

系统管理指南：高级管理

版权所有 © 1998, 2011, Oracle 和/或其附属公司。保留所有权利。

本软件和相关文档是根据许可证协议提供的，该许可证协议中规定了关于使用和公开本软件和相关文档的各种限制，并受知识产权法的保护。除非在许可证协议中明确许可或适用法律明确授权，否则不得以任何形式、任何方式使用、拷贝、复制、翻译、广播、修改、授权、传播、分发、展示、执行、发布或显示本软件和相关文档的任何部分。除非法律要求实现互操作，否则严禁对本软件进行逆向工程设计、反汇编或反编译。

此文档所含信息可能随时被修改，恕不另行通知，我们不保证该信息没有错误。如果贵方发现任何问题，请书面通知我们。

如果将本软件或相关文档交付给美国政府，或者交付给以美国政府名义获得许可证的任何机构，必须符合以下规定：

U.S. GOVERNMENT END USERS:

Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

本软件或硬件是为了在各种信息管理应用领域内的一般使用而开发的。它不应被应用于任何存在危险或潜在危险的应用领域，也不是为此而开发的，其中包括可能会产生人身伤害的应用领域。如果在危险应用领域内使用本软件或硬件，贵方应负责采取所有适当的防范措施，包括备份、冗余和其它确保安全使用本软件或硬件的措施。对于因在危险应用领域内使用本软件或硬件所造成的一切损失或损害，Oracle Corporation 及其附属公司概不负责。

Oracle 和 Java 是 Oracle 和/或其附属公司的注册商标。其他名称可能是各自所有者的商标。

Intel 和 Intel Xeon 是 Intel Corporation 的商标或注册商标。所有 SPARC 商标均是 SPARC International, Inc 的商标或注册商标，并应按照许可证的规定使用。AMD、Opteron、AMD 徽标以及 AMD Opteron 徽标是 Advanced Micro Devices 的商标或注册商标。UNIX 是 The Open Group 的注册商标。

本软件或硬件以及文档可能提供了访问第三方内容、产品和服务的方式或有关这些内容、产品和服务的信息。对于第三方内容、产品和服务，Oracle Corporation 及其附属公司明确表示不承担任何种类的担保，亦不对其承担任何责任。对于因访问或使用第三方内容、产品和服务所造成的任何损失、成本或损害，Oracle Corporation 及其附属公司概不负责。

目录

前言	15
1 管理终端和调制解调器（概述）	19
管理终端和调制解调器方面的新增功能	19
SPARC: 相关控制台	19
SPARC: 对如何设置控制台的 \$TERM 值的更改	20
系统控制台上由 SMF 管理的 ttymon 调用	20
终端、调制解调器、端口和服务	20
终端说明	21
调制解调器描述	21
端口描述	21
服务描述	22
端口监视器	22
终端和调制解调器的管理工具	22
串行端口工具	23
服务访问工具概述	23
2 设置终端和调制解调器（任务）	25
设置终端和调制解调器（任务列表）	25
使用串行端口工具设置终端和调制解调器（概述）	26
设置终端	26
设置调制解调器	27
如何设置终端、调制解调器和初始化端口（任务）	29
▼ 如何设置终端	29
▼ 如何设置调制解调器	30
▼ 如何初始化端口	31
解决终端和调制解调器问题	31

3 使用服务访问工具管理串行端口（任务）	33
管理串行端口（任务列表）	34
使用服务访问工具	34
整体 SAF 管理 (sacadm)	35
服务访问控制器（SAC 程序）	35
SAC 初始化进程	36
端口监视器服务管理 (pmadm)	36
ttymon 端口监视器	36
端口初始化进程	37
双向服务	37
TTY 监视器和网络侦听器端口监视器	37
TTY 端口监视器 (ttymon)	38
ttymon 和控制台端口	38
特定于 ttymon 的管理命令 (ttyadm)	38
网络侦听器服务 (listen)	39
特殊 listen 专用管理命令 (nlsadmin)	39
管理 ttymon 端口监视器	40
▼ 如何设置 ttymon 控制台终端类型	40
▼ 如何在 ttymon 控制台终端上设置波特率速度	40
▼ 如何添加 ttymon 端口监视器	41
▼ 如何查看 ttymon 端口监视器状态	42
▼ 如何停止 ttymon 端口监视器	43
▼ 如何启动 ttymon 端口监视器	43
▼ 如何禁用 ttymon 端口监视器	43
▼ 如何启用 ttymon 端口监视器	44
▼ 如何删除 ttymon 端口监视器	44
管理 ttymon 服务（任务列表）	44
管理 ttymon 服务	45
▼ 如何添加服务	45
▼ 如何查看 TTY 端口服务的状态	46
▼ 如何启用端口监视器服务	48
▼ 如何禁用端口监视器服务	48
服务访问工具管理（参考）	48
与 SAF 相关联的文件	48
/etc/saf/_sactab 文件	49
/etc/saf/pmtab/_pmtab 文件	49

服务状态	51
端口监视器状态	51
端口状态	51
4 管理系统资源（概述）	53
系统资源管理方面的新增功能	53
用于显示产品名称的新增 prtconf 选项	53
识别芯片多线程功能的 psrinfo 命令选项	54
新增 localeadm 命令	54
管理系统资源（指南）	54
5 显示和更改系统信息（任务）	57
显示系统信息（任务列表）	57
显示系统信息	58
▼如何显示系统的物理处理器类型	63
▼如何显示系统的逻辑处理器类型	64
▼如何显示系统中安装的语言环境	64
▼如何确定系统中是否安装了语言环境	65
更改系统信息（任务列表）	65
更改系统信息	66
▼如何手动设置系统的日期和时间	66
▼如何设置每日消息	67
▼如何更改系统的主机名	68
▼如何向系统中添加语言环境	69
▼如何从系统中删除语言环境	69
6 管理磁盘使用（任务）	71
管理磁盘使用（任务列表）	71
显示有关文件和磁盘空间的信息	72
▼如何显示有关文件和磁盘空间的信息	72
检查文件大小	75
▼如何显示文件大小	75
▼如何查找大文件	76
▼如何查找超过指定大小限制的文件	77

检查目录大小	78
▼ 如何显示目录、子目录和文件的大小	78
▼ 如何显示本地 UFS 文件系统的用户所有权	79
查找并删除旧文件或非活动文件	80
▼ 如何列出最新文件	80
▼ 如何查找并删除旧文件或非活动文件	81
▼ 如何清除临时目录	82
▼ 如何查找并删除 core 文件	82
▼ 如何删除故障转储文件	83
7 管理 UFS 配额（任务）	85
什么是 UFS 配额？	85
使用 UFS 配额	85
设置 UFS 配额的软限制和硬限制	86
磁盘块与文件限制之间的区别	86
设置 UFS 配额	86
设置 UFS 配额的原则	87
设置 UFS 配额（任务列表）	88
▼ 如何为 UFS 配额配置文件系统	88
▼ 如何设置一个用户的 UFS 配额	89
▼ 如何设置多个用户的 UFS 配额	90
▼ 如何检查 UFS 配额一致性	90
▼ 如何启用 UFS 配额	91
维护 UFS 配额（任务列表）	92
检查 UFS 配额	92
▼ 如何检查超过的 UFS 配额	92
▼ 如何检查文件系统中的 UFS 配额	93
更改和删除 UFS 配额	94
▼ 如何更改软限制缺省值	95
▼ 如何更改用户的 UFS 配额	96
▼ 如何禁用用户的 UFS 配额	97
▼ 如何禁用 UFS 配额	98
8 调度系统任务（任务）	99
创建和编辑 crontab 文件（任务列表）	99

自动执行系统任务的方法	100
用于调度重复性作业：crontab	100
用于调度单个作业：at	101
调度重复性系统任务(cron)	102
在 crontab 文件内	102
cron 守护进程处理调度的方法	103
crontab 文件项的语法	103
创建和编辑 crontab 文件	104
▼ 如何创建或编辑 crontab 文件	104
▼ 如何验证 crontab 文件是否存在	105
显示 crontab 文件	106
▼ 如何显示 crontab 文件	106
删除 crontab 文件	107
▼ 如何删除 crontab 文件	107
控制对 crontab 命令的访问	108
▼ 如何拒绝 crontab 命令访问	109
▼ 如何将 crontab 命令访问限制于指定的用户	109
如何验证受限的 crontab 命令访问	110
使用 at 命令（任务列表）	111
调度单个系统任务(at)	111
at 命令的说明	112
控制对 at 命令的访问	112
▼ 如何创建 at 作业	112
▼ 如何显示 at 队列	113
▼ 如何验证 at 作业	114
▼ 如何显示 at 作业	114
▼ 如何删除 at 作业	114
▼ 如何拒绝对 at 命令的访问	115
▼ 如何验证 at 命令访问已被拒绝	116
9 管理系统记帐（任务）	117
系统记帐方面的新增功能	117
Oracle Solaris 进程记帐和统计信息改进	117
什么是系统记帐？	118
系统记帐的工作原理	118

系统记帐组件	118
系统记帐（任务列表）	122
设置系统记帐	122
▼ 如何设置系统记帐	123
对用户计费	125
▼ 如何对用户计费	125
维护记帐信息	126
修复损坏的文件并更正 wtmpx 错误	126
▼ 如何修复损坏的 wtmpx 文件	126
更正 tacct 错误	126
▼ 如何修复 tacct 错误	127
重新启动 runacct 脚本	127
▼ 如何重新启动 runacct 脚本	128
停止和禁用系统记帐	128
▼ 如何暂时停止系统记帐	128
▼ 如何永久禁用系统记帐	129
 10 系统记帐（参考）	131
runacct 脚本	131
每日记帐报告	133
每日报告	134
每日使用情况报告	135
每日命令摘要	136
每月命令摘要	137
上次登录报告	137
使用 acctcom 检查 pacct 文件	137
系统记帐文件	139
runacct 脚本生成的文件	141
 11 管理系统性能（概述）	143
管理系统性能方面的新增功能	143
增强的 pfiles 工具	143
CPU 性能计数器	143
有关系统性能任务的参考信息	144
系统性能和系统资源	144

进程和系统性能	145
关于监视系统性能	146
监视工具	146
12 管理系统进程（任务）	149
系统进程管理方面的新增功能	149
伪内核进程	149
管理系统进程（任务列表）	150
用于管理系统进程的命令	150
使用 ps 命令	151
使用 /proc 文件系统和命令	152
使用进程命令管理进程 (/proc)	153
▼ 如何列出进程	154
▼ 如何显示有关进程的信息	155
▼ 如何控制进程	156
终止进程 (pkill, kill)	157
▼ 如何终止进程 (pkill)	157
▼ 如何终止进程 (kill)	158
调试进程 (pargs、preap)	158
管理进程类信息（任务列表）	159
管理进程类信息	160
更改进程的调度优先级 (priocntl)	160
▼ 如何显示有关进程类的基本信息 (priocntl)	161
▼ 如何显示进程的全局优先级	161
▼ 如何指定进程优先级 (priocntl)	162
▼ 如何更改分时进程的调度参数 (priocntl)	162
▼ 如何更改进程的类 (priocntl)	163
更改分时进程的优先级 (nice)	164
▼ 如何更改进程的优先级 (nice)	164
系统进程问题的故障排除	165
13 监视系统性能（任务）	167
显示系统性能信息（任务列表）	167
显示虚拟内存统计信息 (vmstat)	168
▼ 如何显示虚拟内存统计信息 (vmstat)	169

▼ 如何显示系统事件信息 (vmstat -s)	169
▼ 如何显示交换统计信息 (vmstat -S)	170
▼ 如何显示每台设备的中断次数 (vmstat -i)	170
显示磁盘使用率信息 (iostat)	171
▼ 如何显示磁盘使用率信息 (iostat)	171
▼ 如何显示扩展磁盘统计信息 (iostat -xtc)	172
显示磁盘空间统计信息 (df)	173
▼ 如何显示磁盘空间信息 (df -k)	173
监视系统活动 (任务列表)	174
监视系统活动 (sar)	175
▼ 如何检查文件访问 (sar -a)	176
▼ 如何检查缓冲区活动 (sar -b)	176
▼ 如何检查系统调用统计信息 (sar -c)	178
▼ 如何检查磁盘活动 (sar -d)	179
▼ 如何检查页出和内存 (sar -g)	180
检查内核内存分配	181
▼ 如何检查内核内存分配 (sar -k)	182
▼ 如何检查进程间通信 (sar -m)	183
▼ 如何检查页入活动 (sar -p)	184
▼ 如何检查队列活动 (sar -q)	185
▼ 如何检查未使用的内存 (sar -r)	186
▼ 如何检查 CPU 使用率 (sar -u)	187
▼ 如何检查系统表状态 (sar -v)	188
▼ 如何检查交换活动 (sar -w)	189
▼ 如何检查终端活动 (sar -y)	190
▼ 如何检查总体系统性能 (sar -A)	191
自动收集系统活动数据 (sar)	191
引导时运行 sadc 命令	191
使用 sa1 脚本定期运行 sadc 命令	192
使用 sa2 Shell 脚本生成报告	192
设置自动数据收集 (sar)	192
▼ 如何设置自动数据收集	194
 14 软件问题疑难解答 (概述)	195
疑难解答方面的新增内容	195

Common Agent Container 问题	195
x86: SMF 引导归档文件服务可能在系统重新引导期间失败	196
动态跟踪功能	196
kmdb 取代 kadb 作为标准的 Solaris 内核调试程序	196
有关软件故障排除任务的参考信息	197
系统崩溃故障排除	197
系统崩溃时应执行的操作	197
收集故障排除数据	198
系统崩溃故障排除核对表	199
15 管理系统消息	201
查看系统消息	201
▼ 如何查看系统消息	202
系统日志轮转	203
定制系统消息日志	204
▼ 如何定制系统消息日志	205
启用远程控制台消息传送	206
在运行级转换期间使用辅助控制台消息传递	206
在交互式登录会话期间使用 <code>consadm</code> 命令	207
▼ 如何启用辅助（远程）控制台	208
▼ 如何显示辅助控制台的列表	208
▼ 如何在系统重新引导期间启用辅助（远程）控制台	208
▼ 如何禁用辅助（远程）控制台	209
16 管理核心文件（任务）	211
管理核心文件（任务列表）	211
管理核心文件概述	211
配置核心文件路径	212
扩展的核心文件名	212
设置核心文件名称模式	213
启用 <code>setuid</code> 程序以生成核心文件	213
如何显示当前的核心转储配置	214
▼ 如何设置核心文件名称模式	214
▼ 如何启用每进程核心文件路径	214
▼ 如何启用全局核心文件路径	215

核心文件问题故障排除	215
检查核心文件	215
17 管理系统故障转储信息（任务）	217
管理系统故障转储信息中的新增内容	217
快速故障转储工具	217
管理系统故障转储信息（任务列表）	218
系统崩溃（概述）	218
交换区域和转储设备的 Oracle Solaris ZFS 支持	219
x86: GRUB 引导环境中的系统崩溃	219
系统故障转储文件	219
保存故障转储	219
dumpadm 命令	220
dumpadm 命令的工作原理	221
转储设备和卷管理器	221
管理系统故障转储信息	221
▼ 如何显示当前的故障转储配置	221
▼ 如何修改故障转储配置	222
▼ 如何检查故障转储	223
▼ 如何从完整的故障转储目录中恢复（可选的）	224
▼ 如何禁用或启用故障转储的保存	225
18 各种软件问题的故障排除（任务）	227
重新引导失败时应执行的操作	227
忘记 root 用户口令时应执行的操作	228
x86: SMF 引导归档文件服务在系统重新引导期间失败时应执行的操作	231
系统挂起时应执行的操作	232
文件系统占满时应执行的操作	232
由于创建了大文件或目录导致文件系统占满	233
由于系统内存不足导致 TMPFS 文件系统变满	233
复制或恢复后文件 ACL 丢失时应执行的操作	233
备份问题故障排除	233
备份文件系统后根 (/) 文件系统变满	233
确保备份和恢复命令相匹配	234
检查以确保当前目录正确	234

交互式命令	234
Oracle Solaris OS 中 Common Agent Container 问题的故障排除	235
端口号冲突	235
▼ 如何检查端口号	235
超级用户口令的安全性被破坏	236
▼ 如何为 Oracle Solaris OS 生成安全密钥	236
19 文件访问问题故障排除（任务）	237
解决搜索路径的问题 (Command not found)	237
▼ 如何诊断和更正搜索路径问题	238
解决文件访问问题	239
更改文件和组的所有权	239
识别网络访问问题	240
20 解决 UFS 文件系统不一致问题（任务）	241
fsck 错误消息	242
一般 fsck 错误消息	243
初始化阶段的 fsck 消息	245
阶段 1：检查块和大小消息	247
Oracle Solaris 10：阶段 1B：重新扫描更多 DUPS 消息	251
阶段 1B：重新扫描更多 DUPS 消息	252
阶段 2：检查路径名消息	252
阶段 3：检查连通性消息	258
阶段 4：检查引用计数消息	260
阶段 5：检查柱面组消息	263
阶段 5：检查柱面组消息	264
fsck 摘要消息	265
清除阶段消息	265
21 软件包问题故障排除（任务）	267
软件包符号链接问题故障排除	267
特定软件包安装错误	268
一般软件包安装问题	268

索引 269

前言

《系统管理指南：高级管理》是介绍 Oracle Solaris 系统管理信息重要内容的一套文档中的组成部分。该指南包含基于 SPARC 和基于 x86 的系统的信息。

本书假设您已经安装了 Oracle Solaris 操作系统 (operating system, OS)。同时假设您已经设置了任何计划使用的网络软件。

对于 Oracle Solaris 发行版，系统管理员感兴趣的新增功能已在相应各章的名为“... 的新增功能”的各节中加以介绍。

注 – 此 Oracle Solaris 发行版支持使用 SPARC 和 x86 系列处理器体系结构的系统。支持的系统可以在 Oracle Solaris OS: Hardware Compatibility Lists (Oracle Solaris OS: 硬件兼容性列表) 中找到。本文档列举了在不同类型的平台上进行实现时的所有差别。

在本文档中，这些与 x86 相关的术语表示以下含义：

- x86 泛指 64 位和 32 位的 x86 兼容产品系列。
- x64 特指 64 位的 x86 兼容 CPU。
- “32 位 x86”指出了有关基于 x86 的系统的特定 32 位信息。

有关支持的系统，请参见 [Oracle Solaris OS: Hardware Compatibility Lists](#) (Oracle Solaris OS: 硬件兼容性列表)。

目标读者

本书适用于负责管理运行 Oracle Solaris 操作系统 (operating system, OS) 的单一或多个系统的所有用户。要使用本书，您应当具备 1 到 2 年的 UNIX 系统管理经验。参加 UNIX 系统管理培训课程可能会对您有所帮助。

系统管理指南系列书籍的结构

下表列出了系统管理指南系列中各本书包含的主题。

书名	主题
《系统管理指南：基本管理》	用户帐户和组、服务器和客户机支持、关闭和启动系统、管理服务以及管理软件（软件包和修补程序）
《系统管理指南：高级管理》	终端和调制解调器、系统资源（磁盘配额、记帐和 crontab）、系统进程以及 Oracle Solaris 软件问题故障排除
《系统管理指南：设备和文件系统》	可移除介质、磁盘和设备、文件系统以及备份和还原数据
《系统管理指南：IP 服务》	TCP/IP 网络管理、IPv4 和 IPv6 地址管理、DHCP（动态主机配置协议）、Ipsec（Internet 协议安全）、IKE（Internet 密钥交换）、Solaris IP 过滤器、移动 IP、IP 网络多路径 (IP network multipathing, IPMP) 以及 IPQoS
《系统管理指南：名称和目录服务（DNS、NIS 和 LDAP）》	DNS、NIS 和 LDAP 命名和目录服务，包括从 NIS 转换到 LDAP 以及从 NIS+ 转换到 LDAP
《System Administration Guide: Naming and Directory Services (NIS+)》	NIS+ 命名和目录服务
《系统管理指南：网络服务》	Web 高速缓存服务器、与时间相关的服务、网络文件系统（NFS 和 Autofs）、邮件、SLP 和 PPP
《系统管理指南：打印》	打印主题和任务，使用服务、工具、协议和技术来设置及管理打印服务和打印机
《系统管理指南：安全性服务》	审计、设备管理、文件安全、BART（基本审计和报告工具）、Kerberos 服务、PAM（可插拔验证模块）、Solaris 加密框架、权限、RBAC（基于角色的存取控制）、SASL（简单身份认证和安全层）和 Solaris 安全 Shell
《系统管理指南：Oracle Solaris Containers－资源管理和 Oracle Solaris Zones》	资源管理主题项目和任务、扩展记帐、资源控制、公平份额调度器 (fair share scheduler, FSS)、使用资源上限设置守护进程 (rcapd) 的物理内存控制，以及资源池；使用 Solaris Zones 软件分区技术和 1x 标记区域的虚拟功能
《Oracle Solaris ZFS 管理指南》	ZFS（Zettabyte 文件系统）存储工具以及文件系统的创建和管理、快照、克隆、备份、使用访问控制列表 (Access Control List, ACL) 保护 ZFS 文件、在安装区域的 Oracle Solaris 系统中使用 ZFS、仿真卷以及故障排除和数据恢复
《Oracle Solaris Trusted Extensions 管理员规程》	特定于 Oracle Solaris Trusted Extensions 功能的系统管理
《Oracle Solaris Trusted Extensions 配置指南》	从 Solaris 10 5/08 发行版开始，介绍如何规划、启用及初始配置 Oracle Solaris Trusted Extensions 功能

相关的第三方 Web 站点引用

注 - 本软件或硬件以及文档可能提供了访问第三方内容、产品和服务的方式或有关这些内容、产品和服务的信息。对于第三方内容、产品和服务，Oracle Corporation 及其附属公司明确表示不承担任何种类的担保，亦不对其承担任何责任。对于因访问或使用第三方内容、产品或服务所造成的任何损失、成本或损害，Oracle Corporation 及其附属公司概不负责。

获取 Oracle 支持

Oracle 客户可以通过 My Oracle Support 获取电子支持。有关信息，请访问 <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info>，或访问 <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs>（如果您听力受损）。

印刷约定

下表介绍了本书中的印刷约定。

表 P-1 印刷约定

字体或符号	含义	示例
AaBbCc123	命令、文件和目录的名称；计算机屏幕输出	编辑 .login 文件。 使用 ls -a 列出所有文件。 machine_name% you have mail.
AaBbCc123	用户键入的内容，与计算机屏幕输出的显示不同	machine_name% su Password:
<i>aabbcc123</i>	要使用实名或值替换的命令行占位符	删除文件的命令为 <i>rm filename</i> 。
<i>AaBbCc123</i>	保留未译的新词或术语以及要强调的词	这些称为 <i>Class</i> 选项。 注意： 有些强调的项目在联机时以粗体显示。
新词术语强调	新词或术语以及要强调的词	高速缓存 是存储在本地的副本。 请勿保存文件。
《书名》	书名	阅读《用户指南》的第 6 章。

命令中的 shell 提示符示例

下表显示了 Oracle Solaris OS 中包含的缺省 UNIX shell 系统提示符和超级用户提示符。请注意，在命令示例中显示的缺省系统提示符可能会有所不同，具体取决于 Oracle Solaris 发行版。

表 P-2 shell 提示符

shell	提示符
Bash shell、Korn shell 和 Bourne shell	\$
Bash shell、Korn shell 和 Bourne shell 超级用户	#
C shell	machine_name%
C shell 超级用户	machine_name#

一般约定

请注意本书中使用的以下约定。

- 执行步骤或使用示例时，请确保完全按照括号中所示键入双引号 (")、左单引号 (') 和右单引号 (')。
- 在某些键盘上，Return 键被标记为 Enter。
- 已假设根路径包括 /sbin、/usr/sbin、/usr/bin 和 /etc 目录，因此，对于本书中的步骤，在显示这些目录中的命令时不带绝对路径名。对于那些使用其他不太常见目录中命令的步骤，在示例中会显示其绝对路径。

管理终端和调制解调器（概述）

本章提供管理终端和调制解调器的概述信息。

以下是本章中概述信息的列表：

- 第 19 页中的“管理终端和调制解调器方面的新增功能”
- 第 20 页中的“终端、调制解调器、端口和服务”
- 第 22 页中的“终端和调制解调器的管理工具”
- 第 23 页中的“串行端口工具”
- 第 23 页中的“服务访问工具概述”

有关如何使用串行端口工具设置终端和调制解调器的逐步说明，请参见第 2 章，[设置终端和调制解调器（任务）](#)。

有关如何使用服务访问工具 (Service Access Facility, SAF) 设置终端和调制解调器的逐步说明，请参见第 3 章，[使用服务访问工具管理串行端口（任务）](#)。

管理终端和调制解调器方面的新增功能

本节描述 Oracle Solaris 发行版中管理终端和调制解调器方面的新增或已更改的功能。有关新增功能的完整列表以及 Oracle Solaris 发行版的说明，请参见《[Oracle Solaris 10 8/11 新增功能](#)》。

SPARC: 相关控制台

Solaris 10 8/07：相关控制台子系统功能可实现部分内核控制台子系统，以便呈现控制台输出。相关控制台使用 Oracle Solaris 内核机制而不是可编程只读存储器 (Programmable Read-Only Memory, PROM) 接口来呈现控制台输出。这降低了控制台转译对 OpenBoot PROM (OBP) 的依赖性。相关控制台使用内核驻留帧缓冲区驱动程序生成控制台输出。生成的控制台输出比使用 OBP 转译的效率更高。相关控制台还避免了 SPARC 控制台输出过程中使 CPU 处于空闲状态，并且也改善了用户体验。

SPARC: 对如何设置控制台的 \$TERM 值的更改

Solaris 10 8 07：\$TERM 值现在是动态派生出的，具体取决于控制台所使用的终端仿真器。在基于 x86 的系统上，由于始终使用内核的终端仿真器，因此 \$TERM 值为 sun-color。

在基于 SPARC 的系统上，\$TERM 值如下：

sun-color 如果系统使用内核的终端仿真器，则 \$TERM 使用此值。

sun 如果系统使用 PROM 的终端仿真器，则 \$TERM 使用此值。

此更改不会影响为串行端口设置终端类型的方式。如以下示例所示，您仍然可以使用 svccfg 命令修改 \$TERM 值：

```
# svccfg
svc:> select system/console-login
svc:/system/console-login> setprop ttymon/terminal_type = "xterm"
svc:/system/console-login> exit
```

系统控制台上由 SMF 管理的 ttymon 调用

Oracle Solaris 10：系统控制台上的 ttymon 调用由 SMF 管理。通过将属性添加到 svc:/system/console-login:default 服务，可以使用 svccfg 命令指定 ttymon 命令参数。请注意，这些属性特定于 ttymon，不是通用的 SMF 属性。

注 - 您无法再在 /etc/inittab 文件中定义 ttymon 调用。

有关如何使用 SMF 指定 ttymon 命令参数的逐步说明，请参见第 40 页中的“[如何设置 ttymon 控制台终端类型](#)”。

有关 SMF 的完整概述，请参见《[系统管理指南：基本管理](#)》中的第 18 章“[管理服务（概述）](#)”。有关与 SMF 关联的逐步过程的信息，请参见《[系统管理指南：基本管理](#)》中的第 19 章“[管理服务（任务）](#)”。

终端、调制解调器、端口和服务

终端和调制解调器提供对系统和网络资源的本地和远程访问。设置终端和调制解调器访问是系统管理员的重要职责。本节解释 Oracle Solaris 操作系统中调制解调器和终端管理所涉及的一些概念。

终端说明

系统的位映射图形显示器与字母数字终端并不相同。字母数字终端连接到串行端口，并仅显示文本。您不必执行任何特殊步骤以管理图形显示器。

调制解调器描述

可以采用以下三种基本配置来设置调制解调器：

- 拨出
- 拨入
- 双向

连接到家庭计算机的调制解调器可以设置为提供**拨出**服务。通过拨出服务，您可以从自己的家里访问其他计算机。但是，任何人都无法从外部访问您的计算机。

拨入服务正好相反。通过拨入服务，用户可以从远程站点访问系统。但是，它不允许对外进行呼叫。

顾名思义，**双向**访问既提供拨入功能又提供拨出功能。

端口描述

端口是设备与操作系统进行通信的通道。从硬件的角度来看，端口是终端或调制解调器电缆可以用物理方式连入的“插口”。

然而，严格来讲，端口并不是物理容器，而是具有硬件（管脚和连接器）和软件（设备驱动程序）组件的实体。单个物理容器通常提供多个端口，允许连接两个或多个设备。

常见的端口类型包括：串行端口、并行端口、小型计算机系统接口 (Small Computer Systems Interface, SCSI) 端口和以太网端口。

串行端口使用标准的通信协议，通过一条线路逐位传输一个字节的信

息。按照 RS-232-C 或 RS-423 标准设计的设备包括大多数调制解调器、字母数字终端、绘图仪和一些打印机。可以使用标准电缆，将这些设备互连连接到具有类似设计的计算机的串行端口。

当必须将许多串行端口设备连接到单台计算机时，您可能需要为系统添加**适配器板**。适配器板及其驱动程序软件可提供额外串行端口，以连接更多设备。

服务描述

调制解调器和终端使用串行端口软件来访问计算资源。必须设置串行端口软件，以便为连接到端口的设备提供特定的“服务”。例如，您可以设置串行端口来为调制解调器提供双向服务。

端口监视器

获得对服务的访问的主要机制是通过**端口监视器**。端口监视器是持续监视对打印机或文件的登录和访问请求的程序。

当端口监视器检测到请求时，它将设置在操作系统和请求服务的设备之间建立通信所需的任何参数。然后，端口监视器将控制转移到提供所需服务的其他进程。

下表介绍了 Oracle Solaris 发行版中包括的两种类型的端口监视器。

表 1-1 端口监视器类型

手册页	端口监视器	说明
listen(1M)	listen	控制对网络服务的访问，例如在 Solaris 2.6 以前的发行版中处理远程打印请求。缺省的 Oracle Solaris OS 不再使用此端口监视器类型。
ttypmon(1M)	ttypmon	提供对调制解调器和字母数字终端所需的登录服务的访问。串行端口工具自动设置 ttypmon 端口监视器，以处理来自这些设备的登录请求。

您可能熟悉较早的端口监视器 `getty`。新增的 `ttypmon` 端口监视器功能更加强大。一个 `ttypmon` 端口监视器可以取代多个 `getty` 实例。在其他方面，这两个程序具有相同的功能。有关更多信息，请参见 [getty\(1M\)](#) 手册页。

终端和调制解调器的管理工具

下表列出了用于管理终端和调制解调器的工具。

表 1-2 用于管理终端和调制解调器的工具

终端和调制解调器的管理方式	工具	更多信息
最全面	服务访问工具 (Service Access Facility, SAF) 命令	第 23 页中的“服务访问工具概述”

表 1-2 用于管理终端和调制解调器的工具 (续)

终端和调制解调器的管理方式	工具	更多信息
设置最快	Solaris Management Console 的串行端口工具	第 2 章，设置终端和调制解调器 (任务) 和 Solaris Management Console 联机帮助

串行端口工具

串行端口工具使用相应的信息调用 `pmadm` 命令，将串行端口软件设置为使用终端和调制解调器。

该工具还提供以下内容：

- 用于终端和调制解调器常用配置的模板
- 多个端口的设置、修改或删除
- 每个端口的快速可视状态

服务访问工具概述

SAF 是用于管理终端、调制解调器和其他网络设备的工具。

特别是，借助 SAF 可以设置以下各项：

- `ttymon` 和 `listen` 端口监视器 (使用 `sacadm` 命令)
- `ttymon` 端口监视器服务 (使用 `pmadm` 和 `ttyadm` 命令)
- `listen` 端口监视器服务 (使用 `pmadm` 和 `nlsadmin` 命令)
- `tty` 设备故障排除
- 打印服务的传入网络请求故障排除
- 服务访问控制器故障排除 (使用 `sacadm` 命令)

SAF 是一个开放系统解决方案，可控制通过 `tty` 设备和局域网 (local-area network, LAN) 对系统和网络资源的访问。SAF 不是一个程序，而是一个后台进程和管理命令的分层结构。

设置终端和调制解调器（任务）

本章提供了使用 Solaris Management Console 的串行端口工具设置终端和调制解调器的逐步说明。

有关终端和调制解调器的概述信息，请参见第 1 章，[管理终端和调制解调器（概述）](#)。有关管理系统资源的概述信息，请参见第 4 章，[管理系统资源（概述）](#)。

有关使用 Solaris Management Console 的串行端口工具设置终端和调制解调器的关联过程的信息，请参见第 25 页中的[“设置终端和调制解调器（任务列表）”](#)。

设置终端和调制解调器（任务列表）

任务	说明	参考
设置终端。	通过使用 Solaris Management Console 的串行端口工具设置终端。通过从 "Action"（操作）菜单中选择相应的选项来配置终端。	第 29 页中的“如何设置终端”
设置调制解调器。	通过使用 Solaris Management Console 的串行端口工具设置调制解调器。通过从 "Action"（操作）菜单中选择相应的选项来配置调制解调器。	第 30 页中的“如何设置调制解调器”
初始化端口。	要初始化端口，请使用 Solaris Management Console 的串行端口工具。从 "Action"（操作）菜单中选择相应的选项。	第 31 页中的“如何初始化端口”

使用串行端口工具设置终端和调制解调器（概述）

您可以使用 Solaris Management Console 的串行端口工具设置串行端口。

从 "Serial Ports"（串行端口）窗口中选择一个串行端口，然后从 "Action"（操作）菜单中选择 "Configure"（配置）选项以配置以下各项：

- Terminal（终端）
- Modem – Dial-In（调制解调器 – 拨入）
- Modem – Dial-Out（调制解调器 – 拨出）
- Modem – Dial-In/Dial-Out（调制解调器 – 拨入/拨出）
- Initialize Only – No Connection（仅初始化 - 无连接）

通过 "Configure"（配置）选项可以访问用于配置这些服务的模板。您可以查看每个串行端口的两个级别的详细信息："Basic"（基本）和 "Advanced"（高级）。在通过选择串行端口再从 "Action"（操作）菜单中选择 "Properties"（属性）选项配置串行端口后，您可以访问每个串行端口的 "Advanced"（高级）级别的详细信息。在配置串行端口后，可以使用 SAF 命令禁用或启用该端口。有关使用 SAF 命令的信息，请参见第 3 章，使用服务访问工具管理串行端口（任务）。

有关使用串行端口命令行接口的信息，请参见 `smserialport(1M)` 手册页。

设置终端

下表介绍了使用串行端口工具设置终端时的各菜单项（及其缺省值）。

表 2-1 终端缺省值

详细信息	项目	缺省值
基本	Port（端口）	
	Description（描述）	Terminal（终端）
	Service Status（服务状态）	Enabled（已启用）
	Baud Rate（波特率）	9600
	Terminal Type（终端类型）	vi925
	Login Prompt（登录提示）	ttyn login:
高级	Carrier Detection（载体检测）	Software（软件）
	Option: Connect on Carrier（选项：在载体上连接）	Not available（不可用）
	Option: Bidirectional（选项：双向）	Available（可用）

表 2-1 终端缺省值（续）

详细信息	项目	缺省值
	Option: Initialize Only（选项：仅初始化）	Not available（不可用）
	Timeout (seconds)（超时（秒））	Never（从不）
	Port Monitor（端口监视器）	zsmon
	Service Program（服务程序）	/usr/bin/login

设置调制解调器

下表介绍了使用串行端口工具设置调制解调器时可用的三个调制解调器模板。

表 2-2 调制解调器模板

调制解调器配置	说明
Dial-In Only（仅拨入）	用户可以拨入调制解调器但无法拨出。
Dial-Out Only（仅拨出）	用户可以从调制解调器拨出但无法拨入。
Dial-In and Out (Bidirectional)（拨入和拨出（双向））	用户可以拨入调制解调器或从中拨出。

下表介绍了每个模板的缺省值。

表 2-3 调制解调器模板的缺省值

详细信息	项目	Modem - Dial-In Only（调制解调器 - 仅拨入）	Modem - Dial-Out Only（调制解调器 - 仅拨出）	Modem - Dial In and Out（调制解调器 - 拨入和拨出）
基本	Port Name（端口名）			
	Description（描述）	Modem – Dial In Only（调制解调器 – 仅拨入）	Modem – Dial Out Only（调制解调器 – 仅拨出）	Modem – Dial In and Out（调制解调器 – 拨入和拨出）
	Service Status（服务状态）	Enabled（已启用）	Enabled（已启用）	Enabled（已启用）
	Baud Rate（波特率）	9600	9600	9600
	Login Prompt（登录提示）	ttyn login:	ttyn login:	ttyn login:
高级	Carrier Detection（载体检测）	Software（软件）	Software（软件）	Software（软件）

表 2-3 调制解调器模板的缺省值（续）

详细信息	项目	Modem - Dial-In Only（调制解调器 - 仅拨入）	Modem - Dial-Out Only（调制解调器 - 仅拨出）	Modem - Dial In and Out（调制解调器 - 拨入和拨出）
	Option: Connect on Carrier（选项：在载体上连接）	Not available（不可用）	Not available（不可用）	Not available（不可用）
	Option: Bidirectional（选项：双向）	Not available（不可用）	Not available（不可用）	Available（可用）
	Option: Initialize Only（选项：仅初始化）	Not available（不可用）	Available（可用）	Not available（不可用）
	Timeout (seconds)（超时（秒））	Never（从不）	Never（从不）	Never（从不）
	Port Monitor（端口监视器）	zsmon	zsmon	zsmon
	Service Program（服务程序）	/usr/bin/login	/usr/bin/login	/usr/bin/login

下表介绍了 "Initialize Only"（仅初始化）模板的缺省值。

表 2-4 "Initialize Only - No Connection"（仅初始化 - 无连接）的缺省值

详细信息	项目	缺省值
基本	Port Name（端口名）	—
	Description（描述）	Initialize Only – No Connection（仅初始化 - 无连接）
	Service Status（服务状态）	Enabled（已启用）
	Baud Rate（波特率）	9600
	Login Prompt（登录提示）	ttyn login:
高级	Carrier Detection（载体检测）	Software（软件）
	Option: Connect on Carrier（选项：在载体上连接）	Not available（不可用）
	Option: Bidirectional（选项：双向）	Available（可用）
	Option: Initialize Only（选项：仅初始化）	Available（可用）
	Timeout (seconds)（超时（秒））	Never（从不）

表 2-4 "Initialize Only - No Connection"（仅初始化 - 无连接）的缺省值（续）

详细信息	项目	缺省值
	Port Monitor（端口监视器）	zsmon
	Service Program（服务程序）	/usr/bin/login

如何设置终端、调制解调器和初始化端口（任务）

▼ 如何设置终端

- 1 启动 Solaris Management Console（如果尚未运行）。
`% /usr/sadm/bin/smc &`
有关启动 Solaris Management Console 的信息，请参见《系统管理指南：基本管理》中的[“启动 Solaris Management Console”](#)。
- 2 单击“导航”窗格中的“本计算机”图标。
- 3 单击 "Devices and Hardware"（设备和硬件）—> "Serial Ports"（串行端口）。
将显示“串行端口”菜单。
- 4 选择将与终端一起使用的端口。
- 5 从 "Action"（操作）菜单中选择 "Configure"（配置）—> "Terminal"（终端）。
将在基本详细信息模式下显示“配置串行端口”窗口。
有关 "Terminal"（终端）菜单项的说明，请参见[表 2-1](#)。
- 6 单击 "OK"（确定）。
- 7 要配置高级项，请选择配置为终端的端口。
- 8 从 "Action"（操作）菜单中选择 "Properties"（属性）。
- 9 更改模板项的值（如果需要）。
- 10 单击 "OK"（确定）配置端口。
- 11 验证是否已添加了终端服务。
`$ pmadm -l -s ttyn`

▼ 如何设置调制解调器

- 1 启动 Solaris Management Console（如果尚未运行）。

```
% /usr/sadm/bin/smc &
```

有关启动 Solaris Management Console 的信息，请参见《系统管理指南：基本管理》中的[“启动 Solaris Management Console”](#)。

- 2 单击“导航”窗格中的“本计算机”图标。
- 3 单击“Devices and Hardware”（设备和硬件）—>“Serial Ports”（串行端口）。
将显示“串行端口”菜单。
- 4 选择将与调制解调器一起使用的端口。
- 5 从“Action”（操作）菜单中选择以下“Configure”（配置）选项之一。
 - a. “配置”—>“调制解调器（拨入）”
 - b. “Configure”（配置）—>“Modem (Dial Out)”（调制解调器（拨出））”
 - c. “Configure”（配置）—>“Modem (Dial In/Out)”（调制解调器（拨入/拨出））”

将在基本详细信息模式下显示“配置串行端口”窗口。

有关“Modem”（调制解调器）菜单项的说明，请参见[表 2-3](#)。
- 6 单击“OK”（确定）。
- 7 要配置高级项，请选择配置为调制解调器的端口。
- 8 从“Action”（操作）菜单中选择“Properties”（属性）。
- 9 更改模板项的值（如果需要）。
- 10 单击“OK”（确定）配置端口。
- 11 验证是否已配置了调制解调器服务。

```
$ pmadm -l -s tty
```

▼ 如何初始化端口

- 1 启动 Solaris Management Console (如果尚未运行)。

```
% /usr/sadm/bin/smc &
```

有关启动 Solaris Management Console 的信息，请参见《系统管理指南：基本管理》中的“启动 Solaris Management Console”。

- 2 单击“导航”窗格中的“本计算机”图标。
- 3 单击 "Devices and Hardware" (设备和硬件) —> "Serial Ports" (串行端口) 。
将显示“串行端口”菜单。
- 4 选择要初始化的端口。
- 5 选择 "Configure" (配置) —> "Initialize Only - No Connection" (仅初始化 - 无连接) 。
将在基本详细信息模式下显示“串行端口”窗口。
有关 "Initialize Only" (仅初始化) 菜单项的说明，请参见表 2-4。
- 6 单击 "OK" (确定) 。
- 7 要配置高级项，请选择配置为仅初始化的端口。然后，从“操作”菜单中选择“属性”。
- 8 更改模板项的值 (如果需要) 。
- 9 单击 "OK" (确定) 配置端口。
- 10 验证是否已初始化调制解调器服务。

```
$ pmadm -l -s tty
```

解决终端和调制解调器问题

如果在您添加终端或调制解调器并设置适当的服务后，用户无法通过串行端口线路登录，请考虑以下可能的失败原因：

- 与用户进行核对。

终端和调制解调器使用中的故障通常由无法登录或拨入的用户报告。因此，请通过检查桌面上是否存在问题来开始排除故障。

登录失败的一些常见原因包括：

- 登录 ID 或口令不正确
- 终端正在等待 X-ON 流控制键 (Ctrl-Q)

- 串行电缆连接不牢固或者已拔下
- 终端配置不正确
- 终端已关闭或者由于其他原因未通电

- 检查终端。

通过检查终端或调制解调器的配置继续排除故障。确定用于与终端或调制解调器进行通信的正确 *ttylabel*。验证终端或调制解调器设置是否与 *ttylabel* 设置匹配。

- 检查终端服务器。

如果终端没有问题，请继续在终端或调制解调器服务器上搜索问题的根源。使用 `pmadm` 命令验证是否已将一个端口监视器配置为向终端或调制解调器提供服务，以及该端口监视器是否具有与之关联的正确 *ttylabel*。例如：

```
$ pmadm -l -t ttymon
```

检查 `/etc/ttydefs` 文件，并根据终端配置仔细检查标签定义。使用 `sacadm` 命令检查端口监视器的状态。使用 `pmadm` 检查与终端所用的端口关联的服务。

- 检查串行连接。

如果服务访问控制器正在启动 TTY 端口监视器且以下条件为真：

- `pmadm` 命令报告已启用终端端口的服务。
- 终端的配置与端口监视器的配置匹配。

然后，通过检查串行连接继续搜索问题。串行连接由串行端口、电缆和终端组成。通过将一个部件与已知可靠的其他两个部件一起使用，对其中的每个部件进行测试。

测试以下所有部件：

- 串行端口
- 调制解调器
- 电缆
- 连接器
- 如果串行端口用作控制台，请不要使用串行端口工具修改串行端口设置。从 Oracle Solaris 10 开始，控制台上的 `ttymon` 调用由 SMF 进行管理。有关如何更改控制台终端类型的逐步说明，请参见第 40 页中的“如何设置 `ttymon` 控制台终端类型”。

有关 `ttymon` 和 SMF 的更多信息，请参见第 19 页中的“管理终端和调制解调器方面的新增功能”。

使用服务访问工具管理串行端口（任务）

本章描述如何使用服务访问工具 (Service Access Facility, SAF) 管理串行端口服务。

此外，本章还包括有关如何使用服务管理工具 (Service Management Facility, SMF) 执行控制台管理的信息。

注 – SAF 和 SMF 是 Oracle Solaris OS 中的两种不同工具。从 Oracle Solaris 10 开始，系统控制台上的 `ttymon` 调用由 SMF 进行管理。SAF 工具仍用于管理终端、调制解调器和其他网络设备。

以下是本章中概述信息的列表。

- [第 34 页中的“使用服务访问工具”](#)
- [第 35 页中的“整体 SAF 管理 \(sacadm\)”](#)
- [第 36 页中的“端口监视器服务管理 \(pmadm\)”](#)
- [第 37 页中的“TTY 监视器和网络侦听器端口监视器”](#)

有关与管理串行端口关联的逐步过程的信息，请参见以下内容：

- [第 34 页中的“管理串行端口（任务列表）”](#)
- [第 44 页中的“管理 `ttymon` 服务（任务列表）”](#)

有关 SAF 的参考信息，请参见第 48 页中的[“服务访问工具管理（参考）”](#)。

管理串行端口（任务列表）

任务	说明	参考
执行控制台管理。	您可能需要执行以下控制台管理任务： <ul style="list-style-type: none">■ 设置 <code>ttymon</code> 控制台终端类型。 从 Oracle Solaris 10 开始，您必须使用 <code>svccfg</code> 命令指定 <code>ttymon</code> 控制台终端类型。■ 设置 <code>ttymon</code> 控制台终端波特率速度。	第 40 页中的“如何设置 <code>ttymon</code> 控制台终端类型” 第 40 页中的“如何在 <code>ttymon</code> 控制台终端上设置波特率速度”
添加 <code>ttymon</code> 端口监视器。	使用 <code>sacadm</code> 命令添加 <code>ttymon</code> 端口监视器。	第 41 页中的“如何添加 <code>ttymon</code> 端口监视器”
查看 <code>ttymon</code> 端口监视器状态。	使用 <code>sacadm</code> 命令查看 <code>ttymon</code> 端口监视器状态。	第 42 页中的“如何查看 <code>ttymon</code> 端口监视器状态”
停止 <code>ttymon</code> 端口监视器。	使用 <code>sacadm</code> 命令停止 <code>ttymon</code> 端口监视器。	第 43 页中的“如何停止 <code>ttymon</code> 端口监视器”
启动 <code>ttymon</code> 端口监视器。	使用 <code>sacadm</code> 命令启动 <code>ttymon</code> 端口监视器。	第 43 页中的“如何启动 <code>ttymon</code> 端口监视器”
禁用 <code>ttymon</code> 端口监视器。	使用 <code>sacadm</code> 命令禁用 <code>ttymon</code> 端口监视器。	第 43 页中的“如何禁用 <code>ttymon</code> 端口监视器”
启用 <code>ttymon</code> 端口监视器。	使用 <code>sacadm</code> 命令启用 <code>ttymon</code> 端口监视器。	第 44 页中的“如何启用 <code>ttymon</code> 端口监视器”
删除 <code>ttymon</code> 端口监视器。	使用 <code>sacadm</code> 命令删除 <code>ttymon</code> 端口监视器。	第 44 页中的“如何删除 <code>ttymon</code> 端口监视器”

使用服务访问工具

您可以使用 Solaris Management Console 的串行端口工具或 SAF 命令设置终端和调制解调器。

SAF 是一种用于管理终端、调制解调器和其他网络设备的工具。顶层 SAF 程序是服务访问控制器 (Service Access Controller, SAC)。SAC 通过 `sacadm` 命令控制您管理的端口监视器。每个端口监视器可以管理一个或多个端口。

可以通过 `pmadm` 命令管理与端口关联的服务。尽管通过 SAC 提供的服务可能随网络的不同而不同，但是 SAC 及其管理命令 `sacadm` 和 `pmadm` 与网络无关。

下表介绍了 SAF 控制分层结构。sacadm 命令用于管理 SAC（它控制 ttymon 和 listen 端口监视器）。

ttymon 和 listen 的服务又是由 pmadm 命令控制的。一个 ttymon 实例可以为多个端口提供服务。一个 listen 实例可以在一个网络接口上提供多个服务。

表 3-1 SAF 控制分层结构

功能	程序	说明
整体管理	sacadm	用于添加和删除端口监视器的命令
服务访问控制器	sac	SAF 的主程序
端口监视器	ttymon	监视串行端口登录请求
	listen	监视网络服务请求
端口监视器服务管理员	pmadm	用于控制端口监视器服务的命令
服务	登录、远程过程调用	SAF 提供对其访问的服务
控制台管理	控制台登录	控制台服务由 SMF 服务 svc:/system/console-login:default 进行管理。此服务调用 ttymon 端口监视器。不要使用 pmadm 或 sacadm 命令管理控制台。有关更多信息，请参见第 38 页中的“ttymon 和控制台端口”、第 40 页中的“如何设置 ttymon 控制台终端类型”和第 40 页中的“如何在 ttymon 控制台终端上设置波特率速度”。

整体 SAF 管理 (sacadm)

sacadm 命令是 SAF 的顶层。sacadm 命令主要用于添加和删除端口监视器，如 ttymon 和 listen。sacadm 的其他功能包括列出端口监视器的当前状态和管理端口监视器配置脚本。

服务访问控制器（SAC 程序）

服务访问控制器 (Service Access Controller, SAC) 程序可监视所有端口监视器。在进入多用户模式时，系统自动启动 SAC。

SAC 程序在被调用时首先查找并解释每个系统的配置脚本。您可以使用配置脚本定制 SAC 程序环境。缺省情况下，此脚本为空。对 SAC 环境进行的修改由 SAC 的所有“子进程”继承。此继承的环境可能会被子进程修改。

SAC 程序在解释了每系统配置脚本后，将读取其管理文件并启动指定的端口监视器。对于每个端口监视器，SAC 程序运行它自己的副本，从而派生一个子进程。然后，每个子进程解释它的每端口监视器配置脚本（如果存在这样的脚本）。

对在每端口监视器配置脚本中指定的环境进行的任何修改都会影响端口监视器，并将由其所有子进程继承。最后，子进程通过使用在 SAC 程序管理文件中找到的命令运行端口监视器程序。

SAC 初始化进程

以下步骤概括说明了首次启动 SAC 时发生的情况：

1. SAC 程序由 SMF 服务 `svc:/system/sac:default` 启动。
2. SAC 程序读取每系统配置脚本 `/etc/saf/_sysconfig`。
3. SAC 程序读取 SAC 管理文件 `/etc/saf/_sactab`。
4. SAC 程序为它启动的每个端口监视器派生一个子进程。
5. 每个端口监视器读取每端口监视器配置脚本 `/etc/saf/pmtag/_config`。

端口监视器服务管理 (pmadm)

通过 `pmadm` 命令可以管理端口监视器的服务。特别是，使用 `pmadm` 命令可以添加或删除服务以及启用或禁用服务。您还可以安装或替换每服务配置脚本，或列显有关服务的信息。

服务的每个实例必须由端口监视器和端口唯一标识。使用 `pmadm` 命令管理服务时，可以用 `pmtag` 参数指定特定的端口监视器，用 `svctag` 参数指定特定的端口。

对于每种端口监视器类型，SAF 需要一个专用命令来设置端口监视器特定的配置数据的格式。此数据由 `pmadm` 命令使用。对于 `ttymon` 和 `listen` 类型的端口监视器，这些专用命令分别是 `ttyadm` 和 `nlsadmin`。

ttymon 端口监视器

只要您尝试通过使用直接连接的调制解调器或字母数字终端进行登录，`ttymon` 便会开始执行。首先，SMF 启动 SAC 进程。然后 SAC 自动启动在其管理文件 `/etc/saf/_sactab` 中指定的端口监视器。启动 `ttymon` 端口监视器后，该端口监视器对服务请求的串行端口线路进行监视。

当有人尝试通过使用字母数字终端或调制解调器进行登录时，串行端口驱动程序会将该活动传递给操作系统。`ttymon` 端口监视器记录串行端口活动，并尝试建立通信链路。`ttymon` 端口监视器确定与设备进行通信所需的数据传输率、线路规程和握手协议。

确定用于与调制解调器或终端通信的适当参数后，`ttymon` 端口监视器将这些参数传递给登录程序并将控制权转移给它。

端口初始化进程

当 SAC 调用 `ttymon` 端口监视器的实例时，`ttymon` 开始监视其端口。对于每个端口，`ttymon` 端口监视器首先初始化线路规程（如果指定）、速度和终端设置。用于初始化的值是从 `/etc/ttydefs` 文件中的相应项提取的。

然后 `ttymon` 端口监视器写入提示符，并等待用户输入。如果用户通过按 **Break** 键指示速度不合适，则 `ttymon` 端口监视器将尝试下一个速度，并再次写入提示符。

如果为端口启用了 *autobaud*，则 `ttymon` 端口监视器将尝试自动确定端口上的波特率。用户必须按回车键，`ttymon` 端口监视器才能识别波特率并列显提示符。

在收到有效输入时，`ttymon` 端口监视器会执行以下任务：

- 解释端口的每服务配置文件
- 创建 `/etc/utmpx` 项（如果需要）
- 建立服务环境
- 调用与端口关联的服务

服务终止后，`ttymon` 端口监视器将清除 `/etc/utmpx` 项（如果该项存在），并将端口恢复到其初始状态。

双向服务

如果将端口配置为用于双向服务，则 `ttymon` 端口监视器会执行以下操作：

- 允许用户连接到服务。
- 如果端口是空闲的，则允许 `uucico`、`cu` 或 `ct` 命令使用该端口进行拨出。
- 在列显提示符之前等待读取字符。
- 如果设置了“在载体上连接”标志，则在请求连接时，调用端口的关联服务而不发送提示信息。

TTY 监视器和网络侦听器端口监视器

虽然 SAF 提供了用于管理任何将来或第三方端口监视器的通用方法，但是在 Oracle Solaris 发行版中仅实现了两个端口监视器：`ttymon` 和 `listen`。

TTY 端口监视器 (ttymon)

ttymon 端口监视器基于 STREAMS，可执行以下操作：

- 监视端口
- 设置终端模式、波特率和线路规程
- 调用登录进程

ttymon 端口监视器为用户提供的服务与 getty 端口监视器在 SunOS 4.1 软件的早期版本中提供的服务相同。

ttymon 端口监视器在 SAC 程序下运行，并且是使用 `sacadm` 命令配置的。ttymon 的每个实例可以监视多个端口。这些端口在端口监视器的管理文件中指定。管理文件是通过使用 `pmadm` 和 `ttyadm` 命令配置的。

ttymon 和控制台端口

控制台服务既不由服务访问控制器 (Service Access Controller, SAC) 管理，也不由任何显式的 ttymon 管理文件管理。ttymon 调用由 SMF 管理。因此，无法再通过向 `/etc/inittab` 文件中添加项来调用 ttymon。类型为 `application`、名称为 `ttymon` 的属性组已添加到 SMF 服务 `svc:/system/console-login:default` 中。此属性组内的属性由方法脚本 `/lib/svc/method/console-login` 使用。此脚本将属性值用作 ttymon 调用的参数。通常，如果这些值为空，或者如果没有为任一属性定义值，则不会将值用于 ttymon。但是，如果 ttymon 设备值为空或者未设置，则 `/dev/console` 将用作缺省值以便 ttymon 可以运行。

以下属性在 SMF 服务 `svc:/system/console-login:default` 下可用：

<code>ttymon/nohangup</code>	指定 <code>nohangup</code> 属性。如果设置为 <code>true</code> ，则在设置缺省或指定的速度之前，不要通过将线路速度设置为零来强制线路挂起。
<code>ttymon/prompt</code>	指定控制台端口的提示字符串。
<code>ttymon/terminal_type</code>	指定控制台的缺省终端类型。
<code>ttymon/device</code>	指定控制台设备。
<code>ttymon/label</code>	指定 <code>/etc/ttydefs</code> 线路中的 TTY 标签。

特定于 ttymon 的管理命令 (ttyadm)

ttymon 管理文件由 `sacadm` 和 `pmadm` 命令以及 `ttyadm` 命令更新。`ttyadm` 命令设置特定于 ttymon 的信息的格式并将其写入标准输出，从而提供一种向 `sacadm` 和 `pmadm` 命令呈现已设置格式的、特定于 ttymon 的数据的方法。

因此，`ttyadm` 命令不会直接管理 `ttymon`。`ttyadm` 命令是对通用管理命令 `sacadm` 和 `pmadm` 的补充。有关更多信息，请参见 [ttyadm\(1M\)](#) 手册页。

网络侦听器服务 (listen)

`listen` 端口监视器在 SAC 下运行，可执行以下操作：

- 监视网络中的服务请求
- 在请求到达时接受请求
- 作为对那些服务请求的响应，调用服务器

`listen` 端口监视器是通过使用 `sacadm` 命令配置的。`listen` 的每个实例可以提供多个服务。这些服务在端口监视器的管理文件中指定。此管理文件是通过使用 `pmadm` 和 `nlsadmin` 命令配置的。

网络侦听器进程可以用于符合传输层接口 (Transport Layer Interface, TLI) 规范的任何面向连接的传输提供者。在 Oracle Solaris OS 中，`listen` 端口监视器可以提供 `inetd` 服务未提供的其他网络服务。

特殊 listen 专用管理命令 (nlsadmin)

`listen` 端口监视器的管理文件由 `sacadm` 和 `pmadm` 命令以及 `nlsadmin` 命令更新。`nlsadmin` 命令设置特定于 `listen` 的信息的格式并将其写入标准输出，从而提供一种向 `sacadm` 和 `pmadm` 命令呈现已设置格式的、特定于 `listen` 的数据的方法。

因此，`nlsadmin` 命令不会直接管理 `listen`。该命令是对通用管理命令 `sacadm` 和 `pmadm` 的补充。

单独配置的每个网络可以具有至少一个与之关联的网络侦听器进程实例。`nlsadmin` 命令控制 `listen` 端口监视器的操作状态。

`nlsadmin` 命令可以为给定的网络建立 `listen` 端口监视器，配置该端口监视器的特定属性以及对监视器执行 `start` 和 `kill` 操作。`nlsadmin` 命令还可以对计算机上的 `listen` 端口监视器进行报告。

有关更多信息，请参见 [nlsadmin\(1M\)](#) 手册页。

管理 ttymon 端口监视器

ttymon 的控制台管理现在由 SMF 管理。可使用 `svccfg` 命令设置 ttymon 系统控制台属性。仍可使用 SAF 命令 `sacadm` 添加、列出、中止、启动、启用、禁用和删除 ttymon 端口监视器。

▼ 如何设置 ttymon 控制台终端类型

此过程说明如何使用 `svccfg` 命令更改控制台终端类型。

- 1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的“配置 RBAC（任务列表）”。

- 2 运行 `svccfg` 命令以设置要更改的服务实例的属性。

```
# svccfg -s console-login setprop ttymon/terminal_type = "xterm"
```

其中，“xterm”是可能要使用的终端类型。

- 3 （可选的）重新启动服务实例。

```
# svcadm restart svc:/system/console-login:default
```



注意 – 如果您选择立即重新启动服务实例，则将从控制台注销。如果您不立即重新启动服务实例，则下次在控制台上显示登录提示时将应用属性更改。

▼ 如何在 ttymon 控制台终端上设置波特率速度

此过程说明如何在 ttymon 控制台终端上设置波特率速度。基于 x86 的系统上支持的控制台速度取决于具体的平台。

基于 SPARC 的系统上支持以下控制台速度：

- 9600 bps
- 19200 bps
- 38400 bps

- 1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的“配置 RBAC（任务列表）”。

2 使用 `eeeprom` 命令设置适合您的系统类型的波特率速度。

```
# eeeprom ttya-mode=baud-rate,8,n,1,-
```

例如，要将基于 x86 的系统控制台上的波特率更改为 38400，请键入：

```
# eeeprom ttya-mode=38400,8,n,1,-
```

3 按如下所示在 `/etc/ttydefs` 文件中更改控制台一行：

```
console baud-rate hupcl opost onlcr:baud-rate::console
```

4 根据您的系统类型，进行以下附加更改。

请注意，这些更改与平台相关。

- 在基于 SPARC 的系统上：在 `/kernel/drv/options.conf` 文件中更改波特率速度。

使用以下命令将波特率更改为 9600：

```
# 9600          :bd:
ttymodes="2502:1805:bd:8a3b:3:1c:7f:15:4:0:0:0:11:13:1a:19:12:f:17:16";
```

使用以下命令将波特率速度更改为 19200。

```
# 19200         :be:
ttymodes="2502:1805:be:8a3b:3:1c:7f:15:4:0:0:0:11:13:1a:19:12:f:17:16";
```

使用以下命令将波特率速度更改为 38400：

```
# 38400         :bf:
ttymodes="2502:1805:bf:8a3b:3:1c:7f:15:4:0:0:0:11:13:1a:19:12:f:17:16";
```

- 在基于 x86 的系统上：如果启用了 BIOS 串行重定向，请更改控制台速度。用来更改控制台速度的方法与平台有关。

▼ 如何添加 ttymon 端口监视器

1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《[系统管理指南：安全性服务](#)》中的“[配置 RBAC（任务列表）](#)”。

2 添加 ttymon 端口监视器。

```
# sacadm -a -p mbmon -t ttymon -c /usr/lib/saf/ttymon -v 'ttyadm
-V' -y "TTY Ports a & b"
```

- a 指定添加端口监视器的选项。
- p 将 `pmtagmbmon` 指定为端口监视器标记。
- t 将端口监视器 `type` 指定为 `ttymon`。
- c 定义用于启动端口监视器的 `command` 字符串。

- v 指定端口监视器的 *version* 号。
- y 定义描述此端口监视器实例的注释。

▼ 如何查看 ttymon 端口监视器状态

- 1 成为超级用户或承担等效角色。
角色包含授权和具有特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的“配置 RBAC（任务列表）”。
- 2 查看 ttymon 端口监视器的状态。

```
# sacadm -l -p mbmon
-l 指定 list 端口监视器的状态标志。
-p 将 pmtagmbmon 指定为端口监视器标记。
```

示例 3-1 查看 ttymon 端口监视器状态

此示例说明如何查看名为 mbmon 的端口监视器。

```
# sacadm -l -p mbmon
PMTAG PMTYPE FLGS RCNT STATUS  COMMAND
mbmon ttymon - 0 STARTING /usr/lib/saf/ttymon #TTY Ports a & b

PMTAG 标识端口监视器名称 mbmon。
PMTYPE 标识端口监视器类型 ttymon。
FLGS 指示是否设置了以下标志：
      d 不启用新的端口监视器。
      x 不启动新的端口监视器。
      破折号 (-) 未设置任何标志。
RCNT 指示返回计数值。返回计数为 0 指示如果端口监视器出现故障则不重新启动。
STATUS 指示端口监视器的当前状态。
COMMAND 标识用于启动端口监视器的命令。
#TTY Ports a & b 标识用于描述端口监视器的任何注释。
```

▼ 如何停止 ttymon 端口监视器

- 1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的“配置 RBAC（任务列表）”。

- 2 停止 ttymon 端口监视器。

```
# sacadm -k -p mbmon
```

-k 指定 *kill* 端口监视器的状态标志。

-p 将 *pmtagmbmon* 指定为端口监视器标记。

▼ 如何启动 ttymon 端口监视器

- 1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的“配置 RBAC（任务列表）”。

- 2 启动已中止的 ttymon 端口监视器。

```
# sacadm -s -p mbmon
```

-s 指定 *start* 端口监视器的状态标志。

-p 将 *pmtagmbmon* 指定为端口监视器标记。

▼ 如何禁用 ttymon 端口监视器

禁用端口监视器可防止新服务启动，且不影响现有的服务。

- 1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的“配置 RBAC（任务列表）”。

- 2 禁用 ttymon 端口监视器。

```
# sacadm -d -p mbmon
```

-d 指定 *disable* 端口监视器的状态标志。

-p 将 *pmtagmbmon* 指定为端口监视器标记。

▼ 如何启用 ttymon 端口监视器

通过启用 ttymon 端口监视器，可以使其为新请求提供服务。

- 1 成为超级用户或承担等效角色。
角色包含授权和具有特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的“配置 RBAC（任务列表）”。
- 2 启用 ttymon 端口监视器。

```
# sacadm -e -p mbmon
```

-e 指定 *enable* 端口监视器的状态标志。

-p 将 *pmtagmbmon* 指定为端口监视器标记。

▼ 如何删除 ttymon 端口监视器

删除端口监视器将删除与之关联的所有配置文件。

注 - 端口监视器配置文件不能通过使用 `sacadm` 命令进行更新或更改。要重新配置端口监视器，请先将其删除，然后添加一个新端口监视器。

- 1 成为超级用户或承担等效角色。
角色包含授权和具有特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的“配置 RBAC（任务列表）”。
- 2 删除 ttymon 端口监视器。

```
# sacadm -r -p mbmon
```

-r 指定 *remove* 端口监视器的状态标志。

-p 将 *pmtagmbmon* 指定为端口监视器标记。

管理 ttymon 服务（任务列表）

任务	说明	参考
添加 ttymon 服务。	使用 <code>pmadm</code> 命令添加服务。	第 45 页中的“如何添加服务”
查看 TTY 端口服务的状态。	使用 <code>pmadm</code> 命令查看 TTY 端口的状态。	第 46 页中的“如何查看 TTY 端口的状态”

任务	说明	参考
启用端口监视器服务。	使用带有 <code>-e</code> 选项的 <code>pmadm</code> 命令启用端口监视器。	第 48 页中的“如何启用端口监视器服务”
禁用端口监视器服务。	使用带有 <code>-d</code> 选项的 <code>pmadm</code> 命令禁用端口监视器。	第 48 页中的“如何禁用端口监视器服务”

管理 ttymon 服务

使用 `pmadm` 命令可以添加服务、列出与端口监视器关联的一个或多个端口的服务以及启用或禁用服务。

▼ 如何添加服务

- 1 成为超级用户或承担等效角色。
角色包含授权和具有特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的“配置 RBAC（任务列表）”。
- 2 将标准终端服务添加到 `mbmon` 端口监视器。

```
# pmadm -a -p mbmon -s a -i root -v 'ttyadm -V' -m "'ttyadm -i 'Terminal
disabled' -l contty -m ldterm,ttcompat -S y -d /dev/term/a
-s /usr/bin/login'"
```

注 - 在此示例中，输入内容自动换到下一行。请勿使用回车键或换行。

- a 指定 *add* 端口监视器状态标志。
- p 将 *pmtag* `mbmon` 指定为端口监视器标记。
- s 将 *svctag* `a` 指定为端口监视器 *service* 标记。
- i 指定在服务运行时要分配给 *svctag* 的 *identity*。
- v 指定端口监视器的 *version* 号。
- m 指定由 `ttyadm` 设置其格式的特定于 `ttymon` 的配置数据。

前面的 `pmadm` 命令包含嵌入的 `ttyadm` 命令。此嵌入命令中的选项如下所示：

- b 指定 *bidirectional* 端口标志。
- i 指定 *inactive*（禁用）响应消息。
- l 指定要使用 `/etc/ttydefs` 文件中的哪个 TTY *label*。
- m 指定在调用此服务之前要推送的 *STREAMS modules*。

- d 指定要用于 TTY 端口的 *device* 的全路径名。
- s 指定收到连接请求时要调用的 *service* 的全路径名。如果需要参数，请将命令及其参数用引号 (") 括起来。

▼ 如何查看 TTY 端口服务的状态

使用此过程中所示的 `pmadm` 命令可以列出某个 TTY 端口的状态，或与端口监视器关联的所有端口的状态。

- 1 成为超级用户或承担等效角色。
角色包含授权和具有特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的“配置 RBAC（任务列表）”。
- 2 列出端口监视器的一项服务。

```
# pmadm -l -p mbmon -s a
```

 - l 列出有关系统的服务信息。
 - p 将 `pmtagmbmon` 指定为端口监视器标记。
 - s 将 `svctag a` 指定为端口监视器 *service* 标记。

示例 3-2 查看 TTY 端口监视器服务的状态

此示例列出了端口监视器的所有服务。

```
# pmadm -l -p mbmon
PMTAG PMTYPE SVCTAG FLGS ID <PMSPECIFIC>
mbmon ttymon a - root /dev/term/a - - /usr/bin/login - contty
ldterm,ttcompat login: Terminal disabled tvi925 y #
```

PMTAG	标识通过使用 <code>pmadm -p</code> 命令设置的端口监视器名称 <code>mbmon</code> 。
PMTYPE	标识端口监视器类型 <code>ttymon</code> 。
SVCTAG	指示通过使用 <code>pmadm -s</code> 命令设置的服务标签值。
FLAGS	标识是否通过使用 <code>pmadm -f</code> 命令设置了以下标志。 <ul style="list-style-type: none">■ <code>x</code>—不启用服务。■ <code>u</code>—为服务创建 <code>utmpx</code> 项。■ 破折号 (-) —未设置标志。
ID	指示启动服务时为其指定的标识。该值是通过使用 <code>pmadm -i</code> 命令设置的。
<PMSPECIFIC>	信息

<code>/dev/term/a</code>	指示通过使用 <code>ttyadm -d</code> 命令设置的 TTY 端口路径名。
<code>-</code>	指示是否通过使用 <code>ttyadm -c -b -h -I -r</code> 命令设置了以下标志。 <ul style="list-style-type: none"> ■ <code>c</code>—为端口设置“在载体上连接”标志。 ■ <code>b</code>—将端口设置为双向，从而允许传入和传出通信。 ■ <code>h</code>—禁止在收到传入调用之后立即自动挂起。 ■ <code>I</code>—初始化端口。 ■ <code>r</code>—强制 <code>ttymon</code> 在输出 <code>login:</code> 消息之前一直等待，直到它收到来自端口的字符。 ■ 破折号 (<code>-</code>)—未设置标志。
<code>-</code>	指示通过使用 <code>ttyadm -r count</code> 选项设置的值。此选项确定在收到来自端口的数据之后 <code>ttymon</code> 何时显示提示符。如果 <code>count</code> 为 0，则 <code>ttymon</code> 将一直等待，直到它收到某个字符。如果 <code>count</code> 大于 0，则 <code>ttymon</code> 将一直等待，直到收到 <code>count</code> 个换行符。在此示例中未设置值。
<code>/usr/bin/login</code>	标识在收到连接时要调用的服务的全路径名。该值是通过使用 <code>ttyadm -s</code> 命令设置的。
<code>-</code>	标识 <code>ttyadm -t</code> 命令的超时值。此选项指定：如果端口成功打开，且在 <code>timeout</code> 秒内未收到输入数据，则 <code>ttymon</code> 应该关闭该端口。在此示例中没有超时值。
<code>contty</code>	标识 <code>/etc/ttydefs</code> 文件中的 TTY 标签。该值是通过使用 <code>ttyadm -l</code> 命令设置的。
<code>ldterm,ttcompat</code>	标识要推送的 STREAMS 模块。这些模块是通过使用 <code>ttyadmin -m</code> 命令设置的。
<code>login: Terminal disabled</code>	标识在禁用端口时要显示的无效消息。此消息是通过使用 <code>ttyadm -i</code> 命令设置的。
<code>tvi925</code>	标识终端类型（如果使用 <code>ttyadm -T</code> 命令进行了设置）。在此示例中，终端类型为 <code>tvi925</code> 。
<code>y</code>	标识通过使用 <code>ttyadm -S</code> 命令设置的软件载体值。 <code>n</code> 关闭软件载体。 <code>y</code> 打开软件载体。在此示例中打开了软件载体。
<code>#</code>	标识使用 <code>pmadm -y</code> 命令指定的任何注释。在此示例中没有注释。

▼ 如何启用端口监视器服务

- 1 成为超级用户或承担等效角色。
角色包含授权和具有特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的“配置 RBAC（任务列表）”。
- 2 启用已禁用的端口监视器服务。

```
# pmadm -e -p mbmon -s a
```

 - e 指定 *enable* 标志。
 - p 将 *pmtagmbmon* 指定为端口监视器标记。
 - s 将 *svctag a* 指定为端口监视器 *service* 标记。

▼ 如何禁用端口监视器服务

- 1 成为超级用户或承担等效角色。
角色包含授权和具有特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的“配置 RBAC（任务列表）”。
- 2 禁用端口监视器服务。

```
# pmadm -d -p mbmon -s a
```

 - d 指定 *disable* 标志。
 - p 将 *pmtagmbmon* 指定为端口监视器标记。
 - s 将 *svctag a* 指定为端口监视器 *service* 标记。

服务访问工具管理（参考）

本节包括服务访问工具管理的参考信息。

与 SAF 相关联的文件

SAF 使用可通过 `sacadm` 和 `pmadm` 命令修改的配置文件。您无需手动编辑配置文件。

文件名	说明
/etc/saf/_sysconfig	每系统配置脚本

文件名	说明
/etc/saf/_sactab	SAC 的管理文件，该文件包含 SAC 控制的端口监视器的配置数据。
/etc/saf/pmtag	端口监视器 <i>pmtag</i> 的起始目录
/etc/saf/pmtag/_config	端口监视器 <i>pmtag</i> 的每端口监视器配置脚本（如果存在）
/etc/saf/pmtag/_pmtab	端口监视器 <i>pmtag</i> 的管理文件，该文件包含 <i>pmtag</i> 所提供服务的特定于端口监视器的配置数据
/etc/saf/pmtag/svctag	服务 <i>svctag</i> 的每服务配置脚本
/var/saf/log	SAC 的日志文件
/var/saf/pmtag	<i>pmtag</i> 创建的文件（例如日志文件）的目录

/etc/saf/_sactab 文件

/etc/saf/_sactab 文件中的信息如下所示：

# VERSION=1	
zsmon:ttymon::0:/usr/lib/saf/ttymon	
#	
# VERSION=1	指示服务访问工具的版本号。
zsmon	端口监视器的名称。
ttymon	端口监视器的类型。
::	指示是否设置了以下两个标志：
d	不启用端口监视器。
x	不启动端口监视器。在此示例中未设置标志。
0	指示返回代码值。返回计数为 0 指示如果端口监视器出现故障则不重新启动。
/usr/lib/saf/ttymon	指示端口监视器路径名。

/etc/saf/pmtab/_pmtab 文件

/etc/saf/pmtab/_pmtab 文件（如 /etc/saf/zsmon/_pmtab）与如下所示内容类似：

# VERSION=1	
ttya:u:root:reserved:reserved:reserved:/dev/term/a:I::/usr/bin/login::9600: ldterm,ttcompat:ttya login\: ::tvi925:y:#	
# VERSION=1	指示服务访问工具的版本号。
ttya	指示服务标签。
x,u	标识是否设置了以下标志： x 不启用服务。 u 为服务创建 utmpx 项。
root	指示为服务标签指定的标识。
reserved	保留此字段供将来使用。
reserved	保留此字段供将来使用。
reserved	保留此字段供将来使用。
/dev/term/a	指示 TTY 端口路径名。
/usr/bin/login	标识在收到连接时要调用的服务的全路径名。
:c,b,h,I,r:	指示是否设置了以下标志： c 为端口设置“在载体上连接”标志。 b 将端口设置为双向，从而允许传入和传出通信。 h 禁止在收到外来调用之后立即自动挂起。 I 初始化端口。 r 强制 ttymon 在 ttymon 列显 login: 消息之前一直等待，直到它收到来自端口的字符。
9600	标识在 /etc/ttydefs 文件中定义的 TTY 标签。
ldterm,ttcompat	标识要推送的 STREAMS 模块。
ttya login\:	标识要显示的提示符。
:y/n:	指示响应或不响应。
message	标识任何无效（禁用）的响应消息。
tvi925	标识终端类型。
y	指示是否设置了软件载体 (y/n)。

服务状态

sacadm 命令可控制服务的状态。以下列表介绍了服务的可能状态。

Enabled 缺省状态。在添加端口监视器时，服务将运行。

Disabled 缺省状态。在删除端口监视器时，服务将停止。

要确定任何特定服务的状态，请使用以下命令：

```
# pmadm -l -p portmon-name -ssvctag
```

端口监视器状态

sacadm 命令可控制 ttymon 和 listen 端口监视器的状态。下表介绍了可能的端口监视器状态。

状态	说明
Started	缺省状态－在添加端口监视器时，它将自动启动。
Enabled	缺省状态－在添加端口监视器时，它将自动准备接受服务请求。
Stopped	缺省状态－在删除端口监视器时，它将自动停止。
Disabled	缺省状态－在删除端口监视器时，它将自动继续使用现有服务并拒绝添加新服务。
Starting	中间状态－端口监视器处于启动过程中。
Stopping	中间状态－端口监视器已手动终止，但是它尚未完成其关闭过程。端口监视器正在停止。
Notrunning	非活动状态－端口监视器已中止。以前监视的所有端口都是无法访问的。外部用户无法断定端口处于 disabled 还是 notrunning 状态。
Failed	非活动状态－端口监视器无法启动和保持运行。

要确定任何特定端口监视器的状态，请使用以下命令：

```
# sacadm -l -p portmon-name
```

端口状态

根据控制端口的端口监视器的状态，可以启用或禁用端口。

状态	说明
串行 (ttymon) 端口状态	
Enabled	ttymon 端口监视器向端口发送提示消息，并为其提供登录服务。
Disabled	如果 ttymon 已中止或禁用，则为所有端口的缺省状态。如果指定此状态，则 ttymon 在收到连接请求时将发出 disabled 消息。

管理系统资源（概述）

本章简要介绍了 Oracle Solaris OS 中可用的系统资源管理功能，并提供了用于管理系统资源的指南。

使用这些功能，可以显示常规系统信息、监视磁盘空间、设置磁盘配额并使用记帐程序。此外，还可以调度 `cron` 和 `at` 命令以自动运行常规命令。

本节中不包括可用来灵活分配、监视和控制系统资源的资源管理的有关信息。

有关与管理系统资源（不通过资源管理）相关的过程的信息，请参见第 54 页中的“管理系统资源（指南）”。

有关通过资源管理来管理系统资源的信息，请参见《系统管理指南：Oracle Solaris Containers—资源管理和 Oracle Solaris Zones》中的第 1 章“Solaris 10 资源管理介绍”。

系统资源管理方面的新增功能

本节介绍此 Oracle Solaris 发行版中用于管理系统资源的新增功能或已更改的功能。有关新增功能的完整列表以及 Oracle Solaris 发行版的说明，请参见《Oracle Solaris 10 8/11 新增功能》。

用于显示产品名称的新增 `prtconf` 选项

Solaris 10 1/06：`prtconf` 命令中添加了一个新的 `-b` 选项，用于显示系统的产品名称。此选项与 `uname -i` 命令类似。不过，`prtconf -b` 命令专门用于确定产品的市场营销名称。

使用 `prtconf` 命令的 `-b` 选项显示的固件设备树根属性如下所示：

- name
- compatible

- banner-name
- model

要显示可能的其他平台特定输出，请使用 `prtconf -vb` 命令。有关更多信息，请参见 [prtconf\(1M\)](#) 手册页和 [第 62 页](#) 中的“如何显示系统的产品名称”。

识别芯片多线程功能的 `psrinfo` 命令选项

Oracle Solaris 10： `psrinfo` 命令已经过修改，不仅可以提供有关虚拟处理器的信息，还可以提供有关物理处理器的信息。添加此增强功能是为了识别芯片多线程 (chip multithreading, CMT) 功能。新增的 `-p` 选项可报告系统中物理处理器的总数。使用 `psrinfo -pv` 命令可列出系统中所有的物理处理器，以及与每个物理处理器关联的虚拟处理器。 `psrinfo` 命令的缺省输出仍然可显示系统的虚拟处理器信息。

有关更多信息，请参见 [psrinfo\(1M\)](#) 手册页。

有关与此功能关联的过程的信息，请参见 [第 63 页](#) 中的“如何显示系统的物理处理器类型”。

新增 `localeadm` 命令

Oracle Solaris 10：通过新增的 `localeadm` 命令可以更改系统中的语言环境，而无需重新安装 OS 或手动添加和删除软件包。使用此命令还可以查询系统，从而确定已安装的语言环境。要运行 `localeadm` 命令，必须具有超级用户特权或通过基于角色的访问控制 (role-based access control, RBAC) 承担等效角色。

有关更多信息，请参见 [localeadm\(1M\)](#) 手册页。

有关本指南中该命令的更多信息，请参见 [第 5 章](#)，[显示和更改系统信息（任务）](#)。

有关新增功能的完整列表以及 Oracle Solaris 发行版的说明，请参见 [《Oracle Solaris 10 8/11 新增功能》](#)。

管理系统资源（指南）

任务	说明	参考
显示并更改系统信息	使用各种命令显示并更改系统信息，如常规系统信息、语言环境、日期和时间以及系统的主机名。	第 5 章 ， 显示和更改系统信息（任务）
管理磁盘使用	确定磁盘空间的使用情况并采取措施删除旧文件和未使用的文件。	第 6 章 ， 管理磁盘使用（任务）

任务	说明	参考
管理配额	使用 UFS 文件系统配额管理用户使用的磁盘空间量。	第 7 章，管理 UFS 配额（任务）
调度系统事件	使用 cron 和 at 作业帮助调度系统例程，其中可以包括清除旧文件和未使用的文件。	第 8 章，调度系统任务（任务）
管理系统记帐	使用系统记帐来确定用户和应用程序使用系统资源的情况。	第 9 章，管理系统记帐（任务）
管理系统资源（通过 Solaris 资源管理）	使用资源管理器控制应用程序使用可用系统资源的方式，跟踪资源使用情况并对其进行收费。	《系统管理指南：Oracle Solaris Containers—资源管理和 Oracle Solaris Zones》中的第 1 章“Solaris 10 资源管理介绍”

显示和更改系统信息（任务）

本章介绍显示和更改最常用的系统信息所需执行的任务。

有关与显示和更改系统信息相关的过程的信息，请参见以下各节：

- 第 57 页中的“显示系统信息（任务列表）”
- 第 65 页中的“更改系统信息（任务列表）”

有关管理系统资源的概述信息，请参见第 4 章，管理系统资源（概述）。

显示系统信息（任务列表）

任务	说明	参考
确定系统启用了 32 位还是 64 位功能。	使用 <code>isainfo</code> 命令可确定系统启用了 32 位还是 64 位功能。在基于 x86 的系统中，可以使用 <code>isalist</code> 命令来显示此信息。	第 59 页中的“如何确定系统启用了 32 位还是 64 位功能”
显示 Oracle Solaris 发行版信息。	显示 <code>/etc/release</code> 文件的内容，以确定 Oracle Solaris 发行版的版本。	第 61 页中的“如何显示 Oracle Solaris 发行版信息”
显示常规系统信息。	使用 <code>showrev</code> 命令显示常规系统信息。	第 61 页中的“如何显示常规系统信息”
显示系统的主机 ID 号。	使用 <code>hostid</code> 命令显示系统的主机 ID。	第 62 页中的“如何显示系统的主机 ID 号”
显示系统的产品名称。	从 Solaris 10 1/06 发行版开始，可以使用 <code>prtconf -b</code> 命令显示系统的产品名称。	第 62 页中的“如何显示系统的产品名称”

任务	说明	参考
显示系统的已安装内存。	使用 <code>prtconf</code> 命令显示有关系统的已安装内存的信息。	第 63 页中的“如何显示系统的已安装内存”
显示系统的日期和时间。	使用 <code>date</code> 命令显示系统的日期和时间。	第 63 页中的“如何显示日期和时间”
显示系统的物理处理器类型。	使用 <code>psrinfo -p</code> 命令列出系统中物理处理器的总数。 使用 <code>psrinfo -pv</code> 命令列出系统中的所有物理处理器以及与每个物理处理器关联的虚拟处理器。	第 63 页中的“如何显示系统的物理处理器类型”
显示系统的逻辑处理器类型。	使用 <code>psrinfo -v</code> 命令显示系统的逻辑处理器类型。	第 64 页中的“如何显示系统的逻辑处理器类型”
显示系统中安装的语言环境。	使用 <code>localeadm</code> 命令显示系统中安装的语言环境。	第 64 页中的“如何显示系统中安装的语言环境”
确定系统中是否安装了语言环境。	使用 <code>localeadm</code> 命令的 <code>-q</code> 选项和语言环境来确定系统中是否安装了语言环境。	第 65 页中的“如何确定系统中是否安装了语言环境”

显示系统信息

下表介绍了可用于显示常规系统信息的命令。

表 5-1 用于显示系统信息的命令

命令	显示的系统信息	手册页
<code>date</code>	日期和时间	date(1)
<code>hostid</code>	主机 ID 号	hostid(1)
<code>isainfo</code>	正在运行的系统中 本机 应用程序所支持的位数，该位数可以作为标记传递给脚本	isainfo(1)
<code>isalist</code>	用于基于 x86 的系统的处理器类型	psrinfo(1M)
<code>localeadm</code>	系统中安装的语言环境	localeadm(1M)
<code>prtconf</code>	系统配置信息，已安装的内存和产品名称	prtconf(1M)
<code>psrinfo</code>	处理器类型	psrinfo(1M)
<code>showrev</code>	主机名、主机 ID、发行版、内核体系结构、应用程序体系结构、硬件提供者、域和内核版本	showrev(1M)

表 5-1 用于显示系统信息的命令（续）

命令	显示的系统信息	手册页
uname	操作系统名称、发行版、版本、节点名称、硬件名称和处理器类型	uname(1)

▼ 如何确定系统启用了 32 位还是 64 位功能

- 使用 **isainfo** 命令可确定系统启用了 32 位还是 64 位功能。

isainfo options

当运行未指定任何选项的 **isainfo** 命令时，将显示当前 OS 版本所支持应用程序的本机指令集的一个或多个名称。

- v 列显有关其他选项的详细信息。
- b 列显本机指令集的地址空间中的位数。
- n 列显当前版本的 OS 支持的可移植应用程序所使用的本机指令集的名称。
- k 列显 OS 内核组件（例如设备驱动程序和 STREAMS 模块）所使用的一个或多个指令集的名称。

注 – 在基于 x86 的系统中，也可以使用 **isalist** 命令来显示此信息。

有关更多信息，请参见 [isalist\(1\)](#) 手册页。

示例 5-1 SPARC: 确定系统启用了 32 位还是 64 位功能

在运行以前的 Oracle Solaris OS 发行版（使用 32 位内核）的 UltraSPARC 系统中，**isainfo** 命令的输出如下所示：

```
$ isainfo -v
32-bit sparc applications
```

此输出表明，该系统只能支持 32 位应用程序。

在当前的 Oracle Solaris OS 发行版中，只有基于 SPARC 的系统才提供 64 位内核。在运行 64 位内核的 UltraSPARC 系统中，**isainfo** 命令的输出如下所示：

```
$ isainfo -v
64-bit sparcv9 applications
32-bit sparc applications
```

此输出表明，该系统既可以支持 32 位应用程序，又可以支持 64 位应用程序。

使用 **isainfo -b** 命令可以显示正在运行的系统中的本机应用程序所支持的位数。

在运行 32 位 Oracle Solaris OS 的基于 SPARC 的系统、基于 x86 的系统或 UltraSPARC 系统中，`isainfo` 命令的输出如下所示：

```
$ isainfo -b
32
```

在运行 64 位 Oracle Solaris OS 的 64 位 UltraSPARC 系统中，`isainfo` 命令的输出如下所示：

```
$ isainfo -b
64
```

该命令只返回 64。即使 64 位 UltraSPARC 系统可以运行两种类型的应用程序，64 位应用程序也是最适合在 64 位系统中运行的应用程序。

示例 5-2 x86: 确定系统启用了 32 位还是 64 位功能

在运行 64 位内核的基于 x86 的系统中，`isainfo` 命令输出如下所示：

```
$ isainfo
amd64 i386
```

此输出表明，该系统可以支持 64 位应用程序。

使用 `isainfo -v` 命令可以确定基于 x86 的系统是否能够运行 32 位内核。

```
$ isainfo -v
64-bit amd64 applications
    fpu tsc cx8 cmov mmx ammx a3dnow a3dnowx fxsr sse sse2
32-bit i386 applications
    fpu tsc cx8 cmov mmx ammx a3dnow a3dnowx fxsr sse sse2
```

此输出表明，该系统既可以支持 64 位应用程序，又可以支持 32 位应用程序。

使用 `isainfo -b` 命令可以显示正在运行的系统中的本机应用程序所支持的位数。

在运行 32 位 Oracle Solaris OS 的基于 x86 的系统中，`isainfo` 命令的输出如下所示：

```
$ isainfo -b
32
```

在运行 64 位 Oracle Solaris OS 的基于 x86 的系统中，`isainfo` 命令的输出如下所示：

```
$ isainfo -b
64
```

您也可以使用 `isalist` 命令，来确定基于 x86 的系统是以 32 位模式还是以 64 位模式运行。

```
$ isalist
amd64 pentium_pro+mmx pentium_pro pentium+mmx pentium i486 i386 i86
```

在上面的示例中，amd64 表示系统启用了 64 位功能。

▼ 如何显示 Oracle Solaris 发行版信息

- 显示 `/etc/release` 文件的内容，以确定发行版的版本。

```
$ cat /etc/release
Oracle Solaris 10 s10_51 SPARC
Copyright (c) 1983, 2011, Oracle and/or its affiliates. All Rights Reserved.
Use is subject to license terms.
Assembled 28 February 20011
```

▼ 如何显示常规系统信息

- 要显示常规系统信息，请使用 `showrev` 命令。

```
$ showrev options
-a          列显所有可用的系统修订版信息。
-c (command) 列显有关命令的修订版信息。
-p          只列显有关修补程序的修订版信息。
-R (root_path) 定义要用作 root_path 的目录的全路径名。
-s (host name) 对指定的主机名执行此操作。
-w          只列显 OpenWindows 修订版信息。
```

也可以使用 `uname` 命令来显示系统信息。以下示例显示 `uname` 命令输出。-a 选项显示操作系统名称和系统节点名称、操作系统发行版、操作系统版本、硬件名称和处理器类型。

```
$ uname
SunOS
$ uname -a
SunOS starbug 5.10 Generic sun4u sparc SUNW,Ultra-5_10
$
```

示例 5-3 显示常规系统信息

以下示例显示 `showrev` 命令的输出。-a 选项显示所有可用的系统信息。

```
$ showrev -a
Hostname: stonetouch
Hostid: 8099dfb9
Release: 5.10
Kernel architecture: sun4u
```

```
Application architecture: sparc
Hardware provider:
Domain:
Kernel version: SunOS 5.10 s10_46

OpenWindows version:
Solaris X11 Version 6.6.2 19 November 2010
No patches are installed
```

▼ 如何显示系统的主机 ID 号

- 要以十六进制格式显示主机 ID 号，请使用 **hostid** 命令。

示例 5-4 显示系统的主机 ID 号

以下示例显示 **hostid** 命令的样例输出。

```
$ hostid
80a5d34c
```

▼ 如何显示系统的产品名称

Solaris 10 1/06：在 **prtconf** 命令中使用 **-b** 选项可以显示系统的产品名称。有关此功能的更多信息，请参见 [prtconf\(1M\)](#) 手册页。

- 要显示系统的产品名称，请将 **prtconf** 命令与 **-b** 选项结合使用，如下所示：
% prtconf -b

示例 5-5 显示系统的产品名称

此示例显示 **prtconf -b** 命令的样例输出。

```
$ prtconf -b
name: SUNW,Ultra-5_10
model: SUNW,375-0066
banner-name: Sun Ultra 5/10 UPA/PCI (UltraSPARC-IIi 333MHz)
```

此示例显示 **prtconf -vb** 命令的样例输出。

```
$ prtconf -vb
name: SUNW,Ultra-5_10
model: SUNW,375-0066
banner-name: Sun Ultra 5/10 UPA/PCI (UltraSPARC-IIi 333MHz)
idprom: 01800800.20a6c363.00000000.a6c363a9.00000000.00000000.405555aa.aa555500
openprom model: SUNW,3.15
openprom version: 'OBP 3.15.2 1998/11/10 10:35'
```

▼ 如何显示系统的已安装内存

- 要显示系统中已安装的内存量，请使用 **prtconf** 命令。

示例 5-6 显示系统的已安装内存

以下示例显示 **prtconf** 命令的样例输出。**grep Memory** 命令选择 **prtconf** 命令的输出，以便只显示内存信息。

```
$ prtconf | grep Memory
Memory size: 128 Megabytes
```

▼ 如何显示日期和时间

- 要根据系统时钟显示当前的日期和时间，请使用 **date** 命令。

示例 5-7 显示日期和时间

以下示例显示 **date** 命令的样例输出。

```
$ date
Wed Jan 21 17:32:59 MST 2004
$
```

▼ 如何显示系统的物理处理器类型

- 使用 **psrinfo -p** 命令可以显示系统中物理处理器的总数。

```
$ psrinfo -p
1
```

使用 **psrinfo -pv** 命令可以显示有关系统中的每个物理处理器的信息以及与每个物理处理器关联的虚拟处理器。

```
$ psrinfo -pv
The UltraSPARC-IV physical processor has 2 virtual processors (8, 520)
The UltraSPARC-IV physical processor has 2 virtual processors (9, 521)
The UltraSPARC-IV physical processor has 2 virtual processors (10, 522)
The UltraSPARC-IV physical processor has 2 virtual processors (11, 523)
The UltraSPARC-III+ physical processor has 1 virtual processor (16)
The UltraSPARC-III+ physical processor has 1 virtual processor (17)
The UltraSPARC-III+ physical processor has 1 virtual processor (18)
The UltraSPARC-III+ physical processor has 1 virtual processor (19)
```

在基于 x86 的系统中使用 `psrinfo -pv` 命令时，将显示以下输出：

```
$ psrinfo -pv
The i386 physical processor has 2 virtual processors (0, 2)
The i386 physical processor has 2 virtual processors (1, 3)
```

▼ 如何显示系统的逻辑处理器类型

- 使用 `psrinfo -v` 命令可以显示有关系统的处理器类型的信息。

```
$ psrinfo -v
```

在基于 x86 的系统中，使用 `isalist` 命令可以显示虚拟处理器类型。

```
$ isalist
```

示例 5-8 SPARC: 显示系统的处理器类型

此示例说明如何显示有关基于 SPARC 的系统的处理器类型的信息。

```
$ psrinfo -v
Status of virtual processor 0 as of: 04/16/2004 10:32:13
  on-line since 03/22/2004 19:18:27.
  The sparcv9 processor operates at 650 MHz,
  and has a sparcv9 floating point processor.
```

示例 5-9 x86: 显示系统的处理器类型

此示例说明如何显示有关基于 x86 的系统的处理器类型的信息。

```
$ isalist
pentium_pro+mmx pentium_pro pentium+mmx pentium i486 i386 i86
```

▼ 如何显示系统中安装的语言环境

- 1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的“配置 RBAC（任务列表）”。

- 2 使用 `localeadm` 命令显示系统中当前安装的语言环境。`-l` 选项显示系统中安装的语言环境。例如：

```
# localeadm -l
Checking for installed pkgs. This could take a while.

Checking for Australasia region (aau)
```



```
(1of2 pkgs)
|.....|
.
.
.
The following regions are installed on concordance on Wed Dec 17 15:13:00 MST 2003

POSIX (C)

Central Europe (ceu)
[ Austria, Czech Republic, Germany, Hungary, Poland, Slovakia,
Switzerland (German), Switzerland (French) ]

Done.
```

▼ 如何确定系统中是否安装了语言环境

- 1 成为超级用户或承担等效角色。
角色包含授权和具有特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的“配置 RBAC（任务列表）”。
- 2 使用 `localeadm` 命令确定系统中是否安装了语言环境。`-q` 选项和语言环境会查询系统，以确定该系统中是否安装了语言环境。例如，要确定系统中是否安装了中欧 (Central European, ceu) 地区语言环境，可以运行以下命令：

```
# localeadm -q ceu
locale/region name is ceu
Checking for Central Europe region (ceu)
.
.
.
The Central Europe region (ceu) is installed on this system
```

更改系统信息（任务列表）

任务	说明	参考
手动设置系统的日期和时间。	使用 <code>date mmddHHMM[[cc]yy]</code> 命令行语法手动设置系统的日期和时间。	第 66 页中的“如何手动设置系统的日期和时间”
设置每日消息。	通过编辑 <code>/etc/motd</code> 文件在系统中设置每日消息。	第 67 页中的“如何设置每日消息”

任务	说明	参考
更改系统的主机名。	通过编辑以下文件来更改系统的主机名： <ul style="list-style-type: none">■ /etc/nodename■ /etc/hostname.*<i>host-name</i>■ /etc/inet/hosts 注 – 如果运行的是 Solaris 3/05、1/06、6/06 或 11/06 发行版，则还需要更新 /etc/inet/ipnodes 文件。从 Solaris 10 8/07 发行版开始，OS 中不再有两个单独的 hosts 文件。/etc/inet/hosts 文件是同时包含 IPv4 项和 IPv6 项的单个 hosts 文件。	第 68 页中的“如何更改系统的主机名”
向系统中添加语言环境。	使用 localeadm 命令可以向系统中添加语言环境。	如何向系统中添加语言环境
从系统中删除语言环境。	使用 localeadm 命令的 -r 选项和语言环境从系统中删除语言环境。	如何从系统中删除语言环境

更改系统信息

本节介绍可用来更改常规系统信息的命令。

▼ 如何手动设置系统的日期和时间

- 1 成为超级用户或承担等效角色。
角色包含授权和具有特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的“配置 RBAC（任务列表）”。
- 2 输入新的日期和时间。
date mmddHHMM[[cc]yy]
mm 月份，使用两位数。
dd 月份中的某日，使用两位数。
HH 小时，使用两位数和 24 小时制。
MM 分钟，使用两位数。

`cc` 世纪，使用两位数。

`yy` 年份，使用两位数。

有关更多信息，请参见 [date\(1\)](#) 手册页。

- 3 使用不带选项的 `date` 命令验证是否正确重置了系统日期。

示例 5-10 手动设置系统的日期和时间

以下示例说明如何使用 `date` 命令手动设置系统的日期和时间。

```
# date
Wed Mar  3 14:04:19 MST 2004
# date 0121173404
Thu Jan 21 17:34:34 MST 2004
```

▼ 如何设置每日消息

编辑每日消息文件 `/etc/motd`，使其包括在系统的所有用户登录时都会显示的声明或询问。请尽量少使用此功能，并定期编辑此文件，以删除过时的消息。

- 1 成为超级用户或承担等效角色。
角色包含授权和具有特权的命令。有关角色的更多信息，请参见《[系统管理指南：安全性服务](#)》中的“配置 RBAC（任务列表）”。
- 2 编辑 `/etc/motd` 文件并添加您选择的消息。
编辑文本以包括要在用户登录期间显示的消息。包括空格、制表符和回车。
- 3 通过显示 `/etc/motd` 文件的内容来验证更改。

```
$ cat /etc/motd
Welcome to the UNIX Universe. Have a nice day.
```

示例 5-11 设置每日消息

安装 Oracle Solaris 软件时提供的缺省每日消息中包含版本信息。

```
$ cat /etc/motd
Oracle Corporation   SunOS 5.10           Generic  January 2005
```

以下示例显示了一个经过编辑的 `/etc/motd` 文件，该文件可向登录的每个用户提供有关系统可用性的信息。

```
$ cat /etc/motd
The system will be down from 7:00 a.m to 2:00 p.m. on
```

Saturday, July 7, for upgrades and maintenance.
Do not try to access the system during those hours.
Thank you.

▼ 如何更改系统的主机名

系统的主机名在多个不同位置指定。

请记得更新名称服务器数据库，以反映新的主机名。

使用以下过程更改或重命名系统的主机名。

也可以使用 `sys-unconfig` 命令来重新配置系统，包括主机名。有关更多信息，请参见 [sys-unconfig\(1M\)](#) 手册页。

1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《[系统管理指南：安全性服务](#)》中的“[配置 RBAC（任务列表）](#)”。

2 在以下文件中更改系统的主机名：

- `/etc/nodename`
- `/etc/hostname.*interface`
- `/etc/inet/hosts`
- `/etc/inet/ipnodes`（仅适用于某些发行版）

注 - 从 Solaris 10 8/07 发行版开始，不再有两个独立的 `hosts` 文件。`/etc/inet/hosts` 文件是同时包含 IPv4 项和 IPv6 项的单个 `hosts` 文件。无需在始终要求同步的两个 `hosts` 文件中维护 IPv4 项。为了向后兼容，`/etc/inet/ipnodes` 文件被替换为与 `/etc/inet/hosts` 文件同名的符号链接。有关更多信息，请参见 [hosts\(4\)](#) 手册页。

3 可选如果您使用了名称服务，则在 `hosts` 文件中更改系统的主机名。

4 重命名 `/var/crash` 目录中的主机名目录。

```
# cd /var/crash
# mv old-host-name new-host-name
```

5 重新引导系统，以激活新的主机名。

```
# init 6
```

▼ 如何向系统中添加语言环境

1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《[系统管理指南：安全性服务](#)》中的“[配置 RBAC（任务列表）](#)”。

2 使用 `localeadm` 命令添加要在系统中安装的语言环境的软件包。

`-a` 选项后跟语言环境（标识您要添加的语言环境）。`-d` 选项后跟设备（标识含有您要添加的语言环境软件包的设备）。例如，要向系统中添加中欧 (Central European, ceu) 地区语言环境，可运行以下命令：

```
# localeadm -a ceu -d /net/install/latest/Solaris/Product

locale/region name is ceu

Devices are /net/install/latest/Solaris/Product
.
.
.
One or more locales have been added.
To update the list of locales available at
.
.
.
```

▼ 如何从系统中删除语言环境

1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《[系统管理指南：安全性服务](#)》中的“[配置 RBAC（任务列表）](#)”。

2 使用 `localeadm` 命令删除系统中已安装的语言环境软件包。

`-r` 选项后跟语言环境（标识您要从系统中删除的语言环境）。例如，要从系统中删除中欧 (Central European, ceu) 地区语言环境，可运行以下命令：

```
# localeadm -r ceu
locale/region name is ceu
Removing packages for Central Europe (ceu)
.
.
.
One or more locales have been removed.
To update the list of locales available
at the login screen's "Options->Language" menu,
.
.
.
```


管理磁盘使用（任务）

本章介绍如何通过查找未使用的文件和大规模目录来优化磁盘空间。

有关与管理磁盘使用关联的过程的信息，请参见第 71 页中的“管理磁盘使用（任务列表）”。

管理磁盘使用（任务列表）

任务	说明	参考
显示有关文件和磁盘空间的信息。	使用 <code>df</code> 命令显示有关磁盘空间使用情况的信息。	第 72 页中的“如何显示有关文件和磁盘空间的信息”
显示文件的大小。	使用带有 <code>-lh</code> 选项的 <code>ls</code> 命令显示有关文件大小的信息。	第 75 页中的“如何显示文件大小”
查找大文件。	使用 <code>ls -s</code> 命令可按大小对文件进行降序排序。	第 76 页中的“如何查找大文件”
查找超过指定大小限制的文件。	使用带有 <code>-size</code> 选项的 <code>find</code> 命令和指定大小限制值来查找并显示超过指定大小的文件的名称。	第 77 页中的“如何查找超过指定大小限制的文件”
显示目录、子目录和文件的大小。	使用 <code>du</code> 命令显示一个或多个目录、子目录和文件的大小。	第 78 页中的“如何显示目录、子目录和文件的大小”
显示本地 UFS 文件系统的所有权。	使用 <code>quot -a</code> 命令显示文件的所有权。	第 79 页中的“如何显示本地 UFS 文件系统的用户所有权”
列出最新的文件。	使用 <code>ls -t</code> 命令首先显示最新创建或更改的文件。	第 80 页中的“如何列出最新文件”

任务	说明	参考
查找并删除旧文件或非活动文件。	使用带有 <code>-atime</code> 和 <code>-mtime</code> 选项的 <code>find</code> 命令查找在指定天数内未访问的文件。可以使用 <code>rm 'cat filename'</code> 命令来删除这些文件。	第 81 页中的“如何查找并删除旧文件或非活动文件”
清除临时目录。	查找临时目录，然后使用 <code>rm -r *</code> 命令删除整个目录。	第 82 页中的“如何清除临时目录”
查找并删除核心文件。	使用 <code>find . -name core -exec rm {} \;</code> 命令查找并删除核心文件。	第 82 页中的“如何查找并删除 core 文件”
删除故障转储文件。	使用 <code>rm *</code> 命令删除 <code>/var/crash/</code> 目录中的故障转储文件。	第 83 页中的“如何删除故障转储文件”

显示有关文件和磁盘空间的信息

本表概括说明了可用于显示有关文件大小和磁盘空间信息的命令。

命令	说明	手册页
<code>df</code>	报告空闲磁盘块和文件的数量	df(1M)
<code>du</code>	概述分配给每个子目录的磁盘空间	du(1)
<code>find -size</code>	根据 <code>-size</code> 选项指定的大小递归搜索目录	find(1)
<code>ls -lh</code>	以 1024 的幂次方形式列出文件大小	ls(1)

▼ 如何显示有关文件和磁盘空间的信息

- 使用 `df` 命令可显示有关磁盘空间使用情况的信息。

<code>\$ df [directory] [-h] [-t]</code>	
<code>df</code>	没有任何选项时，该命令将列出所有已挂载文件系统及其设备名称、使用的 512 字节块的数量以及文件数量。
<code>directory</code>	指定要检查其文件系统的目录。
<code>-h</code>	以 1024 的幂次方形式显示磁盘空间。
<code>-t</code>	显示总块数以及用于所有已挂载文件系统的块数。

示例 6-1 显示有关文件大小和磁盘空间的信息

在以下示例中，除 `/usr/dist` 外，所列的所有文件系统都是本地挂载。

```
$ df
/                (/dev/dsk/c0t0d0s0 ): 101294 blocks 105480 files
/devices         (/devices          ):      0 blocks      0 files
/system/contract (ctfs              ):      0 blocks 2147483578 files
/proc            (proc                ):      0 blocks   1871 files
/etc/mnttab       (mnttab            ):      0 blocks      0 files
/etc/svc/volatile (swap              ): 992704 blocks 16964 files
/system/object    (objfs             ):      0 blocks 2147483530 files
/usr              (/dev/dsk/c0t0d0s6 ): 503774 blocks 299189 files
/dev/fd           (fd                 ):      0 blocks      0 files
/var/run          (swap              ): 992704 blocks 16964 files
/tmp              (swap              ): 992704 blocks 16964 files
/opt              (/dev/dsk/c0t0d0s5 ): 23914 blocks  6947 files
/export/home      (/dev/dsk/c0t0d0s7 ): 16810 blocks  7160 files
```

示例 6-2 以 1024 字节为单位显示具有 UFS 根文件系统的系统的文件大小信息

在以下示例中，以 1024 字节为单位显示具有 UFS 根文件系统的系统的文件系统信息。

```
$ df -h
Filesystem      size  used  avail capacity  Mounted on
/dev/dsk/c0t0d0s0 249M  200M   25M    90%      /
/devices         0K    0K    0K     0%    /devices
ctfs              0K    0K    0K     0%    /system/contract
proc              0K    0K    0K     0%    /proc
mnttab            0K    0K    0K     0%    /etc/mnttab
swap             485M  376K  485M    1%    /etc/svc/volatile
objfs             0K    0K    0K     0%    /system/object
/dev/dsk/c0t0d0s6 3.2G  2.9G  214M   94%    /usr
fd                0K    0K    0K     0%    /dev/fd
swap             485M  40K   485M    1%    /var/run
swap             485M  40K   485M    1%    /tmp
/dev/dsk/c0t0d0s5 13M   1.7M   10M   15%    /opt
/dev/dsk/c0t0d0s7 9.2M   1.0M   7.3M   13%    /export/home
```

尽管 `/proc` 和 `/tmp` 是本地文件系统，但不是 UFS 文件系统。`/proc` 是 PROCFS 文件系统，`/var/run` 和 `/tmp` 是 TMPFS 文件系统，`/etc/mnttab` 是 MNTFS 文件系统。

示例 6-3 以 1024 字节为单位显示具有 ZFS 根文件系统的系统的文件大小信息

在以下示例中，以 1024 字节为单位显示具有 Oracle Solaris ZFS 根文件系统的系统的文件系统信息。

```
Filesystem      size  used  avail capacity  Mounted on
rpool/ROOT/s1008be 67G  4.6G   58G     8%      /
/devices         0K    0K    0K     0%    /devices
ctfs              0K    0K    0K     0%    /system/contract
proc              0K    0K    0K     0%    /proc
```

mnttab	0K	0K	0K	0%	/etc/mnttab
swap	1.9G	1.5M	1.9G	1%	/etc/svc/volatile
objfs	0K	0K	0K	0%	/system/object
sharefs	0K	0K	0K	0%	/etc/dfs/sharetab
/platform/sun4u-us3/lib/libc_psr/libc_psr_hwcapi.so.1					
	63G	4.6G	58G	8%	/platform/sun4u-us3/lib/libc_psr.so.1
/platform/sun4u-us3/lib/sparcv9/libc_psr/libc_psr_hwcapi.so.1					
	63G	4.6G	58G	8%	/platform/sun4u-us3/lib/sparcv9/libc_psr.so.1
fd	0K	0K	0K	0%	/dev/fd
rpool/ROOT/s1008be/var					
	67G	73M	58G	1%	/var
swap	1.9G	32K	1.9G	1%	/tmp
swap	1.9G	40K	1.9G	1%	/var/run
rpool/export	67G	20K	58G	1%	/export
rpool/export/home	67G	18K	58G	1%	/export/home

示例 6-4 显示文件系统分配的总块数和总文件数

以下示例显示所有已挂载文件系统、设备名称、使用的 512 字节块总数以及文件数量的列表。每个两行项的第二行都显示为文件系统分配的总块数和总文件数。

```
$ df -t
/ (/dev/dsk/c0t0d0s0 ): 101294 blocks 105480 files
                        total: 509932 blocks 129024 files
/devices (/devices ): 0 blocks 0 files
                        total: 0 blocks 113 files
/system/contract (ctfs ): 0 blocks 2147483578 files
                        total: 0 blocks 69 files
/proc (proc ): 0 blocks 1871 files
                        total: 0 blocks 1916 files
/etc/mnttab (mnttab ): 0 blocks 0 files
                        total: 0 blocks 1 files
/etc/svc/volatile (swap ): 992608 blocks 16964 files
                        total: 993360 blocks 17025 files
/system/object (objfs ): 0 blocks 2147483530 files
                        total: 0 blocks 117 files
/usr (/dev/dsk/c0t0d0s6 ): 503774 blocks 299189 files
                        total: 6650604 blocks 420480 files
/dev/fd (fd ): 0 blocks 0 files
                        total: 0 blocks 31 files
/var/run (swap ): 992608 blocks 16964 files
                        total: 992688 blocks 17025 files
/tmp (swap ): 992608 blocks 16964 files
                        total: 992688 blocks 17025 files
/opt (/dev/dsk/c0t0d0s5 ): 23914 blocks 6947 files
                        total: 27404 blocks 7168 files
/export/home (/dev/dsk/c0t0d0s7 ): 16810 blocks 7160 files
                        total: 18900 blocks 7168 files
```

检查文件大小

使用 `ls` 命令可以检查文件的大小并对文件排序。使用 `find` 命令可以查找超过大小限制的文件。有关更多信息，请参见 [ls\(1\)](#) 和 [find\(1\)](#) 手册页。

注- 如果 `/var` 目录中的空间不足，请不要将 `/var` 目录符号链接到文件系统中具有更多磁盘空间的目录。这种做法即使作为一种临时措施，也可能导致某些守护进程和实用程序出现问题。

▼ 如何显示文件大小

- 1 转到要检查的文件所在的目录。
- 2 显示文件的大小。

```
$ ls [-lh] [-s]
-l    以长格式显示文件和目录的列表，以字节为单位显示大小。（请参见以下示例。）
-h    文件或目录大小大于 1024 字节时，请以 KB、MB、GB 或 TB 来表示文件大小和目录大小。该选项还可以修改 -o、-n、-@ 和 -g 选项显示的输出，以使用新格式显示文件或目录大小。有关更多信息，请参见 ls\(1\) 手册页。
-s    显示文件和目录（大小以块为单位）的列表。
```

示例 6-5 显示文件大小

以下示例表明，`lastlog` 和 `messages` 文件比 `/var/adm` 目录中的其他文件大。

```
$ cd /var/adm
$ ls -lh
total 148
drwxrwxr-x  5 adm      adm      512 Nov 26 09:39 acct/
-rw-----  1 uucp     bin      0 Nov 26 09:25 aculog
drwxr-xr-x  2 adm      adm      512 Nov 26 09:25 exacct/
-r--r--r--  1 root     other    342K Nov 26 13:56 lastlog
drwxr-xr-x  2 adm      adm      512 Nov 26 09:25 log/
-rw-r--r--  1 root     root     20K Nov 26 13:55 messages
drwxr-xr-x  2 adm      adm      512 Nov 26 09:25 passwd/
drwxrwxr-x  2 adm      sys      512 Nov 26 09:39 sa/
drwxr-xr-x  2 root     sys      512 Nov 26 09:49 sm.bin/
-rw-rw-rw-  1 root     bin      0 Nov 26 09:25 spellhist
drwxr-xr-x  2 root     sys      512 Nov 26 09:25 streams/
-rw-r--r--  1 root     bin      3.3K Nov 26 13:56 utmpx
-rw-r--r--  1 root     root      0 Nov 26 10:17 vold.log
-rw-r--r--  1 adm      adm      19K Nov 26 13:56 wtmpx
```

以下示例显示 `lpsched.1` 文件使用了两个块。

```
$ cd /var/lp/logs
$ ls -s
total 2          0 lpsched          2 lpsched.1
```

▼ 如何查找大文件

- 1 转到要搜索的目录。
- 2 以块为单位按从大到小的顺序显示文件大小。

- 如果文件的字符或列不同，请使用以下命令按块大小从大到小的顺序对文件列表进行排序。

```
$ ls -l | sort +4rn | more
```

请注意，此命令按照第四个字段中的字符（从左侧开始）对列表中的文件进行排序。

- 如果文件的字符或列相同，请使用以下命令按块大小从大到小的顺序对文件列表进行排序。

```
$ ls -s | sort -nr | more
```

请注意，此命令从最左侧的字符开始对列表中的文件进行排序。

示例 6-6 查找大文件（按第五个字段的字符进行排序）

```
$ cd /var/adm
$ ls -l | sort +4rn | more
-r--r--r-- 1 root root 4568368 Oct 17 08:36 lastlog
-rw-r--r-- 1 adm adm 697040 Oct 17 12:30 pacct.9
-rw-r--r-- 1 adm adm 280520 Oct 17 13:05 pacct.2
-rw-r--r-- 1 adm adm 277360 Oct 17 12:55 pacct.4
-rw-r--r-- 1 adm adm 264080 Oct 17 12:45 pacct.6
-rw-r--r-- 1 adm adm 255840 Oct 17 12:40 pacct.7
-rw-r--r-- 1 adm adm 254120 Oct 17 13:10 pacct.1
-rw-r--r-- 1 adm adm 250360 Oct 17 12:25 pacct.10
-rw-r--r-- 1 adm adm 248880 Oct 17 13:00 pacct.3
-rw-r--r-- 1 adm adm 247200 Oct 17 12:35 pacct.8
-rw-r--r-- 1 adm adm 246720 Oct 17 13:15 pacct.0
-rw-r--r-- 1 adm adm 245920 Oct 17 12:50 pacct.5
-rw-r--r-- 1 root root 190229 Oct 5 03:02 messages.1
-rw-r--r-- 1 adm adm 156800 Oct 17 13:17 pacct
-rw-r--r-- 1 adm adm 129084 Oct 17 08:36 wtmpx
```

示例 6-7 查找大文件（按最左侧的字符进行排序）

在以下示例中，`lastlog` 和 `messages` 文件是 `/var/adm` 目录中最大的文件。

```
$ cd /var/adm
$ ls -s | sort -nr | more
48 lastlog
30 messages
24 wtmpx
18 pacct
8 utmpx
2 vold.log
2 sulog
2 sm.bin/
2 sa/
2 passwd/
2 pacct1
2 log/
2 acct/
0 spellhist
0 aculog
total 144
```

▼ 如何查找超过指定大小限制的文件

- 要查找并显示超过指定大小的文件的名称，请使用 **find** 命令。

```
$ find directory -size +nnn
```

directory 标识要搜索的目录。

-size +nnn 512 字节块的数量。将列出超过此大小的文件。

示例 6-8 查找超过指定大小限制的文件

以下示例显示如何在当前工作目录中查找大于 400 块的文件。-print 选项显示 find 命令的输出。

```
$ find . -size +400 -print
./Howto/howto.doc
./Howto/howto.doc.backup
./Howto/howtotest.doc
./Routine/routineBackupconcepts.doc
./Routine/routineIntro.doc
./Routine/routineTroublefsck.doc
./.record
./Mail/pagination
./Config/configPrintadmin.doc
./Config/configPrintsetup.doc
./Config/configMailappx.doc
./Config/configMailconcepts.doc
./snapshot.rs
```

检查目录大小

可以使用 `du` 命令及选项显示目录的大小。此外，还可以使用 `quot` 命令来查找本地 UFS 文件系统中的用户帐户所使用的磁盘空间量。有关这些命令的更多信息，请参见 [du\(1\)](#) 和 [quot\(1M\)](#) 手册页。

▼ 如何显示目录、子目录和文件的大小

- 使用 `du` 命令显示一个或多个目录、子目录和文件的大小。以 512 字节块为单位显示大小。

```
$ du [-as] [directory ...]
```

<code>du</code>	显示您指定的每个目录的大小，包括目录下的每个子目录。
<code>-a</code>	显示每个文件和子目录的大小，以及指定目录中包含的总块数。
<code>-s</code>	显示指定目录中包含的总块数。
<code>-h</code>	以 1024 字节块为单位显示每个目录的大小。
<code>-H</code>	以 1000 字节块为单位显示每个目录的大小。
<code>[directory ...]</code>	标识要检查的一个或多个目录。命令行语法中的多个目录以空格分隔。

示例 6-9 显示目录、子目录和文件的大小

以下示例显示两个目录的大小。

```
$ du -s /var/adm /var/spool/lp
130    /var/adm
40     /var/spool/lp
```

以下示例显示两个目录的大小，其中包括每个目录下的所有子目录和文件的大小。还显示每个目录中包含的总块数。

```
$ du /var/adm /var/spool/lp
2      /var/adm/exacct
2      /var/adm/log
2      /var/adm/streams
2      /var/adm/acct/fiscal
2      /var/adm/acct/nite
2      /var/adm/acct/sum
8      /var/adm/acct
2      /var/adm/sa
2      /var/adm/sm.bin
258    /var/adm
4      /var/spool/lp/admins
2      /var/spool/lp/requests/printing....
```

```
4      /var/spool/lp/requests
4      /var/spool/lp/system
2      /var/spool/lp/fifos
24     /var/spool/lp
```

以下示例以 1024 字节块为单位显示目录大小。

```
$ du -h /usr/share/audio
796K  /usr/share/audio/samples/au
797K  /usr/share/audio/samples
798K  /usr/share/audio
```

▼ 如何显示本地 UFS 文件系统的用户所有权

1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《[系统管理指南：安全性服务](#)》中的“配置 RBAC（任务列表）”。

2 显示用户、目录或文件系统以及使用的 1024 字节块数。

```
# quot [-a] [filesystem ...]
```

-a 列出每个已挂载 UFS 文件系统的所有用户以及所用的 1024 字节块数。

filesystem 标识 UFS 文件系统。将显示该文件系统的用户以及所用块数。

注 - quot 命令仅适用于本地 UFS 文件系统。

示例 6-10 显示本地 UFS 文件系统的用户所有权

在以下示例中，将显示根 (/) 文件系统的用户。在后续示例中，将显示所有已挂载 UFS 文件系统的用户。

```
# quot /
/dev/rdisk/c0t0d0s0:
43340  root
3142   rimmer
47     uucp
35     lp
30     adm
4      bin
4      daemon

# quot -a
/dev/rdisk/c0t0d0s0 (/):
43340  root
3150   rimmer
47     uucp
```

```
35 lp
30 adm
4 bin
4 daemon
/dev/rdsd/c0t0d0s6 (/usr):
460651 root
206632 bin
791 uucp
46 lp
4 daemon
1 adm
/dev/rdsd/c0t0d0s7 (/export/home):
9 root
```

查找并删除旧文件或非活动文件

清理负载较大的文件系统的工作部分包括查找并删除最近未使用的文件。使用 `ls` 或 `find` 命令可以查找未使用的文件。有关更多信息，请参见 [ls\(1\)](#) 和 [find\(1\)](#) 手册页。

节省磁盘空间的其他方法包括清空临时目录（例如 `/var/tmp` 或 `/var/spool` 中的目录）以及删除 `core` 文件和故障转储文件。有关故障转储文件的更多信息，请参阅 [第 17 章，管理系统故障转储信息（任务）](#)。

▼ 如何列出最新文件

- 使用 `ls -t` 命令列出文件，首先会显示最近创建或更改的文件。

```
$ ls -t [directory]

-t          先按最近的时间戳对文件进行排序。

directory   标识要搜索的目录。
```

示例 6-11 列出最新文件

以下示例显示如何使用 `ls -tl` 命令在 `/var/adm` 目录中查找最近创建或更改的文件。`suolog` 文件是最近创建或编辑的文件。

```
$ ls -tl /var/adm
total 134
-rw----- 1 root    root      315 Sep 24 14:00 suolog
-r--r--r-- 1 root    other    350700 Sep 22 11:04 lastlog
-rw-r--r-- 1 root    bin      4464 Sep 22 11:04 utmpx
-rw-r--r-- 1 adm     adm      20088 Sep 22 11:04 wtmpx
-rw-r--r-- 1 root    other      0 Sep 19 03:10 messages
-rw-r--r-- 1 root    other      0 Sep 12 03:10 messages.0
-rw-r--r-- 1 root    root     11510 Sep 10 16:13 messages.1
-rw-r--r-- 1 root    root      0 Sep 10 16:12 vold.log
```



```
drwxr-xr-x 2 root sys 512 Sep 10 15:33 sm.bin
drwxrwxr-x 5 adm adm 512 Sep 10 15:19 acct
drwxrwxr-x 2 adm sys 512 Sep 10 15:19 sa
-rw----- 1 uucp bin 0 Sep 10 15:17 aculog
-rw-rw-rw- 1 root bin 0 Sep 10 15:17 spellhist
drwxr-xr-x 2 adm adm 512 Sep 10 15:17 log
drwxr-xr-x 2 adm adm 512 Sep 10 15:17 passwd
```

▼ 如何查找并删除旧文件或非活动文件

- 1 成为超级用户或承担等效角色。
角色包含授权和具有特权的命令。有关角色的更多信息，请参见《[系统管理指南：安全性服务](#)》中的“[配置 RBAC（任务列表）](#)”。
- 2 查找在指定的天数内未访问的文件并在一个文件中列出这些文件。

```
# find directory -type f [-atime +nnn] [-mtime +nnn] -print > filename &
```

directory 标识要搜索的目录。还会搜索此目录下面的子目录。

-atime +nnn 查找在您指定的天数内 (*nnn*) 未访问的文件。

-mtime +nnn 查找在您指定的天数内 (*nnn*) 未修改的文件。

filename 标识包含非活动文件列表的文件。
- 3 删除在前面的步骤中列出的非活动文件。

```
# rm 'cat filename'
```

其中 *filename* 标识在前面的步骤中创建的文件。此文件包含非活动文件的列表。

示例 6-12 查找并删除旧文件或非活动文件

以下示例显示 /var/adm 目录及子目录中在过去 60 天内未访问的文件。/var/tmp/deadfiles 文件包含非活动文件的列表。rm 命令删除这些非活动文件。

```
# find /var/adm -type f -atime +60 -print > /var/tmp/deadfiles &
# more /var/tmp/deadfiles
/var/adm/aculog
/var/adm/spellhist
/var/adm/wtmpx
/var/adm/sa/sa13
/var/adm/sa/sa27
/var/adm/sa/sa11
/var/adm/sa/sa23
/var/adm/sulog
/var/adm/vold.log
/var/adm/messages.1
/var/adm/messages.2
/var/adm/messages.3
```

```
# rm 'cat /var/tmp/deadfiles'
#
```

▼ 如何清除临时目录

- 1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《[系统管理指南：安全性服务](#)》中的“[配置 RBAC（任务列表）](#)”。

- 2 转到要清除的目录。

```
# cd directory
```



注意 - 完成步骤 3 之前，请确保您处于正确的目录中。步骤 3 将删除当前目录中的所有文件。

- 3 删除当前目录中的文件和子目录。

```
# rm -r *
```

- 4 转到包含不必要、临时或过时子目录和文件的其他目录。

- 5 重复步骤 3，删除这些子目录和文件。

示例 6-13 清除临时目录

以下示例显示如何清除 mywork 目录以及如何验证是否已删除所有文件和子目录。

```
# cd mywork
# ls
filea.000
fileb.000
filec.001
# rm -r *
# ls
#
```

▼ 如何查找并删除 core 文件

- 1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《[系统管理指南：安全性服务](#)》中的“[配置 RBAC（任务列表）](#)”。

- 2 转到要搜索 `core` 文件的目录。
- 3 查找并删除此目录及其子目录中的所有 `core` 文件。

```
# find . -name core -exec rm {} \;
```

示例 6-14 查找并删除 `core` 文件

以下示例显示如何使用 `find` 命令查找并删除 `jones` 用户帐户的 `core` 文件。

```
# cd /home/jones
# find . -name core -exec rm {} \;
```

▼ 如何删除故障转储文件

故障转储文件可能会很大。如果允许系统存储这些文件，除非必要，否则不要使其保留太长时间。

- 1 成为超级用户或承担等效角色。
角色包含授权和具有特权的命令。有关角色的更多信息，请参见《[系统管理指南：安全性服务](#)》中的“[配置 RBAC（任务列表）](#)”。
- 2 转到存储故障转储文件的目录。

```
# cd /var/crash/system
```

其中 `system` 标识创建了故障转储文件的系统。



注意 – 完成步骤 3 之前，请确保您处于正确的目录中。步骤 3 将删除当前目录中的所有文件。

- 3 删除故障转储文件。

```
# rm *
```
- 4 验证是否已删除故障转储文件。

```
# ls
```

示例 6-15 删除故障转储文件

以下示例显示如何从系统 `venus` 中删除故障转储文件，以及如何验证故障转储文件是否已删除。

```
# cd /var/crash/venus
# rm *
# ls
```


管理 UFS 配额（任务）

本章介绍如何设置和管理磁盘空间和 inode 的 UFS 配额。

有关与管理 UFS 配额相关的信息，请参见以下内容：

- 第 88 页中的“设置 UFS 配额（任务列表）”
- 第 92 页中的“维护 UFS 配额（任务列表）”

有关管理 Oracle Solaris ZFS 配额的信息，请参见《Oracle Solaris ZFS 管理指南》中的“设置 ZFS 配额和预留空间”。

什么是 UFS 配额？

通过 UFS 配额，系统管理员可以控制文件系统的大小。配额会限制磁盘空间量和 inode 数量，这些数量与各个用户可以获取的文件数量大致对应。因此，配额在用户起始目录驻留的文件系统中特别有用。通常，建立配额对 `public` 和 `/tmp` 文件系统不会有明显的益处。

使用 UFS 配额

设置配额后，即可更改配额，以调整用户可以占用的磁盘空间量或 inode 数量。此外，随着系统需要的变化，还可以添加或删除配额。有关更改配额或配额可以超出的时间、禁用各配额或从文件系统中删除配额的说明，请参见第 94 页中的“更改和删除 UFS 配额”。

此外，还可以监视配额状态。使用 UFS 配额命令，管理员可以显示有关文件系统中配额的信息，或搜索已超过配额的用户。有关介绍如何使用这些命令的过程，请参见第 92 页中的“检查 UFS 配额”。

设置 UFS 配额的软限制和硬限制

既可以设置软限制，也可以设置硬限制。系统不允许用户超过其硬限制。但是系统管理员可能会设置软限制，用户可以临时性地超过该软限制。软限制必须低于硬限制。

一旦用户超过软限制，配额计时器便开始计时。在配额计时器计时期间，用户可以使用高于软限制的配额，但不能超过硬限制。一旦用户低于软限制，计时器就将复位。但当计时器过期时，如果用户的使用配额一直在软限制以上，则会将软限制强制作为硬限制。缺省情况下，软限制计时器设置为七天。

`repquota` 和 `quota` 命令中的 `timeleft` 字段显示了计时器的值。

例如，假定某用户的软限制为 10,000 块，硬限制为 12,000 块。如果该用户的块使用量超过 10,000 块并且七天计时器已过期，则在用户的使用量降到软限制以下之前，该用户不能在该文件系统中分配更多磁盘块。

磁盘块与文件限制之间的区别

文件系统可为用户提供两种资源：用于数据的块和用于文件的 `inode`。每个文件占用一个 `inode`。文件数据存储在数据块中。数据块通常由 1 KB 的块组成。

假定不存在任何目录，则用户可以通过创建所有空文件而不使用任何块来超过其 `inode` 配额。用户也可以使用一个 `inode` 超过其块配额，方法是创建一个足够大的文件来占用用户配额中的所有数据块。

设置 UFS 配额

设置配额包括以下常规步骤：

1. 通过向 `/etc/vfstab` 文件项中添加一个配额选项，来确保每次重新引导系统时都会执行配额。此外，在文件系统的顶层目录中创建一个 `quotas` 文件。
2. 为一次使用创建配额后，将其作为样例进行复制即可设置其他用户配额。
3. 在启用配额之前，请先检查建议的配额与当前磁盘使用量之间的一致性，以确保没有冲突。
4. 为一个或多个文件系统启用配额。

有关这些过程的特定信息，请参见第 88 页中的“设置 UFS 配额（任务列表）”。

下表介绍了用于设置磁盘配额的命令。

表 7-1 用于设置 UFS 配额的命令

命令	任务	手册页
edquota	针对每个用户设置 inode 数量和磁盘空间量的硬限制和软限制。	edquota(1M)
quotacheck	检查每个已挂载的 UFS 文件系统，将文件系统的当前磁盘使用量与文件系统磁盘配额文件中存储的信息进行比较。然后，解决不一致问题。	quotacheck(1M)
quotaon	为指定的文件系统激活配额。	quotaon(1M)
quota	显示用户在已挂载文件系统中的 UFS 磁盘配额，以验证是否已正确设置配额。	quota(1M)

设置 UFS 配额的原则

设置 UFS 配额之前，需要确定要分配给每个用户的磁盘空间量和 inode 数量。如果要确保始终不超过文件系统总空间量，可根据用户数来划分文件系统空间。例如，如果三个用户共享 100 MB 的磁盘分片并且所需的磁盘空间相同，则可为每个用户分配 33 MB。

在并非所有用户都希望施加限制的环境中，您可能需要单独设置各用户的配额，以使配额总数大于文件系统的总大小。例如，如果三个用户共享 100 MB 的磁盘分片，则可为每个用户分配 40 MB。

使用 `edquota` 命令为一个用户建立配额后，可以将此配额用作样例，为同一文件系统中的其他用户设置相同配额。

在启用配额之前，请执行以下操作：

- 首先，为配额配置 UFS 文件系统。
- 为每个用户建立配额，并运行 `quotacheck` 命令检查当前磁盘使用量与配额文件之间的一致性。
- 定期运行 `quotacheck` 命令，前提是不会经常重新引导系统。

只有使用 `quotaon` 命令启用配额，才会执行使用 `edquota` 命令设置的配额。如果已正确配置了配额文件，则每次重新引导系统和挂载文件系统时，将自动启用配额。

设置 UFS 配额（任务列表）

任务	说明	参考
1. 为配额配置文件系统。	编辑 <code>/etc/vfstab</code> 文件，以便在每次挂载文件系统时激活配额。此外，还应创建一个 <code>quotas</code> 文件。	第 88 页中的“如何为 UFS 配额配置文件系统”
2. 设置一个用户的 UFS 配额。	使用 <code>edquota</code> 命令为单个用户帐户创建磁盘配额和 inode 配额。	第 89 页中的“如何设置一个用户的 UFS 配额”
3. （可选的）设置多个用户的 UFS 配额。	使用 <code>edquota</code> 命令将样例配额应用于其他用户帐户。	第 90 页中的“如何设置多个用户的 UFS 配额”
4. 检查一致性。	使用 <code>quotacheck</code> 命令将配额与当前磁盘使用量进行比较，以确保一个或多个文件系统中的一致性。	第 90 页中的“如何检查 UFS 配额一致性”
5. 启用 UFS 配额。	使用 <code>quotaon</code> 命令在一个或多个文件系统中启动 UFS 配额。	第 91 页中的“如何启用 UFS 配额”

▼ 如何为 UFS 配额配置文件系统

- 1 成为超级用户或承担等效角色。
角色包含授权和具有特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的“配置 RBAC（任务列表）”。
- 2 对于要获取配额的每个 UFS 文件系统，编辑 `/etc/vfstab` 文件并向 `mount options` 字段中添加 `rq`。
- 3 将目录转到将获取配额的文件系统的根目录。
- 4 创建一个名为 `quotas` 的文件。
`# touch quotas`
- 5 更改读取/写入权限，以便仅供超级用户访问。
`# chmod 600 quotas`

示例 7-1 为 UFS 配额配置文件系统

以下 `/etc/vfstab` 示例显示，系统 `pluto` 中的 `/export/home` 目录在本地系统中作为 NFS 文件系统挂载。通过 `mount options` 列下的 `rq` 项可以判断已启用了配额。


```
# device device mount FS fsck mount mount
# to mount to fsck point type pass at boot options
# pluto:/export/home - /export/home nfs - yes rq
```

/etc/vfstab 文件中的以下示例行说明本地 /work 目录在挂载时已启用配额，mount options 列下的 rq 项指明了这一点。

```
#device device mount FS fsck mount mount
#to mount to fsck point type pass at boot options
#/dev/dsk/c0t4d0s0 /dev/rdisk/c0t4d0s0 /work ufs 3 yes rq
```

- 另请参见
- [第 89 页中的“如何设置一个用户的 UFS 配额”](#)
 - [第 90 页中的“如何设置多个用户的 UFS 配额”](#)
 - [第 90 页中的“如何检查 UFS 配额一致性”](#)
 - [第 91 页中的“如何启用 UFS 配额”](#)

▼ 如何设置一个用户的 UFS 配额

- 1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《[系统管理指南：安全性服务](#)》中的“[配置 RBAC（任务列表）](#)”。

- 2 使用配额编辑器创建一个临时文件，对于在文件系统的根目录中包含 **quotas** 文件的每个已挂载的 UFS 文件系统，该临时文件都包含一行配额信息。

```
# edquota username
```

其中，*username* 是要为其设置配额的用户。

- 3 将 1 KB 磁盘块的数量（软限制和硬限制）更改为针对每个文件系统指定的配额。
- 4 将 inode 的数量（软限制和硬限制）从缺省值 0 更改为针对每个文件系统指定的配额。
- 5 验证用户的 UFS 配额。

```
# quota -v username
```

-v 在存在配额的所有已挂载文件系统中显示用户的配额信息。

username 指定要查看配额限制的用户名。

示例 7-2 设置一个用户的 UFS 配额

以下示例显示了 **edquota** 在某个系统中打开的临时文件的内容，在该系统中，/files 是根目录中包含 **quotas** 文件的唯一已挂载文件系统。

```
fs /files blocks (soft = 0, hard = 0) inodes (soft = 0, hard = 0)
```

以下示例显示了设置配额后临时文件中的同一行。

```
fs /files blocks (soft = 50, hard = 60) inodes (soft = 90, hard = 100)
```

▼ 如何设置多个用户的 UFS 配额

- 1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《[系统管理指南：安全性服务](#)》中的“[配置 RBAC（任务列表）](#)”。

- 2 使用配额编辑器将已为样例用户建立的配额应用到您指定的其他用户。

```
# edquota -p prototype-user username ...
```

prototype-user 已为其设置了配额的帐户的用户名。

username ... 指定其他帐户的一个或多个用户名。通过以空格分隔每个用户名来指定多个用户名。

示例 7-3 设置多个用户的样例 UFS 配额

以下示例说明如何将为用户 bob 建立的配额应用到用户 mary 和 john。

```
# edquota -p bob mary john
```

▼ 如何检查 UFS 配额一致性

重新引导系统时会自动运行 quotacheck 命令。通常不需要在具有配额的空文件系统中运行 quotacheck 命令。但是，如果要在包含现有文件的文件系统中设置配额，则需要运行 quotacheck 命令，以使配额数据库与文件系统中已存在的文件或 inode 同步。

此外应谨记，在大型文件系统中运行 quotacheck 命令会非常耗时。

注 - 为确保磁盘数据的准确性，在手动运行 quotacheck 命令时，检查的文件系统应处于停顿状态。

- 1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《[系统管理指南：安全性服务](#)》中的“[配置 RBAC（任务列表）](#)”。

- 2 对 UFS 文件系统运行一致性检查。

```
# quotacheck [-va] filesystem
```

-v （可选的）确定每个用户在特定文件系统中的磁盘配额。

-a 检查 /etc/vfstab 文件中包含 rq 项的所有文件系统。

filesystem 指定要检查的文件系统。

有关更多信息，请参见 [quotacheck\(1M\)](#) 手册页。

示例 7-4 检查 UFS 配额一致性

以下示例说明如何检查 /dev/rdisk/c0t0d0s7 磁盘分片中 /export/home 文件系统的配额。/export/home 文件系统是 /etc/vfstab 文件中具有 rq 项的唯一文件系统。

```
# quotacheck -va
*** Checking quotas for /dev/rdisk/c0t0d0s7 (/export/home)
```

▼ 如何启用 UFS 配额

1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《[系统管理指南：安全性服务](#)》中的“配置 RBAC（任务列表）”。

2 启用文件系统配额。

```
# quotaon [-v] -a filesystem ...
```

-v 启用配额后为每个文件系统显示一条消息。

-a 为在 /etc/vfstab 文件中包含 rq 项的所有文件系统启用配额。

filesystem... 为指定的一个或多个文件系统启用配额。指定多个文件系统时，需要用空格来分隔每个文件系统的名称。

示例 7-5 启用 UFS 配额

以下示例说明如何为 /dev/dsk/c0t4d0s7 和 /dev/dsk/c0t3d0s7 磁盘分片上的文件系统启用配额。

```
# quotaon -v /dev/dsk/c0t4d0s7 /dev/dsk/c0t3d0s7
/dev/dsk/c0t4d0s7: quotas turned on
/dev/dsk/c0t3d0s7: quotas turned on
```

维护 UFS 配额（任务列表）

任务	说明	参考
检查超过的 UFS 配额。	使用 <code>quota</code> 命令显示已激活 UFS 配额的文件系统中各用户的 UFS 配额和磁盘使用情况。	第 92 页中的“如何检查超过的 UFS 配额”
检查文件系统中的 UFS 配额。	使用 <code>repquota</code> 命令显示一个或多个文件系统中所有用户的 UFS 配额和磁盘使用情况。	第 93 页中的“如何检查文件系统中的 UFS 配额”
更改软限制缺省值。	使用 <code>edquota</code> 命令更改用户可以超过其磁盘空间配额或 <code>inode</code> 配额的时间长度。	第 95 页中的“如何更改软限制缺省值”
更改用户的 UFS 配额。	使用配额编辑器 <code>edquota</code> 更改单个用户的配额。	第 96 页中的“如何更改用户的 UFS 配额”
禁用用户的 UFS 配额。	使用配额编辑器 <code>edquota</code> 为单个用户禁用配额。	第 97 页中的“如何禁用用户的 UFS 配额”
禁用 UFS 配额。	使用 <code>quotaoff</code> 命令禁用 UFS 配额。	第 98 页中的“如何禁用 UFS 配额”

检查 UFS 配额

设置并启用 UFS 磁盘配额和 `inode` 配额后，即可检查是否有用户超过了其配额。此外，还可以检查整个文件系统的配额信息。

下表介绍了用于检查配额的命令。

表 7-2 用于检查 UFS 配额的命令

命令	任务
quota(1M)	显示用户配额和当前磁盘使用情况，以及有关超过其配额的用户的信息
repquota(1M)	显示指定文件系统所拥有的配额、文件和空间量

▼ 如何检查超过的 UFS 配额

您可以使用 `quota` 命令显示已激活配额的文件系统中各用户的 UFS 配额和磁盘使用情况。

- 1 成为超级用户或承担等效角色。
角色包含授权和具有特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的“配置 RBAC（任务列表）”。
- 2 显示启用配额的已挂载文件系统的用户配额。
`# quota [-v] username`
`-v` 显示在具有配额的所有已挂载文件系统中的—个或多个用户的配额。
`username` 用户帐户的登录名或 UID。

示例 7-6 检查超过的 UFS 配额

以下示例显示由 UID 301 标识的用户帐户有一个 1KB 的配额，但尚未使用任何磁盘空间。

```
# quota -v 301
Disk quotas for bob (uid 301):
Filesystem  usage  quota limit timeleft files quota  limit timeleft
/export/home  0      1      2      0      2      3

Filesystem  文件系统的挂载点。
usage        当前的块使用情况。
quota        软块限制。
limit        硬块限制。
timeleft     配额计时器中剩余的时间，以天为单位。
files        当前的 inode 使用情况。
quota        软 inode 限制。
limit        硬 inode 限制。
timeleft     配额计时器中剩余的时间，以天为单位。
```

▼ 如何检查文件系统上的 UFS 配额

使用 repquota 命令显示一个或多个文件系统中所有用户的 UFS 配额和磁盘使用情况。

- 1 成为超级用户或承担等效角色。
角色包含授权和具有特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的“配置 RBAC（任务列表）”。

2 显示一个或多个文件系统的所有 UFS 配额（即使未使用也显示）。

```
# repquota [-v] -a filesystem
-v          报告所有用户的 UFS 配额，包括那些不占用资源的用户。
-a          报告所有文件系统。
filesystem  报告指定的文件系统。
```

示例 7-7 检查文件系统上的 UFS 配额

以下示例显示在仅对一个文件系统 (/export/home) 启用配额的系统中 repquota 命令的输出。

```
# repquota -va
/dev/dsk/c0t3d0s7 (/export/home):
      Block limits          File limits
User      used  soft  hard  timeleft  used  soft  hard  timeleft
#301  --      0      1  2.0 days  0      2      3
#341  --    57    50   60  7.0 days  2     90   100

Block limits  定义
used          当前的块使用情况。
soft          软块限制。
hard          硬块限制。
timeleft      配额计时器中剩余的时间，以天为单位。

File limits   是文件限制的定义。
used          当前的 inode 使用情况。
soft          软 inode 限制。
hard          硬 inode 限制。
timeleft      配额计时器中剩余的时间，以天为单位。
```

更改和删除 UFS 配额

您可以更改配额，以调整用户可以占用的磁盘空间量或 inode 数量。还可以根据需要，为各用户或从整个文件系统中删除配额。

下表介绍了用于更改配额或删除配额的命令。

表 7-3 用于更改和删除 UFS 配额的命令

命令	手册页	说明
edquota	edquota(1M)	更改每个用户的 inode 数量或磁盘空间量的硬限制和软限制。此外，还要更改具有配额的每个文件系统的软限制。
quotaoff	quotaon(1M)	禁用指定文件系统的配额。

▼ 如何更改软限制缺省值

缺省情况下，用户可以超过其 UFS 配额软时间限制一周。因此，在重复违反磁盘空间配额或 inode 配额软时间限制一周后，系统将禁止用户使用更多的 inode 或磁盘块。

使用 edquota 命令可以更改用户可超过其磁盘空间配额或 inode 配额的时间长度。

- 1 成为超级用户或承担等效角色。
角色包含授权和具有特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的“配置 RBAC（任务列表）”。
- 2 使用配额编辑器创建包含软时间限制的临时文件。
`# edquota -t`
其中，-t 选项指定为每个文件系统编辑软时间限制。
- 3 将时间限制从 0（缺省值）更改为指定的时间限制。因此，可使用数字和关键字 month、week、day、hour、min 或 sec。

注 - 此过程不会影响当前的配额违反者。

示例 7-8 更改软限制缺省值

以下示例显示了 edquota 命令在某一系统（其中 /export/home 是唯一具有配额的已挂载文件系统）中打开的临时文件的内容。缺省值 0 表示使用的缺省时间限制为一周。

```
fs /export/home blocks time limit = 0 (default), files time limit = 0 (default)
```

以下示例显示了在将超过块配额的时间限制更改为 2 周之后的同一临时文件。而且，超过文件数的时间限制已更改为 16 天。

```
fs /export/home blocks time limit = 2 weeks, files time limit = 16 days
```

▼ 如何更改用户的 UFS 配额

- 1 成为超级用户或承担等效角色。
角色包含授权和具有特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的“配置 RBAC（任务列表）”。
- 2 使用配额编辑器打开临时文件，对于在文件系统的根目录中具有 `quotas` 文件的每个已挂载文件系统，该文件都包含一行配额信息。

```
# edquota username
```

其中，`username` 指定要更改其配额的用户名。



注意 – 可以将多个用户指定为 `edquota` 命令的参数。但不显示此信息所属的用户。为避免混乱，请仅指定一个用户名。

- 3 指定 1 KB 磁盘块的数量（软限制和硬限制）和 `inode` 数量（软限制和硬限制）。
- 4 验证是否已正确更改了用户的 UFS 配额。

```
# quota -v username
```

`-v` 显示已启用配额的所有已挂载文件系统上的用户 UFS 配额信息。

`username` 指定要检查其配额的用户名。

示例 7-9 更改用户的 UFS 配额

以下示例显示了由 `edquota` 命令打开的临时文件的内容。在打开此临时文件的系统中，`/files` 是在文件系统的根目录中包含 `quotas` 文件的唯一已挂载文件系统。

```
fs /files blocks (soft = 0, hard = 0) inodes (soft = 0, hard = 0)
```

以下输出显示了更改配额后的同一临时文件。

```
fs /files blocks (soft = 0, hard = 500) inodes (soft = 0, hard = 100)
```

示例 7-10 验证是否已更改了硬 UFS 配额

以下示例解释如何验证用户 `smith` 的硬配额已更改为 500 个 1 KB 块和 100 个 `inode`。

```
# quota -v smith
Disk quotas for smith (uid 12):
Filesystem  usage  quota  limit  timeleft  files  quota  limit  timeleft
    /files      1      0    500          1      0    100
```


▼ 如何禁用用户的 UFS 配额

- 1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的“配置 RBAC（任务列表）”。

- 2 使用配额编辑器创建一个临时文件，对于在顶层目录中包含 `quotas` 文件的每个已挂载文件系统，该临时文件都包含一行配额信息。

```
# edquota username
```

其中，`username` 指定要禁用其配额的用户名。



注意 – 可以将多个用户指定为 `edquota` 命令的参数。但不显示此信息所属的用户。为避免混乱，请仅指定一个用户名。

- 3 将 1 KB 磁盘块的数量（软限制和硬限制）更改为 0。

- 4 将 inode 的数量（软限制和硬限制）更改为 0。

注 – 确保将上述值更改为零。请勿从文本文件中删除该行。

- 5 验证是否已禁用用户的 UFS 配额。

```
# quota -v username
```

`-v` 显示已启用配额的所有已挂载文件系统上的用户 UFS 配额信息。

`username` 指定要检查其 UFS 配额的用户名 (UID)。

示例 7-11 禁用用户的 UFS 配额

以下示例显示了由 `edquota` 命令在某一系统中打开的临时文件的内容，在该系统中，`/files` 是在文件系统的根目录中包含配额文件的唯一已挂载文件系统。

```
fs /files blocks (soft = 50, hard = 60) inodes (soft = 90, hard = 100)
```

以下示例显示了禁用 UFS 配额后的同一临时文件。

```
fs /files blocks (soft = 0, hard = 0) inodes (soft = 0, hard = 0)
```

▼ 如何禁用 UFS 配额

1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《[系统管理指南：安全性服务](#)》中的“[配置 RBAC（任务列表）](#)”。

2 禁用文件系统配额。

```
# quotaoff [-v] -a filesystem ...
```

-v 显示禁用 UFS 配额时每个文件系统上的消息。

-a 禁用所有文件系统的 UFS 配额。

filesystem 禁用指定的一个或多个文件系统的 UFS 配额。指定多个文件系统时，需要用空格来分隔每个文件系统的名称。

示例 7-12 禁用配额

以下示例说明如何为 `/export/home` 文件系统禁用配额。

```
# quotaoff -v /export/home
/export/home: quotas turned off
```

调度系统任务（任务）

本章介绍如何使用 `crontab` 和 `at` 命令来调度例程或单个（一次性）系统任务。

本章还说明如何使用以下文件来控制对上述命令的访问：

- `cron.deny`
- `cron-allow`
- `at.deny`

有关与调度系统任务关联的过程信息，请参见以下内容：

- 第 99 页中的“创建和编辑 `crontab` 文件（任务列表）”
- 第 111 页中的“使用 `at` 命令（任务列表）”

创建和编辑 `crontab` 文件（任务列表）

任务	说明	参考
创建或编辑 <code>crontab</code> 文件。	使用 <code>crontab -e</code> 命令可创建或编辑 <code>crontab</code> 文件。	第 104 页中的“如何创建或编辑 <code>crontab</code> 文件”
验证 <code>crontab</code> 文件是否存在。	使用 <code>ls -l</code> 命令验证 <code>/var/spool/cron/crontabs</code> 文件的内容。	第 105 页中的“如何验证 <code>crontab</code> 文件是否存在”
显示 <code>crontab</code> 文件。	使用 <code>ls -l</code> 命令显示 <code>crontab</code> 文件。	第 106 页中的“如何显示 <code>crontab</code> 文件”
删除 <code>crontab</code> 文件。	<code>crontab</code> 文件设置时使用了受限权限。使用 <code>crontab -r</code> 命令而不是 <code>rm</code> 命令删除 <code>crontab</code> 文件。	第 107 页中的“如何删除 <code>crontab</code> 文件”

任务	说明	参考
拒绝 crontab 访问。	要拒绝用户对 crontab 命令的访问，请通过编辑 /etc/cron.d/cron.deny 文件将用户名添加到该文件中。	第 109 页中的“如何拒绝 crontab 命令访问”
将 crontab 访问限定于指定的用户。	要允许用户访问 crontab 命令，请将用户名添加到 /etc/cron.d/cron.allow 文件中。	第 109 页中的“如何将 crontab 命令访问限制于指定的用户”

自动执行系统任务的方法

可以设置多个要自动执行的系统任务。其中的某些任务应该以固定间隔执行。其他任务只需运行一次，可能是在晚间或周末等非高峰时间。

本节包含有关 crontab 和 at 这两个命令的概括性信息，使用这两个命令可以调度要自动执行的例行任务。crontab 命令用于调度重复性任务。at 命令则用于调度只执行一次的任务。

下表概括说明了 crontab 和 at 命令，以及可用来控制对这些命令的访问的文件。

表 8-1 命令摘要：调度系统任务

命令	调度对象	文件位置	控制访问的文件
crontab	按固定间隔执行的多个系统任务	/var/spool/cron/crontabs	/etc/cron.d/cron.allow 和 /etc/cron.d/cron.deny
at	单个系统任务	/var/spool/cron/atjobs	/etc/cron.d/at.deny

也可以使用 Solaris Management Console 的调度作业工具来调度例程任务。有关使用和启动 Solaris Management Console 的信息，请参见《系统管理指南：基本管理》中的第 2 章“使用 Solaris Management Console（任务）”。

用于调度重复性作业：crontab

可以使用 crontab 命令来调度例行系统管理任务，使其每日、每周或每月执行一次。

每日 crontab 系统管理任务可能包括以下内容：

- 从临时目录中删除几天前的文件
- 执行记帐摘要命令
- 使用 df 和 ps 命令捕获系统快照
- 执行每日安全监视

- 运行系统备份

每周 `crontab` 系统管理任务可能包括以下内容：

- 重新生成 `catman` 数据库以供 `man -k` 命令使用
- 运行 `fsck -n` 命令以列出任何磁盘问题

每月 `crontab` 系统管理任务可能包括以下内容：

- 列出在特定月份中未使用的文件
- 生成每月记帐报告

此外，用户还可以调度 `crontab` 命令执行其他例行系统任务，例如发送提醒和删除备份文件。

有关调度 `crontab` 作业的逐步说明，请参见第 104 页中的“[如何创建或编辑 crontab 文件](#)”。

用于调度单个作业：at

通过 `at` 命令可以调度要在以后执行的作业。该作业可由单个命令或脚本组成。

与 `crontab` 类似，使用 `at` 命令可以调度例行任务自动执行。但与 `crontab` 文件不同的是，`at` 文件只执行任务一次。然后，便从目录中删除这些文件。因此，`at` 命令非常适合于运行将输出定向到单独文件中（供以后检查）的简单命令或脚本。

提交 `at` 作业包括键入命令并按照 `at` 命令语法指定选项来调度执行作业的时间。有关提交 `at` 作业的更多信息，请参见第 112 页中的“[at 命令的说明](#)”。

`at` 命令在 `/var/spool/cron/atjobs` 目录中存储您运行的命令或脚本以及当前环境变量的副本。`at` 作业文件名是一个长数字，用于指定该文件在 `at` 队列中的位置，后跟 `.a` 扩展名，例如 `793962000.a`。

`cron` 守护进程在启动时检查 `at` 作业并侦听是否提交了新作业。`cron` 守护进程执行 `at` 作业后，将从 `atjobs` 目录中删除 `at` 作业的文件。有关更多信息，请参见 [at\(1\)](#) 手册页。

有关调度 `at` 作业的逐步说明，请参见第 112 页中的“[如何创建 at 作业](#)”。

调度重复性系统任务 (cron)

以下各节介绍如何创建、编辑、显示和删除 crontab 文件，以及如何控制对这些文件的访问。

在 crontab 文件内

cron 守护进程会根据在每个 crontab 文件中找到的命令来调度系统任务。crontab 文件由命令组成，每个命令占据一行，这些命令将以固定间隔执行。每行开头包含日期和时间信息，以告知 cron 守护进程何时执行命令。

例如，在 SunOS 软件安装期间将提供名为 root 的 crontab 文件。该文件的内容包括以下命令行：

```
10 3 * * * /usr/sbin/logadm          (1)
15 3 * * 0 /usr/lib/fs/nfs/nfsfind    (2)
1 2 * * * [ -x /usr/sbin/rtc ] && /usr/sbin/rtc -c > /dev/null 2>&1      (3)
30 3 * * * [ -x /usr/lib/gss/gsscred_clean ] && /usr/lib/gss/gsscred_clean  (4)
```

下面介绍了上述每个命令行的输出：

- 第一行在每天凌晨 3:10 运行 logadm 命令。
- 第二行在每个星期日凌晨 3:15 执行 nfsfind 脚本。
- 第三行在每天凌晨 2:10 运行用于检查夏时制时间（并根据需要进行更正）的脚本。如果没有 RTC 时区，也没有 /etc/rtc_config 文件，则此项不执行任何操作。

仅适用于 x86 – /usr/sbin/rtc 脚本只能在基于 x86 的系统上运行。

- 第四行在每天凌晨 3:30 检查（并删除）通用安全服务表 /etc/gss/gsscred_db 中的重复项。

有关 crontab 文件中各行语法的更多信息，请参见第 103 页中的“crontab 文件项的语法”。

crontab 文件存储在 /var/spool/cron/crontabs 目录中。在 Oracle Solaris 软件安装期间，会提供包括 root 在内的若干个 crontab 文件。请参见下表。

表 8-2 缺省 crontab 文件

crontab 文件	功能
adm	记帐
lp	打印

表 8-2 缺省 crontab 文件 (续)

crontab 文件	功能
root	一般系统功能和文件系统清除
sys	性能数据收集
uucp	一般 uucp 清除

除了缺省的 crontab 文件之外，用户还可以创建 crontab 文件，以调度自己的系统任务。其他 crontab 文件按创建它们的用户帐户名称（如 bob、mary、smith 或 jones）命名。

要访问属于 root 或其他用户的 crontab 文件，需要具有超级用户特权。

说明如何创建、编辑、显示和删除 crontab 文件的过程在后续几节进行介绍。

cron 守护进程处理调度的方法

cron 守护进程可管理 crontab 命令的自动调度。cron 守护进程将检查 /var/spool/cron/crontab 目录中是否存在 crontab 文件。

cron 守护进程在启动时执行以下任务：

- 检查新的 crontab 文件。
- 阅读文件中列出的执行时间。
- 在适当时间提交执行命令。
- 侦听来自 crontab 命令的有关更新的 crontab 文件的通知。

cron 守护进程以几乎相同的方式来控制 at 文件的调度。这些文件存储在 /var/spool/cron/atjobs 目录中。cron 守护进程还侦听来自 crontab 命令的有关已提交的 at 作业的通知。

crontab 文件项的语法

crontab 文件由命令组成，每个命令占据一行，这些命令将按每个命令行的前五个字段指定的时间自动执行。下表中介绍了这五个字段，它们以空格分隔。

表 8-3 crontab 时间字段的可接受值

时间字段	值
分钟	0-59
小时	0-23
月中某日	1-31

表 8-3 crontab 时间字段的可接受值 (续)

时间字段	值
月份	1-12
星期中某日	0-6 (0 = 星期日)

在 crontab 时间字段中使用特殊字符时请遵循以下规则：

- 使用空格分隔每个字段。
- 使用逗号分隔多个值。
- 使用连字符指定某一范围的值。
- 使用星号作为通配符来包括所有可能值。
- 在一行开头使用注释标记 (#) 来表示注释或空白行。

例如，以下 crontab 命令项将于每月第一天和第十五天下午 4 点在用户的控制台窗口中显示提醒。

```
0 16 1,15 * * echo Timesheets Due > /dev/console
```

crontab 文件中的每个命令必须只占据一行，即使这一行非常长也是如此。crontab 文件不识别额外的回车。有关 crontab 项和命令选项的更多详细信息，请参阅 [crontab\(1\)](#) 手册页。

创建和编辑 crontab 文件

创建 crontab 文件的最简单方法是使用 crontab -e 命令。此命令会调用已为系统环境设置的文本编辑器。系统环境的缺省编辑器在 EDITOR 环境变量中定义。如果尚未设置此变量，crontab 命令将使用缺省编辑器 ed。最好选择您熟悉的编辑器。

以下示例说明如何确定是否已定义编辑器，以及如何将 vi 设置为缺省值。

```
$ which $EDITOR
$
$ EDITOR=vi
$ export EDITOR
```

创建 crontab 文件时，该文件会自动放入 /var/spool/cron/crontabs 目录，并以您的用户名命名。如果具有超级用户特权，则可为其他用户或 root 创建或编辑 crontab 文件。

▼ 如何创建或编辑 crontab 文件

开始之前 如果要创建或编辑属于 root 或其他用户的 crontab 文件，您必须成为超级用户或承担等效角色。角色包含授权和具有特权的命令。有关角色的更多信息，请参见《[系统管理指南：安全性服务](#)》中的“[配置 RBAC（任务列表）](#)”。

无需成为超级用户，便可编辑自己的 crontab 文件。

1 创建新的 crontab 文件，或编辑现有文件。

```
$ crontab -e [username]
```

其中，*username* 指定您要为其创建或编辑 crontab 文件的用户帐户的名称。无需具有超级用户特权便可创建自己的 crontab 文件，但如果要为 root 或其他用户创建或编辑 crontab 文件，则必须具有超级用户特权。



注意 – 如果意外键入了不带选项的 crontab 命令，请按下编辑器的中断字符。使用此字符将退出而不会保存更改。如果保存了更改并退出文件，现有的 crontab 文件将被空文件覆盖。

2 向 crontab 文件中添加命令行。

按照第 103 页中的“crontab 文件项的语法”中所述的语法操作。将 crontab 文件放入 /var/spool/cron/crontabs 目录。

3 验证 crontab 文件更改。

```
# crontab -l [username]
```

示例 8-1 创建 crontab 文件

以下示例说明如何为其他用户创建 crontab 文件。

```
# crontab -e jones
```

添加到新 crontab 文件中的以下命令项将在每个星期日的凌晨 1:00 自动删除用户起始目录中的所有日志文件。由于该命令项不重定向输出，因此将重定向字符添加到 *.log 之后的命令行中。这样可以确保正常执行命令。

```
# This command helps clean up user accounts.
1 0 * * 0 rm /home/jones/*.log > /dev/null 2>&1
```

▼ 如何验证 crontab 文件是否存在

- 要验证用户的 crontab 文件是否存在，请在 /var/spool/cron/crontabs 目录中使用 ls -l 命令。例如，以下输出说明用户 jones 和 smith 的 crontab 文件存在。

```
$ ls -l /var/spool/cron/crontabs
-rw-r--r-- 1 root sys 190 Feb 26 16:23 adm
-rw----- 1 root staff 225 Mar 1 9:19 jones
-rw-r--r-- 1 root root 1063 Feb 26 16:23 lp
-rw-r--r-- 1 root sys 441 Feb 26 16:25 root
-rw----- 1 root staff 60 Mar 1 9:15 smith
-rw-r--r-- 1 root sys 308 Feb 26 16:23 sys
```

使用 `crontab -l` 命令验证用户的 `crontab` 文件的内容，如第 106 页中的“如何显示 `crontab` 文件”中所述。

显示 crontab 文件

`crontab -l` 命令显示 `crontab` 文件内容的方式与 `cat` 命令显示其他类型文件内容的方式非常相似。无需转到 `/var/spool/cron/crontabs` 目录（`crontab` 文件所在的目录），便可使用此命令。

缺省情况下，`crontab -l` 命令显示您自己的 `crontab` 文件。要显示属于其他用户的 `crontab` 文件，您必须是超级用户。

▼ 如何显示 crontab 文件

开始之前 成为超级用户或同等角色，以显示属于 `root` 或其他用户的 `crontab` 文件。

无需成为超级用户或承担等效角色，即可显示您自己的 `crontab` 文件。

- 显示 `crontab` 文件。

```
$ crontab -l [username]
```

其中，`username` 指定了要显示其 `crontab` 文件的用户帐户的名称。显示其他用户的 `crontab` 文件需要超级用户特权。



注意 – 如果意外键入了无选项的 `crontab` 命令，请按下编辑器的中断字符。使用此字符将退出而不会保存更改。如果保存了更改并退出文件，现有的 `crontab` 文件将被空文件覆盖。

示例 8-2 显示 crontab 文件

此示例说明如何使用 `crontab -l` 命令来显示用户的缺省 `crontab` 文件的内容。

```
$ crontab -l
13 13 * * * chmod g+w /home1/documents/*.book > /dev/null 2>&1
```

示例 8-3 显示缺省的 root crontab 文件

此示例说明如何显示缺省的 `root` `crontab` 文件。

```
$ suPassword:
Oracle Corporation   SunOS 5.10           Generic Patch      January 2005
# crontab -l
#ident  "@(#)root      1.19    98/07/06 SMI"    /* SVr4.0 1.1.3.1    */
```

```
#
# The root crontab should be used to perform accounting data collection.
#
#
10 3 * * * /usr/sbin/logadm
15 3 * * 0 /usr/lib/fs/nfs/nfsfind
30 3 * * * [ -x /usr/lib/gss/gsscred_clean ] && /usr/lib/gss/gsscred_clean
#10 3 * * * /usr/lib/krb5/kprop_script ___slave_kdcs___
```

示例 8-4 显示其他用户的 crontab 文件

此示例说明如何显示属于其他用户的 crontab 文件。

```
$ su
Password:
Oracle Corporation   SunOS 5.10       Generic Patch   January 2005
# crontab -l jones
13 13 * * * cp /home/jones/work_files /usr/backup/. > /dev/null 2>&1
```

删除 crontab 文件

缺省情况下，设置了 crontab 文件保护，以防止使用 rm 命令意外删除 crontab 文件。请改用 crontab -r 命令删除 crontab 文件。

缺省情况下，crontab -r 命令会删除您自己的 crontab 文件。

无需转到 /var/spool/cron/crontabs 目录（crontab 文件所在的目录），便可使用此命令。

▼ 如何删除 crontab 文件

开始之前 成为超级用户或同等角色，以删除属于 root 或其他用户的 crontab 文件。角色包含授权和具有特权的命令。

无需成为超级用户或承担等效角色，即可删除您自己的 crontab 文件。

1 删除 crontab 文件。

```
$ crontab -r [username]
```

其中 *username* 指定要为其删除 crontab 文件的用户帐户的名称。为其他用户删除 crontab 文件需要超级用户特权。



注意 – 如果意外键入了无选项的 `crontab` 命令，请按下编辑器的中断字符。使用此字符将退出而不会保存更改。如果保存了更改并退出文件，现有的 `crontab` 文件将被空文件覆盖。

2 验证是否已删除 `crontab` 文件。

```
# ls /var/spool/cron/crontabs
```

示例 8-5 删除 `crontab` 文件

以下示例说明用户 `smith` 如何使用 `crontab -r` 命令删除他的 `crontab` 文件。

```
$ ls /var/spool/cron/crontabs
adm    jones    lp       root     smith    sys      uucp
$ crontab -r
$ ls /var/spool/cron/crontabs
adm    jones    lp       root     sys      uucp
```

控制对 `crontab` 命令的访问

可以使用 `/etc/cron.d` 目录中的以下两个文件来控制对 `crontab` 命令的访问：`cron.deny` 和 `cron.allow`。这些文件只允许指定的用户执行 `crontab` 命令任务，例如创建、编辑、显示或删除自己的 `crontab` 文件。

`cron.deny` 和 `cron.allow` 文件包含用户名的列表，每行一个用户名。

这些访问控制文件按以下方式协同工作：

- 如果存在 `cron.allow`，则只有此文件中列出的用户可以创建、编辑、显示或删除 `crontab` 文件。
- 如果不存在 `cron.allow`，则所有用户都可以提交 `crontab` 文件（`cron.deny` 中列出的用户除外）。
- 如果 `cron.allow` 和 `cron.deny` 都不存在，则运行 `crontab` 命令需要超级用户特权。

编辑或创建 `cron.deny` 和 `cron.allow` 文件需要超级用户特权。

在 SunOS 软件安装期间创建的 `cron.deny` 文件包含以下用户名：

```
$ cat /etc/cron.d/cron.deny
daemon
bin
smtp
nuucp
listen
nobody
noaccess
```

缺省 `cron.deny` 文件中的用户名都不能访问 `crontab` 命令。您可以编辑此文件，以添加被拒绝访问 `crontab` 命令的其他用户名。

未提供缺省的 `cron.allow` 文件。因此，在安装 Oracle Solaris 软件后，所有用户（缺省 `cron.deny` 文件中列出的用户除外）都可以访问 `crontab` 命令。如果创建 `cron.allow` 文件，则只有这些用户可以访问 `crontab` 命令。

▼ 如何拒绝 `crontab` 命令访问

- 1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《[系统管理指南：安全性服务](#)》中的“配置 RBAC（任务列表）”。

- 2 编辑 `/etc/cron.d/cron.deny` 文件并添加用户名，每个用户占据一行。将拒绝访问 `crontab` 命令的用户包括在内。

```
daemon
bin
smtp
nuucp
listen
nobody
noaccess
username1
username2
username3
.
.
.
```

- 3 验证 `/etc/cron.d/cron.deny` 文件是否包含新项。

```
# cat /etc/cron.d/cron.deny
daemon
bin
nuucp
listen
nobody
noaccess
```

▼ 如何将 `crontab` 命令访问限制于指定的用户

- 1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《[系统管理指南：安全性服务](#)》中的“配置 RBAC（任务列表）”。

- 2 创建 `/etc/cron.d/cron.allow` 文件。

3 将 root 用户名添加到 cron.allow 文件中。

如果未将 root 添加到该文件中，则会拒绝超级用户访问 crontab 命令。

4 添加用户名，每行一个用户名。

将允许使用 crontab 命令的用户包括在内。

```
root
username1
username2
username3
.
.
.
```

示例 8-6 将 crontab 命令访问限制于指定的用户

以下示例显示一个 cron.deny 文件，该文件用于禁止用户名 jones、temp 和 visitor 访问 crontab 命令。

```
$ cat /etc/cron.d/cron.deny
daemon
bin
smtp
nuucp
listen
nobody
noaccess
jones
temp
visitor
```

以下示例显示一个 cron.allow 文件。用户 root、jones、lp 和 smith 是仅有的可以访问 crontab 命令的用户。

```
$ cat /etc/cron.d/cron.allow
root
jones
lp
smith
```

如何验证受限的 crontab 命令访问

要验证特定用户是否可以访问 crontab 命令，请在使用该用户帐户登录后执行 crontab -l 命令。

```
$ crontab -l
```

如果用户可以访问 crontab 命令并已创建 crontab 文件，则会显示该文件。否则，如果用户可以访问 crontab 命令但不存在 crontab 文件，则会显示与以下消息类似的消息：

```
crontab: can't open your crontab file
```

此用户已列在 `cron.allow` 文件中（如果存在该文件），或者该用户未列在 `cron.deny` 文件中。

如果用户不能访问 `crontab` 命令，则无论是否存在以前的 `crontab` 文件，都会显示以下消息：

```
crontab: you are not authorized to use cron. Sorry.
```

此消息表明，该用户未列在 `cron.allow` 文件（如果该文件存在）中，或者该用户已列在 `cron.deny` 文件中。

使用 at 命令（任务列表）

任务	说明	参考
创建 at 作业。	使用 <code>at</code> 命令执行以下操作： <ul style="list-style-type: none"> 从命令行启动 <code>at</code> 实用程序。 键入要执行的命令或脚本，每行一个。 退出 <code>at</code> 实用程序并保存作业。 	第 112 页中的“如何创建 at 作业”
显示 at 队列。	使用 <code>atq</code> 命令显示 at 队列。	第 113 页中的“如何显示 at 队列”
验证 at 作业。	使用 <code>atq</code> 命令确认属于特定用户的 at 作业已提交至队列。	第 114 页中的“如何验证 at 作业”
显示 at 作业。	使用 <code>at -l [job-id]</code> 命令显示已提交至队列的 at 作业。	第 114 页中的“如何显示 at 作业”
删除 at 作业。	使用 <code>at -r [job-id]</code> 命令从队列中删除 at 作业。	第 114 页中的“如何删除 at 作业”
拒绝访问 at 命令。	要拒绝用户访问 <code>at</code> 命令，请编辑 <code>/etc/cron.d/at.deny</code> 文件。	第 115 页中的“如何拒绝对 at 命令的访问”

调度单个系统任务 (at)

以下各节介绍如何使用 `at` 命令来执行下列任务：

- 调度作业（命令和脚本）供以后执行
- 如何显示和删除这些作业
- 如何控制对 `at` 命令的访问

缺省情况下，用户可以创建、显示和删除自己的 `at` 作业文件。要访问属于 `root` 或其他用户的 `at` 文件，您必须具有超级用户特权。

提交 `at` 作业时，会为该作业指定作业标识号和 `.a` 扩展名。此指定将成为该作业的文件名，以及其队列编号。

at 命令的说明

提交 `at` 作业文件需执行以下步骤：

1. 调用 `at` 实用程序并指定命令执行时间。
2. 键入以后要执行的命令或脚本。

注 - 如果此命令或脚本的输出很重要，请确保将输出定向到一个文件中，以便以后检查。

例如，以下 `at` 作业将在 7 月的最后一天接近午夜时删除用户帐户 `smith` 的 `core` 文件。

```
$ at 11:45pm July 31
at> rm /home/smith/*core*
at> Press Control-d
commands will be executed using /bin/csh
job 933486300.a at Tue Jul 31 23:45:00 2004
```

控制对 at 命令的访问

您可以设置一个文件来控制对 `at` 命令的访问，只允许指定的用户创建、删除或显示有关 `at` 作业的队列信息。控制对 `at` 命令的访问的文件 `/etc/cron.d/at.deny` 由用户名列表构成，每个用户名占据一行。此文件中列出的用户不能访问 `at` 命令。

在 SunOS 软件安装期间创建的 `at.deny` 文件包含以下用户名：

```
daemon
bin
smtp
nuucp
listen
nobody
noaccess
```

使用超级用户特权，您可以编辑 `at.deny` 文件，以添加要限制其对 `at` 命令访问的其他用户名。

▼ 如何创建 at 作业

- 1 启动 `at` 实用程序，指定所需的作业执行时间。

```
$ at [-m] time [date]
```


- m 在作业完成后发送邮件。
 - time 指定要调度作业的小时。如果不按 24 小时制指定小时，请添加 am 或 pm。可接受的关键字包括 midnight、noon 和 now。分钟是可选的选项。
 - date 指定月份的前三个或更多字母、一周中的某日或关键字 today 或 tomorrow。
- 2 在 at 提示符下，键入要执行的命令或脚本，每行一个。
通过在每行结尾处按回车键，可以键入多个命令。
 - 3 按 Ctrl-D 组合键，以退出 at 实用程序并保存 at 作业。
at 作业将被指定一个队列编号，它也是该作业的文件名。退出 at 实用程序时将显示该编号。

示例 8-7 创建 at 作业

以下示例显示了用户 jones 创建的 at 作业，该作业用于在下午 7:30 删除其备份文件。由于她使用了 -m 选项，因此她会在该作业完成后收到电子邮件。

```
$ at -m 1930
at> rm /home/jones/*.backup
at> Press Control-D
job 897355800.a at Thu Jul 12 19:30:00 2004
```

她收到一封确认已执行 at 作业的电子邮件。

```
Your "at" job "rm /home/jones/*.backup"
completed.
```

以下示例说明 jones 如何调度在星期六凌晨 4:00 执行的大型 at 作业。该作业输出被定向到名为 big.file 的文件中。

```
$ at 4 am Saturday
at> sort -r /usr/dict/words > /export/home/jones/big.file
```

▼ 如何显示 at 队列

- 要检查在 at 队列中等待的作业，请使用 atq 命令。

```
$ atq
```

此命令可以显示您已创建的 at 作业的状态信息。

▼ 如何验证 at 作业

- 要验证您是否已创建了 **at** 作业，请使用 **atq** 命令。在以下示例中，**atq** 命令确认已将属于 **jones** 的 **at** 作业提交至队列。

```
$ atq
Rank      Execution Date      Owner      Job      Queue      Job Name
1st       Jul 12, 2004 19:30      jones      897355800.a      a      stdin
2nd       Jul 14, 2004 23:45      jones      897543900.a      a      stdin
3rd       Jul 17, 2004 04:00      jones      897732000.a      a      stdin
```

▼ 如何显示 at 作业

- 要显示有关 **at** 作业的执行时间信息，请使用 **at -l** 命令。

```
$ at -l [job-id]
```

其中，**-l job-id** 选项表示要显示其状态的作业的标识号。

示例 8-8 显示 at 作业

以下示例显示 **at -l** 命令的输出，该输出提供有关用户已提交的所有作业的状态信息。

```
$ at -l
897543900.a      Sat Jul 14 23:45:00 2004
897355800.a      Thu Jul 12 19:30:00 2004
897732000.a      Tue Jul 17 04:00:00 2004
```

以下示例显示使用 **at -l** 命令指定单个作业时所显示的输出。

```
$ at -l 897732000.a
897732000.a      Tue Jul 17 04:00:00 2004
```

▼ 如何删除 at 作业

开始之前 成为超级用户或同等角色，以删除属于 **root** 或其他用户的 **at** 作业。角色包含授权和具有特权的命令。

无需成为超级用户或承担等效角色，便可删除自己的 **at** 作业。

- 1 在作业执行之前从队列中删除 **at** 作业。

```
$ at -r [job-id]
```

其中，**-r job-id** 选项指定要删除的作业的标识号。

2 使用 `at -l` (或 `atq`) 命令, 验证是否已删除 `at` 作业。

`at -l` 命令显示 `at` 队列中剩余的作业。不应显示已指定标识号的作业。

```
$ at -l [job-id]
```

示例 8-9 删除 `at` 作业

在以下示例中, 用户要删除计划在 7 月 17 日凌晨 4 点执行的 `at` 作业。首先, 该用户显示 `at` 队列, 以找到作业标识号。然后, 用户从 `at` 队列中删除此作业。最后, 该用户验证是否已从队列中删除此作业。

```
$ at -l
897543900.a    Sat Jul 14 23:45:00 2003
897355800.a    Thu Jul 12 19:30:00 2003
897732000.a    Tue Jul 17 04:00:00 2003
$ at -r 897732000.a
$ at -l 897732000.a
at: 858142000.a: No such file or directory
```

▼ 如何拒绝对 `at` 命令的访问

1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息, 请参见《[系统管理指南: 安全性服务](#)》中的“配置 RBAC (任务列表)”。

2 编辑 `/etc/cron.d/at.deny` 文件并添加要禁止其使用 `at` 命令的用户名, 每行一个用户名。

```
daemon
bin
smtp
nuucp
listen
nobody
noaccess
username1
username2
username3
.
.
.
```

示例 8-10 拒绝 `at` 访问

以下示例显示了一个 `at.deny` 文件, 该文件已被编辑过, 因此用户 `smith` 和 `jones` 无法访问 `at` 命令。

```
$ cat at.deny
daemon
bin
smtp
nuucp
listen
nobody
noaccess
jones
smith
```

▼ 如何验证 at 命令访问已被拒绝

- 要验证是否已将用户名正确添加到 `/etc/cron.d/at.deny` 文件，请在以该用户身份登录后使用 `at -l` 命令。如果用户 `smith` 无法访问 `at` 命令，将显示以下消息：

```
# su smith
Password:
$ at -l
at: you are not authorized to use at. Sorry.
```

类似地，如果该用户尝试提交 `at` 作业，则将显示以下消息：

```
$ at 2:30pm
at: you are not authorized to use at. Sorry.
```

此消息确认该用户已列在 `at.deny` 文件中。

如果允许访问 `at` 命令，则 `at -l` 命令不会返回任何内容。

管理系统记帐（任务）

本章介绍如何设置和维护系统记帐。

以下是本章中概述信息的列表。

- [第 118 页](#)中的“什么是系统记帐？”
- [第 122 页](#)中的“设置系统记帐”

有关使用扩展记帐的信息，请参见《[系统管理指南：Oracle Solaris Containers—资源管理和 Oracle Solaris Zones](#)》中的第 4 章“扩展记帐（概述）”。

有关与系统记帐相关的逐步过程的信息，请参见第 122 页中的“系统记帐（任务列表）”。

有关各种系统记帐报告的参考信息，请参见第 10 章，[系统记帐（参考）](#)。

系统记帐方面的新增功能

本节介绍此 Oracle Solaris 发行版中用于系统记帐的新增功能或已更改的功能。有关新增功能的完整列表以及 Oracle Solaris 发行版的说明，请参见《[Oracle Solaris 10 8/11 新增功能](#)》。

Oracle Solaris 进程记帐和统计信息改进

Oracle Solaris 10：对平均负荷的内部实现 `cpu usr/sys/idle` 以及记帐功能进行了更改。微状态记帐取代了原有的记帐机制，并在缺省情况下一直启用。因此，您可能会注意到进程使用情况和计时统计信息与以往稍有不同。

改用微状态记帐，可针对用户进程及其在各个状态上所花费的时间来提供更为准确的信息。此外，这些信息还用于从 `/proc` 文件系统中生成更精确的负荷平均值和统计信息。有关更多信息，请参见 [proc\(4\)](#) 手册页。

什么是系统记帐？

Oracle Solaris OS 中的系统记帐软件是一组程序，用于收集和记录有关用户连接时间、进程占用的 CPU 时间和磁盘使用情况的数据。收集到此数据后，即可生成报告并针对系统使用情况收取费用。

可以使用每日或每月系统记帐。或者，也可以按用户跟踪磁盘使用情况。

使用记帐程序可以执行以下任务：

- 监视系统使用情况
- 查找并更正性能问题
- 维护系统安全

在设置系统记帐程序后，这些程序大部分时间会自行运行。

系统记帐的工作原理

设置自动记帐时，先要将记帐启动脚本放入根的 `crontab` 文件中。然后，`cron` 命令即可自动启动该记帐启动脚本。

以下概述介绍了系统记帐过程。

1. 在系统启动和关闭期间，会在记帐文件中收集有关系统使用（如用户登录、运行进程和数据存储）的原始数据。
2. `/usr/lib/acct/runacct` 脚本会定期（通常一天一次）处理各种记帐文件并生成累积概要文件和每日记帐报告。然后，`/usr/lib/acct/prdaily` 脚本将打印每日报告。
有关 `runacct` 脚本的更多信息，请参见第 131 页中的“`runacct` 脚本”。
3. 可以通过执行 `monacct` 脚本每月处理并打印一次累积 `runacct` 摘要文件。`monacct` 脚本生成的摘要报告提供了一种按月或按其他财务周期对用户计费的有效方法。

系统记帐组件

系统记帐软件提供了可将数据组织为摘要文件和报告的 C 语言程序和 Shell 脚本。这些程序驻留在 `/usr/lib/acct` 目录中。记帐报告驻留在 `/var/adm/acct` 目录中。

每日记帐有助于执行四种类型的审计：

- 连接记帐
- 进程记帐
- 磁盘记帐
- 费用计算

连接记帐

通过连接记帐可以确定以下信息：

- 用户已登录的时间
- `tty` 线的使用情况
- 系统重新引导的次数
- 已关闭和打开记帐软件的次数

要提供有关连接会话的记帐信息，系统需要存储以下数据：

- 时间调整的记录
- 引导时间
- 关闭和打开记帐软件的次数
- 运行级的更改
- 用户进程（`login` 进程和 `init` 进程）的创建
- 进程的终止

这些记录是从系统程序（如 `date`、`init`、`login`、`ttymon` 和 `acctwtmp`）的输出中生成的。这些记录存储在 `/var/adm/wtmpx` 文件中。

`wtmpx` 文件中的项可以包含以下信息：

- 登录名称
- 设备名称
- 进程 ID
- 项类型
- 指明项创建时间的时间戳

进程记帐

通过进程记帐可以跟踪有关系统中运行的每个进程的以下数据：

- 使用进程的用户的用户 ID 和组 ID
- 进程的启动时间和已用时间
- 进程的 CPU 时间（用户时间和系统时间）
- 进程使用的内存量
- 进程运行的命令
- 控制进程的 `tty`

每次进程终止时，`exit` 程序便会收集此信息并将其写入 `/var/adm/pacct` 文件。

磁盘记帐

通过磁盘记帐可以收集有关每个用户在磁盘中的文件的以下数据并设置数据格式：

- 用户的用户名和用户 ID
- 用户的文件占用的块数

/usr/lib/acct/dodisk shell 脚本以一定的时间间隔收集上述数据，该时间间隔是由您向 /var/spool/cron/crontabs/root 文件中添加的项确定的。反过来，dodisk 脚本会调用 acctdisk 和 acctdusg 命令。这些命令按登录名称收集磁盘使用情况的信息。



注意 – 通过运行 dodisk 脚本收集的信息存储在 /var/adm/acct/nite/disktacct 文件中。下次运行 dodisk 脚本时将覆盖此信息。因此，请避免在同一天中两次运行 dodisk 脚本。

对于随机写入的文件，acctdusg 命令计算的磁盘占用量可能比实际多，这会在文件中形成空洞。出现这种问题是由于 acctdusg 命令在确定文件大小时未读取文件的间接块。更确切地说，acctdusg 文件是通过检查文件 inode 中的当前文件大小值来确定文件大小的。

费用计算

chargefee 实用程序在 /var/adm/fee 文件中存储为用户提供的特殊服务的费用。例如，特殊服务可以是文件恢复。文件中的每项都由用户登录名、用户 ID 和费用组成。runacct 每天都会检查此文件，并将新项合并到记帐记录中。有关运行 chargefee 脚本以对用户计费的说明，请参见第 125 页中的“如何对用户计费”。

每日记帐的工作原理

以下是每日记帐的工作原理的分步概要说明：

1. 将系统切换到多用户模式时，执行 /usr/lib/acct/startup 程序。startup 程序会执行多个调用每日记帐的其他程序。
2. acctwtmp 程序向 /var/adm/wtmpx 文件中添加一条“引导”记录。在此记录中，系统名显示为 wtmpx 记录中的用户名。下表汇总了原始记帐数据的收集方式及存储位置。

/var/adm 中的文件	存储的信息	写入程序	格式
wtmpx	连接会话	login、init	二进制
	更改	date	二进制
	重新引导	acctwtmp	二进制
	关闭	shutacct	二进制
pacctn	进程	Kernel（当进程结束时）	二进制
		turnacct switch（当原有文件达到 500 块时创建一个新文件）	二进制
fee	特殊费用	chargefee	ASCII

/var/adm中的文件	存储的信息	写入程序	格式
acct/nite/diskacct	使用的磁盘空间	dodisk	二进制

- 3. 使用 -on 选项调用的 turnacct 脚本开始进程记帐。具体来讲，就是 turnacct 脚本使用 /var/adm/pacct 参数执行 accton 程序。
- 4. 删除 Shell 脚本会“清除”由 runacct 脚本留在 sum 目录中的已保存的 pacct 和 wtmpx 文件。
- 5. login 和 init 程序通过向 /var/adm/wtmpx 文件中写入记录来记录连接会话。日期更改（使用带参数的日期）也会写入 /var/adm/wtmpx 文件中。使用 acctwtmp 命令进行重新引导和关闭也会记录到 /var/adm/wtmpx 文件中。
- 6. 进程结束时，内核会使用 acct.h 格式在 /var/adm/pacct 文件中为每个进程写入一条记录。

cron 命令每小时执行一次 ckpacct 脚本，以检查 /var/adm/pacct 文件的大小。如果文件大小超出了 500 块（缺省值），则会执行 turnacct switch 命令。（程序将 pacct 文件移至 pacctn 文件并创建一个新文件。）如果由于处理这些记录时出现故障而尝试重新启动 runacct 脚本，则有多个小型 pacct 文件的优点是显而易见的。

- 7. cron 命令每晚都会执行 runacct 脚本。runacct 脚本会对记帐文件进行处理，从而按用户名生成命令概要和使用情况概要。将处理以下记帐文件：
/var/adm/pacctn、/var/adm/wtmpx、/var/adm/fee 和
/var/adm/acct/nite/diskacct。
- 8. runacct 脚本每天执行一次 /usr/lib/acct/prdaily 脚本，以便将每日记帐信息写入 /var/adm/acct/sum/rprtMMDD 文件中。
- 9. monacct 脚本应每月执行一次（或以您确定的时间间隔执行，例如在每个财务周期结束时执行）。monacct 脚本会基于 sum 目录中存储的数据创建一个报告，该目录每天通过 runacct 脚本更新一次。创建报告后，monacct 脚本将“清除”sum 目录，以便为新的 runacct 数据准备目录文件。

系统关闭时执行的操作

如果使用 shutdown 命令关闭系统，则会自动执行 shutacct 脚本。shutacct 脚本会向 /var/adm/wtmpx 文件中写入一条原因记录并关闭进程记帐。

系统记帐（任务列表）

任务	说明	参考
设置系统记帐。	通过执行以下任务来设置系统记帐： <ul style="list-style-type: none">■ 创建 /etc/rc0.d/K22acct 和 /etc/rc2.d/S22acct 文件。■ 修改 /var/spool/cron/crontabs/adm 和 /var/spool/cron/crontabs/root crontab 文件。	第 123 页中的“如何设置系统记帐”
对用户计费。	运行 <code>/usr/lib/acct/chargefee username amount</code> 命令。	第 125 页中的“如何对用户计费”
修复损坏的 wtmpx 文件。	将 wtmpx 文件从二进制转换为 ASCII 格式。	第 126 页中的“如何修复损坏的 wtmpx 文件”
修复 tacct 错误。	运行 prtacct 脚本以检查 /var/adm/acct/sum/tacctprev 文件。然后，修补最新的 /var/adm/acct/sum/tacctMMDD 文件。您将需要重新创建 /var/adm/acct/sum/tacct 文件。	第 127 页中的“如何修复 tacct 错误”
重新启动 runacct 脚本。	删除 lastdate 文件以及任何锁定文件。然后，手动重新启动 runacct 脚本。	第 128 页中的“如何重新启动 runacct 脚本”
临时禁用系统记帐。	编辑 adm crontab 文件，以阻止 ckpacct、runacct 和 monacct 程序运行。	第 128 页中的“如何暂时停止系统记帐”
永久禁用系统记帐。	删除 adm 和 crontab 文件中用于 ckpacct、runacct 和 monacct 程序的项。	第 129 页中的“如何永久禁用系统记帐”

设置系统记帐

可将系统记帐设置为在系统处于多用户模式时运行（运行级 2）。通常，此任务涉及以下步骤：

1. 创建 /etc/rc0.d/K22acct 和 /etc/rc2.d/S22acct 启动脚本
2. 修改 /var/spool/cron/crontabs/adm 和 /var/spool/cron/crontabs/root crontab 文件

下表介绍了缺省记帐脚本。

表 9-1 缺省记帐脚本

用途	记帐脚本	手册页	运行频率
检查 /usr/adm/pacct 日志文件的大小并确保该文件不会太大。	ckpacct	acctsh(1M)	定期
处理连接、磁盘和费用记帐的信息。如果不想执行某些记帐功能，则可从此脚本中删除相应命令。	runacct	runacct(1M)	每日
每月生成一次财务记帐摘要报告。您可以确定此脚本的运行频率。如果不想使用某些记帐功能，则可从此脚本中删除相应命令。	monacct	acctsh(1M)	按财务周期

可以选择缺省情况下运行的记帐脚本。将这些项添加到 `crontab` 文件后，系统记帐应自动运行。

▼ 如何设置系统记帐

- 1 成为超级用户或承担等效角色。
角色包含授权和具有特权的命令。有关角色的更多信息，请参见《[系统管理指南：安全性服务](#)》中的“[配置 RBAC（任务列表）](#)”。
- 2 如有必要，请使用 `pkgadd` 命令在系统中安装 `SUNWaccr` 和 `SUNWaccu` 软件包。
- 3 安装 `/etc/init.d/acct` 作为运行级 2 的启动脚本。

```
# ln /etc/init.d/acct /etc/rc2.d/S22acct
```
- 4 安装 `/etc/init.d/acct` 作为运行级 0 的停止脚本。

```
# ln /etc/init.d/acct /etc/rc0.d/K22acct
```
- 5 向 `adm crontab` 文件中添加以下行，以自动启动 `ckpacct`、`runacct` 和 `monacct` 脚本。

```
# EDITOR=vi; export EDITOR
# crontab -e adm
0 * * * * /usr/lib/acct/ckpacct
30 2 * * * /usr/lib/acct/runacct 2> /var/adm/acct/nite/fd2log
30 7 1 * * /usr/lib/acct/monacct
```

- 6 向 `root crontab` 文件中添加以下行，以自动启动 `dodisk` 脚本。

```
# crontab -e
30 22 * * 4 /usr/lib/acct/dodisk
```

- 7 编辑 `/etc/acct/holidays` 以包括国定假日和当地假日。
有关更多信息，请参见 [holidays\(4\)](#) 手册页及后面的示例。

- 8 重新引导系统或键入以下内容手动启动系统记帐：

```
# /etc/init.d/acct start
```

示例9-1 设置记帐 (adm crontab)

这一修改的 `adm crontab` 包含 `ckpacct`、`runacct` 和 `monacct` 脚本的项。

```
#ident "@(#)adm      1.5      92/07/14 SMI"    /* SVr4.0 1.2 */
#
# The adm crontab file should contain startup of performance
# collection if the profiling and performance feature has been
# installed.
0 * * * * /usr/lib/acct/ckpacct
30 2 * * * /usr/lib/acct/runacct 2> /var/adm/acct/nite/fd2log
30 7 1 * * /usr/lib/acct/monacct
```

示例9-2 设置记帐 (root crontab)

这一修改的 `root crontab` 包含 `dodisk` 程序的项。

```
#ident "@(#)root      1.19      98/07/06 SMI"    /* SVr4.0 1.1.3.1 */
#
# The root crontab should be used to perform accounting data collection.
#
#
10 3 * * * /usr/sbin/logadm
15 3 * * 0 /usr/lib/fs/nfs/nfsfind
30 3 * * * [ -x /usr/lib/gss/gsscred_clean ] && /usr/lib/gss/gsscred_clean
30 22 * * 4 /usr/lib/acct/dodisk
```

示例9-3 设置记帐 (/etc/acct/holidays)

以下示例显示一个 `/etc/acct/holidays` 文件样例。

```
* @(#)holidays      January 1, 2004
*
* Prime/Nonprime Table for UNIX Accounting System
*
* Curr      Prime      Non-Prime
* Year      Start      Start
*
*      2004      0800      1800
*
```

```
* only the first column (month/day) is significant.
*
* month/day      Company
*              Holiday
*
1/1             New Years Day
7/4             Indep. Day
12/25           Christmas
```

对用户计费

如果您根据需求提供特别用户服务，则需要运行 `chargefee` 实用程序对用户计费。特殊服务包括恢复文件或远程打印。`chargefee` 实用程序会在 `/var/adm/fee` 文件中记录费用。每次执行 `runacct` 实用程序时，都会将新项合并到总记帐记录中。

有关更多信息，请参见 [acctsh\(1M\)](#) 手册页。

▼ 如何对用户计费

1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《[系统管理指南：安全性服务](#)》中的“[配置 RBAC（任务列表）](#)”。

2 向用户收取特殊服务的费用。

```
# /usr/lib/acct/chargefee username amount
```

username 指定要对其计费的用户帐户。

amount 指定要对用户计费的单位数量。该值是设置的用于根据打印或恢复文件等任务向用户收费的任意单位。必须编写一个脚本，以调用 `chargefee` 实用程序并向用户收取特定任务的费用。

示例 9-4 对用户计费

在以下示例中，向用户 `print_customer` 收取了 10 个单位的费用。

```
# /usr/lib/acct/chargefee print_customer 10
```

维护记帐信息

本节介绍如何修复损坏的系统记帐文件以及如何重新启动 `runacct` 脚本。

修复损坏的文件并更正 `wtmpx` 错误

遗憾的是，系统记帐并不十分安全。有时候，文件会损坏或丢失。有些文件可以忽略，也可以从备份中恢复。但是，有些文件则必须修复，才能维护系统记帐的完整性。

`wtmpx` 文件可能会引发每日系统记帐操作中的大多数问题。手动更改日期并且系统处于多用户模式时，会向 `/var/adm/wtmpx` 文件中写入一组日期更改记录。`wtmpfix` 实用程序旨在出现日期更改时调整 `wtmp` 记录中的时间戳。但是，部分日期更改和重新引导的组合会跳过 `wtmpfix` 实用程序并导致 `acctcon` 程序失败。

▼ 如何修复损坏的 `wtmpx` 文件

- 1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的“配置 RBAC（任务列表）”。

- 2 转到 `/var/adm` 目录。
- 3 将 `wtmpx` 文件从二进制格式转换为 ASCII 格式。
- 4 编辑 `wtmpx.ascii` 文件以删除损坏的记录。
- 5 将 `wtmpx.ascii` 文件转换回二进制文件。

```
# /usr/lib/acct/fwtmp -ic < wtmpx.ascii > wtmpx
```

有关更多信息，请参见 `fwtmp(1M)` 手册页。

更正 `tacct` 错误

如果要向用户收取系统资源费用，则 `/var/adm/acct/sum/tacct` 文件的完整性极为重要。有时候，会出现异常的 `tacct` 记录，其中包括负数、重复的用户 ID 或用户 ID 65535。首先，使用 `prtacct` 脚本列显 `/var/adm/acct/sum/tacctprev` 文件，以检查该文件。如果内容没有问题，则修补最新的 `/var/adm/acct/sum/tacct` `MMDD` 文件。然后，重新创建 `/var/adm/acct/sum/tacct` 文件。以下步骤概括了一个简单的修补过程。

▼ 如何修复 tacct 错误

- 1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的“配置 RBAC（任务列表）”。

- 2 转到 `/var/adm/acct/sum` 目录。

- 3 将 `tacctMMDD` 文件从二进制格式转换为 ASCII 格式。

```
# /usr/lib/acct/acctmerg -v < tacctMMDD > xtacct
```

`MMDD` 是一对表示月份和日期的两位数。

- 4 编辑 `xtacct` 文件，删除损坏的记录并将重复记录写入另一个文件。

- 5 将 `xtacct` 文件从 ASCII 格式转换为二进制格式。

```
# /usr/lib/acct/acctmerg -i < xtacct > tacctMMDD
```

- 6 将文件 `tacctprev` 和 `tacct.MMDD` 合并到 `tacct` 文件中。

```
# /usr/lib/acct/acctmerg < tacctprev tacctMMDD > tacct
```

重新启动 runacct 脚本

`runacct` 脚本失败的原因可能有多种。

以下是最常见的原因：

- 系统崩溃
- `/var` 目录空间不足
- `wtmpx` 文件损坏

如果存在 `active.MMDD` 文件，请先检查该文件中是否有错误消息。如果存在 `active` 和 `lock` 文件，请检查 `fd2log` 文件中是否有任何相关消息。

如果不带任何参数运行 `runacct` 脚本，则该脚本会假定这一次调用是当天的第一次调用。如果 `runacct` 脚本正在重新启动并指定 `runacct` 脚本重新运行记帐的月份和日期，则参数 `MMDD` 是必需的。进行处理的入口点基于 `statefile` 文件的内容。要覆盖 `statefile` 文件，请在命令中包括所需的状态。有关可用状态的说明，请参见 [runacct\(1M\)](#) 手册页。



注意 – 手动运行 `runacct` 程序时，请确保以用户身份 `adm` 运行该程序。

▼ 如何重新启动 runacct 脚本

- 1 转到 `/var/adm/acct/nite` 目录。

```
$ cd /var/adm/acct/nite
```

- 2 删除 `lastdate` 文件和所有 `lock*` 文件（如果有）。

```
$ rm lastdate lock*
```

`lastdate` 文件包含上次运行 `runacct` 程序的日期。在下一步中重新启动 `runacct` 脚本将重新创建此文件。

- 3 重新启动 `runacct` 脚本。

```
$ /usr/lib/acct/runacct MMDD [state] 2> /var/adm/acct/nite/fd2log &
```

`MMDD` 由两位数指定的月份和日期。

`state` 指定 `runacct` 脚本处理开始的状态或起始点。

停止和禁用系统记帐

可以暂时停止或永久禁用系统记帐。

▼ 如何暂时停止系统记帐

- 1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的“配置 RBAC（任务列表）”。

- 2 编辑 `adm crontab` 文件，通过注释掉相应行来停止运行 `ckpacct`、`runacct` 和 `monacct` 程序。

```
# EDITOR=vi; export EDITOR
# crontab -e adm
#0 * * * * /usr/lib/acct/ckpacct
#30 2 * * * /usr/lib/acct/runacct 2> /var/adm/acct/nite/fd2log
#30 7 1 * * /usr/lib/acct/monacct
```

- 3 编辑 `root crontab` 文件，通过注释掉相应行来停止运行 `dodisk` 程序。

```
# crontab -e
#30 22 * * 4 /usr/lib/acct/dodisk
```

- 4 停止系统记帐程序。

```
# /etc/init.d/acct stop
```


5 （可选的）从 `crontab` 文件中删除新增的注释符号。

6 重新启动系统记帐程序，以重新启用系统记帐。

```
# /etc/init.d/acct start
```

▼ 如何永久禁用系统记帐

1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《[系统管理指南：安全性服务](#)》中的“[配置 RBAC（任务列表）](#)”。

2 编辑 `adm crontab` 文件并删除 `ckpacct`、`runacct` 和 `monacct` 程序的项。

```
# EDITOR=vi; export EDITOR
# crontab -e adm
```

3 编辑 `root crontab` 文件并删除 `dodisk` 程序的项。

```
# crontab -e
```

4 删除运行级 2 的启动脚本。

```
# unlink /etc/rc2.d/S22acct
```

5 删除运行级 0 的停止脚本。

```
# unlink /etc/rc0.d/K22acct
```

6 停止系统记帐程序。

```
# /etc/init.d/acct stop
```


系统记帐（参考）

本章提供有关系统记帐的参考信息。

以下是本章中参考信息的列表。

- [第 131 页中的“runacct 脚本”](#)
- [第 133 页中的“每日记帐报告”](#)
- [第 139 页中的“系统记帐文件”](#)

有关系统记帐任务的更多信息，请参见[第 9 章，管理系统记帐（任务）](#)。

runacct 脚本

主要的每日记帐脚本 `runacct` 通常由 `cron` 命令在正常工作时间之外调用。`runacct` 脚本可以处理连接、费用、磁盘和进程记帐文件。此脚本还可准备每日摘要文件和累积摘要文件，以供 `prdaily` 和 `monacct` 脚本记帐。

`runacct` 脚本可在发生错误时防止文件受到损坏。

使用一系列保护机制来执行以下任务：

- 识别错误
- 提供智能诊断
- 以特定方式完成处理，从而以最少的介入重新启动 `runacct` 脚本

该脚本通过向 `active` 文件中写入描述性消息来记录其进度。除非另有说明，否则假定 `runacct` 脚本所使用的文件位于 `/var/adm/acct/nite` 目录中。`runacct` 脚本执行期间的所有诊断输出都写入 `fd2log` 文件中。

调用 `runacct` 脚本时，将会创建 `lock` 和 `lock1` 文件。这些文件可防止同时执行 `runacct` 脚本。如果调用 `runacct` 程序时已经存在上述文件，则该程序将列显错误消息。`lastdate` 文件中包含上次调用 `runacct` 脚本的月份和日期，以避免每天多次执行脚本。

如果 runacct 脚本检测到错误，则会发生以下情况：

- 向控制台写入一条消息
- 向 root 和 adm 发送电子邮件
- 可能删除锁定
- 保存诊断
- 执行结束

有关如何重新启动 runacct 脚本的说明，请参见第 128 页中的“[如何重新启动 runacct 脚本](#)”。

要允许重新启动 runacct 脚本，需要将处理细分为不同的可重复执行状态。statefile 文件用于跟踪完成的上一个状态。当每个状态完成时，都会更新 statefile 文件，以反映下一个状态。该状态的处理完成后，将读取 statefile 文件并处理下一个状态。当 runacct 脚本达到 CLEANUP 状态时，它将删除锁定并结束。将按下表所示来执行状态。

表 10-1 runacct 脚本的状态

状态	说明
SETUP	执行 turnacct switch 命令以创建新的 pacct 文件。/var/adm/pacct n 进程记帐文件（pacct 文件除外）被移至 /var/adm/Spacct n.MMDD 文件。/var/adm/wtmpx 文件被移至 /var/adm/acct/nite/wtmp.MMDD 文件（在结尾添加了当前时间记录）并创建一个新 /var/adm/wtmp 文件。closewtmp 和 utmp2wtmp 程序将向 wtmp.MMDD 文件和新的 wtmpx 文件中添加记录，以便 对当前登录的用户进行说明。
WTMPFIX	wtmpfix 程序检查 nite 目录中的 wtmp.MMDD 文件是否正确。由于某些日期更改会导致 acctcon 程序失败，所以 wtmpfix 程序会在有日期更改记录时尝试调整 wtmpx 文件中的时间戳。此程序还可删除 wtmpx 文件中的任何已损坏项。将 wtmp.MMDD 文件的已修复版本写入 tmpwtmp 文件。
CONNECT	acctcon 程序用于在文件 ctacct.MMDD 中记录连接记帐记录。这些记录使用 tacct.h 格式。此外，acctcon 程序还创建 lineuse 和 reboots 文件。reboots 文件会记录在 wtmpx 文件中找到的所有引导记录。
PROCESS	acctprc 程序用于将 /var/adm/Spacctn.MMDD 进程记帐文件转换为 ptacctn.MMDD 文件中的完整记帐记录。Spacct 和 ptacct 文件通过数字进行关联，这样在 runacct 脚本失败时，将不处理 Spacct 文件。
MERGE	acctmerg 程序将进程记帐记录与连接记帐记录合并，以形成 daytacct 文件。
FEES	acctmerg 程序将 fee 文件中的 ASCII tacct 记录合并到 daytacct 文件中。
DISK	dodisk 脚本可生成 diskacct 文件。如果 dodisk 脚本已运行并生成 diskacct 文件，则 DISK 程序会将该文件合并到 daytacct 文件中，并将 diskacct 文件移至 /tmp/diskacct.MMDD 文件。

表 10-1 runacct 脚本的状态 (续)

状态	说明
MERGETACCT	acctmerg 程序将 daytacct 文件与 sum/tacct 文件（累计总记帐文件）合并。每天都将 daytacct 文件保存至 sum/tacct.MMDD 文件中，这样便可在 sum/tacct 文件损坏或丢失时重新创建该文件。
CMS	acctcms 程序多次运行。第一次运行该程序是为了使用 Spacct <i>n</i> 文件生成命令摘要，并向 sum/daycms 文件写入数据。第二次运行 acctcms 程序是为了将 sum/daycms 文件与 sum/cms 累积命令摘要文件合并。最后，再运行 acctcms 程序，以生成 nite/daycms 和 nite/cms，即分别来自 sum/daycms 和 sum/cms 的 ASCII 文件。lastlogin 程序用于创建 /var/adm/acct/sum/loginlog 日志文件。此文件将报告每个用户的上次登录时间。如果 runacct 脚本在午夜后运行，则显示某些用户上次登录时间的日期会与正确日期相差一天。
USEREXIT	任何与安装相关（本地）的记帐程序都可在此点运行。runacct 脚本预计此程序为 /usr/lib/acct/runacct.local 程序。
CLEANUP	此状态可清除临时文件、运行 prdaily 脚本、将其输出保存至 sum/rpt.MMDD 文件、删除锁定，然后退出。



注意 - 如果以 CLEANUP 状态重新启动 runacct 脚本，请删除最后一个 ptacct 文件，因为该文件将不完整。

每日记帐报告

runacct shell 脚本可在每次调用时生成五个基本报告。下表对这些报告进行了描述。

表 10-2 每日记帐报告

报告类型	说明
第 134 页中的“每日报告”	按 tty 数显示终端线使用率。
第 135 页中的“每日使用情况报告”	按用户指明系统资源的使用情况（按用户 ID 的顺序列出）。
第 136 页中的“每日命令摘要”	按命令指明系统资源的使用率，按内存使用的降序列出。换言之，使用内存最多的命令列在最前面。在每月命令摘要中会为当月报告此信息。
第 137 页中的“每月命令摘要”	反映自上次调用 monacct 程序以来所累积数据的累计摘要。
第 137 页中的“上次登录报告”	显示每位用户上次登录的时间（按年代顺序列出）。

每日报告

此报告提供了所使用的每个终端线的信息。下面是一个每日报告样例：

Jan 16 02:30 2004 DAILY REPORT FOR venus Page 1

from Mon Jan 15 02:30:02 2004
to Tue Jan 16 02:30:01 2004
1 runacct
1 acctcon

TOTAL DURATION IS 1440 MINUTES

LINE	MINUTES	PERCENT	# SESS	# ON	# OFF
console	868	60	1	1	2
TOTALS	868	--	1	1	2

from 和 to 行指定报告中反映的时间段。此时间段包含自上次生成每日报告到生成当前每日报告之间的时间。然后，报告将显示有关系统重新引导、关机、电源断电恢复的日志以及由 acctwtmp 程序写入 /var/adm/wtmpx 文件的其他任何记录。有关更多信息，请参见 [acct\(1M\)](#) 手册页。

报告的第二部分是对终端线使用情况的细分。TOTAL DURATION 指出系统处于多用户模式（可通过终端线访问）的时间。下表介绍了每日报告所提供的数据。

LINE	终端线或访问端口。
MINUTES	在记帐期间使用该线的分钟数。
PERCENT	按 MINUTES 数划分的 TOTAL DURATION。
# SESS	登录会话访问此线或端口的次数。
# ON	与 SESS 相同。（此列不再有意义。以前，此列可列出用户使用某线或端口登录的次数。）
# OFF	用户注销的次数以及该线中出现的任何中断。通常，在系统进入多用户模式后初次调用 ttymon 时，端口中会发生中断。如果 # OFF 超出 # SESS 很多，则多路复用器、调制解调器或电缆可能存在问题。或者，某处存在连接问题。最常见的原因是电缆与多路复用器的连接不牢固。

在实际使用中，应监视 /var/adm/wtmpx 文件，因为连接记帐源自该文件。如果 wtmpx 文件快速增大，请执行以下命令确定哪条 tty 线噪音最大。

/usr/lib/acct/acctcon -l file < /var/adm/wtmpx

如果不断发生中断，则总体系统性能将受到影响。此外，还可能损坏 wtmp 文件。要更正此问题，请参见第 126 页中的“如何修复损坏的 wtmpx 文件”。

每日使用情况报告

每日使用情况报告按用户细分系统资源使用情况。下面是此报告的一个样例：

Jan 16 02:30 2004 DAILY USAGE REPORT FOR skisun Page 1

UID	LOGIN NAME	CPU PRIME	(MINS) NPRIME	KCORE-PRIME	MINS NPRIME	CONNECT PRIME	(MINS) NPRIME	DISK BLOCKS	# OF PROCS	# OF SESS	# DISK SAMPLES	FEE
0	TOTAL	72	148	11006173	51168	26230634	57792	539	330	0	2150	1
0	root	32	76	11006164	33664	26230616	22784	0	0	0	127	0
4	adm	0	0	22	51	0	0	0	420	0	0	0
101	rimmer	39	72	894385	1766020	539	330	0	1603	1	0	0

下表描述每日使用情况报告所提供的数据。

表 10-3 每日使用情况报告数据

列	说明
UID	用户 ID 号。
LOGIN NAME	用户的登录（或用户）名。标识有多个登录名的用户。
CPU (MINS)	用户的进程使用中央处理器的时间长度（分钟）。使用情况分为 PRIME 和 NPRIME（非主要）。此数据的记帐系统版本位于 /etc/acct/holidays 文件中。
KCORE-MINS	进程运行时每分钟所使用内存量（千字节）的累计度量。使用情况分为 PRIME 和 NPRIME。
CONNECT (MINS)	用户登录到系统的时间长度（分钟），或“实际时间”。使用情况分为 PRIME 和 NPRIME。如果这些数字很大而 # OF PROCS 很小，则可推断出，用户早晨先登录，然后在一天的其余时间内几乎未连接终端。
DISK BLOCKS	acctdusg 程序的输出，该程序运行磁盘记帐程序并合并记帐记录 (daytacct)。用于记帐的块的大小为 512 字节。
# OF PROCS	用户调用的进程数。如果出现的数字很大，则表明用户可能有一个 shell 过程在运行时无法受到控制。
# OF SESS	用户登录系统的次数。
# DISK SAMPLES	运行磁盘记帐以获得 DISK BLOCKS 平均数的次数。
FEE	此字段表示 chargefee 脚本对用户收取费用的总累积单位，通常不使用此字段。

每日命令摘要

每日命令摘要报告按命令显示系统资源使用情况。通过此报告，可以确定使用率最高的命令。根据这些命令使用系统资源的方式，可以准确把握对系统进行调优的最佳方式。

这些报告按 TOTAL KCOREMIN 排序，它是任意衡量，但常用于计算系统中的垃圾箱。

下面是一个每日命令摘要样例：

COMMAND NAME	NUMBER CMDS	TOTAL COMMAND SUMMARY							
		TOTAL KCOREMIN	TOTAL CPU-MIN	TOTAL REAL-MIN	MEAN SIZE-K	MEAN CPU-MIN	HOG FACTOR	CHARS TRNSFD	BLOCKS READ
TOTALS	2150	1334999.75	219.59	724258.50	6079.48	0.10	0.00	397338982	419448
netscape	43	2456898.50	92.03	54503.12	26695.51	2.14	0.00	947774912	225568
adeptedi	7	88328.22	4.03	404.12	21914.95	0.58	0.01	93155160	8774
dtmail	1	54919.17	5.33	17716.57	10308.94	5.33	0.00	213843968	40192
acroread	8	31218.02	2.67	17744.57	11682.66	0.33	0.00	331454464	11260
dtwm	1	16252.93	2.53	17716.57	6416.05	2.53	0.00	158662656	12848
dtterm	5	4762.71	1.30	76300.29	3658.93	0.26	0.00	33828352	11604
dtaction	23	1389.72	0.33	0.60	4196.43	0.01	0.55	18653184	539
dtsessio	1	1174.87	0.24	17716.57	4932.97	0.24	0.00	23535616	5421
dtcm	1	866.30	0.18	17716.57	4826.21	0.18	0.00	3012096	6490

以下列表介绍了每日命令摘要提供的数据。

COMMAND NAME	命令的名称。由于进程记帐系统只报告对象模块，因此所有 Shell 过程都集中在名称 sh 下。应对名为 a.out、core 或其他任何意外名称的程序的频率进行监视。可以使用 acctcom 程序来确定哪个用户执行了名称奇怪的命令以及是否使用了超级用户特权。
NUMBER CMDS	运行此命令的总次数。
TOTAL KCOREMIN	进程运行时每分钟所使用千字节内存段的累计度量。
TOTAL CPU-MIN	此程序累积的总处理时间。
TOTAL REAL-MIN	此程序累积的总实时（挂钟）分钟数。
MEAN SIZE-K	在 NUMBER CMDS 所反映的调用次数中 TOTAL KCOREMIN 的平均值（平均数）。
MEAN CPU-MIN	源自 NUMBER CMDS 和 TOTAL CPU-MIN 的平均值（平均数）。
HOG FACTOR	CPU 总时间除以已用时间。显示系统可用性与系统使用率之比，提供进程在执行期间可用的总 CPU 时间的相对测量。
CHARS TRNSFD	由读和写系统调用传送的字符数。可能因溢出而为负值。
BLOCKS READ	一个进程执行的物理块读取和写入总数。

每月命令摘要

每日命令摘要报告的格式实际上与每月命令摘要报告的格式相同。但是，每日摘要只针对当前记帐期间进行报告，而每月摘要则针对从财务期间开始到当前日期之间的时间段进行报告。换言之，每月报告是一个累计摘要，它反映自上次调用 `monacct` 程序以来累积的数据。

下面是一个每月命令摘要样例。

Jan 16 02:30 2004 MONTHLY TOTAL COMMAND SUMMARY Page 1

COMMAND NAME	NUMBER CMDS	TOTAL COMMAND SUMMARY							
		TOTAL KCOREMIN	TOTAL CPU-MIN	TOTAL REAL-MIN	MEAN SIZE-K	MEAN CPU-MIN	HOG FACTOR	CHARS TRNSFD	BLOCKS READ
TOTALS	42718	4398793.50	361.92	956039.00	12154.09	0.01	0.00	16100942848	825171
netscape	789	3110437.25	121.03	79101.12	25699.58	0.15	0.00	3930527232	302486
adeptedi	84	1214419.00	50.20	4174.65	24193.62	0.60	0.01	890216640	107237
acroread	145	165297.78	7.01	18180.74	23566.84	0.05	0.00	1900504064	26053
dtmail	2	64208.90	6.35	20557.14	10112.43	3.17	0.00	250445824	43280
dtaction	800	47602.28	11.26	15.37	4226.93	0.01	0.73	640057536	8095
soffice.	13	35506.79	0.97	9.23	36510.84	0.07	0.11	134754320	5712
dtwm	2	20350.98	3.17	20557.14	6419.87	1.59	0.00	190636032	14049

有关每月命令摘要所提供数据的描述，请参见第 136 页中的“每日命令摘要”。

上次登录报告

此报告给出上次进行特定登录的日期。使用此信息可以找出未使用的登录以及可以存档和删除的登录目录。下面是一个上次登录报告。

Jan 16 02:30 2004 LAST LOGIN Page 1

01-06-12	kryten	01-09-08	protoA	01-10-14	ripley
01-07-14	lister	01-09-08	protoB	01-10-15	scutter1
01-08-16	pmorph	01-10-12	rimmer	01-10-16	scutter2

使用 `acctcom` 检查 `pacct` 文件

您随时都可以使用 `acctcom` 程序来检查 `/var/adm/pacct n` 文件的内容，或含有 `acct.h` 格式的记录的任文件的内容。如果运行此命令时不指定任何文件也不提供任何标准输入，`acctcom` 命令将读取 `pacct` 文件。`acctcom` 命令所读取的每条记录都表示有关已终止进程的信息。可通过运行 `ps` 命令来检查活动进程。

acctcom 命令的缺省输出提供以下信息：

# acctcom			START	END	REAL	CPU	MEAN
COMMAND			TIME	TIME	(SECS)	(SECS)	SIZE(K)
NAME	USER	TTYNAME					
#accton	root	?	02:30:01	02:30:01	0.03	0.01	304.00
turnacct	adm	?	02:30:01	02:30:01	0.42	0.01	320.00
mv	adm	?	02:30:01	02:30:01	0.07	0.01	504.00
utmp_upd	adm	?	02:30:01	02:30:01	0.03	0.01	712.00
utmp_upd	adm	?	02:30:01	02:30:01	0.01	0.01	824.00
utmp_upd	adm	?	02:30:01	02:30:01	0.01	0.01	912.00
utmp_upd	adm	?	02:30:01	02:30:01	0.01	0.01	920.00
utmp_upd	adm	?	02:30:01	02:30:01	0.01	0.01	1136.00
utmp_upd	adm	?	02:30:01	02:30:01	0.01	0.01	576.00
closewtm	adm	?	02:30:01	02:30:01	0.10	0.01	664.00

以下列表介绍了每个字段：

COMMAND	命令名称（如果以超级用户权限执行命令，则带有井号（#））
NAME	
USER	用户名
TTYNAME	tty 名称（如果未知则列为？）
START TIME	命令执行开始时间
END TIME	命令执行结束时间
REAL (SECS)	实时（秒）
CPU (SECS)	CPU 时间（秒）
MEAN SIZE (K)	平均大小（千字节）

通过使用 acctcom 命令选项可以获得以下信息。

- fork/exec 标志的状态（1 表示不带 exec 的 fork）
- 系统退出状态
- 扰乱因子
- kcore 总分钟数
- CPU 因子
- 传送的字符
- 已读的块

以下列表介绍 acctcom 命令选项。

- a 显示有关选定进程的平均统计信息。记录输出后将列显该统计信息。
- b 向后读取文件，先显示最新命令。如果读取标准输入，此选项将无效。
- f 列显 fork/exec 标志和系统退出状态列。输出是一个八进制数字。

-h	显示扰乱因子而不是平均内存大小，扰乱因子是进程在执行期间使用的总可用 CPU 时间的分数。扰乱因子 = 总 CPU 时间/已用时间。
-i	列显输出中包含 I/O 计数的列。
-k	显示总 kcore minutes 而不是内存大小。
-m	显示平均核心转储大小。此大小是缺省值。
-q	列显平均统计信息，而不是输出记录。
-r	显示 CPU 因子： $user-time/(system-time + user-time)$ 。
-t	显示单独的系统和用户 CPU 时间。
-v	从输出中排除列标题。
-C sec	只显示总 CPU 时间（系统加用户）超过 sec 秒的进程。
-e time	显示在 time（给定格式 hr[: min[: sec]]）时或之前存在的进程。
-E time	显示在 time（给定格式 hr[: min[: sec]]）时或之前开始的进程。对 -S 和 -E 使用相同时间，以显示在该时间存在的进程。
-g group	只显示属于 group 的进程。
-H factor	只显示超出 factor 的进程，其中 factor 是“扰乱因子”（请参见 -h 选项）。
-I chars	只显示传送的字符数大于 chars 所指定的分界数的进程。
-l line	只显示属于终端 /dev/line 的进程。
-n pattern	只显示与 pattern（正则表达式，只是 "+" 表示出现一次或多次）匹配的命令。
-o ofile	不列显记录，而以 acct.h 格式将其复制到 ofile 文件中。
-O sec	只显示 CPU 系统时间超出 sec 秒的进程。
-s time	显示在 time（给定格式 hr[: min[: sec]]）时或之后存在的进程。
-S time	显示在 time（给定格式 hr[: min[: sec]]）时或之后开始的进程。
-u user	只显示属于 user 的进程。

系统记帐文件

/var/adm 目录包含活动的的数据收集文件。下表介绍了此目录中的记帐文件。

dtmp	acctdusg 程序的输出
fee	chargefee 程序的输出，该输出为 ASCII tacct 记录
pacct	活动进程记帐文件

`pacct n` 通过运行 `turnacct` 脚本切换的进程记帐文件

`Spacctn.MMDD` 执行 `runacct` 脚本期间用于 `MMDD` 的进程记帐文件

`/var/adm/acct` 目录包含 `nite`、`sum` 和 `fiscal` 目录。这些目录包含实际的数据收集文件。例如，`nite` 目录包含 `runacct` 脚本每天重用的文件。下面是 `/var/adm/acct/nite` 目录中文件的简短摘要。

表 10-4 `/var/adm/acct/nite` 目录中的文件

文件	说明
<code>active</code>	由 <code>runacct</code> 脚本用来记录进度并显警告和错误消息
<code>active.MMDD</code>	与 <code>runacct</code> 脚本检测到错误后的 <code>active</code> 文件相同
<code>cms</code>	<code>prdaily</code> 脚本使用的 ASCII 总命令摘要
<code>ctacct.MMDD</code>	<code>tacct.h</code> 格式的连接记帐记录
<code>ctmp</code>	<code>acctcon1</code> 程序的输出，其中包括 <code>ctmp.h</code> 格式的连接会话记录（提供 <code>acctcon1</code> 和 <code>acctcon2</code> 是为了兼容）
<code>daycms</code>	<code>prdaily</code> 脚本使用的 ASCII 每日命令摘要
<code>daytacct</code>	<code>tacct.h</code> 格式的一天总计帐记录
<code>disktacct</code>	<code>tacct.h</code> 格式的磁盘记帐记录，由 <code>dodisk</code> 脚本创建
<code>fd2log</code>	执行 <code>runacct</code> 脚本期间的诊断输出
<code>lastdate</code>	上次执行 <code>runacct</code> 脚本的日期（以 <code>date +%m%d</code> 格式）
<code>lineuse</code>	<code>prdaily</code> 脚本使用的 <code>tty</code> 线使用情况报告
<code>lock</code>	用于控制 <code>runacct</code> 脚本的串行使用
<code>log</code>	<code>acctcon</code> 程序的诊断输出
<code>log.MMDD</code>	与 <code>runacct</code> 脚本检测到错误后的 <code>log</code> 文件相同
<code>owtmpx</code>	前一天的 <code>wtmpx</code> 文件
<code>reboots</code>	<code>wtmpx</code> 文件中的开始和结束日期，以及重新启动的列表
<code>statefile</code>	用于在执行 <code>runacct</code> 脚本期间记录当前状态
<code>tmpwtmp</code>	由 <code>wtmpfix</code> 程序更正的 <code>wtmpx</code> 文件
<code>wtmperror</code>	包含 <code>wtmpfix</code> 错误消息
<code>wtmperror MMDD</code>	与 <code>runacct</code> 脚本检测到错误后的 <code>wtmperror</code> 文件相同
<code>wtmp MMDD</code>	<code>runacct</code> 脚本的 <code>wtmpx</code> 文件副本

sum 目录中包含由 runacct 脚本更新并由 monacct 脚本使用的累积摘要文件。下表汇总了 /var/adm/acct/sum 目录中的文件。

表 10-5 /var/adm/acct/sum 目录中的文件

文件	说明
cms	当前财务期间的二进制格式的总命令摘要文件
cmsprev	没有最新更新的命令摘要文件
daycms	表示日使用情况的内部摘要格式的命令摘要文件
loginlog	用户上次登录日期的记录；由 lastlogin 脚本创建，在 prdaily 脚本中使用
rprt.MMDD	已保存的 prdaily 脚本的输出
tacct	当前财务期间的累计总记帐文件
tacctprev	与没有最新更新的 tacct 文件相同
tacct.MMDD	MMDD 的总记帐文件

fiscal 目录中包含由 monacct 脚本创建的定期摘要文件。下表汇总了 /var/adm/acct/fiscal 目录中的文件。

表 10-6 /var/adm/acct/fiscal 目录中的文件

文件	说明
cmsn	财务期间 <i>n</i> 的内部摘要格式的总命令摘要文件
fiscrptn	与财务期间 <i>n</i> 的 <i>rprt</i> <i>n</i> 类似的报告
tacctn	财务期间 <i>n</i> 的总记帐文件

runacct 脚本生成的文件

下表汇总了由 runacct 脚本生成的最有用的文件。这些文件位于 /var/adm/acct 目录中。

表 10-7 runacct 脚本创建的文件

文件	说明
nite/daytacct	某日的 tacct.h 格式的总记帐文件。

表 10-7 runacct 脚本创建的文件 (续)

文件	说明
nite/lineuse	runacct 脚本调用 acctcon 程序以从 /var/adm/acct/nite/tmpwtmp 文件中收集有关终端线使用情况的数据，并将数据写入 /var/adm/acct/nite/lineuse 文件。prdaily 脚本使用此数据来报告线使用情况。此报告对于检测故障线特别有用。如果退出数与登录数之比大于三比一，该线很可能出现了故障。
sum/cms	此文件是每天的命令摘要的累积。该累积在 monacct 脚本执行时重新启动。ASCII 版本是 nite/cms 文件。
sum/daycms	runacct 脚本调用 acctcms 程序以处理该日使用的命令，从而创建每日命令摘要报告并将数据存储在 /var/adm/acct/sum/daycms 文件中。ASCII 版本是 /var/adm/acct/nite/daycms 文件。
sum/loginlog	runacct 脚本调用 lastlogin 脚本，以便对 /var/adm/acct/sum/loginlog 文件中的登录更新上次登录日期。lastlogin 命令还从此文件中删除不再有效的所有登录。
sum/rprt.MMDD	每次执行 runacct 脚本都会保存 prdaily 脚本所列显的每日报告的副本。
sum/tacct	包含每天的 nite/daytacct 数据的累积并用于记帐目的。monacct 脚本在每月或每个财务期间重新开始累积此数据。

管理系统性能（概述）

从计算机或网络中获取良好性能是系统管理的重要部分。本章概述与管理计算机系统的性能有关的一些因素。

以下是本章中概述信息的列表。

- 第 143 页中的“管理系统性能方面的新增功能”
- 第 144 页中的“有关系统性能任务的参考信息”
- 第 144 页中的“系统性能和系统资源”
- 第 145 页中的“进程和系统性能”
- 第 146 页中的“关于监视系统性能”

管理系统性能方面的新增功能

本节介绍此 Oracle Solaris 发行版中用于管理系统性能的新增功能或已更改的功能。有关新增功能的完整列表以及 Oracle Solaris 发行版的说明，请参见《[Oracle Solaris 10 8/11 新增功能](#)》。

增强的 pfiles 工具

Oracle Solaris 10：/proc 文件系统已得到增强，目前在 /proc/pic/path 目录中包含文件名信息。pfiles 使用此信息显示进程中每个文件的文件名。这一更改使用户能以全新的方式洞察进程的行为。有关更多信息，请参见第 155 页中的“[如何显示有关进程的信息](#)”和 [proc\(1\)](#) 手册页。

CPU 性能计数器

Oracle Solaris 10：CPU 性能计数器 (CPU Performance Counter, CPC) 系统进行了更新，从而能够更好地访问运行 Oracle Solaris OS 的 SPARC 和 x86 平台中提供的性能分析功能。

CPC 命令 `cpustat` 和 `cputrack` 增强了用于指定 CPU 信息的命令行语法。例如，在以前的 Oracle Solaris OS 版本中，您需要指定两个计数器。现在，这两个命令的配置都允许您仅指定一个计数器，如以下示例中所示：

```
# cputrack -c pic0=Cycle_cnt ls -d .
time lwp      event      pic0      pic1
.
0.034   1          exit      841167
```

对于简单的测量，您甚至可以省略计数器配置，如以下示例中所示：

```
# cputrack -c Cycle_cnt ls -d .
time lwp      event      pic0      pic1
.
0.016   1          exit      850736
```

有关使用 `cpustat` 命令的更多信息，请参见 [cpustat\(1M\)](#) 手册页。有关使用 `cputrack` 命令的更多信息，请参见 [cputrack\(1\)](#) 手册页。

有关系统性能任务的参考信息

系统性能任务	更多信息
管理进程	第 12 章，管理系统进程（任务）
监视系统性能	第 13 章，监视系统性能（任务）
更改可调参数	《Oracle Solaris Tunable Parameters Reference Manual》
管理系统性能任务	《系统管理指南：Oracle Solaris Containers—资源管理和 Oracle Solaris Zones》 中的第 2 章“项目和任务（概述）”
使用 FX 和 FS 调度程序管理进程	《系统管理指南：Oracle Solaris Containers—资源管理和 Oracle Solaris Zones》 中的第 8 章“公平共享调度程序（概述）”

系统性能和系统资源

计算机系统的性能取决于该系统使用和分配其资源的方式。定期监视系统的性能，以便了解系统在正常情况下的行为。应当明确了解预期情况，并能够在出现问题时进行识别。

下表介绍了影响性能的系统资源。

系统资源	说明
中央处理器 (Central processing unit, CPU)	CPU 通过从内存中提取并执行指令来对指令进行处理。
输入/输出 (Input/Output, I/O) 设备	I/O 设备可向计算机传入信息，并可从中传出信息。此类设备可能是终端和键盘、磁盘驱动器或打印机。
内存	物理（或主）内存是系统中的随机访问存储器 (Random Access Memory, RAM) 量。

第 13 章，[监视系统性能（任务）](#) 对显示有关系统活动和性能统计信息的工具进行了介绍。

进程和系统性能

下表介绍了与进程相关的术语。

表 11-1 进程术语

术语	说明
进程	任何系统活动或作业。每次引导系统、执行命令或启动应用程序时，系统都会激活一个或多个进程。
Lightweight Process, LWP（轻量级进程）	虚拟 CPU 或执行资源。LWP 由内核预定，以根据其预定类和优先级来使用可用的 CPU 资源。LWP 包括一个内核线程和一个 LWP。内核线程包含必须始终位于内存中的信息。LWP 包含可交换的信息。
Application thread（应用程序线程）	具有单独栈的一系列指令，这些指令可在用户的地址空间中独立执行。可在 LWP 顶部复用应用程序线程。

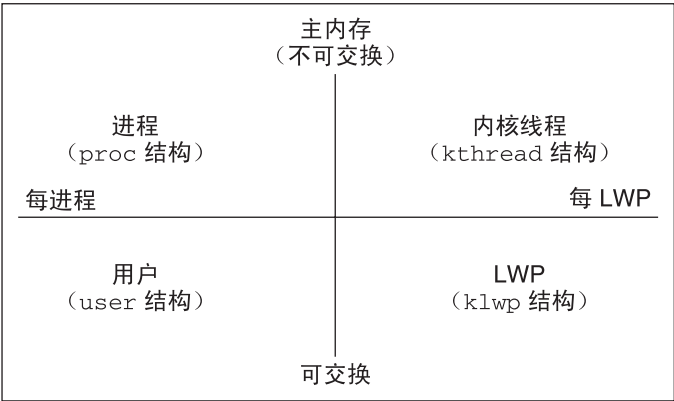
一个进程可以由多个 LWP 和多个应用程序线程组成。内核预定内核线程结构，该结构是 SunOS 环境中的预定实体。下表介绍了各种进程结构。

表 11-2 进程结构

结构	说明
proc	包含与整个进程有关的信息，该信息必须始终位于主内存中
kthread	包含与 LWP 有关的信息，该信息必须始终位于主内存中
user	包含可交换的“每进程”信息
klwp	包含可交换的“每 LWP 进程”信息

下图显示了这些进程结构之间的关系。

图 11-1 进程结构之间的关系



进程中的所有线程都可以访问大多数进程资源。几乎所有进程虚拟内存都是共享的。一个线程对共享数据的更改可供进程中的其他线程使用。

关于监视系统性能

计算机运行时，操作系统中的计数器会增加，以跟踪各种系统活动。

跟踪的系统活动如下所示：

- 中央处理器 (Central Processing Unit, CPU) 使用率
- 缓冲区使用情况
- 磁盘和磁带输入/输出 (Input/Output, I/O) 活动
- 终端设备活动
- 系统调用活动
- 上下文切换
- 文件访问
- 队列活动
- 内核表
- 进程间通信
- 分页
- 可用内存和交换空间
- 内核内存分配 (Kernel Memory Allocation, KMA)

监视工具

Oracle Solaris 软件提供了多种工具，以帮助您跟踪系统的执行情况。下表介绍了这些工具。

表 11-3 性能监视工具

命令	说明	更多信息
cpustat 和 cputrack 命令	使用 CPU 性能计数器监视系统或进程的性能。	cpustat(1M) 和 cputrack(1)
netstat 和 nfsstat 命令	显示有关网络性能的信息。	netstat(1M) 和 nfsstat(1M)
ps 和 prstat 命令	显示有关活动进程的信息。	第 12 章，管理系统进程（任务）
sar 和 sadc 命令	收集并报告系统活动数据。	第 13 章，监视系统性能（任务）
Sun Enterprise SyMON	收集有关 Sun 的企业级系统的系统活动数据。	《Sun Enterprise SyMON 2.0.1 Software User's Guide》
swap 命令	显示有关系统中的可用交换空间的信息。	《系统管理指南：设备和文件系统》中的第 21 章“配置其他交换空间（任务）”
vmstat 和 iostat 命令	汇总系统活动数据，如虚拟内存统计信息、磁盘使用情况和 CPU 活动。	第 13 章，监视系统性能（任务）
cputrack 和 cpustat 命令	协助访问微处理器提供的硬件性能计数器功能。	cputrack(1) 和 cpustat(1M) 手册页
kstat 和 mpstat 命令	检查系统中可用的内核统计信息或 kstats，并报告与命令行上指定的条件匹配的统计信息。mpstat 命令以表格形式报告处理器统计信息。	kstat(1M) 和 mpstat(1M) 手册页

管理系统进程（任务）

本章介绍用于管理系统进程的过程。

有关与管理系统进程相关的过程的信息，请参见以下主题：

- [第 149 页中的“系统进程管理方面的新增功能”](#)
- [第 150 页中的“管理系统进程（任务列表）”](#)
- [第 159 页中的“管理进程类信息（任务列表）”](#)

有关管理系统进程的概述信息，请参见以下主题：

- [第 150 页中的“用于管理系统进程的命令”](#)
- [第 160 页中的“管理进程类信息”](#)

系统进程管理方面的新增功能

伪内核进程

Oracle Solaris 包括一些在系统中运行的进程。这些进程执行特定任务，但不需要管理。请注意，没有介绍这些进程的参考手册文档（手册页）。

下表介绍了其中的每个进程。

进程	说明
fsflush	将页面刷新到磁盘的系统守护进程
init	启动和重新启动其他进程和 SMF 组件的初始系统进程
intrad	由于中断而监视并平衡系统负载的系统进程
kmem_task	监视内存缓存大小的系统进程

进程	说明
pageout	控制到磁盘的内存分页的系统进程
sched	负责 OS 调度和进程交换的系统进程
vm_tasks	用于在各个 CPU 间平衡和分配与虚拟内存相关的工作负荷以提高性能的系统进程，该进程针对每个处理器有一个线程
zpool-pool-name	包含关联池的 I/O taskq 线程的每个 ZFS 存储池的系统进程

管理系统进程（任务列表）

任务	说明	参考
列出进程。	使用 <code>ps</code> 命令列出系统中的所有进程。	第 154 页中的“如何列出进程”
显示有关进程的信息。	使用 <code>pgrep</code> 命令获取要显示其详细信息的进程的进程 ID。	第 155 页中的“如何显示有关进程的信息”
控制进程。	使用 <code>pgrep</code> 命令找到进程。然后，使用相应的 <code>pcommand (/proc)</code> 来控制进程。有关 <code>(/proc)</code> 命令的说明，请参见 表 12-3 。	第 156 页中的“如何控制进程”
中止进程。	通过进程名称或进程 ID 找到进程。然后可以使用 <code>pkill</code> 或 <code>kill</code> 命令终止进程。	第 157 页中的“如何终止进程 (pkill)” 第 158 页中的“如何终止进程 (kill)”

用于管理系统进程的命令

下表介绍了用于管理系统进程的命令。

表 12-1 用于管理进程的命令

命令	说明	手册页
<code>ps</code> 、 <code>pgrep</code> 、 <code>prstat</code> 、 <code>pkill</code>	检查系统中活动进程的状态，并显示有关这些进程的详细信息。	ps(1) 、 pgrep(1) 和 prstat(1M)

表 12-1 用于管理进程的命令 (续)

命令	说明	手册页
kill	功能与 pgrep 相同，但通过名称或其他属性来查找进程或向进程发送信号，然后终止进程。像 kill 命令一样向每个匹配的进程发送信号，而不列显进程 ID。	pgrep(1) 和 pkill(1) kill(1)
pargs、preap	协助进行进程调试。	pargs(1) 和 preap(1)
dispadmin	列出缺省进程调度策略。	dispadmin(1M)
priocntl	为优先级类指定进程并管理进程优先级。	priocntl(1)
nice	更改分时进程的优先级。	nice(1)
psrset	将特定进程组绑定到一组处理器而非一个处理器。	psrset(1M)

Solaris Management Console 的进程工具使您可通过用户友好的界面来管理进程。有关使用和启动 Solaris Management Console 的信息，请参见《系统管理指南：基本管理》中的第 2 章“使用 Solaris Management Console（任务）”。

使用 ps 命令

使用 [ps](#) 命令可以检查系统中活动进程的状态，并可显示有关进程的技术信息。此数据对于管理任务（例如，确定设置进程优先级的方式）很有用。

根据您所使用的选项，[ps](#) 命令会报告以下信息：

- 进程的当前状态
- 进程 ID
- 父进程 ID
- 用户 ID
- 调度类
- 优先级
- 进程的地址
- 已用内存
- 已用 CPU 时间

下表介绍了 [ps](#) 命令报告的一些字段。具体显示哪些字段取决于您选择的选项。有关所有可用选项的说明，请参见 [ps\(1\)](#) 手册页。

表 12-2 ps 报告中字段的汇总

字段	说明
UID	进程所有者的有效用户 ID。
PID	进程 ID。
PPID	父进程 ID。
C	用于调度的处理器使用率。使用 -c 选项时，将不显示此字段。
CLS	进程所属的调度类，例如实时、系统或分时。只有 -c 选项包括此字段。
PRI	内核线程的调度优先级。数值越大，表示优先级越高。
NI	进程的 nice 数值，该数值对其调度优先级有影响。进程的 nice 值越高，意味着其优先级越低。
ADDR	proc 结构的地址。
SZ	进程的虚拟地址大小。
WCHAN	进程休眠的事件或锁定的地址。
STIME	以小时、分钟和秒表示的进程开始时间。
TTY	从中启动进程或其父级的终端。问号表示没有控制终端。
TIME	进程自开始以来使用的 CPU 时间总量。
CMD	生成进程的命令。

使用 /proc 文件系统和命令

使用进程命令可以显示有关 /proc 目录中所列进程的详细信息。下表列出了 /proc 进程命令。/proc 目录也称为进程文件系统 (Process File System, PROCFS)。活动进程的映像按进程 ID 号存储在此位置。

表 12-3 进程命令 (/proc)

进程命令	说明
psred	显示进程凭证信息
pfiles	为进程中打开的文件报告 fstat 和 fcntl 信息
pflags	列显 /proc 跟踪标志、悬挂信号和保留信号以及其他状态信息
pldd	列出链接至进程中的动态库

表 12-3 进程命令 (/proc) (续)

进程命令	说明
pmap	列显每个进程的地址空间图
psig	列出每个进程的信号操作和处理程序
prun	启动每个进程
pstack	显示每个进程中的每个 lwp 的十六进制符号栈跟踪
pstop	停止每个进程
ptime	使用微状态记帐记录进程的时间
ptree	显示包含该进程的进程树
pwait	在进程终止后显示状态信息
pwdx	显示进程的当前工作目录

有关更多信息，请参见 [proc\(1\)](#)。

进程工具与 ps 命令的某些选项相似，区别在于这些命令提供的输出更详细。

通常，进程命令可执行以下操作：

- 显示有关进程的更多信息，例如 `fstat` 和 `fcntl`、工作目录以及父进程和子进程树。
- 通过允许用户停止或恢复进程来提供对进程的控制。

使用进程命令管理进程 (/proc)

通过使用一些进程命令，可以显示有关进程的详细技术信息或者控制活动进程。表 12-3 列出了一些 /proc 命令。

如果某个进程陷入无限循环或者执行时间过长，则可能需要停止（中止）该进程。有关使用 `kill` 或 `pkill` 命令停止进程的更多信息，请参见第 12 章，[管理系统进程（任务）](#)。

/proc 文件系统是一个目录分层结构，其中包括状态信息和控制功能的附加子目录。

/proc 文件系统还提供了 `xwatchpoint` 功能，用于在进程地址空间中的各页上重新映射读写权限。该工具没有限制，并且是 MT-safe（多线程安全）的。

为了能使用 /proc 的 `xwatchpoint` 功能，调试工具已进行了修改，这意味着整个 `xwatchpoint` 进程会更快。

使用 dbx 调试工具设置 xwatchpoint 时，已取消以下限制：

- 由于基于 SPARC 的系统注册窗口而在栈的本地变量中设置 xwatchpoint。
- 在多线程进程中设置 xwatchpoint。

有关更多信息，请参见 [proc\(4\)](#) 和 [mdb\(1\)](#) 手册页。

▼ 如何列出进程

- 可使用 **ps** 命令列出系统中的所有进程。

\$ ps [-efc]

ps 只显示与您的登录会话关联的进程。

-ef 显示系统中正在执行的所有进程的完整信息。

-c 显示进程调度程序信息。

示例 12-1 列出进程

以下示例显示 **ps** 命令在不带任何选项时的输出。

```
$ ps
  PID TTY          TIME CMD
 1664 pts/4        0:06  csh
 2081 pts/4        0:00  ps
```

以下示例显示 **ps -ef** 命令的输出。此输出显示，系统启动时最先执行的进程是 **sched**（交换程序），然后是 **init** 进程、**pageout** 等。

```
$ ps -ef
  UID  PID  PPID  C   STIME TTY          TIME CMD
  root     0     0   0   Dec 20 ?         0:17  sched
  root     1     0   0   Dec 20 ?         0:00  /etc/init -
  root     2     0   0   Dec 20 ?         0:00  pageout
  root     3     0   0   Dec 20 ?         4:20  fsflush
  root   374   367   0   Dec 20 ?         0:00  /usr/lib/saf/ttymon
  root   367     1   0   Dec 20 ?         0:00  /usr/lib/saf/sac -t 300
  root   126     1   0   Dec 20 ?         0:00  /usr/sbin/rpcbind
  root    54     1   0   Dec 20 ?         0:00  /usr/lib/sysevent/syseventd
  root    59     1   0   Dec 20 ?         0:00  /usr/lib/picl/picld
  root   178     1   0   Dec 20 ?         0:03  /usr/lib/autofs/automountd
  root   129     1   0   Dec 20 ?         0:00  /usr/sbin/keyserv
  root   213     1   0   Dec 20 ?         0:00  /usr/lib/lpsched
  root   154     1   0   Dec 20 ?         0:00  /usr/sbin/inetd -s
  root   139     1   0   Dec 20 ?         0:00  /usr/lib/netsvc/yp/ypbind ...
  root   191     1   0   Dec 20 ?         0:00  /usr/sbin/syslogd
  root   208     1   0   Dec 20 ?         0:02  /usr/sbin/nscd
  root   193     1   0   Dec 20 ?         0:00  /usr/sbin/cron
  root   174     1   0   Dec 20 ?         0:00  /usr/lib/nfs/lockd
 daemon  175     1   0   Dec 20 ?         0:00  /usr/lib/nfs/statd
```

```

root  376      1  0  Dec 20 ?      0:00 /usr/lib/ssh/sshd
root  226      1  0  Dec 20 ?      0:00 /usr/lib/power/powerd
root  315      1  0  Dec 20 ?      0:00 /usr/lib/nfs/mountd
root  237      1  0  Dec 20 ?      0:00 /usr/lib/utmpd
.
.
.

```

▼ 如何显示有关进程的信息

- 1 获得要显示更多信息的进程的进程 ID。

```
# pgrep process
```

其中，*process* 是要显示更多信息的进程的名称。

进程 ID 显示在输出的第一列中。

- 2 显示所需进程信息。

```
# /usr/bin/pcommand pid
```

pcommand 是要运行的 (/proc) 命令。表 12-3 列出并介绍了这些命令。

pid 标识进程 ID。

示例 12-2 显示有关进程的信息

以下示例显示如何使用进程命令来显示有关 *cron* 进程的更多信息。

```

# pgrep cron      1
4780
# pwdx 4780      2
4780: /var/spool/cron/atjobs
# ptree 4780      3
4780 /usr/sbin/cron
# pfiles 4780      4
4780: /usr/sbin/cron
Current rlimit: 256 file descriptors
0: S_IFCHR mode:0666 dev:290,0 ino:6815752 uid:0 gid:3 rdev:13,2
   O_RDONLY|O_LARGEFILE
   /devices/pseudo/mm@0:null
1: S_IFREG mode:0600 dev:32,128 ino:42054 uid:0 gid:0 size:9771
   O_WRONLY|O_APPEND|O_CREAT|O_LARGEFILE
   /var/cron/log
2: S_IFREG mode:0600 dev:32,128 ino:42054 uid:0 gid:0 size:9771
   O_WRONLY|O_APPEND|O_CREAT|O_LARGEFILE
   /var/cron/log
3: S_IFIFO mode:0600 dev:32,128 ino:42049 uid:0 gid:0 size:0

```

```

O_RDWR|O_LARGEFILE
/etc/cron.d/FIFO
4: S_IFIFO mode:0000 dev:293,0 ino:4630 uid:0 gid:0 size:0
O_RDWR|O_NONBLOCK
5: S_IFIFO mode:0000 dev:293,0 ino:4630 uid:0 gid:0 size:0
O_RDWR

```

1. 获得 cron 进程的进程 ID
2. 显示 cron 进程的当前工作目录
3. 显示包含 cron 进程的进程树
4. 显示 fstat 和 fcntl 信息

▼ 如何控制进程

- 1 获得要控制的进程的进程 ID。

```
# pgrep process
```

其中，*process* 是要控制的进程的名称。

进程 ID 显示在输出的第一列中。

- 2 使用相应的进程命令来控制进程。

```
# /usr/bin/pcommand pid
```

pcommand 是要运行的进程 (/proc) 命令。表 12-3 列出并介绍了这些命令。

pid 标识进程 ID。

- 3 验证进程状态。

```
# ps -ef | grep pid
```

示例 12-3 控制进程

以下示例显示如何使用进程命令来停止和重新启动 dtpad 进程。

```

# pgrep dtpad      1
2921
# pstop 2921      2
# prun 2921      3

```

1. 获得 dtpad 进程的进程 ID
2. 停止 dtpad 进程
3. 重新启动 dtpad 进程

终止进程 (pkill, kill)

有时，您可能需要停止（中止）进程。该进程可能处于无限循环中。或者，您可能启动了一个大型作业，并且希望在完成之前使其停止。您可以中止属于您的任何进程。超级用户可以中止系统中的任何进程，但进程 ID 为 0、1、2、3 和 4 的进程除外。中止这些进程很可能会使系统崩溃。

有关更多信息，请参见 `pgrep(1)`、`pkill(1)` 和 `kill(1)` 手册页。

▼ 如何终止进程 (pkill)

- 1 可选要终止另一个用户的进程，请成为超级用户或承担等效角色。
- 2 获得要终止的进程的进程 ID。

```
$ pgrep process
```

其中，*process* 是要终止的进程的名称。

例如：

```
$ pgrep netscape
587
566
```

在输出中显示进程 ID。

注 - 要获得有关 Sun Ray 的进程信息，请使用以下命令：

```
# ps -fu user
```

此命令将列出所有用户进程。

```
# ps -fu user | grep process
```

此命令可为用户找到特定进程。

- 3 终止进程。

```
$ pkill [signal] process
```

signal 当 `pkill` 命令行语法中不包含任何信号时，使用的缺省信号为 -15 (SIGKILL)。将 -9 信号 (SIGTERM) 与 `pkill` 命令一起使用，可以确保进程快速终止。但是，不应使用 -9 信号来中止特定进程，例如数据库进程或 LDAP 服务器进程。这样做可能丢失数据。

process 是要停止的进程的名称。

提示 – 使用 `pkill` 命令终止进程时，先尝试使用该命令本身，而不包括信号选项。等待几分钟确定进程是否终止，然后再使用带有 `-9` 信号的 `pkill` 命令。

4 验证进程是否已终止。

```
$ pgrep process
```

`pgrep` 命令的输出中应不再列出已终止的进程。

▼ 如何终止进程 (`kill`)

1 可选要终止另一个用户的进程，请成为超级用户或承担等效角色。

2 获得要终止的进程的进程 ID。

```
$ ps -fu user
```

其中，`user` 是要为其显示进程的用户。

进程 ID 显示在输出的第一列中。

3 终止进程。

```
$ kill [signal-number] pid
```

signal 当 `kill` 命令行语法中不包括任何信号时，使用的缺省信号为 `-15` (`SIGKILL`)。将 `-9` 信号 (`SIGTERM`) 与 `kill` 命令一起使用，可以确保进程快速终止。但是，不应使用 `-9` 信号来中止特定进程，例如数据库进程或 LDAP 服务器进程。这样做可能丢失数据。

pid 是要终止的进程的进程 ID。

提示 – 使用 `kill` 命令停止进程时，先尝试使用该命令本身，而不包括信号选项。等待几分钟以确定进程是否终止，然后再使用带有 `-9` 信号的 `kill` 命令。

4 验证进程是否已终止。

```
$ pgrep pid
```

`pgrep` 命令的输出中应不再列出已终止的进程。

调试进程 (`pargs`、 `preap`)

`pargs` 命令和 `preap` 命令可以改进进程调试。`pargs` 命令可以列显与实时进程或核心文件关联的参数和环境变量。`preap` 命令可以删除不再存在（僵停）的进程。僵停进程尚

不具有由父进程请求的退出状态。这些进程通常是无害的，但如果数量众多，则会占用系统资源。可以使用 `pargs` 和 `preap` 命令检查您有权检查的任何进程。作为超级用户，您可以检查任何进程。

有关使用 `preap` 命令的信息，请参见 [preap\(1\)](#) 手册页。有关使用 `pargs` 命令的信息，请参见 [pargs\(1\)](#) 手册页。另请参见 [proc\(1\)](#) 手册页。

示例 12-4 调试进程 (pargs)

`pargs` 命令可以解决一个长期存在的问题，即使用 `ps` 命令无法显示传递给进程的所有参数。以下示例显示如何通过结合使用 `pargs` 命令和 `pgrep` 命令来显示传递给进程的参数。

```
# pargs 'pgrep ttymon'
579:      /usr/lib/saf/ttymon -g -h -p system-name console login:
-T sun -d /dev/console -l
argv[0]: /usr/lib/saf/ttymon
argv[1]: -g
argv[2]: -h
argv[3]: -p
argv[4]: system-name console login:
argv[5]: -T
argv[6]: sun
argv[7]: -d
argv[8]: /dev/console
argv[9]: -l
argv[10]: console
argv[11]: -m
argv[12]: ldterm,ttcompat
548:      /usr/lib/saf/ttymon
argv[0]: /usr/lib/saf/ttymon
```

以下示例显示如何使用 `pargs -e` 命令来显示与某一进程关联的环境变量。

```
$ pargs -e 6763
6763: tcsh
envp[0]: DISPLAY=:0.0
```

管理进程类信息（任务列表）

任务	说明	参考
显示有关进程类的基本信息。	使用 <code>prctl -l</code> 命令可以显示进程调度类和优先级范围。	第 161 页中的“如何显示有关进程类的基本信息 (prctl)”
显示进程的全局优先级。	使用 <code>ps -ecl</code> 命令可以显示进程的全局优先级。	第 161 页中的“如何显示进程的全局优先级”

任务	说明	参考
指定进程优先级。	使用 <code>priocntl -e -c</code> 命令可以指定的优先级启动进程。	第 162 页中的“如何指定进程优先级 (priocntl)”
更改分时进程的调度参数。	使用 <code>priocntl -s -m</code> 命令可以更改分时进程中的调度参数。	第 162 页中的“如何更改分时进程的调度参数 (priocntl)”
更改进程的类。	使用 <code>priocntl -s -c</code> 命令可以更改进程的类。	第 163 页中的“如何更改进程的类 (priocntl)”
更改进程的优先级。	使用 <code>/usr/bin/nice</code> 命令及相应选项可以降低或提高进程的优先级。	第 164 页中的“如何更改进程的优先级 (nice)”

管理进程类信息

以下列表列出了可在系统中配置的进程调度类。其中还包括分时类的用户优先级范围。

可能的进程调度类如下所示：

- 公平份额 (FSS)
- 固定 (FX)
- 系统 (SYS)
- 交互 (IA)
- 实时 (RT)
- 分时 (Timesharing, TS)
 - 用户提供的优先级范围为 -60 到 +60。
 - 进程的优先级是从父进程继承的。此优先级称为**用户模式优先级**。
 - 系统会在分时分发参数表中查找用户模式优先级。然后，系统将添加任何 `nice` 或 `priocntl`（用户提供的）优先级并确保范围在 0–59 之间，以创建**全局优先级**。

更改进程的调度优先级 (priocntl)

进程的调度优先级是进程调度程序根据调度策略指定的优先级。`dispadmin` 命令可以列出缺省调度策略。有关更多信息，请参见 [dispadmin\(1M\)](#) 手册页。

可以使用 `priocntl` 命令将进程指定给优先级类并管理进程优先级。有关使用 `priocntl` 命令管理进程的说明，请参见 [第 162 页中的“如何指定进程优先级 \(priocntl\)”](#)。

▼ 如何显示有关进程类的基本信息 (priocntl)

- 使用 **priocntl -l** 命令可显示进程调度类和优先级范围。

```
$ priocntl -l
```

示例 12-5 显示有关进程类的基本信息 (priocntl)

以下示例显示 **priocntl -l** 命令的输出。

```
# priocntl -l
CONFIGURED CLASSES
=====

SYS (System Class)

TS (Time Sharing)
    Configured TS User Priority Range: -60 through 60

FX (Fixed priority)
    Configured FX User Priority Range: 0 through 60

IA (Interactive)
    Configured IA User Priority Range: -60 through 60
```

▼ 如何显示进程的全局优先级

- 使用 **ps** 命令可显示进程的全局优先级。

```
$ ps -ecl
```

PRI 列下列出了全局优先级。

示例 12-6 显示进程的全局优先级

以下示例显示 **ps -ecl** 命令的输出。PRI 列中的值显示，**pageout** 进程的优先级最高，而 **sh** 进程的优先级最低。

```
$ ps -ecl
 F S  UID  PID  PPID  CLS  PRI  ADDR      SZ  WCHAN      TTY      TIME  CMD
19 T  0    0    0    SYS  96   f00d05a8  0           ?         0:03  sched
 8 S  0    1    0    TS   50   ff0f4678 185  ff0f4848 ?        36:51  init
19 S  0    2    0    SYS  98   ff0f4018  0  f00c645c ?         0:01  pageout
19 S  0    3    0    SYS  60   ff0f5998  0  f00d0c68 ?       241:01  fsflush
 8 S  0   269  1    TS   58   ff0f5338 303  ff49837e ?         0:07   sac
 8 S  0   204  1    TS   43   ff2f6008  50  ff2f606e console 0:02    sh
```

▼ 如何指定进程优先级 (prctl)

1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的“配置 RBAC（任务列表）”。

2 以指定的优先级启动进程。

```
# prctl -e -c class -m user-limit -p pri command-name
```

-e 执行该命令。

-c class 指定要在其中运行进程的类。有效类包括 TS（timesharing，分时）、RT（real time，实时）、IA（interactive，交互）、FSS（fair share，公平份额）和 FX（fixed priority，固定优先级）。

-m user-limit 使用 -p 选项时，可以指定可将优先级提高或降低的最大量。

-p pri command-name 允许您为实时线程指定在 RT 类中的相对优先级。对于分时进程而言，使用 -p 选项可以指定用户提供的优先级，该优先级范围是 -60 到 +60。

3 验证进程状态。

```
# ps -ecl | grep command-name
```

示例 12-7 指定进程优先级 (prctl)

以下示例显示如何以用户提供的最高优先级启动 find 命令。

```
# prctl -e -c TS -m 60 -p 60 find . -name core -print
# ps -ecl | grep find
```

▼ 如何更改分时进程的调度参数 (prctl)

1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的“配置 RBAC（任务列表）”。

2 更改正在运行的分时进程的调度参数。

```
# prctl -s -m user-limit [-p user-priority] -i idtype idlist
```

-s 允许设置用户优先级范围的上限并更改当前优先级。

-m user-limit 使用 -p 选项时，可以指定可将优先级提高或降低的最大量。

- p *user-priority* 允许指定优先级。
- i *xidtype xidlist* 结合使用 *xidtype* 和 *xidlist* 可以标识一个或多个进程。*xidtype* 指定 ID 的类型，例如进程 ID 或用户 ID。使用 *xidlist* 可以标识进程 ID 或用户 ID 的列表。

3 验证进程状态。

```
# ps -ecl | grep idlist
```

示例 12-8 更改分时进程的调度参数 (priosctl)

以下示例显示如何以 500 毫秒的时间分片、RT 类中的优先级 20 以及全局优先级 120 来执行命令。

```
# priosctl -e -c RT -m 500 -p 20 myprog
# ps -ecl | grep myprog
```

▼ 如何更改进程的类 (priosctl)

1 可选成为超级用户或承担等效角色。

2 更改进程的类。

- ```
priosctl -s -c class -i idtype idlist
```
- s                      允许设置用户优先级范围的上限并更改当前优先级。
  - c *class*              指定进程所要改用的类，分时则指定 TS，实时则指定 RT。
  - i *idtype idlist*      结合使用 *xidtype* 和 *xidlist* 可以标识一个或多个进程。*xidtype* 指定 ID 的类型，例如进程 ID 或用户 ID。使用 *xidlist* 可以标识进程 ID 或用户 ID 的列表。

---

注 - 只有超级用户或在实时 Shell 中工作的用户才能将某一进程更改为实时进程或将实时进程更改为该进程。如果超级用户将用户进程更改为实时类，那么该用户以后不能使用 `priosctl -s` 命令来更改实时调度参数。

---

### 3 验证进程状态。

```
ps -ecl | grep idlist
```

## 示例 12-9 更改进程的类 (priosctl)

以下示例显示如何将属于用户 15249 的所有进程都更改为实时进程。

```
priocntl -s -c RT -i uid 15249
ps -ecl | grep 15249
```

## 更改分时进程的优先级(nice)

支持 `nice` 命令的目的只是为了向下兼容以前的 Solaris 发行版。`priocntl` 命令在管理进程方面提供了更多的灵活性。

进程的优先级由其调度类的策略和 *nice* 数值决定。每个分时进程都有全局优先级。全局优先级的计算方法是用户提供的优先级（该优先级可能受 `nice` 或 `priocntl` 命令影响）加上系统计算的优先级。

进程的执行优先级数由操作系统指定。优先级数由多个因素决定，其中包括进程的调度类、进程使用的 CPU 时间以及进程的 *nice* 数值（对于分时进程的情况）。

每个分时进程都以缺省 *nice* 数值启动，它从父进程中继承该数值。*nice* 数值显示在 `ps` 报告的 **NI** 列中。

用户可通过提高用户提供的优先级来降低进程的优先级。但是，只有超级用户能够通过降低 *nice* 数值来提高进程的优先级。此限制使用户无法提高自己所拥有进程的优先级，因此会独占更大的 CPU 份额。

*nice* 数值的范围是从 0 到 +39，其中 0 表示最高优先级。每个分时进程的缺省 *nice* 值为 20。有两个可用的命令版本：标准版本 `/usr/bin/nice` 以及 C shell 内置命令。

### ▼ 如何更改进程的优先级(nice)

通过使用此过程，用户可以降低进程的优先级。而超级用户可以提高或降低进程的优先级。

---

注 - 本节介绍的是 `/usr/bin/nice` 命令而非 C-shell 内置 *nice* 命令的语法。有关 C-shell *nice* 命令的信息，请参见 `cs(1)` 手册页。

---

- 1 确定要以用户身份还是超级用户身份更改进程的优先级。然后，选择以下操作之一：

- 以用户身份，按照步骤 2 中的示例降低命令的优先级。
- 以超级用户的身份，按照步骤 3 中的示例提高或降低命令的优先级。

- 2 以用户身份，通过增大 *nice* 数值降低命令的优先级。

以下 *nice* 命令使 *nice* 数值增大 5 个单位，从而以更低的优先级执行 *command-name*。

```
$ /usr/bin/nice -5 command-name
```

在前面的命令中，减号指定后面的内容是一个选项。也可以通过以下方式指定此命令：

```
% /usr/bin/nice -n 5 command-name
```

以下 `nice` 命令使 `nice` 数值增大 10 个单位（缺省增量）但未超出最大值 39，从而降低了 `command-name` 的优先级。

```
% /usr/bin/nice command-name
```

### 3 作为超级用户或同等角色，通过更改 `nice` 数值来提高或降低命令的优先级。

以下 `nice` 命令使 `nice` 数值降低 10 个单位但不低于最小值 0，从而提高了 `command-name` 的优先级。

```
/usr/bin/nice --10 command-name
```

在前面的命令中，第一个减号指定后面的内容是一个选项。第二个减号指示一个负数。

以下 `nice` 命令使 `nice` 数值增大 5 个单位但未超出最大值 39，从而降低了 `command-name` 的优先级。

```
/usr/bin/nice -5 command-name
```

另请参见 有关更多信息，请参见 [nice\(1\)](#) 手册页。

## 系统进程问题的故障排除

下面就您可能遇到的突出问题给出了一些提示：

- 查找同一用户所有的多个相同作业。出现此问题可能是由于正在运行的某个脚本启动了多个后台作业，而未等待任何作业完成。
- 查找累积了大量 CPU 时间的进程。通过检查 `ps` 输出中的 `TIME` 字段可以确定此问题。此进程可能处于无限循环中。
- 查找运行优先级过高的进程。使用 `ps -c` 命令检查 `CLS` 字段，其中显示每个进程的调度类。作为实时 (RT) 进程执行的进程会独占 CPU。或者，查找 `nice` 数值很大的分时 (TS) 进程。具有超级用户特权的用户可能提高了某一进程的优先级。系统管理员可以使用 `nice` 命令来降低优先级。
- 查找失控进程。失控进程会逐渐占用越来越多的 CPU 时间。通过查看进程启动时的时间 (STIME) 并监视一会儿 CPU 时间累积 (TIME)，便可确定此问题。



# 监视系统性能（任务）

本章介绍使用 `vmstat`、`iostat`、`df` 和 `sar` 命令监视系统性能的过程。

有关与监视系统性能相关的过程的信息，请参见以下主题：

- [第 167 页中的“显示系统性能信息（任务列表）”](#)
- [第 174 页中的“监视系统活动（任务列表）”](#)

## 显示系统性能信息（任务列表）

| 任务          | 说明                                                         | 参考                                                    |
|-------------|------------------------------------------------------------|-------------------------------------------------------|
| 显示虚拟内存统计信息。 | 使用 <code>vmstat</code> 命令收集虚拟内存统计信息。                       | <a href="#">第 169 页中的“如何显示虚拟内存统计信息 (vmstat)”</a>      |
| 显示系统事件信息。   | 使用 <code>vmstat</code> 命令和 <code>-s</code> 选项显示系统事件信息。     | <a href="#">第 169 页中的“如何显示系统事件信息 (vmstat -s)”</a>     |
| 显示交换统计信息。   | 使用 <code>vmstat</code> 命令和 <code>-S</code> 选项显示交换统计信息。     | <a href="#">第 170 页中的“如何显示交换统计信息 (vmstat -S)”</a>     |
| 显示每台设备的中断。  | 使用 <code>vmstat</code> 命令和 <code>-i</code> 选项显示每台设备的中断次数。  | <a href="#">第 170 页中的“如何显示每台设备的中断次数 (vmstat -i)”</a>  |
| 显示磁盘使用率。    | 使用 <code>iostat</code> 命令报告磁盘输入和输出统计信息。                    | <a href="#">第 171 页中的“如何显示磁盘使用率信息 (iostat)”</a>       |
| 显示扩展磁盘统计信息。 | 使用 <code>iostat</code> 命令和 <code>-xtc</code> 选项显示扩展磁盘统计信息。 | <a href="#">第 172 页中的“如何显示扩展磁盘统计信息 (iostat -xtc)”</a> |
| 显示磁盘空间信息。   | <code>df -k</code> 命令以千字节为单位显示磁盘空间信息。                      | <a href="#">第 173 页中的“如何显示磁盘空间信息 (df -k)”</a>         |

# 显示虚拟内存统计信息 (vmstat)

可以使用 `vmsta` 命令报告虚拟内存统计信息，以及有关系统事件（例如 CPU 负载、分页、上下文切换数、设备中断和系统调用）的信息。`vmstat` 命令还可以显示有关交换、高速缓存刷新和中断的统计信息。

下表介绍了 `vmstat` 命令输出中的字段。

表 13-1 vmstat 命令的输出

| 类别     | 字段名  | 说明                                                  |
|--------|------|-----------------------------------------------------|
| procs  |      | 报告以下内容：                                             |
|        | r    | 分发队列中的内核线程数                                         |
|        | b    | 正在等待资源的阻塞内核线程数                                      |
|        | w    | 正在等待处理中的资源完成的换出 LWP 数                               |
| memory |      | 报告实际内存和虚拟内存的使用情况：                                   |
|        | swap | 可用交换空间                                              |
|        | free | 可用列表的大小                                             |
| page   |      | 以秒为单位报告缺页和分页活动：                                     |
|        | re   | 回收的页面                                               |
|        | mf   | 次要错误和主要错误                                           |
|        | pi   | 页入的千字节数                                             |
|        | po   | 页出的千字节数                                             |
|        | fr   | 释放的千字节数                                             |
|        | de   | 最近换入的进程所需的预计内存                                      |
|        | sr   | 由 page 守护程序扫描的当前未使用的页数。如果 sr 不等于零，则 page 守护程序一直在运行。 |
| disk   |      | 报告每秒的磁盘操作数，最多显示四个磁盘中的数据                             |
| faults |      | 报告每秒的陷阱/中断速率：                                       |
|        | in   | 每秒的中断次数                                             |
|        | sy   | 每秒的系统调用数                                            |
|        | cs   | CPU 上下文切换速率                                         |
| cpu    |      | 报告 CPU 使用时间：                                        |



表 13-1 vmstat 命令的输出 (续)

| 类别 | 字段名 | 说明   |
|----|-----|------|
|    | us  | 用户时间 |
|    | sy  | 系统时间 |
|    | id  | 空闲时间 |

有关此命令的更多详细说明，请参见 [vmstat\(1M\)](#) 手册页。

▼ 如何显示虚拟内存统计信息 (vmstat)

- 使用 **vmstat** 命令以秒为时间间隔单位收集虚拟内存统计信息。

`$ vmstat n`  
其中，*n* 是两次报告之间的间隔秒数。

示例 13-1 显示虚拟内存统计信息

以下示例显示了利用 **vmstat** 报告的以 5 秒为间隔收集的统计信息：

```
$ vmstat 5
kthr memory page disk faults cpu
 r b w swap free re mf pi po fr de sr dd f0 s1 -- in sy cs us sy id
0 0 0 863160 365680 0 3 1 0 0 0 0 0 0 0 0 406 378 209 1 0 99
0 0 0 765640 208568 0 36 0 0 0 0 0 0 0 0 0 479 4445 1378 3 3 94
0 0 0 765640 208568 0 0 0 0 0 0 0 0 0 0 0 423 214 235 0 0 100
0 0 0 765712 208640 0 0 0 0 0 0 0 3 0 0 0 412 158 181 0 0 100
0 0 0 765832 208760 0 0 0 0 0 0 0 0 0 0 0 402 157 179 0 0 100
0 0 0 765832 208760 0 0 0 0 0 0 0 0 0 0 0 403 153 182 0 0 100
0 0 0 765832 208760 0 0 0 0 0 0 0 0 0 0 0 402 168 177 0 0 100
0 0 0 765832 208760 0 0 0 0 0 0 0 0 0 0 0 402 153 178 0 0 100
0 0 0 765832 208760 0 18 0 0 0 0 0 0 0 0 0 407 165 186 0 0 100
```

▼ 如何显示系统事件信息 (vmstat -s)

- 运行 **vmstat -s** 命令，以显示自上次引导系统以来发生的系统事件数。

```
$ vmstat -s
 0 swap ins
 0 swap outs
 0 pages swapped in
 0 pages swapped out
522586 total address trans. faults taken
17006 page ins
 25 page outs
23361 pages paged in
```

```
28 pages paged out
45594 total reclaims
45592 reclaims from free list
0 micro (hat) faults
522586 minor (as) faults
16189 major faults
98241 copy-on-write faults
137280 zero fill page faults
45052 pages examined by the clock daemon
0 revolutions of the clock hand
26 pages freed by the clock daemon
2857 forks
78 vforks
1647 execs
34673885 cpu context switches
65943468 device interrupts
711250 traps
63957605 system calls
3523925 total name lookups (cache hits 99%)
92590 user cpu
65952 system cpu
16085832 idle cpu
7450 wait cpu
```

▼ 如何显示交换统计信息 (vmstat -S)

- 运行 **vmstat -S**，以显示交换统计信息。

```
$ vmstat -S
kthr memory page disk faults cpu
r b w swap free si so pi po fr de sr dd fo sl -- in sy cs us sy id
0 0 0 862608 364792 0 0 1 0 0 0 0 0 0 0 406 394 213 1 0 99
```

以下列表介绍了交换统计信息字段。有关其他字段的说明，请参见[表 13-1](#)。

- si 每秒换入的平均 LWP 数
- so 换出的完整进程数

---

注 - **vmstat** 命令会截断 **si** 和 **so** 字段的输出。应使用 **sar** 命令显示更精确的交换统计信息记录。

---

▼ 如何显示每台设备的中断次数 (vmstat -i)

- 运行 **vmstat -i** 命令，以显示每台设备的中断次数。

示例 13-2 显示每台设备的中断次数

以下示例显示 **vmstat -i** 命令的输出。

```
$ vmstat -i
interrupt total rate

clock 52163269 100
esp0 2600077 4
zsc0 25341 0
zsc1 48917 0
cgsixc0 459 0
lec0 400882 0
fdc0 14 0
bppc0 0 0
audiocs0 0 0

Total 55238959 105
```

## 显示磁盘使用率信息 ( iostat)

使用 `iostat` 命令可以报告有关磁盘输入和输出的统计信息，以及生成对吞吐量、使用率、队列长度、事务速率和服务时间的测量值。有关此命令的详细说明，请参阅 [iostat\(1M\)](#) 手册页。

### ▼ 如何显示磁盘使用率信息 (iostat)

- 使用 `iostat` 命令和时间间隔秒数可以显示磁盘使用率信息。

```
$ iostat 5
 tty fd0 sd3 nfs1 nfs31 cpu
tin tout kps tps serv kps tps serv kps tps serv kps tps serv us sy wt id
 0 1 0 0 410 3 0 29 0 0 9 3 0 47 4 2 0 94
```

输出的第一行显示自上次引导系统以来的统计信息。后面的每行显示间隔的统计信息。缺省情况下会显示终端 (tty)、磁盘 (fd 以及 sd) 和 CPU (cpu) 的统计信息。

### 示例 13-3 显示磁盘使用率信息

以下示例显示每隔 5 秒收集的磁盘统计信息。

```
$ iostat 5
 tty sd0 sd6 nfs1 nfs49 cpu
tin tout kps tps serv kps tps serv kps tps serv kps tps serv us sy wt id
 0 0 1 0 49 0 0 0 0 0 0 0 0 15 0 0 0 100
 0 47 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 100
 0 16 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 100
 0 16 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 100
 0 16 44 6 132 0 0 0 0 0 0 0 0 0 0 0 1 99
 0 16 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 100
 0 16 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 100
 0 16 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 100
 0 16 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 100
```

```
0 16 0 0 0 0 0 0 0 0 0 0 0 0 0 0 100
0 16 3 1 23 0 0 0 0 0 0 0 0 0 0 0 1 99
0 16 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 100
0 16 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 100
0 16 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 100
```

下表介绍了 `iostat n` 命令输出中的字段。

| 设备类型 | 字段名  | 说明            |
|------|------|---------------|
| 终端   | 设备类型 |               |
|      | tin  | 终端输入队列中的字符数   |
|      | tout | 终端输出队列中的字符数   |
| 磁盘   | 设备类型 |               |
|      | bps  | 每秒块数          |
|      | tps  | 每秒事务数         |
|      | serv | 平均服务时间，以毫秒为单位 |
| CPU  | 设备类型 |               |
|      | us   | 在用户模式下        |
|      | sy   | 在系统模式下        |
|      | wt   | 等待 I/O        |
|      | id   | 空闲            |

▼ 如何显示扩展磁盘统计信息 (iostat -xtc)

- 运行 `iostat -xtc` 命令，以显示扩展磁盘统计信息。

```
$ iostat -xtc
extended device statistics
device r/s w/s kr/s kw/s wait actv svc_t %w %b tty cpu
 tin tout us sy wt id
fd0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0 0 0 0 0 100
sd0 0.0 0.0 0.4 0.4 0.0 0.0 49.5 0 0
sd6 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0 0
nfs1 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0 0
nfs49 0.0 0.0 0.0 0.0 0.0 0.0 15.1 0 0
nfs53 0.0 0.0 0.4 0.0 0.0 0.0 24.5 0 0
nfs54 0.0 0.0 0.0 0.0 0.0 0.0 6.3 0 0
nfs55 0.0 0.0 0.0 0.0 0.0 0.0 4.9 0 0
```

`iostat -xtc` 命令对每个磁盘显示一行输出。以下列表介绍了输出字段。

r/s          每秒读取次数

w/s          每秒写入次数

|       |                  |
|-------|------------------|
| kr/s  | 每秒读取的千字节数        |
| kw/s  | 每秒写入的千字节数        |
| wait  | 等待服务的平均事务数（队列长度） |
| actv  | 处于活动服务状态的平均事务数   |
| svc_t | 平均服务时间，以毫秒为单位    |
| %w    | 队列不为空的时间百分比      |
| %b    | 磁盘繁忙的时间百分比       |

## 显示磁盘空间统计信息 (df)

使用 `df` 命令可以显示挂载的每个磁盘中的可用磁盘空间量。`df` 报告的可用磁盘空间只反映全部容量的 90%，因为报告统计信息留出总可用空间的 10% 以上。此头空间通常保持为空，以实现更好的性能。

`df` 命令实际报告的磁盘空间百分比是已用空间除以可用空间。

如果文件系统超出容量的 90%，则可使用 `cp` 命令将文件转移至具有可用空间的磁盘。或者，使用 `tar` 或 `cpio` 命令将文件转移至磁带。也可以删除文件。

有关此命令的详细说明，请参见 [df\(1M\)](#) 手册页。

### ▼ 如何显示磁盘空间信息 (df -k)

- 使用 `df -k` 命令以千字节为单位显示磁盘空间信息。

```
$ df -k
Filesystem kbytes used avail capacity Mounted on
/dev/dsk/c0t3d0s0 192807 40231 133296 24% /
```

#### 示例 13-4 显示文件系统信息

以下示例显示 `df -k` 命令的输出。

```
$ df -k
Filesystem kbytes used avail capacity Mounted on
/dev/dsk/c0t0d0s0 254966 204319 25151 90% /
/devices 0 0 0 0% /devices
ctfs 0 0 0 0% /system/contract
proc 0 0 0 0% /proc
mnttab 0 0 0 0% /etc/mnttab
swap 496808 376 496432 1% /etc/svc/volatile
objfs 0 0 0 0% /system/object
```

|                   |         |         |        |     |              |
|-------------------|---------|---------|--------|-----|--------------|
| /dev/dsk/c0t0d0s6 | 3325302 | 3073415 | 218634 | 94% | /usr         |
| fd                | 0       | 0       | 0      | 0%  | /dev/fd      |
| swap              | 496472  | 40      | 496432 | 1%  | /var/run     |
| swap              | 496472  | 40      | 496432 | 1%  | /tmp         |
| /dev/dsk/c0t0d0s5 | 13702   | 1745    | 10587  | 15% | /opt         |
| /dev/dsk/c0t0d0s7 | 9450    | 1045    | 7460   | 13% | /export/home |

下表介绍了 `df -k` 命令的输出。

| 字段名        | 说明               |
|------------|------------------|
| kbytes     | 文件系统中可用空间的总大小    |
| used       | 已用空间量            |
| avail      | 可用空间量            |
| capacity   | 已用空间量，表示为总容量的百分比 |
| mounted on | 挂载点              |

## 监视系统活动（任务列表）

| 任务          | 说明                                                                                                            | 参考                                               |
|-------------|---------------------------------------------------------------------------------------------------------------|--------------------------------------------------|
| 检查文件访问。     | 使用带有 <code>-a</code> 选项的 <code>sar</code> 命令显示文件访问操作状态。                                                       | <a href="#">第 176 页中的“如何检查文件访问 (sar -a)”</a>     |
| 检查缓冲区活动。    | 使用带有 <code>-b</code> 选项的 <code>sar</code> 命令显示缓冲区活动统计信息。                                                      | <a href="#">第 176 页中的“如何检查缓冲区活动 (sar -b)”</a>    |
| 检查系统调用统计信息。 | 使用带有 <code>-c</code> 选项的 <code>sar</code> 命令显示系统调用统计信息。                                                       | <a href="#">第 178 页中的“如何检查系统调用统计信息 (sar -c)”</a> |
| 检查磁盘活动。     | 使用带有 <code>-d</code> 选项的 <code>sar</code> 命令检查磁盘活动。                                                           | <a href="#">第 179 页中的“如何检查磁盘活动 (sar -d)”</a>     |
| 检查页出和内存。    | 使用带有 <code>-g</code> 选项的 <code>sar</code> 命令显示页出内存释放活动。                                                       | <a href="#">第 180 页中的“如何检查页出和内存 (sar -g)”</a>    |
| 检查内核内存分配。   | 内核内存分配 (Kernel Memory Allocation, KMA) 允许内核子系统根据需要分配和释放内存。使用带有 <code>-k</code> 选项的 <code>sar</code> 命令检查 KMA。 | <a href="#">第 182 页中的“如何检查内核内存分配 (sar -k)”</a>   |
| 检查进程间通信。    | 使用带有 <code>-m</code> 选项的 <code>sar</code> 命令报告进程间通信活动。                                                        | <a href="#">第 183 页中的“如何检查进程间通信 (sar -m)”</a>    |
| 检查页入活动。     | 使用带有 <code>-p</code> 选项的 <code>sar</code> 命令报告页入活动。                                                           | <a href="#">第 184 页中的“如何检查页入活动 (sar -p)”</a>     |

| 任务          | 说明                                                                                                                                                                    | 参考                               |
|-------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------|
| 检查队列活动。     | 使用带有 -q 选项的 sar 命令检查以下内容： <ul style="list-style-type: none"><li>■ 占用队列时的平均队列长度</li><li>■ 占用队列时的时间百分比</li></ul>                                                        | 第 185 页中的“如何检查队列活动 (sar -q)”     |
| 检查未使用的内存。   | 使用带有 -r 选项的 sar 命令报告当前使用的内存页数和交换文件磁盘块数。                                                                                                                               | 第 186 页中的“如何检查未使用的内存 (sar -r)”   |
| 检查 CPU 使用率。 | 使用带有 -u 选项的 sar 命令显示 CPU 使用率统计信息。                                                                                                                                     | 第 187 页中的“如何检查 CPU 使用率 (sar -u)” |
| 检查系统表状态。    | 使用带有 -v 选项的 sar 命令报告有关以下系统表的状态： <ul style="list-style-type: none"><li>■ 进程</li><li>■ inode</li><li>■ 文件</li><li>■ 共享内存记录</li></ul>                                    | 第 188 页中的“如何检查系统表状态 (sar -v)”    |
| 检查交换活动。     | 使用带有 -w 选项的 sar 命令检查交换活动。                                                                                                                                             | 第 189 页中的“如何检查交换活动 (sar -w)”     |
| 检查终端活动。     | 使用带有 -y 选项的 sar 命令监视终端设备活动。                                                                                                                                           | 第 190 页中的“如何检查终端活动 (sar -y)”     |
| 检查总体系统性能。   | sar -A 命令可通过显示所有选项的统计信息提供总体系统性能信息。                                                                                                                                    | 第 191 页中的“如何检查总体系统性能 (sar -A)”   |
| 设置自动数据收集。   | 要设置系统以自动收集数据并运行 sar 命令，请执行以下操作： <ul style="list-style-type: none"><li>■ 运行 svcadm enable system/sar:default 命令</li><li>■ 编辑 /var/spool/cron/crontabs/sys 文件</li></ul> | 第 194 页中的“如何设置自动数据收集”            |

## 监视系统活动 (sar)

使用 sar 命令可执行以下任务：

- 组织并查看有关系统活动的数据。
- 根据特殊请求访问系统活动数据。
- 生成自动报告以测量和监视系统性能，并生成特殊请求报告以确定特定性能问题。有关如何设置要在系统上运行的 sar 命令的信息以及相应工具的说明，请参见第 191 页中的“自动收集系统活动数据 (sar)”。

有关此命令的详细说明，请参见 [sar\(1\)](#) 手册页。

## ▼ 如何检查文件访问 (sar -a)

- 使用 **sar -a** 命令可显示文件访问操作统计信息。

```
$ sar -a

SunOS balmyday 5.10 s10_51 sun4u 03/18/2004

00:00:00 iget/s namei/s dirbk/s
01:00:00 0 3 0
02:00:00 0 3 0
03:00:00 0 3 0
04:00:00 0 3 0
05:00:00 0 3 0
06:00:00 0 3 0
07:00:00 0 3 0
08:00:00 0 3 0
08:20:01 0 3 0
08:40:00 0 3 0
09:00:00 0 3 0
09:20:01 0 10 0
09:40:01 0 1 0
10:00:02 0 5 0

Average 0 4 0
```

以下列表介绍了 **sar -a** 命令报告的操作系统例程的字段名和说明。

|         |                                                                                             |
|---------|---------------------------------------------------------------------------------------------|
| iget/s  | 对不位于目录名称查找高速缓存 (Directory Name Look-up Cache, DNLC) 中的 inode 发出的请求数。                        |
| namei/s | 每秒搜索的文件系统路径数。如果 namei 在 DNLC 中找不到目录名称，它会调用 iget 以获取文件或目录的 inode。因此，大多数 igets 都是 DNLC 遗漏的结果。 |
| dirbk/s | 每秒发出的目录块读取数。                                                                                |

这些操作系统例程的报告值越大，内核访问用户文件所用的时间就越多。时间的长短将反映程序和应用程序使用文件系统的程度。-a 选项有助于查看磁盘与应用程序的相关情况。

## ▼ 如何检查缓冲区活动 (sar -b)

- 使用 **sar -b** 命令可显示缓冲区活动统计信息。

缓冲区用于高速缓存元数据。元数据包括 inode、柱面组块和间接块。

```
$ sar -b
00:00:00 bread/s lread/s %rcache bwrit/s lwrit/s %wcache pread/s pwrit/s
01:00:00 0 0 100 0 0 55 0 0
```



示例13-5 检查缓冲区活动 (sar -b)

下面的 sar -b 命令输出示例说明 %rcache 和 %wcache 缓冲区未引起速率下降。所有数据都在可接受的限制范围内。

```
$ sar -b

SunOS balmyday 5.10 s10_51 sun4u 03/18/2004

00:00:04 bread/s lread/s %rcache bwrit/s lwrit/s %wcache pread/s pwrit/s
01:00:00 0 0 100 0 0 94 0 0
02:00:01 0 0 100 0 0 94 0 0
03:00:00 0 0 100 0 0 92 0 0
04:00:00 0 1 100 0 1 94 0 0
05:00:00 0 0 100 0 0 93 0 0
06:00:00 0 0 100 0 0 93 0 0
07:00:00 0 0 100 0 0 93 0 0
08:00:00 0 0 100 0 0 93 0 0
08:20:00 0 1 100 0 1 94 0 0
08:40:01 0 1 100 0 1 93 0 0
09:00:00 0 1 100 0 1 93 0 0
09:20:00 0 1 100 0 1 93 0 0
09:40:00 0 2 100 0 1 89 0 0
10:00:00 0 9 100 0 5 92 0 0
10:20:00 0 0 100 0 0 68 0 0
10:40:00 0 1 98 0 1 70 0 0
11:00:00 0 1 100 0 1 75 0 0

Average 0 1 100 0 1 91 0 0
```

下表介绍了 -b 选项显示的缓冲区活动。

| 字段名     | 说明                                                 |
|---------|----------------------------------------------------|
| bread/s | 从磁盘提交至高速缓存存储区的每秒平均读取数                              |
| lread/s | 每秒从高速缓存存储区进行的平均逻辑读取数                               |
| %rcache | 在高速缓存存储区中找到的逻辑读取的分数（100 % 减去 bread/s 与 lread/s 之比） |
| bwrit/s | 每秒平均从高速缓存存储区写入磁盘的物理块数（512 块）                       |
| lwrit/s | 每秒平均对高速缓存存储区进行的逻辑写入数                               |
| %wcache | 在高速缓存存储区中找到的逻辑写入的分数（100 % 减去 bwrit/s 与 lwrit/s 之比） |
| pread/s | 每秒平均使用字符设备接口的物理读取数                                 |
| pwrit/s | 每秒平均使用字符设备接口的物理写入请求数                               |

最重要的项是高速缓存命中率%rcache和%wcache。这两项用于度量系统缓冲的有效性。如果%rcache低于90%或者%wcache低于65%，则可通过增加缓冲区空间来改善性能。

▼ 如何检查系统调用统计信息 (sar -c)

- 使用 sar -c 命令可显示系统调用统计信息。

```
$ sar -c
00:00:00 scall/s sread/s swrit/s fork/s exec/s rchar/s wchar/s
01:00:00 38 2 2 0.00 0.00 149 120
```

示例 13-6 检查系统调用统计信息 (sar -c)

以下示例显示 sar -c 命令的输出。

```
$ sar -c

SunOS balmyday 5.10 s10_51 sun4u 03/18/2004

00:00:04 scall/s sread/s swrit/s fork/s exec/s rchar/s wchar/s
01:00:00 89 14 9 0.01 0.00 2906 2394
02:00:01 89 14 9 0.01 0.00 2905 2393
03:00:00 89 14 9 0.01 0.00 2908 2393
04:00:00 90 14 9 0.01 0.00 2912 2393
05:00:00 89 14 9 0.01 0.00 2905 2393
06:00:00 89 14 9 0.01 0.00 2905 2393
07:00:00 89 14 9 0.01 0.00 2905 2393
08:00:00 89 14 9 0.01 0.00 2906 2393
08:20:00 90 14 9 0.01 0.01 2914 2395
08:40:01 90 14 9 0.01 0.00 2914 2396
09:00:00 90 14 9 0.01 0.01 2915 2396
09:20:00 90 14 9 0.01 0.01 2915 2396
09:40:00 880 207 156 0.08 0.08 26671 9290
10:00:00 2020 530 322 0.14 0.13 57675 36393
10:20:00 853 129 75 0.02 0.01 10500 8594
10:40:00 2061 524 450 0.08 0.08 579217 567072
11:00:00 1658 404 350 0.07 0.06 1152916 1144203

Average 302 66 49 0.02 0.01 57842 55544
```

下表介绍了 -c 选项报告的系统调用类别。通常，读取和写入占系统调用总数的一半。但是，该百分比会因系统所执行的活动而产生极大的变化。

| 字段名     | 说明                                                |
|---------|---------------------------------------------------|
| scall/s | 每秒中所有类型的系统调用数，在具有 4 到 6 位用户的系统中，通常每秒大约有 30 个系统调用。 |
| sread/s | 每秒的 read 系统调用数。                                   |

| 字段名     | 说明                                                                         |
|---------|----------------------------------------------------------------------------|
| swrit/s | 每秒的 write 系统调用数。                                                           |
| fork/s  | 每秒的 fork 系统调用数，在具有 4 到 6 位用户的系统中，每秒中大约有 0.5 个该系统调用。如果正在运行 Shell 脚本，此数字会增加。 |
| exec/s  | 每秒的 exec 系统调用数。如果 exec/s 除以 fork/s 的结果大于 3，请确定是否存在无效的 PATH 变量。             |
| rchar/s | 每秒由 read 系统调用传送的字符数（字节）。                                                   |
| wchar/s | 每秒由 write 系统调用传送的字符数（字节）。                                                  |

▼ 如何检查磁盘活动 (sar -d)

- 使用 sar -d 命令可显示磁盘活动统计信息。

```
$ sar -d
00:00:00 device %busy avque r+w/s blks/s await aserv
```

示例 13-7 检查磁盘活动

此缩写示例演示了 sar -d 命令的输出。

```
$ sar -d
SunOS balmyday 5.10 s10_51 sun4u 03/18/2004

12:36:32 device %busy avque r+w/s blks/s await aserv
12:40:01 dad1 15 0.7 26 399 18.1 10.0
 dad1,a 15 0.7 26 398 18.1 10.0
 dad1,b 0 0.0 0 1 1.0 3.0
 dad1,c 0 0.0 0 0 0.0 0.0
 dad1,h 0 0.0 0 0 0.0 6.0
 fd0 0 0.0 0 0 0.0 0.0
 nfs1 0 0.0 0 0 0.0 0.0
 nfs2 1 0.0 1 12 0.0 13.2
 nfs3 0 0.0 0 2 0.0 1.9
 nfs4 0 0.0 0 0 0.0 7.0
 nfs5 0 0.0 0 0 0.0 57.1
 nfs6 1 0.0 6 125 4.3 3.2
 nfs7 0 0.0 0 0 0.0 6.0
 sd1 0 0.0 0 0 0.0 5.4
 ohci0,bu 0 0.0 0 0 0.0 0.0
 ohci0,ct 0 0.0 0 0 0.0 0.0
 ohci0,in 0 0.0 7 0 0.0 0.0
 ohci0,is 0 0.0 0 0 0.0 0.0
 ohci0,to 0 0.0 7 0 0.0 0.0
```

下表介绍了 -d 选项报告的磁盘设备活动。

| 字段名    | 说明                                                    |
|--------|-------------------------------------------------------|
| device | 监视的磁盘设备的名称。                                           |
| %busy  | 设备忙于为传送请求提供服务的时间份额。                                   |
| avque  | 设备忙于为传送请求提供服务期间的平均请求数。                                |
| r+w/s  | 每秒对设备进行的读取和写入传送数。                                     |
| blks/s | 每秒传送给设备的 512 字节块的数量。                                  |
| await  | 传送请求在队列中空闲等待的平均时间，以毫秒为单位。仅当队列被占用时才测量此时间。              |
| avserv | 设备完成传送请求所需的平均时间，以毫秒为单位。对于磁盘而言，此值包括查找时间、旋转延迟时间和数据传送时间。 |

请注意，在队列不为空时测量队列长度和等待时间。当 %busy 很小时，如果队列和服务时间很大，则可能表示系统进行周期性的努力，以确保将警报块快速写入磁盘。

▼ 如何检查页出和内存 (sar -g)

- 使用 sar -g 命令可显示平均页出和内存释放活动。

```
$ sar -g
00:00:00 pgout/s ppgout/s pgfree/s pgscan/s %ufs_ipf
01:00:00 0.00 0.00 0.00 0.00 0.00
```

sar -g 命令的输出可以明确指示是否需要更多内存。使用 ps -elf 命令显示 page 守护程序使用的周期数。如果周期数很大，并且 pgfree/s 和 pgscan/s 字段的值也很大，则表明内存不足。

sar -g 命令还可表明是否回收 inode 的速率过快而引起可重用页丢失。

示例 13-8 检查页出和内存 (sar -g)

以下示例显示 sar -g 命令的输出。

```
$ sar -g

SunOS balmyday 5.10 s10_51 sun4u 03/18/2004

00:00:00 pgout/s ppgout/s pgfree/s pgscan/s %ufs_ipf
01:00:00 0.00 0.00 0.00 0.00 0.00
02:00:00 0.01 0.01 0.01 0.00 0.00
03:00:00 0.00 0.00 0.00 0.00 0.00
04:00:00 0.00 0.00 0.00 0.00 0.00
05:00:00 0.00 0.00 0.00 0.00 0.00
06:00:00 0.00 0.00 0.00 0.00 0.00
```

|          |      |      |       |       |      |
|----------|------|------|-------|-------|------|
| 07:00:00 | 0.00 | 0.00 | 0.00  | 0.00  | 0.00 |
| 08:00:00 | 0.00 | 0.00 | 0.00  | 0.00  | 0.00 |
| 08:20:01 | 0.00 | 0.00 | 0.00  | 0.00  | 0.00 |
| 08:40:00 | 0.00 | 0.00 | 0.00  | 0.00  | 0.00 |
| 09:00:00 | 0.00 | 0.00 | 0.00  | 0.00  | 0.00 |
| 09:20:01 | 0.05 | 0.52 | 1.62  | 10.16 | 0.00 |
| 09:40:01 | 0.03 | 0.44 | 1.47  | 4.77  | 0.00 |
| 10:00:02 | 0.13 | 2.00 | 4.38  | 12.28 | 0.00 |
| 10:20:03 | 0.37 | 4.68 | 12.26 | 33.80 | 0.00 |
| Average  | 0.02 | 0.25 | 0.64  | 1.97  | 0.00 |

下表介绍了 -g 选项的输出。

| 字段名      | 说明                                                                                                                                       |
|----------|------------------------------------------------------------------------------------------------------------------------------------------|
| pgout/s  | 每秒的页出请求数。                                                                                                                                |
| ppgout/s | 每秒调出的页的实际数量。单个页出请求可能涉及多个页的调出。                                                                                                            |
| pgfree/s | 每秒放置在可用列表中的页数。                                                                                                                           |
| pgscan/s | page 守护程序每秒扫描的页数。如果此值很大，则表明 page 守护程序花费大量时间来检查可用内存。此情况暗示，可能需要更多内存。                                                                       |
| %ufs_ipf | 具有关联的可重用页的 iget 从可用列表中取消的 ufs inode 的百分比。这些页面被刷新，并且不能由进程回收。因此，此字段表示具有页面刷新的 igets 的百分比。如果该值很大，则表明 inode 的可用列表页面密集，并且可能需要增加 ufs inode 的数量。 |

## 检查内核内存分配

KMA 允许内核子系统根据需要分配和释放内存。

KMA 并不是静态分配在峰值载荷下预计所需的最大内存量，而是将内存请求划分为三个类别：

- 小型（少于 256 字节）
- 大型（512 字节至 4 千字节）
- 超大型（大于 4 千字节）

KMA 保留两个内存池，以满足小型和大型请求。超大型请求则通过从系统页面分配器中分配内存来满足。

如果您所检查的系统用来编写使用 KMA 资源的驱动程序或 STREAMS，则 `sar -k` 命令可能很有用。否则，您可能不需要它所提供的信息。使用 KMA 资源但不一定在退出前返回资源的所有驱动程序或模块都可能产生内存泄漏。内存泄漏会导致 KMA 分配的内

存量随事件而增加。因此，如果 sar -k 命令的 alloc 字段随时间稳定增加，则可能存在内存泄漏。表明存在内存泄漏的另一种情况是请求失败。如果出现此问题，内存泄漏很可能导致 KMA 无法保留和分配内存。

如果似乎存在内存泄漏，则应检查可能从 KMA 请求内存但未返回内存的所有驱动程序或 STREAMS。

▼ 如何检查内核内存分配 (sar -k)

- 使用 sar -k 命令可报告内核内存分配器 (Kernel Memory Allocator, KMA) 的以下活动。

```
$ sar -k
00:00:00 sml_mem alloc fail lg_mem alloc fail ovsz_alloc fail
01:00:00 2523136 1866512 0 18939904 14762364 0 360448 0
02:00:02 2523136 1861724 0 18939904 14778748 0 360448 0
```

示例 13-9 检查内核内存分配 (sar -k)

下面是 sar -k 输出的缩写示例。

```
$ sar -k

SunOS balmyday 5.10 s10_51 sun4u 03/18/2004

00:00:04 sml_mem alloc fail lg_mem alloc fail ovsz_alloc fail
01:00:00 6119744 4852865 0 60243968 54334808 156 9666560 0
02:00:01 6119744 4853057 0 60243968 54336088 156 9666560 0
03:00:00 6119744 4853297 0 60243968 54335760 156 9666560 0
04:00:00 6119744 4857673 0 60252160 54375280 156 9666560 0
05:00:00 6119744 4858097 0 60252160 54376240 156 9666560 0
06:00:00 6119744 4858289 0 60252160 54375608 156 9666560 0
07:00:00 6119744 4858793 0 60252160 54442424 156 9666560 0
08:00:00 6119744 4858985 0 60252160 54474552 156 9666560 0
08:20:00 6119744 4858169 0 60252160 54377400 156 9666560 0
08:40:01 6119744 4857345 0 60252160 54376880 156 9666560 0
09:00:00 6119744 4859433 0 60252160 54539752 156 9666560 0
09:20:00 6119744 4858633 0 60252160 54410920 156 9666560 0
09:40:00 6127936 5262064 0 60530688 55619816 156 9666560 0
10:00:00 6545728 5823137 0 62996480 58391136 156 9666560 0
10:20:00 6545728 5758997 0 62996480 57907400 156 9666560 0
10:40:00 6734144 6035759 0 64389120 59743064 156 10493952 0
11:00:00 6996288 6394872 0 65437696 60935936 156 10493952 0

Average 6258044 5150556 0 61138340 55609004 156 9763900 0
```

下表介绍了 -k 选项的输出。

| 字段名     | 说明                                             |
|---------|------------------------------------------------|
| sml_mem | KMA 在小型内存请求池中可用的内存量，以字节为单位。在此池中，小型请求小于 256 字节。 |

| 字段名        | 说明                                                      |
|------------|---------------------------------------------------------|
| alloc      | KMA 已从其小型内存请求池向小型内存请求分配的内存量，以字节为单位。                     |
| fail       | 请求少量内存并失败的请求数。                                          |
| lg_mem     | KMA 在大型内存请求池中可用的内存量，以字节为单位。在此池中，大型请求介于 512 字节到 4 千字节之间。 |
| alloc      | KMA 已从其大型内存请求池向大型内存请求分配的内存量，以字节为单位。                     |
| fail       | 请求大量内存并失败的请求数。                                          |
| ovsz_alloc | 为大于 4 千字节的超大型请求分配的内存量。这些请求可通过页面分配器来满足。因此，不存在池。          |
| fail       | 因请求超大量内存而失败的请求数。                                        |

▼ 如何检查进程间通信 (sar -m)

- 使用 sar -m 命令可报告进程间通信活动。

```
$ sar -m
00:00:00 msg/s sema/s
01:00:00 0.00 0.00
```

除非运行使用消息或信号的应用程序，否则这些数字通常都为零 (0.00)。

以下列表介绍了 -m 选项的输出。

msg/s        每秒的消息操作（发送和接收）数  
sema/s       每秒的信号操作数

示例 13-10 检查进程间通信 (sar -m)

以下缩写示例显示 sar -m 命令的输出。

```
$ sar -m

SunOS balmyday 5.10 s10_51 sun4u 03/18/2004

00:00:00 msg/s sema/s
01:00:00 0.00 0.00
02:00:02 0.00 0.00
03:00:00 0.00 0.00
04:00:00 0.00 0.00
05:00:01 0.00 0.00
06:00:00 0.00 0.00

Average 0.00 0.00
```

▼ 如何检查页入活动 (sar -p)

- 使用 **sar -p** 命令可报告页入活动，其中包括保护错误和转换错误。

```
$ sar -p
00:00:00 atch/s pgin/s ppgin/s pflt/s vflt/s slock/s
01:00:00 0.07 0.00 0.00 0.21 0.39 0.00
```

示例 13-11 检查页入活动 (sar -p)

以下示例显示 **sar -p** 命令的输出。

```
$ sar -p

SunOS balmyday 5.10 s10_51 sun4u 03/18/2004

00:00:04 atch/s pgin/s ppgin/s pflt/s vflt/s slock/s
01:00:00 0.09 0.00 0.00 0.78 2.02 0.00
02:00:01 0.08 0.00 0.00 0.78 2.02 0.00
03:00:00 0.09 0.00 0.00 0.81 2.07 0.00
04:00:00 0.11 0.01 0.01 0.86 2.18 0.00
05:00:00 0.08 0.00 0.00 0.78 2.02 0.00
06:00:00 0.09 0.00 0.00 0.78 2.02 0.00
07:00:00 0.08 0.00 0.00 0.78 2.02 0.00
08:00:00 0.09 0.00 0.00 0.78 2.02 0.00
08:20:00 0.11 0.00 0.00 0.87 2.24 0.00
08:40:01 0.13 0.00 0.00 0.90 2.29 0.00
09:00:00 0.11 0.00 0.00 0.88 2.24 0.00
09:20:00 0.10 0.00 0.00 0.88 2.24 0.00
09:40:00 2.91 1.80 2.38 4.61 17.62 0.00
10:00:00 2.74 2.03 3.08 8.17 21.76 0.00
10:20:00 0.16 0.04 0.04 1.92 2.96 0.00
10:40:00 2.10 2.50 3.42 6.62 16.51 0.00
11:00:00 3.36 0.87 1.35 3.92 15.12 0.00

Average 0.42 0.22 0.31 1.45 4.00 0.00
```

下表介绍了通过 **-p** 选项报告的统计信息。

| 字段名     | 说明                                                                                   |
|---------|--------------------------------------------------------------------------------------|
| atch/s  | 每秒通过回收当前在内存中的页来满足的缺页数（每秒附加数）。例如从可用列表中回收无效的页，以及共享其他进程当前正在使用的文本页。例如，两个或多个进程同时访问同一程序文本。 |
| pgin/s  | 文件系统每秒接收页入请求的次数。                                                                     |
| ppgin/s | 每秒调进的页数。单个页入请求（例如软件锁定请求，请参见 <b>slock/s</b> ）或块大小很大时可能涉及多个页的调进。                       |



| 字段名    | 说明                                                                              |
|--------|---------------------------------------------------------------------------------|
| pflt/s | 因保护错误引起的缺页数。保护错误实例表明非法访问页面和“写复制”。通常，此数目主要包含“写复制”。                               |
| vflt/s | 每秒的地址转换缺页数。这些错误称为有效性错误。当给定虚拟地址的有效进程表项不存在时，会发生有效性错误。                             |
| slck/s | 每秒内由要求物理 I/O 的软件锁定请求引起的错误数。从磁盘向内存传送数据时，就会出现软件锁定请求。系统锁定了要接收数据的页，因此其他进程无法请求和使用该页。 |

▼ 如何检查队列活动 (sar -q)

- 使用 sar -q 命令可报告以下信息：
  - 队列被占用时的平均队列长度。
  - 队列处于占用状态的时间百分比。

```
$ sar -q
00:00:00 runq-sz %runocc swpq-sz %swpocc
```

以下列表介绍了 -q 选项的输出。

- runq-sz      内存中等待 CPU 以便运行的内核线程数。通常，此值应小于 2。如果此值持续偏高，则表明系统可能计算密集 (CPU-bound)。
- %runocc      占用分发队列的时间百分比。
- swpq-sz      换出进程的平均数量。
- %swpocc      进程处于换出状态的时间百分比。

示例 13-12 检查队列活动

以下示例显示 sar -q 命令的输出。如果 %runocc 值较大（大于 90%）并且 runq-sz 值大于 2，则表明 CPU 负载较大，并且响应变慢。在此情况下，可能需要提供附加的 CPU 容量，才能获得可接受的系统响应速度。

```
sar -q
SunOS system2 5.10 Generic_142909-13 sun4u 06/28/2010

00:00:00 runq-sz %runocc swpq-sz %swpocc
01:00:00 1.0 7 0.0 0
02:00:00 1.0 7 0.0 0
03:00:00 1.0 7 0.0 0
04:00:00 1.0 7 0.0 0
05:00:00 1.0 6 0.0 0
06:00:00 1.0 7 0.0 0
```

Average      1.0      7      0.0      0

## ▼ 如何检查未使用的内存 (sar -r)

- 使用 **sar -r** 命令可报告当前未使用的内存页数和交换文件磁盘块数。

```
$ sar -r
00:00:00 freemem freeswap
01:00:00 2135 401922
```

以下列表介绍了 **-r** 选项的输出：

**freemem**      在该命令采样的时间间隔内可供用户进程使用的平均内存页数。页面大小与计算机有关。

**freeswap**      可用于页交换的 512 字节磁盘块数。

### 示例 13-13 检查未使用的内存 (sar -r)

以下示例显示 **sar -r** 命令的输出。

```
$ sar -r

SunOS balmyday 5.10 s10_51 sun4u 03/18/2004

00:00:04 freemem freeswap
01:00:00 44717 1715062
02:00:01 44733 1715496
03:00:00 44715 1714746
04:00:00 44751 1715403
05:00:00 44784 1714743
06:00:00 44794 1715186
07:00:00 44793 1715159
08:00:00 44786 1714914
08:20:00 44805 1715576
08:40:01 44797 1715347
09:00:00 44761 1713948
09:20:00 44802 1715478
09:40:00 41770 1682239
10:00:00 35401 1610833
10:20:00 34295 1599141
10:40:00 33943 1598425
11:00:00 30500 1561959

Average 43312 1699242
```

## ▼ 如何检查 CPU 使用率 (sar -u)

- 使用 **sar -u** 命令可显示 CPU 使用率统计信息。

```
$ sar -u
00:00:00 %usr %sys %wio %idle
01:00:00 0 0 0 100
```

没有任何选项的 **sar** 命令与 **sar -u** 命令等效。在任意给定时刻，处理器都会处于繁忙或空闲状态。繁忙时，处理器可能处于用户模式或系统模式。空闲时，处理器可能在等待 I/O 完成，或“静止”而不执行任何操作。

以下列表介绍了 **-u** 选项的输出：

- %usr**      列出处理器处于用户模式的时间百分比。
  - %sys**      列出处理器处于系统模式的时间百分比。
  - %wio**      列出处理器空闲并等待 I/O 完成的时间百分比。
  - %idle**     列出处理器空闲并且未等待 I/O 的时间百分比。
- %wio** 值越大，通常表示磁盘速率变慢。

### 示例 13-14    检查 CPU 使用率 (sar -u)

以下示例显示 **sar -u** 命令的输出。

```
$ sar -u

SunOS balmyday 5.10 s10_51 sun4u 03/18/2004

00:00:04 %usr %sys %wio %idle
01:00:00 0 0 0 100
02:00:01 0 0 0 100
03:00:00 0 0 0 100
04:00:00 0 0 0 100
05:00:00 0 0 0 100
06:00:00 0 0 0 100
07:00:00 0 0 0 100
08:00:00 0 0 0 100
08:20:00 0 0 0 99
08:40:01 0 0 0 99
09:00:00 0 0 0 99
09:20:00 0 0 0 99
09:40:00 4 1 0 95
10:00:00 4 2 0 94
10:20:00 1 1 0 98
10:40:00 18 3 0 79
11:00:00 25 3 0 72

Average 2 0 0 98
```

## ▼ 如何检查系统表状态 (sar -v)

- 使用 **sar -v** 命令可报告进程表、inode 表、文件表和共享内存记录表的状态。

```
$ sar -v
00:00:00 proc-sz ov inod-sz ov file-sz ov lock-sz
01:00:00 43/922 0 2984/4236 0 322/322 0 0/0
```

### 示例 13-15 检查系统表状态 (sar -v)

以下缩写示例显示 **sar -v** 命令的输出。此示例表明，所有表都足够大，因此没有溢出。这些表都基于物理内存量进行动态分配。

```
$ sar -v

SunOS balmyday 5.10 s10_51 sun4u 03/18/2004

00:00:04 proc-sz ov inod-sz ov file-sz ov lock-sz
01:00:00 69/8010 0 3476/34703 0 0/0 0 0/0
02:00:01 69/8010 0 3476/34703 0 0/0 0 0/0
03:00:00 69/8010 0 3476/34703 0 0/0 0 0/0
04:00:00 69/8010 0 3494/34703 0 0/0 0 0/0
05:00:00 69/8010 0 3494/34703 0 0/0 0 0/0
06:00:00 69/8010 0 3494/34703 0 0/0 0 0/0
07:00:00 69/8010 0 3494/34703 0 0/0 0 0/0
08:00:00 69/8010 0 3494/34703 0 0/0 0 0/0
08:20:00 69/8010 0 3494/34703 0 0/0 0 0/0
08:40:01 69/8010 0 3494/34703 0 0/0 0 0/0
09:00:00 69/8010 0 3494/34703 0 0/0 0 0/0
09:20:00 69/8010 0 3494/34703 0 0/0 0 0/0
09:40:00 74/8010 0 3494/34703 0 0/0 0 0/0
10:00:00 75/8010 0 4918/34703 0 0/0 0 0/0
10:20:00 72/8010 0 4918/34703 0 0/0 0 0/0
10:40:00 71/8010 0 5018/34703 0 0/0 0 0/0
11:00:00 77/8010 0 5018/34703 0 0/0 0 0/0
```

下表介绍了 **-v** 选项的输出。

| 字段名     | 说明                                                       |
|---------|----------------------------------------------------------|
| proc-sz | 内核中当前正在使用或已分配的进程项（proc 结构）数。                             |
| inod-sz | 与内核中分配的最大 inode 数相比，内存中的 inode 总数。此数字不是严格的高水位标记。该数字可以溢出。 |
| file-sz | 打开的系统文件表的大小。由于文件表的空间是动态分配的，因此 <b>sz</b> 被给定为 0。          |
| ov      | 在每个表的采样点之间发生的溢出。                                         |

| 字段名     | 说明                                                        |
|---------|-----------------------------------------------------------|
| lock-sz | 内核中当前正在使用或分配的共享内存记录表项的数量。由于共享内存记录表的空间是动态分配的，因此 sz 被给定为 0。 |

▼ 如何检查交换活动 (sar -w)

- 使用 sar -w 命令可报告交换和切换活动。

```
$ sar -w
00:00:00 swpin/s bswin/s swpot/s bswot/s pswch/s
01:00:00 0.00 0.0 0.00 0.0 22
```

以下列表介绍了 sar -w 命令输出的目标值和观测值。

|         |                                                                |
|---------|----------------------------------------------------------------|
| swpin/s | 每秒传入内存的 LWP 数。                                                 |
| bswin/s | 每秒为换入传送的块数 /* (float)PGTOBLK(xx->cvmi.pgswapin) / sec_diff */。 |
| swpot/s | 每秒换出内存的平均进程数。如果该数字大于 1，则可能需要增大内存。                              |
| bswot/s | 每秒为换出传送的块数。                                                    |
| pswch/s | 每秒的内核线程切换数。                                                    |

注 – 所有进程换入都包括进程初始化。

示例 13-16 检查交换活动 (sar -w)

以下示例显示 sar -w 命令的输出。

```
$ sar -w

SunOS balmyday 5.10 s10_51 sun4u 03/18/2004

00:00:04 swpin/s bswin/s swpot/s bswot/s pswch/s
01:00:00 0.00 0.0 0.00 0.0 132
02:00:01 0.00 0.0 0.00 0.0 133
03:00:00 0.00 0.0 0.00 0.0 133
04:00:00 0.00 0.0 0.00 0.0 134
05:00:00 0.00 0.0 0.00 0.0 133
06:00:00 0.00 0.0 0.00 0.0 133
07:00:00 0.00 0.0 0.00 0.0 132
08:00:00 0.00 0.0 0.00 0.0 131
08:20:00 0.00 0.0 0.00 0.0 133
08:40:01 0.00 0.0 0.00 0.0 132
09:00:00 0.00 0.0 0.00 0.0 132
09:20:00 0.00 0.0 0.00 0.0 132
09:40:00 0.00 0.0 0.00 0.0 335
```

|          |      |     |      |     |     |
|----------|------|-----|------|-----|-----|
| 10:00:00 | 0.00 | 0.0 | 0.00 | 0.0 | 601 |
| 10:20:00 | 0.00 | 0.0 | 0.00 | 0.0 | 353 |
| 10:40:00 | 0.00 | 0.0 | 0.00 | 0.0 | 747 |
| 11:00:00 | 0.00 | 0.0 | 0.00 | 0.0 | 804 |
| Average  | 0.00 | 0.0 | 0.00 | 0.0 | 198 |

▼ 如何检查终端活动 (sar -y)

- 使用 sar -y 命令可监视终端设备活动。

```
$ sar -y
00:00:00 rawch/s canch/s outch/s rcvin/s xmtin/s mdmin/s
01:00:00 0 0 0 0 0 0
```

如果有多个终端 I/O，则可使用此报告来确定是否存在任何错误行。以下列表中定义了记录的活动。

- rawch/s      每秒输入字符数（原始队列）。
- canch/s      canon（规则队列）每秒处理的输入字符数。
- outch/s      每秒输出字符数（输出队列）。
- rcvin/s      每秒接收器硬件中断次数。
- xmtin/s      每秒传送器硬件中断次数。
- mdmin/s      每秒调制解调器中断次数。

每秒调制解调器中断次数 (mdmin/s) 应接近于零。每秒的接收和传送中断次数（rcvin/s 和 xmtin/s）应分别小于或等于传入或传出字符数。否则，请检查是否存在错误行。

示例 13-17 检查终端活动 (sar -y)

以下示例显示 sar -y 命令的输出。

```
$ sar -y

SunOS balmyday 5.10 s10_51 sun4u 03/18/2004

00:00:04 rawch/s canch/s outch/s rcvin/s xmtin/s mdmin/s
01:00:00 0 0 0 0 0 0
02:00:01 0 0 0 0 0 0
03:00:00 0 0 0 0 0 0
04:00:00 0 0 0 0 0 0
05:00:00 0 0 0 0 0 0
06:00:00 0 0 0 0 0 0
07:00:00 0 0 0 0 0 0
08:00:00 0 0 0 0 0 0
08:20:00 0 0 0 0 0 0
08:40:01 0 0 0 0 0 0
09:00:00 0 0 0 0 0 0
```

|          |   |   |    |   |   |   |
|----------|---|---|----|---|---|---|
| 09:20:00 | 0 | 0 | 0  | 0 | 0 | 0 |
| 09:40:00 | 0 | 0 | 1  | 0 | 0 | 0 |
| 10:00:00 | 0 | 0 | 37 | 0 | 0 | 0 |
| 10:20:00 | 0 | 0 | 0  | 0 | 0 | 0 |
| 10:40:00 | 0 | 0 | 3  | 0 | 0 | 0 |
| 11:00:00 | 0 | 0 | 3  | 0 | 0 | 0 |
| Average  | 0 | 0 | 1  | 0 | 0 | 0 |

▼ 如何检查总体系统性能 (sar -A)

- 使用 **sar -A** 命令可显示所有选项的统计信息，以提供总体系统性能的综览。  
此命令可提供更具全局性的透视。如果显示来自多个单时间段的数据，则该报告会包括平均值。

自动收集系统活动数据 (sar)

自动收集系统活动数据时需要使用三个命令：**sadc**、**sa1** 和 **sa2**。

**sadc** 数据收集实用程序定期收集系统数据，并以二进制格式的文件保存数据，每 24 小时保存一个文件。可以将 **sadc** 命令设置为定期运行（通常每小时一次），并在系统引导到多用户模式时运行。数据文件放置在 **/var/adm/sa** 目录中。每个文件都命名为 **sadd**，其中 **dd** 是当前日期。命令的格式如下：

```
/usr/lib/sa/sadc [t n] [ofile]
```

该命令以 **t** 秒为间隔采样 **n** 次，两次采样之间的间隔应大于 5 秒。然后，此命令将向二进制 **ofile** 文件或标准输出中写入数据。

引导时运行 **sadc** 命令

**sadc** 命令应在系统引导时运行，以记录自计数器重置为零以来的统计信息。为确保在引导时运行 **sadc** 命令，**svcadm enable system/sar:default** 命令会向每日数据文件中写入一条记录。

该命令项格式如下：

```
/usr/bin/su sys -c "/usr/lib/sa/sadc /var/adm/sa/sa`date +%d`"
```

## 使用 sa1 脚本定期运行 sadc 命令

为了生成定期记录，您需要定期运行 `sadc` 命令。最简单的方法是在 `/var/spool/cron/crontabs/sys` 文件中取消对下列行的注释：

```
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
```

`sys` `crontab` 项执行以下操作：

- 前两个 `crontab` 项可在星期一到星期五从上午 8 点到下午 5 点，每 20 分钟或每小时向 `/var/adm/sa/sadd` 文件写入一条记录。
- 第三项可从星期一到星期五每小时向 `/var/adm/sa/sar dd` 文件写入一条记录，并且可以包括所有 `sar` 选项。

可以更改这些缺省设置，以满足您的需要。

## 使用 sa2 Shell 脚本生成报告

另一个 shell 脚本 `sa2` 可生成报告，而不是二进制数据文件。`sa2` 命令调用 `sar` 命令，并将 ASCII 输出写入报告文件。

## 设置自动数据收集 (sar)

`sar` 命令可用于自行收集系统活动数据，或报告 `sadc` 命令所创建的每日活动文件中收集的内容。

`sar` 命令格式如下：

```
sar [-aAbcdgkmpqruvw] [-o file] t [n]
```

```
sar [-aAbcdgkmpqruvw] [-s time] [-e time] [-i sec] [-f file]
```

以下 `sar` 命令每隔 `t` 秒对操作系统中的累积活动计数器进行采样，共进行 `n` 次。`t` 应大于或等于 5 秒。否则，命令本身会对样本产生影响。必须指定采样的时间间隔。否则，命令将根据第二种格式运行。`n` 的缺省值为 1。以下示例以 10 秒为间隔抽取两个样本。如果指定 `-o` 选项，则以二进制格式保存样本。

```
$ sar -u 10 2
```



有关 sar 命令的其他重要信息包括：

- 如果不指定采样间隔或样本数，sar 命令将从以前记录的文件中提取数据。该文件是由 -f 选项为最近一天指定的文件，或对应于最近一天的标准每日活动文件 /var/adm/sa/sa dd（缺省设置）。
- -s 和 -e 选项定义报告的开始时间和结束时间。开始时间和结束时间的格式为 hh[:mm[:ss]]，其中 hh、mm 和 ss 表示小时、分钟和秒。
- -i 选项指定记录所选内容之间的时间间隔（以秒为单位）。如果不包括 -i 选项，则报告在每日活动文件中找到的所有间隔。

下表列出了 sar 选项及其操作。

表 13-2 sar 命令的选项

| 选项  | 操作                   |
|-----|----------------------|
| -a  | 检查文件访问操作             |
| -b  | 检查缓冲区活动              |
| -c  | 检查系统调用               |
| -d  | 检查每个块设备的活动           |
| -g  | 检查页出和内存释放            |
| -k  | 检查内核内存分配             |
| -m  | 检查进程间通信              |
| -nv | 检查系统表状态              |
| -p  | 检查交换和分发活动            |
| -q  | 检查队列活动               |
| -r  | 检查未使用的内存             |
| -u  | 检查 CPU 使用率           |
| -w  | 检查交换和切换卷             |
| -y  | 检查终端活动               |
| -A  | 报告总体系统性能，这与输入所有选项等效。 |

不使用任何选项等效于调用带 -u 选项的 sar 命令。

## ▼ 如何设置自动数据收集

### 1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《[系统管理指南：安全性服务](#)》中的“[配置 RBAC（任务列表）](#)”。

### 2 运行 `svcadm enable system/sar:default` 命令。

此版本的 `sadc` 命令会写入一条特殊记录，其中标记了将计数器重置为零的时间（引导时间）。

### 3 编辑 `/var/spool/cron/crontabs/sys` crontab 文件。

---

注 – 不要直接编辑 crontab 文件。而要改用 `crontab -e` 命令对现有 crontab 文件进行更改。

---

```
crontab -e sys
```

### 4 取消对以下行的注释：

```
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
```

有关更多信息，请参见 [crontab\(1\)](#) 手册页。

## 软件问题疑难解答（概述）

---

本章提供软件问题故障排除的一般概述，其中包括有关系统崩溃的故障排除以及查看系统消息的信息。

以下是本章中的信息列表。

- 第 195 页中的“疑难解答方面的新增内容”
- 第 197 页中的“有关软件故障排除任务的参考信息”
- 第 197 页中的“系统崩溃故障排除”
- 第 199 页中的“系统崩溃故障排除核对表”

### 疑难解答方面的新增内容

本节介绍此发行版中新增或已更改的故障排除信息。

有关 Oracle Solaris 10 发行版中新增或已更改的故障排除功能的信息，请参见以下内容：

- 第 196 页中的“动态跟踪功能”
- 第 196 页中的“`kldb` 取代 `kadb` 作为标准的 Solaris 内核调试程序”

有关新增功能的完整列表以及 Oracle Solaris 发行版的说明，请参见《[Oracle Solaris 10 8/11 新增功能](#)》。

### Common Agent Container 问题

**Oracle Solaris 10 6/06**：Common Agent Container 是现已包括在 Oracle Solaris OS 中的独立 Java 程序。此程序可为 Java 管理应用程序实现容器。Common Agent Container 提供一种针对基于 Java Management Extensions (JMX) 和 Java Dynamic Management Kit (Java DMK) 的功能设计的管理基础结构。此软件由 `SUNWcacao` 软件包安装，它驻留在 `/usr/lib/cacao` 目录中。

通常，容器是不可见的。但在以下两种情况下，您可能需要与容器守护进程交互：

- 其他应用程序可能会尝试使用为 Common Agent Container 保留的网络端口。
- 如果证书库遭到破坏，则可能需要重新生成 Common Agent Container 证书密钥。

有关如何解决这些问题的信息，请参见第 235 页中的“[Oracle Solaris OS 中 Common Agent Container 问题的故障排除](#)”。

## x86: SMF 引导归档文件服务可能在系统重新引导期间失败

**Solaris 10 1/06**：如果系统在基于 GRUB 的引导环境中发生崩溃，可能是 SMF 服务 `svc:/system/boot-archive:default` 在系统重新引导时失败。如果出现此问题，请重新引导系统并在 GRUB 引导菜单中选择故障安全归档文件。请按照提示重新生成引导归档文件。重新生成归档文件后，重新引导系统。要继续引导过程，可以使用 `svcadm` 命令清除 `svc:/system/boot-archive:default` 服务。有关基于 GRUB 进行引导的更多信息，请参见《系统管理指南：基本管理》中的“[使用 GRUB 引导基于 x86 的系统（任务列表）](#)”。

## 动态跟踪功能

Oracle Solaris 动态跟踪 (Dynamic Tracing, DTrace) 功能是一项全面的动态跟踪功能，借助该功能，可在一个全新级别观察 Solaris 内核和用户进程。使用 Dtrace 可以动态检测操作系统内核和用户进程，并记录您在所关注的位置（称为**探测器**）指定的数据，从而帮助您了解系统。每个探测器均可与用新的 D 编程语言编写的定制程序相关。DTrace 的所有检测过程都是完全动态的，并且可用于产品化的系统。有关更多信息，请参见 [dtrace\(1M\)](#) 手册页和《[Solaris 动态跟踪指南](#)》。

## kldb 取代 kadb 作为标准的 Solaris 内核调试程序

kldb 已取代 kadb，作为标准的“现场”Solaris 内核调试程序。

在进行实时内核调试时，kldb 可实现 mdb 的所有强大功能和灵活性。kldb 支持以下功能：

- 调试程序命令 (dcmds)
- 调试程序模块 (dmods)
- 访问内核类型数据
- 内核执行控制
- 检查
- 修改

有关更多信息，请参见 [kmdb\(1\)](#) 手册页。有关使用 `kmdb` 对系统进行故障排除的逐步说明，请参见《系统管理指南：基本管理》中的“如何使用内核调试器 (`kmdb`) 引导系统”和《系统管理指南：基本管理》中的“如何在 GRUB 引导环境中使用内核调试器 (`kmdb`) 引导系统”。

## 有关软件故障排除任务的参考信息

| 故障排除任务                | 更多信息                                                                  |
|-----------------------|-----------------------------------------------------------------------|
| 管理系统故障转储信息            | <a href="#">第 17 章，管理系统故障转储信息（任务）</a>                                 |
| 管理核心文件                | <a href="#">第 16 章，管理核心文件（任务）</a>                                     |
| 重新引导失败和备份问题等软件问题的故障排除 | <a href="#">第 18 章，各种软件问题的故障排除（任务）</a>                                |
| 文件访问问题故障排除            | <a href="#">第 19 章，文件访问问题故障排除（任务）</a>                                 |
| 打印问题故障排除              | <a href="#">《系统管理指南：打印》中的第 13 章“Oracle Solaris OS 中的打印问题故障排除（任务）”</a> |
| 解决 UFS 文件系统不一致问题      | <a href="#">第 20 章，解决 UFS 文件系统不一致问题（任务）</a>                           |
| 软件包问题故障排除             | <a href="#">第 21 章，软件包问题故障排除（任务）</a>                                  |

## 系统崩溃故障排除

如果运行 Oracle Solaris OS 的系统崩溃，请向服务提供商提供尽可能多的信息，包括故障转储文件。

## 系统崩溃时应执行的操作

要记住的最重要的事情如下：

1. 记录系统控制台消息。

如果系统崩溃，则使其重新运行可能是最紧迫的事情。但是，在重新引导系统之前，请先检查控制台屏幕上的消息。这些消息可能有助于了解导致崩溃的原因。即使系统自动重新引导并且控制台消息已从屏幕上消失，仍然可以通过查看系统错误日志（`/var/adm/messages` 文件）来检查这些消息。有关查看系统错误日志文件的更多信息，请参见 [第 202 页](#) 中的“如何查看系统消息”。

如果系统频繁发生崩溃并且无法确定其原因，请收集可从系统控制台或 `/var/adm/messages` 文件中获取的所有信息，并准备好以供客户服务代表检查。有关要为服务提供商收集的故障排除信息的完整列表，请参见 [第 197 页](#) 中的“系统崩溃故障排除”。

如果系统在崩溃后无法成功重新引导，请参见第 18 章，各种软件问题的故障排除（任务）。

2. 同步磁盘并重新引导。

`ok sync`

如果系统在崩溃后无法成功重新引导，请参见第 18 章，各种软件问题的故障排除（任务）。

查看系统崩溃后是否生成了系统故障转储。缺省情况下，会保存系统故障转储。有关故障转储的信息，请参见第 17 章，管理系统故障转储信息（任务）。

## 收集故障排除数据

请回答以下问题，以帮助查出系统问题。有关收集崩溃系统的故障排除数据的信息，请参见第 199 页中的“系统崩溃故障排除核对表”。

表 14-1 确定系统崩溃数据

| 问题                            | 说明                                                                      |
|-------------------------------|-------------------------------------------------------------------------|
| 是否可以重现问题？                     | 这一点很重要，因为在调试很难的问题时，可重现的测试案例是必不可少的。通过重现问题，服务提供商可以使用特殊设备构造内核，以触发、诊断和更正错误。 |
| 是否使用了任何第三方驱动程序？               | 具有相同特权的多个驱动程序在内核所在的同一地址空间中运行，如果这些驱动程序存在已知问题，则会导致系统崩溃。                   |
| 系统在崩溃之前正在执行什么操作？              | 如果系统在执行异常的操作，例如运行新的负荷测试或遇到特别高的负荷，则可能导致系统崩溃。                             |
| 在系统崩溃之前，是否有任何异常的控制台消息？        | 有时，系统会在实际崩溃前显示故障信号，此信息通常很有用。                                            |
| 是否向 /etc/system 文件中添加了任何调优参数？ | 有时，调优参数（如增大共享内存段，以使系统尝试分配比实际拥有内存更多的内存）会导致系统崩溃。                          |
| 问题是在最近开始的吗？                   | 如果是这样，问题是否与对系统的更改同时出现？例如新的驱动程序、新软件、不同工作负荷、CPU 升级或内存升级。                  |

# 系统崩溃故障排除核对表

可在为崩溃的系统收集系统数据时使用此核对表。

| 项目                                      | 数据 |
|-----------------------------------------|----|
| 系统故障转储是否可用？                             |    |
| 确定操作系统发行版以及相应软件应用程序的发行版级别。              |    |
| 确定系统硬件。                                 |    |
| 包括 sun4u 系统的 prtdiag 输出。包括其他系统的资源管理器输出。 |    |
| 是否安装了修补程序？如果已安装，请包括 showrev -p 输出。      |    |
| 问题是否可重现？                                |    |
| 系统中是否有任何第三方驱动程序？                        |    |
| 系统在崩溃前正在执行什么操作？                         |    |
| 在系统崩溃前是否有任何异常的控制台消息？                    |    |
| 是否向 /etc/system 文件中添加了任何参数？             |    |
| 问题是在最近开始的吗？                             |    |





## 管理系统消息

---

本章介绍 Oracle Solaris OS 中的系统消息传送功能。

### 查看系统消息

系统消息显示在控制台设备中。大多数系统消息的文本如下所示：

[ID *msgid facility.priority*]

例如：

[ID 672855 kern.notice] syncing file systems...

如果消息来自内核，则会显示内核模块名称。例如：

Oct 1 14:07:24 mars ufs: [ID 845546 kern.notice] alloc: /: file system full

当系统崩溃时，系统控制台可能会显示如下消息：

panic: *error message*

少数情况下，可能会显示以下消息而非故障消息：

Watchdog reset !

错误日志守护进程 `syslogd` 可在消息文件中自动记录各种系统警告和错误。缺省情况下，其中许多系统消息都会显示在系统控制台中，并存储在 `/var/adm` 目录中。通过设置系统消息日志可以指示这些消息的存储位置。有关更多信息，请参见第 204 页中的“定制系统消息日志”。这些消息可以提醒您系统出现问题，例如设备将要出现故障。

`/var/adm` 目录中包含若干个消息文件。最新消息位于 `/var/adm/messages` 文件中（和 `messages.*` 中），而最旧的消息位于 `messages.3` 文件中。经过一段时间后（通常为每隔十天），会创建一个新的 `messages` 文件。`messages.0` 文件被重命名为

messages.1, messages.1 被重命名为 messages.2, 而 messages.2 被重命名为 messages.3。当前的 /var/adm/messages.3 文件将被删除。

由于 /var/adm 目录存储包含消息、故障转储和其他数据的大型文件，因此该目录可能会占用许多磁盘空间。为防止 /var/adm 目录变得过大，并确保可以保存将来的故障转储，应定期删除不需要的文件。可以使用 crontab 文件自动执行此任务。有关自动执行此任务的更多信息，请参见第 83 页中的“如何删除故障转储文件”和第 8 章，调度系统任务（任务）。

## ▼ 如何查看系统消息

- 使用 dmesg 命令显示由系统崩溃或重新引导生成的最新消息。

```
$ dmesg
```

或者，使用 more 命令逐屏显示消息。

```
$ more /var/adm/messages
```

### 示例 15-1 查看系统消息

以下示例显示 dmesg 命令的输出。

```
$ dmesg
Jan 3 08:44:41 starbug genunix: [ID 540533 kern.notice] SunOS Release 5.10 ...
Jan 3 08:44:41 starbug genunix: [ID 913631 kern.notice] Copyright 1983-2003 ...
Jan 3 08:44:41 starbug genunix: [ID 678236 kern.info] Ethernet address ...
Jan 3 08:44:41 starbug unix: [ID 389951 kern.info] mem = 131072K (0x8000000)
Jan 3 08:44:41 starbug unix: [ID 930857 kern.info] avail mem = 121888768
Jan 3 08:44:41 starbug rootnex: [ID 466748 kern.info] root nexus = Sun Ultra 5/
10 UPA/PCI (UltraSPARC-IIi 333MHz)
Jan 3 08:44:41 starbug rootnex: [ID 349649 kern.info] pcipsy0 at root: UPA 0x1f0x0
Jan 3 08:44:41 starbug genunix: [ID 936769 kern.info] pcipsy0 is /pci@1f,0
Jan 3 08:44:41 starbug pcipsy: [ID 370704 kern.info] PCI-device: pci@1,1, simba0
Jan 3 08:44:41 starbug genunix: [ID 936769 kern.info] simba0 is /pci@1f,0/pci@1,1
Jan 3 08:44:41 starbug pcipsy: [ID 370704 kern.info] PCI-device: pci@1, simbal
Jan 3 08:44:41 starbug genunix: [ID 936769 kern.info] simbal is /pci@1f,0/pci@1
Jan 3 08:44:57 starbug simba: [ID 370704 kern.info] PCI-device: ide@3, uata0
Jan 3 08:44:57 starbug genunix: [ID 936769 kern.info] uata0 is /pci@1f,0/pci@1,
1/ide@3
Jan 3 08:44:57 starbug uata: [ID 114370 kern.info] dad0 at pci1095,6460
.
.
.
```

另请参见 有关更多信息，请参见 [dmesg\(1M\)](#) 手册页。

# 系统日志轮转

使用 `root crontab` 中的一个项内的 `logadm` 命令，可以轮转系统日志文件。不再使用 `/usr/lib/newsyslog` 脚本。

系统日志轮转在 `/etc/logadm.conf` 文件中定义。此文件包含用于 `syslogd` 等进程的日志轮转项。例如，`/etc/logadm.conf` 文件中的一个项指定，除非 `/var/log/syslog` 文件为空，否则该文件每周轮转一次。最新的 `syslog` 文件成为 `syslog.0`，下一个最新的文件成为 `syslog.1`，依此类推。会保留八个以前的 `syslog` 日志文件。

`/etc/logadm.conf` 文件还包含记录最后一次日志轮转发生时间的时间戳。

可以使用 `logadm` 命令来定制系统日志，并可根据需要在 `/etc/logadm.conf` 文件中添加其他日志。

例如，要轮转 Apache 访问和错误日志，请使用以下命令：

```
logadm -w /var/apache/logs/access_log -s 100m
logadm -w /var/apache/logs/error_log -s 10m
```

在此示例中，Apache `access_log` 文件会在大小达到 100 MB 时进行轮转，以 `.0`、`.1` 等作为后缀，并保留旧 `access_log` 文件的 10 个副本。`error_log` 会在大小达到 10 MB 时进行轮转，后缀和副本数与 `access_log` 文件相同。

用于上述 Apache 日志轮转示例的 `/etc/logadm.conf` 项与以下示例类似：

```
cat /etc/logadm.conf
.
.
.
/var/apache/logs/error_log -s 10m
/var/apache/logs/access_log -s 100m
```

有关更多信息，请参见 [logadm\(1M\)](#)。

可以超级用户身份或通过承担等效角色（具有日志管理权限）来使用 `logadm` 命令。通过基于角色的访问控制 (role-based access control, RBAC)，您可以通过提供对 `logadm` 命令的访问来授予非 `root` 用户维护日志文件的特权。

例如，可通过向 `/etc/user_attr` 文件添加以下项，授予用户 `andy` 使用 `logadm` 命令的权限：

```
andy:::profiles=Log Management
```

也可以使用 Solaris Management Console 设置一个用于日志管理的角色。有关设置角色的更多信息，请参见《系统管理指南：安全性服务》中的“基于角色的访问控制（概述）”。

# 定制系统消息日志

通过修改 `/etc/syslog.conf` 文件，可以捕获各个系统进程生成的其他错误消息。缺省情况下，`/etc/syslog.conf` 文件会将许多系统进程消息定向到 `/var/adm/messages` 文件。崩溃和引导消息也存储在这些文件中。要查看 `/var/adm` 消息，请参见第 202 页中的“如何查看系统消息”。

`/etc/syslog.conf` 文件有两个通过制表符分隔的列：

*facility.level ... action*

*facility.level*      消息或情况的工具或系统源。可能是由逗号分隔的工具列表。表 15-1 中列出了工具值。*level*，表示所记录情况的严重程度或优先级。表 15-2 中列出了优先级。

如果同一工具的两个项用于不同优先级，则不要将这两个项放在同一行中。在 `syslog` 文件中放置优先级表示将记录该优先级或更高优先级的所有消息，最后一条消息优先。对于给定的工具和级别，`syslogd` 将匹配该级别以及所有更高级别的所有消息。

*action*              操作字段表示将消息转发到的位置。

以下示例显示缺省的 `/etc/syslog.conf` 文件中的样例行。

```
user.err /dev/sysmsg
user.err /var/adm/messages
user.alert 'root, operator'
user.emerg *
```

这意味着将自动记录以下用户消息：

- 将用户错误列显到控制台，同时将其记录到 `/var/adm/messages` 文件中。
- 将需要立即操作的用户消息 (`alert`) 发送给 `root` 用户和操作员用户。
- 将用户紧急消息发送给各用户。

注 - 如果在 `/etc/syslog.conf` 文件中多次指定一个日志目标，则将各项分别放置在不同的行中可能会导致消息的记录顺序混乱。请注意，可在一个行项中指定多个选择器，每个选择器之间用分号分隔。

下表中显示了最常见的错误情况源。表 15-2 按严重程度显示最常见的优先级。

表 15-1 `syslog.conf` 消息的源工具

| 源    | 说明 |
|------|----|
| kern | 内核 |

表 15-1 syslog.conf 消息的源工具 (续)

| 源      | 说明     |
|--------|--------|
| auth   | 验证     |
| daemon | 所有守护进程 |
| mail   | 邮件系统   |
| lp     | 假脱机系统  |
| user   | 用户进程   |

注 – 可在 /etc/syslog.conf 文件中激活的 syslog 工具数没有限制。

表 15-2 syslog.conf 消息的优先级

| 优先级   | 说明        |
|-------|-----------|
| emerg | 系统紧急情况    |
| alert | 需要立即更正的错误 |
| crit  | 严重错误      |
| err   | 其他错误      |
| info  | 信息性消息     |
| debug | 用于调试的输出   |
| none  | 此设置不记录输出  |

## ▼ 如何定制系统消息日志

- 1 成为超级用户或承担等效角色。  
角色包含授权和具有特权的命令。有关角色的更多信息，请参见《[系统管理指南：安全性服务](#)》中的“[配置 RBAC（任务列表）](#)”。
- 2 编辑 /etc/syslog.conf 文件，根据 [syslog.conf\(4\)](#) 中介绍的语法添加或更改消息源、优先级和消息位置。
- 3 退出文件，保存更改。

## 示例 15-2 定制系统消息日志

此样例 `/etc/syslog.conf` `user.emerg` 工具可向 `root` 用户和个人用户发送用户紧急消息。

```
user.emerg
```

```
'root, *'
```

## 启用远程控制台消息传送

以下新增控制台功能可以提高您对远程系统进行故障排除的能力：

- 利用 `consadm` 命令，您可以选择串行设备作为**辅助**（或远程）控制台。使用 `consadm` 命令，系统管理员可以配置一个或多个串行端口，以便当系统在不同运行级之间转换时显示重定向的控制台消息，并托管 `sulogin` 会话。借助此功能，您可用调制解调器拨入并连接到串行端口，以监视控制台消息并参与 `init` 状态转换。（有关更多信息，请参见 [sulogin\(1M\)](#) 和后面的逐步过程。）

使用配置为辅助控制台的端口登录系统时，它主要用作输出设备，其中显示的信息也在缺省控制台中显示。如果引导脚本或其他应用程序从缺省控制台中读取内容或向其中写入内容，则写入输出将在所有辅助控制台上显示，但输入只从缺省控制台中读取。（有关在交互式登录会话过程中使用 `consadm` 命令的更多信息，请参见 [第 207 页中的“在交互式登录会话期间使用 `consadm` 命令”](#)。）

- 控制台输出包括内核和写入新的伪设备 `/dev/sysmsg` 的 `syslog` 消息。此外，`rc` 脚本启动消息也将写入 `/dev/msglog`。以前，所有这些消息都写入 `/dev/console`。

如果希望看到显示在辅助控制台中的脚本消息，那么需要把将控制台输出定向到 `/dev/console` 的脚本更改为 `/dev/msglog`。如果希望将消息重定向到辅助设备，则应该将引用 `/dev/console` 的程序显式修改为使用 `syslog()` 或 `strlog()`。

- `consadm` 命令运行守护进程来监视辅助控制台设备。指定为辅助控制台并且已断开、挂起或失去载体的任何显示设备都将从辅助控制台设备列表中删除，并且不再处于活动状态。启用一个或多个辅助控制台不会禁用缺省控制台上的消息显示，消息将继续在 `/dev/console` 中显示。

## 在运行级转换期间使用辅助控制台消息传递

在运行级转换期间使用辅助控制台消息传递时，请记住以下几点：

- 如果在系统引导时运行的 `rc` 脚本期望用户输入，则输入不能来自辅助控制台。输入必须来自缺省控制台。
- 由 `init` 调用以用于在运行级之间转换时提示输入超级用户口令的 `sulogin` 程序已被修改，除了缺省控制台设备之外，还可以向每个辅助设备发送超级用户口令提示。

- 当系统处于单用户模式并且使用 `consadm` 命令启用一个或多个辅助控制台时，将在第一个设备上运行控制台登录会话，以便为 `sulogin` 提示提供正确的超级用户口令。从控制台设备收到正确口令时，`sulogin` 将禁用来自所有其他控制台设备的输入。
- 如果其中一个控制台承担了单用户特权，将在缺省控制台和其他辅助控制台上显示一条消息。此消息指出已通过接受正确的超级用户口令而成为控制台的设备。如果运行单用户 `shell` 的辅助控制台中丢失载体，则会执行以下两种操作之一：
  - 如果辅助控制台代表一个处于运行级 1 的系统，则系统会继续到缺省运行级。
  - 如果辅助控制台代表一个处于运行级 S 的系统，则系统会显示已通过 `Shell` 输入 `init s` 或 `shutdown` 命令的设备中的 `ENTER RUN LEVEL (0-6, s or S):` 消息。如果该设备中也没有任何载体，则必须重新建立载体并输入正确的运行级。`init` 或 `shutdown` 命令不再重新显示运行级提示。
- 如果使用串行端口登录系统，并发出 `init` 或 `shutdown` 命令以转换到其他运行级，则无论此设备是否为辅助控制台，登录会话都将丢失。此情况与没有辅助控制台功能的发行版相同。
- 一旦使用 `consadm` 命令将设备选作辅助控制台，该设备将一直用作辅助控制台，直到重新引导系统或取消选中辅助控制台。但是，`consadm` 命令有一个选项，可在系统重新引导期间将设备设置为辅助控制台。（有关逐步说明，请参见以下过程。）

## 在交互式登录会话期间使用 `consadm` 命令

如果要通过使用与串行端口连接的终端登录系统，再使用 `consadm` 命令查看终端的控制台消息的方式来运行交互式登录会话，请注意以下行为。

- 如果在辅助控制台处于活动状态时将终端用于交互式登录会话，则会向 `/dev/sysmsg` 或 `/dev/msglog` 设备发送控制台消息。
- 在终端发出命令时，输入将转到交互式会话而非缺省控制台 (`/dev/console`)。
- 如果运行 `init` 命令更改运行级，远程控制台软件将中止交互式会话并运行 `sulogin` 程序。此时，只接受来自终端的输入，并将其视为来自控制台设备的输入。这样您就可以按第 206 页中的“在运行级转换期间使用辅助控制台消息传递”中所述为 `sulogin` 程序输入口令。

然后，如果您在（辅助）终端中输入正确口令，辅助控制台将运行交互式 `sulogin` 会话，并锁定缺省控制台和任何竞争性的辅助控制台。这意味着，终端基本上可用作系统控制台。

- 此时，您可以更改到运行级 3 或转到其他运行级。如果更改运行级，`sulogin` 将在所有控制台设备中再次运行。如果您退出或指定系统应达到运行级 3，则所有辅助控制台都将丧失提供输入的能力。它们将恢复为控制台消息的显示设备。

随着系统的提升，您必须为缺省控制台设备中的 `rc` 脚本提供信息。在系统恢复启动后，`login` 程序将在串行端口上运行，您可以重新登录到其他交互式会话中。如果已将该设备指定为辅助控制台，您将继续在终端中获得控制台消息，但来自该终端的所有输入都将转至交互式会话。

## ▼ 如何启用辅助（远程）控制台

在您使用 `consadm` 命令添加辅助控制台之前，`consadm` 守护进程不会开始监视端口。作为一种安全功能，在载体脱机或取消选择辅助控制台设备之前，控制台消息只能重定向。这意味着必须在端口中建立载体，才能成功使用 `consadm` 命令。

有关启用辅助控制台的更多信息，请参见 [consadm\(1m\)](#) 手册页。

- 1 以超级用户身份登录系统。
- 2 启用辅助控制台。  
`# consadm -a devicename`
- 3 验证当前连接是否为辅助控制台。  
`# consadm`

### 示例 15-3 启用辅助（远程）控制台

```
consadm -a /dev/term/a
consadm
/dev/term/a
```

## ▼ 如何显示辅助控制台的列表

- 1 以超级用户身份登录系统。
- 2 选择以下步骤之一：
  - a. 显示辅助控制台的列表。  
`# consadm
/dev/term/a`
  - b. 显示持久性辅助控制台的列表。  
`# consadm -p
/dev/term/b`

## ▼ 如何在系统重新引导期间启用辅助（远程）控制台

- 1 以超级用户身份登录系统。
- 2 在系统重新引导期间启用辅助控制台。  
`# consadm -a -p devicename`



这将会向持久性辅助控制台列表中添加设备。

- 3 验证设备是否已添加至持久性辅助控制台的列表中。

```
consadm
```

#### 示例 15-4 在系统重新引导期间启用辅助（远程）控制台

```
consadm -a -p /dev/term/a
consadm
/dev/term/a
```

## ▼ 如何禁用辅助（远程）控制台

- 1 以超级用户身份登录系统。

- 2 选择以下步骤之一：

- a. 禁用辅助控制台。

```
consadm -d devicename
```

或

- b. 禁用辅助控制台并从持久性辅助控制台列表中将其删除。

```
consadm -p -d devicename
```

- 3 验证是否已禁用辅助控制台。

```
consadm
```

#### 示例 15-5 禁用辅助（远程）控制台

```
consadm -d /dev/term/a
consadm
```



## 管理核心文件（任务）

本章介绍如何使用 `coreadm` 命令管理核心文件。

有关与管理核心转储文件相关的过程的信息，请参见第 211 页中的“管理核心文件（任务列表）”。

### 管理核心文件（任务列表）

| 任务              | 说明                                                                | 参考                                                                             |
|-----------------|-------------------------------------------------------------------|--------------------------------------------------------------------------------|
| 1. 显示当前的核心转储配置。 | 使用 <code>coreadm</code> 命令显示当前的核心转储配置。                            | 第 214 页中的“如何显示当前的核心转储配置”                                                       |
| 2. 修改核心转储配置。    | 修改核心转储配置，以执行下列操作之一：<br>设置核心文件名称模式。<br>启用每进程核心文件路径。<br>启用全局核心文件路径。 | 第 214 页中的“如何设置核心文件名称模式”<br>第 214 页中的“如何启用每进程核心文件路径”<br>第 215 页中的“如何启用全局核心文件路径” |
| 3. 检查核心转储文件。    | 使用 <code>proc</code> 工具查看核心转储文件。                                  | 第 215 页中的“检查核心文件”                                                              |

### 管理核心文件概述

核心文件是在异常终止进程或应用程序时生成的。使用 `coreadm` 命令可以管理核心文件。

例如，可以使用 `coreadm` 命令来配置系统，以便将所有进程核心文件都放在一个系统目录中。这意味着，当进程或守护进程异常终止时，可通过检查特定目录中的核心文件来跟踪问题。

## 配置核心文件路径

可以独立启用或禁用的两个新的可配置 `core` 文件路径是：

- 每进程核心文件路径，缺省为 `core`，缺省情况下处于启用状态。如果启用，则每进程核心文件路径会导致在进程异常终止时生成 `core` 文件。每进程路径由新进程从其父进程处继承。

生成每进程核心文件时，该文件由具有所有者读/写权限的进程所有者所有。只有所有者用户可以查看此文件。

- 全局核心文件路径，缺省为 `core`，缺省情况下处于禁用状态。如果启用，则会使用全局核心文件路径生成内容与每进程核心文件相同的**附加**核心文件。

生成全局核心文件时，该文件只属于具有超级用户读/写权限的超级用户所有。非特权用户不能查看此文件。

当进程异常终止时，缺省情况下会在当前目录中生成一个核心文件。如果启用了全局核心文件，则每个异常终止的进程可能会生成两个文件，一个在当前工作目录中，另一个在全局核心文件位置。

缺省情况下，`setuid` 进程不使用全局或每进程路径生成核心文件。

## 扩展的核心文件名

如果启用了全局核心转储文件目录，则可使用下表中介绍的变量来区分 `core` 文件。

| 变量名 | 变量定义                                      |
|-----|-------------------------------------------|
| %d  | 可执行文件目录名，最多包含 <code>MAXPATHLEN</code> 个字符 |
| %f  | 可执行文件名，最多包含 <code>MAXCOMLEN</code> 个字符    |
| %g  | 有效组 ID                                    |
| %m  | 计算机名 ( <code>uname -m</code> )            |
| %n  | 系统节点名 ( <code>uname -n</code> )           |
| %p  | 进程 ID                                     |
| %t  | 时间的十进制值 (2)                               |
| %u  | 有效用户 ID                                   |
| %z  | 在其中执行进程的区域的名称 ( <code>zonename</code> )   |
| %%  | 字面值 %                                     |

例如，如果全局核心文件路径设置为：

```
/var/core/core.%f.%p
```

并且 PID 为 12345 的 `sendmail` 进程异常终止，则会生成以下 `core` 文件：

```
/var/core/core.sendmail.12345
```

## 设置核心文件名称模式

可以按全局、区域或进程设置核心文件名称模式。此外，还可以设置在系统重新引导后仍持续存在的每进程缺省值。

例如，以下 `coreadm` 命令将设置缺省的每进程核心文件模式。此设置适用于未显式覆盖缺省核心文件模式的所有进程。此设置在系统重新引导后继续存在。

```
coreadm -i /var/core/core.%f.%p
```

以下 `coreadm` 命令可为任何进程设置每进程核心文件名称模式：

```
$ coreadm -p /var/core/core.%f.%p $$
```

`$$` 符号表示当前正在运行的 Shell 的进程 ID 的占位符。所有子进程都会继承每进程核心文件名称模式。

一旦设置了全局或每进程核心文件模式，就必须使用 `coreadm -e` 命令启用该模式。有关更多信息，请参见以下过程。

通过在 `$HOME/.profile` 或 `.login` 文件中放置该命令，便可为用户登录会话期间运行的所有进程设置核心文件名称模式。

## 启用 `setuid` 程序以生成核心文件

可以使用 `coreadm` 命令启用或禁用 `setuid` 程序，以便通过设置以下路径来为所有系统进程或每个进程生成核心文件：

- 如果启用了全局 `setuid` 选项，则全局核心文件路径允许系统中的所有 `setuid` 程序生成 `core` 文件。
- 如果启用了每进程 `setuid` 选项，则每进程核心文件路径只允许特定 `setuid` 进程生成 `core` 文件。

缺省情况下，这两个标志都被禁用。由于安全原因，全局核心文件路径必须为全路径名，以斜杠 `/` 开头。如果超级用户禁用每进程核心文件，则各个用户都无法获取核心文件。

`setuid` 核心文件只属于具有超级用户读/写权限的超级用户。即使生成 `setuid` 核心文件的进程属于普通用户，常规用户也不能访问这些文件。

有关更多信息，请参见 [coreadm\(1M\)](#) 手册页。

## 如何显示当前的核心转储配置

使用不带任何选项的 `coreadm` 命令可以显示当前的核心转储配置。

```
$ coreadm
 global core file pattern:
global core file content: default
 init core file pattern: core
 init core file content: default
 global core dumps: disabled
per-process core dumps: enabled
 global setid core dumps: disabled
per-process setid core dumps: disabled
 global core dump logging: disabled
```

## ▼ 如何设置核心文件名称模式

- 确定要设置每进程核心文件还是全局核心文件，并选择下列项之一：

- a. 设置每进程文件名称模式。

```
$ coreadm -p $HOME/corefiles/%f.%p $$
```

- b. 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《[系统管理指南：安全性服务](#)》中的“[配置 RBAC（任务列表）](#)”。

- c. 设置全局文件名称模式。

```
coreadm -g /var/corefiles/%f.%p
```

## ▼ 如何启用每进程核心文件路径

- 1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《[系统管理指南：安全性服务](#)》中的“[配置 RBAC（任务列表）](#)”。

- 2 启用每进程核心文件路径。

```
coreadm -e process
```

- 3 显示当前的进程核心文件路径，以验证配置。

```
$ coreadm $$
1180: /home/kryten/corefiles/%f.%p
```

## ▼ 如何启用全局核心文件路径

### 1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的“配置 RBAC（任务列表）”。

### 2 启用全局核心文件路径。

```
coreadm -e global -g /var/core/core.%f.%p
```

### 3 显示当前的进程核心文件路径，以验证配置。

```
coreadm
 global core file pattern: /var/core/core.%f.%p
 global core file content: default
 init core file pattern: core
 init core file content: default
 global core dumps: enabled
 per-process core dumps: enabled
 global setid core dumps: disabled
 per-process setid core dumps: disabled
 global core dump logging: disabled
```

## 核心文件问题故障排除

错误消息

```
NOTICE: 'set allow_setid_core = 1' in /etc/system is obsolete
NOTICE: Use the coreadm command instead of 'allow_setid_core'
```

原因

存在一个过时的参数，它允许 setuid 核心文件位于 /etc/system 文件中。

解决方法

从 /etc/system 文件中删除 allow\_setid\_core=1。然后使用 coreadm 命令启用全局 setuid 核心文件路径。

## 检查核心文件

有些 proc 工具已得到增强，可以检查进程核心转储文件和实时进程。这些 proc 工具是可以管理 /proc 文件系统功能的实用程序。

通过在命令行中指定核心文件的名称（方法与向命令指定进程 ID 的方法相似），可将 /usr/proc/bin/pstack、pmap、pldd、pflags 和 pcred 工具应用于核心文件。

有关使用 proc 工具检查核心转储文件的更多信息，请参见 [proc\(1\)](#)。

示例 16-1 使用 proc 工具检查核心转储文件

```
$./a.out
Segmentation Fault(coredump)
$ /usr/proc/bin/pstack ./core
core './core' of 19305: ./a.out
000108c4 main (1, ffbef5cc, ffbef5d4, 20800, 0, 0) + 1c
00010880 _start (0, 0, 0, 0, 0, 0) + b8
```



## 管理系统故障转储信息（任务）

---

本章介绍如何在 Oracle Solaris OS 中管理系统故障转储信息。

有关与管理系统故障转储信息相关的过程信息，请参见第 218 页中的“管理系统故障转储信息（任务列表）”。

### 管理系统故障转储信息中的新增内容

本节介绍此 Oracle Solaris 发行版中用于管理系统资源的新增功能或已更改的功能。

#### 快速故障转储工具

**Oracle Solaris 10 9/10：**此功能增强使系统能够在更短的时间内使用更少的空间保存故障转储信息。完成故障转储所需的时间现在缩短了 2 至 10 分钟，具体取决于平台。在 `savecore` 目录中保存故障转储所需的磁盘空间量以相同的系数缩减。要加快故障转储文件的创建和压缩，快速故障转储工具可利用大型系统中较少使用的 CPU。新的故障转储文件 `vmdump.n` 是 `vmcore.n` 和 `unix.n` 文件的压缩版本。经过压缩的故障转储文件在网络上传输的速度更快，而且随后可离线分析。请注意，转储文件必须先解压缩，然后 `mdb` 实用程序之类的工具才能使用此文件。您可以通过使用 `savecore` 命令从本地或远程解压缩转储文件。

为了支持新的故障转储工具，已将 `-z` 选项添加至 `dumpadm` 命令。使用此选项可指定是以压缩格式还是未压缩格式保存转储文件。缺省值为压缩格式。

有关更多信息，请参见 [dumpadm\(1M\)](#) 和 [savecore\(1M\)](#) 手册页。

# 管理系统故障转储信息（任务列表）

以下任务列表提供了管理系统故障转储信息所需的过程。

| 任务                      | 说明                                                                                      | 参考                              |
|-------------------------|-----------------------------------------------------------------------------------------|---------------------------------|
| 1. 显示当前的故障转储配置。         | 使用 <code>dumpadm</code> 命令显示当前的故障转储配置。                                                  | 第 221 页中的“如何显示当前的故障转储配置”        |
| 2. 修改故障转储配置。            | 使用 <code>dumpadm</code> 命令指定要转储的数据类型、系统是否使用专用转储设备和用于保存故障转储文件的目录，以及在写入故障转储文件后必须保持可用的空间量。 | 第 222 页中的“如何修改故障转储配置”           |
| 3. 检查故障转储文件。            | 使用 <code>mdb</code> 命令查看故障转储文件。                                                         | 第 223 页中的“如何检查故障转储”             |
| 4. （可选的）从完整的故障转储目录中恢复。  | 系统崩溃，但 <code>savecore</code> 目录中没有可用空间，并且您需要保存一些关键的系统故障转储信息。                            | 第 224 页中的“如何从完整的故障转储目录中恢复（可选的）” |
| 5. （可选的）禁用或启用故障转储文件的保存。 | 使用 <code>dumpadm</code> 命令禁用或启用故障转储文件的保存。缺省情况下，会启用故障转储文件的保存。                            | 第 225 页中的“如何禁用或启用故障转储的保存”       |

## 系统崩溃（概述）

系统崩溃可能是由于存在硬件故障、I/O 问题和软件错误而引起的。如果系统崩溃，则会在控制台中显示一条错误消息，然后向转储设备中写入物理内存的副本。然后，将自动重新引导系统。重新引导系统时，将执行 `savecore` 命令，以从转储设备中检索数据，并将保存的故障转储文件写入 `savecore` 目录。保存的故障转储文件为支持提供商提供了非常有价值的信息，可帮助诊断问题的原因所在。

故障转储信息以压缩格式写入 `vmdump.n` 文件中，其中 *n* 是标识故障转储的一个整数。随后，可在同一系统或其他系统中调用 `savecore` 命令以将压缩的故障转储文件扩展为名为 `unix.n` 和 `vmcore.n` 的文件对。还可以使用 `dumpadm` 命令配置重新引导时保存故障转储文件的目录。

对于具有 UFS 根文件系统的系统，缺省转储设备配置为相应的交换分区。交换分区是保留为操作系统虚拟内存备份存储的磁盘分区。因此，永久信息不会位于将被故障转储覆写的交换分区中。对于具有 Oracle Solaris ZFS 根文件系统的系统，专用的 ZFS 卷将用于交换和转储区域。有关更多信息，请参见第 219 页中的“交换区域和转储设备的 Oracle Solaris ZFS 支持”。

## 交换区域和转储设备的 Oracle Solaris ZFS 支持

如果安装 Oracle Solaris ZFS 根文件系统或使用 Oracle Solaris Live Upgrade 程序从 UFS 根文件系统迁移到 ZFS 根文件系统，将在两个 ZFS 卷上创建交换设备和转储设备。例如，使用缺省根池名称 `rpool` 时，将自动创建 `/rpool/swap` 和 `/rpool/dump` 卷。可以将交换卷和转储卷的大小调整为所选择的大小，只要新的大小支持系统运作。有关更多信息，请参见《[Oracle Solaris ZFS 管理指南](#)》中的“[对于交换和转储设备的 ZFS 支持](#)”。

如果在安装后需要修改 ZFS 交换设备或转储设备，请像在以前的发行版中那样使用 `swap` 或 `dumpadm` 命令。

有关本文中管理转储设备的信息，请参见第 221 页中的“[管理系统故障转储信息](#)”。

## x86: GRUB 引导环境中的系统崩溃

如果在 GRUB 引导环境中基于 x86 的系统发生系统崩溃，则管理 GRUB 引导归档文件的 SMF 服务 `svc:/system/boot-archive:default` 可能在下一次系统重新引导时失败。有关基于 GRUB 进行引导的更多信息，请参见《[系统管理指南：基本管理](#)》中的“[使用 GRUB 引导基于 x86 的系统（任务列表）](#)”。

## 系统故障转储文件

`savecore` 命令在系统崩溃后自动运行，以便从转储设备检索故障转储信息，并写入名为 `unix.X` 和 `vmcore.X` 的两个文件，其中 `X` 标识转储序列号。这些文件共同表示保存的系统故障转储信息。

故障转储文件有时容易与**核心转储**文件混淆，后者是在应用程序异常终止时写入的用户应用程序的映像。

故障转储文件保存在预先确定的目录中，该目录缺省为 `/var/crash/hostname`。在先前的发行版中，除非手动使系统将物理内存的映像保存到故障转储文件中，否则系统重新引导时会覆盖故障转储文件。现在，缺省情况下便可保存故障转储文件。

使用 `dumpadm` 命令可以管理系统故障转储信息。有关更多信息，请参见第 220 页中的“[dumpadm 命令](#)”。

## 保存故障转储

使用 `mdb` 实用程序可以检查控制结构、活动表、正常运行或崩溃的系统内核的内存映像，以及有关内核运行的其他信息。要最大程度地发挥 `mdb` 的作用，需要具备有关内核的详细信息，这超出了本手册的范围。有关使用此实用程序的信息，请参见 [mdb\(1\)](#) 手册页。

此外，还可以将 `savecore` 保存的故障转储发送给客户服务代表，这有助于他们分析系统崩溃的原因。

## dumpadm 命令

使用 `dumpadm` 命令可以管理 Oracle Solaris OS 中的系统故障转储信息。

- 使用 `dumpadm` 命令可以配置操作系统的故障转储。`dumpadm` 配置参数包括转储内容、转储设备和保存故障转储文件的目录。
- 转储数据以压缩格式存储在转储设备中。内核故障转储映像的大小可以是 4 GB 或更大。压缩数据意味着转储速度更快，且转储设备所需的磁盘空间更小。
- 当专用转储设备（而不是交换区域）属于转储配置的一部分时，会在后台运行故障转储文件的保存。这意味着引导系统不等待 `savecore` 命令完成，便转到下一步。在较大的内存系统中，可在 `savecore` 完成前使用系统。
- 缺省情况下，会保存 `savecore` 命令生成的系统故障转储文件。
- `savecore -L` 命令是一个新增功能，通过该功能可以获取当前运行的 Oracle Solaris OS 的故障转储。此命令用于通过在某些发生故障的状态下（例如瞬态性能问题或服务故障）捕获内存快照，对正在运行的系统进行故障排除。如果系统已启动，并且您仍可以运行一些命令，则可执行 `savecore -L` 命令将系统快照保存到转储设备，然后立即将故障转储文件写入 `savecore` 目录。由于系统仍在运行，因此如果配置了专用转储设备，则只能使用 `savecore -L` 命令。

下表介绍了 `dumpadm` 的配置参数。

| 转储参数                     | 说明                                                                            |
|--------------------------|-------------------------------------------------------------------------------|
| 转储设备                     | 在系统崩溃时临时存储转储数据的设备。当转储设备不是交换区域时， <code>savecore</code> 将在后台运行，这样可以加快引导过程进行的速度。 |
| <code>savecore</code> 目录 | 存储系统故障转储文件的目录。                                                                |
| 转储内容                     | 要转储的内存数据的类型。                                                                  |
| 最小空闲空间                   | 保存故障转储文件后 <code>savecore</code> 目录中所需的最小空闲空间量。如果未配置最小空闲空间，则缺省值为 1 MB。         |

有关更多信息，请参见 [dumpadm\(1M\)](#)。

转储配置参数由 `dumpadm` 命令管理。

## dumpadm 命令的工作原理

在系统启动过程中，`svc:/system/dumpadm:default` 服务调用 `dumpadm` 命令以配置故障转储参数。

具体地说，`dumpadm` 通过 `/dev/dump` 接口来初始化转储设备和转储内容。

完成转储配置后，`savecore` 脚本会查找故障转储文件目录的位置。然后，会调用 `savecore` 来检查故障转储，并检查故障转储目录中 `minfree` 文件的内容。

## 转储设备和卷管理器

由于可访问性和性能原因，请不要配置受卷管理产品（例如 Solaris Volume Manager）控制的专用转储设备。可以使交换区域受 Solaris Volume Manager 控制，这是推荐的做法，但应使转储设备保持独立。

## 管理系统故障转储信息

在处理系统故障转储信息时，请记住以下几点重要内容：

- 只有成为超级用户或同等角色的用户，才能访问和管理系统故障转储信息。
- 不要禁用保存系统故障转储的选项。系统故障转储文件提供了用于确定系统崩溃原因的非常有价值的方法。
- 在将重要的系统故障转储信息发送给客户服务代表之前，不要删除这些信息。

### ▼ 如何显示当前的故障转储配置

- 1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《[系统管理指南：安全性服务](#)》中的“[配置 RBAC（任务列表）](#)”。

- 2 显示当前的故障转储配置。

```
dumpadm
Dump content: kernel pages
Dump device: /dev/dsk/c0t3d0s1 (swap)
Savecore directory: /var/crash/venus
Savecore enabled: yes
Saved compressed: on
```

前面的示例输出表明：

- 转储内容是内核内存页面。
- 内核内存将在交换设备 `/dev/dsk/c0t3d0s1` 中转储。使用 `swap -l` 命令可以确定所有交换区域。
- 系统故障转储文件将写入 `/var/crash/venus` 目录。
- 已启用故障转储文件的保存。
- 以压缩格式保存故障转储。

## ▼ 如何修改故障转储配置

### 1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的“配置 RBAC（任务列表）”。

### 2 确定当前的故障转储配置。

```
dumpadm
Dump content: kernel pages
Dump device: /dev/dsk/c0t3d0s1 (swap)
Savecore directory: /var/crash/pluto
Savecore enabled: yes
Save compressed: on
```

此输出标识了运行 Oracle Solaris 10 发行版的系统的缺省转储配置。

### 3 修改故障转储配置。

```
/usr/sbin/dumpadm [-nuy] [-c content-type] [-d dump-device] [-m mink | minm | min%]
[-s savecore-dir] [-r root-dir] [-z on | off]
```

**-c content** 指定要转储的数据类型。使用 `kernel` 转储所有内核内存，使用 `all` 转储所有内存，或使用 `curproc` 转储内核内存以及在发生崩溃时其线程正在执行的进程的内存页面。缺省转储内容是内核内存。

**-d dump-device** 指定在系统崩溃时临时存储转储数据的设备。主交换设备是缺省转储设备。

**-m nnnk | nnnm | nnn%** 通过在当前的 `savecore` 目录中创建 `minfree` 文件，指定用于保存故障转储文件的最小空闲磁盘空间。可以 KB (`nnnk`)、MB (`nnnm`) 或文件系统大小百分比 (`nnn%`) 的形式指定此参数。`savecore` 命令会在写入故障转储文件之前访问此文件。如果写入故障转储文件（根据大小）会减少空闲空间量并使其低于 `minfree` 阈值，则不写入转储文件，并记录一条错误消息。有关从此情况中恢复的信息，请参见第 224 页中的“如何从完整的故障转储目录中恢复（可选的）”。

|           |                                                                                                                                                 |
|-----------|-------------------------------------------------------------------------------------------------------------------------------------------------|
| -n        | 指定重新引导系统时不应运行 <b>savecore</b> 。不推荐使用此转储配置。如果已将系统故障转储信息写入交换设备并且未启用 <b>savecore</b> ，则系统开始交换时将覆盖故障转储信息。                                           |
| -s        | 指定用于存储故障转储文件的备用目录。缺省目录为 <b>/var/crash/hostname</b> ，其中 <b>hostname</b> 是 <b>uname -n</b> 命令的输出。                                                 |
| -u        | 强制更新基于 <b>/etc/dumpadm.conf</b> 文件内容的内核转储配置。                                                                                                    |
| -y        | 修改转储配置以在重新引导时自动执行 <b>savecore</b> 命令，即此转储设置的缺省值。                                                                                                |
| -z on off | 修改转储配置以控制重新引导时 <b>savecore</b> 命令的操作。 <b>on</b> 设置允许以压缩格式保存核心文件。 <b>off</b> 设置会自动解压缩故障转储文件。由于故障转储文件可能非常大，而以压缩格式保存需要的文件系统空间较少，因此缺省值为 <b>on</b> 。 |

### 示例 17-1 修改故障转储配置

在此示例中，所有内存都会转储到专用转储设备 **/dev/dsk/c0t1d0s1** 中，并且在保存故障转储文件后必须可用的最小空闲空间为文件系统空间的 10%。

```
dumpadm
 Dump content: kernel pages
 Dump device: /dev/dsk/c0t3d0s1 (swap)
Savecore directory: /var/crash/pluto
 Savecore enabled: yes
 Save compressed: on
dumpadm -c all -d /dev/dsk/c0t1d0s1 -m 10%
 Dump content: all pages
 Dump device: /dev/dsk/c0t1d0s1 (dedicated)
Savecore directory: /var/crash/pluto (minfree = 77071KB)
 Savecore enabled: yes
 Save compressed: on
```

## ▼ 如何检查故障转储

### 1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《[系统管理指南：安全性服务](#)》中的“[配置 RBAC（任务列表）](#)”。

### 2 使用 **mdb** 实用程序检查故障转储。

```
/usr/bin/mdb [-k] crashdump-file
```

-k 通过假定文件为操作系统故障转储文件来指定内核调试模式。

`crashdump-file` 指定操作系统故障转储文件。

### 3 显示崩溃状态信息。

```
/usr/bin/mdb file-name
> ::status
.
.
.
> ::system
.
.
.
```

## 示例 17-2 检查故障转储

以下示例显示 `mdb` 实用程序的样例输出，其中包括系统信息，并列在此系统的 `/etc/system` 文件中设置的可调参数。

```
/usr/bin/mdb -k unix.0
Loading modules: [unix krtld genunix ip nfs ipc ptm]
> ::status
debugging crash dump /dev/mem (64-bit) from ozlo
operating system: 5.10 Generic (sun4u)
> ::system
set ufs_ninode=0x9c40 [0t40000]
set ncsiz=0x4e20 [0t20000]
set pt_cnt=0x400 [0t1024]
```

## ▼ 如何从完整的故障转储目录中恢复（可选的）

在此情况下，系统崩溃，但 `savecore` 目录中没有任何可用空间，并且您要保存一些关键的系统故障转储信息。

- 1 系统重新引导后，以超级用户身份登录或成为同等角色。
- 2 通过删除已发送给服务提供商的现有故障转储文件来清除 `savecore` 目录，通常为 `/var/crash/hostname`。
  - 或者，手动运行 `savecore` 命令以指定拥有足够磁盘空间的备用目录。
 

```
savecore [directory]
```



## ▼ 如何禁用或启用故障转储的保存

### 1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《[系统管理指南：安全性服务](#)》中的“[配置 RBAC（任务列表）](#)”。

### 2 在系统中禁用或启用故障转储的保存。

```
dumpadm -n | -y
```

#### 示例 17-3 禁用故障转储的保存

此示例说明如何在系统中禁用故障转储的保存。

```
dumpadm -n
Dump content: all pages
Dump device: /dev/dsk/c0t1d0s1 (dedicated)
Savecore directory: /var/crash/pluto (minfree = 77071KB)
Savecore enabled: no
Save Compressed: on
```

#### 示例 17-4 启用故障转储的保存

此示例说明如何在系统中启用故障转储的保存。

```
dumpadm -y
Dump content: all pages
Dump device: /dev/dsk/c0t1d0s1 (dedicated)
Savecore directory: /var/crash/pluto (minfree = 77071KB)
Savecore enabled: yes
Save compressed: on
```



# 各种软件问题的故障排除（任务）

本章介绍可能偶然发生并且相对易于修复的各种软件问题。对各种软件问题的故障排除包括解决不与特定软件应用程序或主题相关的问题，如不成功的重新引导或整个文件系统。以下各节中介绍了这些问题的解决方法。

以下是本章中信息的列表。

- 第 227 页中的“重新引导失败时应执行的操作”
- 第 231 页中的“x86: SMF 引导归档文件服务在系统重新引导期间失败时应执行的操作”
- 第 232 页中的“系统挂起时应执行的操作”
- 第 232 页中的“文件系统占满时应执行的操作”
- 第 233 页中的“复制或恢复后文件 ACL 丢失时应执行的操作”
- 第 233 页中的“备份问题故障排除”
- 第 235 页中的“Oracle Solaris OS 中 Common Agent Container 问题的故障排除”

## 重新引导失败时应执行的操作

如果系统未完全重新引导，或者重新引导后再次崩溃，则可能存在使系统无法成功引导的软件或硬件问题。

| 系统未引导的原因                                                                                        | 解决该问题的办法                                                                                                          |
|-------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------|
| 系统找不到 <code>/platform/‘uname -m’/kernel/unix。</code>                                            | 您可能需要在基于 SPARC 的系统上更改 PROM 中的 <code>boot-device</code> 设置。有关更改缺省引导设备的信息，请参见《系统管理指南：基本管理》中的“如何使用引导 PROM 更改缺省引导设备”。 |
| <b>Oracle Solaris 10</b> ：基于 x86 的系统中没有缺省引导设备。该消息显示为：<br><br><code>Not a UFS filesystem.</code> | <b>Oracle Solaris 10</b> ：使用配置辅助/引导磁盘引导系统，并选择要进行引导的磁盘。                                                            |

| 系统未引导的原因                                                                                          | 解决该问题的办法                                                                                                                                                                       |
|---------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Solaris 10 1/06</b> ：GRUB 引导归档文件已损坏。或者，SMF 引导归档文件服务失败。如果运行 <code>svcs -x</code> 命令，则会显示一条错误消息。 | <b>Solaris 10 1/06</b> ：引导故障安全归档文件。                                                                                                                                            |
| <code>/etc/passwd</code> 文件中存在无效项。                                                                | 有关从无效 <code>passwd</code> 文件恢复的信息，请参见《系统管理指南：基本管理》中的第 12 章“引导 Oracle Solaris 系统（任务）”。                                                                                          |
| 磁盘或其他设备存在硬件问题。                                                                                    | 检查硬件连接： <ul style="list-style-type: none"><li>■ 确保已插入设备。</li><li>■ 确保所有交换机都已正确设置。</li><li>■ 查看所有连接器和电缆，包括以太网电缆。</li><li>■ 如果所有上述操作都失败，请关闭系统电源，等待 10 到 20 秒，然后再次打开电源。</li></ul> |

如果采用上述建议仍无法解决问题，请与当地服务提供商联系。

# 忘记 root 用户口令时应执行的操作

如果忘记了 root 用户口令并且无法登录系统，必须执行以下操作：

- 使用键盘停止序列停止系统。
- **Oracle Solaris 10**：从引导服务器或安装服务器或者本地 CD-ROM 引导系统。
- 挂载根 (/) 文件系统。
- 从 `/etc/shadow` 文件中删除 root 用户口令。
- 重新引导系统。
- 登录并设置 root 用户的口令。

如果忘记了 root 用户口令并且无法登录系统，必须执行以下操作：

- 使用键盘停止序列停止系统。
- **从 Solaris 10 1/06 发行版开始**：在基于 x86 的系统上，在 Solaris 故障安全归档文件中引导系统。
- **Oracle Solaris 10**：从引导服务器或安装服务器或者本地 CD-ROM 引导系统。
- 挂载根 (/) 文件系统。
- 从 `/etc/shadow` 文件中删除 root 用户口令。
- 重新引导系统。
- 登录并设置 root 用户的口令。

《系统管理指南：基本管理》中的第 12 章“引导 Oracle Solaris 系统（任务）”完整介绍了这些过程。

---

注 – 在此发行版中，基于 SPARC 的系统不能使用基于 GRUB 的引导。

---

以下示例介绍在基于 SPARC 和 x86 的系统中，如何恢复忘记的 root 用户口令。

**示例 18-1 SPARC: 忘记 root 用户口令时应执行的操作**

以下示例说明在忘记超级用户口令时，如何通过从网络引导来进行恢复。此示例假定引导服务器已经可用。请确保在重新引导系统后应用一个新的 root 用户口令。

```
(Use keyboard abort sequence--Press Stop A keys to stop the system)
ok boot net -s
mount /dev/dsk/c0t3d0s0 /a
cd /a/etc
TERM=vt100
export TERM
vi shadow
(Remove root's encrypted password string)
cd /
umount /a
init 6
```

**示例 18-2 x86: 忘记 root 用户口令时执行基于 GRUB 的引导**

此示例假定引导服务器已经可用。请确保在重新引导系统后应用一个新的 root 用户口令。

```
GNU GRUB version 0.95 (637K lower / 3144640K upper memory)
+-----+
| be1
| be1 failsafe
| be3
| be3 failsafe
| be2
| be2 failfafa
+-----+
 Use the ^ and v keys to select which entry is highlighted.
 Press enter to boot the selected OS, 'e' to edit the
 commands before booting, or 'c' for a command-line.
```

Searching for installed OS instances...

```
An out of sync boot archive was detected on /dev/dsk/c0t0d0s0.
The boot archive is a cache of files used during boot and
should be kept in sync to ensure proper system operation.
```

```
Do you wish to automatically update this boot archive? [y,n,?] n
Searching for installed OS instances...
```

Multiple OS instances were found. To check and mount one of them read-write under /a, select it from the following list. To not mount any, select 'q'.

```
1 pool10:13292304648356142148 ROOT/be10
```

示例 18-2 x86: 忘记 root 用户口令时执行基于 GRUB 的引导 (续)

```

2 rpool:14465159259155950256 ROOT/be01

Please select a device to be mounted (q for none) [?,??,q]: 1
mounting /dev/dsk/c0t0d0s0 on /a
starting shell.
.
.
.
cd /a/etc
vi shadow
(Remove root's encrypted password string)
cd /
umount /a
reboot

```

示例 18-3 x86: 忘记 root 用户口令时引导系统

**Oracle Solaris 10** : 以下示例说明在忘记超级用户口令时, 如何通过从网络引导来进行恢复。此示例假定引导服务器已经可用。请确保在重新引导系统后应用一个新的 root 用户口令。

```

Press any key to reboot.
Resetting...
.
.
.
Initializing system
Please wait...

<<< Current Boot Parameters >>>
Boot path: /pci@0,0/pci-ide@7,1/ide@0/cmdk@0,0:a
Boot args:

Type b [file-name] [boot-flags] <ENTER> to boot with options
or i <ENTER> to enter boot interpreter
or <ENTER> to boot with defaults

<<< timeout in 5 seconds >>>

Select (b)oot or (i)nterpreter: b -s
SunOS Release 5.10 Version amd64-gate-2004-09-30 32-bit
Copyright (c) 1983, 2011, Oracle and/or its affiliates. All rights reserved.
Use is subject to license terms.
DEBUG enabled
Booting to milestone "milestone/single-user:default".
Hostname: venus
NIS domain name is example.com
Requesting System Maintenance Mode
SINGLE USER MODE

Root password for system maintenance (control-d to bypass): xxxxxx
Entering System Maintenance Mode
.

```

示例 18-3 x86: 忘记 root 用户口令时引导系统 (续)

```
.
.
mount /dev/dsk/c0t0d0s0 /a
.
.
cd /a/etc
vi shadow
(Remove root's encrypted password string)
cd /
umount /a
init 6
```

## x86: SMF 引导归档文件服务在系统重新引导期间失败时应执行的操作

**Solaris 10 1/06**：在此发行版中，如果系统崩溃，SMF 引导归档文件服务 `svc:/system/boot-archive:default` 可能会在系统重新引导时失败。如果引导归档文件服务已失败，则运行 `svcs -x` 命令时将显示以下类似消息：

```
svc:/system/boot-archive:default (check boot archive content)
 State: maintenance since Fri Jun 03 10:24:52 2005
 Reason: Start method exited with $SMF_EXIT_ERR_FATAL.
 See: http://sun.com/msg/SMF-8000-KS
 See: /etc/svc/volatile/system-boot-archive:default.log
 Impact: 48 dependent services are not running. (Use -v for list.)

svc:/network/rpc/gss:default (Generic Security Service)
 State: uninitialized since Fri Jun 03 10:24:51 2005
 Reason: Restarter svc:/network/inetd:default is not running.
 See: http://sun.com/msg/SMF-8000-5H
 See: gssd(1M)
 Impact: 10 dependent services are not running. (Use -v for list.)

svc:/application/print/server:default (LP print server)
 State: disabled since Fri Jun 03 10:24:51 2005
 Reason: Disabled by an administrator.
 See: http://sun.com/msg/SMF-8000-05
 See: lpsched(1M)
 Impact: 1 dependent service is not running. (Use -v for list.)
```

要解决问题，请执行以下操作：

1. 重新引导系统并从 GRUB 引导菜单中选择故障安全归档文件选项。
2. 系统提示重新构造引导归档文件时，回答 `y`。  
重新构造引导归档文件后，系统准备引导。
3. 要继续引导，请使用以下命令清除 SMF 引导归档文件服务。

```
svcadm clear boot-archive
```

请注意，只有超级用户或等效角色才能运行此命令。

有关重新生成 GRUB 引导归档文件的更多信息，请参见《系统管理指南：基本管理》中的[“如何在故障安全模式下引导基于 x86 的系统”](#)和 [bootadm\(1M\)](#) 手册页。

## 系统挂起时应执行的操作

如果某些软件进程出现问题，系统可能会冻结或挂起，而不是完全崩溃。遵循以下步骤可以从挂起的系统中进行恢复。

1. 确定系统是否正在运行窗口环境并遵循以下建议。如果这些建议无法解决问题，请转到步骤 2。
  - 确保指针位于正在键入命令的窗口中。
  - 如果用户意外按下了可冻结屏幕的 `Ctrl-s`，请按下 `Ctrl-q`。`Ctrl-s` 仅会冻结窗口，而不是整个屏幕。如果一个窗口被冻结，请尝试使用其他窗口。
  - 如果可能，请从网络中的其他系统中远程登录。使用 `pgrep` 命令查找挂起的进程。如果窗口系统看似已挂起，请标识进程并将其中止。
2. 按 `Ctrl-\` 强制从运行的程序中“退出”，并（可能）写出 `core` 文件。
3. 按 `Ctrl-c` 中断正在运行的程序。
4. 远程登录并尝试确定和中止使系统挂起的进程。
5. 远程登录，成为超级用户或同等角色，并重新引导系统。
6. 如果系统仍然无法响应，请强制进行故障转储并重新引导。有关强制进行故障转储和引导的信息，请参见《系统管理指南：基本管理》中的[“强制实施崩溃转储和系统重新引导”](#)。
7. 如果系统仍然无法响应，请关闭电源，等待一分钟左右，然后重新打开电源。
8. 如果根本无法使系统做出响应，请联系当地服务提供商以获取帮助。

## 文件系统占满时应执行的操作

当根 (/) 文件系统或任何其他文件系统占满时，您将在控制台窗口中看到以下消息：

```
.... file system full
```

系统占满的原因可能有多种。以下各节介绍了从已满文件系统中进行恢复的几种方案。有关定期清除旧文件和未使用的文件以防止文件系统变满的信息，请参见[第 6 章，管理磁盘使用（任务）](#)。



## 由于创建了大文件或目录导致文件系统占满

| 出现错误的原因                                                              | 解决该问题的办法                                                                                                                                 |
|----------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------|
| 有人意外地将文件或目录复制到错误位置。当应用程序崩溃并将大型 <code>core</code> 文件写入文件系统时，也会出现这种情况。 | 以超级用户身份登录或成为同等角色，在特定文件系统中使用 <code>ls -tl</code> 命令确定新创建的大文件并将其删除。有关删除 <code>core</code> 文件的信息，请参见第 82 页中的“如何查找并删除 <code>core</code> 文件”。 |

## 由于系统内存不足导致 **TMPFS** 文件系统变满

| 出现错误的原因                                               | 解决该问题的办法                                                                 |
|-------------------------------------------------------|--------------------------------------------------------------------------|
| 如果 <b>TMPFS</b> 尝试写入的内容比允许量多或者有些当前进程使用了大量内存，则可能出现此情况。 | 有关利用与 <code>tmpfs</code> 相关的错误消息进行恢复的信息，请参见 <code>tmpfs(7FS)</code> 手册页。 |

## 复制或恢复后文件 **ACL** 丢失时应执行的操作

| 出现错误的原因                                                                                                                                            | 解决该问题的办法                             |
|----------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------|
| 如果将具有 <b>ACL</b> 的文件或目录复制或恢复到 <code>/tmp</code> 目录，则会丢失 <b>ACL</b> 属性。 <code>/tmp</code> 目录通常作为临时文件系统挂载，而临时文件系统不支持 <b>ACL</b> 等 <b>UFS</b> 文件系统属性。 | 将文件复制或恢复到 <code>/var/tmp</code> 目录中。 |

## 备份问题故障排除

本节介绍在备份和恢复数据时使用的一些基本的故障排除方法。

### 备份文件系统后根 (/) 文件系统变满

备份一个文件系统，然后根 (/) 文件系统变满。未向介质中写入任何内容，`ufsdump` 命令会提示您插入介质的第二个卷。

| 出现错误的原因                                                                                                                                                                                                       | 解决该问题的办法                                                           |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------|
| 如果将无效的目标设备名用于 -f 选项， <code>ufsdump</code> 命令将向根 (/) 文件系统中的 <code>/dev</code> 目录写入内容，将其填满。例如，如果键入的是 <code>/dev/rmt/st0</code> 而不是 <code>/dev/rmt/0</code> ，则会在磁盘中创建备份文件 <code>/dev/rmt/st0</code> ，而不将其发送至磁带机。 | 在 <code>/dev</code> 目录中使用 <code>ls -tl</code> 命令确定新创建的特别大的文件并将其删除。 |

## 确保备份和恢复命令相匹配

仅能使用 `ufsrestore` 命令恢复通过 `ufsdump` 命令备份的文件。如果使用 `tar` 命令备份，则可使用 `tar` 命令进行恢复。如果使用 `ufsrestore` 命令恢复通过其他命令写入的磁带，则会显示一条错误消息，指明磁带格式不是 `ufsdump`。

## 检查以确保当前目录正确

很容易将文件恢复到错误位置。由于 `ufsdump` 命令复制文件时总是使用相对于文件系统根目录的全路径名，因此在运行 `ufsrestore` 命令之前，通常应转到文件系统的根目录。如果转到较低级别的目录，则在恢复文件后，将看到在该目录下创建的一棵完整的文件树。

## 交互式命令

使用交互命令时，将显示 `ufsrestore>` 提示符，如下例所示：

```
ufsrestore ivf /dev/rmt/0
Verify volume and initialize maps
Media block size is 126
Dump date: Fri Jan 30 10:13:46 2004
Dumped from: the epoch
Level 0 dump of /export/home on starbug:/dev/dsk/c0t0d0s7
Label: none
Extract directories from tape
Initialize symbol table.
ufsrestore >
```

# Oracle Solaris OS 中 Common Agent Container 问题的故障排除

本节将解决使用 common agent container 共享组件时可能遇到的问题。在此 Oracle Solaris 发行版中，Common Agent Container Java 程序已包括在 Oracle Solaris OS 中。该程序将为 Java 管理应用程序实现容器。通常，容器对于用户是不可见的。

以下是可能遇到的问题：

- 端口号冲突
- 超级用户口令的安全性被破坏

## 端口号冲突

缺省情况下，common agent container 占用以下端口号：

- JMX 端口 (TCP) = 11162
- SNMPAdaptor 端口 (UDP) = 11161
- SNMPAdaptor 陷阱端口 (UDP) = 11162
- Commandstream 适配器端口 (TCP) = 11163
- RMI 连接器端口 (TCP) = 11164

---

注 – 如果对 Oracle Solaris Cluster 的安装进行故障排除，端口指定会有所不同。

---

如果安装已保留其中所有的端口号，则更改 common agent container 占用的端口号，如下过程所述。

## ▼ 如何检查端口号

此过程说明如何验证端口。

- 1 成为超级用户或承担等效角色。

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的“配置 RBAC（任务列表）”。

- 2 停止 common agent container 管理守护进程。

```
/usr/sbin/cacaoadm stop
```

- 3 使用以下语法更改端口号：

```
/usr/sbin/cacaoadm set-param param=value
```

例如，要将 SNMPAdaptor 占用的端口由缺省的 11161 更改为 11165，请键入：

```
/usr/sbin/cacaoadm set-param snmp-adaptor-port=11165
```

**4 重新启动 common agent container 管理守护进程。**

```
/usr/sbin/cacaoadm start
```

## 超级用户口令的安全性被破坏

可能需要在运行 Java ES 的主机上重新生成安全密钥。例如，如果存在超级用户口令被曝露或破坏的风险，则应重新生成安全密钥。common agent container 服务使用的密钥存储在 /etc/cacao/instances/instance-name/security 目录中。以下任务说明如何为 Oracle Solaris OS 生成安全密钥。

### ▼ 如何为 Oracle Solaris OS 生成安全密钥

**1 成为超级用户或承担等效角色。**

角色包含授权和具有特权的命令。有关角色的更多信息，请参见《[系统管理指南：安全性服务](#)》中的“配置 RBAC（任务列表）”。

**2 停止 common agent container 管理守护进程。**

```
/usr/sbin/cacaoadm stop
```

**3 重新生成安全密钥。**

```
/usr/sbin/cacaoadm create-keys --force
```

**4 重新启动 common agent container 管理守护进程。**

```
/usr/sbin/cacaoadm start
```

---

注 – 对于 Oracle Sun Cluster 软件，必须将此更改传播到群集中的所有节点。

---

## 文件访问问题故障排除（任务）

---

本章提供有关解决文件访问问题（例如与不正确的权限和搜索路径有关的问题）的信息。

以下是本章中故障排除主题列表。

- 第 237 页中的“解决搜索路径的问题 (Command not found)”
- 第 239 页中的“解决文件访问问题”
- 第 240 页中的“识别网络访问问题”

用户经常由于无法访问以前可用的程序、文件或目录等问题，而向系统管理员寻求帮助。

当您遇到此类问题时，请检查是否是由以下三个方面的某一问题引起：

- 用户的搜索路径可能已更改，或者搜索路径中的目录顺序不正确。
- 文件或目录可能不具有正确的权限或所有权。
- 通过网络访问的系统配置可能已更改。

本章简要介绍如何识别上述每个方面的问题，并提出可能的解决方案。

### 解决搜索路径的问题 (Command not found)

"Command not found" 消息表示存在以下情况之一：

- 命令在系统中不可用。
- 命令目录不在搜索路径中。

要解决搜索路径问题，您需要知道存储命令的目录的路径名。

如果找到了该命令的错误版本，则一个包含同名命令的目录会出现在搜索路径中。在这种情况下，正确的目录可能稍后会出现在搜索路径中，或者根本不会出现。

使用 `echo $PATH` 命令可以显示当前的搜索路径。例如：

```
$ echo $PATH
/home/kryten/bin:/sbin:/usr/sbin:/usr/bin:/usr/dt:/usr/dist/exe
```

使用 which 命令可以确定您所运行的命令是否是一个错误版本。例如：

```
$ which acroread
/usr/doctools/bin/acroread
```

注 - which 命令在 .cshrc 文件中查找路径信息。如果从 Bourne 或 Korn shell 中执行 which 命令，并且有一个 .cshrc 文件中包含 which 命令的别名，则该命令可能会提供误导性的结果。为确保结果的准确性，请在 C shell 中使用 which 命令，或在 Korn shell 中使用 whence 命令。

## ▼ 如何诊断和更正搜索路径问题

- 1 显示当前搜索路径，以验证命令的目录是否不在路径中，或验证该路径是否有拼写错误。

```
$ echo $PATH
```

- 2 检查以下各项：

- 搜索路径是否正确？
- 该搜索路径是否在找到了该命令另一个版本的其他搜索路径之前列出？
- 命令是否位于其中一个搜索路径中？

如果需要更正路径，请转到步骤 3。否则，请转到步骤 4。

- 3 将路径添加到相应文件，如下表所示。

| Shell         | 文件                                  | 语法                                                            | 说明        |
|---------------|-------------------------------------|---------------------------------------------------------------|-----------|
| Bourne 和 Korn | \$HOME/.profile                     | \$ PATH=\$HOME/bin:/sbin:/usr/local/bin ...<br>\$ export PATH | 用冒号分隔路径名。 |
| C             | \$HOME/.cshrc<br>或<br>\$HOME/.login | hostname% set path=( ~bin /sbin /usr/local/bin ...)           | 用空格分隔路径名。 |

- 4 按以下方式激活新路径：

| Shell         | 路径所在的文件  | 激活路径的命令        |
|---------------|----------|----------------|
| Bourne 和 Korn | .profile | \$ . ./profile |

| Shell | 路径所在的文件 | 激活路径的命令                        |
|-------|---------|--------------------------------|
| C     | .cshrc  | hostname% <b>source .cshrc</b> |
|       | .login  | hostname% <b>source .login</b> |

5 验证新路径。  
\$ **which command**

示例 19-1 诊断和更正搜索路径问题

此示例显示执行 which 命令后发现，mytool 可执行文件并不存在于搜索路径中的任何目录内。

```
venus% mytool
mytool: Command not found
venus% which mytool
no mytool in /sbin /usr/sbin /usr/bin /etc /home/ignatz/bin .
venus% echo $PATH
/sbin /usr/sbin /usr/bin /etc /home/ignatz/bin
venus% vi ~/.cshrc
(Add appropriate command directory to the search path)
venus% source .cshrc
venus% mytool
```

如果找不到某一命令，请在手册页中查看其目录路径。例如，如果找不到 lpsched 命令（lp 打印机守护进程），[lpsched\(1M\)](#) 手册页会告知您路径是 /usr/lib/lp/lpsched。

# 解决文件访问问题

当用户无法访问以前可以访问的文件或目录时，可能是这些文件或目录的权限或所有权已经更改。

## 更改文件和组的所有权

通常，文件和目录的所有权会因为有人以超级用户的身份编辑文件而发生变化。在为新用户创建起始目录时，务必要使该用户成为起始目录中的点 (.) 文件的所有者。如果用户不是 "." 的所有者，他们将无法在自己的起始目录中创建文件。

当组所有权发生变化或从 /etc/group 数据库中删除了某用户所属的组时，也可能会发生访问问题。

有关如何更改存在访问问题的文件的权限或所有权的信息，请参见《[系统管理指南：安全性服务](#)》中的第 6 章“控制对文件的访问（任务）”。

## 识别网络访问问题

如果用户在使用 `rcp` 远程复制命令通过网络复制文件时遇到问题，则可能是通过设置权限对远程系统中的目录和文件进行了访问限制。另一个可能的原因是未将远程系统和本地系统配置为允许访问。

有关网络访问以及通过 AutoFS 访问系统时遇到的问题的信息，请参见 [《系统管理指南：网络服务》](#) 中的“NFS 故障排除的策略”。



## 解决 UFS 文件系统不一致问题（任务）

---

本章介绍 `fsck` 错误消息以及为解决错误消息问题所能做出的可能响应。

---

注 – 从 Solaris 10 6/06 发行版开始，运行 `fsck` 命令时显示的错误消息已更改。本章包括修改过的 `fsck` 错误消息。有关在版本低于 Solaris 10 6/06 发行版的系统上运行 `fsck` 命令的适用信息，请参阅标为“Oracle Solaris 10”的章节中的错误消息。有关当前发行版中所有 `fsck` 改进的详细说明，请参见《系统管理指南：设备和文件系统》。

---

以下是本章中信息的列表：

- 第 243 页中的“一般 `fsck` 错误消息”
- 第 245 页中的“初始化阶段的 `fsck` 消息”
- 第 247 页中的“阶段 1：检查块和大小消息”
- 第 252 页中的“阶段 1B：重新扫描更多 DUPS 消息”
- **Oracle Solaris 10**：第 251 页中的“Oracle Solaris 10：阶段 1B：重新扫描更多 DUPS 消息”
- 第 252 页中的“阶段 2：检查路径名消息”
- 第 258 页中的“阶段 3：检查连通性消息”
- 第 260 页中的“阶段 4：检查引用计数消息”
- 第 263 页中的“阶段 5：检查柱面组消息”
- **Oracle Solaris 10**：第 264 页中的“阶段 5：检查柱面组消息”
- 第 265 页中的“`fsck` 摘要消息”
- **Oracle Solaris 10**：第 265 页中的“清除阶段消息”

有关 `fsck` 命令以及如何使用该命令检查文件系统完整性的信息，请参见《系统管理指南：设备和文件系统》中的第 22 章“检查 UFS 文件系统一致性（任务）”。

# fsck 错误消息

通常，fsck 命令在系统突然停机（其中未将最新文件系统更改写入磁盘）之后以非交互方式运行，以便整理文件系统。整理会自动修复所有基本的文件系统不一致问题，但不会尝试修复更严重的错误。整理文件系统时，fsck 命令将修复它所预期的突然停机所导致的不一致问题。对于更严重的情况，该命令将报告错误并终止。

以交互方式运行 fsck 命令时，它会报告发现的每个不一致问题并修复无害的错误。但对于更严重的错误，该命令将报告一致性问题并提示您选择响应。运行带有 -y 或 -n 选项的 fsck 命令时，响应会预定义为执行或不执行 fsck 命令针对每种错误状态建议的缺省响应。

某些更正操作会导致数据丢失。数据丢失的数量和严重性可通过 fsck 诊断输出确定。

fsck 命令是一个多遍文件系统检查程序。每遍检查操作都将调用具有不同消息集的 fsck 命令的不同阶段。初始化之后，fsck 命令会对每个文件系统执行多遍操作，检查块和大小、路径名、连通性、引用计数和空闲块图（可能重新生成该图）。此外还会执行一些清除操作。

fsck 命令的 UFS 版本所执行的操作阶段（各遍操作）包括：

- 初始化
- 阶段 1—检查块和大小
- 阶段 2a—检查重复的名称
- 阶段 2b—检查路径名
- 阶段 3—检查连通性
- 阶段 3b—验证阴影/ACL
- 阶段 4—检查引用计数
- 阶段 5—检查柱面组

以下各节介绍在每个阶段中可能检测到的错误状态、生成的消息和提示以及可以做出的可能响应。

第 243 页中的“一般 fsck 错误消息”中介绍了可能在多个阶段中出现的消息。对于其他消息，将根据消息产生的阶段按字母顺序分别加以介绍。

下表列出了 fsck 错误消息中包括的许多缩写。

表 20-1 错误消息缩写

| 缩写  | 含义                             |
|-----|--------------------------------|
| BLK | Block number（块编号）              |
| DUP | Duplicate block number（重复的块编号） |
| DIR | Directory name（目录名）            |
| CG  | Cylinder group（柱面组）            |

表 20-1 错误消息缩写 (续)

| 缩写    | 含义                                      |
|-------|-----------------------------------------|
| MTIME | Time file was last modified (上次修改文件的时间) |
| UNREF | Unreferenced (未引用)                      |

许多消息还包括变量字段，例如 inode 编号，在本书中变量字段以斜体字表示，例如 *inode-number*。例如，以下屏幕消息：

INCORRECT BLOCK COUNT I=2529

显示为：

INCORRECT BLOCK COUNT I=*inode-number*

## 一般 fsck 错误消息

在初始化之后，任何阶段中都可能显示本节的错误消息。尽管这些消息提供了可供继续的选项，但通常最佳做法是将其视为致命错误。它们反映了严重的系统故障，应立即进行处理。遇到此类消息时，请通过输入 **n(o)** 终止程序。如果无法确定引起问题的原因，请与当地服务提供商或其他有资格的人员联系。

CANNOT SEEK: BLK *disk-block-number* (CONTINUE)

### Oracle Solaris 10：

CANNOT SEEK: BLK *block-number* (CONTINUE)

#### 原因

移动到文件系统中指定块编号 *disk-block-number* 的请求失败。此消息表示存在严重问题，很可能是硬件故障。

**Oracle Solaris 10：**移动到文件系统中指定块编号 *block-number* 的请求失败。此消息表示存在严重问题，很可能是硬件故障。

如果要继续进行文件系统检查，**fsck** 将重试该移动并显示无法移动的扇区编号的列表。如果块属于虚拟内存高速缓存存储区的一部分，**fsck** 将终止并显示致命 I/O 错误消息。

#### 操作

- 如果磁盘遇到硬件问题，该问题将一直存在。再次运行 **fsck**，重新检查文件系统。
- 如果重新检查失败，请与当地服务提供商或其他有资格的人员联系。

CANNOT READ: DISK BLOCK *disk-block-number*: I/O ERROR  
CONTINUE?

**Oracle Solaris 10 :**

CANNOT READ: DISK BLOCK *block-number*: I/O ERROR  
CONTINUE?

**原因**

读取文件系统中指定块编号 *disk-block-number* 的请求失败。此消息表示存在严重问题，可能是硬件故障。

**Oracle Solaris 10 :** 读取文件系统中指定块编号 *block-number* 的请求失败。此消息表示存在严重问题，可能是硬件故障。

如果要继续进行文件系统检查，**fsck** 将重试读取并显示无法读取的扇区编号列表。如果块属于虚拟内存高速缓存存储区的一部分，**fsck** 将终止并显示致命 I/O 错误消息。如果 **fsck** 尝试回写读取失败的一个块，则会显示以下消息：

WRITING ZERO'ED BLOCK *sector-numbers* TO DISK

**操作**

如果磁盘遇到硬件问题，该问题将一直存在。再次运行 **fsck**，重新检查文件系统。如果重新检查失败，请与当地服务提供商或其他有资格的人员联系。

CANNOT WRITE: BLK *disk-block-number* (CONTINUE)

**Oracle Solaris 10 :**

CANNOT WRITE: BLK *block-number* (CONTINUE)

**原因**

写入文件系统中指定块编号 *disk-block-number* 的请求失败。

如果要继续进行文件系统检查，**fsck** 将重试写入并显示无法写入的扇区编号列表。如果块属于虚拟内存高速缓存存储区的一部分，**fsck** 将终止并显示致命 I/O 错误消息。

**Oracle Solaris 10 :** 写入文件系统中指定块编号 *block-number* 的请求失败。

如果要继续进行文件系统检查，**fsck** 将重试写入并显示无法写入的扇区编号列表。如果块属于虚拟内存高速缓存存储区的一部分，**fsck** 将终止并显示致命 I/O 错误消息。

**操作**

磁盘可能设置了写保护。检查驱动器上的写保护锁。如果磁盘出现硬件问题，该问题可能一直存在。再次运行 **fsck**，重新检查文件系统。如果写保护不会导致问题或者重新检查失败，请与当地服务提供商或其他有资格的人员联系。

## 初始化阶段的 fsck 消息

在初始化阶段，将检查命令行语法。在可以执行文件系统检查之前，fsck 将设置表并打开文件。

本节中的消息与命令行选项、内存请求、打开文件、文件状态、文件系统大小检查和创建临时文件所导致的错误状态相关。当 fsck 整理文件系统时，所有此类初始化错误都会使其终止。

Can't roll the log for *device-name*.

DISCARDING THE LOG MAY DISCARD PENDING TRANSACTIONS.  
DISCARD THE LOG AND CONTINUE?

### 原因

fsck 在检查文件系统错误之前，无法刷新日志 UFS 文件系统的事务日志。

### 操作

回答是意味着在日志中但尚未应用于文件系统的文件系统操作将丢失。在此情况下，fsck 将运行它通常运行的相同检查，并在阶段 5 中提出以下问题：

FREE BLK COUNT(S) WRONG IN SUPERBLK (SALVAGE)

在此时回答是将回收用于日志的块。下次在启用日志的情况下挂载文件系统时，将重新创建日志。

回答否将保留日志并退出，但文件系统不可挂载。

bad inode number *inode-number* to ginode

### 原因

由于不存在 inode *inode-number* 而发生内部错误。fsck 退出。

### 操作

请与当地服务提供商或其他有资格的人员联系。

cannot alloc *size-of-block map* bytes for blockmap  
cannot alloc *size-of-free map* bytes for freemap  
cannot alloc *size-of-state map* bytes for statemap  
cannot alloc *size-of-lncntp* bytes for lncntp

### 原因

对内部表的内存请求失败。fsck 终止。此消息表示存在严重的系统故障，应立即进行处理。如果其他进程正在使用大量系统资源，则可能会出现此情况。

### 操作

中止其他进程可能会解决该问题。如果未解决，请与当地服务提供商或其他有资格的人员联系。

Can't open checklist file: *filename*

**原因**

无法打开文件系统核对表文件 *filename*（通常为 */etc/vfstab*）以供读取。fsck 终止。

**操作**

检查该文件是否存在以及其访问模式是否允许读取访问。

**Can't open *filename***

**原因**

fsck 无法打开文件系统 *filename*。以交互方式运行时，fsck 将忽略此文件系统并继续检查给定的下一个文件系统。

**操作**

检查是否允许对文件系统的原始设备文件进行读取和写入访问。

**Can't stat root**

**原因**

对有关根目录的统计信息的 fsck 请求失败。fsck 终止。

**操作**

此消息表示存在严重的系统故障。请与当地服务提供商或其他有资格的人员联系。

**Can't stat *filename***

**Can't make sense out of name *filename***

**原因**

对有关文件系统 *filename* 的统计信息的 fsck 请求失败。以交互方式运行时，fsck 将忽略此文件系统并继续检查给定的下一个文件系统。

**操作**

检查该文件系统是否存在并检查其访问模式。

***filename*: (NO WRITE)**

**原因**

已指定 -n 选项，或者 fsck 无法打开文件系统 *filename* 以供写入。当 fsck 以禁止写入模式运行时，将显示所有诊断消息，但 fsck 不会尝试进行任何修复。

**操作**

如果未指定 -n，请检查所指定文件的类型。它可能是一个常规文件的名称。

**IMPOSSIBLE MINFREE=*percent* IN SUPERBLOCK (SET TO DEFAULT)**

**原因**

超级块最小空间百分比大于 99% 或小于 0%。

**操作**

要将 minfree 参数设置为缺省的 10%，请在缺省提示符下键入 y。要忽略该错误状态，请在缺省提示符下键入 n。

*filename*: BAD SUPER BLOCK: *message*  
 USE AN ALTERNATE SUPER-BLOCK TO SUPPLY NEEDED INFORMATION;  
 e.g., fsck[-f ufs] -o b=# [special ...]  
 where # is the alternate superblock. See fsck\_ufs(1M)

#### 原因

超级块已损坏。

#### 操作

可能显示以下消息之一：

```
CPG OUT OF RANGE
FRAGS PER BLOCK OR FRAGSIZE WRONG
INODES PER GROUP OUT OF RANGE
INOPB NONSENSICAL RELATIVE TO BSIZE
MAGIC NUMBER WRONG
NCG OUT OF RANGE
NCYL IS INCONSISTENT WITH NCG*CPG
NUMBER OF DATA BLOCKS OUT OF RANGE
NUMBER OF DIRECTORIES OUT OF RANGE
ROTATIONAL POSITION TABLE SIZE OUT OF RANGE
SIZE OF CYLINDER GROUP SUMMARY AREA WRONG
SIZE TOO LARGE
BAD VALUES IN SUPERBLOCK
```

尝试使用备用超级块重新运行 fsck。最佳选择是指定块 32。通过在分片中运行 newfs -N 命令可以找到超级块的备用副本。务必要指定 -N 选项；否则，newfs 将覆写现有的文件系统。

UNDEFINED OPTIMIZATION IN SUPERBLOCK (SET TO DEFAULT)

#### 原因

超级块优化参数既不是 OPT\_TIME 也不是 OPT\_SPACE。

#### 操作

要最大程度地减少在文件系统中执行操作的时间，请在 SET TO DEFAULT 提示符下键入 y。要忽略此错误状态，请键入 n。

## 阶段 1：检查块和大小消息

此阶段检查 inode 列表。它会报告在以下情况下遇到的错误状态：

- 检查 inode 类型
- 设置零链接计数表
- 检查坏块或重复块的 inode 块编号
- 检查 inode 大小
- 检查 inode 格式

整理文件系统时，此阶段中除了 INCORRECT BLOCK COUNT、PARTIALLY TRUNCATED INODE、PARTIALLY ALLOCATED INODE 和 UNKNOWN FILE TYPE 之外的所有错误都会终止 fsck。

阶段 1 中可能出现以下消息（按字母顺序）：

*block-number* BAD I=*inode-number*

原因

Inode *inode-number* 包含一个块编号 *block-number*，其编号小于文件系统中第一个数据块的编号或大于文件系统中最后一个块的编号。如果 inode *inode-number* 有太多的块编号超出文件系统范围，则此错误状态可能会在阶段 1 中生成 EXCESSIVE BAD BLKS 错误消息。此错误状态会在阶段 2 和 4 中生成 BAD/DUP 错误消息。

操作

N/A

BAD MODE: MAKE IT A FILE?

原因

给定 inode 的状态都设置为 1，表示文件系统损坏。除非此消息在 `fsck -y` 运行后重复显示，否则此消息不表示物理磁盘损坏。

操作

键入 y，以便将 inode 重新初始化为合理值。

BAD STATE *state-number* TO BLKERR

原因

内部错误使 fsck 状态图变得杂乱，因此它会显示不可能的值 *state-number*。fsck 将立即退出。

操作

请与当地服务提供商或其他有资格的人员联系。

*fragment-number* DUP I=*inode-number*

**Oracle Solaris 10：**

*block-number* DUP I=*inode-number*

原因

Inode *inode-number* 包含一个块编号 *fragment-number*，该块编号已被该 inode 或其他 inode 请求。如果 inode *inode-number* 具有太多已被该 inode 或其他 inode 请求的块编号，此错误状态可能会在阶段 1 中生成 EXCESSIVE DUP BLKS 错误消息。此错误状态将调用阶段 1B 并在阶段 2 和 4 中生成 BAD/DUP 错误消息。

**Oracle Solaris 10：** Inode *inode-number* 包含一个块编号 *block-number*，该块编号已被该 inode 或其他 inode 请求。如果 inode *inode-number* 具有太多已被该 inode 或其他 inode 请求的块编号，此错误状态可能会在阶段 1 中生成 EXCESSIVE DUP BLKS 错误消息。此错误状态将调用阶段 1B 并在阶段 2 和 4 中生成 BAD/DUP 错误消息。

操作

N/A



## DUP TABLE OVERFLOW (CONTINUE)

## 原因

fsck 无法分配用于跟踪重复片段的内存。如果指定了 `-o p` 选项，程序将终止。

**Oracle Solaris 10**：包含重复块编号的 fsck 的内部表中没有更多空间。如果指定了 `-o p` 选项，程序将终止。

## 操作

要继续该程序，请在 `CONTINUE` 提示符下键入 `y`。如果出现此错误，则无法进行完整的文件系统检查。如果找到其他的重复片段，将重复出现此错误状态。增加可用的虚拟内存量（通过中止一些进程，增加交换空间）并再次运行 fsck，以重新检查文件系统。要终止该程序，请键入 `n`。

**Oracle Solaris 10**：要继续该程序，请在 `CONTINUE` 提示符下键入 `y`。如果出现此错误，则无法进行完整的文件系统检查。如果找到其他重复块，将重复出现此错误状态。增加可用的虚拟内存量（通过中止一些进程，增加交换空间）并再次运行 fsck，以重新检查文件系统。要终止该程序，请键入 `n`。

## EXCESSIVE BAD FRAGMENTS I=inode-number (CONTINUE)

**Oracle Solaris 10**：

## EXCESSIVE BAD BLOCKS I=inode-number (CONTINUE)

## 原因

太多（通常多于 10 个）片段表示磁盘地址无效。如果指定 `-o p`（整理）选项，程序将终止。

**Oracle Solaris 10**：太多（通常多于 10 个）块的编号小于文件系统中第一个数据块的编号或大于文件系统中与 inode *inode-number* 关联的最后一个块的编号。如果指定 `-o p`（整理）选项，程序将终止。

## 操作

要继续该程序，请在 `CONTINUE` 提示符下键入 `y`。如果出现此错误，则无法进行完整的文件系统检查。应再次运行 fsck，以重新检查文件系统。要终止该程序，请键入 `n`。

## EXCESSIVE DUP BLKSDUPLICATE FRAGMENTS I=inode-number (CONTINUE)

**Oracle Solaris 10**：

## EXCESSIVE DUP BLKS I=inode-number (CONTINUE)

## 原因

该 inode、其他 inode 或可用列表已请求太多（通常多于 10 个）片段。如果指定了 `-o p` 选项，程序将终止。

**Oracle Solaris 10**：该 inode、其他 inode 或可用列表已请求太多（通常多于 10 个）块。如果指定了 `-o p` 选项，程序将终止。

**操作**

要继续该程序，请在 **CONTINUE** 提示符下键入 **y**。如果出现此错误，则无法进行完整的文件系统检查。应再次运行 **fsck**，以重新检查文件系统。要终止该程序，请键入 **n**。

INCORRECT DISK BLOCK COUNT I=inode-number (*number-of-BAD-DUP-or-missing-blocks* should be *number-of-blocks-in-filesystem*) (CORRECT)

**Oracle Solaris 10 :**

INCORRECT BLOCK COUNT I=inode-number (*number-of-BAD-DUP-or-missing-blocks* should be *number-of-blocks-in-filesystem*) (CORRECT)

**原因**

inode *inode-number* 的磁盘块计数不正确。整理时，**fsck** 将更正该计数。

**Oracle Solaris 10 :** inode *inode-number* 的块计数是 *number-of-BAD-DUP-or-missing-blocks*，但应该是 *number-of-blocks-in-filesystem*。整理时，**fsck** 将更正该计数。

**操作**

要按 *number-of-blocks-in-file* 更正 inode *inode-number* 的磁盘块计数，请在 **CORRECT** 提示符下键入 **y**。

**Oracle Solaris 10 :** 要用 *number-of-blocks-in-filesystem* 替换 inode *inode-number* 的块计数，请在 **CORRECT** 提示符下键入 **y**。要终止该程序，请键入 **n**。

LINK COUNT TABLE OVERFLOW (CONTINUE)

**原因**

对于包含链接计数为零的已分配 inode 的 **fsck** 而言，内部表内已没有可用空间。如果指定了 **-o p**（整理）选项，程序将退出，必须手动完成 **fsck**。

**操作**

要继续该程序，请在 **CONTINUE** 提示符下键入 **y**。如果找到具有零链接计数的其他已分配 inode，则会重复出现此错误状态。如果出现此错误，则无法进行完整的文件系统检查。应再次运行 **fsck**，以重新检查文件系统。通过中止一些进程或增加交换空间来增加可用虚拟内存，然后再次运行 **fsck**。要终止该程序，请键入 **n**。

PARTIALLY ALLOCATED INODE I=inode-number (CLEAR)

**原因**

未分配 Inode *inode-number*，也未取消其分配。如果指定了 **-o p**（整理）选项，则会清除该 inode。

**操作**

要通过清空内容来取消分配 inode *inode-number*，请键入 **y**。这可能会在阶段 2 中对指向此 inode 的每个目录项生成 **UNALLOCATED** 错误状态。要忽略该错误状态，请键入 **n**。仅当要采取其他措施来修复问题时，才可以不做出响应。

PARTIALLY TRUNCATED INODE *I=inode-number* (SALVAGE)

#### 原因

fsck 发现大小小于为其分配的片段数的 inode *inode-number*。仅当系统在截断文件的过程中崩溃时，才会出现此情况。整理文件系统时，fsck 将以指定大小完成截断。

**Oracle Solaris 10**：fsck 发现 inode *inode-number* 的大小小于为其分配的块数。仅当系统在截断文件的过程中崩溃时，才会出现此情况。整理文件系统时，fsck 将以指定大小完成截断。

#### 操作

要以 inode 中指定的大小完成截断，请在 SALVAGE 提示符下键入 y。要忽略此错误状态，请键入 n。

UNKNOWN FILE TYPE *I=inode-number* (CLEAR)

#### 原因

inode *inode-number* 的模式字表明，该 inode 不是管道、字符设备、块设备、常规文件、符号链接、FIFO 文件或目录 inode。如果指定了 -o p 选项，则会清除 inode。

**Oracle Solaris 10**：inode *inode-number* 的模式字表明，该 inode 不是管道、特殊字符 inode、特殊块 inode、常规 inode、符号链接、FIFO 文件或目录 inode。如果指定了 -o p 选项，则会清除 inode。

#### 操作

要通过清空其内容来取消分配 inode *inode-number*（这将导致阶段 2 中指向此 inode 的每个目录项都出现 UNALLOCATED 错误状态），请在 CLEAR 提示符下键入 y。要忽略此错误状态，请键入 n。

## Oracle Solaris 10：阶段 1B：重新扫描更多 DUPS 消息

本节包含当前发行版中的阶段 1B fsck 消息。

在文件系统中找到重复片段时，将显示此消息：

fragment DUP *I=inode-number*

#### 原因

Inode *inode-number* 包含已由该 inode 或其他 inode 请求的片段编号 *fragment-number*。此错误状态会在阶段 2 中生成 BAD/DUP 错误消息。通过检查此错误状态和阶段 1 中的 DUP 错误状态，可以确定具有重复片段的 inode。在 fsck 运行时生成的重复片段报告可以简化此操作。

#### 操作

发现重复块时，将重新扫描文件系统，以查找以前请求过该块的 inode。

## 阶段 1B：重新扫描更多 DUPS 消息

本节包含在 Oracle Solaris 10 发行版和其他支持的 Solaris 发行版中显示的 fsck 消息。

在文件系统中发现重复块时，将显示此消息：

```
block-number DUP I=inode-number
```

原因

Inode *inode-number* 包含已由该 inode 或其他 inode 请求的块编号 *block-number*。此错误状态会在阶段 2 中生成 BAD/DUP 错误消息。通过检查此错误状态和阶段 1 中的 DUP 错误状态，可以确定具有重复块的 inode。

操作

发现重复块时，将重新扫描文件系统，以查找以前请求过该块的 inode。

## 阶段 2：检查路径名消息

此阶段将删除指向阶段 1 和 1B 中所找到的坏 inode 的目录项。它将报告下列原因所导致的错误状态：

- 不正确的根 inode 模式和状态
- 目录 inode 指针超出范围
- 目录项指向坏 inode
- 目录完整性检查

整理（-o -p 选项）文件系统时，此阶段中的所有错误都会终止 fsck，但与不是块大小倍数的目录、重复块和坏块、超出范围的 inode 以及多余的硬链接相关的错误除外。

阶段 2 中可能出现以下消息（按字母顺序）：

```
BAD INODE state-number TO DESCEND
```

原因

fsck 内部错误将一个无效状态 *state-number* 传递给由文件系统目录结构向下派生的例程。fsck 退出。

操作

如果显示此错误消息，请与当地服务提供商或其他有资格的人员联系。

```
BAD INODE NUMBER FOR '.' I=inode-number OWNER=UID MODE=file-mode
SIZE=file-size MTIME=modification-time DIR=filename (FIX)
```

原因

找到一个目录 *inode-number*，其用于 "." 的 inode 编号不等于 *inode-number*。

操作

要更改 "." 的 inode 编号以使其等于 *inode-number*，请在 FIX 提示符下键入 y。要使 "." 的 inode 编号保持不变，请键入 n。

BAD INODE NUMBER FOR '..' I=*inode-number* OWNER=*UID* MODE=*file-mode*  
SIZE=*file-size* MTIME=*modification-time* DIR=*filename* (FIX)

#### 原因

找到一个目录 *inode-number*，其用于".."的 inode 编号不等于 *inode-number* 的父级。

#### 操作

要更改".."的 inode 编号以使其等于 *inode-number* 的父级，请在 **FIX** 提示符下键入 **y**。（请注意，根 inode 中的".."指向本身。）要使".."的 inode 编号保持不变，请键入 **n**。

BAD RETURN STATE *state-number* FROM DESCEND

#### 原因

fsck 内部错误从文件系统目录结构向下派生的例程中返回不可能的状态 *state-number*。fsck 退出。

#### 操作

如果显示此消息，请与当地服务提供商或其他有资格的人员联系。

BAD STATE *state-number* FOR ROOT INODE

#### 原因

内部错误为根 inode 指定了不可能的状态 *state-number*。fsck 退出。

#### 操作

如果显示此错误消息，请与当地服务提供商或其他有资格的人员联系。

BAD STATE *state-number* FOR INODE=*inode-number*

#### 原因

内部错误为 inode *inode-number* 指定了不可能的状态 *state-number*。fsck 退出。

#### 操作

如果显示此错误消息，请与当地服务提供商或其他有资格的人员联系。

DIRECTORY TOO SHORT I=*inode-number* OWNER=*UID* MODE=*file-mode*  
SIZE=*file-size* MTIME=*modification-time* DIR=*filename* (FIX)

#### 原因

发现大小 *file-size* 小于最小目录大小的目录 *filename*。此时将显示所有者 *UID*、模式 *file-mode*、大小 *file-size*、修改时间 *modification-time* 和目录名称 *filename*。

#### 操作

要将目录的大小增大至最小目录大小，请在 **FIX** 提示符下键入 **y**。要忽略此目录，请键入 **n**。

DIRECTORY *filename*: LENGTH *file-size* NOT MULTIPLE OF *disk-block-size* (ADJUST)

#### Oracle Solaris 10 :

DIRECTORY *filename*: LENGTH *file-size* NOT MULTIPLE OF *block-number* (ADJUST)

**原因**

找到一个目录 *filename*，其大小 *file-size* 不是目录块大小 *disk-block-size* 的倍数。

**Oracle Solaris 10 :**

找到一个目录 *filename*，其大小 *file-size* 不是目录块大小 *block-number* 的倍数。

**操作**

要将长度舍入为合适的磁盘块大小，请键入 *y*。整理文件系统（*-o p* 选项）时，*fsck* 只显示警告并调整目录。要忽略此状态，请键入 *n*。

**Oracle Solaris 10 :**

要将长度舍入为合适的块大小，请键入 *y*。整理文件系统（*-o p* 选项）时，*fsck* 只显示警告并调整目录。要忽略此状态，请键入 *n*。

```
DIRECTORY CORRUPTED I=inode-number OWNER=UID MODE=file-mode
SIZE=file-size MTIME=modification-time DIR=filename (SALVAGE)
```

**原因**

找到了具有不一致的内部状态的目录。

**操作**

要丢弃达到下一目录边界（通常为 512 字节边界）的所有项，请在 *SALVAGE* 提示符下键入 *y*。此极端操作最多可丢弃 42 个项。仅在其他恢复努力失败之后，才能采取此操作。要跳到下一个目录边界并继续阅读，但不修改目录，请键入 *n*。

```
DUP/BAD I=inode-number OWNER=O MODE=M SIZE=file-size
MTIME=modification-time TYPE=filename (REMOVE)
```

**原因**

阶段 1 或阶段 1B 发现与目录或文件项 *filename*、inode *inode-number* 相关的重复片段或坏片段。此时将显示所有者 *UID*、模式 *file-mode*、大小 *file-size*、修改时间 *modification-time* 以及目录名称或文件名 *filename*。如果指定了 *-op*（整理）选项，则会删除重复/坏片段。

**Oracle Solaris 10 :**

阶段 1 或阶段 1B 发现与目录或文件项 *filename*、inode *inode-number* 相关的重复块或坏块。此时将显示所有者 *UID*、模式 *file-mode*、大小 *file-size*、修改时间 *modification-time* 以及目录名称或文件名 *filename*。如果指定了 *-op*（整理）选项，则会删除重复/坏块。

**操作**

要删除该目录或文件项 *filename*，请在 *REMOVE* 提示符下键入 *y*。要忽略此错误状态，请键入 *n*。

```
DUPS/BAD IN ROOT INODE (REALLOCATE)
```

**原因**

阶段 1 或阶段 1B 在文件系统的根 inode（inode 编号 20）中发现重复片段或坏片段。

**Oracle Solaris 10 :**

阶段 1 或阶段 1B 在文件系统的根 inode（通常为 inode 编号 2）中发现重复块或坏块。

**操作**

要清除根 inode 的现有内容并对其重新分配，请在 **REALLOCATE** 提示符下键入 **y**。通常在根 inode 中找到的文件和目录将在阶段 3 中恢复，并被放入 **lost+found** 目录中。如果尝试分配根时失败，**fsck** 将退出并显示以下消息：**CANNOT ALLOCATE ROOT INODE**。键入 **n** 获取 **CONTINUE** 提示符。键入 **y** 以对 **CONTINUE** 提示符做出响应，忽略根 inode 中的 **DUPS/BAD** 错误状态并继续运行文件系统检查。如果根 inode 不正确，则可能会生成许多其他错误消息。键入 **n** 终止程序。

```
EXTRA '..' ENTRY I=inode-number OWNER=UID MODE=file-mode
SIZE=file-size MTIME=modification-time DIR=filename (FIX)
```

**原因**

找到一个 "." 有多个项的目录 *inode-number*。

**操作**

要删除 "." 的多余项，请在 **FIX** 提示符下键入 **y**。要使该目录保持不变，请键入 **n**。

```
EXTRA '..' ENTRY I=inode-number OWNER=UID MODE=file-mode
SIZE=file-size MTIME=modification-time DIR=filename (FIX)
```

**原因**

找到一个 ".."（父目录）有多个项的目录 *inode-number*。

**操作**

要删除 ".."（父目录）的多余项，请在 **FIX** 提示符下键入 **y**。要使该目录保持不变，请键入 **n**。

```
hard-link-number IS AN EXTRANEIOUS HARD LINK TO A DIRECTORY filename (REMOVE)
```

**原因**

**fsck** 发现链接到目录 *filename* 的多余硬链接 *hard-link-number*。整理（**-op** 选项）时，**fsck** 将忽略多余硬链接。

**操作**

要删除多余项 *hard-link-number*，请在 **REMOVE** 提示符下键入 **y**。要忽略该错误状态，请键入 **n**。

```
inode-number OUT OF RANGE I=inode-number NAME=filename (REMOVE)
```

**原因**

目录项 *filename* 有一个 inode 编号 *inode-number* 大于 inode 列表的最后一个数。如果指定了 **-p**（整理）选项，则会自动删除该 inode。

**操作**

要删除目录项 *filename*，请在 REMOVE 提示符下键入 y。要忽略该错误状态，请键入 n。

```
MISSING '.' I=inode-number OWNER=UID MODE=file-mode SIZE=file-size
MTIME=modification-time DIR=filename (FIX)
```

**原因**

找到一个第一项（用于 "." 的项）尚未分配的目录 *inode-number*。

**操作**

要为 "." 生成一个 inode 编号等于 *inode-number* 的项，请在 FIX 提示符下键入 y。要使该目录保持不变，请键入 n。

```
MISSING '.' I=inode-number OWNER=UID MODE=file-mode SIZE=file-size
MTIME=modification-time DIR=filename CANNOT FIX, FIRST ENTRY IN
DIRECTORY CONTAINS filename
```

**原因**

找到一个第一项为 *filename* 的目录 *inode-number*。fsck 无法解决此问题。

**操作**

如果显示此错误消息，请与当地服务提供商或其他有资格的人员联系。

```
MISSING '.' I=inode-number OWNER=UID MODE=file-mode SIZE=file-size
MTIME=modification-time DIR=filename CANNOT FIX, INSUFFICIENT
SPACE TO ADD '.'
```

**原因**

找到一个第一项不是 "." 的目录 *inode-number*。fsck 无法解决此问题。

**操作**

如果显示此错误消息，请与当地服务提供商或其他有资格的人员联系。

```
MISSING '..' I=inode-number OWNER=UID MODE=file-mode SIZE=file-size
MTIME=modification-time DIR=filename (FIX)
```

**原因**

找到一个第二项尚未分配的目录 *inode-number*。

**操作**

要为 ".." 生成一个 inode 编号等于 *inode-number* 的父级的项，请在 FIX 提示符下键入 y。（请注意，根 inode 中的 ".." 指向本身。）要使该目录保持不变，请键入 n。

```
MISSING '..' I=inode-number OWNER=UID MODE=file-mode SIZE=file-size
MTIME=modification-time DIR=filename CANNOT FIX, SECOND ENTRY IN
DIRECTORY CONTAINS filename
```

**原因**

找到一个第二项为 *filename* 的目录 *inode-number*。fsck 无法解决此问题。

**操作**

如果显示此错误消息，请与当地服务提供商或其他有资格的人员联系。



```
MISSING '..' I=inode-number OWNER=UID MODE=file-mode SIZE=file-size
MTIME=modification-time DIR=filename CANNOT FIX, INSUFFICIENT SPACE
TO ADD '..'
```

#### 原因

找到一个第二项不是 ".."（父目录）的目录 *inode-number*。fsck 无法解决此问题。

#### 操作

如果显示此错误消息，请与当地服务提供商或其他有资格的人员联系。

```
NAME TOO LONG filename
```

#### 原因

发现一个超长的路径名，这通常表示文件系统名称空间中的循环。如果特权用户对目录创建了循环链接，则会出现此错误。

#### 操作

删除循环链接。

```
ROOT INODE UNALLOCATED (ALLOCATE)
```

#### 原因

根 *inode*（通常为 *inode* 编号 2）没有分配模式位。

#### 操作

要将 *inode* 2 分配为根 *inode*，请在 *ALLOCATE* 提示符下键入 *y*。通常在根 *inode* 中找到的文件和目录将在阶段 3 中恢复，并被放入 *lost+found* 目录中。如果尝试分配根 *inode* 时失败，fsck 将显示此消息并退出： *CANNOT ALLOCATE ROOT INODE*。要终止该程序，请键入 *n*。

```
ROOT INODE NOT DIRECTORY (REALLOCATE)
```

#### 原因

文件系统的根 *inode*（通常为 *inode* 编号 2）不是目录 *inode*。

#### 操作

要清除根 *inode* 的现有内容并对其重新分配，请在 *REALLOCATE* 提示符下键入 *y*。通常在根 *inode* 中找到的文件和目录将在阶段 3 中恢复，并被放入 *lost+found* 目录中。如果尝试分配根 *inode* 时失败，fsck 将显示此消息并退出： *CANNOT ALLOCATE ROOT INODE*。要使 fsck 提供带 *FIX* 的提示，请键入 *n*。

```
UNALLOCATED I=inode-number OWNER=UID MODE=file-mode SIZE=file-size
MTIME=modification-time type=filename(REMOVE)
```

#### 原因

目录或文件项 *filename* 指向未分配的 *inode inode-number*。此时将显示所有者 *UID*、模式 *file-mode*、大小 *file-size*、修改时间 *modification-time* 和文件名 *filename*。

#### 操作

要删除目录项 *filename*，请在 *REMOVE* 提示符下键入 *y*。要忽略该错误状态，请键入 *n*。

ZERO LENGTH DIRECTORY I=*inode-number* OWNER=*UID* MODE=*file-mode*  
 SIZE=*file-size* MTIME=*modification-time* DIR=*filename* (REMOVE)

#### 原因

目录项 *filename* 的大小 *file-size* 为零。此时将显示所有者 *UID*、模式 *file-mode*、大小 *file-size*、修改时间 *modification-time* 和目录名称 *filename*。

#### 操作

要删除目录项 *filename*，请在 REMOVE 提示符下键入 y。这将导致阶段 4 中出现 BAD/DUP 错误消息。要忽略该错误状态，请键入 n。

## 阶段 3：检查连通性消息

此阶段将检查在阶段 2 中检查的目录，并报告以下原因所导致的错误状态：

- 未引用的目录
- 缺失的或完整的 lost+found 目录

阶段 3 中可能出现以下消息（按字母顺序）：

BAD INODE *state-number* TO DESCEND

#### 原因

内部错误导致将不可能的状态 *state-number* 传递给从文件系统目录结构向下派生的例程。fsck 退出。

#### 操作

如果出现此情况，请与当地服务提供商或其他有资格的人员联系。

DIR I=*inode-number1* CONNECTED. PARENT WAS I=*inode-number2*

#### 原因

这是一条建议性消息，表明一个目录 inode *inode-number1* 已成功连接到 lost+found 目录。目录 inode *inode-number1* 的父级 inode *inode-number2* 已由 lost+found 目录的 inode 编号替换。

#### 操作

N/A

DIRECTORY *filename* LENGTH *file-size* NOT MULTIPLE OF *disk-block-size* (ADJUST)

#### Oracle Solaris 10：

DIRECTORY *filename* LENGTH *file-size* NOT MULTIPLE OF *block-number* (ADJUST)

#### 原因

找到一个目录 *filename*，其大小 *file-size* 不是目录块大小 B 的倍数。（如果未在阶段 2 调整此问题，还会在阶段 3 再次出现。）

**操作**

要将长度舍入为合适的磁盘块大小，请在 **ADJUST** 提示符下键入 **y**。整理时，**fsck** 将显示一条警告并调整目录。要忽略此错误状态，请键入 **n**。

**Oracle Solaris 10 :**

要将长度舍入为合适的块大小，请在 **ADJUST** 提示符下键入 **y**。整理时，**fsck** 将显示一条警告并调整目录。要忽略此错误状态，请键入 **n**。

**lost+found IS NOT A DIRECTORY (REALLOCATE)**

**原因**

用于 **lost+found** 的项不是目录。

**操作**

要分配一个目录 **inode** 并更改 **lost+found** 目录以引用该 **inode**，请在 **REALLOCATE** 提示符下键入 **y**。**lost+found** 目录引用的上一个 **inode** 未清除，它将作为未引用的 **inode** 回收或在此阶段的稍后部分调整其链接计数。无法创建 **lost+found** 目录时，将会显示以下消息：**SORRY. CANNOT CREATE lost+found DIRECTORY**，并放弃尝试链接到已丢失的 **inode**（这将在阶段 4 中生成 **UNREF** 错误消息）。要放弃尝试链接到已丢失的 **inode**（这将在阶段 4 中生成 **UNREF** 错误消息），请键入 **n**。

**NO lost+found DIRECTORY (CREATE)**

**原因**

文件系统的根目录中没有 **lost+found** 目录。整理时，**fsck** 将尝试创建一个 **lost+found** 目录。

**操作**

要在文件系统的根目录中创建一个 **lost+found** 目录，请在 **CREATE** 提示符下键入 **y**。这可能会生成消息 **NO SPACE LEFT IN / (EXPAND)**。如果无法创建 **lost+found** 目录，**fsck** 将显示以下消息：**SORRY. CANNOT CREATE lost+found DIRECTORY**，并放弃尝试链接到已丢失的 **inode**。这反过来会在稍后在阶段 4 中生成 **UNREF** 错误消息。要放弃尝试链接到已丢失的 **inode**，请键入 **n**。

**NO SPACE LEFT IN /lost+found (EXPAND)**

**原因**

由于没有可用空间，因此无法将另一项添加到文件系统根目录的 **lost+found** 目录中。整理时，**fsck** 将扩展 **lost+found** 目录。

**操作**

要扩展 **lost+found** 目录以便为新项留出空间，请在 **EXPAND** 提示符下键入 **y**。如果尝试扩展时失败，**fsck** 将显示：**SORRY. NO SPACE IN lost+found DIRECTORY**，并放弃将文件链接到 **lost+found** 目录的请求。此错误会在稍后在阶段 4 中生成 **UNREF** 错误消息。删除 **lost+found** 目录中的任何不必要项。整理生效时，此错误将终止 **fsck**。要放弃尝试链接到已丢失的 **inode**，请键入 **n**。

UNREF DIR I=*inode-number* OWNER=*UID* MODE=*file-mode* SIZE=*file-size*  
MTIME=*modification-time* (RECONNECT)

#### 原因

遍历文件系统时，目录 inode *inode-number* 未连接到目录项。此时将显示所有者 *UID*、模式 *file-mode*、大小 *file-size* 以及目录 inode *inode-number* 的修改时间 *modification-time*。整理时，如果目录大小不为零，fsck 将重新连接到非空目录 inode。否则，fsck 将清除该目录 inode。

#### 操作

要将目录 inode *inode-number* 重新连接到 `lost+found` 目录，请在 RECONNECT 提示符下键入 `y`。如果目录重新连接成功，则会显示 CONNECTED 消息。否则，将显示一条 `lost+found` 错误消息。要忽略此错误状态，请键入 `n`。此错误会导致阶段 4 中出现 UNREF 错误状态。

## 阶段 4：检查引用计数消息

此阶段将检查在阶段 2 和 3 中获得的链接计数信息。它将报告下列原因所导致的错误状态：

- 未引用的文件
- 缺失的或完整的 `lost+found` 目录
- 文件、目录、符号链接或特殊文件的链接计数不正确
- 未引用的文件、符号链接和目录
- 文件或目录中存在坏片段或重复片段

#### Oracle Solaris 10：

文件或目录中存在坏块或重复块

- 可用 inode 总计数不正确

整理文件系统时，此阶段的所有错误（`lost+found` 目录中空间不足的错误除外）都可以更正。

阶段 4 中可能出现以下消息（按字母顺序）：

BAD/DUP type I=*inode-number* OWNER=*UID* MODE=*file-mode* SIZE=*file-size*  
MTIME=*modification-time* (CLEAR)

#### 原因

阶段 1 或阶段 1B 发现与文件或目录 inode *inode-number* 相关的重复片段或坏片段。此时将显示所有者 *UID*、模式 *file-mode*、大小 *file-size* 以及 inode *inode-number* 的修改时间 *modification-time*。

#### Oracle Solaris 10：

阶段 1 或阶段 1B 发现与文件或目录 inode *inode-number* 相关的重复块或坏块。此时将显示所有者 *UID*、模式 *file-mode*、大小 *file-size* 以及 inode *inode-number* 的修改时间 *modification-time*。

#### 操作

要通过清空其内容来取消分配 inode *inode-number*，请在 CLEAR 提示符下键入 y。要忽略此错误状态，请键入 n。

(CLEAR)

#### 原因

上面的 UNREF 错误消息中提及的 inode 无法重新连接。如果由于缺少重新连接文件的空间而终止 fsck，则整理文件系统时不会显示此消息。

#### 操作

要通过清空其内容取消分配 inode，请在 CLEAR 提示符下键入 y。要忽略前面的错误状态，请键入 n。

```
LINK COUNT type I=inode-number OWNER=UID MODE=file-mode
SIZE=file-size
MTIME=modification-time COUNT link-count SHOULD BE
corrected-link-count (ADJUST)
```

#### 原因

目录或文件 inode *inode-number* 的链接计数为 *link-count*，但应该为 *corrected-link-count*。此时将显示所有者 *UID*、模式 *file-mode*、大小 *file-size* 以及 inode *inode-number* 的修改时间 *modification-time*。如果指定了 -op 选项，则除非引用数在增加，否则将调整链接计数。除非发生硬件故障，否则不会出现此情况。如果在整理期间引用数增加，fsck 会显示 LINK COUNT INCREASING 消息，然后退出。

#### 操作

要用 *corrected-link-count* 替换目录或文件 inode *inode-number* 的链接计数，请在 ADJUST 提示符下键入 y。要忽略此错误状态，请键入 n。

lost+found IS NOT A DIRECTORY (REALLOCATE)

#### 原因

用于 lost+found 的项不是目录。

#### 操作

要分配一个目录 inode 并更改 lost+found 目录以引用该 inode，请在 REALLOCATE 提示符下键入 y。lost+found 目录以前引用的 inode 尚未清除。它将作为未引用的 inode 回收或在此阶段的稍后部分调整其链接计数。无法创建 lost+found 目录时，将会显示以下消息：SORRY. CANNOT CREATE lost+found DIRECTORY，并放弃尝试链接到已丢失的 inode。此错误会在稍后在阶段 4 中生成 UNREF 错误消息。要放弃尝试链接到已丢失的 inode，请键入 n。

NO lost+found DIRECTORY (CREATE)

### 原因

文件系统的根目录中没有 `lost+found` 目录。整理时，`fsck` 将尝试创建一个 `lost+found` 目录。

### 操作

要在文件系统的根目录中创建一个 `lost+found` 目录，请在 `CREATE` 提示符下键入 `y`。如果无法创建 `lost+found` 目录，`fsck` 将显示以下消息：`SORRY. CANNOT CREATE lost+found DIRECTORY`，并放弃尝试链接到已丢失的 `inode`。此错误反过来会在稍后在阶段 4 中生成 `UNREF` 错误消息。要放弃尝试链接到已丢失的 `inode`，请键入 `n`。

`NO SPACE LEFT IN / lost+found (EXPAND)`

### 原因

没有空间可用于将另一项添加到文件系统根目录的 `lost+found` 目录中。整理时，`fsck` 将扩展 `lost+found` 目录。

### 操作

要扩展 `lost+found` 目录以便为新项留出空间，请在 `EXPAND` 提示符下键入 `y`。如果尝试扩展时失败，`fsck` 将显示以下消息：`SORRY. NO SPACE IN lost+found DIRECTORY`，并放弃将文件链接到 `lost+found` 目录的请求。此错误会在稍后在阶段 4 中生成 `UNREF` 错误消息。删除 `lost+found` 目录中的任何不必要项。当整理（`-op` 选项）生效时，此错误将终止 `fsck`。要放弃尝试链接到已丢失的 `inode`，请键入 `n`。

`UNREF FILE I=inode-number OWNER=UID MODE=file-mode SIZE=file-size  
MTIME=modification-time (RECONNECT)`

### 原因

遍历文件系统时，文件 `inode inode-number` 未连接到目录项。此时将显示所有者 `UID`、模式 `file-mode`、大小 `file-size` 以及 `inode inode-number` 的修改时间 `modification-time`。当 `fsck` 进行整理时，如果文件的大小或链接计数为零，则清除该文件；否则，将重新连接该文件。

### 操作

要将 `inode inode-number` 重新连接到文件系统的 `lost+found` 目录，请键入 `y`。如果将 `inode inode-number` 连接到 `lost+found` 目录时出现问题，此错误可能会在阶段 4 中生成 `lost+found` 错误消息。要忽略此错误状态，请键入 `n`。此错误始终会在阶段 4 中引发 `CLEAR` 错误状态。

`UNREF type I=inode-number OWNER=UID MODE=file-mode SIZE=file-size  
MTIME=modification-time (CLEAR)`

### 原因

遍历文件系统时，`inode inode-number`（其 `type` 是目录或文件）未连接到目录项。此时将显示所有者 `UID`、模式 `file-mode`、大小 `file-size` 以及 `inode inode-number` 的修改时间 `modification-time`。当 `fsck` 进行整理时，如果文件的大小或链接计数为零，则清除该文件；否则，将重新连接该文件。

**操作**

要通过清空其内容来取消分配 inode *inode-number*，请在 CLEAR 提示符下键入 y。要忽略此错误状态，请键入 n。

```
ZERO LENGTH DIRECTORY I=inode-number OWNER=UID MODE=file-mode
SIZE=file-size MTIME=modification-time(CLEAR)
```

**原因**

目录项 *filename* 的大小 *file-size* 为零。此时将显示所有者 *UID*、模式 *file-mode*、大小 *file-size*、修改时间 *modification-time* 和目录名称 *filename*。

**操作**

要通过清空其内容来取消分配目录 inode *inode-number*，请键入 y。要忽略该错误状态，请键入 n。

## 阶段 5：检查柱面组消息

本节包含当前 Oracle Solaris 发行版中的阶段 5 fsck 消息。

此阶段将检查可用片段和已用 inode 图。它将报告下列原因所导致的错误状态：

- 已用 inode 图中缺少已分配 inode
- 可用片段图中缺少可用片段
- 已用 inode 图中有可用 inode
- 可用片段总计数不正确
- 已用 inode 总计数不正确

阶段 5 中可能出现以下消息（按字母顺序）：

```
FRAG BITMAP WRONG (CORRECTED)
```

**原因**

柱面组片段图缺少一些可用片段。在整理期间，fsck 将重新构造这些图。

**操作**

要重新构造可用片段图，请在 SALVAGE 提示符下键入 y。要忽略此错误状态，请键入 n。

```
CG cg-number: BAD MAGIC NUMBER
```

**原因**

柱面组的魔数 *cg-number* 不正确。此错误通常表示柱面组图已被销毁。以交互方式运行时，柱面组标记为需要重新构造。如果正在整理文件系统，fsck 将终止。

**操作**

如果出现此情况，请与当地服务提供商或其他有资格的人员联系。

```
CORRECT GLOBAL SUMMARY (SALVAGE)
```

**原因**

摘要信息不正确。整理时，fsck 将重新计算摘要信息。

**操作**

要重新构造摘要信息，请在 SALVAGE 提示符下键入 y。要忽略此错误状态，请键入 n。

## 阶段 5：检查柱面组消息

本节包含 Solaris 10 3/05 初始发行版中的阶段 5 fsck 消息。

此阶段检查可用块和已用 inode 图。它将报告下列原因所导致的错误状态：

- 已用 inode 图中缺少已分配 inode
- 可用块图中缺少可用块
- 已用 inode 图中有可用 inode
- 可用块总计数不正确
- 已用 inode 总计数不正确

阶段 5 中可能出现以下消息（按字母顺序）：

BLK(S) MISSING IN BIT MAPS (SALVAGE)

**原因**

柱面组块图缺少一些可用块。在整理期间，fsck 将重新构造这些图。

**操作**

要重新构造可用块图，请在 SALVAGE 提示符下键入 y。要忽略此错误状态，请键入 n。

CG *character-for-command-option*: BAD MAGIC NUMBER

**原因**

柱面组的魔数 *character-for-command-option* 不正确。此错误通常表示柱面组图已被销毁。以交互方式运行时，柱面组标记为需要重新构造。如果正在整理文件系统，fsck 将终止。

**操作**

如果出现此情况，请与当地服务提供商或其他有资格的人员联系。

FREE BLK COUNT(S) WRONG IN SUPERBLK (SALVAGE)

**原因**

实际可用块计数与文件系统超级块中的可用块计数不匹配。如果指定了 -o p 选项，则会自动修复超级块中的可用块计数。

**操作**

要重新构造超级块可用块信息，请在 SALVAGE 提示符下键入 y。要忽略此错误状态，请键入 n。



## SUMMARY INFORMATION BAD (SALVAGE)

## 原因

摘要信息不正确。整理时，fsck 将重新计算摘要信息。

## 操作

要重新构造摘要信息，请在 SALVAGE 提示符下键入 y。要忽略此错误状态，请键入 n。

## fsck 摘要消息

本节包含当前 Oracle Solaris 发行版中的 fsck 摘要消息。如果运行的版本低于 Solaris 10 6/06 发行版，则会在清除阶段显示这些消息。有关更多信息，请参见第 265 页中的“清除阶段消息”。

检查文件系统后，会显示一些摘要消息。

```
number-of files, number-of-files
used, number-of-files free (number-of frags, number-of blocks,
percent fragmentation)
```

此消息表明，所检查的文件系统中包含 *number-of* 个使用 *number-of* 个片段大小块的文件，该文件系统中还有 *number-of* 个片段大小块可用。括号中的数字将可用计数细分为 *number-of* 个可用片段、*number-of* 个可用的完整大小块和 *percent* 分段。

```
***** FILE SYSTEM WAS MODIFIED *****
```

此消息表示，fsck 对文件系统进行了修改。如果看到此消息，则不需要重新运行 fsck。此消息只是有关 fsck 的更正操作的信息。

## 清除阶段消息

本节包含 Oracle Solaris 10 发行版中的 fsck 清除阶段消息。在此发行版中，可在 fsck 摘要阶段找到类似消息。有关更多信息，请参见第 265 页中的“fsck 摘要消息”。

检查文件系统后，将执行一些清除功能。清除阶段将显示以下状态消息。

```
number-of files, number-of-files
used, number-of-files free (number-of frags, number-of blocks,
percent fragmentation)
```

此消息表明，所检查的文件系统中包含 *number-of* 个使用 *number-of* 个片段大小块的文件，该文件系统中还有 *number-of* 个片段大小块可用。括号中的数字将可用计数细分为 *number-of* 个可用片段、*number-of* 个可用的完整大小块和 *percent* 分段。

```
***** FILE SYSTEM WAS MODIFIED *****
```

此消息表示，`fsck` 对文件系统进行了修改。如果文件系统已挂载或为当前的根 (/) 文件系统，请重新引导。如果已挂载文件系统，则可能需要将其卸载并再次运行 `fsck`；否则，内核中的表副本可能会撤消 `fsck` 已执行的操作。

*filename* FILE SYSTEM STATE SET TO OKAY

此消息表示文件系统 *filename* 已标记为稳定。使用 `fsck -m` 命令确定文件系统是否需要检查。

*filename* FILE SYSTEM STATE NOT SET TO OKAY

此消息表明，文件系统文件名未标记为稳定。使用 `fsck -m` 命令确定文件系统是否需要检查。

## 软件包问题故障排除（任务）

---

本章介绍在安装或删除软件包时可能遇到的问题。“特定软件包安装错误”一节介绍了可能遇到的软件包安装和管理错误。“一般软件包安装问题”一节介绍了可能不显示错误消息的行为问题。

以下是本章中的信息列表：

- 第 268 页中的“特定软件包安装错误”
- 第 268 页中的“一般软件包安装问题”

有关管理软件包的信息，请参见《系统管理指南：基本管理》中的第 20 章“管理软件（概述）”。

## 软件包符号链接问题故障排除

在以前的 Solaris 发行版中，创建软件包时无法在 `pkgmap` 文件中指定符号链接目标。这意味着在使用 `pkgadd` 命令添加软件包时，与软件包或修补程序相关的符号链接始终指向符号链接的源，而不是符号链接的目标。在升级更改符号链接目标所需的软件包或修补程序软件包时，这将产生问题。

现在，缺省行为是，如果软件包需要将符号链接的目标更改为其他内容，则 `pkgadd` 命令将检查符号链接的目标，而不检查符号链接的源。

不过，这意味着部分软件包可能符合或不符合新的 `pkgadd` 行为。

`PKG_NONABI_SYMLINKS` 环境变量将有助于在新旧 `pkgadd` 符号链接行为之间进行转换。如果此环境变量设置为 `true`，`pkgadd` 将指向符号链接的源。

如果在使用 `pkgadd` 命令添加软件包之前管理员已设置了软件包的行为，则设置此变量可使不一致的软件包恢复至原有行为。

新的 `pkgadd` 符号链接行为可能会导致使用 `pkgadd` 命令添加现有软件包失败。在此情况下，可能会看到以下错误消息：

unable to create symbolic link to <path>

如果软件包因为此问题而未安装，请执行以下操作：

- 1. 如果是 Oracle 提供的软件包，请致电 [My Oracle Support](#) 并报告不一致软件包的名称。
- 2. 设置 PKG\_NONABI\_SYMLINKS 环境变量并尝试使用 pkgadd 命令再次添加软件包。

```
PKG_NONABI_SYMLINKS=true
export PKG_NONABI_SYMLINKS
pkgadd pkg-name
```

## 特定软件包安装错误

WARNING: filename <not present on Read Only file system>

| 出现错误的原因                                                                                                      | 解决该问题的办法                                                                   |
|--------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------|
| 此错误消息指示并非该软件包的所有文件都可以安装。使用 pkgadd 在客户机上安装软件包时，通常会出现这种情况。在此情况下，pkgadd 会尝试在从服务器挂载的文件系统中安装软件包，但 pkgadd 无权执行此操作。 | 如果在软件包安装期间看到此警告消息，还必须在服务器上安装软件包。有关详细信息，请参见《系统管理指南：基本管理》中的第 20 章“管理软件（概述）”。 |

## 一般软件包安装问题

| 出现错误的原因                                                                                          | 解决该问题的办法                                       |
|--------------------------------------------------------------------------------------------------|------------------------------------------------|
| 添加或删除在 Solaris 2.5 发行版和兼容版本之前开发的一些软件包时，存在一个已知问题。有时候，添加或删除这些软件包时，在用户交互期间安装会失败，或者提示您进行用户交互并忽略您的响应。 | 设置以下环境变量并尝试再次添加软件包。<br><br>NONABI_SCRIPTS=TRUE |

# 索引

---

## A

- acct.h 格式文件, 137, 138
- acctcms 命令, 132, 141
- acctcom 命令, 137, 138
- acctcon 命令, 126, 132, 140
- acctdusg 命令, 120, 135, 139
- acctprc 命令, 132
- acctwtmp 命令, 119, 120, 134
- active.MMDD 文件, 127, 140
- active 文件, 127, 140
- at.deny 文件, 112, 115
  - 说明, 100
- at 命令, 111, 112, 115
  - l 选项 (列表), 114
  - m 选项 (邮件), 112, 113
  - 错误消息, 116
  - 概述, 100, 101, 111
  - 拒绝访问, 115–116
  - 控制访问, 112, 115
    - 概述, 100
  - 自动调度, 103
- at 作业文件, 111, 115
  - 创建, 112, 113
  - 删除, 115
  - 说明, 101
  - 提交, 112
  - 位置, 101
  - 显示, 114
- atjobs 目录, 103
  - 说明, 100

## C

- chargefee 脚本, 120, 135
  - 对用户计费, 125
- 重复性系统任务, 109
- 重新启动
  - runacct 脚本, 127, 128, 131, 133
- 重新引导
  - 崩溃后失败, 227–228
  - 和 /var/adm/wtmpx 文件, 120
- 重新启动
  - 进程, 153
- 重新引导
  - 连接记帐和, 119
  - 每日报告和, 134
- 重新引导基于 x86 的系统, 引导归档文件 SMF 服务失败, 196
- ckpacct 脚本, 121, 122, 124
- closewtmp 命令, 132
- cmsprev 文件, 141
- Command not found 错误消息, 237
- Common Agent Container
  - Oracle Solaris OS 中的故障排除, 195–196
  - 故障排除, 235–236
- Common Agent Container 共享, 共享组件, 235–236
- Common Agent Container 共享组件
  - 端口编号 (如何检查), 235–236
  - 问题类型
    - 超级用户口令的安全性, 235–236
- Common Agent container 共享组件
  - 问题类型
    - 端口号冲突, 235–236
- consadm 命令, 208

**consadm 命令 (续)**

- 禁用辅助控制台, 209
- 启用辅助控制台, 208
  - 在系统重新引导期间, 208-209
- 显示辅助控制台的列表(如何), 208

**core 文件**

- 查找并删除, 83
- 使用 `proc` 工具检查, 215

**core 文件, 自动删除, 112****coreadm 命令, 211**

- 管理核心文件, 211
- 设置核心文件名称模式, 214
- 显示核心转储配置, 214

**CPU (central processing unit, 中央处理器)**

- 使用率高的进程, 165
- 显示信息
  - 时间使用情况, 135, 151, 165

**cron.allow 文件, 108, 109, 110****cron.deny 文件, 108, 109**

- 缺省值, 108

**cron 守护进程, 101, 103****crontab 命令, 109**

- cron 守护进程和, 103
- e 选项(编辑), 104, 105
- l 选项(列表), 106
- r 选项(删除), 107, 108
- `/var/adm` 维护和, 202
- 错误消息, 110
- 调度, 103
- 概述, 100, 101
- 控制访问, 108, 109, 110
  - 概述, 100, 108, 109
  - 仅限特定用户访问, 108, 109, 110
  - 拒绝访问, 108, 109
- 每日任务, 100
- 使用的文件, 103
- 退出而不保存更改, 105
- 运行的记帐脚本, 122, 124

**crontab 文件**

- 编辑, 104, 105
- 创建, 104, 105
- 创建和编辑, 99-100
- 拒绝访问, 109
- 缺省值, 102

**crontab 文件 (续)**

- 删除, 107, 108
  - 说明, 103
  - 位置, 102
  - 显示, 106
  - 语法, 103, 104
- `ctacct.MMDD` 文件, 132, 140
- `ctmp` 文件, 140

**D****date 命令**

- 记帐数据和, 119, 120

**daytacct 文件**

- `runacct` 脚本和, 132, 141
- `/var/adm/acct/nite` 目录, 位于, 140
- 每日使用情况报告和, 135

**df 命令, 173**

- h 选项, 73
- k 选项(千字节), 173
- t 选项(总块数), 74
- 概述, 72, 173
- 示例, 73, 173

**disktacct.MMDD 文件, 132****disktacct 文件, 121****disktacct 文件, 120, 132, 140****dispadm 命令, 概述, 160****dmesg 命令, 202****dodisk 脚本, 119**

- 创建的文件, 120, 121, 132
- 概述, 119, 120, 121
- 文件创建, 140
- 运行的 `crontab` 项, 124
- 注意, 120

**dtmp 文件, 139****DTrace 功能, 196****du 命令, 78, 79****dumpadm, 管理系统故障转储信息, 220****E****edquota 命令**

- p 选项(样例), 90

**edquota 命令 (续)**

- t 选项 (时间限制), 95

- 概述, 86, 87, 94

- 设置用户配额, 90

- 为单个用户禁用配额, 97

eeeprom 命令, 用于在 ttymon 终端上设置波特率, 41

/etc/acct/holidays 文件, 124

/etc/cron.d/at.deny 文件, 112, 115

/etc/cron.d/cron.allow 文件, 108, 109, 110

/etc/cron.d/cron.deny 文件, 108, 109

/etc/init.d/acct 文件, 123

/etc/syslog.conf 文件, 204

/etc/utmpx 文件, 37

/etc/vfstab 文件, 88

**F**

fcntl 信息, 153, 155

fd2log 文件, 127, 131, 140

fee 文件, 120, 125, 132, 139

files, fstat 和 fcntl 信息显示, 153

find 命令

- core 文件, 83

- 查找超过大小限制的文件, 77

- 旧文件/非活动文件, 80, 81

fiscrptn 文件, 141

fsck 命令, 101

fstat 信息, 153, 155

**G**

getty, 22

**H**

holidays 文件, 124

hostid 命令, 58

**I**

iostat 命令

- 概述, 171

- 基本信息显示, 171

**K**

klwp 结构, 145

kmdb 实用程序, 229–230, 230–231

kthread 结构, 145

**L**

lastdate 文件, 131, 140

lastlogin 命令, 132

lineuse 文件, 请参见 /var/adm/acct/nite/lineuse 文件

localeadm 命令, 54

lock 文件, 127, 131

lock1 文件, 131

log.MMDD 文件, 140

loginlog 文件, 132, 141

ls 命令

- 检查目录大小, 75

- l 选项 (以字节表示的大小), 76

- s 选项 (以块表示的大小), 76

- t 选项 (最新文件), 80

LWP (lightweight process, 轻量级进程)

- 结构, 145

- 进程和, 145

- 显示信息, 153

- 已定义, 145

**M**

mdb 实用程序, 223, 224

messages.n 文件, 201

messages 文件, 197, 204

monacct?本, 使用的文件/生成的文件, 141

monacct 脚本

- runacct 脚本和, 121, 131

- 计划运行, 122

**monacct 脚本（续）**

- 每月命令摘要和, 136, 137
- 使用的文件/生成的文件, 141
- 运行的 crontab 项, 124

motd 文件, 67-68

motd 文件, 67

MOTD（message of the day, 每日消息）功能, 67-68

**N**

nice 命令, 164, 165

nice 数值, 151, 164

nlsadmin 命令, 39

**O**

Oracle Solaris 进程记帐和统计信息改进, 117

owtmpx 文件, 141

**P**

pacctn 文件

- 概述, 120, 132, 139
- 监视大小, 121, 131
- 显示, 137, 138

perf 文件, 191

pfiles 命令, 153, 155

pflags 命令, 153

pkill 命令, 153, 157

pldd 命令, 153

pmadm 命令

- 禁用 ttymon 服务, 48
- 列出 ttymon 服务, 46
- 启用 ttymon 服务, 48
- 说明, 36
- 添加 ttymon 服务, 45

pmap 命令, 153

prdaily 脚本

- runacct 脚本和, 131, 141
- 概述, 131
- 使用的文件, 140

**prdaily 脚本（续）**

- 文件使用, 141
- 线使用情况报告和, 141

priocntl 命令

- c 选项（调度类指定）, 163
- i 选项（ID 类型）, 162, 163
- l 选项（调度类显示）, 161
- m 选项（最高/最低优先级）, 162
- p 选项（优先级指定）, 162
- s 选项（优先级上限/更改优先级）, 162, 163
- 概述, 160

/proc 目录, 152

proc 工具, 检查核心转储文件, 215

proc 结构, 145, 151

PROCFS（process file system, 进程文件系统）, 152

prtconf 命令, 58, 63

- 显示系统的产品名称, 62

ps 命令, 151, 154

- c 选项（调度类）, 151, 165
- ecl 选项（全局优先级）, 161
- ef 选项（全部信息）, 154
- 报告的字段, 152
- 概述, 151

psig 命令, 153

pstack 命令, 153

ptacctn.MMDD 文件, 133

ptime 命令, 153

ptree 命令, 153, 155

pwait 命令, 153

pwdx 命令, 153, 155

**Q**

quot 命令, 79

quotacheck 命令, 86, 91

quotaon 命令, 87, 91

quotas 文件, 86, 88

**R**

reboots 文件, 132, 140

repquota 命令, 92, 93-94, 94

rm 命令, 81, 82



root 用户口令, 忘记

x86

基于 GRUB 的引导, 229-230

rppt.MMDD 文件, 121, 141

rpt.MMDD 文件, 132, 141

RS-232-C, 请参见 串行端口

runacct 脚本, 126, 131

重新启动, 127, 128, 131, 133

monacct 脚本和, 131

prdaily 脚本和, 131, 141

错误防护, 131

错误消息, 127

概述, 121

计划运行, 122

进程文件, 131

上次执行时间, 140

失败, 127

使用的文件/生成的文件, 139, 141

修复损坏的文件, 126, 132

用户费用计算和, 125, 135

运行的 crontab 项, 131

诊断文件, 131

状态, 132

## S

sal 命令, 191

sa2 命令, 191, 192

SAC, 请参见 服务访问控制器

sacadm 命令, 44

启动 ttymon 端口监视器, 43

说明, 35

添加 ttymon 端口监视器, 41

中止 ttymon 端口监视器, 43

sadc 命令, 191, 192

sadd 文件, 192

SAF, 请参见 服务访问工具

sar 命令, 176, 192

-A 选项 (总体性能), 191, 193

-a 选项 (文件访问), 176

-b 选项 (缓冲区), 177

-c 选项 (系统调用), 178

-e 选项 (结束时间), 193

-f 选项 (要从中提取数据的文件), 193

sar 命令 (续)

-i 选项 (时间间隔), 193

-m 选项 (进程间通信), 183

-p 选项 (页入/缺页), 184

-q 选项 (队列), 185

-r 选项 (未使用的内存), 186

-s 选项 (开始时间), 193

-u 选项 (CPU 使用情况), 187

-v 选项 (系统表), 188

-y 选项 (终端设备), 190

概述, 175, 192

列出的选项, 193

所有选项的说明, 193

shutacct 脚本, 120, 121

shutdown 命令, 121

Spacctn.MMDD 文件, 132, 139

startup 命令, acct, 120

statefile 文件, 127, 132, 140

svcadm enable system/sar:default 命令, 191

sys crontab, 192

syslog.conf 文件, 204

syslogd 守护进程, 201

## T

tacct.MMDD 文件, 127, 132, 141

tacct 文件, 127, 132, 141

tacctn 文件, 141

tacctprev 文件, 141

调制解调器, 30

拨出服务, 21

拨入服务, 21

不同的使用方式, 21

串行端口工具调制解调器模板, 27

串行端口工具概述, 26

调优, 每日命令摘要和, 136

调制解调器

双向服务, 21, 37

已定义, 21

/tmp/disktacct.MMDD 文件, 132

tmpwtmp 文件, 132, 140, 141

tty 线

故障线故障排除, 134

tty 线, 使用情况监视, 134

**tty 线**

使用情况监视, 119, 134, 141

ttynam 命令, 38–39

ttymon 端口监视器, 44

(图), 36

启动, 43

双向调制解调器服务和, 37

添加, 41

中止, 43

ttymon 服务

禁用, 48

列出, 46

启用, 48

添加, 45

turnacct switch 脚本, 120, 121

turnacct switch 脚本, 132

**U**

UFS 配额, 93–94, 95

检查, 92

命令, 92

软限制时间

更改, 95

为单个用户更改, 96

为单个用户禁用, 97

要求, 87

UFS 配额命令, 87

UFS 文件系统, 显示信息, 79

UNIX 系统 (故障转储信息), 219

user 结构, 145

/usr/adm/messages 文件, 197

/usr/bin/mdb 实用程序, 223

/usr/proc/bin 目录, 152, 153

utmp2wtmp 命令, 132

/var/adm/acct/nite/cms 文件, 140

/var/adm/acct/nite/ctacct.MMDD 文件, 132, 140

/var/adm/acct/nite/ctmp 文件, 140

/var/adm/acct/nite/daycms 文件, 132, 140, 141

/var/adm/acct/nite/daytacct 文件, 请参见 daytacct 文件

/var/adm/acct/nite 目录, 140

/var/adm/acct/nite/disktacct 文件, 121

/var/adm/acct/nite/disktacct 文件, 120, 121, 132, 140

/var/adm/acct/nite/disktacct.MMDD 文件, 132

/var/adm/acct/nite/fd2log 文件, 127, 131, 140

/var/adm/acct/nite/lastdate 文件, 131, 140

/var/adm/acct/nite/lineuse 文件, 132, 140, 141

/var/adm/acct/nite/lock 文件, 127, 131, 140

/var/adm/acct/nite/lock1 文件, 131

/var/adm/acct/nite/log 文件, 140

/var/adm/acct/nite/log.MMDD 文件, 140

/var/adm/acct/nite/owtmpx 文件, 141

/var/adm/acct/nite/reboots 文件, 132, 140

/var/adm/acct/nite/statefile 文件, 127, 132, 140

/var/adm/acct/nite/tmpwtmp 文件, 132, 140, 141

/var/adm/acct/nite/wtmp.MMDD 文件, 132, 140

/var/adm/acct/nite/wtmperror 文件, 140

/var/adm/acct/nite/wtmperror.MMDD 文件, 140

/var/adm/acct/sum/cms 文件, 132

/var/adm/acct/sum/cms 文件, 141

/var/adm/acct/sum/cmsprev 文件, 141

/var/adm/acct/sum/daycms 文件, 132, 141

/var/adm/acct/sum 目录, 121, 140, 141

/var/adm/acct/sum/loginlog 文件, 132, 141

/var/adm/acct/sum/rprt.MMDD 文件, 141

/var/adm/acct/sum/rprtMMDD 文件, 121

/var/adm/acct/sum/rpt.MMDD 文件, 132

/var/adm/acct/sum/tacct 文件, 132

/var/adm/acct/sum/tacct 文件, 127, 141

/var/adm/acct/sum/tacct.MMDD 文件, 132, 141

/var/adm/acct/sum/tacctMMDD 文件, 127

/var/adm/acct/sum/tacctprev 文件, 127, 141

/var/adm/dtmp 文件, 139

/var/adm/fee 文件, 120, 125, 132, 139

/var/adm/messages 文件, 197, 204

/var/adm/messages.*n* 文件, 201

/var/adm/sa/sadd 文件, 192

**V**

/var/adm/acct 目录, 140

/var/adm/acct/fiscal 目录, 140

/var/adm/acct/nite/active 文件, 127, 131, 140

/var/adm/acct/nite/active.MMDD 文件, 131, 140

/var/adm/acct/nite/cms 文件, 132

/var/adm/Spacctn.MMDD 文件, 132, 139  
 /var/adm 目录  
     控制大小, 81  
     说明, 139  
     原始记帐数据, 120  
 /var/spool/cron/atjobs 目录, 100, 101, 103  
 /var/spool/cron/crontabs 目录, 102, 103  
 /var/spool/cron/crontabs/root 文件, 102, 120  
 /var/spool/cron/crontabs/sys crontab, 192  
 vfstab 文件, 配额和, 88  
 vmstat 命令  
     报告中的字段, 168  
     概述, 168

## W

Watchdog reset! 消息, 201  
 wtmp.MMDD 文件, 132, 140  
 wtmperror.MMDD 文件, 140  
 wtmperror 文件, 140  
 wtmpfix 命令, 126, 132, 140  
 wtmpx 文件, 126  
     概述, 120, 126, 132  
     关闭和, 121  
     每日报告和, 134  
     修复损坏的, 126, 132

## Z

ZFS, 交换区域和转储卷要求, 219

## 安

安全性  
     at 命令, 112  
     crontab 命令, 109

## 保

保存故障转储信息, 220

## 崩

崩溃, 204, 232  
     重新引导失败, 之后, 227-228  
     保存故障转储信息, 219  
     保存其他系统信息, 202  
     检查故障转储, 223, 224  
     客户服务和, 197, 220  
     显示生成的系统信息, 201, 224  
     以下过程, 197, 232

## 编

### 编辑

crontab 文件, 104, 105

## 拨

拨出调制解调器服务, 21  
 拨入调制解调器服务, 21

## 波

### 波特率

如何使用 eeprom 命令设置, 41  
 如何在 ttymon 终端上设置, 40-41

## 查

### 查找

并删除旧文件/非活动文件  
     请参见删除  
 超过大小限制的文件, 77  
 大文件, 76

## 超

超级用户 (root) 口令, 忘记  
     SPARC, 229  
     x86, 229, 230

超级用户口令, 忘记

SPARC, 229

x86, 229, 230

超级用户口令的安全性

Common Agent Container 共享组件

故障排除, 235–236

## 程

程序

磁盘相关性, 176

强制退出运行, 232

中断, 232

## 初

初始化 UFS 配额, 87

初始化配额, 91

## 串

串行端口

适配器板, 21

已定义, 21

## 创

创建

at 作业, 112

at 作业, 113

crontab 文件, 104, 105

## 磁

磁盘记帐, 请参见记帐, 磁盘

磁盘空间

查找并删除旧文件/非活动文件, 80, 83

查找超过大小限制的文件, 77

查找大文件, 76

磁盘空间 (续)

显示信息

df 命令, 173

挂载点, 174

每个用户拥有的磁盘空间, 79

目录大小, 78, 79

文件大小, 75, 76, 78

磁盘空间的用户所有权, 79

磁盘块和文件限制, 区别, 86

磁盘驱动器

查找并删除旧文件/非活动文件, 105

显示信息

可用磁盘空间, 173

## 从

从完整的故障转储目录中恢复, 224

## 错

错误消息

at 命令, 116

crontab 命令, 110

runacct 脚本, 127

崩溃消息, 202

定制日志, 204

日志文件, 197, 201

优先级, 205

与崩溃相关的, 201

源, 204

指定存储位置, 201, 204

## 大

大文件, 76

大小

目录, 78, 79

文件, 75, 76, 78, 79

## 登

### 登录监视

- 登录次数, 135
- 上次登录, 132, 137, 141
- 时间使用情况, 119, 121, 135

## 地

地址空间映射, 153

## 调

### 调度

- 另请参见 `crontab` 命令、`at` 命令
- 重复性系统任务, 100, 102
- 一次性系统任务, 101, 111

### 调度类, 160

- 更改, 163
- 更改优先级, 162, 164
- 显示信息, 151, 161
- 优先级和, 160, 162
- 指定, 162

## 定

### 定制

- 系统消息日志, 204
- 系统消息日志（如何）, 205–206

## 动

动力循环, 232

## 端

### 端口, 31

- 初始化进程, 37
- 已定义, 21
- 状态（表）, 51

### 端口号（如何检查）

- Common Agent Container 共享组件
- cacao, 235–236

### 端口号冲突

- Common Agent container 共享组件
- 故障排除, 235–236

### 端口监视器

- `ttymon` 和 `listen`（已定义）, 22, 37–39
- 定义, 22
- 状态（表）, 51

## 断

断电恢复, 134

## 对

对失败的 SMF 引导归档文件服务进行故障排除

- x86
- GRUB failsafe 归档文件, 231–232

对用户计费, 125

另请参见 `chargefee` 脚本

## 费

费用, 用户, 120, 125

费用（用户）, 135

## 分

### 分时进程

- 更改调度参数, 162
- 优先级
- 范围, 160
- 概述, 160
- 更改, 162, 164

## 服

服务访问工具

概述, 23, 34

关联的程序 (表), 35

所控制的服务

状态 (表), 51

用于, 23, 34

服务访问控制器, 35, 36

## 辅

辅助 (远程) 控制台, 206

## 根

根 crontab 文件, 120

## 跟

跟踪标志, 153

## 更

更改

crontab 文件, 104

单个用户的 UFS 配额, 96

调度类, 163

每日消息, 67

日期, 67

软限制时间, 95

系统的主机名, 68

优先级, 162, 164

分时进程, 164

## 工

工具

进程, 153

系统性能监视, 146

用于显示进程信息, 152

## 共

共享内存, 进程虚拟内存, 146

## 故

故障排除

Common Agent Container, 195–196

Common Agent container 共享组件

问题类型, 235–236

tty 线, 134

进程, 165

软件包安装/删除, 267

故障排除任务, 参考信息, 197

故障消息, 201

故障转储目录, 从完整目录中恢复, 224

## 关

关闭

监视, 120, 121, 134

## 管

管理系统故障转储信息, 使用 dumpadm, 220

管理系统资源, 指南, 53

## 核

核心文件, 使用 coreadm 管理, 211

核心文件名称模式, 使用 coreadm 设置, 213

核心转储配置, 使用 coreadm 显示, 214

## 恢

恢复, 使用匹配的命令, 234

恢复超级用户口令

SPARC, 229

x86, 229, 230

## 活

活动文件, 131

## 基

基于 GRUB 的引导

SMF 引导归档文件服务失败故障排除, 196

系统崩溃

失败的 SMF 引导归档文件服务, 219

## 技

技术支持

发送故障转储信息, 197

故障转储分析, 220

## 记

记帐, 126, 128, 141

报告, 133

概述, 133

每日报告 (tty 线使用情况), 134

每日命令摘要, 136, 141

每日使用情况报告, 135

全部命令摘要 (每月), 137, 141

上次登录报告, 137

磁盘, 119, 120, 121

acctdusg 程序, 135

概述, 118

禁用, 129

进程, 119, 120, 135

连接, 119

runacct 状态和, 132

/var/adm/wtmpx, 134

每日, 120, 141

另请参见记帐, 报告

分步概要说明, 121

启动, 124

设置为自动运行 (如何), 123

停止, 128-129

维护, 128

文件, 139, 141

## 记帐 (续)

修复损坏的文件

tacct 文件, 127

wtmpx 文件, 126, 132

用户费用计算, 120

另请参见对用户计费

原始数据, 120

记帐

连接

/var/adm/acct/nite/directory 和, 140

## 检

检查核心转储文件, 使用 proc 工具, 215

## 交

交换卷大小, 对于具有 ZFS 根文件系统的系统, 219

## 禁

禁用

单个用户的配额, 97

辅助控制台, 使用 consadm 命令, 209

系统记帐, 129

禁用配额, 87

## 进

进程

重新启动, 153

nice 数值, 151, 164, 165

proc 工具命令, 152

打开文件的 fstat 和 fcntl 信息, 153, 155

当前工作目录, 153, 155

地址空间映射, 153

调度类, 160

更改, 163

更改优先级, 162, 164

显示信息, 151, 161

优先级和, 160, 162

## 进程, 调度类 (续)

- 指定, 162
- 跟踪标志, 153
- 工具命令, 153
- 故障排除, 165
- 记帐实用程序, 119, 120, 135
- 结构, 145, 151
- 控制, 156
- 库链接到, 153
- 失控, 165
- 使用 `proc` 工具命令显示信息, 153
- 使用 `proc` 工具显示信息, 152
- 树, 153, 155
- 显示信息, 151
  - `acctcom` 命令, 137, 138
  - `LWP`, 153
  - `priocntl` 命令, 161
  - `ps` 命令, 151, 154, 161
  - 列出进程, 154
  - 列出所执行的进程, 154
  - 每日使用情况报告, 135
  - 停用进程, 137
- 显示信息 (如何), 155–156
- 信号操作, 153
- 已打开文件的 `fstat` 和 `fcntl` 信息, 153
- 已定义, 145
- 应用程序线程和, 145, 146
- 优先级, 164
  - 调度类和, 160, 162
  - 概述, 160, 164
  - 更改, 162, 164
  - 更改分时进程优先级, 162, 164
  - 全局优先级, 160, 161
  - 显示信息, 151, 161
  - 用户模式优先级, 160
  - 指定, 162
- 暂时停止, 153
- 栈跟踪, 153
- 中止, 153, 157
- 术语, 145, 146
- 进程记帐, 119, 120, 135
  - 原因记录, 121
- 进程文件系统 (process file system, PROCFS), 152

## 警

- 警报信息优先级 (对于 `syslogd`), 205

## 客

- 客户服务, 发送故障转储信息, 197

## 控

## 控制

- 对 `at` 命令的访问, 100, 112, 115
- 对 `crontab` 命令的访问, 109, 110
  - 概述, 100
- 进程, 156
- 控制台
  - 辅助
    - 在系统重新引导期间启用, 208–209
- 控制台终端, 如何设置波特率, 40–41
- 控制台终端波特率, 使用 `eeprom` 命令设置, 41

## 口

- 口令安全性冲突, 超级用户, Common Agent Container, 235–236

## 连

- 连接记帐, 请参见记帐, 连接

## 列

## 列出

- 进程, 154
- 所执行的进程, 154
- 文件和目录, 75, 76, 80

## 临

- 临时目录, 80, 82



## 每

每进程核心文件路径,使用 `coreadm` 设置, 212  
 每日记帐, **请参见** 记帐, 每日  
 每日任务 (使用 `crontab` 调度), 100  
 每日消息 (message of the day, MOTD) 功能, 67-68  
 每月命令摘要, 137  
 每月任务 (使用 `crontab` 调度), 101  
 每周任务 (使用 `crontab` 调度), 101

## 命

命令, 监视使用情况, 140

## 目

### 目录

大小, 78, 79  
 进程的当前工作目录, 153  
 临时, 清除, 80, 82  
 显示信息, 75, 76, 78, 79

## 内

### 内存

#### 共享

进程虚拟内存, 146  
 进程结构和, 145  
 显示信息的示例, 63

#### 虚拟

进程, 146  
 用于显示信息的命令, 58

### 内核线程

调度和, 151  
 结构, 145, 151

## 配

### 配额

初始化, 91  
 多个用户的样例, 90  
 概述, 85

### 配额 (续)

更改, 94  
 检查超过的, 92-93  
 检查超过的用户配额, 93  
 启用, 86  
 启用, 示例, 91  
 启用和禁用, 87  
 删除, 94  
 设置, 86  
 设置软限制, 86  
 设置硬限制, 86  
 显示, 92-93  
 显示信息, 92  
 验证, 86, 92, 96  
 一致性检查, 91  
 用户

检查超过的, 93  
 设置, 90  
 为单个用户更改, 96

### 配额, UFS

初始化, 87  
 更改软限制缺省值, 95  
 使用, 85-86  
 在文件系统中检查, 94

## 启

### 启用

辅助控制台, 使用 `consadm` 命令, 208  
 辅助控制台, 在系统重新引导期间, 208-209

### 启用配额, 87

启用配额, 示例, 91

## 强

强制程序退出, 232

## 全

全部命令摘要, 137, 141  
 全局核心文件路径, 使用 `coreadm` 设置, 212

## 全局优先级

- 显示, 161
- 已定义, 160

## 缺

### 缺省值

- nice 数值, 164
- 每日消息, 67
- 配额, 95
- 软限制时间, 95

## 日

- 日志文件, 140
- 日志文件, 自动删除, 105

## 软

- 软件包, 安装故障排除, 267
- 软限制时间, 更改, 95

## 删

### 删除

- at 作业, 115
- core 文件, 83
- crontab 文件, 107, 108
- 查找并删除旧文件/非活动文件, 80
- 旧文件/非活动文件, 101
- 临时文件, 82
- 日志文件, 105

## 上

- 上次登录报告, 137

## 设

- 设置, 核心文件名称模式, 使用 `coreadm`, 214
- 设置终端和调制解调器, 任务列表, 25–26

## 失

- 失败, 基于 x86 的系统重新引导, SMF 引导归档文件服务, 196
- 失败的 SMF 引导归档文件服务, 基于 GRUB 的引导的故障排除, 219
- 失控进程, 165

## 实

- 实时进程, 更改类, 163

## 时

### 时间

- CPU 使用情况, 135, 151, 165
- 累积大量 CPU 时间的进程, 165

## 使

- 使用 SAF 管理串行端口, 任务列表, 34
- 使用 UFS 配额, 85–86

## 适

- 适配器板（串行端口）, 21

## 双

- 双向调制解调器服务, 21, 37

## 搜

- 搜索路径, 用于设置的文件, 238

## 停

### 停止

- 进程, 暂时, 153
- 系统记帐, 128-129

## 退

- 退出, 强制程序退出, 232

## 网

- 网络, 识别访问问题, 240

## 忘

### 忘记超级用户口令

- SPARC, 229
- x86, 229, 230

## 文

### 文件

- `fstat` 和 `fcntl` 信息显示, 153, 155
- 查找超过大小限制的文件, 77
- 大小, 75, 76, 78, 79
- 记帐, 139, 141
- 检查访问操作, 176
- 删除
  - 请参见删除
- 使用情况监视, 119, 120, 135
- 显示大小, 75-76
- 显示信息
  - 大小, 75, 76, 78, 79
  - 列出, 75, 76
- 修复损坏的
  - `wtmpx` 文件, 132
- 用于设置搜索路径, 238
- 文件或组的所有权, 解决文件访问问题, 239
- 文件系统
  - 磁盘空间使用情况, 173
  - 挂载点, 174

## 文件系统 (续)

- 恢复, 135
- 正在恢复, 125

## 系

### 系统崩溃故障排除

#### GRUB

- 引导归档文件服务在重新引导时失败, 219

系统的产品名称, 使用 `prtconf` 命令显示, 62

系统故障转储信息, 使用 `dumpadm` 管理, 220

### 系统活动

- 跟踪活动的列表, 146
- 手动收集数据, 192
- 自动收集数据, 191

系统记帐, 任务列表, 122

### 系统任务

另请参见 `crontab` 命令、`at` 命令

#### 调度

- 重复性任务, 100, 102
- 一次性任务, 101, 111
- 自动调度, 100

### 系统消息

- 定制日志 (如何), 205-206
- 指定存储位置, 201

系统消息日志 (定制), 204

### 系统资源

- 概述, 144
- 记帐
  - 概述, 118
- 监视, 112
  - UFS 配额, 94
  - 崩溃, 204, 232
  - 记帐, 128
  - 记帐系统, 141
  - 自动, 112

## 显

### 显示

- `acct.h` 格式文件, 137, 138
- `at` 作业, 114
- `crontab` 文件, 106

## 显示（续）

- LWP 信息, 153
- pacctn 文件, 137, 138
- 崩溃信息, 201
- 调度类信息, 151, 161
- 故障转储信息, 224
- 核心转储配置, 使用 `coreadm`, 214
- 进程信息（如何）, 155–156
- 链接的库, 153
- 链接库, 153
- 目录信息, 75, 76, 78
- 配额, 92–93
- 配额信息, 86, 92, 93
- 日期和时间, 63
- 文件大小, 75–76
- 文件系统信息, 79
- 文件信息
  - 列出最新, 80
  - 使用 `du` 命令, 78
  - 文件大小, 75, 76
- 系统的已安装内存, 63
- 系统活动信息, 175, 192
- 系统信息
  - 命令, 58, 63
- 引导消息, 202
- 优先级信息, 151, 161
- 主机 ID, 62

显示产品名称信息, `prtconf` 命令, 62

显示系统的物理处理器类型, `psrinfo -p`, 63–64

## 线

线路规程, 36

线使用情况

- `/var/adm/acct/nite/lineuse` 文件, 142
- 连接记帐和, 119
- 每日报告和, 134

线使用情况监视, 134

## 新

新增功能

- CPU 性能计数器, 143–144

## 新增功能（续）

- `svcadm enable system/sar:default` 命令, 191
- 增强的 `pfiles` 工具, 143

## 性

### 性能

- 报告, 175
- 跟踪的活动, 146
- 监视工具, 146
- 进程管理, 145, 153, 164
- 手动收集活动数据, 176, 192
- 文件访问, 176
- 系统活动监视, 146, 176, 191
- 自动收集活动数据, 191

## 修

修复, 126

- 损坏的 `tacct` 文件, 127
- 损坏的 `wtmpx` 文件, 126

## 验

### 验证

- UFS 配额, 92
- 配额, 96

## 要

要求, UFS 配额, 87

## 引

### 引导

- 显示生成的消息, 202
- 运行 `sadc` 命令, 191

引导归档文件, SMF 服务在重新引导时失败, 196

引导归档文件服务失败

x86

GRUB 故障排除, 231-232

引导归档文件服务失败时应执行的操作

x86

引导 failsafe 归档文件, 231-232

## 应

应用程序线程, 145, 146

## 用

用户登录

登录次数, 135

上次登录监视, 132, 137, 141

时间监视, 119, 132, 135

用户费用, 120, 135

**另请参见**对用户计费

用户进程

CPU 使用情况, 135

更改优先级, 164

优先级, 160

用户模式优先级, 160

用户配额, 92-93

设置, 90

为单个用户更改, 96

为单个用户禁用, 97

## 优

优先级 (进程)

调度类和, 162

概述, 160, 164

更改, 162, 164

分时进程, 162, 164

全局

显示, 161

已定义, 160

显示信息, 151, 161

用户模式优先级, 160

指定, 162

## 原

原因记录, 进程记帐, 121

## 在

在 ttymon 控制台终端上设置波特率, 如何, 40-41

## 正

正在打印, 用户费用计算, 125

正在记帐

**另请参见**对用户计费

类型, 125

正在远程打印, 用户费用计算, 125

## 识

识别网络访问问题, 240

识别芯片多线程功能的 `psrinfo` 命令选项, `psrinfo`

-p, 54

## 中

中断程序, 232

中止进程, 153, 157

## 终

终端, 29

串行端口工具概述, 26

串行端口工具中各项的说明, 26

故障线故障排除, 134

进程控制, 151

类型之间的区别, 21

线使用情况

`/var/adm/acct/nite/lineuse` 文件, 142

连接记帐和, 119

每日报告和, 134

已定义, 21

字母数字, 21

## 主

主机名,更改, 68

## 转

转储卷大小,对于具有 ZFS 根文件系统的系统, 219

## 状

状态, (runacct 脚本), 132

## 字

字母数字终端,请参见终端

## 自

自动报告系统活动, 191, 192

自动启用配额, 86

自动收集系统活动数据, 191

自动执行例行任务(概述), 100

自动执行系统任务, 100

    重复性任务, 109, 110

    单个任务, 111, 112, 115

## 最

最大值

    nice 数值, 164

    查找超过最大大小的文件, 77

最小值, nice 数值, 164