 /dev 和 /devices	129
7.1.2 存储设备	132
7.1.3 串行设备	137
7.2 设备检测	138
7.2.1 启动时的情况	140
7.2.2 当系统已经启动时	142
7.3 增加设备	149
7.3.1 硬盘	150
7.3.2 CD-ROM	154
7.3.3 CD-R 和CD-RW	154

7.4 小结	155
7.5 如何得到更多信息	155
第8章 基本文件系统管理	156
8.1 文件系统基础	156
8.1.1 特殊的磁盘块	157
8.1.2 虚拟文件系统接口	159
8.2 创建文件系统	162
8.2.1 确定硬盘的设备	162
8.2.2 格式化和分区	163
8.2.3 生成一个新的UFS文件系统	169
8.3 加载和卸载文件系统	171
8.3.1 加载文件系统	171
8.3.2 卸载文件系统	174
8.3.3 加载配置文件	176
8.3.4 空间管理	176
8.4 小结	178
8.5 如何得到更多信息	178
第9章 高级文件系统管理	179
9.1 监测使用情况	179
9.1.1 监测使用情况	179
9.1.2 实行配额	187
9.2 用fsck解决问题	189
9.2.1 fsck的步骤	189
9.2.2 fsck实例	191
9.3 系统记账	193
9.3.1 数据收集	194
9.3.2 生成报告	194
9.3.3 费率	197
9.4 小结	197
9.5 如何得到更多信息	197
第10章 备份	200
10.1 理解备份	200
10.1.1 为什么需要备份	200
10.1.2 确定备份策略	201
10.2 选择备份媒介	203
10.2.1 磁带驱动器	203
10.2.2 硬盘驱动器	204

10.2.3 Zip/Jaz 盘	205	12.4 IP过滤	258
10.2.4 CD-R 和 CD-RW	205	12.4.1 什么是IP过滤	259
10.3 执行备份和恢复	205	12.4.2 IP过滤数据包	260
10.3.1 标准Solaris工具	206	12.5 小结	262
10.3.2 集中和分布式工具	213	12.6 如何得到更多信息	262
10.4 小结	216	第13章 域名服务	263
10.5 如何得到更多信息	216	13.1 DNS概述	263
第三部分 网络管理基础知识		13.1.1 互联网历史回顾	263
第11章 网络基础	217	13.1.2 DNS中的域结构	265
11.1 TCP/IP概述	217	13.1.3 DNS查询	267
11.2 TCP/IP协议族结构	218	13.2 配置DNS用户	268
11.2.1 层和协议	218	13.2.1 域名解析配置文件	269
11.2.2 应用层	219	13.2.2 网络服务配置文件	270
11.2.3 传输层	219	13.3 配置DNS服务器	271
11.2.4 网络层	220	13.3.1 master服务器	272
11.2.5 网络访问层	221	13.3.2 slave服务器	272
11.2.6 IP地址	222	13.3.3 hint服务器	273
11.2.7 网络传输协议	224	13.3.4 名字服务器配置文件	273
11.3 TCP/IP启动和调试	227	13.3.5 区数据库和资源记录	279
11.3.1 TCP/IP启动	227	13.3.6 样本区数据库文件	284
11.3.2 网络调试工具	229	13.3.7 名字服务器的启动和结束	287
11.4 小结	242	13.3.8 测试名字服务器	287
11.5 如何得到更多信息	242	13.3.9 保护DNS服务器	290
第12章 路由	244	13.4 小结	292
12.1 网络接口	244	13.5 如何得到更多信息	292
12.1.1 检查接口	250	第14章 动态主机配置协议	293
12.1.2 修改接口参数	251	14.1 什么是DHCP	293
12.2 IP路由	253	14.1.1 DHCP如何工作	293
12.2.1 数据包传输概述	254	14.1.2 DHCP的局限性	295
12.2.2 路由追踪	255	14.2 配置DHCP服务器	296
12.2.3 静态路由	256	14.2.1 分配IP地址块	296
12.2.4 路由协议	256	14.2.2 使用dhcpconfig	297
12.3 内核路由表	257	14.2.3 自动启动和停止DHCP服务器	304
12.3.1 查看路由表	257	14.3 配置DHCP客户机	304
12.3.2 处理路由表	257	14.3.1 手动配置DHCP客户机	305
12.3.3 动态路由	258	14.3.2 引导时自动配置DHCP	307

14.4 小结	307
14.5 如何得到更多信息	307
第15章 网络信息服务	308
15.1 NIS+概述	308
15.1.1 NIS+为何可以改进网络数据管理	309
15.1.2 NIS映像文件	311
15.1.3 NIS+表	312
15.1.4 名字服务开关	314
15.2 配置NIS	315
15.2.1 建立根域	316
15.2.2 填充表格	316
15.2.3 建立客户机	316
15.2.4 设置服务器	317
15.3 NIS+工具	318
15.3.1 nisdefaults	318
15.3.2 nischmod	318
15.3.3 nislsl	319
15.3.4 niscat	320
15.4 小结	322
15.5 如何得到更多信息	322

第四部分 服务管理

第16章 管理远程访问	323
16.1 标准远程访问工具	323
16.1.1 Telnet	323
16.1.2 R-命令	326
16.1.3 安全问题	327
16.2 安全工具	328
16.2.1 安全Shell	328
16.2.2 Kerberos	330
16.2.3 SRP	332
16.2.4 TCP包装器	332
16.3 Syslog	334
16.3.1 什么是Syslog	334
16.3.2 检查日志文件	334
16.3.3 syslog.conf文件	335
16.3.4 监视系统访问	336

16.4 小结	336
16.5 如何得到更多信息	337
第17章 使用NFS与UNIX系统共享文件	338
17.1 什么是NFS	338
17.1.1 为什么使用NFS	339
17.1.2 NFS的守护进程	340
17.1.3 远程过程调用协议	340
17.2 建立NFS服务器	341
17.2.1 NFS服务器的启动和关闭	341
17.2.2 输出文件系统	343
17.2.3 输出文件系统的规则	348
17.3 建立NFS客户机	349
17.3.1 使用mount命令	350
17.3.2 启动与关闭NFS客户机	354
17.3.3 排除NFS文件系统故障	354
17.4 使用Automounter工具	359
17.4.1 Automounter映像	359
17.4.2 启动和关闭Automounter	361
17.5 打印	362
17.5.1 配置本地打印机	363
17.5.2 配置远程打印机	364
17.5.3 用OpenWindows打印	365
17.6 小结	366
17.7 如何得到更多信息	366
第18章 用SAMBA和Netatalk与PC和 MAC共享文件	367
18.1 Windows和MacOS的文件共享	367
18.1.1 NetBIOS和服务器消息块	367
18.1.2 AppleTalk	371
18.2 SAMBA	373
18.2.1 安装SAMBA	373
18.2.2 配置SAMBA	377
18.2.3 配置高级服务器访问	381
18.2.4 SAMBA的替代品	383
18.3 Netatalk	384
18.3.1 安装Netatalk	384
18.3.2 配置Netatalk	386

18.3.3 Netatalk的替代品	389	20.1.6 激活FTP访问	420
18.4 小结	390	20.1.7 建立FTP服务	420
18.5 如何得到更多信息	390	20.1.8 建立匿名FTP	420
第19章 管理邮件	391	20.1.9 提高FTP的安全性	421
19.1 理解电子邮件协议	391	20.1.10 问题解答	422
19.1.1 什么是SMTP	392	20.2 其他FTP服务器	422
19.1.2 邮件头	395	20.2.1 WU-FTPD	422
19.1.3 MIME	395	20.2.2 BeroFTPD	423
19.1.4 什么是POP	397	20.2.3 Troll FTPD	423
19.1.5 什么是IMAP	398	20.3 小结	424
19.2 使用sendmail	399	20.4 如何得到更多信息	424
19.2.1 什么是sendmail	399	第21章 Web服务	425
19.2.2 配置sendmail	399	21.1 HTTP介绍	425
19.2.3 运行sendmail	402	21.1.1 HTTP请求	426
19.2.4 使用别名	402	21.1.2 HTTP回应	428
19.2.5 问题解答	403	21.1.3 CGI和Servlet	430
19.2.6 安全	405	21.2 安装Apache	434
19.3 邮件客户机	405	21.2.1 获得Apache的方法	435
19.3.1 本地客户机程序	405	21.2.2 编译和安装前的注意事项	435
19.3.2 远程客户机程序	407	21.2.3 配置、编译和安装Apache	436
19.4 小结	409	21.2.4 测试Apache	438
19.5 如何得到更多信息	409	21.3 为用户配置Apache	439
第20章 FTP管理	410	21.3.1 全局配置	439
20.1 理解FTP	410	21.3.2 服务器特性配置	442
20.1.1 什么是FTP	410	21.3.3 虚拟主机配置	447
20.1.2 FTP内部命令	412	21.3.4 完整的httpd.conf文件	448
20.1.3 FTP过程实例	415	21.4 小结	451
20.1.4 使用匿名FTP	418	21.5 如何得到更多信息	451
20.1.5 GUI FTP客户机	419		

第一部分 Solaris基础知识

第1章 Solaris 8 简介

操作系统是计算机系统的组成部分之一，它提供了用户程序和计算机硬件之间的接口。Solaris 8是由Sun Microsystems (<http://www.sun.com/>)开发和销售的一种多用户、多任务操作系统，它是UNIX操作系统的一种变体，并且吸取了System V (AT&T)和Berkeley (BSD)两家的优点，起初它只不过是一个研究项目，如今已成为国际市场上UNIX操作系统的主导。Solaris 8是基于SunOS操作系统（目前已是版本5.8了）的一系列操作环境的一个新成员。Solaris常用于大公司或教育机构中，这些地方往往要求多用户同时访问一台主机或者要求主机之间通过Internet相连，但是，通过Sun推出的“免费Solaris”行动，Solaris正越来越被小企业和个人开发者采用。在本书中，虽然我们针对的是Solaris 8，许多命令和进程也同样适用于早前的Solaris 2.x 版本，当关系到Solaris 1.x的命令时，本书将特别注明。

许多桌面计算机用户从未听说过在计算领域还有一个“Sun”，通常他们也不熟悉“Solaris”是一种操作环境，然而，当一个Internet用户发送一条电子邮件消息或从一台运行Sun的NFS (Network File System) 的连网服务器上打开文件时，几乎每次Solaris 8都在支持着现今的Internet应用。在企业计算领域，Sun是高可用性和高可靠性硬件的同义词，而Solaris 8常是运行数据库和应用服务器的首选操作环境。Sun的硬件基于采用 SPARC和 UltraSPARC的集成电路技术，它们目前可以在单个服务器系统中支持64个处理器 (E10000 StarFire 的配置)。如图1-1所示，在软件开发中，Sun 也支持包括CDE (Common Desktop Environment) 在内的通用业界标准。

最近，Sun的两个创新使得人们把注意力从服务器领域转到了桌面环境。第一，Sun开发了Java编程语言，它保证用Java编写的应用程序可以在任何支持Java虚拟机的平台上运行，这给联网应用的开发带来了革命。另外，现在在许多Web页面上都有Java applets (运行在客户端的嵌入式小应用)，并且许多三层应用的后端都由servlets制成，如CRM 和复杂的 HR 应用。

其次，Sun正在推行SPARC和Intel硬件平台上的“免费” Solaris 8版本，这表明，以前那些使用Intel平台而不得不选用Microsoft Windows 或 SCO OpenServer 的机构，可以依旧使用装着这些操作系统的服务器，而在上面改装基于Intel的Solaris 8。Intel硬件平台上的“免费” Solaris 8版本也使得更多的桌面用户可以使用Solaris 8，从而这些用户可以如他们所期望的那样使用Sun的其他免费产品，如Star Office产品包。Star Office完全可以和Microsoft Office相媲美，因为它完全集成了字处理、表格、演示幻灯片和数据库等功能。Star Office接口如图1-2所示。另处，Star Office可以运行在多种不同的平台上，并支持8种语言，这样Sun SPARCstation上的用户可以毫无困难地和Linux及Microsoft Windows用户共享文件。将强健的操作系统和最好的产品套件结合在一起使Solaris 8开拓了新的桌面市场。

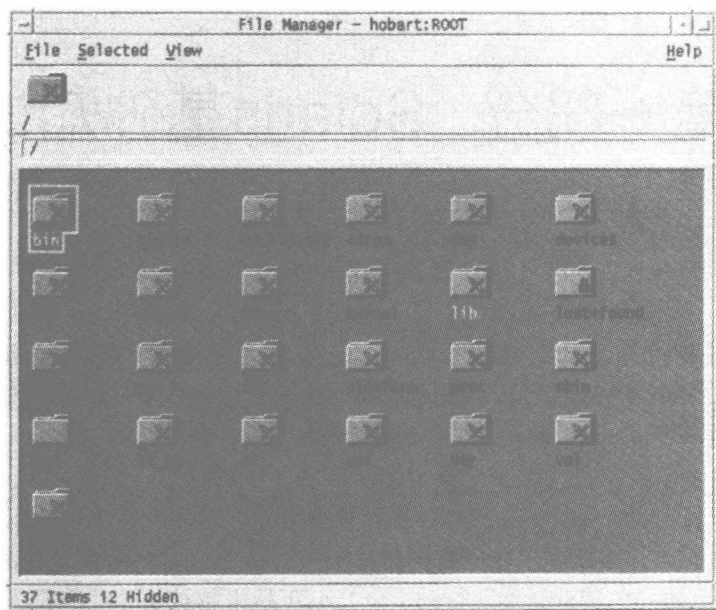


图1-1 公共桌面环境

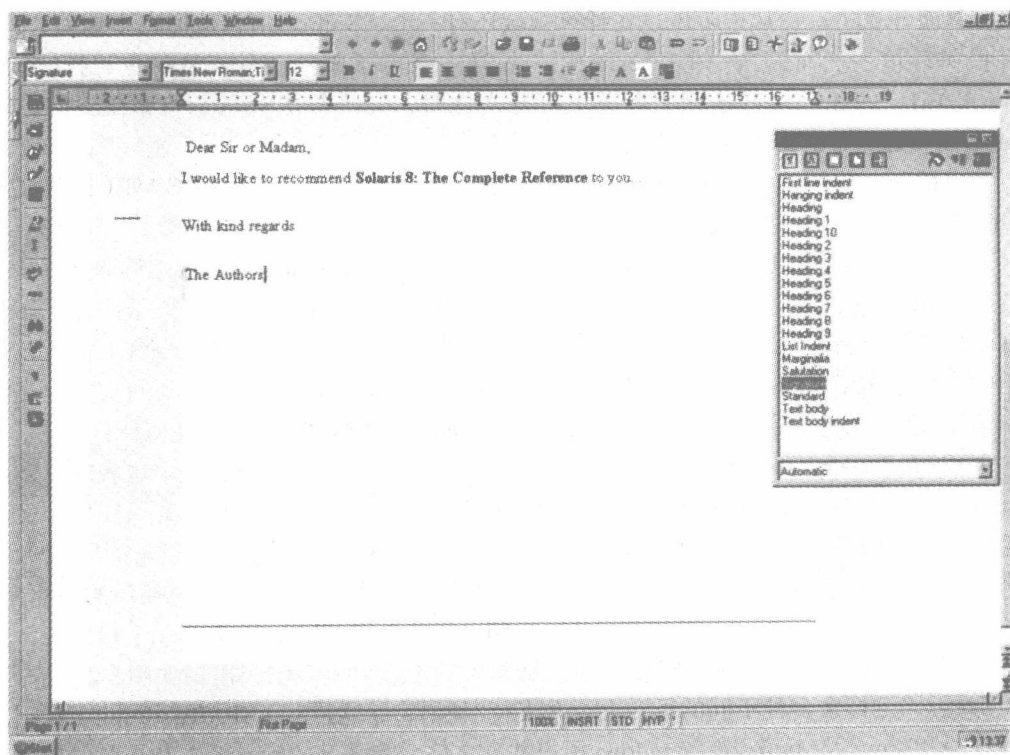


图1-2 Star Office办公套件

本书是 Solaris 8 运行环境和 SunOS 5.8 操作系统的参考大全，我们试着全面而又详细地描述Solaris和SunOS的操作运行。如果你只是想简单地查看一下某条命令的选项，你可以通过键入 `man command` 来使用Sun自己的在线手册，其中`command`是你查看的命令，也可以像图1-3中所示那样，利用 `http://docs.sun.com/` 的搜索功能，在线检索联机文本和用户手册。当你想实现一种特定的解决方案，并且需要对这个方案进行实践、尝试和测试时，本参考大全就特别有用。例如，在第20章中我们研究了如何建立一个匿名FTP服务器，在第16章中我们了解了安全的远程访问机制。虽然Solaris 8缺省带有一系列的远程访问工具和服务器，但从安全角度来看，当用于产品中时，这些并不总是最好的工具。例如，在局域网中完全可以用FTP进行文件传送，但远程数据交换却应采用安全的文件传送系统，如sftp。在全面介绍对一个问题的解决办法时，我们通常先介绍Sun提供的软件，然后再讨论其他可选用的第三方产品的安装和配置。



图1-3 在docs.sun.com中找Solaris参考材料

如果你一直注意着最近发布的新闻，你可能会奇怪为什么 SunOS 的版本号为5.8，而Solaris却有一个为8的版本号。从Sun发布前一个Solaris版本（Solaris 7）后，Sun决定根据以前版本号中的小数位只用一个版本数字顺序标注它发布的新产品，这表示发布的 Solaris 序号为2.5.1, 2.6, 7和现在的 8。这样，许多站点将继续运行Solaris 2.6也不会感觉到落后太多，特别是当它们并不要求使用Solaris 7及更高版本为Ultra SPARC处理器提供的64位功能时。Sun确实为以前发布的操

作系统提供了大量的补丁，为保证尽快地解除漏洞（特别是安全漏洞）就需要安装它。在本书中谈到的绝大多数 Solaris 8 命令和一些问题也同样适用于 Solaris 7 以及以前所发布的所有 SunOS 5.x。

然而，只要可能，我们也包括进了 SunOS 4.x操作系统的内容，SunOS 4.x被标记为 Solaris 1.x平台。安装的许多用户只是在最近几年才开始用 SunOS 5.x，并且直到 SunOS 4.x出现了“千年虫”问题以前，总是许多站点还在这种平台上运行老的应用（尤其是当与 System V风格的 SunOS 5.x操作系统相比他们更钟爱 BSD 风格的 SunOS 4.x时）。许多 Internet 防火墙、邮件服务器和新闻服务器仍在各种 SPARC结构的 CPU上运行，而其中某些结构并未得到 SunOS 5.x操作系统的支持。对于许多 Sun用户来说，是否从 SunOS 4.x升级到 SunOS 5.x 是一个艰难的决定。

还好，随着Solaris 7和8的发布，64位计算的时代到来了，我们有了一个稳定的平台，Solaris 7和8的许多区别只是锦上添花型的，例如，发布 Solaris 8 时第一次将 Larry Wall的 Perl解释器包括了进来，这意味着新一代的系统管理员失去了新系统安装后的第一件快乐任务。但是，在网络和管理方面仍有许多非常重要的发展（例如Solstice AdminSuite，如图1-4所示），它们虽然可能不会影响所有的用户，但对于企业应用来说却非常重要。

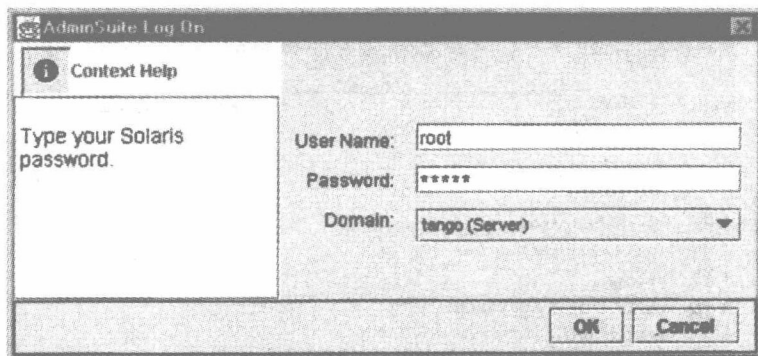


图1-4 Solstice AdminSuite启动界面

在这一章中，我们将介绍Solaris 8运行环境的背景，这实际上要从UNIX操作系统的发明和被广泛采用说起。另外，我们还将讨论Solaris 8之所以可以运行跨平台应用的原因，例如，通过使用一个称为Lxrun的应用，基于Intel的Solaris 8可以运行Linux二进制应用。Lxrun可以免费从Sun那里得到。虽然早些那些试图模仿其他操作系统的努力大多数没有成功（如想模仿Microsoft Windows的WABI），但Sun开发Java也可看作是对跨平台兼容性的有力许诺。Solaris 8还提供了许多网络管理功能，使得Solaris 8服务器可以作为管理 Windows NT客户的主用或备用域控制器，例如，当你想让广泛采用的Microsoft Windows桌面操作系统与Solaris 8的可靠性相结合时就可以这么做。

最后，我们将总结Internet上提供有用信息、软件包的大量站点中的一部分，以及与本书中许多内容有关的参考书。

1.1 什么是UNIX

定义什么是UNIX是件很难的事,由于历史的原因,不同的UNIX厂商在各自的UNIX实现中引进了不同的特性,而多数UNIX用户总是习惯上将这些特性当作UNIX的一部分。但是,将所有UNIX和类UNIX系统都具备的一些基本共性罗列一下倒不是件困难的事。

- 它们有一个内核,使用C编程语言,主要用来管理输入/输出进程,而不作为一个完整的操作系统。内核拥有最高的权限来分配系统资源去完成各种不同的任务。
- 它们有一个分层的文件系统,从根目录开始,在此之上加载由其他所有目录(和文件系统)构成的分支。
- 系统硬件设备以逻辑的形式表达为文件系统上的特殊文件(如 `/dev/pty`, 表示伪终端)。
- 它们都是基于进程的,用一个标识数字(进程ID或PID)表示所有服务和用户 shell。
- 它们都有一套命令行工具,用于各种各样的文本和数字处理,如 `troff`、`col`、`cat`、`head`和`tbl`。
- 用户进程可以从一个shell启动,如Bourne shell,通过它可以交互式地执行应用程序。
- 每个用户可以同时运行多个进程,并且可以使用“&”操作符将其转入后台运行。
- 多个用户可以通过从伪终端登录的方式同时执行命令。

虽然大多数UNIX支持X11和通用桌面环境(CDE)图形界面,但是图形用户界面(GUI)并不是UNIX的定义特征,这与一些其他的操作系统不一样,它们感觉上过分强调图形化的界面。图1-5是一个典型的CDE登录屏幕。

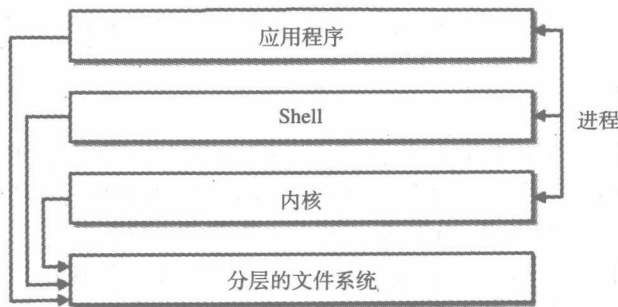


图1-5 UNIX系统组件

这种区别主要来源于历史原因,与本章后面将要介绍的UNIX体系的设计哲学有关系。图1-6中给出了UNIX系统各组件的分层关系。由于UNIX是由一些活跃的开发人员而不是由一些操作系统的权威们开发的,因此开发人员们特别致力于制作一种适合于程序开发员的操作系统。在1978年的Bell System Technical Journal 中列出了一些在UNIX开发中指导性的原则:

- 生成小巧的,可以独立完成一个单一任务的程序。当需要解决一个新的任务时,可以生成一个新的程序完成这个任务,也可以从已有的工具集中组合出一个解决方案来。这与当前封装和独立组件构造的趋势非常相似(例如Enterprise Java Beans),也就是说复杂的系统由小巧的、相互作用但又互相独立的模块来组成。
- 程序应该从标准输入来接收数据并输出到标准输出,因而,程序可以串起来使用,后面的程序处理前面程序的输出。交互式输入要尽量避免,而使用命令行选项来指定该程序要执

行的动作。表示和程序要实现的目标应该分开。这些观点和管道的概念非常一致。管道对用户shell的操作非常关键。例如，“ls”命令（列出一个目录中的所有文件）的输出可以被管道定向到（使用管道“|”符号）另外的程序，例如“grep”，来进行模式匹配。在一个命令行中的管道数目不受限制。

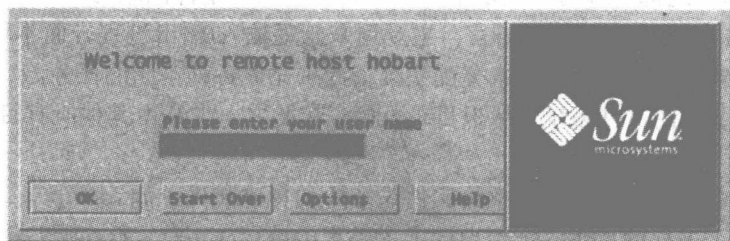


图1-6 CDE登录界面

- 创建一个新的操作系统或程序的时间应该在数个星期的范围内，而不是几年。要充分利用这种能够导致继承性设计和实现的创新精神。如果软件不能工作，一定要勇于去创建更好的软件。程序的这种重复修订过程在近几年来体现为面向对象开发的兴起。
- 尽可能地利用现有的工具，而不是要求过多的帮助。UNIX背后的动机就是建立一个能够支持成功的开发所需的各种各样工具的操作系统。

这里并不想过多罗列定义UNIX的特征，然而，这些特征对于理解UNIX开发者经常强调的操作系统的重要性非常关键。UNIX从设计开始就是一个对程序员非常友好的操作系统。

1.2 UNIX的历史

在20世纪60年代后期开始的一个专用研究项目中，Bell实验室的一个小组最早开始开发UNIX。在前面的10年中，这个小组尝试了若干个不同的相关操作系统的研究。他们的UNIX项目的目标是使用新出现的第三代编程语言设计一个透明、简单、易修改的操作系统。当时，典型的厂家提供的操作系统随厂家而异，大多非常巨大，使用汇编语言编写，非常难于维护。虽然，最初的UNIX内核编写是基于汇编语言的，然而随后的版本都是使用同时代发展起来的一个高级语言C来编写的。即使到了今天，多数现代操作系统的内核，例如Linux内核，也是使用C写的。在使用第一个C编译器开发完内核后，接着开发出来一个完整的操作系统，包括今天的UNIX仍然使用的许多应用工具（例如可视化编辑器vi）。在本节，我们将考察导致UNIX开发的时间线索以及两种不同的UNIX风格的起源：AT&T (System V) 和BSD。

1.2.1 UNIX的起源

1969年，贝尔实验室的Ken Thompson 在一台DEC PDP-7上编写了UNIX操作系统的第一个版本。鉴于对Multics（Multiplexed Information and Computing Service）项目效率低下的反省，Thompson 决定创建一个对程序员友好的操作系统，它限制内核中包含的函数数量，允许应用的设计和实现有更大的灵活性。对于建设一个新的操作系统来说，PDP-7是一个不错的系统。它只有一个汇编器和一个加载器，并且只允许单一用户登录。它甚至没有一个硬盘；开发商被迫将

物理内存分为一个操作系统部分和一个RAM盘部分。因此，第一个UNIX文件系统整个都在一个RAM中实现。

在成功的在PDP-7上发明了一个单用户版本的UNIX之后，Thompson和他的同事Dennis Ritchie在1970年将这个系统转移到一个规模大很多的DEC PDP-11/20系统上面。该计划的经费来源于这样一个需求，即为专利权建立一个文本处理系统，作为那些已经在例如troff一类的文本过滤器中存在的系统的后代来筹措经费。创建应用程序的需要最终导致Ritchie开发了第一个C编译器，这是建立在B语言的基础上的。C语言的设计很注重可移植性，在源代码中可用同一函数调用表示特定平台的库，而源代码也同样可以在另一不同的硬件平台上面编译。尽管PDP-11比PDP-7性能优良，但是与今天的科学计算器相比，它还是差得很遥远，它仅有24K的可寻址内存，其中12K为操作系统保留。在1972年以前，全世界的UNIX装机数已经增长到10。

UNIX发展的下一个主要的里程碑是内核用C语言的重写，这在1973年由Ritchie 和Thompson完成。这解释了为什么UNIX和C关系很牢固：即使在今天，大多数的UNIX应用程序都是用C语言写的，虽然早已有许多编程语言可供使用了。随着C语言内核的发展，UNIX的拥有者(AT&T)开始在美国和国外将源代码授权给教育机构。但是，这些许可经常受到限制，并且这种授权应用得并不广泛。没有任何支持，并且没有机构去正式地修改漏洞。由于用户可以得到源代码，攻击代码的天赋——这个天赋现在还在像Linux那样的群体项目中存在——汇聚成了一种潮流，特别是在伯克利大学里，对UNIX的授权和AT&T对它的控制的结果决定了在以后的日子里操作系统将要面临的分裂。

在1975年，第一次UNIX软件的分发包由一个伯克利小组成功实现，并且被称为Berkeley Software Distribution (BSD)。Berkeley是Ken Thompson的母校，他和两个研究生（Bill Joy和Chuck Haley）组成了一个小组，他们后来成为UNIX世界中的领导人物。他们完成了一个被作为BSD的一部分发表的UNIX Pascal编译器，并且Bill Joy还书写了vi的第一个版本，后来vi（虚拟编辑器）一直非常受欢迎。

在1978年，操作系统的第七个版本发表，它支持很多不同的硬件结构，包括IBM 360, Interdata 8/32和Interdata 7/32。第七版的内核仅仅有40K大小，它包括以下系统调用：_exit, access, acct, alarm, brk, chdir, chmod, chown, chroot, close, creat, dup, dup2, exec*, exit, fork, fstat, ftime, getegid, geteuid, getgid, getpid, getuid, gtty, indir, ioctl, kill, link, lock, lseek, mknod, mount, mpxcall, nice, open, pause, phys, pipe, pkoff, pkon, profil, ptrace, read, sbrk, setgid, setuid, signal, stat, stime, stty, sync, tell, time, times, umask, umount, unlink, utime, wait, write。第七版的完全手册在 <http://plan9.bell-labs.com/7thEdMan/index.html> 上在线提供。

随着UNIX第七版在全球的流行，AT&T意识到UNIX可能是一个很有商业价值的产品，并且试图限制在大学课程里面传授UNIX的源代码，通过这个方法保护宝贵的智力财富，另外，AT&T第一次开始对UNIX源代码的访问索要授权的费用。这促使UCB组织去创建他们自己的UNIX变体。BSD分包除了拥有传统的、开始就构成这个包的应用程序外，还包括一个完整的操作系统。结果，第七版成为现在所有正在使用的UNIX版本的基础。该版的UNIX同样包括一个完整的Brian Kernighan和Ritchie的C语言编译器，还有Bourne shell。UNIX分裂成AT&T和BSD特色的两个分支，这个过程一直到今天还在继续，像下面讨论的那样，很多商业系统，例