

AIX 5L™ V5.3



# 命令参考大全，卷 5，s - u



AIX 5L™ V5.3



# 命令参考大全，卷 5，s - u

**注意**

使用本资料及其支持的产品前，请阅读第 621 页的『声明』中的信息。

**第四版（2006 年 7 月）**

本版本适用于 AIX 5L V5.3 及其所有后续发行版，直到在新版本中另有声明为止。

在本出版物的后面提供了读者意见表。如果该表已被删除，请将意见寄往 IBM 中国公司上海分公司，汉化部；中国上海市淮海中路 333 号瑞安广场 10 楼；邮政编码：200021。要通过电子形式发送意见，请使用以下商业互联网地址：[ctscrcf@cn.ibm.com](mailto:ctscrcf@cn.ibm.com)。我们可以使用您提供的任何信息，而无需对您承担任何责任。

**© Copyright International Business Machines Corporation 1997, 2006. All rights reserved.**

---

# 目录

关于本书 . . . . .	ix
如何使用本书 . . . . .	ix
ISO 9000 . . . . .	xi
Single UNIX Specification 的 32 位和 64 位支持 . . . . .	xi
相关信息 . . . . .	xii
<b>按字母排列的命令列表 . . . . .</b>	<b>1</b>
sa 命令 . . . . .	1
sa1 命令 . . . . .	3
sa2 命令 . . . . .	3
sact 命令 . . . . .	4
sadc 命令 . . . . .	5
sar 命令 . . . . .	6
savebase 命令 . . . . .	13
savecore 命令 . . . . .	14
savevg 命令 . . . . .	15
scan 命令 . . . . .	18
sccs 命令 . . . . .	19
sccsdiff 命令 . . . . .	23
scscshelp 命令 . . . . .	24
schedo 命令 . . . . .	25
scls 命令 . . . . .	39
script 命令 . . . . .	40
sctpctrl 命令 . . . . .	40
sdiff 命令 . . . . .	42
secldapclntd 守护程序 . . . . .	45
secldifconv 命令 . . . . .	46
sectoldif 命令 . . . . .	48
securetcpip 命令 . . . . .	50
sed 命令 . . . . .	51
sedmgr 命令 . . . . .	55
send 命令 . . . . .	59
sendbug 命令 . . . . .	61
sendmail 命令 . . . . .	61
setclock 命令 . . . . .	67
setea 命令 . . . . .	68
setgroups 命令 . . . . .	69
setmaps 命令 . . . . .	70
setsenv 命令 . . . . .	73
settime 命令 . . . . .	75
setuname 命令 . . . . .	76
sh 命令 . . . . .	77
shconf 命令 . . . . .	77
shell 命令 . . . . .	78
show 命令 . . . . .	80
showmount 命令 . . . . .	82
shutacct 命令 . . . . .	83
shutdown 命令 . . . . .	84
size 命令 . . . . .	86

skulker 命令	87
slattach 命令	87
sleep 命令	89
slibclean 命令	90
sliplogin 命令	90
slocal 命令	93
smcaprop 命令	94
smdefca 命令	95
smdemon.cleanu 命令	96
smexpacert 命令	97
smgenkeycr 命令	97
smgenprivkr 命令	98
smimpcacert 命令	99
smimpservercert 命令	100
sminstkey 命令	101
smit 命令	101
smitty 命令	104
smlistcerts 命令	106
smrsh 命令	106
smserverprop 命令	108
smsigncert 命令	108
smtctl 命令	109
smundefca 命令	111
snap 命令	112
snapcore 命令	116
snapshot 命令	117
snapsplit 命令	119
snmpevent 命令	120
snmpd 守护程序	123
snmpdv1 守护程序	123
snmpdv3 守护程序	127
snmpinfo 命令	129
snmpmibd 守护程序	132
snmptrap 命令	134
snmpv3_ssw 命令	135
sno 命令	136
sodebug 命令	137
soelim 命令	138
sort 命令	139
sortbib 命令	145
sortm 命令	146
spell 命令	148
spellin 命令	150
spellout 命令	150
split 命令	151
splat 命令	152
splitvg 命令	160
splitvcopy 命令	161
splp 命令	163
spost 命令	165
spray 命令	166
sprayd 守护程序	167

srcmstr 守护程序	168
startcondresp 命令	170
start-secldapclntd 命令	172
stop-secldapclntd 命令	174
startprdomain 命令	174
startprnode 命令	177
startsrc 命令	179
startup 命令	181
startvsd 命令	181
startx 命令	183
statd 守护程序	185
statvsd 命令	186
stopcondresp 命令	188
stopprdomain 命令	190
stopprnode 命令	192
stopsrc 命令	194
stopvsd 命令	196
stpinet 方法	197
strace 命令	198
strchg 命令	199
strclean 命令	201
strconf 命令	201
strerr 守护程序	202
strinfo 命令	204
strings 命令	205
strip 命令	206
stripnm 命令	208
strload 命令	210
strreset 命令	213
strtune 命令	214
struct 命令	216
stinet 方法	217
stty 命令	218
stty-cxma 命令	225
style 命令	227
su 命令	228
subj 命令	230
sum 命令	231
suma 命令	232
survd 守护程序	242
suspendvsd 命令	243
svmon 命令	244
swap 命令	270
swapoff 命令	271
swapon 命令	272
swcons 命令	274
swts 命令	275
sync 命令	276
synclvodm 命令	276
syncvg 命令	277
syscall 命令	278
sysck 命令	280

syscorepath 命令	282
sysdumpdev 命令	284
sysdumpstart 命令	288
sysline 命令	289
syslogd 守护程序	290
tab 命令	293
tabs 命令	294
tail 命令	298
talk 命令	300
talkd 守护程序	301
tapechk 命令	303
tar 命令	304
tbl 命令	309
tc 命令	312
tcbck 命令	313
tcopy 命令	318
tcpdump 命令	318
tctl 命令	328
tee 命令	331
telinit 或 init 命令	332
telnet、tn 或 tn3270 命令	335
telnetd 守护程序	345
termdef 命令	347
test 命令	348
ftfp 或 utftp 命令	350
ftfpd 守护程序	355
tic 命令	358
time 命令	359
timed 守护程序	360
timedc 命令	362
timex 命令	364
tip 命令	365
tokstat 命令	370
topas 命令	374
topasout 命令	385
topsvcs 命令	386
topsvcsctrl 命令	388
touch 命令	391
tprof 命令	393
tput 命令	402
tr 命令	404
跟踪守护程序	407
traceroute 命令	413
tracesoff 命令	415
traceson 命令	416
trbsd 命令	417
trectl 命令	419
tredead 命令	420
trcevgrp 命令	422
trcnm 命令	423
trcrpt 命令	424
trestop 命令	430



trcupdate 命令	431
troff 命令	432
trpt 命令	485
true 或 false 命令	490
truss 命令	491
tset 命令	494
tsh 命令	496
tsm 命令	498
tsort 命令	499
ttt 命令	500
tty 命令	501
tunchange 命令	502
tuncheck 命令	504
tundefault 命令	505
tunrestore 命令	506
tunsave 命令	508
turnacct 命令	509
turnoff 命令	510
turnon 命令	510
tvi 命令	511
twconvdict 命令	514
twconvfont 命令	515
type 命令	516
ucgif 方法	517
ucfginet 方法	517
ucfgqos 方法	518
ucfgvsd 命令	518
uconvdef 命令	520
udefif 方法	521
udefinet 方法	522
udfcheck 命令	522
udfcreate 命令	523
udflabel 命令	524
uil 命令	524
uimx 命令	525
ul 命令	527
ulimit 命令	527
umask 命令	529
umcode_latest 命令	530
umount 或 unmount 命令	532
umountall 命令	533
unalias 命令	534
uname 命令	535
uncompress 命令	538
undefvsd 命令	539
unexpand 命令	540
unfencevsd 命令	541
unget 命令 (SCCS)	542
unifdef 命令	543
uniq 命令	545
units 命令	547
unlink 命令	549

unloadipsec 命令	550
unmirrorvg 命令	551
unpack 命令	552
untab 命令	554
update 命令	554
updatevsnode 命令	555
updatevsdtab 命令	557
updatevsdvg 命令	558
uprintfd 守护程序	560
uptime 命令	560
useradd 命令	561
userdel 命令	563
usermod 命令	564
users 命令	566
usrck 命令	567
utmpd 守护程序	572
uuccheck 命令	573
uucico 守护程序	575
uuclean 命令	576
uucleanup 命令	578
uucp 命令	580
uucpadm 命令	583
uucpd 守护程序	586
uudecode 命令	586
uudemon.admin 命令	587
uudemon.cleanu 命令	588
uudemon.hour 命令	590
uudemon.poll 命令	591
uuencode 命令	593
uuid_gen 命令 (NCS)	594
uukick 命令	595
uulog 命令	596
uuname 命令	598
uupick 命令	599
uupoll 命令	601
uuq 命令	602
uusched 守护程序	604
uusend 命令	605
uusnap 命令	606
uustat 命令	607
uuto 命令	610
uutry 命令	612
Uutry 命令	613
uux 命令	615
uuxqt 守护程序	618
<b>附录. 声明</b>	<b>621</b>
商标	622
<b>索引</b>	<b>623</b>

---

## 关于本书

本书为最终用户提供了关于 AIX® 操作系统的命令的完整且详细的信息。这些命令以字母顺序按类别列出，并且提供了有关命令及其可用标志的完整描述。每个列出的命令都适当地包含一些示例。本卷包含从字母 s 到 u 开头的 AIX 命令。操作系统附带的文档 CD 中也提供了本出版物。

---

## 如何使用本书

命令是执行操作或运行程序的请求。可以使用命令告诉操作系统所要执行的任务。当输入命令后，它们由命令解释器（也称为 shell）进行解密，然后处理该任务。

有些命令只需输入一个字。也可能将命令组合起来，使一个命令的输出成为另一个命令的输入。这就称为流水线技术。

标志进一步定义命令的操作。标志是修饰符，在命令行中与命令名称一起使用，通常由破折号开头。

命令也可以分组在一起并存储在文件中。这些就称为 shell 过程或 shell 脚本。执行包含命令的文件，而不是单独执行命令。

可以使用基于 Web 的系统管理器应用程序或“系统管理界面工具”（SMIT）来构造一些命令。

## 突出显示

本书使用以下突出显示约定：

<b>粗体</b>	标识命令、子例程、关键字、文件、结构、目录和由系统预定义名称的其他项。也标识图形对象，例如用户选择的按钮、标签以及图标。
<i>斜体</i>	标识由用户供给实际名称或值的参数。
等宽	标识特定数据值示例、与您可能看到的显示内容相似的文本示例、与您可能作为程序员编写的内容相似的部分程序代码示例、来自系统的消息或应实际输入的信息。

## 格式

每个命令可能包含以下的任一部分：

用途	每个命令主要功能的描述。
语法	显示命令行选项的语法语句。
描述	命令的讨论详细描述其功能和使用。
标志	命令行标志和相关变量列表，并附有标志如何修改命令操作的解释。
参数	命令行参数及其描述的列表。
子命令	说明子命令（交互式命令）用法的子命令列表。
退出状态	命令返回的退出值的描述。
安全性	指定运行命令所需的任何许可权。
示例	关于如何可以使用命令的特定示例。
文件	由命令使用的文件的列表。
相关信息	本书中相关命令和其他书中相关讨论的列表。

## 读取语法语句

语法语句是一种表示命令语法的方法，且由例如括号（[ ]）、花括号（{ }）、竖线（|）的符号组成。以下是 **unget** 命令语法语句的示例：

**unget** [ *-rSID* ] [ *-s* ] [ *-n* ] *File* ...

命令语法语句使用以下约定:

- 在命令行中必须以**粗体**按原文输入项。这些项包含命令名称、标志和文字符号。
- 必须用名称替换的表示变量的项以*斜体*表示。这些项包含紧随标志之后的参数以及该命令读取的参数, 例如*文件*和*目录*。
- 括号中所附的参数是可选的参数。
- 花括号中所附的参数是必需的参数。
- 不附在括号或花括号中的参数是必需的参数。
- 竖杠表示仅选择了一个参数。例如: [ *a | b* ] 表示可以选择 *a*、*b* 或不选。同样, { *a | b* } 表示必须选择 *a* 或 *b* 中的一个。
- 省略号 (...) 表示可以在命令行里重复使用该参数。
- 破折号 (-) 表示标准输入。

## 可安装软件包的列表

要列出个别命令的可安装软件包(文件集), 请使用带有 **-w** 标志的 **lsipp** 命令。例如, 要列出拥有 **installp** 命令的文件集, 请输入:

```
lsipp -w /usr/sbin/installp
```

输出类似以下显示:

File	Fileset	Type
/usr/sbin/installp	bos.rte.install	File

要列出拥有包含 **installp** 的所有文件名的文件集, 请输入:

```
lsipp -w "*installp*"
```

输出类似以下显示:

File	Fileset	Type
/usr/sbin/installp	bos.rte.install	File
/usr/clvm/sbin/linstallpv	prpq.clvm	File
/usr/lpp/bos.sysmgmt/nim/methods/c_installp	bos.sysmgmt.nim.client	File

## 在后台运行命令

如果要运行一个需要花费很长时间进行处理的命令, 可以指定命令在后台运行。后台进程是运行慢处理程序的一个有用的方法。要在后台运行命令, 请在命令结束处使用 **&** 运算符:

*Command*&

一旦进程在后台运行, 就可以继续在系统上工作并输入其他命令。

有时, 可能想在指定时间或特定日期运行命令。使用 **cron** 守护程序, 可以调度命令自动运行。或者, 使用 **at** 和 **batch** 命令, 可以在以后或在系统装入级别允许时运行命令。

## 输入命令

一般在命令行中的 **shell** 提示符后输入命令。**shell** 提示符可以改变。在以下示例中, **\$** 是提示符。

要显示当前目录的内容列表, 应输入 **ls** 并按 **Enter** 键:

```
$ ls
```

当输入并运行命令时，操作系统不显示 shell 提示符。命令完成其操作时，系统再次显示提示符。这表示可以输入另一个命令。

输入命令的一般格式是：

*Command Flag(s) Parameter*

标志改变命令的工作方法。很多命令拥有几个标志。例如，如果在 **ls** 命令后输入 **-l**（长）标志，系统将提供有关当前目录内容的附加信息。以下示例显示如何将 **ls** 命令与 **-l** 标志一起使用：

```
$ ls -l
```

参数由命令或标志后的字符串构成。它指定数据，例如文件或目录的名称或值。在以下示例中，名为 **/usr/bin** 的目录是一个参数：

```
$ ls -l /usr/bin
```

输入命令时，记住以下内容很重要：

- 通常用小写输入命令。
- 通常标志的前缀是 -（减号）。
- 如果用 ;（分号）分隔命令，可以在命令行中输入多个命令。
- 命令的长序列可以使用 \（反斜杠）继续到下一行。反斜杠位于第一行的结束处。以下示例显示反斜杠的位置：

```
$ cat /usr/ust/mydir/mydata > \  
/usr/usts/yourdir/yourdata
```

输入某些命令时，shell 提示符将更改。因为一些命令实际上是程序（例如 **telnet** 命令），在该命令中操作时，提示符会更改。在程序中发出的任何命令称为子命令。当您退出程序时，提示符返回为 shell 提示符。

操作系统可以与不同的 shell（例如 Bourne、C 或 Korn）一起操作，并且由 shell 解释所输入的命令。因此，必须了解正在使用的是什么 shell，才能用正确的格式输入命令。

## 停止命令

如果输入命令后决定停止运行该命令，可以中止命令的进一步处理。要停止处理命令，按下“中断”按键顺序（通常是 Ctrl-C 或 Alt-Pause）。进程停止时，返回 shell 提示符，然后可以输入另一个命令。

---

## ISO 9000

在此产品的开发和制造中使用了 ISO 9000 注册质量体系。

---

## Single UNIX Specification 的 32 位和 64 位支持

从版本 5.2 开始，此操作系统被设计用来支持 The Open Group 的 Single UNIX Specification V3（UNIX 03），提供了对基于 UNIX 操作系统的可移植性。添加了许多新接口或增强了某些当前接口以满足此规范，使版本 5.2 对应用程序具有更强的开放性和可移植性，同时保留了与先前 AIX 发行版的兼容性。

要确定开发可移植到 UNIX 03 的应用程序的正确方法，可能需要参考 The Open Group 的 UNIX 03 规范，可以通过在线访问或从 <http://www.unix.org/> 下载此规范。

---

## 相关信息

以下书籍包含与命令相关或有关的信息:

- 《AIX 5L V5.3 命令参考大全, 卷 1》
- 《AIX 5L V5.3 命令参考大全, 卷 2》
- 《AIX 5L V5.3 命令参考大全, 卷 3》
- 《AIX 5L V5.3 命令参考大全, 卷 4》
- 《AIX 5L V5.3 命令参考大全, 卷 5》
- 《AIX 5L V5.3 命令参考大全, 卷 6》
- *AIX 5L Version 5.3 Files Reference*
- 《打印机和打印指南》
- 《安装与迁移》
- 《AIX 5L V5.3 分区环境中的 AIX 安装》
- *AIX 5L Version 5.3 Network Information Services (NIS and NIS+) Guide*
- 《性能管理》
- *AIX 5L Version 5.3 Performance Tools Guide and Reference*
- 《安全性》
- 《网络与通信管理》
- 《操作系统与设备管理》
- *AIX 5L Version 5.3 Technical Reference: Base Operating System and Extensions Volume 1*
- *AIX 5L Version 5.3 Technical Reference: Base Operating System and Extensions Volume 2*
- *AIX 5L Version 5.3 Technical Reference: Communications Volume 1*
- *AIX 5L Version 5.3 Technical Reference: Communications Volume 2*
- *AIX 5L Version 5.3 Technical Reference: Kernel and Subsystems Volume 1*
- *AIX 5L Version 5.3 Technical Reference: Kernel and Subsystems Volume 2*
- 《AIX 5L V5.3 基于 web 的系统管理器管理指南》
- *Performance Toolbox Version 2 and 3 for AIX: Guide and Reference*

---

# 按字母排列的命令列表

---

## sa 命令

### 用途

总结记帐记录。

### 语法

```
/usr/sbin/sa [ -a ] [ -b ] [ -c ] [ -C ] [ -d ] [ -D ] [ -i ] [ -j ] [ -k ] [ -K ] [ -l ] [ -m ] [ -n ] [ -r ] [ -s ] [ -t ] [ -u ] [ -vNumber [ -f ] ] [ -SSaveFile ] [ -UUserFile ] [ File ... ]
```

### 描述

**sa** 命令对收集原始记帐数据的文件中的信息进行总结，该文件是 **/var/adm/pacct** 文件，或是由 *File* 参数指定的文件，然后将使用情况摘要报告写入 **/var/adm/savacct** 文件。接着，**sa** 命令删除 **/var/adm/pacct** 文件中的数据，以便收集新的记帐信息。下次执行 **sa** 命令时，它读取使用情况摘要和新的数据，并将所有信息都合并到报告中。

与 **sa** 命令一起使用的标志改变报告信息的类型。报告可以包含以下字段：

avio	表示每次执行的平均 I/O 操作数。
cpu	表示用户和系统时间（以分钟计）之和。
k	表示每次执行的 CPU 时间的平均千字节块数。
k*sec	表示 CPU 存储量整数，单位为千核心秒。
re	表示实时的分钟数。
s	表示系统 CPU 时间的分钟数。
tio	表示总的 I/O 操作数。
u	表示用户 CPU 时间的分钟数。

如果不指定任何标志而运行 **sa** 命令，摘要报告将包含每个命令的调用次数，以及 re、cpu、avio 和 k 字段。

**注：****-b**、**-d**、**-D**、**-k**、**-K** 和 **-n** 标志确定如何进行输出分类。如果在命令行中指定了不止一个这些标志，则只有最后指定的那个标志生效。

在此基本操作系统发行版下创建的摘要文件以支持大型用户标识（8 个字符或更长）的格式保存。在前发行版下创建的摘要文件可能为旧格式，它只支持最多 7 个字符的用户标识。**sa** 命令可识别和支持摘要文件的以上两种格式。如果需要将旧格式的摘要文件转换为新格式，可使用 **-C** 标志代替 **-s** 标志。只需进行一次转换。转换后，既可使用 **-s** 标志，也可使用 **-C** 标志。

### 标志

<b>-a</b>	显示所有命令的名称（包括那些带有不可显示字符的命令名）。将用过一次的命令放在 <b>other</b> 类别中。
<b>-b</b>	将输出按用户和系统时间的总和除以调用次数来进行分类。否则，输出为用户时间和系统时间之和。
<b>-c</b>	将每个命令使用时间显示为相对于所有命令使用时间的百分比。此外，还有用户、系统和实时。
<b>-C</b>	将记帐文件合并到摘要文件中。如果摘要文件为旧格式，则将它转换为新格式。
<b>-d</b>	将输出按平均磁盘 I/O 操作数进行分类。

<b>-D</b>	将输出按总的磁盘 I/O 操作数进行分类和显示。
<b>-f</b>	不要强制进行交互式阈值压缩。此标志必须与 <b>-v</b> 标志一起使用。
<b>-i</b>	仅读取原始数据，不读取摘要文件。
<b>-j</b>	显示每个调用的秒数，而不是每个类别的总的分钟数。
<b>-k</b>	将输出按平均 CPU 时间进行分类。
<b>-K</b>	将输出按 CPU 存储量整数进行分类和显示。
<b>-l</b>	将系统时间和用户时间分离，而不是组合它们。
<b>-m</b>	显示每个用户的进程数和 CPU 分钟数。
<b>-n</b>	按调用数对输出进行分类。
<b>-r</b>	将分类的顺序倒置。
<b>-s</b>	将记帐文件合并到摘要文件中。
<b>-S SaveFile</b>	使用指定的已保存文件作为命令摘要文件，而不是使用 <b>/var/adm/savacct</b> 文件。
<b>-t</b>	显示每个命令实时与用户和系统时间和之比。
<b>-u</b>	暂挂所有其他标志并且显示每个命令的用户数字标识和命令名。
<b>-U UserFile</b>	使用指定文件代替 <b>/var/adm/usracct</b> 文件来积累由 <b>-m</b> 标志显示的每个用户的统计信息。
<b>-v Number</b>	输入使用了指定次数或少于指定次数的每个命令的名称。当查询时，如果输入 y (是)，则将命令添加到 junk 类别中，并让它作为此类别的一部分显示在将来的摘要中。

## 示例

1. 要对 **/var/adm/pacct** 文件中的所有命令进行记帐记录总结，输入：

```
sa -a
```

将仅用过一次的命令放在 **other** 字段下。

2. 要按平均 CPU 时间进行记帐记录总结，输入：

```
sa -k
```

## 文件

<b>/usr/sbin/sa</b>	包含 <b>sa</b> 命令。
<b>/etc/sa</b>	包含至 <b>sa</b> 命令的符号链接。
<b>/var/adm/pacct</b>	包含原始记帐记录。
<b>/var/adm/savacct</b>	包含记帐记录摘要。
<b>/var/adm/usracct</b>	包含用户记录的记帐摘要。

## 相关信息

**acctcms** 命令、**acctcom** 命令、**acctcon1** 或 **acctcon2** 命令、**acctmerg** 命令、**acctprc1**、**acctprc2** 或 **accton** 命令、**fwtmp** 命令、**runacct** 命令。

有关“记帐系统”、每日和每月报告的准备以及记帐文件的更多信息，请参阅《操作系统与设备管理》中的『系统记帐』。

《操作系统与设备管理》中的『设置记帐子系统』描述了建立记帐系统必须采取的步骤。

有关可以自动运行或从键盘输入的记帐命令列表的信息，请参阅《操作系统与设备管理》中的『记帐命令』。



---

## sa1 命令

### 用途

将二进制数据收集并保存在 `/var/adm/sa/sadd` 文件中。

### 语法

```
/usr/lib/sa/sa1 [ Interval Number ]
```

### 描述

**sa1** 命令是 **sadc** 命令的 shell 过程变体，并处理该命令的所有标志和参数。**sa1** 命令将二进制数据收集并存储在 `/var/adm/sa/sadd` 文件中，其中 *dd* 表示每月的日期。*Interval* 和 *Number* 参数指定应以 *Interval* 秒的间隔，写入记录 *Number* 次。如果不指定这些参数，将写一个记录。要使用本命令，必须具有写入 `/var/adm/sa` 目录的许可权。

**sa1** 命令设计为由 **cron** 命令自动启动。如果 **sa1** 命令不是每天从 **cron** 命令运行，则 **sar** 命令显示 `/usr/lib/sa/sa1` 数据文件不存在的消息。

### 示例

要创建 **sar** 活动的每日记录，将以下条目放入 `adm crontab` 文件中：

```
0 8-17 * * 1-5 /usr/lib/sa/sa1 1200 3 &
```

### 文件

<code>/var/adm/sa</code>	指定包含每日数据文件的目录。
<code>/var/adm/sa/sadd</code>	包含每日数据文件，其中 <i>dd</i> 参数是一个表示每月日期的数字。
<code>/usr/lib/sa/sa1</code>	包含 <b>sa1</b> 命令。

### 相关信息

**sadc** 命令、**sar** 命令和 **sa2** 命令。

**cron** 守护程序。

有关“记帐系统”、每日和每月报告的准备以及记帐文件的更多信息，请参阅《操作系统与设备管理》中的『系统记帐』。

《操作系统与设备管理》中的『设置记帐子系统』描述了建立记帐系统必须采取的步骤。

有关可以自动运行或从键盘输入的记帐命令列表的信息，请参阅《操作系统与设备管理》中的『记帐命令』。

---

## sa2 命令

### 用途

将每日报告写入 `/var/adm/sa/sar` 文件中。

## 语法

`/usr/lib/sa/sa2`

## 描述

**sa2** 命令是 **sar** 命令的一个变体 shell 过程，该过程将每日报告写入 `/var/adm/sa/sardd` 文件，其中，*dd* 表示每月的日期。**sa2** 命令处理 **sar** 命令的所有标志与参数。

**sa2** 命令设计为由 **cron** 命令自动运行，并与 **sa1** 命令并行运行。

## 示例

要每天运行 **sa2** 命令，可将以下条目放入根 **crontab** 文件：

```
5 18 * * 1-5 /usr/lib/sa/sa2 -s 8:00 -e 18:01 -i 3600 -ubcwyaqvm &
```

这将生成每日报告，称为 `/var/adm/sa/sardd`。同时将除去一周以前的报告。

## 文件

<code>/var/adm/sa</code>	指定包含每日数据文件的目录。
<code>/var/adm/sa/sar<i>dd</i></code>	包含每日数据文件，其中 <i>dd</i> 参数是一个表示每月日期的数字。
<code>/usr/lib/sa/sa2</code>	到 <b>sa2</b> 命令的 shell 脚本的路径。

## 相关信息

**sa1** 命令、**sadc** 命令和 **sar** 命令。

**cron** 守护程序。

《操作系统与设备管理》中的『系统记帐』和『设置帐户子系统』。

《操作系统与设备管理》中的『记帐命令』列出了可以自动运行或从键盘输入的记帐命令。

---

## sact 命令

### 用途

显示当前 SCCS 文件编辑状态。

### 语法

**sact** *File* ...

### 描述

**sact** 命令读取“源代码控制系统”（SCCS）文件，并且将与 *File* 变量指定值有关的 P 文件内容写入标准输出，（如果有的话）。**get -e** 命令创建 P 文件。如果将 **-**（减号）指定为 *File* 值，则 **sact** 命令读取标准输入并将每一行解释为一个 SCCS 文件的名称。如果 *File* 值是目录，**sact** 命令将执行所有 SCCS 文件上的操作。

## 退出状态

该命令返回以下退出值:

- 0 成功完成。
- >0 发生错误。

## 示例

要显示 P 文件的内容, 输入:

```
sact File
```

## 文件

`/usr/bin/sact` 包含指向 SCCS `sact` 命令的路径。

## 相关信息

`delta` 命令、`get` 命令、`sccs` 命令和 `unget` 命令。

`sccsfile` 文件格式。

《AIX 5L V5.3 通用编程概念: 编写并调试程序》中的『SCCS 命令列表』和『源代码控制系统 (SCCS) 概述』。

---

## sadc 命令

### 用途

提供系统数据收集器报表。

### 语法

```
/usr/lib/sa/sadc [ Interval Number ] [ Outfile ]
```

```
/usr/lib/sa/sa1 [ Interval Number ]
```

```
/usr/lib/sa/sa2
```

### 描述

`sadc` 命令, 即数据收集器, 按照指定的时间间隔 (以秒计) (*Interval*) 以指定次数 (*Number*) 采样系统数据。它以二进制格式写入指定的输出文件或标准输出。当 *Interval* 和 *Number* 都没有指定时, 将写成哑元记录, 用于系统启动时标记计数器重新从 0 开始时的时间。`sadc` 命令设计作为 `sar` 命令的后端进行使用。

操作系统包含多个计数器, 这些计数器会随各种系统操作的出现而增加。各种系统操作包含:

- 系统配置参数
- 系统部件使用率计数器
- 缓冲区使用情况计数器
- 磁盘和磁带 I/O 活动计数器
- tty 设备活动计数器

- 转换与子例程计数器
- 文件存取计数器
- 队列活动计数器
- 进程间通信计数器

注: **sadc** 命令仅报告本地活动。

## 安全性

访问控制: 这些命令应只将执行 (x) 访问权授予 **adm** 组的成员。

## 示例

要将以一秒为间隔的 10 个记录写入 **/tmp/rpt** 二进制文件, 输入:

```
sadc 1 10 /tmp/rpt
```

## 文件

<b>/var/adm/sa/sadd</b>	包含每日数据文件, <i>dd</i> 表示每月的日期。
<b>/var/adm/sa/sar<i>dd</i></b>	包含每日报告文件, <i>dd</i> 表示每月的日期。
<b>/tmp/rpt</b>	包含用于 <b>sar</b> 命令输入的二进制文件。
<b>/tmp/sa.adrf1</b>	包含地址文件。

## 相关信息

**sar** 命令、**sa1** 命令、**sa2** 命令和 **timex** 命令。

**cron** 守护程序。

《操作系统与设备管理》中的『设置帐户子系统』。

《操作系统与设备管理》中的『系统记帐』描述了系统记帐, 而『记帐命令』列出可以自动运行或从键盘输入的记帐命令。

## sar 命令

### 用途

收集、报告或保存系统活动信息。

### 语法

```
/usr/sbin/sar [ { -A [ -M ] | [ -a ] [ -b ] [ -c ] [ -d ] [ -k ] [ -m ] [ -q ] [ -r ] [ -u ] [ -v ] [ -w ] [ -y ] [ -M ] } ] [ -P ProcessorIdentifier, ... | ALL ] [ -ehh [ :mm [ :ss ] ] ] [ -XFile ] [ -fFile ] [ -iSeconds ] [ -oFile ] [ -shh [ :mm [ :ss ] ] ] [ Interval [ Number ] ]
```

### 描述

**sar** 命令将操作系统中选定的累计活动计数器的内容写到标准输出。基于 *Number* 和 *Interval* 参数的值, 记帐系统以指定的时间间隔 (以秒为单位) 和指定间隔次数写入信息。*Number* 参数的缺省采样时间间隔为 1 秒。收集的数据也可以保存在由 **-o File** 标志所指定的文件中。

**sar** 命令抽取并写入以前保存在文件里的标准输出记录。此文件既可以由 **-f** 标志指定的文件，或者也可以是缺省情况下标准系统活动每日数据文件，即 `/var/adm/sa/sadd` 文件，其中 *dd* 参数表示当前日期。

没有 **-P** 标志的 **sar** 命令报告整个系统（所有处理器）的统计信息，对于以百分比表示的数值计算其平均值，否则计算其总数。如果给定 **-P** 标志，则 **sar** 命令报告与指定的一个或多个处理器相关的活动。如果给定 **-P ALL**，则 **sar** 命令报告每个单独的处理器的统计信息，跟在整个系统的统计信息后面。

可以使用标志选择关于特定系统活动的信息。没有指定任何标志，则仅选择系统部件活动。指定 **-A** 标志，则选择所有活动。**sar** 命令在开始显示统计信息之前显示 CPU 的数目和当前活动的磁盘的数目。

**sar** 命令的缺省版本（CPU 使用率报告）可能是用户开始系统活动调查首先要运行的工具之一，因为它监视主要的系统资源。如果 CPU 使用率接近 100%（用户 + 系统），采样的工作负荷为 CPU 受限。如果 I/O 等待花费相当大的时间百分比，这意味着 CPU 执行受阻，等待磁盘 I/O。该 I/O 可能是必要的文件存取，也可能与由于内存不足而进行页面调度有关。

**注：**系统用于等待远程文件存取的时间没有计入 I/O 等待时间。如果一项任务的 CPU 使用率和 I/O 等待时间相对较低，且响应时间不能令人满意，则可考虑调查有多少时间耗费在等待远程 I/O 上。由于没有一个高级命令提供对远程 I/O 等待的统计，所以跟踪数据可能对这种观察有用。如果系统配置中出现了影响 **sar** 命令输出的更改，**sar** 将显示直至当前迭代的平均值，然后显示关于配置更改的警告消息。该命令在显示已更新的系统配置信息之后将继续输出。

## 计算 CPU 磁盘 I/O 等待时间的方法

AIX 4.3.3 及以后的版本包含用于计算等待磁盘 I/O 所花费 CPU 时间（*wio* 时间）百分比的方法的改进。在 AIX 4.3.2 及早期操作系统版本中采用的方法在某些情况下会在 SMP 上显示出夸大的 *wio* 时间。*wio* 时间是由 **sar**（`%wio`）命令、**vmstat**（`wa`）命令和 **iostat**（`%iowait`）命令报告的。

AIX 4.3.2 及早期版本中使用的方法如下：每个处理器上的每次时钟中断（每个处理器 100 次/秒）时，都必须确定要将最后 10 ms 时间花费在四个类别（`usr/sys/wio/idle`）中的哪个类别上。如果时钟中断时 CPU 正处于 `usr` 模式下，则会将该时钟信号加入 `usr` 的类别中。如果时钟中断时 CPU 处于 `kernel` 模式下，则 `sys` 类别获取该信号。如果 CPU 不忙，则检查是否有任何磁盘 I/O 正在进行。如果有正在进行的磁盘 I/O，则增加 *wio* 类别。如果没有正在进行的磁盘 I/O 且 CPU 不忙，则 `idle` 类别取得该信号。*wio* 时间的夸大是由于所有空闲 CPU 都被视为 *wio*，而不管等待 I/O 的线程数。例如，只有一个线程正在执行 I/O 的系统可能报告大于 90% 的 *wio* 时间，而不管该系统拥有的 CPU 数量。

操作系统 AIX 4.3.3 及以后版本中使用的方法如下：在操作系统 AIX 4.3.3 中所作的更改是仅当某空闲 CPU 上有未完成的 I/O 开始时才将其标记为 *wio*。当只有几个线程正在执行 I/O，或者系统为空闲时，此方法所报告的 *wio* 时间要低得多。例如，系统中有四个 CPU 且一个线程正在执行 I/O 时，最多只报告 25% 的 *wio* 时间。系统中有十二个 CPU 且一个线程正在执行 I/O 时，最多报告 8% 的 *wio* 时间。NFS 客户机通过 VMM 进行读/写，现在将 `biod` 在 VMM 等待 I/O 完成所花费的时间报告为 I/O 等待时间。

如果期望多个采样和多个报告，可为 **sar** 命令指定一个输出文件，这样就方便多了。将 **sar** 命令的标准输出数据定向到 `/dev/null`，并将 **sar** 命令作为后台进程运行。以上操作的语法是：

```
sar -A -o data.file interval count > /dev/null &
```

以二进制格式获取所有数据并将其保存在文件（`data.file`）中。然后，使用带有 **-f** 选项的 **sar** 命令可以有选择地显示数据。

**sar** 命令调用一个称为 **sadc** 的进程来存取系统数据。构建两个 shell 脚本（`/usr/lib/sa/sa1` 和 `/usr/lib/sa/sa2`），它们由 **cron** 命令运行并提供每日统计信息和报告。样本节包括在

`/var/spool/cron/crontabs/adm` crontab 文件中（但被注释掉），用来指定 **cron** 守护程序应在何时运行 shell 脚本。以这种方式收集的数据对于描述一段时间内系统的使用特征以及确定使用高峰期是很有用的。

您可以在系统启动时通过取消 `/etc/rc` 脚本中相应行的注释将一条哑元记录插入标准系统活动每日数据文件中。对于 CPU 时间小于前一个记录的任何记录，**sar** 命令都将报告时间变化为负值。如果用在 `/etc/rc` 中注释掉的哑元记录插入行重新引导系统，将出现这种情况。

从 AIX 5.3 开始，**sar** 命令报告使用率度量值 `physc` 和 `%entc`，这些度量值与微分区和同时多线程环境相关联。这些度量值只显示在微分区和同时多线程环境中。`physc` 表明分区消耗的物理处理器的数目（如果达到系统范围使用率）或者逻辑 CPU 的数目（如果指定了 **-P** 标志），`%entc` 表明分配的授权容量（如果达到系统范围的使用率）或授予的授权容量（如果指定了 **-P** 标志）的百分比。当分区在限制方式下运行时，分区获取的容量无法比分配到的容量更大。在未限制方式中，分区可以获取比实际分配更大的容量。这称为授予的授权容量。如果指定了 **-P** 标志并且存在未使用的容量，**sar** 将未使用的容量作为带有 `cpu id U` 的单独 CPU 进行显示。

注：**sar** 命令只报告本地活动。

可以使用基于 Web 的系统管理器（wsm）中的系统应用程序来运行此命令。

也可以使用“系统管理界面工具”（SMIT）**smit sar** 的快速路径来运行此命令。

## 标志

- A** 没有使用 **-P** 标志时，使用 **-A** 标志相当于指定 **-abckmqruvwy**。当与 **-P** 标志一起使用时，**-A** 相当于指定 **-acmuw**。没有使用 **-M** 标志时，在第一个时间间隔内，在分组在一起的多行中，标题将仅在数据之前打印一次。当此标志与 **-M** 标志一起使用时，每个迭代的每行数据前都有相应的标题。
- a** 报告使用文件存取系统例程指定每秒调用各种系统文件存取例程的次数。当与 **-P** 标志一起使用时，为每个指定的处理器提供此信息；否则，只在系统范围内提供此信息。将显示以下数值：

### **dirblk/s**

目录搜索例程为特定文件定位目录项所读取的 512 字节块的数量。

**iget/s** 调用支持多文件系统类型的几个 i-node 查询例程中的任何一个例程。**iget** 例程返回一个指向文件或设备的 i-node 结构的指针。

### **lookuppn/s**

调用目录搜索例程，根据给定的路径名该例程可以查找 v-node 的地址。

- b** 报告缓冲区活动中每秒用于传输、存取以及高速缓存（内核块高速缓存）的命中率。存取第 3 版本中的大多数文件会忽略内核块缓冲，因此不生成这些统计信息。然而，如果程序打开块设备或裸字符设备进行 I/O，则使用传统的存取机制使生成的统计信息变得有意义。将显示以下数值：
- bread/s, bwrit/s**  
报告 I/O 操作的块的数目。这些 I/O 通常由内核执行来管理块高速缓存区，正如在 **lread/s** 值的描述中所论述的那样。
- lread/s, lwrit/s**  
报告逻辑 I/O 请求的数量。当执行对块设备的逻辑读或写时，可能请求小于整块大小的逻辑传输大小。系统对所有块的物理设备单元进行存取，并将这些块缓冲在为此目的而留出的内核缓冲区（块 I/O 高速缓存区）中。这个高速缓存区由内核管理，因此对块设备的多个逻辑读写可以存取以前缓冲在高速缓存中的数据，而且不需要对该设备进行真正的 I/O。应用程序对块的读写请求作为逻辑读写而统计地报告。由管理高速缓存区的内核所执行的块 I/O 作为块读写而报告。
- pread/s, pwrit/s**  
报告对裸设备的 I/O 操作的数量。对裸字符设备的 I/O 请求不像对块设备的 I/O 请求那样缓冲。直接对设备执行 I/O。
- %rcache, %wcache**  
报告高速缓存的效率（高速缓存命中百分比）。此百分比由下式计算得出：  

$$[(100) \times (\text{lreads} - \text{bread}) / \text{lreads}]$$
- c** 报告系统调用。当与 **-P** 标志一起使用时，为每个指定的处理器提供此信息；否则，只在系统范围内提供此信息。将显示以下数值：
- exec/s, fork/s**  
报告 **fork** 和 **exec** 系统调用的总数。
- sread/s, swrit/s**  
报告读 / 写系统调用的总数。
- rchar/s, wchar/s**  
报告读 / 写系统调用传送的字符总数。
- scall/s** 报告系统调用的总数。  
**注：**根据 **sar** 命令运行的时间间隔，该命令本身可生成相当多的读和写。在无工作负荷的情况下运行 **sar** 统计，以了解 **sar** 命令在总统计信息中的份额。
- d** 报告每个块设备（除磁带驱动器外）的活动。所报告的活动数据是：
- %busy** 报告设备忙于执行传送请求所用的那一部分时间。
- avque** 在 AIX 5.3 之前：报告发送到磁盘但尚未完成的请求瞬间数量。AIX 5.3：报告正等待发送到磁盘的请求的平均数量。
- read/s, write/s, blk/s**  
以千字节 / 秒报告设备之间的读写传送。
- await, avserv**  
每个请求的平均等待时间和服务时间以毫秒计。
- e hh[:mm][:ss]** 设置报告的结束时间。缺省结束时间是 18:00。
- X File** 从 *File* 文件中抽取记录，该文件由 **sar/sadc** 命令的 AIX 4.3 或 AIX 4.2 版本生成。
- f File** 从 *File*（由 **-o File** 标志创建）文件中抽取记录。*File* 参数的缺省值是当前每日数据文件，即 **/var/adm/sa/sadd** 文件。
- i Seconds** 以尽可能接近 *Seconds* 参数所指定的秒数选择数据记录。否则，**sar** 命令将报告在该数据文件中找到的所有秒数。

- k** 报告内核进程活动。将显示以下数值：  
**kexit/s** 报告每秒内核进程终止数。  
**kproc-ov/s**  
 报告由于实施进程阈值限制而不能创建内核进程的次数。  
**ksched/s**  
 报告每秒分配执行任务的内核进程数。
- M** 当与 **abckmqruvwy]** 的至少两种组合一起使用时，或者与 **-A** 标志一起使用时，在输出中可以有多个标题。在这种方式下，每个迭代的每行数据前都有相应的标题。当不带 **[Interval [Number]]** 使用此标志时，此标志会被忽略。
- m** 报告每秒消息（发送和接收）以及信号量（创建、使用或破坏）活动。当与 **-P** 标志一起使用时，为每个指定的处理器提供此信息；否则，只在系统范围内提供此信息。将显示以下数值：  
**msg/s** 报告 IPC 消息原语的数量。  
**sema/s**  
 报告 IPC 信号量原语的数量。
- o File** 将读取的内容以二进制格式保存在文件里。每个读取为一条单独的记录，而且每个记录都含有一个标识该读取时间的标记。
- P ProcessorIdentifier, ... | ALL** 报告指定的一个或多个处理器的每个处理器的统计信息。指定 **ALL** 关键字报告每个处理器的统计信息以及所有处理器的全部统计信息。在指定要报告的统计信息的标志中，只有 **-a**、**-c**、**-m**、**-u** 和 **-w** 标志对 **-P** 标志有意义。
- q** 报告队列统计信息。将显示以下数值：  
**runq-sz**  
 报告在运行队列中的平均内核线程数。  
**%runocc**  
 报告运行队列占用的时间百分比。  
**swpq-sz**  
 报告等待页面调入的内核线程数的平均值。  
**%swpocc**  
 报告调动队列占用的时间百分比。  
 注：任何列为空值表示相关的队列为空。
- r** 报告页面调度统计信息。将显示以下数值：  
**cycle/s** 报告每秒页面替换循环的数量。  
**fault/s** 报告每秒缺页故障的数量。它没有对生成 I/O 的缺页故障进行计数，因为一些缺页故障可以在没有 I/O 的情况下解决。  
**slots** 报告在调页空间中空闲页的数量。  
**odio/s** 报告每秒非页面调度磁盘 I/O 的数量。
- s hh[:mm[:ss]]** 设置数据开始时间，使 **sar** 命令以（遵循）指定时间抽取有时间标记的记录。缺省开始时间是 08:00。



**-u** 报告每个处理器或整个系统的统计信息。当与 **-P** 标志一起使用时，为每个指定的处理器提供此信息；否则，只在系统范围内提供此信息。因为 **-u** 标志信息以百分比表示，所以系统范围信息只是每个处理器统计信息的平均值。并且，I/O 等待状态是以整个系统而不是每个处理器来定义的。将显示以下数值：

**%idle** 报告 CPU 为空闲（且没有未完成的 I/O 请求）所占的时间百分比。

**%sys** 报告 CPU 在系统（或内核）级别上执行所花费的时间的百分比。

**%usr** 报告 CPU 在用户（或应用程序）级别上执行所花费的时间的百分比。

**%wio** 报告 CPU 空闲（但系统有未完成的磁盘 / NFS I/O 请求）所占的时间的百分比。请参阅以上的详细描述。

**physc** 报告物理处理器的消耗数量。仅当在启用了共享处理器或同时多线程的情况下运行分区，才会作此报告。

**%entc** 报告消耗的授权容量的百分比。仅当在共享处理器的情况下运行分区，才会作此报告。因为计算该数据所依赖的时基可能变化，所以授权容量百分比有时超过 100%。仅在采样时间间隔短时才会注意到超过。

**注：**如果没有请求其他特定内容选项，则 **sar** 命令将报告系统部件的活动情况。如果使用了 **-P** 标志并且在共享处理器的情况下运行分区，并且如果使用的分区容量是分配的容量，则将报告带有 **cpuid U** 的 CPU 行以显示系统范围的未使用容量。如果分区在共享处理器的情况下以未限制方式运行，则 **%entc** 将报告就每个 CPU 行所授予的授权容量的百分比，以及在系统范围的 CPU 行中分配的授权容量的百分比。

**-v** 报告进程、内核线程、i-node 和文件表的状态。将显示以下数值：

**file-sz, inod-sz, proc-sz, thrd-sz**

报告每个表中正在使用的条目数。

**-w** 报告系统切换的活动。当与 **-P** 标志一起使用时，为每个指定的处理器提供此信息；否则，只在系统范围内提供此信息。将显示以下数值：

**pswch/s**

报告每秒上下文转接的数量。

**-y** 报告每秒 tty 设备的活动。

**canch/s**

报告 tty 规范输入队列字符。对于 AIX V4 及以后的版本，本字段总是 0（零）。

**mdmin/s**

报告 tty 调制解调器中断。

**outch/s**

报告 tty 输出队列字符。

**rawch/s**

报告 tty 输入队列字符。

**revin/s** 报告 tty 接收中断。

**xmtin/s**

报告 tty 发送中断。

## 安全性

访问控制：这些命令应只将执行（x）访问权授予 **adm** 组的成员。

## 示例

1. 要报告系统部件活动，请输入：

```
sar
```

2. 要在随后的 40 秒内每隔 2 秒报告当前的 tty 活动，请输入：

```
sar-y -r 2 20
```

3. 要观察系统部件 10 分钟，并对数据进行排序，请输入：

```
sar -o temp 60 10
```

4. 要报告最前面的两个处理器的 cpu 活动，请输入：

```
sar -u -P 0,1
```

这样将产生与以下类似的输出：

cpu	%usr	%sys	%wio	%idle
0	45	45	5	5
1	27	65	3	5

5. 要报告所有处理器和系统范围内的消息、信号量及 CPU 活动，请输入：

```
sar -mu -P ALL
```

而在一个有四个处理器的系统上，将产生与以下类似的输出（最后一行表示对所有处理器的系统范围内的统计）：

cpu	msgs/s	sema/s	%usr	%sys	%wio	%idle
0	7	2	45	45	5	5
1	5	0	27	65	3	5
2	3	0	55	40	1	4
3	4	1	48	41	4	7
-	19	3	44	48	3	5

6. 要查看系统范围内所有处理器的物理处理器消耗和授权消耗，请在共享处理器逻辑分区机器中运行 **sar** 命令，如下所示：

```
sar -P ALL
```

在双逻辑处理器系统上，它会产生类似以下的输出（最后一行表明所有处理器的系统范围的统计信息，带有 **cpuid U** 的行表明系统范围未使用的容量）：

cpu	%usr	%sys	%wio	%idle	physc	%entc
0	0	0	0	100	0.02	3.1
1	0	0	0	100	0.00	1.0
U	-	-	0	96	0.48	96.0
-	0	0	0	100	0.02	4.0

7. 要在随后的 40 秒内每 2 秒以单独的标题报告每次迭代的 3 行数据中每行数据的系统调用、内核进程和页面调度活动，请输入：

```
sar -Mckr 2 20
```

8. 要在随后的 40 秒内每 2 秒以多组标题报告所有活动，请输入：

```
sar -MA 2 20
```

## 文件

**/usr/sbin/sar**

包含 **sar** 命令。

**/bin/sar**

表示与 **sar** 命令的符号链接。

**/var/adm/sa/sadd**

表示每日数据文件，其中 *dd* 参数是表示该月的第几日的数字。

## 相关信息

**sadc** 命令、**sa1** 命令和 **sa2** 命令。

《操作系统与设备管理》中的『系统记帐』。

《操作系统与设备管理》中的『设置帐户子系统』。

《操作系统与设备管理》中的『记帐命令』列出了可以自动运行或从键盘输入的记帐命令。

有关安装基于 Web 的系统管理器的信息，请参阅《AIX 5L V5.3 基于 web 的系统管理器管理指南》中的『第二章：安装与系统需求』。

---

## savebase 命令

### 用途

将“设备配置”数据库中的基本定制设备信息保存到引导设备中。

### 语法

```
savebase [ -o Path ] [ -d Fine ] [ -v ]
```

### 描述

**savebase** 命令存储在系统引导第一阶段使用的基本设备的定制信息。缺省情况下，**savebase** 命令从 `/etc/objrepos` 目录检索这个信息。然而，也可以使用 **-o** 标志指定一个 ODM 目录来重设此操作。**savebase** 命令运行时通常不使用任何参数。它使用 `/dev/ipl_blv` 特殊文件链接来确定输出目标。

也可以使用 **-d** 标志来指定目标文件或设备，例如 `/dev/hdisk0` 设备文件。要确定特定输出目标，**-d** 标志会确定 **savebase** 在其中写入基本定制设备数据的文件。该文件可以是常规文件，也可以是设备特殊文件。设备特殊文件确定磁盘设备特殊文件或引导逻辑卷设备特殊文件。

当磁盘上只有一个引导逻辑卷时可以使用磁盘设备特殊文件。**savebase** 命令确保给定磁盘只有一个引导逻辑卷并且可引导。如果这些条件都不为真，则 **savebase** 不会将基本定制设备数据保存到磁盘，且退出并出现错误。

当磁盘上有第二个引导逻辑卷时，必须将引导逻辑卷设备特殊文件用作目标设备，以确定基本定制设备数据将存储到哪一个引导映像中。即使磁盘上只有一个引导逻辑卷，也可以使用引导逻辑卷设备特殊文件。在将任何数据保存到给定设备特殊文件之前，**savebase** 可命令确保该设备特殊文件是引导逻辑卷并且可引导。如果这些检查中任何一个检查失败，则 **savebase** 退出并出现错误。

**savebase** 命令使用与 **CuDv** 对象类别中的每个条目对应的 `PdDv.base` 字段，确定需要保存的设备信息。特定地，`PdDv.base` 字段是位掩码，代表此设备作为基本设备时的引导类型。**savebase** 命令通过存取 **CuAT** 对象类别中的 `boot_mask` 属性来确定当前的引导类型。此属性值是位掩码，适用于 `PdDv.base` 字段以确定哪些设备是基本设备。

**注：**基本设备是在第一阶段的引导过程中就已配置的那些设备；它们根据引导类型（掩码）的不同而不同。例如，如果掩码是 **NETWORK\_BOOT**，则网络设备就是基本设备；如果掩码是 **DISK\_BOOT**，则磁盘设备就是基本设备。`/usr/include/sys/cfgdb.h` 文件定义了引导类型掩码。

注: **savebase** 命令不再使用 **-m** 标志。出于兼容性方面的原因, 可以指定该标志, 但 **savebase** 可有效地忽略它。

## 标志

**-d** *File* 指定要写入基本信息的目的地文件或设备。  
**-o** *Path* 指定包含“设备配置”数据库的目录。  
**-v** 将冗长的输出写到标准输入中。

## 示例

1. 要保存基本定制信息和查看冗长的输出, 请输入:

```
savebase -v
```

2. 要指定一个非 **/usr/lib/objrepos** 目录的 ODM 目录, 请输入:

```
savebase -o /tmp/objrepos
```

3. 要将基本定制信息保存在 **/dev/hdisk0** 设备文件 (而不是引导盘) 中, 请输入:

```
savebase -d /dev/hdisk0
```

## 文件

**/usr/include/sys/cfgdb.h**  
**/usr/lib/objrepos/PdDv**  
**/etc/objrepos/CuDv**  
**/etc/objrepos/CuAt**  
**/etc/objrepos/CuDep**  
**/etc/objrepos/CuDvDr**

定义基本设备的引导掩码类型。  
包含系统支持的所有已知设备类型的条目。  
包含系统中定义的所有设备实例的条目。  
包含定制的特定于设备的属性信息。  
描述由其他设备实例决定的设备实例。  
通过使用“设备配置库”例程, 存储需要并发管理的关键资源的信息。

## 相关信息

**bosboot** 命令和 **restbase** 命令。

《AIX 5L V5.3 通用编程概念: 编写并调试程序》中的『程序员对象数据管理器 (ODM) 概述』。

AIX 5L Version 5.3 Kernel Extensions and Device Support Programming Concepts 中的 Device Configuration Subsystem: Programming Introduction 和 List of Device Configuration Commands。

---

## savecore 命令

### 用途

保存系统转储。

### 语法

```
savecore { [ [ -c ] [ -d ] [ -f ] ] | [ -F [ -d ] ] } DirectoryName SystemName
```

### 描述

**savecore** 命令的功能是保存系统转储, 并且通常在系统启动时运行。

**savecore** 命令检查是否已经进行新的转储并且有足够的空间来保存。系统转储保存在 *DirectoryName/vmcore.n* 文件中，系统保存在 *DirectoryName/vmunix.n* 文件中。*n* 变量在 *DirectoryName/bounds* 文件中指定。如果这个文件不存在，将创建这个文件，其缺省值为 **0**，*n* 变量使用该值。随后每保存一个转储，*n* 变量就增加 1。

**savecore** 命令也检查是否压缩过当前转储。如果压缩过，则压缩的转储将被复制到一个名为 *DirectoryName/vmcore.n.Z* 的文件中，其中 **.Z** 是一个标准指示，表明该文件被压缩过。

注：以上适用于 AIX 4.3.2 及更新版本。

如果系统转储不是来自 *linux* 系统，则必须将系统名称作为 *SystemName* 而给出。

注：**savecore** 命令只保存当前转储和前一次转储。

该目录可能包含一个名为 **minfree** 的文件。这个文件包含该目录中剩余空闲的千字节数。**minfree** 文件用来保证在复制该转储之后，剩余可用空间的最小量。

## 标志

- c** 标记转储无效（不是最新的），但不复制。
- d** 只复制转储。不复制系统。
- f** 即使转储看起来是无效的，也复制它。
- F** 报告复制目录中用于转储的可用空间大小。由于 **savecore** 命令保留当前转储和前一次转储，删除其他转储，故报告的可用空间数可能比实际可用空间大。如果指定了 **-F** 标志，则不能进行复制。这个标志只有与 **-d** 标志一起使用时才有效。

## 示例

1. 要复制转储（而非系统）到 *DirectoryName* 中，请输入：

```
savecore -d DirectoryName
```

2. 即使转储无效，也要复制它，请输入：

```
savecore -f -d DirectoryName
```

3. 要标记该转储为无效，请输入：

```
savecore -c
```

4. 要复制转储和系统，请输入：

```
savecore -d DirectoryName SystemName
```

5. 要查看可用于转储的空间有多大，请输入：

```
savecore -d -F DirectoryName
```

## 相关信息

**sysdumpdev** 命令和 **sysdumpstart** 命令。

---

## savevg 命令

### 用途

查找和备份属于指定卷组的所有文件。

## 语法

```
savevg [ -a ] [ -A ] [ -b Blocks ] [ -e ] [ -f Device ] [ -i | -m ] [ -p ] [ -r ] [ -v ] [ -V ] [ -X ]  
VGName
```

## 描述

**savevg** 命令查找和备份属于指定卷组的所有文件。卷组必须是联机的卷组，且必须安装文件系统。**savevg** 命令使用由 **mkvgdata** 命令创建的数据文件。数据文件可以是以下之一：

### **/image.data**

包含根卷组 (**rootvg**) 的信息。**savevg** 命令使用这个文件创建备份映像，“网络安装管理” (NIM) 使用备份映像将卷组重新安装到当前的系统或新的系统中。

### **/tmp/vgdata/vgname/vgname.data**

包含用户卷组信息。*VGName* 变量表示卷组的名称。**savevg** 命令使用这个文件创建备份映像，**restvg** 命令使用备份映像重新制作用户卷组。

要在 CD 中创建操作系统备份，请使用 **mkcd** 命令。

**注：**如果卷组是根卷组，**savevg** 命令将不生成可引导磁带。虽然磁带不是可引导的，但是磁带开头的三个映像是哑元，用以替代通常在可引导磁带中的映像。实际的系统备份是第四个映像。

## 标志

- a** 不备份扩展属性或 NFS4 ACL。
- A** 备份 DMAPI 文件系统的文件。
- b *Blocks*** 指定在单一输出操作中写入 512 字节块的数量。如果不指定这个参数，则 **backup** 命令使用与选定物理设备相应的缺省值。值越大，到磁带设备的物理传输也越大。指定值必须是正在使用的设备的物理块大小的倍数。
- e** 将 **/etc/exclude.vgname** 文件中指定的文件从该命令所做的备份中排除出去。

**注：**如果不想备份某些文件，可使用 ASCII 编辑器创建 **/etc/exclude.rootvg** 文件，并输入不想包括在系统备份映像中的文件名模式。这个文件中的模式将被输入到 **grep** 命令的模式匹配约定中，以确定不备份哪些文件。如果想排除 **/etc/exclude.rootvg** 文件中列出的文件，请选择“排除文件”字段，并按下 Tab 键一次，将缺省值改为“yes”。

例如，要排除名为 **scratch** 的目录下的所有内容，编辑排除文件如下：

```
/scratch/
```

例如，要排除目录为 **/tmp** 的内容，并避免将其他路径名中含有 **/tmp** 的目录都排除，编辑排除文件如下：

```
^./tmp/
```

备份所有与 **.** (当前工作目录) 相关的文件。要排除那些对于该文件或目录而言，搜索字符与命令行的开头字符串相匹配是尤为重要的任意文件或目录，请使用 **^** (插入符号) 作为搜索字符串的第一个字符，后面跟随 **.** (点字符) 和要排除的文件名或目录。

如果排除的文件名或目录是另一个文件名或目录的子串，请使用 **^.** (插入字符，后面跟随点字符)，以表示搜索应从行的开头处开始，和 **/** 或使用 **\$** (美元字符) 以表示搜索应结束于行尾。

- f *Device*** 指定存储映像的设备或文件名。缺省值是 **/dev/rmt0** 设备。
- i** 通过调用 **mkvgdata** 命令创建数据文件。
- m** 通过调用带 **-m** 标志的 **mkvgdata** 命令以使用映射文件创建数据文件。
- p** 备份时禁用文件的软件压缩功能。某些磁带驱动器使用自己的压缩或压缩算法。

- r** 备份用户卷组信息和管理数据文件文件。此标志将备份类似 `/tmp/vgdata/vgname/vgname.data` 的文件和映射文件（如果有的话）。但不备份用户数据文件。此备份可用于在不恢复用户数据文件的情况下创建用户卷组。不能对 `rootvg` 执行此操作。
- v** 详细方式。备份时列出备份文件。
- V** 验证磁带备份。这个标志使 `savevg` 验证备份磁带上每个文件的文件头，并报告出现的任何读取错误。
- X** 如果需要，可指定自动扩展 `/tmp` 文件系统。当在磁带上创建可引导备份时，可能需要扩展 `/tmp` 文件系统，以便给引导映像创建空间。

## 参数

`VGName` 指定要备份的卷组名称。

## SMIT 快速路径

1. 要列出由 `savevg` 命令创建的根卷组备份内容，请输入以下 SMIT 快速路径：  
`smit lsmksysb`
2. 要列出由 `savevg` 命令创建的用户卷组备份内容，请输入以下 SMIT 快速路径：  
`smit lsbackvg`
3. 要从根卷组备份中恢复个别文件，请输入以下 SMIT 快速路径：  
`smit restmksysb`
4. 要从用户卷组备份中恢复个别文件，请输入以下 SMIT 快速路径：  
`smit restsavevg`

## 示例

1. 要将根卷组（操作系统映像）备份到 `/mysys/myvg/myroot` 备份文件中，并创建 `/image.data` 文件，请输入：  
`savevg -i -f/mysys/myvg/myroot rootvg`
2. 要将 `uservg` 卷组备份到缺省磁带驱动器（`dev/rmt0`）中，并创建新的 `uservg.data` 文件，请输入：  
`savevg -i uservg`
3. 要备份 `data2` 卷组并在 `rmt1` 设备中创建映射文件和新的 `data2.data` 文件，请输入：  
`savevg -mf/dev/rmt1 data2`
4. 要备份 `data2` 卷组，排除 `/etc/exclude.data2` 中列出的文件，请输入：  
`savevg -ief/dev/rmt1 data2`
5. 要将卷组 `my_vg` 备份到 `/dev/rmt0` 磁带中，并验证文件头的可读性，请输入：  
`savevg -f /dev/rmt0 -V my_vg`

## 文件

<code>/image.data</code>	当卷组是 <code>rootvg</code> 时使用。
<code>/tmp/vgdata/vgname /vgname.data</code>	当卷组不是 <code>rootvg</code> ，而卷组名是 <code>vgname</code> 时使用。

## 相关信息

`backup` 命令、`bosboot` 命令、`mkcd` 命令和 `mkszfile` 命令。

---

## scan 命令

### 用途

产生一个每次扫描一行消息的列表。

### 语法

```
scan [ +Folder ] [ Messages ] [ -form FormFile | -format String ] [ -noheader | -header ] [ -clear | -noclear ] [ -help ]
```

### 描述

**scan** 命令显示指定文件夹中消息的一行信息。每行提供消息号、日期、发送方、主题和尽可能多的消息体。缺省情况下，**scan** 命令显示当前文件夹中所有消息的信息。

如果在消息号后显示 +（加号），则该消息是文件夹中的当前消息。如果在消息号后显示 -（减号），说明已经对这个消息做了应答。如果日期后显示 \*（星号），则不显示 **Date:** 字段，并且显示的日期是消息最后一次更改的日期。

### 标志

<b>-clear</b>	在发送输出之后清除显示。 <b>scan</b> 命令使用 <b>\$TERM</b> 环境变量的值来确定如何清除显示。如果标准输出不是显示器，则 <b>scan</b> 命令发送输出之后发送一个格式反馈字符。
<b>+Folder</b>	指定要扫描的文件夹。缺省值是当前文件夹。
<b>-form FormFile</b>	将 <b>scan</b> 命令的输出以 <i>FormFile</i> 变量指定的备用格式显示。
<b>-format String</b>	将 <b>scan</b> 命令的输出以 <i>String</i> 变量指定的备用格式显示。
<b>-header</b>	将列有文件夹名、当前日期和时间的标题显示出来。
<b>-help</b>	列出命令语法、可用的参数选项（开关）和版本信息。 <b>注：</b> 对于消息处理程序（MH），必须完整拼写此标志的名称。
<i>Messages</i>	显示指定文件夹中每个指定消息的信息。指定消息时，可以使用以下引用： <i>Number</i> 指定消息的数目。 <i>Sequence</i> 指定一组由用户指定的消息。识别值包括： <b>all</b> 文件夹中的所有消息。这是缺省值。 <b>cur</b> 或 <b>.</b> （句点） 当前®消息。 <b>first</b> 文件夹中的第一条消息。 <b>last</b> 文件夹中的最后一条消息。 <b>next</b> 当前消息的下一条消息。 <b>prev</b> 当前消息的上一条消息。
<b>-noclear</b>	防止在发送输出之后将终端清除。这是缺省值。
<b>-noheader</b>	不让标题显示出来。这是缺省值。
<b>-width Number</b>	设置 <b>scan</b> 命令输出的列数。缺省值是显示器的宽度。



## 概要文件条目

在 `UserMhDirectory/.mh_profile` 文件中输入以下条目:

Alternate-Mailboxes:           指定邮箱。  
Current-Folder:                设置缺省当前文件夹。  
Path:                           指定 `UserMhDirectory`。

## 示例

1. 要获取当前文件夹中所有消息的单行列表, 请输入:

```
scan
```

系统使用消息响应如下:

```
3 04/17 dale@athena Status meeting <<The weekly status meeting
5 04/20 tom@venus Due Dates <<Your project is due to
6 04/21 dawn@tech Writing Clas <<There will be a writing
```

2. 要获取 `test` 文件夹中从 11 到 15 条消息的单行列表, 请输入:

```
scan +test 11-15
```

系统使用消息响应如下:

```
11 04/16 karen@anchor Meeting <<Today's meeting is at 2 p.m.
12 04/18 tom@venus Luncheon <<There will be a luncheon to
14 04/20 dale@athena First Draft <<First drafts are due
15 04/21 geo@gtwn Examples <<The examples will be written
```

## 文件

<code>\$HOME/.mh_profile</code>	包含消息处理程序 (MH) 用户概要文件。
<code>/etc/mh/scan.size</code>	包含扫描格式化字符串样本。
<code>/etc/mh/scan.time</code>	包含扫描格式化字符串样本。
<code>/etc/mh/scan.timely</code>	包含扫描格式化字符串样本。
<code>/usr/bin/scan</code>	包含 <code>scan</code> 命令的可执行格式。

## 相关信息

`inc` 命令、`pick` 命令和 `show` 命令。

`.mh_alias` 文件格式和 `.mh_profile` 文件格式。

《网络与通信管理》中的『邮件应用程序』。

---

## SCCS 命令

### 用途

SCCS 命令的管理程序。

## 语法

**sccs** [ **-r** ] [ **-dPath** ] [ **-pPath** ] *Command* [ *CommandFlags* ] *File* ...

## 描述

**sccs** 命令是一个管理程序，它将“源代码控制系统”（SCCS）命令集并入操作系统中。另外，**sccs** 命令可以用来分配或重新分配文件所有权（参阅 **-r** 标志）。

**sccs** 命令激活具有指定标志和参数的指定的命令。通常每个文件放在名为 **SCCS** 的目录下，并且文件命名为 **s.filename**。假定与工作目录有关的 **SCCS** 目录存在（除非使用了 **-p** 标志）。

在 **sccs** 命令语法句子中可以使用两类命令。第一类由 14 个可以在提示符下输入的 **sccs** 命令组成。第二类，即伪命令，只能用作 **sccs** 命令语法的一部分。有 12 个伪命令，执行以下操作：

<b>edit</b>	相当于 <b>get -e</b> 命令。
<b>delget</b>	在指定的文件上执行 <b>delta</b> 命令，然后获取新版本。文件的新版本具有扩展的识别关键字，而且不可编辑。
	标志:
	<b>-m, -p, -r, -s, -y</b> 可以传递给 <b>delta</b> 命令。
	<b>-b, -c, -i, -l, -s, -x</b> 可以传递给 <b>get</b> 命令。
<b>deledit</b>	除了 <b>get</b> 这一部分的语句中包括了 <b>-e</b> 标志外，它相当于 <b>delget</b> 伪命令。 <b>deledit</b> 伪命令对于在当前编辑会话中创建检查点很有用。
	标志:
	<b>-m, -p, -r, -s, -y</b> 可以传递给 <b>delta</b> 命令。
	<b>-b, -c, -i, -l, -s, -x</b> 可以传递给 <b>get</b> 命令。
<b>create</b>	通过复制一个具有相同名称的文件的初始内容，创建一个 <b>SCCS</b> 文件。如果文件创建成功，则先前的文件将被重命名，前面加一个逗号。不必一定要象用 <b>admin</b> 命令一样移动或删除原始文件。
	标志:
<b>fix</b>	接受与 <b>admin</b> 命令相同的标志。隐含 <b>-i</b> 标志。 除去一个指定的 <b>delta</b> ，但保留一份该 <b>delta</b> 的副本，并保持更改完整无缺。此伪命令对于修订小的编译器错误很有用。此伪命令不对文件更改作记录。
	标志:
<b>clean</b>	<b>-rSID</b> 表示一个必要的标志。 将所有文件从当前目录中除去，或者从一个可以从 <b>SCCS</b> 文件重新创建的指定目录中除去。不删除正在编辑的文件。
	标志:
<b>unedit</b>	<b>-b</b> 在确定哪些文件正在编辑时，将忽略分支。在同一目录下编辑的分支将会丢失。 相当于 <b>unget</b> 命令。将失去使用 <b>get</b> 命令后进行的任何更改。

<b>info</b>	<p>列出被编辑的全部文件。</p> <p>标志:</p> <p><b>-b</b> 在确定哪些文件正在编辑时, 将忽略分支。</p> <p><b>-u [Argument]</b> 只列出您或 <i>Argument</i> 参数所指定的用户正在编辑的文件。</p>
<b>check</b>	<p>显示正在编辑的全部文件。如果有正在编辑的文件, 则返回一个非零退出状态。可以在制作文件中使用检查程序, 以确保在安装版本前文件是完整的。安装前, 要检查返回代码。</p> <p>标志:</p> <p><b>-b</b> 在确定哪些文件正在编辑时, 将忽略分支。</p> <p><b>-u [Argument]</b> 只列出您或 <i>Argument</i> 参数指定的用户正在编辑的文件。</p>
<b>tell</b>	<p>在标准输出中列出正在编辑的全部文件, 每个条目后进行换行。</p> <p>标志:</p> <p><b>-b</b> 在确定哪些文件正在编辑时, 将忽略分支。</p> <p><b>-u [Argument]</b> 只列出您或 <i>Argument</i> 参数指定的用户正在编辑的文件。</p>
<b>diffs</b>	<p>显示正在编辑的当前程序版本与先前 <i>deltas</i> 的差别。</p> <p>标志:</p> <p><b>-r, -c, -l, -x, -t</b> 可以传递给 <b>get</b> 命令。</p> <p><b>-l, -s, -e, -f, -h, -b</b> 可以传递给 <b>diff</b> (而非 <b>sccsdiff</b>) 命令。</p> <p><b>-C</b> 可以传递给 <b>diff</b> (而非 <b>sccsdiff</b>) 命令, 以作为 <b>-c</b> 标志。 显示指定文件的 <b>verbose</b> 信息。</p>
<b>print (filename(s))</b>	<p>如果已设置了 <b>PROJECTDIR</b> 环境变量, 则它的值将确定工作目录。如果该值以 / (斜杠) 开头, 则直接使用它。否则, 将该值解释为用户名, 检查其主目录下名为 <b>src</b> 或 <b>source</b> 的子目录。如果找到子目录, 则将该子目录用作工作目录。</p>

## 标志

**-dPath** 为 SCCS 文件指定工作目录。缺省值是当前目录。将 **-d** 标志作为文件的完整路径名的前缀。当已设置了 **PROJECTDIR** 环境变量, 而且使用了 **-d** 标志时, 该命令行在确定工作目录时将覆盖环境值。

**-p** 为 SCCS 文件指定路径名。缺省值是 SCCS 目录。在路径名的最后部分的前面插入 **-p** 标志。

在执行过程中，此命令后所指定的全部标志都将被传递给该命令。要了解命令标志的描述，请参阅相应的命令描述。

示例:

```
sccs -d/x -py get a/b
```

转换为:

```
get /x/a/y/s.b
```

此选项用来创建别名。例如:

```
alias syssccs sccs -d/usr/src
```

使 **syssccs** 命令成为别名命令，别名命令可以按以下使用:

```
syssccs get cmd/who.c
```

当这样使用时，以上命令将检查 **s.who.c** 文件的 **/usr/src/cmd/SCCS** 目录。

**-r** 作为一个真正的用户，而不是作为 **sccs** 命令设置（使用 **set user id** 命令）的有效用户来运行 **sccs** 命令。

一些命令，如 **admin** 命令不能用作 **set user id**，否则任何人都可以改变授权。这样的命令始终由真正的用户来运行。

## 退出状态

该命令返回以下退出值:

**0** 成功完成。  
**>0** 发生错误。

## 示例

1. 要获取用于编辑的文件，对其进行编辑，然后产生一个新的 **delta**，请输入:

```
sccs get -e file.c  
ex file.c  
sccs delta file.c
```

2. 要从另一个目录取得文件，请输入:

```
sccs -p/usr/src/sccs/ get cc.c
```

或

```
sccs get /usr/src/sccs/s.cc.c
```

3. 要获取不在分支上、正在被编辑的文件列表，请输入:

```
sccs info -b
```

## 文件

**/usr/bin/sccs** 包含 **sccs** 命令，它是 SCCS 命令的管理程序。

## 相关信息

**admin** 命令、**cdc** 命令、**comb** 命令、**delta** 命令、**diff** 命令、**get** 命令、**prs** 命令、**rmdel** 命令、**sact** 命令、**sccsdiff** 命令、**sccshelp** 命令、**unget** 命令、**val** 命令、**vc** 命令和 **what** 命令。

**sccsfile** 文件格式。

《AIX 5L V5.3 通用编程概念：编写并调试程序》中的『SCCS 命令列表』和『源代码控制系统（SCCS）概述』。

---

## sccsdiff 命令

### 用途

比较 SCCS 文件的两个版本。

### 语法

```
sccsdiff -rSID1 -rSID2 [ -p ] [ -sNumber ] File ...
```

### 描述

**sccsdiff** 命令读取“源代码控制系统”（SCCS）文件的两个版本，对它们进行比较，并将两个版本的差别写到标准输出中。可以指定任意 SCCS 文件数，但是所有的文件要使用相同的参数。

### 标志

<b>-p</b>	通过 <b>pr</b> 命令进行输出。
<b>-rSID1</b>	将 <i>SID1</i> 指定为 SCCS 文件的一个 delta，以便使用 <b>sccsdiff</b> 命令进行比较。
<b>-rSID2</b>	将 <i>SID2</i> 指定为一个 SCCS 文件的另一个 delta，以便使用 <b>sccsdiff</b> 命令进行比较。
<b>-sNumber</b>	指定文件节大小，以便使用 <b>bdiff</b> 命令传递给 <b>diff</b> 命令。这对于 <b>diff</b> 命令由于高系统负载而失效尤为有用。

### 示例

要显示 SCCS 文件 `s.test.c` V1.1 和 V1.2 的差别，请输入：

```
sccsdiff -r1.1 -r1.2 s.test.c
```

### 文件

<code>/usr/bin/sccsdiff</code>	包含 SCCS <b>sccsdiff</b> 命令。 <b>sccsdiff</b> 命令支持多字节字符集（MBCS）数据作为文件名。
--------------------------------	--

## 相关信息

**bdiff** 命令、**diff** 命令、**get** 命令、**prs** 命令和 **sccshelp** 命令。

**sccsfile** 文件格式。

《AIX 5L V5.3 通用编程概念：编写并调试程序》中的『SCCS 命令列表』和『源代码控制系统（SCCS）概述』。

---

## sccshelp 命令

### 用途

提供关于 SCCS 消息或命令的信息。

### 语法

**sccshelp** [ *ErrorCode* ] [ *Command* ]

### 描述

**sccshelp** 命令显示关于使用指定的“源代码控制系统”（SCCS）命令的信息，或关于使用此命令所生成的消息的信息。每条消息都有一个关联代码，可以作为参数的一部分提供给 **sccshelp** 命令。可以提供零个或多个参数。如果不提供参数，则 **sccshelp** 命令将提示需要一个参数。可以将任意 SCCS 命令作为参数包括在 **sccshelp** 命令中。

*ErrorCode* 参数对代码进行指定，代码由数字和字母组成，出现在消息的结尾。例如，以下消息中，代码是 (cm7)：

```
There are no SCCS identification keywords in the file. (cm7)
```

### 示例

要在 **rmDEL** 命令和两个错误代码上获取 **sccshelp**，请输入：

```
$ sccshelp rmDEL gee ad3
```

**sccshelp** 命令应答：

```
rmDEL:
rmDEL -r<SID> <file> ...
ERROR:
1255-141 gee is not a valid parameter. Specify a valid command or error code.
ad3:
The header flag you specified is not recognized.
The header flag you supplied with the -d or the -f flag is not correct.
Choose a valid header flag.
```

### 文件

**/usr/bin/sccshelp** 包含 SCCS **sccshelp** 命令。

### 相关信息

**admin** 命令、**cdc** 命令、**comb** 命令、**delta** 命令、**get** 命令、**prs** 命令、**rmDEL** 命令、**sccsdiff** 命令和 **what** 命令。

**sccsfile** 文件格式。

《AIX 5L V5.3 通用编程概念：编写并调试程序》中的『SCCS 命令列表』和『源代码控制系统（SCCS）概述』。

---

## schedo 命令

### 用途

管理 CPU 调度程序的可调参数。

### 语法

```
schedo [ -p | -r ] { -o Tunable[=Newvalue]}
```

```
schedo [ -p | -r ] { -d Tunable }
```

```
schedo [ -p | -r ] -D
```

```
schedo [ -p | -r ] -a
```

```
schedo -h [ Tunable ]
```

```
schedo -L [Tunable ]
```

```
schedo -x [Tunable ]
```

注：允许多个 **-o**、**-d**、**-x** 标志和 **-L** 标志

### 描述

注：**schedo** 命令只能由 root 用户来执行。

使用 **schedo** 命令来配置调度程序的微调参数。此命令设置或显示所有调度程序的微调参数的当前值或下一个引导值。此命令也可以示参数永久更改，或将更改推迟到下一个重新引导之后生效。此命令是设置参数还是显示参数是由所带标志确定的。**-o** 标志执行这两个操作。它既可以显示参数的值，也可以为参数设置新值。

### 了解更改可调参数的效果

误用此命令可能导致性能下降或操作系统故障。在使用 **schedo** 更改系统参数前，请务必研究《性能管理》中相应的调整部分。

在修改任何可调参数之前，应该先仔细阅读以下可调参数部分中有关它的所有特性，然后按照任何“请参考”指针进行操作，以便完全了解其用途。

然后必须确保此参数的“诊断”和“调整”部分确实适用于您的情况，并确保更改此参数值能够有助于提高您的系统性能。

如果“诊断”部分和“调整”部分都仅包含“不适用”，则您可能应该永远不要更改此参数，除非 AIX 开发有明确的指示。

### 优先级计算参数

大多数用户进程的优先级随进程最近使用的 CPU 时间的数量而改变。CPU 调度程序的优先级计算是基于 **schedo** 所设置的两个参数，即 *sched\_R* 和 *sched\_D*。*sched\_R* 和 *sched\_D* 的值以 30 秒（1/32）为单位；即，调度程序使用下式计算作为对最近 CPU 使用的损失而加入到进程优先级值中的数量：

CPU 处罚 = (进程最近使用的 CPU 值) \* (r/32)

而对每个进程最近使用的 CPU 值进行每秒一次重新计算的公式是：

新的最近使用的 CPU 值 = ( 进程最近使用的旧的 CPU 值 ) \* ( d/32 )

$r$  (*sched\_R* 参数) 和  $d$  (*sched\_D* 参数) 的缺省值都是 16。这样将保持操作系统以前版本的 CPU 调度行为。在尝试这些值之前, 应该熟悉《性能管理指南》中『调节 CPU 调度程序』。

## 内存装入控制参数

当内存过量使用时, 操作系统调度程序通过暂挂进程来进行内存装入控制。系统并不将进程换出; 而是偷页, 因为需要它们来实现当前内存要求。通常, 从暂挂进程偷页。当满足下列条件时, 认为内存使用过量:

$p * h s$                     其中:

$p$  是最后一秒写入调页空间的页数  
 $h$  是由 *v\_repage\_hi* 参数指定的一个整数  
 $s$  是最后一秒发生的偷页数

当内存使用过度, 且满足下列条件时, 将暂挂进程:

$r * p f$                     其中:

$r$  是最后一秒进程累积的重新调页数  
 $p$  是由 *v\_repage\_proc* 参数指定的一个整数  
 $f$  是最后一秒进程所经历的缺页故障数

此外, 免除固定优先级进程和内核进程的暂挂。

术语重新调页是指属于某个进程的页由该进程管理后不久就重新引用页数。

用户还可以用 *v\_min\_process* 参数来指定最低的多道程序设计级别。这样做可以在整个进程暂挂期间确保有最少的进程数正在运行。活动的进程是那些可以运行、正在等待页 I/O 的进程。正在等待事件的进程和暂挂的进程不认为是活动的, 等待进程也不能认为是活动的。

当系统处于过量使用阈值以下  $n$  秒时, 暂挂的进程可以被加回到组合中, 其中  $n$  由 *v\_sec\_wait* 参数指定。首先根据进程的优先级, 其次根据进程暂挂时间的长短来将进程加回到系统中。

在试验这些值之前, 您应该非常熟悉《性能管理指南》中的“通过 *schedo* 命令调整 VMM 内存装入控制”。

## 时间片增量参数

**schedo** 命令也可以用来改变在调用分派器以选择运行另一个进程运行之前, 操作系统允许某个给定进程运行的时间量 (时间片)。这个时间间隔的缺省值是一个时钟周期 (10 毫秒)。时间片微调参数允许用户指定时间片长度要增加的钟信号量。

在 AIX V4 中, 此参数仅适用于使用 *SCHED\_RR* 调度策略的线程。请参阅《线程调度策略》。

## fork() 重试时间间隔参数

如果因为没有足够的调页空间用于创建新的进程而使 **fork()** 子例程调用失败, 则系统将在等待一段指定的时间后重试该调用。时间间隔由步速子进程微调参数来设置。

## 对称多线程的专门术语

支持多个运行队列。在此模式下, 每个 CPU 都具有自己的运行队列。Power 5 处理器支持对称多线程, 其中每个物理处理器都有两个执行引擎, 称为硬件线程。每个硬件线程本质上等价于单个的 CPU。缺省情况下启用对称多线程, 但是也可以动态地禁用 (或者重新启用) 对称多线程。如果启用了对称多线程, 那么每个硬件线



程为单独的运行队列服务。例如，在一个 4 路系统上，如果对称多线程被禁用或者当前无法使用，那么除全局运行队列之外还有 4 个运行队列。如果启动了对称多线程，那么除全局运行队列之外还有 8 个运行队列。

属于同一个物理处理器的硬件线程称为兄弟线程。主兄弟线程是指物理处理器的第一个硬件线程。辅助兄弟线程是指物理处理器的第二个硬件线程。

## 虚拟处理器管理

可以定义比处理分区中工作所需的虚拟处理器更多的虚拟处理器。可以不在降低整体处理器使用率或导致缺少虚拟处理器的情况下通过使用较少的虚拟处理器来减少分派虚拟处理器的开销。虚拟处理器并不是动态地从分区中除去的，而是不再使用，只在提供了更多工作时才再次可用。每个虚拟处理器最多使用一个物理处理器。所需的虚拟处理器的数目是通过物理处理器利用率与可调整的 **vpm\_xvcpus** 的和进行上舍入来确定的：

```
number = ceiling( p_util + vpm_xvcpus)
```

其中 *number* 是所需的虚拟处理器的数目，*p\_util* 是物理处理器利用率，而 **vpm\_xvcpus** 是可调节的（它指定要启用的更多虚拟处理器的数目）。如果 *number* 小于当前已启用的虚拟处理器的数目，则将禁用某个虚拟处理器。如果 *number* 大于当前已启用的虚拟处理器的数目，将启用已禁用的虚拟处理器。连接到已禁用的虚拟处理器的线程仍然可以在已禁用的虚拟处理器上运行。

## 节点装入

节点装入，或者简称装入，是指所有运行队列（包括全局运行队列）的平均运行队列深度乘以 256，随着时间将大幅度地变得平滑。例如，装入为 256 意味着，如果有 16 个 CPU（包括对称多线程 CPU），那么在过去几毫秒内，在系统中大约有 16 个可运行作业。

## 标志

- a** 显示所有可调参数的当前值、重新引导值（当与 **-r** 一起使用时）或永久值（当与 **-p** 一起使用时），每行一对 *Tunable = Value*。对于永久选项，如果参数的重新引导和当前值相等，则对于一个参数只显示一个值。否则，该值显示为 **NONE**。
- d *Tunable*** 将 *Tunable* 重新设置为缺省值。如果可调参数值（即，它当前的值不是缺省值）需要更改，但在该组合中没有使用 **-r**，则不对它做更改，而是显示一个警告。
- D** 将所有的可调参数重新设置为缺省值。如果需要更改的可调参数为 **Bosboot** 或“重新引导”类型或“增量”类型，并是从其缺省值更改的，而且组合中没有使用 **-r**，则不做更改，而是显示一个警告。
- h [*Tunable*]** 如果指定了一个 *Tunable* 参数，则显示有关该参数的帮助。否则，显示 **schedo** 命令用法语句。
- L [ *Tunable* ]** 列出一个或所有可调参数的特征，每个一行，使用以下格式：

```

NAME          CUR  DEF  BOOT  MIN  MAX  UNIT  TYPE
DEPENDENCIES
-----
v_repage_hi   0   0   0     0   2047M          D
-----
v_repage_proc 4   4   4     0   2047M          D
-----
v_sec_wait    1   1   1     0   2047M seconds  D
-----
...
where:
CUR = current value
DEF = default value
BOOT = reboot value
MIN = minimal value
MAX = maximum value
UNIT = tunable unit of measure
TYPE = parameter type: D (for Dynamic), S (for Static), R (for Reboot),
      B (for Bosboot), M (for Mount), I (for Incremental), C (for Connect), and d (for Deprecated)
DEPENDENCIES = list of dependent tunable parameters, one per line

```

**-o** *Tunable* [=Newvalue] 显示值或将 *Tunable* 设置为 *Newvalue*。如果需要更改一个可调参数（所指定的值与当前值不同），而且它是 **Bosboot** 或 **Reboot** 类型，或者是 **Incremental** 类型，并且当前值比指定值要大，在组合中没有使用 **-r**，则不做更改，而是显示一个警告。

当在组合中使用 **-r** 却没有指定新值时，显示可调参数的下一个引导值。当在组合中使用 **-p** 却没有指定新值时，仅当可调参数的当前值和下一个引导值相同时才显示值。否则，该值显示为 **NONE**。

**-p** 当与 **-o**、**-d** 或 **-D** 组合使用时，将所做的更改应用到当前值和重新引导值中，即打开更新 **/etc/tunables/nextboot** 文件并更新当前值。这些组合无法用在 **Reboot** 和 **Bosboot** 类型参数上，因为它们的当前值不能更改。

当与 **-a** 或 **-o** 一起使用而没有指定新值时，仅当参数的当前值与下一个引导值相同的情况下才会显示该值。否则，该值显示为 **NONE**。

**-r** 在与 **-o**、**-d** 或 **-D** 组合使用时，将更改应用到重新引导值上，即打开更新 **/etc/tunables/nextboot** 文件。如果对任何 **Bosboot** 类型的参数做了更改，则提示用户运行 **bosboot**。

当与 **-a** 或 **-o** 一起使用而没有指定新值时，显示可调参数的下一个引导值而不是当前值。列出一个或所有可调参数的特征，每行一个，使用以下（电子表格）格式：

**-x** [*Tunable*]

```
tunable,current,default,reboot,min,max,unit,type,{dtunable }
```

where:

```
current = current value
default = default value
reboot = reboot value
min = minimal value
max = maximum value
unit = tunable unit of measure
type = parameter type: D (for Dynamic), S (for Static), R (for Reboot),
      B (for Bosboot), M (for Mount), I (for Incremental),
      C (for Connect), and d (for Deprecated)
dtunable = space separated list of dependent tunable parameters
```

对 **Mount**（安装）类型参数的任何更改（使用 **-o**、**-d** 或 **-D**）将显示一条消息，警告用户该更改仅对以后的安装有效。

对 **Connect**（连接）类型参数的任何更改（使用 **-o**、**-d** 或 **-D** 标志）都会导致重新引导 **inetd**，并显示一条消息，警告用户该更改仅对以后套接字连接有效。

在不使用 **-r** 的情况下，任何试图对 **Bosboot** 或 **Reboot** 类参数的更改（使用 **-o**、**-d** 或 **-D**）都会导致一个错误消息。

在不使用 **-r** 的情况下，任何试图用比 **Incremental**（增量）类型参数当前值小的新值来更改它的值（使用 **-o**、**-d** 或 **-D**）都会导致一个错误消息。

## 可调参数类型

所有由调整命令（**no**、**nfso**、**vmo**、**ioo**、**raso** 和 **schedo**）操作的可调参数分为以下几种类别：

动态	如果能在任何时间更改参数
静态	如果参数从来不能更改
重新引导	如果只能在重新引导期间更改参数
<b>Bosboot</b>	如果只能通过运行 <b>bosboot</b> 和重新引导机器来更改参数

安装	如果对参数的更改仅对以后的文件系统或目录安装有效。
增量	如果除了引导时间只能增量参数
连接	如果对参数的更改仅对以后套接字连接有效
不推荐	如果 AIX 的当前发行版不再支持更改此参数。

对于 Bosboot 类型的参数，无论何时进行更改，调整命令都会自动提示用户，询问他们是否想要执行 **bosboot** 命令。对于 Connect（连接）类型的参数，调整命令自动重新启动 **inetd** 守护程序。

请注意，受管于 **schedo** 命令的当前参数集仅包括 Dynamic 和 Reboot 类型。

## 兼容性方式

当以 5.2 以前版本的兼容性方式（由 **sys0** 的 **pre520tune** 属性控制，请参阅《性能管理》中的『AIX 5.2 的兼容性方式』）运行时，参数（除了 Bosboot 类型的那些参数）的重新引导值并不真正有意义，因为在该方式下不会在引导时应用这些参数。

在 5.2 版本以前的兼容性方式下，仍然可以通过将微调命令调用嵌入引导序列期间所调用的脚本中来设置微调参数的重新引导值。因此，不用 **-r** 标志，也可以设置 **Reboot** 类型的参数，这样现有的脚本可以继续工作。

当机器“迁移”到 AIX 5.2 时，此模式自动“打开”。对于完全安装，它是“关闭”的，而参数的重新引导值是通过在重新引导序列期间应用 **/etc/tunables/nextboot** 文件的内容来设置的。只有在该模式下，**-r** 和 **-p** 标志才可以完全起作用。请参阅 *AIX 5L Version 5.3 Performance Tools Guide and Reference* 中的 Kernel Tuning 以获取更多信息。

## 可调参数

affinity_lim	<b>用途:</b>	设置在介入分派后 SCHEN_FIFO2 策略不再对线程有利的介入分配数。
	<b>值:</b>	缺省值: 7 范围: 0 至 100 类型: 动态
	<b>诊断:</b>	不适用
	<b>调整:</b>	一旦某个线程采用 SCHED_FIFO2 策略运行，对此变量进行调节就可能（也可能不）影响该线程和工作负荷性能。应通过试验和出错来确定理想值。
allowMCMmigrate	<b>请参考:</b>	线程的调度策略
	<b>用途:</b>	在负载不重的情况下，在 MCM 之间迁移线程时启用（1）或禁用（0）障碍弱化功能。
	<b>值:</b>	缺省值: 0（禁用） 范围: 0 至 1 类型: 布尔型
	<b>诊断:</b>	不适用
	<b>调整:</b>	不适用

big_tick_size	<p><b>用途:</b> 设置经过 cpu 的物理时标时间间隔和同步时标。</p> <p><b>值:</b></p> <p style="padding-left: 20px;">缺省值: 1</p> <p style="padding-left: 20px;">范围: 1 至 100</p> <p style="padding-left: 20px;">类型: 动态</p> <p><b>诊断:</b> 不适用</p> <p><b>调整:</b> 此值乘以 10 ms 即为时标时间间隔, 应平均分为 100 份。使用此参数将降低系统统计信息的准确性。</p>
fixed_pri_global	<p><b>用途:</b> 使固定优先级线程保持在全局运行队列中。</p> <p><b>值:</b></p> <p style="padding-left: 20px;">缺省值: 0</p> <p style="padding-left: 20px;">范围: 0 至 1</p> <p style="padding-left: 20px;">类型: 动态</p> <p><b>诊断:</b> 不适用</p> <p><b>调整:</b> 如果是 1, 则将固定优先级线程放在全局运行队列中。</p> <p><b>请参考:</b></p> <p style="padding-left: 20px;">调度程序运行队列</p>
force_grq	<p><b>用途:</b> 将非 MPI 线程保留在全局运行队列中。</p> <p><b>值:</b></p> <p style="padding-left: 20px;">缺省值: 0</p> <p style="padding-left: 20px;">范围: 0 或 1</p> <p style="padding-left: 20px;">类型: 动态</p> <p><b>诊断:</b> 不适用</p> <p><b>调整:</b> 如果设置为 1, 将只有 MPI 和边界线程使用局部运行队列, 这可能将降低性能。</p>
hotlocks_enable	<p><b>用途:</b> 启用 (1) 或禁用 (0) 热锁的硬件优先级增强功能。</p> <p><b>值:</b></p> <p style="padding-left: 20px;">缺省值: 0 (禁用)</p> <p style="padding-left: 20px;">范围: 0..1</p> <p style="padding-left: 20px;">类型: 动态</p> <p><b>诊断:</b> 不适用</p> <p><b>调整:</b> 不适用</p>
idle_migration_barrier	<p><b>用途:</b> 用来确定何时可以将线程迁移到其他处理器上。</p> <p><b>值:</b></p> <p style="padding-left: 20px;">缺省值: 4</p> <p style="padding-left: 20px;">范围: 0 至 100</p> <p style="padding-left: 20px;">类型: 动态</p> <p><b>诊断:</b> 不适用</p> <p><b>调整:</b> 该值除以 16, 然后乘以负载平均值。所得的结果用来确定是否将作业迁移到其他节点上 (基本上进行负载均衡)。</p>

krlock_confer2self	<p><b>用途:</b> 尝试获取 krlock krlock_spinb4confer 次数之后启用 (1) 或禁用 (0) 授予自身。此参数只适用于 64 位内核。</p> <p><b>值:</b></p> <p style="padding-left: 20px;">缺省值: 0 (禁用)</p> <p style="padding-left: 20px;">范围: 0...1</p> <p style="padding-left: 20px;">类型: 动态 (当 32 位内核正在运行时为重新引导。)</p> <p><b>诊断:</b> 不适用</p> <p><b>调整:</b> 不适用</p>
krlock_conferb4alloc	<p><b>用途:</b> 在轮转 <b>slock_spinb4confer</b> 之后, 尝试获取或分配 krlock 之前, 启用 (1) 或禁用 (0) 授予。此参数只适用于 64 位内核。</p> <p><b>值:</b></p> <p style="padding-left: 20px;">缺省值: 0 (禁用)</p> <p style="padding-left: 20px;">范围: 0...1</p> <p style="padding-left: 20px;">类型: 动态 (当 32 位内核正在运行时为重新引导。)</p> <p><b>诊断:</b> 不适用</p> <p><b>调整:</b> 不适用</p>
krlock_enable	<p><b>用途:</b> 启用 (1) 或禁用 (0) krlock。此参数只适用于 64 位内核。</p> <p><b>值:</b></p> <p style="padding-left: 20px;">缺省值: 1 (启用)</p> <p style="padding-left: 20px;">范围: 0...1</p> <p style="padding-left: 20px;">类型: 动态 (当 32 位内核正在运行时为重新引导。)</p> <p><b>诊断:</b> 不适用</p> <p><b>调整:</b> 不适用</p>
krlock_spinb4alloc	<p><b>用途:</b> 在轮转 <b>slock_spinb4confer</b> 和授予 (如果打开 <b>krlock_conferb4alloc</b>) 之后, 分配 krlock 之前其他获取尝试的数目。此参数只适用于 64 位内核。</p> <p><b>值:</b></p> <p style="padding-left: 20px;">缺省值: 1</p> <p style="padding-left: 20px;">范围: 1...MAXINT</p> <p style="padding-left: 20px;">类型: 动态 (当 32 位内核正在运行时为重新引导。)</p> <p><b>诊断:</b> 不适用</p> <p><b>调整:</b> 不适用</p>
krlock_spinb4confer	<p><b>用途:</b> 授予 krlock 占位符 (或本身) 之前 krlock 获取尝试的数目。此参数只适用于 64 位内核。</p> <p><b>值:</b></p> <p style="padding-left: 20px;">缺省值: 1024</p> <p style="padding-left: 20px;">范围: 0...MAXINT</p> <p style="padding-left: 20px;">类型: 动态 (当 32 位内核正在运行时为重新引导。)</p> <p><b>诊断:</b> 不适用</p> <p><b>调整:</b> 不适用</p>

maxspin	<p><b>用途:</b> 设置进入休眠状态之前进行内核锁定而轮转的次数。</p> <p><b>值:</b></p> <p>缺省值: 在单处理器系统上为 1, 在多处理器系统上为 -1, 这意味着轮转可多达 <math>2^{32}</math> 次</p> <p>范围: -1 至 <math>2^{32}</math></p> <p>类型: 动态</p> <p><b>诊断:</b> 不适用</p> <p><b>调整:</b> 在多处理器系统上将该值增加或将它设置为 -1 可能减少空闲时间, 而在某些情况下也可能浪费 CPU 时间。建议不要在单处理器系统上增加该值。</p> <p><b>请参考:</b></p>
n_idle_loop_vlopri	<p>使用 schedo 命令来修改 MAXSPIN 参数</p> <p><b>用途:</b> 没有找到新工作时, 每次在空闲循环中运行低硬件优先级循环的次数。</p> <p><b>值:</b></p> <p>缺省值: 100</p> <p>范围: 0...1000000</p> <p>类型: 动态</p> <p><b>诊断:</b> 不适用</p> <p><b>调整:</b> 不适用</p>
pacefork	<p><b>用途:</b> 由于调页空间不够而导致对失败的子进程重新创建前等待的时钟信号数。</p> <p><b>值:</b></p> <p>缺省值: 10</p> <p>范围: 比 10 大的时钟信号数</p> <p>类型: 动态</p> <p><b>诊断:</b> 系统已用完调页空间, 不能创建该进程的子进程。</p> <p><b>调整:</b> 系统对失败的子进程重新创建 5 次。例如, 如果 <b>fork()</b> 子例程调用因为没有足够的调页空间可用来创建新进程而失败, 则系统在等待了指定的时钟信号数之后将重试该调用。</p> <p><b>请参考:</b></p>
sched_D	<p><b>fork()</b> 重试时间间隔参数</p> <p><b>用途:</b> 设置短期 CPU 使用延迟率。</p> <p><b>值:</b></p> <p>缺省值: 16</p> <p>范围: 0 至 32</p> <p>类型: 动态</p> <p><b>诊断:</b> 不适用</p> <p><b>调整:</b> 缺省值是每秒衰减短期 CPU 使用的 <math>1/2</math> (<math>16/32</math>)。减少此值使前台进程能够避免长时间与后台进程竞争。</p> <p><b>请参考:</b></p>
	<p>线程优先级值的计算</p>

sched_R	<p><b>用途:</b> 在优先级计算中, 设置短期 CPU 使用的权重因子。</p> <p><b>值:</b></p> <p style="padding-left: 20px;">缺省值: 16</p> <p style="padding-left: 20px;">范围: 0 至 32</p> <p style="padding-left: 20px;">类型: 动态</p> <p><b>诊断:</b> 运行: <b>ps al</b>。如果发现 <b>PRI</b> 列具有前台进程 (其 <b>NI</b> 值为 20) 的优先级值高于某些后台进程 (<b>NI</b> 值 &gt; 20) 的 <b>PRI</b> 值, 则可以减少 <b>r</b> 值。</p> <p><b>调整:</b> 缺省值是将短期 CPU 使用的 1/2 (16/32) 包括在优先级计算中。降低此值使前台进程更容易进行竞争。</p> <p><b>请参考:</b></p> <p style="padding-left: 20px;">线程优先级值的计算</p>
search_globalrq_mload	<p><b>用途:</b> 最小负载 (高于此负载时辅助兄弟线程将在分派器的全局运行队列中查找工作)。</p> <p><b>值:</b></p> <p style="padding-left: 20px;">缺省值: 256</p> <p style="padding-left: 20px;">范围: 0...4294967040</p> <p style="padding-left: 20px;">类型: 动态</p> <p><b>诊断:</b> 不适用</p> <p><b>调整:</b> 不适用</p>
search_smtrunq_mload	<p><b>用途:</b> 最小负载, 高于此负载时分派器还将搜索属于其兄弟硬件线程的运行队列。这是为了在物理处理器上进行负载均衡, 它与空闲负载均衡不同, 因为该检查是分派器在选择下一个要分派的作业时进行的。该检查与可调参数 <b>smtrunq_load_diff</b> 共同起作用。</p> <p><b>值:</b></p> <p style="padding-left: 20px;">缺省值: 256</p> <p style="padding-left: 20px;">范围: 0...4294967040</p> <p style="padding-left: 20px;">类型: 动态</p> <p><b>诊断:</b> 不适用</p> <p><b>调整:</b> 不适用</p>
setnewrq_sidle_mload	<p><b>用途:</b> 最小系统负载 (即使主兄弟线程未在空闲状态, 高于此负载时将考虑新工作的空闲辅助兄弟线程)。</p> <p><b>值:</b></p> <p style="padding-left: 20px;">缺省值: 384</p> <p style="padding-left: 20px;">范围: 0...4294967040</p> <p style="padding-left: 20px;">类型: 动态</p> <p><b>诊断:</b> 不适用</p> <p><b>调整:</b> 不适用</p>

shed_primrunq_mload	<p><b>用途:</b> 最大负载, 低于此负载时辅助兄弟线程将尝试将工作发送到主兄弟线程的运行队列上。</p> <p><b>值:</b></p> <p style="padding-left: 20px;">缺省值: 64</p> <p style="padding-left: 20px;">范围: 0...4294967040</p> <p style="padding-left: 20px;">类型: 动态</p> <p><b>诊断:</b> 不适用</p> <p><b>调整:</b> 不适用</p>
sidle_s1runq_mload	<p><b>用途:</b> 最小负载, 高于此负载时辅助兄弟线程的空闲负载均衡将在主兄弟线程的运行队列中搜索工作。</p> <p><b>值:</b></p> <p style="padding-left: 20px;">缺省值: 64</p> <p style="padding-left: 20px;">范围: 0...4294967040</p> <p style="padding-left: 20px;">类型: 动态</p> <p><b>诊断:</b> 不适用</p> <p><b>调整:</b> 不适用</p>
sidle_S2runq_mload	<p><b>用途:</b> 最小负载, 高于此负载时辅助兄弟线程将在空闲负载均衡的过程中, 在 CPU 拥有的其他运行队列中查找在其 S2 相似性域中的工作。</p> <p><b>值:</b></p> <p style="padding-left: 20px;">缺省值: 134</p> <p style="padding-left: 20px;">范围: 0...4294967040</p> <p style="padding-left: 20px;">类型: 动态</p> <p><b>诊断:</b> 不适用</p> <p><b>调整:</b> 建议绝不要将此可调参数设置为小于 sidle_S1runq_mload 的值。</p>
sidle_S3runq_mload	<p><b>用途:</b> 最小负载, 高于此负载时辅助兄弟线程将在空闲负载均衡的过程中, 在 CPU 拥有的其他运行队列中查找在其 S3 相似性域中的工作。</p> <p><b>值:</b></p> <p style="padding-left: 20px;">缺省值: 134</p> <p style="padding-left: 20px;">范围: 0...4294967040</p> <p style="padding-left: 20px;">类型: 动态</p> <p><b>诊断:</b> 不适用</p> <p><b>调整:</b> 建议绝不要将此可调参数设置为小于 sidle_S2runq_mload 的值。</p>
sidle_S4runq_mload	<p><b>用途:</b> 最小负载, 高于此负载时辅助兄弟线程将在任何本地运行队列中查找工作。</p> <p><b>值:</b></p> <p style="padding-left: 20px;">缺省值: 4294967040</p> <p style="padding-left: 20px;">范围: 0...4294967040</p> <p style="padding-left: 20px;">类型: 动态</p> <p><b>诊断:</b> 不适用</p> <p><b>调整:</b> 建议绝不要将此可调参数设置为小于 sidle_S3runq_mload 的值。</p>



slock_spinb4confer	<p><b>用途:</b> 授予之前尝试简单锁定的次数。</p> <p><b>值:</b></p> <p style="padding-left: 20px;">缺省值: 1024</p> <p style="padding-left: 20px;">范围: 0...MAXINT</p> <p style="padding-left: 20px;">类型: 动态</p> <p><b>诊断:</b> 不适用</p> <p><b>调整:</b> 不适用</p>
smt_snooze_delay	<p><b>用途:</b> 休眠（调用 <b>h_cede</b>）之前不做有用工作的空闲循环中的时间长短（以微秒计）。-1 值指示禁用休眠，而 0 值指示立即休眠。</p> <p><b>值:</b></p> <p style="padding-left: 20px;">缺省值: 0</p> <p style="padding-left: 20px;">范围: -1..100000000（100 秒）</p> <p style="padding-left: 20px;">类型: 动态</p> <p><b>诊断:</b> 不适用</p> <p><b>调整:</b> 不适用</p>
smtrunq_load_diff	<p><b>用途:</b> 要从兄弟运行队列偷取的任务的兄弟运行队列负载之间的最小负载差异。只有当负载大于可调参数 <code>search_smtrunq_mload</code> 的值时，才启用该参数。</p> <p><b>值:</b></p> <p style="padding-left: 20px;">缺省值: 2</p> <p style="padding-left: 20px;">范围: 1 至 4294967040</p> <p style="padding-left: 20px;">类型: 动态</p> <p><b>诊断:</b> 不适用</p> <p><b>调整:</b> 不适用</p>
tb_balance_s0	<p><b>用途:</b> 控制 SMT 核心繁忙均衡。该均衡尝试保持扩展到物理资源的计算绑定的线程。</p> <p><b>值:</b></p> <p style="padding-left: 20px;">缺省值: 0（禁用均衡）</p> <p style="padding-left: 20px;">范围: 0、1 或 2。1 值表示 MCM（S2 组）中启用了均衡。</p> <p style="padding-left: 20px;">类型: 动态</p> <p><b>诊断:</b> 不适用</p> <p><b>调整:</b> 不适用</p>
tb_balance_s1	<p><b>用途:</b> 控制芯片组繁忙均衡。该均衡尝试保持扩展到物理资源的计算绑定的线程。</p> <p><b>值:</b></p> <p style="padding-left: 20px;">缺省值: 1（系统范围内启用了均衡）</p> <p style="padding-left: 20px;">范围: 0、1 或 2。值 0 表示禁用均衡。值 1 表示 MCM（S2 组）中启用了均衡。</p> <p style="padding-left: 20px;">类型: 动态</p> <p><b>诊断:</b> 不适用</p> <p><b>调整:</b> 不适用</p>

tb_threshold	<p><b>用途:</b> 用于考虑一个线程的繁忙程度以优化 thread_busy 负载均衡的时标数。该均衡尝试保持扩展到物理资源的计算绑定的线程。</p> <p><b>值:</b></p> <p style="padding-left: 20px;">缺省值: 100 (1 秒)</p> <p style="padding-left: 20px;">范围: 10 到 1000 (0.1 到 10 秒)</p> <p style="padding-left: 20px;">类型: 动态</p> <p><b>诊断:</b> 不适用</p> <p><b>调整:</b> 不适用</p>
timeslice	<p><b>用途:</b> 线程放回运行队列之前可以运行的时钟信号数。</p> <p><b>值:</b></p> <p style="padding-left: 20px;">缺省值: 1</p> <p style="padding-left: 20px;">范围: 正整数值</p> <p style="padding-left: 20px;">类型: 动态</p> <p><b>诊断:</b> 不适用</p> <p><b>调整:</b> 增加此值可以减少分派线程的开销。该值是指一个时间片中时钟信号的总数, 且仅对固定优先级进程有影响。</p> <p><b>请参考:</b></p> <p style="padding-left: 20px;">通过 schedo 命令来修改调度程序时间片</p>
unboost_inflih	<p><b>用途:</b> 启用 (1) 或禁用 (0) <b>flih</b> 中热锁优先级的不增强功能。禁用时, 在分派器中产生不增强功能。</p> <p><b>值:</b></p> <p style="padding-left: 20px;">缺省值: 1 (启用)</p> <p style="padding-left: 20px;">范围: 0...1</p> <p style="padding-left: 20px;">类型: 动态</p> <p><b>诊断:</b> 不适用</p> <p><b>调整:</b> 不适用</p>
%usDelta	<p><b>用途:</b> 用来调整系统时钟, 使每个时钟周期都处于 -1 至 +1 秒的误差范围内。</p> <p><b>值:</b></p> <p style="padding-left: 20px;">缺省值: 100</p> <p style="padding-left: 20px;">范围: 0 至 100</p> <p style="padding-left: 20px;">类型: 动态</p> <p><b>诊断:</b> 不适用</p> <p><b>调整:</b> 用来调整时钟漂移。</p>

v_exempt_secs	<p><b>用途:</b> 设置以前被暂挂而最近恢复的进程的免除暂挂的秒数。</p> <p><b>值:</b></p> <p style="padding-left: 20px;">缺省值: 2</p> <p style="padding-left: 20px;">范围: 0 或正数</p> <p style="padding-left: 20px;">类型: 动态</p> <p><b>诊断:</b> 不适用</p> <p><b>调整:</b> 仅在出现系统颠簸现象时才检查此参数。</p> <p><b>请参考:</b></p> <p>VMM 内存装入控制工具和通过 schedo 命令调整 VMM 内存装入控制。</p>
v_min_process	<p><b>用途:</b> 设置被免除暂挂的最少进程数。</p> <p><b>值:</b></p> <p style="padding-left: 20px;">缺省值: 2</p> <p style="padding-left: 20px;">范围: 0 或正数</p> <p style="padding-left: 20px;">类型: 动态</p> <p><b>诊断:</b> 不适用</p> <p><b>调整:</b> 除了指内核进程外, 此数还指固定优先级低于 60 的进程、具有固定内存的进程或等待事件的进程。仅当在暂挂队列中有线程时才检查此参数。</p> <p><b>请参考:</b></p> <p>VMM 内存装入控制工具和通过 schedo 命令调整 VMM 内存装入控制。</p>
v_repage_hi	<p><b>用途:</b> 设置用以确定进程暂挂何时开始以及何时结束 (系统颠簸) 的系统范围内的标准。</p> <p><b>值:</b></p> <p style="padding-left: 20px;">缺省值: 6, 系统 RAM 为 128 MB 或更大时除外 (在该情况下值为 0)</p> <p style="padding-left: 20px;">范围: 0 或正数</p> <p style="padding-left: 20px;">类型: 动态</p> <p><b>诊断:</b> 如果 <math>v\_repage\_hi * page\_outs/sec &gt; page\_steals</math>, 那么进程可能被暂挂。</p> <p><b>调整:</b> 如果系统正在调页并使调度程序认为系统颠簸, 而实际上没有发生系统颠簸, 则可通过减少 <b>-h</b> 的值或将其设为 0 从而降低算法的敏感性, 这可能很有用。</p> <p><b>请参考:</b></p> <p>VMM 内存装入控制工具和通过 schedo 命令调整 VMM 内存装入控制。</p>
v_repage_proc	<p><b>用途:</b> 设置用于确定要暂挂哪些进程的每个进程标准。</p> <p><b>值:</b></p> <p style="padding-left: 20px;">缺省值: 4</p> <p style="padding-left: 20px;">范围: 0 或正数</p> <p style="padding-left: 20px;">类型: 动态</p> <p><b>诊断:</b> 不适用</p> <p><b>调整:</b> 这要求在某个给定进程要被内存装入控制暂挂前需要对该页进行更高级别的重新调页。仅在出现系统颠簸现象时才检查此参数。</p> <p><b>请参考:</b></p> <p>VMM 内存装入控制工具和通过 schedo 命令调整 VMM 内存装入控制。</p>

v_sec_wait	<p><b>用途:</b> 设置在系统颠簸结束后, 使暂挂进程成为可运行之前要等待的秒数。</p> <p><b>值:</b></p> <p style="padding-left: 20px;">缺省值: 1</p> <p style="padding-left: 20px;">范围: 0 或正数</p> <p style="padding-left: 20px;">类型: 动态</p> <p><b>诊断:</b> 不适用</p> <p><b>调整:</b> 仅在出现系统颠簸现象时才检查此参数。</p> <p><b>请参考:</b></p> <p>VMM 内存装入控制工具和通过 schedo 命令调整 VMM 内存装入控制。</p>
vpm_xvcpus	<p><b>用途:</b> 指定要启用的虚拟处理器的数目以及占用物理 CPU 利用率所需的虚拟处理器的数目。</p> <p><b>值:</b></p> <p style="padding-left: 20px;">缺省值: 0 (启用)</p> <p style="padding-left: 20px;">范围: -1 至 <b>INT_MAX</b></p> <p style="padding-left: 20px;">类型: 动态</p> <p><b>诊断:</b> 不适用</p> <p><b>调整:</b> 值为 <b>-1</b> 表示禁用此功能。</p>

## 示例

1. 要列出由 schedo 命令管理的全部可调参数的当前和重新引导的值、范围、单位、类型以及相关性, 请输入:  

```
schedo -L
```
2. 要列出 (电子表格格式) 由 **schedo** 命令管理的全部可调参数的当前和重新引导的值、范围、单元、类型以及相关性, 请输入:  

```
schedo -x
```
3. 要将 v\_sec\_wait 重新设置为缺省值, 请输入:  

```
schedo -d v_sec_wait
```
4. 要显示有关 sched\_R 的帮助, 请输入:  

```
schedo -h sched_R
```
5. 要在下一次重新引导之后将 v\_min\_process 设置为 4, 请输入:  

```
schedo -r -o v_min_process=4
```
6. 要将全部的 schedo 可调参数永久重新设置为缺省值, 请输入:  

```
schedo -p -D
```
7. 要列出所有 schedo 参数的重新引导值, 请输入:  

```
schedo -r -a
```

## 相关信息

**vmo** 命令、**ioo** 命令、**no** 命令、**nfso** 命令、**raso** 命令、**tunchange** 命令、**tunsave** 命令、**tunrestore** 命令、**tuncheck** 命令和 **tundefault** 命令。

*AIX 5L Version 5.3 Performance Tools Guide and Reference* 中的 Kernel Tuning

《性能管理》中的『AIX 5.2 兼容性方式』。

---

## scls 命令

### 用途

产生模块和驱动程序名的列表。

### 语法

```
scls [ -c | -l ] [ Module ... ]
```

### 描述

**scls** 命令为客户提供了查询当前可移植流环境（PSE）配置的方法。**scls** 命令产生模块和驱动程序名的列表。标志能用于产生增强型列表。在该命令行中的任何更多的参数为模块或驱动程序名，并且产生的输出只针对那些名称。

注: **scls** 命令需要 **sc** STREAMS 模块和 **nuls** 驱动程序。如果两者都不可用，那么 **scls** 命令不会成功。

### 标志

- c** 产生显示调用接口例程的次数的列表。
- l** 产生长列表，它显示扩展名类型、主号码以及关于 **module\_info** 结构的信息。

**-c** 和 **-l** 标志是互斥的标志。

### 参数

模块 指定输出信息的模块或驱动程序名。

### 文件

- sc** 动态可装载 STREAMS 配置模块
- nuls** 动态可装载 STREAMS 空设备

### 相关信息

**strload** 命令。

*AIX 5L Version 5.3 Communications Programming Concepts* 中的 List of Streams Commands。

*AIX 5L Version 5.3 Communications Programming Concepts* 中的 Configuring Drivers and Modules in the Portable Streams Environment (PSE)、STREAMS Overview。

---

## script 命令

### 用途

制作终端对话期的 typescript。

### 语法

**script** [ **-a** ] [ **-q** ] [ *File* ]

### 描述

**script** 命令制作显示在终端上的所有内容的 typescript。将 typescript 写到 *File* 参数指定的文件中。随后可将 typescript 发送到行式打印机。如果没有给定文件名，就会将 typescript 以文件名 **typescript** 保存在当前目录中。

该脚本在派生 shell 退出时结束。

当硬拷贝终端供应不足时，该命令用于产生硬拷贝记录。例如，当您在 CRT 显示器上工作并且需要对话的硬拷贝记录时，可以使用 **script** 命令。

因为 **script** 命令设置 **SetUserID** 方式位，出于安全性原因，调用此命令时将取消设置 **LIBPATH** 变量的值。不过，如果在环境文件中定义了 **LIBPATH**，那么在派生 shell 中会自动对其重新设置。此行为对 **NLSPATH** 环境变量而言也为真。相关信息可参阅 **exec** 子例程。

### 标志

- a** 将 typescript 附加到指定的文件或 **typescript** 文件。
- q** 禁止诊断消息。

### 文件

**/usr/bin/script** 包含 **script** 命令。

### 相关信息

**tee** 命令。

《操作系统与设备管理》中的『输入和输出重定向』描述了操作系统如何处理输入和输出，以及如何使用重定向和管道符号。

---

## sctpctl 命令

### 用途

控制和配置 SCTP。

## 语法

**sctpctrl** {load|unload|set}

**sctpctrl** stats [reset] [interval]

**sctpctrl** set {name=value|default [name]}

**sctpctrl** get [name]

## 描述

**sctpctrl** 命令用来控制和配置 SCTP 内核扩展。此命令可用来装入和卸装 SCTP 内核扩展。它还可以用来转储 SCTP 数据和集合，或者可以检索各种 SCTP 参数。另外，**sctpctrl** 命令可以用来读取和重新设置特定于 SCTP 的网络统计信息。

## 参数

*load*

*unload*

*stats* [reset] [interval]

*set* {name=value|default [name]}

*get* [name]

如果未装入 SCTP 内核扩展，则执行装入操作。

如果已经装入 SCTP 内核扩展，则执行卸装操作。

显示 SCTP 统计信息。**reset** 命令（可选）将清除（置零）统计信息。如果添加了 *interval* 参数（以秒计），则程序将不退出，但输出每隔 [*interval*] 秒的统计信息。

将 SCTP 可调参数设置为值。如果指定了 *default*，那么它会将所有的可调参数设置为其缺省值。如果在 *default* 的后面指定了可选 [*name*]，那么它会将 *name* 描述的可调参数设置为其缺省值。

获取由可选 *name* 参数描述的该可调参数的值。如果不指定 *name* 参数，那么它将获取所有可调参数的值。

## 示例

1. 要装入 SCTP 内核扩展，请输入以下内容：

```
sctpctrl load
```

2. 要卸装 SCTP 内核扩展，请输入以下内容：

```
sctpctrl unload
```

3. 要重置 SCTP 统计信息，请输入以下内容：

```
sctpctrl stats reset
```

此命令将使所有的 SCTP 统计信息置零。

4. 要获取 SCTP 可调参数值，请输入以下内容：

```
sctpctrl get
```

它将列出所有 SCTP 可调参数以及其值。以下是样本输出。

```
sctp_assoc_maxerr = 10
sctp_cookie_life = 60
sctp_delack_timer = 4
sctp_dontdelayack = 1
sctp_ecn = 1
sctp_ephemeral_high = 65535
sctp_ephemeral_low = 32768
sctp_instreams = 2048
sctp_maxburst = 8
sctp_outstreams = 10
```

```
sctp_path_maxerr = 5
sctp_pmtu_discover = 1
sctp_rttmax = 60
sctp_rttmin = 1
sctp_recvspace = 65536
sctp_sendspace = 65536
sctp_send_fewsacks = 0
```

5. 要将 **sctp\_path\_maxerr** 设置为值 6, 请输入以下内容:

```
sctpctrl set sctp_path_maxerr=6
```

## 位置

**/usr/sbin/sctpctrl**

## 文件

**/usr/sbin/sctpctrl** 包含 **sctpctrl** 命令。  
**/usr/lib/drivers/sctp** 包含 SCTP 内核扩展。

## 相关信息

**sctp\_peeloff** 子例程和 **sctp\_opt\_info** 子例程。

《网络与通信管理》中的『流控制传输协议』。

---

## sdiff 命令

### 用途

比较两个文件并以并排的格式显示差别。

### 语法

```
sdiff [ -l | -s ] [ -o OutFile ] [ -w Number ] File1 File2
```

### 描述

**sdiff** 命令读取由 *File1* 和 *File2* 参数指定的文件, 使用 **diff** 命令比较二者, 并将结果以并排格式写到标准输出中。如果行相同, 那么 **sdiff** 命令就会显示两个文件的每一行, 其间有一系列的空格。如果行只存在于 *File1* 参数指定的文件中, 那么在空白字段里就会显示 < (小于号), 如果行只存在于 *File2* 参数指定的文件中, 那么就显示 > (大于号), 如果两行不同就显示 | (竖线)。

当您指定了 **-o** 标志时, **sdiff** 命令就会将 *File1* 和 *File2* 参数指定的文件合并, 产生第三个文件。

注: **sdiff** 命令调用 **diff -b** 命令来比较两个输入文件。 **-b** 标志会引起 **diff** 命令忽略尾随空格和制表符, 并将其他空格字符串视为相同的字符串。

### 标志

**-l** 当行相同时只显示左侧内容。



- o OutFile** 通过将 *File1* 和 *File2* 参数指定的两个文件在受控下逐行合并，来创建 *OutFile* 变量指定的第三个文件。以下子命令管理该文件的创建：
- e** 以空文件开始 **ed** 命令。
  - e b or e l** 从两侧开始 **ed** 命令。
  - e l or e <** 从左侧开始 **ed** 命令。
  - e r or e >** 从右侧开始 **ed** 命令。
  - l** 将左侧添加到输出文件。
  - r** 将右侧添加到输出文件。
  - s** 停止显示相同的行。
  - v** 开始显示相同的行。
  - q** 执行以下功能之一：
    - 退出 **ed** 命令。
    - 如果没有运行 **ed** 命令，则退出 **sdiff** 命令。
    - 退出两条命令。不再有合并至输出文件的行时，执行该操作。
- 每次退出 **ed** 命令时，**sdiff** 命令将作为结果的已编辑文件写到 *OutFile* 变量指定的文件结尾。如果您在退出之前没有保存更改（例如，按下 **Ctrl-C** 按键顺序），那么 **sdiff** 命令将初始的输入写到输出文件中。
- s** 不显示相同的行。
- w Number** 设置输出行的宽度。*Number* 变量的缺省值是 130 个字符。*Number* 变量的最大宽度是 2048。*Number* 变量的最小宽度是 20。如果指定的值大于 2048，那么 **sdiff** 命令使用 2048。

## 示例

1. 要显示两个文件的比较，请输入：

```
sdiff chap1.bak chap1
```

**sdiff** 命令显示并排的列表，它比较 *chap1.bak* 和 *chap1* 文件的每行。

2. 要只显示不同的行，请输入：

```
sdiff -s -w 80 chap1.bak chap1
```

**sdiff** 命令在工作站上显示差别。**-w 80** 标志和变量将页宽设置为 80 列。**-s** 标志表示不显示两个文件中相同的行。

3. 要有选择地组合两个文件的部分，请输入：

```
sdiff -s -w 80 -o chap1.combo chap1.bak chap1
```

**sdiff** 命令将 *chap1.bak* 和 *chap1* 文件组合成名为 *chap1.combo* 的新文件。对于每组不同的行，**sdiff** 命令会提示您保持哪个组或是否想使用 **ed** 命令来对其进行编辑。

4. 要组合和编辑两个文件（*staff.jan* 和 *staff.apr*），并将结果写到 *staff.year* 文件中，请执行标明的步骤。

*staff.jan* 文件包含以下行：

```
会计部门成员
Andrea
George
Karen
Sam
Thomas
```

staff.apr 文件包含以下行:

```
会计部门成员
Andrea
Fred
Mark
Sam
Wendy
```

a. 请输入以下命令:

```
sdiff -o staff.year staff.jan staff.apr
```

**sdiff** 命令会开始比较 staff.jan 和 staff.apr 文件的内容, 并将结果写到 staff.year 文件中。**sdiff** 命令显示以下内容:

```
会计部门成员      会计部门成员
Andrea              Andrea
George              | Fred
%
```

% (百分号) 是命令提示符。

b. 输入 **e b** 子命令以开始使用 **ed** 命令编辑输出文件。

**sdiff** 命令显示数字序列, 它表示合并的行的字节数。在这种情况下, 字节数是 23。

c. 输入 **q** 子命令以退出 **ed** 命令, 并继续组合和编辑这两个文件。**sdiff** 命令显示以下内容:

```
Sam                Sam
Thomas             | Wendy
```

d. 请再次输入 **e b** 子命令。每当两个原始文件的一组行合并成输出文件时, 必须运行 **ed** 命令。在本例中的字节数是 13。

e. 请输入 **q** 子命令来保存更改。当将两个文件的所有行都合并成输出文件时, **q** 子命令退出 **ed** 和 **sdiff** 命令。

现在 staff.year 文件包含以下内容:

```
会计部门成员
Andrea
George
Karen
Fred
Mark
Sam
Thomas
Wendy
```

## 文件

`/usr/bin/sdiff` 包含 **sdiff** 命令。

## 相关信息

**diff** 命令、**ed** 命令。

《操作系统与设备管理》中的『文件』和『输入和输出重定向』。

---

## secdapclntd 守护程序

### 用途

提供和管理连接，并处理 LDAP 装入模块和“LDAP 安全信息服务器”之间的事务。

### 语法

```
/usr/sbin/secdapclntd [ -C CacheSize ] [ -p NumOfThread ] [ -t CacheTimeOut ] [ -T HeartBeatIntv ] [ -o ldapTimeOut ]
```

### 描述

**secdapclntd** 守护程序接受来自 LDAP 装入模块的请求，将该请求转发至“LDAP 安全信息服务器”，并将来自该服务器的结果发送回 LDAP 装入模块。该守护程序在其启动过程中读取 **/etc/security/ldap/ldap.cfg** 文件中所定义的配置信息，使用指定服务器专有名称和密码来认证到“LDAP 安全性信息服务器”，并在本地主机和服务器之间建立连接。

如果在 **/etc/security/ldap/ldap.cfg** 文件中指定了多个服务器，那么 **secdapclntd** 守护程序就连接到所有的服务器。然而，在特定的时间，它只跟这些服务器的其中之一进行谈话。服务器连接的优先级由它在服务器列表（最高优先级服务器在其中首先列出）中的位置确定。一旦当前与 **secdapclntd** 守护进程通信的服务器关闭，该守护进程就能发觉，并随之自动切换至另外的可用服务器。它还能检测服务器何时再次可用，并重新建立与该服务器的连接。如果重新连接的服务器具有比当前服务器更高的优先级，那么通信将切换至重新连接的服务器。通过 **secdapclntd** 守护程序在每个服务器上周期性地检测，可完成这种自动检测功能。后继检查之间的时间间隔缺省值为 300 秒，可以在守护程序启动时从带有 **-T** 选项的命令行为其进行更改，或者通过修改 **/etc/security/ldap/ldap.cfg** 文件中的 **heartbeatinterval** 值对其进行更改。

在启动时，**secdapclntd** 守护程序尝试建立与 LDAP 服务器的连接。如果它不能连接到任何服务器，那么就会休眠，并在 30 秒后重试。它重复这个过程两次，如果仍无法建立任何连接，那么就会退出 **secdapclntd** 守护程序进程。

**secdapclntd** 守护程序是多线程程序。守护程序所用的线程数的缺省值是 10。管理员可以通过调整守护程序所用的线程数来微调系统性能。

**secdapclntd** 守护程序高速缓存从“LDAP 安全信息服务器”上检索到的信息用于性能用途。如果能在高速缓存中查找到请求的数据，并且高速缓存条目没有失效，那么高速缓存中的数据就会返回给请求者。否则，**secdapclntd** 守护程序向“LDAP 安全信息服务器”发出请求以得到此信息。

用户的高速缓存条目的有效数值是 100-10,000，组的高速缓存条目的有效数值是 10-1,000。用户的缺省值是 1000 个条目，而组则是 100 个条目。

高速缓存超时或 TTL（生存时间）可以从 60 秒到 1 小时（60\*60=3600 秒）。缺省情况下，高速缓存条目在 300 秒后失效。如果将高速缓存超时设置为 0，那么也就禁用了高速缓存功能。

通过使用异步方法来执行 **secdapclntd** 守护程序和 LDAP 服务器之间的通信。它允许守护程序从服务器请求信息，然后在等待请求返回的过程中执行其他步骤。管理员可以配置客户机将等待服务器响应的的时间长度，时间长度缺省值为 60 秒。

当连接到 LDAP 服务器时，**secdapclntd** 守护程序需要进行主机查找。**nis\_ldap** 解析器可能导致查找被传递回守护程序本身，结果出现挂起的情况。要避免该问题，**secdapclntd** 守护程序忽略名称解析的系统指令。相反，它使用 **/etc/security/ldap/ldap.cfg** 文件中的 **nsorder** 属性定义的指令。

## 标志

注：缺省情况下，**secdapclntd** 守护程序在启动时读取 **/etc/security/ldap/ldap.cfg** 文件中指定的配置信息。如果在启动 **secdapclntd** 进程时命令行上给出了以下选项，那么命令行的选项会覆盖 **/etc/security/ldap/ldap.cfg** 文件中的值。

<b>-C</b> <i>CacheSize</i>	将 <b>secdapclntd</b> 守护程序所用的最大的高速缓存条目设置为 <i>CacheSize</i> 条目的数字。用户高速缓存的有效范围是 100-10,000 个条目。缺省值是 1000。组高速缓存项是用户高速缓存项的 10%。
<b>-o</b> <i>ldapTimeOut</i>	LDAP 客户机向服务器请求的超时周期（以秒计）。该值确定客户机将等待 LDAP 服务器响应的的时间长度。有效范围为 0 - 3600（1 小时）。缺省值为 60 秒。将该值设置为 0 来禁用超时并迫使客户机无限期等待。
<b>-p</b> <i>NumOfThread</i>	将 <b>secdapclntd</b> 守护程序所用的线程数设置为 <i>NumOfThread</i> 个线程。有效范围是 1-1000。缺省值是 10。
<b>-t</b> <i>CacheTimeout</i>	将高速缓存设置为在 <i>CacheTimeout</i> 秒后到期。有效范围是 60-3600 秒。缺省值是 300 秒。
<b>-T</b> <i>HeartBeatIntv</i>	设置此客户机与 LDAP 服务器之间脉动信号的时间间隔。有效值是 60-3,600 秒。缺省值是 300。

## 示例

1. 要启动 **secdapclntd** 守护程序，请输入：

```
/usr/sbin/secdapclntd
```

2. 要通过使用 20 个线程和 600 秒高速缓存超时值来启动 **secdapclntd**，请输入：

```
/usr/sbin/secdapclntd -p 20 -t 600
```

建议使用 **start-secdapclntd** 命令来启动 **secdapclntd** 守护程序。还建议在 **/etc/security/ldap/ldap.cfg** 文件中指定配置值，而不使用命令行标志，这样每次启动 **secdapclntd** 进程时都将使用这些值。

## 相关信息

**mksecdap**、**start-secdapclntd**、**stop-secdapclntd**、**restart-secdapclntd**、**ls-secdapclntd** 和 **flush-secdapclntd** 命令。

**/etc/security/ldap/ldap.cfg** 文件。

---

## secdifconv 命令

### 用途

将 LDIF 的用户和组条目从一种模式类型转换为另一种模式类型。

### 语法

```
secdifconv [-R load_module ] -S schematype -i inputFile [ -r ]
```

## 描述

**secldifconv** 命令读取 **-i** 选项指定的 **ldif** 格式化输入文件，使用 **-S** 选项指定的模式类型转换用户和组数据，并将结果显示到标准输出。如果将结果重定向到文件，那么可以使用 **ldapadd** 命令或 **ldif2db** 命令将结果添加到 LDAP 服务器。

**-S** 选项指定用于 **ldif** 输出的转换模式类型。**secldifconv** 命令接受以下模式类型：

- **AIX** - AIX 模式 (**aixaccount** 和 **aixaccessgroup** 对象类)
- **RFC2307** - RFC 2307 模式 (**posixaccount**、**shadowaccount** 和 **posixgroup** 对象类)
- **RFC2307AIX** - 具有完全 AIX 支持的 RFC 2307 模式 (**posixaccount**、**shadowaccount** 和 **posixgroup** 对象类，以及 **aixauxaccount** 和 **aixauxgroup** 对象类)。

用 **-i** 选项指定的输入文件可以包括以上任何受支持模式中的条目。**secldifconv** 命令将根据 **/etc/security/ldap/\*.map** 文件中定义的属性映射转换相应模式类型的用户和组条目。将只转换用户和组条目，其他条目输出时未经更改。

使用 **-r** 选项允许除去未包括在指定输出模式中的用户和组条目中的属性。如果未指定该选项，则假定无法识别的属性有效并且在输出时未经更改。请注意，如果在 **secldifconv** 从其转换的模式（而不是请求转换到其中的模式）中定义了用户或组属性，那么将不输出属性。此行为使得 **AIX** 模式和 **RFC2307AIX** 模式能够与包含属性子集的 **RFC2307** 模式进行转换。

在 AIX 5.3 之前的发行版中，当使用 **AIX** 模式在 LDAP 中存储条目时，也存储了不带有加密前缀（即，**db2ldif** 命令用来生成 **secldifconv** 的输入文件，不带加密前缀的密码以 **{IMASK}** 格式输出。要将 **{imask}** 格式转换为正确的 **{crypt}** 格式，应该使用 **-R** 选项来指定可装入的 I&A 模块以读取从 **AIX** 模式类型转换的密码，并假定先前已经将系统配置为 LDAP 客户机。

使用 **secldifconv** 命令输出将用户和组从其他系统添加到 LDAP 服务器时，应当小心。**ldapadd** 和 **ldif2db** 命令在添加条目时只检查条目名称（用户名或组名），而不检查数字标识。使用 **secldifconv** 输出以合并多台服务器的用户和组可能导致多个帐户共享一个数字标识，这是安全性违规。请注意，IBM® Directory Server 5.2 以及以后的版本都支持可用来避免此问题的唯一属性功能。

## 标志

<b>-R</b> <i>load_module</i>	指定用来在需要时检索用户密码的可装入的 I&A 模块。
<b>-S</b> <i>schematype</i>	指定输出 LDAP 模式类型。有效值为 <b>AIX</b> 、 <b>RFC2307</b> 和 <b>RFC2307AIX</b> 。
<b>-i</b> <i>inputFile</i>	指定包含要转换的用户和组数据的 <b>ldif</b> 格式的输入文件。
<b>-r</b>	指定除去未在指定模式类型中定义的任何属性。

## 退出状态

该命令返回以下退出值：

<b>0</b>	命令成功完成。
<b>&gt;0</b>	发生错误。
<b>-1</b>	内存故障（即，内存分配故障）。

## 示例

1. 要将 ldif 格式文件中的条目转换为 rfc2307 模式，请输入以下内容：

```
secldifconv -S rfc2307 -i input.ldif
```

它使转换的文件显示为 ldif 格式的标准输出。用户条目和组条目转换为 **rfc2307** 模式类型。

2. 要将 ldif 格式文件中的条目转换为 rfc2307aix 模式并除去无法识别的属性，请输入以下内容：

```
secldifconv -R LDAP -S rfc2307aix -i input.ldif -r > convert.ldif
```

它会将命令的输出内容发送到 ldif 格式的 convert.ldif 文件。在转换过程中除去了无法识别的属性，如果需要，将从 LDAP 模块请求用户密码。

## 位置

**/usr/sbin/secldifconv**

## 文件

方式	文件
r	<b>/etc/security/ldap/2307aixgroup.map</b>
r	<b>/etc/security/ldap/2307aixuser.map</b>
r	<b>/etc/security/ldap/2307group.map</b>
r	<b>/etc/security/ldap/2307user.map</b>
r	<b>/etc/security/ldap/aixgroup.map</b>
r	<b>/etc/security/ldap/aixuser.map</b>

## 相关信息

db2ldif、ldif2db 和 ldapadd 命令。

*AIX 5L Version 5.3 Files Reference* 中的 LDAP Attribute Mapping File Format。

---

## sectoldif 命令

### 用途

以 ldif 格式将本地定义的用户和组显示到标准输出。

### 语法

```
sectoldif -d baseDN [ -S schematype ] [ -u username ]
```

### 描述

**sectoldif** 命令读取本地定义的用户和组，并以 ldif 格式将结果显示到标准输出中。如果将结果重定向到文件，那么可以使用 **ldapadd** 命令或 **ldif2db** 命令将结果添加到 LDAP 服务器。

**-S** 选项指定用于 ldif 输出的模式类型。 **sectoldif** 命令接受三种模式类型：

- **AIX** - AIX 模式 (**aixaccount** 和 **aixaccessgroup** 对象类)
- **RFC2307** - RFC 2307 模式 (**posixaccount**、**shadowaccount** 和 **posixgroup** 对象类)

- **RFC2307AIX** - 具有完全 AIX 支持的 RFC 2307 模式 (**posixaccount**、**shadowaccount** 和 **posixgroup** 对象类, 以及 **aixauxaccount** 和 **aixauxgroup** 对象类)。

在 LDAP 服务器设置过程中, **sectoldif** 命令由 **mksecldap** 命令调用以导出用户和组。使用 **sectoldif** 输出将附加用户和组从其他系统导出到 LDAP 服务器时, 需要特别谨慎。**ldapadd** 和 **ldif2db** 命令在添加条目时只检查条目名称 (用户名或组名), 而不检查数字标识。使用 **sectoldif** 输出从多个系统导出用户和组可能导致多个帐户共享一个数字标识, 这是安全性违规。

**sectoldif** 命令读取 **/etc/security/ldap/sectoldif.cfg** 文件来确定数据将导出到的用户、组和系统子树的名称。**sectoldif** 命令仅将数据导出为 USER、GROUP 和 SYSTEM 类型。将使用文件中的指定名称在用 **-d** 标志指定的基础 DN 下创建子树。有关更多信息, 请参考 **/etc/security/ldap/sectoldif.cfg** 文件文档。

## 标志

- d baseDN** 指定放置用户和组数据的基本 DN。
- S schematype** 指定用于代表 LDAP 服务器中的用户 / 组条目的 LDAP 模式。有效值为 AIX、RFC2307 和 RFC2307AIX。缺省值为 AIX。
- u username** 指定显示特定的用户。

## 示例

1. 要显示所有本地定义的用户和组, 请输入以下内容:

```
sectoldif -d cn=aixsecdb,cn=aixdata -S rfc2307aix
```

这会以 **ldif** 格式将所有本地定义的用户和组显示到**标准输出**。使用 **rfc2307aix** 模式类型表示用户条目和组条目。将基本 DN 设置为 **cn=aixsecdb, cn=aixdata**。

2. 要只显示本地定义的用户占位符, 请输入以下内容:

```
sectoldif -d cn=aixsecdb,cn=aixdata -u foo
```

这会将本地定义的用户占位符以 **ldif** 格式显示到**标准输出**。如果不使用 **-S** 选项, 则使用缺省的 AIX 模式类型来表示 **foo** 的 **ldif** 输出。

3. 要以与 AIX 4.3 和 AIX 5.1 客户机兼容的格式导出数据, 请执行以下操作:

- a. 编辑 **/etc/security/ldap/sectoldif.cfg** 文件以包括以下条目:

```
USER          ou      organizationalUnit    aixuser
GROUP         ou      organizationalUnit    aixgroup
ID            ou      organizationalUnit    System
```

- b. 调用如下 **sectoldif** 命令:

```
sectoldif -d cn=aixsecdb,cn=aixdata -S aix
```

它会将本地定义的所有用户和组显示到 **ldif** 格式的标准输出。通过使用 **aix** 模式类型表示用户条目和组条目。将基础 DN 设置为带有用户子树 **ou=aixuser** 和组子树 **ou=aixgroup** 的 **cn=aixsecdb, cn=aixdata**。

## 文件

- |    |                             |
|----|-----------------------------|
| 方式 | 文件                          |
| r  | <b>/etc/passwd</b>          |
| r  | <b>/etc/group</b>           |
| r  | <b>/etc/security/passwd</b> |

方式	文件
r	/etc/security/limits
r	/etc/security/user
r	/etc/security/environ
r	/etc/security/user.roles
r	/etc/security/lastlog
r	/etc/security/smitacl.user
r	/etc/security/mac_user
r	/etc/security/group
r	/etc/security/smitacl.group
r	/etc/security/login.cfg

## 相关信息

`mksecdap` 和 `nistoldif` 命令。

`/etc/security/ldap/sectoldif.cfg` 文件。

---

## securetcpip 命令

### 用途

启用操作系统网络安全性功能。

### 语法

`securetcpip`

### 描述

`securetcpip` 命令为网络提供增强的安全性。该命令执行以下内容：

1. 运行 `tcback -a` 命令，此命令禁用不可信的命令和守护程序：`rcp`、`rlogin`、`rlogind`、`rsh`、`rshd`、`tftp` 和 `tftpd`。不删除禁用的命令和守护程序；而是将其更改为方式 `0000`。您可以通过重新建立有效的方式来启用特定的命令或守护程序。
2. 将 TCP/IP 安全节添加到 `/etc/security/sysck.cfg` 文件。该节具有以下格式：

```
tcip:
  netrc = ftp,rexec      /* functions disabling netrc */
```

在运行 `securetcpip` 命令之前，通过作为 `root` 用户登录并执行 `killall` 命令停止所有的网络守护程序来默许系统。

**注：**`killall` 命令杀死除调用进程之外的所有进程。如果已经登录或正在运行应用程序，那么在执行 `killall` 命令之前退出或完成这些应用程序。

发出 `securetcpip` 命令之后，关闭并重新启动系统。系统重新启动之后，已正确地配置了所有的 TCP/IP 命令和网络接口。

### 文件

`/etc/security/config` 包含安全系统的信息。



## 相关信息

**killall** 命令、**tcbck** 命令。

**.netrc** 文件格式。

《安全性》中的 Trusted Processes。

---

## sed 命令

### 用途

流编辑器。

### 语法

**sed** [ **-n** ] *Script* [ *File* ... ]

**sed** [ **-n** ] [ **-e** *Script* ] ... [ **-f** *ScriptFile* ] ... [ *File* ... ]

### 描述

**sed** 命令根据编辑脚本修改来自指定的 *File* 参数的行，并将其写到标准输出。**sed** 命令包含很多功能，用于选择要修改的行，并只对选择的行作更改。

**sed** 命令使用两个工作空间来保留修改的行：保留选定行的模式空间和暂时存储行的保留空间。

编辑脚本由单独的子命令构成，每个单独的行上有一个子命令。**sed** 子命令的一般格式如下：

```
[address-range] function[modifiers]
```

**sed** 命令通过将输入行读入方式空间，依次应用所有的 **sed** 子命令（这些子命令的地址选择了该行），并将方式空间写到标准输出来处理每个输入的 *File* 参数。然后清除方式空间，并对在输入的 *File* 参数中指定的每行重复该过程。一些 **sed** 子命令使用保留空间来保存后继检索的所有或部分模式空间。

当命令包含地址（行号或搜索模式）时，该命令只会对被寻址的行起作用。否则，该命令适用于所有的行。

地址是十进制行号、\$（美元符号），它表示输入的最后行的地址或上下文地址。上下文地址是与 **ed** 命令中所使用的表达式相似的正则表达式，除了具有以下的差别之外：

- 您可以选择模式的字符定界符。表达式的一般格式是：

```
\?pattern?
```

其中？（问号）是可选择的字符定界符。除了空格或换行符，您可以从当前的语言环境中选择任何字符。仅在第一次出现？时要求 \（反斜杠）字符（问号）。

模式的缺省格式如下：

```
/pattern/
```

\（反斜杠）字符不是必需的。

- \n 序列同模式空间中的换行字符相匹配（终止换行字符除外）。
- .（句点）与任何字符相匹配（终止换行字符除外）。也就是说，与 **ed** 命令不同（**ed** 命令不能匹配行中间的换行符），**sed** 命令能够匹配模式空间中的换行符。

某些叫做 *addressed* 的命令允许您指定该命令适用的一行或多行。下列的规则适用于寻址命令:

- 没有地址的命令行选择每一行。
- 具有以上下文格式表示的一个地址的命令行选择与该地址匹配的每一行。
- 具有以逗号分隔的两个地址的命令行选择从与第一个地址匹配的第一行到与第二个地址匹配的下一行之间的整个范围。(如果第二个地址小于或等于第一个选择的行号,那么只会选择一行。)此后就会重复这个过程以再次寻找第一个地址。

## 标志

<b>-e</b> <i>Script</i>	使用 <i>Script</i> 变量作为编辑脚本。如果您只使用一个 <b>-e</b> 标志并且不使用 <b>-f</b> 标志,则可以省略 <b>-e</b> 标志。
<b>-f</b> <i>ScriptFile</i>	使用 <i>ScriptFile</i> 变量作为编辑脚本源文件。 <i>ScriptFile</i> 变量是适用于 <i>File</i> 参数的一组准备好的编辑命令。
<b>-n</b>	禁止所有正常地写到标准输出的信息。

注: 您可以指定多个 **-e** 和 **-f** 标志。按指定的顺序将所有的子命令添加到脚本中,而不管其来源。

## sed 子命令

**sed** 命令包含下列的 **sed** 脚本子命令。子命令前圆括号内的数字表示子命令可允许的地址的最大数。

注:

1. 如果除了最后一行以外的所有行以 \ (反斜杠) 结束以引用换行字符,则带有 **a\**、**c\** 以及 **i\** 子命令的 *Text* 变量就可以超过一行。文本中的反斜杠被视为 **s** 命令的替换字符串中的反斜杠,并且能够用于保护初始的空格和制表符,以避免在处理每个脚本行时被去除。*RFile* 和 *WFile* 变量必须结束命令行,并且前面必须正好有一个空格。在开始处理前创建每个 *WFile* 变量。
2. **sed** 命令可以在模式文件中处理最多 999 个子命令。

(1) <b>a\</b>	
<i>Text</i>	在读取下一个输入行之前将 <i>Text</i> 变量放在输出中。
(2) <b>b[label]</b>	转移到具有 <i>label</i> 变量的: 命令。如果 <i>label</i> 变量为空,则转移到脚本的结尾。
(2) <b>c\</b>	
<i>Text</i>	删除模式空间。用 0 或 1 地址或 2 地址范围的结尾将 <i>Text</i> 变量放置在输出中,然后开始下一个周期。
(2) <b>d</b>	删除模式空间,然后开始下一个周期。
(2) <b>D</b>	删除通过第一个换行符的模式空间的初始段,然后开始下一个周期。
(2) <b>g</b>	用保留空间的内容替换模式空间的内容。
(2) <b>G</b>	将保留空间的内容附加到模式空间。
(2) <b>h</b>	用模式空间的内容替换保留空间的内容。
(2) <b>H</b>	将模式空间的内容附加到保留空间。
(1) <b>i\</b>	
<i>Text</i>	在将下一行读到模式空间之前将 <i>Text</i> 变量写到标准输出。
(2) <b>l</b>	将模式空间写到标准输出中,将不可显示字符以 4 位十六进制值显示。将长行折起来。
(2) <b>l</b>	以肉眼明确的格式将模式空间写到标准输出。将字符 \\, \\a, \\b, \\f, \\r, \\t 以及 \\v 以对应的转义序列写出。对于非显示字符,将其字符中的每个字节都以一个 3 位八进制数(前面具有反斜杠字符)写出(最重要的字节优先)。该格式也用于多字节字符。该子命令折叠长行。跟在换行符后面的反斜杠表示折叠点。折叠出现在第 72 列位置。\$ (美元符号) 标记每行的结束。
(2) <b>n</b>	如果不禁止缺省的输出,则将模式空间写到标准输出。它用输入的下一行替换模式空间。

- (2)**N** 使用嵌入式换行符（更改当前的行号）将输入的下一行附加到模式空间。您可以用它来搜索分割成两行的模式。
- (2)**p** 将模式空间写到标准输出。
- (2)**P** 删除通过第一个换行符的模式空间的初始段写到标准输出。
- (1)**q** 转移到脚本的结尾。它不启动新的周期。
- (2)**r** *RFile* 读取 *RFile* 变量的内容。在读取下一输入行之前将内容放置到输出。
- (2)**s***/pattern/replacement/flags* 用 *replacement* 字符串代替在模式空间中首次出现的 *pattern* 参数。除了空格或换行符，在 **s** 子命令之后显示的任何字符都能代替 */*（斜杠）分隔符。
- 请参阅 **ed** 命令的“模式匹配”部分。
- flags* 变量的值必须是零或大于零：
- g** 代替 *pattern* 参数的所有不重叠的实例，而不仅仅是第一个实例。
- n** 仅代替第 *n* 次出现的 *pattern* 参数。
- p** 如果进行了替代，则将模式空间写到标准输出。
- w** *WFile* 如果进行了替换，将模式空间写到 *WFile* 变量。将模式空间附加到 *WFile* 变量中。如果 **sed** 脚本的前一次写入还没有创建 *WFile* 变量，则 **sed** 命令就会创建它。
- (2)**t***label* 如果自从读取执行 **t** 子命令的输入行以来做了任何替代，则转移到该脚本文件中的 *:label* 变量。如果您不指定 *label* 变量，则控制转移到脚本的结尾。
- (2)**w***WFile* 将模式空间附加到 *WFile* 变量中。
- (2)**x** 交换模式空间和保留空间的内容。
- (2)**y***/pattern1/pattern2* 将所有 *pattern1* 变量中出现的字符替换为相应的 *pattern2* 字符。在 *pattern1* 和 *pattern2* 变量中的字符数必须相等。换行符用 **\n** 表示。
- (2)**!***sed-cmd* 只将指定的 **sed** 子命令应用于未被地址选择的行。
- (0):*label* 标记 **b** 和 **t** 子命令所引用的分支点。该标号可以是任何八个或更少字节的序列。
- (1)= 将当前行号作为一行写到标准输出。
- (2){*subcmd* } 将 {}（花括号）括起的子命令分组。
- (0) 忽略空命令。
- (0)# 如果 #（镑符）是出现在脚本文件的某行的第一个字符，则将整个行视为注释，但有一个例外。只对脚本文件的第一行来说，如果在 # 后的字符是 **n**，就会禁止缺省的输出。忽略 **#n** 之后的行的剩余部分。

## 退出状态

该命令返回以下退出值：

- 0** 成功完成。
- >0** 发生错误。

## 示例

- 要执行全局更改，请输入：

```
sed "s/happy/enchanted/g" chap1 >chap1.new
```

该命令序列用字 **enchanted** 替换每个在文件 **chap1** 中出现的字 **happy**。它将已编辑的版本写入名为 **chap1.new** 的单独文件中。**s** 子命令结尾的 **g** 字符指示 **sed** 命令在每行中尽可能多地进行替代。如果没有 **g** 字符，则 **sed** 命令只会替换行中首次出现的字 **happy**。

**sed** 命令作为过滤器操作。它从标准输入或命令行指定的文件中（本例中是 `chap1`）读取文本、修改文本、并将其写到标准输出。不同于大多数的编辑器，它不替换原始文件。这会使 **sed** 命令在用于管道时成为一个强大的命令。

2. 要在管道中将 **sed** 命令作为过滤器使用，请输入：

```
pr chap2 | sed "s/Page *[0-9]*$/(&)/" | enq
```

该命令序列在显示文件 `chap2` 之前将页码包含在圆括号中。**pr** 命令将标题和页码放置在每页的顶部，然后 **sed** 命令将页码放在圆括号中，**enq** 命令显示已编辑的列表。

**sed** 命令模式 `/Page *[0-9]*$/` 与在行尾出现的页码相匹配。**s** 子命令将其更改为 `(&)`，其中 `&` 代表匹配的页码。

3. 要显示文件的选定行，请输入：

```
sed -n "/food/p" chap3
```

**sed -n** 显示文件 `chap3` 中包含字 `food` 的每一行。通常，**sed** 命令将编辑过的每行复制到标准输出。**-n** 终止 **sed** 命令进行该操作。然后，您使用子命令（如 **p**）写入文本的特定部分。如果没有 **-n** 标志，该示例会显示文件 `chap3` 中的所有行，并显示两次包含 `food` 的每个行。

4. 要执行复杂的编辑，请输入：

```
sed -f script.sed chap4 >chap4.new
```

当您想做任何复杂的编辑时，该命令序列创建一个 **sed** 脚本文件。然后，您可以在使用该脚本之前对其进行测试和修改。您也可以重用脚本来编辑其他文件。用交互式文本编辑器创建脚本文件。

5. 样本 **sed** 脚本文件：

```
:join
/\$/ {N
s/\\n//
b join
}
```

**sed** 脚本将每个以 `\`（反斜杠）结束的行连接到其后的行上。首先，模式 `/\$/` 选择以 `\` 结束的行，其用于分组以 `{}`（花括号）括起的命令。然后，**N** 子命令附加下一行，嵌入换行符。`s/\\n//` 删除 `\` 和嵌入的换行符。最后，`b join` 转移回到标号 `:join` 以在新连接的行末检查 `\`。如果没有转移，那么 **sed** 命令在检查第二个 `\` 之前写入连接的行并读取下一行。

**注：**如果没有更多的输入行（也就是说，如果 **N** 子命令读取文件结束符），**N** 子命令将导致 **sed** 命令立刻停止。在停止前它不将模式空间复制到标准输出。这意味着如果输入的最后行结束于 `\`，就不将其复制到输出。

6. 要将现有的文件（`oldfile`）复制到新文件（`newfile`），并将所有出现的 `testpattern` 文本字符串替换为 `$REPL shell` 变量的内容，请输入：

```
cat oldfile | sed -e "s/testpattern/$REPL/g" > newfile
```

7. 要替换输入文件中出现的所有带有 `a` 的 `A`、带有 `b` 的 `B`、带有 `c` 的 `C` 和出现的所有带有字符 `Z` 的换行，请输入：

```
$ sed -f command.file input.file
```

其中 `command.file` 是脚本文件，`input.file` 是输入文件。

```
$cat command.file
y/ABC\n/abcZ/
```

另外，还可以执行以下命令来获取相同的功能：

```
sed "y/ABC\n/abcZ/" input.file
```

## 相关信息

**awk** 命令、**ed** 命令以及 **grep** 命令。

《AIX 5L V5.3 通用编程概念: 编写并调试程序》中的『用 sed 操作字符串』。

《AIX 5L V5.3 本地语言支持指南和参考大全》中的『本地语言支持』。

---

## sedmgr 命令

### 用途

显示和设置系统或可执行文件的“堆栈执行禁用”标志。

### 语法

```
sedmgr [-m {off | all | select | setidfiles}] [-o {on | off}] [-c {system | request | exempt}] {file_name | file_group} [-d {file_name | directory_name}] [-h]
```

### 描述

**sedmgr** 命令是“堆栈执行禁用”（SED）工具的管理器。您可以使用该命令启用和控制系统中完成的堆栈执行的级别。该命令还可以用来设置可执行文件中的各种标志，控制堆栈执行禁用。对系统范围方式设置的任何更改都只在系统重新引导后才会生效。

只有 root 用户可以修改系统范围的设置。对单独可执行文件的其他设置和重新设置选项仅在用户对该文件具有写权限时才会成功。SED 工具仅在 AIX 64 位内核操作系统中可用。

如果调用不带有任何参数，则 **sedmgr** 命令将根据堆栈执行禁用环境显示当前设置。

有关更多信息，请参阅《安全性》中的 Login control 中有关 *Stack Execution Disable* 的部分。

# 标志

-c

在可执行文件的头中设置或重新设置“request”和“exempt” SED 标志。另外，在 *file\_group* 中的所有可执行文件的头中设置或重新设置 SED 请求和免除检查标志。此选项需要对文件的写特权，或者，在指定了 *file\_group* 时需要引导特权。

可能的值如下：

## system

如果文件在可执行文件的头中具有系统标志，则操作系统将根据系统范围 SED 标志决定进程的操作。当文件未指定任何标志时，操作系统也根据系统范围 SED 标志决定进程的操作。

## exempt

在可执行文件头中设置一个标志表明该文件执行基于堆栈/头的操作，因此需要 SED 机制的免除。SED 请求检查位关闭。

## request

在可执行文件头设置一个标志表明该文件不执行任何基于堆栈/数据区域的操作，因此具有 SED 功能。SED 免除检查位关闭。

您可以指定表示一组文件的文件组，例如 TCB 文件。如果指定的文件名字符串不能确定文件，则假定该字符串确定 *file\_group*。当前仅定义了 **TCB\_files** 文件组。您可以为 32 位和 64 位可执行文件设置或重新设置 SED 请求和免除标志。**-c** 标志不能与 **-m**、**-o** 和 **-d** 标志一起使用。

-d

显示可执行文件的 SED 请求和免除检查标志。SED 请求和免除标志在可执行文件的文件头中。如果指定了目录，则该目录及其子目录下的所有可执行文件都将与它们的 SED 相关标志一起显示。该标志需要对 *file\_name* 或 *directory\_name* 的读特权。**-d** 标志不能与 **-m**、**-o** 和 **-c** 标志一起使用。

-h

显示 **sedmgr** 命令的语法。

**-m**

如果处理器支持 SED，则设置系统范围堆栈执行禁用方式。对系统范围设置的任何更改都需要重新引导系统后才能生效。该选项接受以下值之一：

- all** 对请求（标记为）免除的文件之外的所有文件强制执行堆栈执行禁用。
- off** 关闭系统上的堆栈执行禁用功能。
- select** 设置操作方式选择一组将被启用和被监视堆栈执行禁用的进程。仅选择文件头中设置了“request” SED 标志的文件中的进程。

#### setidfiles

设置操作方式以使操作系统为设置了“request” SED 标志的文件执行 SED，并且为具有以下特征的可执行文件启用 SED：

- 属于 root 用户的 **setuid** 文件。
- 带有如“system”或“security”主组的 **setid** 文件。

已配置的 SED 属性在下一 64 位内核引导时有效。由于 ODM 中的 SED 属性不影响 32 位内核，SED 监视标志在这种情况下是关闭的。如果处理器不支持 SED，则 **sedmgr** 命令带 **-m** 标志执行时返回错误。**-m** 标志不能与 **-c** 和 **-d** 标志一起使用。

**-o**

该选项启用 SED 在发生异常时监视进程而不要终止进程。该选项允许评估可执行文件是否正在操作任何合法的堆栈执行。该设置与使用 **-c** 选项设置的系统范围方式一起工作。SED 监视控制标志是 ODM 中存储的系统范围 SED 设置的一部分。更改该设置需要 root 特权。该标志可能的值如下：

- on** 打开 SED 工具的监视功能。在该方式下操作时，即使出现与 SED 相关的异常，系统也将允许进程继续操作。操作系统不会终止进程，而是将异常记录在 AIX 错误日志子系统中。
- off** 关闭 SED 工具的监视方式。在该方式下，操作系统终止任何违反和导致 SED 工具异常的进程。

已配置的 SED 属性在下一 64 位内核引导时有效。由于 ODM 中的 SED 属性不影响 32 位内核，SED 监视标志在这种情况下是关闭的。如果处理器不支持 SED，则 **sedmgr** 命令带 **-m** 标志执行时返回错误。**-o** 标志不能与 **-c** 和 **-d** 标志一起使用。

无

如果没有指定标志，则 **sedmgr** 命令根据堆栈执行禁用环境显示当前设置。它显示内核 **var** 结构中的当前 SED 设置和 ODM 中的系统范围 SED 设置。

## 参数

*file\_name*

*file\_group*

*directory\_name*

更改了 SED 设置的可执行文件的名称。需要写特权。

在没有指定文件名时，指更改了 SED 设置的可执行文件的组。需要引导特权。

可执行文件的目录和将 SED 检查标志与 **-d** 标志一起显示的可执行文件的任何子目录。

## 退出状态

0  
255

命令成功完成。  
发生错误。

## 安全性

访问控制：此命令应该是标准用户命令并且具有可信计算库属性。

## 示例

1. 要将系统范围的 SED 方式标志更改为 **setidfiles** 并且打开 SED 控制标志，请输入：

```
sedmgr -m setidfiles -o on
```

2. 要将 **plans** 文件的 SED 检查标志更改为 **exempt**，请输入：

```
sedmgr -c exempt plans
```

3. 要将标注为 TCB 文件的所有可执行文件的 SED 检查标志更改为 **select**，请输入：

```
sedmgr -c request TCB_files
```

4. 要显示 **plans** 文件的 SED 检查标志，请输入：

```
sedmgr -d plans
```

## 限制

审计事件：如果审计子系统已正确配置并启用，则 **sedmgr** 命令生成以下审计记录（事件）：

事件	信息
SEDMGR_Odm	系统范围的 SED 设置。
SEDMGR_File	可执行文件头中的 SED 设置。

关于如何对审计事件进行正确选择和分组，以及如何配置审计事件数据收集的更多详细信息，请参阅《安全性》中的 Auditing overview 这一节中的 Setting up auditing。

## 位置

**/usr/sbin/sedmgr**

## 文件

**/usr/bin/tcbck**  
**/usr/bin/ldedit**

以可执行方式访问。  
以可执行方式访问。

## 相关信息

**ldedit** 命令，第 313 页的『tcbck 命令』。

《安全性》中 Auditing overview 和 Login control 中的 Stack Execution Disable(SED) Protection 部分。



---

## send 命令

### 用途

发送消息。

### 语法

```
send [ File ... | { -draft | -nodraftfolder | -draftfolder +Folder | -draftmessage Message } ] [ -alias File ] [ -format | -noformat ] [ -nomsgid | -msgid ] [ -nofilter | -filter File ] [ -nopush | -push ] [ -forward | -noforward ] [ -noverbose | -verbose ] [ -nowatch | -watch ]
```

### 描述

**send** 命令通过邮件传送系统发送消息。如果发送失败，则 **send** 命令显示一条错误消息。缺省情况下，From: 与 Date: 字段添加到每条指定的消息中。除非有 **\$SIGNATURE** 环境变量或 signature: 概要文件条目存在，**send** 命令将发送者地址放入 From: 字段中。

**send** 命令将当前日期写到 Date: 字段中。如果 **dist** 命令调用 **send** 命令，那么 **send** 命令将 Resent- 添加到 From:、Date: 以及 Message-ID: 字段中。

发送成功之后，**send** 命令通过重命名活动状态下的消息来除去消息。系统通过在当前消息号前放置 , (逗号) 来重命名消息。非活动文件不能使用“消息处理程序” (MH) 包。然而，系统命令仍然可以操作非活动文件。再次使用 **send** 命令之前，仍可以检索非活动文件。

### 标志

<b>-alias</b> <i>File</i>	指定要搜索的邮件别名文件。使用“消息处理程序” (MH) 别名需要三个“消息处理程序” (MH) 概要文件条目: <pre>ali: -alias Aliases send: -alias Aliases whom: -aliasAliases</pre>
<b>-draft</b>	其中 Aliases 是要被搜索的文件。缺省别名文件是 <i>/etc/mh/MailAliases</i> 。如果没有指定文件，则使用当前草稿消息。没有此标志而且没有指定具体文件时， <b>send</b> 命令会询问用户是否使用当前的草稿消息。
<b>-draftfolder</b> <i>+Folder</i>	指定包含要发送的草稿消息的草稿文件夹。带有 <i>Message</i> 参数的 <b>-draftfolder</b> <i>+Folder</i> 标志和指定 <b>-draftmessage</b> 标志相同。
<b>-draftmessage</b> <i>Message</i>	指定要发送的消息。您可以使用以下消息参考值之一作为 <i>Message</i> 参数的值: <i>Number</i> 消息号。 <b>cur</b> 或 . (句点) 当前消息。这是缺省值。 <b>first</b> 文件夹中的第一条消息。 <b>last</b> 文件夹中的最后一条消息。 <b>next</b> 当前消息的下一条消息。 <b>prev</b> 当前消息的上一条消息。
<b>-filter</b> <i>File</i>	使用指定文件中的格式化指令对发送到列在 Bcc: 字段中的收件人的消息副本重定格式。
<b>-format</b>	以标准格式将所有的收件人地址提交给交付传送系统。此标志是缺省值。

<b>-forward</b>	如果 <b>send</b> 命令不能成功发送草稿，则将失败消息添加到草稿消息中并返回给发送者。此标志是缺省值。
<b>-help</b>	列出命令语法、可用的参数选项（开关）和版本信息。 <b>注：</b> 对于“消息处理程序”（MH），必须完整拼写此标志的名称。
<b>-msgid</b>	将消息标识组件（例如 Message-ID:）添加到消息中。
<b>-nodrafftfolder</b>	撤销最后一次出现的 <b>-drafftfolder + Folder</b> 标志。此标志是缺省值。
<b>-nofilter</b>	将 Bcc: 头字段从列出在 To: 与 cc: 字段内的收件人消息中除去。然后此标志将带有最小头部分的消息发送到 Bcc: 字段中的收件人。此标志是缺省值。
<b>-noformat</b>	防止更改接收方地址的格式。
<b>-noforward</b>	如果传送失败，则防止草稿消息返回发送方。
<b>-nomsgid</b>	阻止添加消息标识组件。此标志是缺省值。
<b>-nopush</b>	在前台运行 <b>send</b> 命令。此标志是缺省值。
<b>-noverbose</b>	在将消息传送给 <b>sendmail</b> 命令的过程中，防止显示信息。此标志是缺省值。
<b>-nowatch</b>	使用 <b>sendmail</b> 命令传送时，防止显示信息。此标志是缺省值。
<b>-push</b>	在后台运行 <b>send</b> 命令。如果发送失败，则 <b>send</b> 命令在终端不显示错误消息。使用 <b>-forward</b> 标志将没有发送的消息返回给您。
<b>-verbose</b>	在将消息传送给 <b>sendmail</b> 命令的过程中，显示信息。此信息允许监视发送邮件时所涉及到的步骤。
<b>-watch</b>	在使用 <b>sendmail</b> 命令传送消息的过程中，显示信息。此信息允许监视发送邮件时所涉及到的步骤。

## 概要文件条目

在 *UserMhDirectory/mh\_profile* 文件中输入以下条目：

Draft-Folder:	为草稿设置缺省文件夹。
mailproc:	指定用于公布失败通告的程序。
Path:	指定用户的“消息处理程序”目录。
postproc:	指定用于公布消息的程序。
Signature:	设置邮件签名。

## 示例

要发送 **\$HOME/Mail/draft** 文件中的草稿消息，请输入：

```
send
```

系统使用消息响应如下：

```
Use "/home/david/Mail/draft"?
```

如果输入 **yes**，则发送草稿消息，并返回到 shell 提示符。在本示例中，**\$HOME** 目录的名称是 **/home/david**。

## 文件

<b>\$HOME/mh_profile</b>	指定“消息处理程序”用户概要文件。
<b>/usr/bin/send</b>	包含 <b>send</b> 命令。

## 相关信息

**ali** 命令、**comp** 命令、**dist** 命令、**forw** 命令、**post** 命令、**sendmail** 命令和 **spost** 命令。

**.mh\_alias** 文件格式和 **.mh\_profile** 文件格式。

《网络与通信管理》中的『邮件应用程序』。

---

## sendbug 命令

### 用途

将系统错误报表邮寄到指定的地址。

### 语法

**sendbug** [ *Address* ]

### 描述

**sendbug** 命令是 shell 脚本，用来帮助用户以正确的格式撰写和邮寄错误报表。

**sendbug** 命令启动错误报表格式大纲中的 **EDITOR** 环境变量所指定的编辑器。缺省编辑器是 **vi**。

填写错误报表格式大纲的适当字段并退出编辑器。**sendbug** 命令将完成的报表邮寄到 *Address* 参数所指定的地址。缺省地址是 **POSTMASTER**。

### 文件

**/usr/lib/bugformat** 包含错误报表大纲。

### 相关信息

**bugfiler** 命令、**env** 命令、**sendmail** 命令。

《网络与通信管理》中的『邮件管理』。

---

## sendmail 命令

### 用途

为本地或网络交付传送邮件。

### 语法

**sendmail** [ **-ba** | **-bd** | **-bD** | **-bh** | **-bH** | **-bi** | **-bm** | **-bp** | **-bs** | **-bv** | **-bt** [ **-CFile** ] [ **-dValue** ] ] [ **-BType** ] [ **-FFullName** ] [ **-fName** ] [ **-hNumber** ] [ **-i** ] [ **-Mx Value** ] [ **-n** ] [ **-N Dsn** ] [ **-OOption=Value** ] [ **-o Option [ Value ]** ] [ **-pProtocol** ] [ **-q [ Time]** ] [ **-qISubstr** ] [ **-qRSubstr** ] [ **-qSSubstr** ] [ **-R Return** ] [ **-r addr** ] [ **-t** ] [ **-U** ] [ **-V Envid** ] ] [ **-v** ] [ **-X LogFile** ] *Address*

注: **-bd**、**-bi**、**-bp**、**-bt**、**-q [ Time]** 标志 和 *Address* 参数是可选的。

### 描述

注: 在 **sendmail V8.7** (AIX 4.2 及以后的版本) 中, 名称解析顺序是 **DNS**、**NIS**, 然后是本地。如果您想覆盖它, 则指定 **/etc/netsvc.conf** 文件或 **NSORDER** 环境变量。

**sendmail** 命令接收格式化文本消息并将它们发送到一个或更多的用户。在网络上使用时，**sendmail** 命令转换消息的头信息格式以匹配目的系统的要求。程序使用地址的语法和内容来确定目的系统的网络。

**sendmail** 命令可以将消息传送到：

- 本地系统用户
- 使用 TCP/IP 协议连接到本地系统的用户。
- 使用基本网络实用程序 (BNU) 命令协议连接到本地系统的用户。

**sendmail** 命令并不是用来作为一个用户界面例程；其他的命令提供用户友好的界面。使用 **sendmail** 命令只是用来传送预格式化的消息。

**sendmail** 命令从标准输入中读取消息文本。当读到消息字符的结尾（文件结束符 (Ctrl-D) 控制序列或者一行中的单一句点）时，**sendmail** 命令会将消息副本发送到所列的全部地址。

### **sendmail 邮件过滤 API (Milter)**

**sendmail** 邮件过滤 API 提供对正在处理的邮件消息的访问，这样第三方程序可以过滤元信息和内容。使用 **sendmail** 邮件过滤 API 开发的过滤器使用线程，所以可能需要更改过滤器中的每个进程限制。例如，如果将大量使用过滤器，则您可能想使用 **setrlimit()** 子例程来增加打开的文件描述符的数量。

**在 sendmail 配置中指定过滤器：** 使用关键字 **X**（表示外部）指定过滤器。以下是三个过滤器示例：

```
Xfilter1, S=local:/var/run/f1.sock, F=R
```

```
Xfilter2, S=inet6:999@localhost, F=T, T=C:10m;S:1s;R:1s;E:5m
```

```
Xfilter3, S=inet:3333@localhost
```

您可以在 `.mc` 文件中指定过滤器。以下过滤器与 `/var/run` 目录中的 UNIX<sup>®</sup> 域套接字连接：

```
INPUT_MAIL_FILTER(`filter1', `S=local:/var/run/f1.sock, F=R')
```

以下过滤器使用本地主机端口 999 上的 IPv6 套接字：

```
INPUT_MAIL_FILTER(`filter2', `S=inet6:999@localhost, F=T, T=C:10m;S:1s;R:1s;E:5m')
```

以下过滤器使用本地主机端口 3333 上的 IPv4 套接字：

```
INPUT_MAIL_FILTER(`filter3', `S=inet:3333@localhost')
```

#### **sendmail 邮件过滤标志：**

**R** 如果过滤器不可用，则拒绝连接。

**T** 如果过滤器不可用，则暂时断开连接。

如果未指定 `F=R` 和 `F=T`，则 **sendmail** 按照过滤器不存在的情况来传送消息。分隔符是逗号 (,)。

**sendmail 邮件过滤超时：** 您可以使用 `T=x` 覆盖缺省 **sendmail** 超时。 `T= statement` 中有四个字段：

**C** 连接到过滤器的超时（如果为 0，则使用系统超时）。

**S** 从 MTA 向过滤器发送信息的超时。

**R** 从过滤器读取回复的超时。

**E** 从向过滤器发送消息结束到等到最终应答之间的总超时时间。

每个项目之间的分隔符是分号 (;)。

缺省值为：

- T=C:0m;S:10s;R:10s;E:5m

InputMailFilters 选项确定调用哪些过滤器和这些过滤器如何排序:

```
InputMailFilters=filter1, filter2, filter3
```

这根据 .mc 文件中 INPUT\_MAIL\_FILTER 命令的顺序自动设置。您还可以通过设置 .mc 文件中的 confINPUT\_MAIL\_FILTERS 来重新设置该值。该选项以指定过滤器的顺序调用这三个过滤器。

您可以通过使用 .mc 文件中的 **MAIL\_FILTER()** (而不是 **INPUT\_MAIL\_FILTER()**) 来定义过滤器, 但是不会将过滤器添加到输入过滤器列表中。

注: 如果定义了 InputMailFilters, 则不使用过滤器。

## 使用配置文件

**sendmail** 命令使用配置文件 (缺省情况下为 **/etc/mail/sendmail.cf** 文件) 设置可操作的参数并且确定命令如何分析地址。此文件是文本文件, 您可以用其他的文本编辑器对它进行编辑。修改 **sendmail.cf** 以后, 请刷新 **sendmail** 守护程序。

**sendmail** 命令的当前进程标识存储在 **/etc/mail/sendmail.pid** 文件中。如下发出 **kill -15** 命令, 使 **sendmail** 命令重新读取新编辑的 **sendmail.cf**:

```
kill -15 `head -1 /etc/mail/sendmail.pid`
```

如果 **srcmstr** 命令正在运行, 则您可以发出如下的 **refresh** 命令来重新构建配置数据库、别名数据库和 NLS 数据库。

```
refresh-s sendmail
```

**sendmail** 命令重新读取这些数据库并且继续操作新数据。

## 定义别名

当 **sendmail** 命令处理本地邮件时, **sendmail** 命令允许您定义使用的别名。别名是备用名称, 您可用它来代替详细的网络地址。您也可以使用别名构建分发列表。

在 **/etc/mail/aliases** 文件中定义别名。此文件是可以编辑的文本文件。**sendmail** 命令使用此文件的数据库版本。必须在 **/etc/mail/aliases** 文件的任和更改生效之前, 运行 **sendmail -bi** 命令或 **newaliases** 命令来构建新的别名数据库。

现在, 在 AIX 5.1 for Sendmail 8.11.0 上提供了 Berkeley DB 支持。Sendmail 将继续读取 DBM 格式的别名直到别名数据库获得重建。一旦重建, sendmail 将读取 Berkeley DB 格式的别名并将它们存储在 **/etc/mail/aliases.db** 文件中。

注: 当在 **/etc/mail/aliases** 文件中定义别名时, 对嵌套的别名只能用小写字符。别名右手边的大写字符在存储到别名数据库之前转换为小写字母。在以下示例中, 由于存储第二行时 TEST 转换成 test, 因此发送到 testalias 的邮件失败。

```
TEST: user@machine  
testalias: TEST
```

每个系统必须有一个被指定为 **postmaster** 别名的用户或用户别名。**postmaster** 别名的缺省值是一个 root 文件。在 **/etc/mail/aliases** 文件中您可以将此别名指定给不同的用户。**postmaster** 别名允许您系统以外的其他用户将邮件发送到一个已知的标识并获取有关邮寄到您系统上的用户的信息。您系统上的用户也可以将问题通知发送到 **postmaster** 标识。

## 标志

- B***Type* 将正文主体类型设置为 *type*。当前合法值是 7BI 或 8BITMIME。  
注: **-b** 标志是互斥的
- ba** 以 ARPANET 方式启动 **sendmail** 命令。命令的所有输入行必须以回车符和换行符 (CR-LF) 结束。**sendmail** 命令生成以 CR-LF 结尾的消息, 并查看 From: 与 Sender: 字段来查找发件人的名字。
- bd** 启动 **sendmail** 命令使其像“简单邮件传输协议”(SMTP) 邮件路由器一样作为守护程序在后台运行。
- bD** 启动 **sendmail** 命令使其像“简单邮件传输协议”(SMTP) 邮件路由器一样作为守护程序在前台运行。
- bh** 显示持久主机状态数据库。
- bH** 清除持久主机状态数据库。
- bi** 根据 **/etc/mail/aliases** 文件中定义的信息构建别名数据库。用此标志运行 **sendmail** 命令和运行 **/usr/sbin/newaliases** 命令一样。
- bm** 以普通方式传送邮件。(这是缺省值。)
- bp** 显示邮件队列列表。用此标志运行 **sendmail** 命令和运行 **/usr/sbin/mailq** 命令一样。
- bs** 使用如 RFC821 中描述的简单邮件传输协议 (SMTP) 收集来自标准输入的邮件。此标志也包含所有与 SMTP 兼容的 **-ba** 标志的操作。
- bt** 以地址测试方式启动 **sendmail** 命令。此方式允许您输入交互式地址并在 **sendmail** 命令显示其分析地址所采取的步骤时进行监视。在测试方式提示符下, 输入一个规则集或用逗号分割的多个规则集并输入地址。使用此方式来调试新配置文件中的地址分析规则。
- bv** 用验证此命令 *Address* 参数字段所提供的用户标识符的请求来启动 **sendmail** 命令。**sendmail** 命令返回一条消息作为响应, 告诉哪些标识可由邮件程序解析。它并不尝试收集或传送消息。使用此模式验证用户标识、别名或邮递列表的格式。
- C***File* 用 *File* 变量指定的备用配置文件启动 **sendmail** 命令。在安装用作运行的配置文件的新配置文件之前, 和 **-bt** 一起使用此标识来测试该文件。
- d***Value* 将调试值设置为 *Value* 变量所指定的值。唯一的有效值是 21.*n*, 其中 *n* 是任意非零整数。它产生关于地址分析的信息并且通常和 **-bt** 标志一起使用。*n* 的更高值生成更多的详细信息。使用此标志需要引导许可权。
- F***FullName* 将发件人的全名设置为 *FullName* 变量所提供的字符串。
- f***Name* 设置发件人的名字 (邮件的信包发件人)。如果在初始提交过程中 From: 头丢失, 则在该头中仍可以使用这个地址。信包发件人的地址用作传送状态通知的接收方, 也可以出现在 Return-path: 头中。只有可信的用户 (一般是 root 用户、守护程序以及 uucp) 或者如果您试图成为的那个人就是您本人的话才能使用此标志。否则, X-Authentication-Warning 头将添加到消息中。
- h***Number* 将跳计数设置为 *Number* 变量所指定的值。跳计数是 SMTP 路由器 (不仅仅是 **sendmail** 命令的本地副本) 所处理消息的次数。每次消息处理后, 邮件路由器就增加跳计数。当它到达限制时, 就会返回一条错误消息以在防止邮件系统中的无限循环。
- i** 忽略进入消息的行中单独的点。如果您正在从文件读取数据的话, 必须设置它。
- L** 将系统日志消息中使用的标识设置为供给的标记。
- M***x Value* 将宏 *x* 设置为指定的 *Value*。
- N** *Dsn* 将传送状态通知条件设置为 DSN。传送状态通知条件可以是: never 表示没有通知或者用逗号分隔的值的列表, failure 表示发送失败的通知, delay 表示发送延迟的通知, 以及 success 表示消息成功发送的通知。
- n** 阻止 **sendmail** 命令解释别名。
- O** *Option=Value* 将 *Option* 设置为指定的 *Value*。使用长格式选项名称。此标志仅应用于 AIX 4.2 或者以后的版本。
- o***Option [ Value ]* 设置 *Option* 变量。如果选项是一个有值的选项, 您也必须为 *Value* 变量指定一个值。  
注: 对于有效值, 请查看《网络与通信管理》中的『sendmail.cf 文件中 sendmail 的命令选项』。

<b>-p</b> <i>Protocol</i>	设置发送协议。建议您设置此项。您可以以 <i>Protocol:Host</i> 格式设置发送协议和发送主机来设置 <i>Protocol</i> 。例如, <b>-pUUCP:uunet</b> 将发送协议设置为 UUCP 并将发送主机设置为 uunet。一些现有的程序使用 <b>-oM</b> 标志设置 <b>r</b> 和 <b>s</b> 宏, 它和使用 <b>-p</b> 标志是等效的。此标志仅适用于 4.2 版本或以后的版本。
<b>-qI</b> <i>Substr</i>	将进程作业限制为那些包含作为队列标识的 <i>Substr</i> 。
<b>-qR</b> <i>Substr</i>	将进程作业限制为那些包含作为收件人之一的子串 of <i>Substr</i> 。
<b>-qS</b> <i>Substr</i>	将进程作业限制为那些包含作为发件人的子串 of <i>Substr</i> 。
<b>-q</b> [ <i>Time</i> ]	在 <i>Time</i> 变量指定的间隔中处理队列中保存的消息。如果没有指定 <i>Time</i> 变量, 则该标志立即处理队列。
<b>-R</b> <i>Return</i>	设置如果消息弹回要返回的消息的量。 <i>Return</i> 参数可以为 full 以表示返回整个信息或者为 hdrs 以表示仅能返回头部分。
<b>-r</b> <i>addr</i>	<b>-f</b> 的废弃格式。
<b>-t</b>	将消息发送到消息头的 To:、Cc: 以及 Bcc: 字段中指定的收件人, 也可以发送到命令行中指定的任何用户。
<b>-U</b>	设置初始 (用户) 提交。当被诸如 Mail 或 exmh 用户代理调用时, 应当设置此项, 但是当被诸如 rmail 网络传送代理调用时切勿设置此项。
<b>-V</b> <i>Envid</i>	设置原始信包标识。它可以通过 SMTP 传播到支持 DSN 的服务器上, 并且以 DSN 兼容的错误消息返回。
<b>-v</b>	以详细的方式启动 <b>sendmail</b> 命令。 <b>sendmail</b> 命令显示有关传送状态和别名扩展的消息。
<b>-X</b> <i>LogFile</i>	为了调试邮件程序问题, 记录 <i>LogFile</i> 中 <b>sendmail</b> 的所有进出流量。由于它迅速地产生大量的数据, 因此尽可能不使用此标志。此标志仅适用于 4.2 版本或以后的版本。

您也可以设置或除去 **sendmail** 配置处理选项。负责邮件系统的人使用这些选项。为设置这些选项, 在命令行中使用 **-o** 标志或者使用配置文件 (*/etc/mail/sendmail.cf*) 中的 **O** 控制行。

## 示例

运行以下命令可显示 **sendmail** 版本:

```
echo \$$ | sendmail -d0
```

系统使用消息响应如下:

```
Version AIX5.2/8.11.6p2
Compiled with: LDAPMAP MAP_REGEX LOG MATCHGECOS MIME7T08 MIME8T07
              NAMED_BIND NDBM NETINET NETINET6 NETUNIX NEWDB NIS NISPLUS
              QUEUE SCANF SMTP USERDB XDEBUG
```

```
===== SYSTEM IDENTITY (after readcf) =====
(short domain name) $w = dodgers
(canonical domain name) $j = dodgers.usca.ibm.com
(subdomain name) $m = usca.ibm.com
(node name) $k = dodgers
=====
```

```
Recipient names must be specified
# oslevel -r
5200-02
#
```

## 安全性

审计事件:

事件	信息
<b>SENDMAIL_Config</b>	配置事件
<b>SENDMAIL_ToFile</b>	创建文件事件

## 退出状态

**sendmail** 命令返回退出状态值。这些返回值在 `/usr/include/sysexits.h` 文件中定义。下列表总结了这些返回值的含义：

<b>EX_CANTCREAT</b>	<b>sendmail</b> 命令不能创建用户指定的文件。
<b>EX_CONFIG</b>	在配置文件的格式中发现到错误。
<b>EX_DATAERR</b>	某种方式下输入的数据不正确。
<b>EX_IOERR</b>	在 I/O 过程中产生错误。
<b>EX_NOHOST</b>	<b>sendmail</b> 命令无法识别指定的主机名。
<b>EX_NOINPUT</b>	输入文件（非系统文件）不存在或者不可读。
<b>EX_NOPERM</b>	用户没有执行所请求的操作的许可权。
<b>EX_NOUSER</b>	<b>sendmail</b> 命令无法识别指定的用户标识。
<b>EX_OK</b>	<b>sendmail</b> 命令成功完成。
<b>EX_OSERR</b>	出现一个临时操作系统错误。这种错误的一个示例是创建一个新的进程失败。
<b>EX_OSFILE</b>	出现系统文件错误。例如，系统文件（例如 <code>/etc/passwd</code> ）不存在、无法打开或者有另一种类型的错误阻止使用它。
<b>EX_PROTOCOL</b>	在协议交换过程中，远程系统返回一些不正确的东西。
<b>EX_SOFTWARE</b>	出现一个内部软件错误（包含坏的参数）。
<b>EX_TEMPFAIL</b>	<b>sendmail</b> 命令无法创建与远程系统的连接。稍后重新尝试请求。
<b>EX_UNAVAILABLE</b>	<b>sendmail</b> 命令所需的服务或资源不可用。
<b>EX_USAGE</b>	命令语法不正确。

## 文件

<code>/usr/sbin/sendmail</code>	包含 <b>sendmail</b> 命令。
<code>/usr/sbin/mailq</code>	包含邮件队列。
<code>/usr/sbin/newaliases</code>	包含别名数据库。
<code>/usr/sbin/mailstats</code>	包含在 <code>/usr/lib/sendmail.st</code> 文件中查找到的统计信息。
<code>/etc/mail/aliases</code>	包含 <b>sendmail</b> 命令别名的文字版。
<code>/etc/mail/aliases.db</code>	包含 Berkeley DB 格式化的别名数据库。
<code>/etc/mail/aliases.dir</code>	包含 DBM 格式化的别名数据库。
<code>/etc/mail/aliases.pag</code>	包含 DBM 格式化的别名数据库。
<code>/etc/mail/sendmail.cf</code>	包含 <b>sendmail</b> 配置文件的文字版。
<code>/etc/sendmail.st</code>	包含邮件路由的统计信息。
<code>/usr/lib/smdemon.cleanu</code>	维护在 <code>/var/spool/mqueue</code> 目录下查找到的日志文件的老化副本。
<code>/var/spool/mqueue</code>	包含与邮件队列中的消息相关的临时文件和日志文件。
<code>/usr/bin/uux</code>	包含用于传送“基本联网实用程序”（BNU）的邮件程序命令。
<code>/usr/bin/bellmail</code>	包含用于传送本地邮件的邮件程序命令。

## 相关信息

**bellmail** 命令、**kill** 命令、**mail**、**Mail** 命令、**mailq** 命令、**mailstats** 命令、**newaliases** 命令、**refresh** 命令以及 **uux** 命令。

**srmcstr** 守护程序。

《网络与通信管理》中的『BNU 概述』、『邮件管理』和『传输控制协议 / 因特网协议』。

《网络与通信管理》中的『邮件应用程序』。



---

## setclock 命令

### 用途

为网络中的主机设置时间与日期。

### 语法

```
/usr/sbin/setclock [ TimeServer ]
```

### 描述

**/usr/sbin/setclock** 命令从网络时间服务器上获取时间，如果用户使用 root 用户权限运行，则相应地设置本地时间与日期。

**setclock** 命令从时间服务器获得第一次响应，然后转换在服务器上读取的日历时钟并且显示本地日期与时间。如果 root 用户运行 **setclock** 命令，则它能调用标准工作站条目点来设置系统的日期与时间。

如果没有时间服务器响应或者网络不可操作，则 **setclock** 命令会显示表示那种效果的消息并且保持系统的当前日期与时间设置不变。

注：任意运行 **inetd** 守护程序的主机都可以充当时间服务器。

### 参数

*TimeServer* 服务 TIME 请求的主机名称或网络主机地址。**setclock** 命令向时间服务器主机发送因特网 TIME 服务请求。如果省略 *TimeServer* 名称，则 **setclock** 命令会向缺省时间服务器发送请求。名称服务器指定 DOMAIN 环境下的缺省时间服务器。否则，**/etc/hosts** 文件指定缺省时间服务器。

### 示例

1. 要使用 **/etc/hosts** 文件指定的时间服务器主机来显示日期与时间，请输入：

```
setclock
Sat Mar 11 15:31:05 1988
```

**setclock** 命令显示正确的日期与时间。

2. 要设置日期与时间，请输入：

```
su root
setclock host1
Thu Jan 12 15:24:15 1990
```

在按照 host1 的时间服务器设置时间之前，必须使用 **su** 命令或以 root 用户登录。

### 相关信息

**timedc** 命令。

**inetd** 守护程序、**timed** 守护程序。

**hosts** 文件格式。

《网络与通信管理》中的『TCP/IP 守护程序』。

---

## setea 命令

### 用途

撰写或删除文件的指定扩展属性。

### 语法

```
setea -n Name [ -l ] { -v Value | -d | -f EAFile } FileName ...
```

### 描述

**setea** 命令撰写或删除文件的指定扩展属性。文件必须位于支持指定扩展属性的文件系统中，例如，使用 **v2** 扩展属性格式的 JFS2。

注：为了阻止命名冲突，JFS2 已保留了系统定义的扩展属性的 8 字符前缀 (0xf8)SYSTEM(0xF8)。避免使用该前缀来命名用户定义的扩展属性。

不使用此命令来设置 ACL。要设置 ACL，请使用 **aclput** 命令。

### 标志

<b>-d</b>	指定从文件中删除命名的扩展属性。
<b>-f EAFile</b>	<i>EAFile</i> 指定包含 EA 值的文件。如果 <i>FileName</i> 的同指定名称相匹配的扩展属性已经存在，则该值将更改为指定的值。
<b>-l</b>	指定从符号链接本身（而不是它指向的文件）撰写或删除扩展属性。
<b>-n Name</b>	指定要撰写的扩展属性的名称。
<b>-v Value</b>	为命名的扩展属性指定值。如果文件的同指定名称相匹配的扩展属性已经存在，则将该值更改为指定的值。值将作为字符串进行处理。如果它包含空格，则应为其加上引号。
<i>FileName ...</i>	指定从其中撰写或删除扩展属性的一个或多个文件。

### 退出状态

<b>0</b>	成功完成。
正整数	发生错误。

### 示例

- 要为文件 `design.html` 创建名为 `Approver` 且值为 `Grover` 的扩展属性。

```
setea -n Approver -v Grover design.html
```
- 要将文件 `design.html` 的名为 `Approver` 的扩展属性修改为 `Joon` 新值。

```
setea -n Approver -v Joon design.html
```
- 要从文件 `design.html` 除去名为 `Approver` 的扩展属性。

```
setea -n Approver -d design.html
```
- 要为符号链接 `design.html` 创建名为 `Approver` 且值为 `Zach` 的扩展属性，请输入：

```
setea -n Approver -v Zach -l design.html
```

### 位置

**/usr/sbin**

## 相关信息

《AIX 5L V5.3 命令参考大全, 卷 1》中的 **chfs** 和 **crfs** 命令。

《AIX 5L V5.3 命令参考大全, 卷 2》中的 **getea** 命令。

---

## setgroups 命令

### 用途

重新设置会话进程组集。

### 语法

```
setgroups [ - ] [ -a GroupSet ] [ -d GroupSet ] [ -r [ Group ] ] [ GroupSet ]
```

### 描述

缺省情况下, **setgroups** 命令为当前 shell 显示用户当前组集和进程组集。用户数据库文件定义用户组集。当给定标志和 *GroupSet* 参数时, 该命令会重新设置 *GroupSet* 参数列出的进程组集。*GroupSet* 参数是一个逗号分隔的组名列表。用户数据库文件中定义了可用的组。

您也可以使用 **setgroups** 命令从当前组集中添加或删除组。使用 **-r** 标志, 您能重新设置实组标识。如果指定了 *Groupset* 参数但没有指定标志, 则 **setgroups** 命令重新设置所有的组并且将第一组放在实际组列表中。**setgroups** 命令不更改控制终端的安全性特征。

当运行 **setgroups** 命令时, 系统始终用新的 shell 代替旧的 shell。不管命令是否成功, 该命令都会替换 shell。因此, 命令不会返回错误代码。

**setgroups -r** 命令和 **newgrp** 命令相同。

### 标志

<b>-a</b> <i>GroupSet</i>	将 <i>GroupSet</i> 参数指定的组添加到当前的会话中。新集合中的组数不能超过 <b>NGROUPS_MAX</b> 组数 (一个在 <b>limits.h</b> 文件中定义的值)。实际组标识不变。
<b>-d</b> <i>GroupSet</i>	从当前会话中除去 <i>GroupSet</i> 参数指定的组。如果除去了实际组, 则当前集内所列的下一组成为实际组。
<b>-r</b> <i>Group</i>	为当前进程重新设置实际组。如果未指定 <i>Group</i> 参数, 并且当前实际组并非主组, 则 <b>-r</b> 标志除去当前实际组并且将其复位成原始主组。如果指定了 <i>Group</i> 参数, 则这与 <b>newgrp</b> 命令的行为相同。
<b>-</b>	将会话的组集重新初始化为它的初始登录状态。

### 安全性

存取控制: 此命令是常规用户程序。该命令应该作为一个程序安装在“可信计算库”(TCB)上。带有 **setuid** (SUID) 位设置的 root 用户应该拥有此命令。

访问的文件:

方式	文件
<b>r</b>	<b>/etc/passwd</b>
<b>r</b>	<b>/etc/group</b>

审计事件:

事件 信息

**USER\_SetGroups** realgroup, groupset

## 示例

1. 作为用户 sah, 您可以显示当前组员身份和进程组集, 请输入:

```
setgroups
```

出现与以下内容类似的输出:

sah:

```
user groups = staff, payroll
process groups = staff, payroll
```

2. 要将 finance 组添加到当前会话的进程组中, 请输入:

```
setgroups -a finance
```

3. 要将实际组设置为 finance, 请输入:

```
setgroups finance, staff, payroll
```

这样就将 finance 设置为实际组。staff 与 payroll 组构成补充组列表。

4. 要从当前进程组集中删除 payroll 组, 请输入:

```
setgroups -d payroll
```

5. 要将进程组集更改回缺省设置, 请输入:

```
setgroups -
```

登录之后会将当前会话复位为原始状态。

## 文件

**/usr/bin/setgroups** 包含 **setgroups** 命令。

**/etc/group** 包含组的基本属性。

**/etc/passwd** 包含基本用户属性。

## 相关信息

**login** 命令、**newgrp** 命令、**setsenv** 命令、**tsm** 命令。

---

## setmaps 命令

### 用途

设置终端映射或代码集映射。

## 语法

要使用 `setmaps` 而不指定输入或输出映射文件，请输入以下内容：

```
setmaps [ -v ] [ -c | -h ]
```

要从缺省目录中选择一个文件作为代码集映射文件，请输入以下内容：

```
setmaps [ -v ] -s -i MapName
```

要选择一个指定文件作为代码集映射文件，请输入以下内容：

```
setmaps [ -v ] -s -i File1
```

要从缺省目录中选择一个文件作为输入或输出终端映射文件，请输入以下内容：

```
setmaps [ -v ] [ -D ] [ -k KeyName ] [ -d DirectoryPath ] { -i | -o } MapName
```

要从缺省目录中选择文件作为输入或输出终端映射文件，请输入以下内容：

```
setmaps [ -v ] [ -D ] [ -d DirectoryPath ] -t MapName
```

要选择一个指定文件作为输入或输出终端映射文件，请输入以下内容：

```
setmaps [ -v ] [ -D ] [ -k KeyName ] { -i | -o } File1
```

要装入缺省终端映射文件以备以后使用，请输入以下内容：

```
setmaps [ -v ] [ -D ] [ -k KeyName ] [ -r ] -i File2
```

要装入一个指定的终端映射文件以备以后使用，请输入以下内容：

```
setmaps [ -v ] [ -D ] [ -k KeyName ] [ -r ] -L File1
```

## 描述

**注：** 如果没有 `root` 用户权限运行该命令，则不能装入代码集映射，仅仅可以调试。

`setmaps` 命令处理终端和代码集映射。`-s` 标志必须用于代码集映射。操作系统使用输入和输出终端映射将内部数据表示转换为异步终端支持的 ASCII 字符。如果您输入没有标志的 `setmaps` 命令，则它显示出当前输入和输出终端映射的名称。

终端映射是一个文本文件，它包含将模式字符串和替换字符串关联的规则列表。这个文件通常存在于 `/usr/lib/nls/termmap` 目录中。操作系统使用输入映射文件将键盘输入映射到应用程序，并使用输出文件将应用程序的输出映射到显示器。

终端映射工作流程如下：

1. 系统将字符收集到缓冲区，直到映射文件中的规则所指定的模式与缓冲区中的子串相匹配。
2. 然后，系统构造并返回规则所指定的替换字符串。

该过程继续处理缓冲区中的剩余字符。

终端映射规则可以测试和更改模式处理器的状态。由单字节字符，通常是数字（0-9）来识别该状态。当系统装入一个新的映射、刷新终端输入或输出缓冲区时（比如当它处理 `KILL` 或 `INTR` 字符或者当程序发出 `ioctl`

系统调用时)，状态复位为 0，即初始状态。终端映射可以使用状态以在其他任务中检测多字节转义序列。您可以通过在模式中指定 @x 来测试状态 x。您可以通过将 @x 包含在替换字符串中来将设置状态为 x。

当使用 **-s** 标志时，**setmaps** 命令将代码映射指定给标准输入设备。操作系统使用代码集映射确定一个字符所需的内存字节数和所需的显示列数。

## 标志

<b>-c</b>	在终端上清除所有的映射。
<b>-d DirectoryPath</b>	使 <i>DirectoryPath</i> 变量用作包含 <i>MapName</i> 变量的目录的路径。指定这个标志和变量覆盖 <b>/usr/lib/nls/termmap</b> 目录。
<b>-D</b>	装入映射前在标准输出设备上生成指定映射的调试程序显示输出。在新的映射上运行此调试程序时，直到映射被完全调试后才能使用 root 用户权限来运行以防止映射被实际加载。
<b>-h</b>	显示 <b>setmaps</b> 命令（高级用户将其和 <b>-v</b> 标志一起使用）的使用信息。
<b>-i MapName</b>	选择 <b>/usr/lib/nls/termmap/MapName.in</b> 文件作为输入映射。当和 <b>-s</b> 标志一起使用时，该标志会选择 <b>/usr/lib/nls/csmmap/MapName</b> 文件作为终端代码集映射文件。
<b>-I File1</b>	选择 <i>File1</i> 变量的内容作为输入映射。 <i>File1</i> 变量指定的文件可以是全路径名或者是当前工作目录的相对路径名。当和 <b>-s</b> 标志一起使用时，该标志会选择 <i>File1</i> 变量的内容作为终端代码页面映射文件。
<b>-k KeyName</b>	将 <i>KeyName</i> 变量的内容和所选定的映射相关联。这个密钥名称覆盖缺省密钥，该缺省密钥一般设置为 <i>MapName</i> 变量的值。
<b>-l File2</b>	装入 <b>/usr/lib/nls/termmap/File2</b> 文件以备后用。 <i>File2</i> 变量包含映射文件的全路径名和后缀（如果有的话）。 <b>注：</b> 您必须具有 root 用户权限才能指定这个标志。
<b>-L File1</b>	装入指定的映射以备后用。 <i>File1</i> 变量包含映射文件的全路径名和后缀（如果有的话）。 <b>注：</b> 您必须具有 root 用户权限才能指定这个标志。
<b>-o MapName</b>	选择 <b>/usr/lib/nls/termmap/MapName.out</b> 文件作为终端输出映射。
<b>-O File1</b>	选择 <i>File1</i> 变量的内容作为终端输出映射。 <i>File1</i> 变量包含映射文件的全路径名和后缀（如果有的话）。
<b>-r</b>	即使指定的映射已经装入，也要强制重新装入。使用旧映射的终端继续装入直到注销或者其映射已明确复位。如果您不指定这个标志，则仅当映射以前没有装入到内核时才能装入该映射。 <b>注：</b> 您必须具有 root 用户权限才能指定这个标志。
<b>-s</b>	将任意映射看成是一个代码集映射。
<b>-t MapName</b>	选择 <b>/usr/lib/nls/termmap/MapName.in</b> 文件作为终端输入映射， <b>/usr/lib/nls/termmap/MapName.out</b> 文件作为终端输出映射。
<b>-v</b>	选择冗长的输出。

所有装入的映射必须有唯一的名称。使用 **-k** 标志消除命名冲突。只有 **-i**、**-o** 和 **-t** 标志隐含地添加一个后缀。如果适当的话，其他指定映射名称的标志应该包括一个后缀。如果内核中已经装入所请求的映射名称，那么即使命令行中提供的路径信息暗示一个不同的映射，仍然会使用该映射。

为了将代码集映射重新设置成原始状态，应该使用 **/usr/lib/nls/csmmap/sbcs** 代码集映射。

## 示例

1. 要显示该终端的当前映射设置，请输入：

```
setmaps
```

2. 要清除当前终端的所有映射，请输入：

```
setmaps -c
```

3. 要为 `ibm3161-C` 终端设置映射（输入和输出映射），请输入：  
`setmaps -t ibm3161-C`
4. 要将 `vt220` 输入映射作为 `fred` 映射装入内核中，请输入：  
`setmaps -k fred -i vt220`
5. 要在名为 `bob.dump` 文件中收集名为 `bob` 的新映射的调试输出，请输入：  
`setmaps -D -L /tmp/bob > bob.dump`
6. 要为终端设置符合 `IBM-932` 代码页的代码集映射，请输入：  
`setmaps -s -i IBM-932`
7. 要为终端设置符合 `IBM-943` 代码页的代码集映射，请输入：  
`setmaps -s -i IBM-943`
8. 要为终端设置来自文件 `myEUC` 的代码集映射，请输入：  
`setmaps -s -I myEUC`

## 文件

<code>/usr/bin/setmaps</code>	包含 <code>setmaps</code> 命令。
<code>/usr/lib/nls/termmap/*.in</code>	包含输入映射文件。
<code>/usr/lib/nls/termmap/*.out</code>	包含输出映射文件。
<code>/usr/lib/nls/csmmap/sbcs</code>	包含单字节代码页的代码集映射。
<code>/usr/lib/nls/csmmap/IBM-943</code>	包含 <code>IBM-943</code> 代码页的代码集映射。
<code>/usr/lib/nls/csmmap/IBM-eucJP</code>	包含 <code>IBM-eucJP</code> 代码页的代码集映射。

## 相关信息

`stty` 命令。

`setmaps` 文件格式、`termios.h` 文件。

`setcsmap` 子例程。

《*AIX 5L V5.3 本地语言支持指南和参考大全*》中的『本地语言支持』。

## setsenv 命令

### 用途

复位用户受保护的状态环境。

### 语法

`setsenv [ - ] NewEnvironment`

### 描述

登录时，`setsenv` 命令复位受保护的状态环境。受保护的状态环境定义为一个变量集。这些变量保留在内核中，只有 `SETUINFO` 系统调用才能修改它们。`setsenv` 命令使用 `NewEnvironment` 参数指定的变量。变量由一个空格分隔的 `EnvironmentVariable= Value` 定义构成。关于环境变量的信息，请参阅 `environment` 文件。

您无法用 **setsenv** 命令复位以下环境变量:

<b>NAME</b>	您最后认证的用户名称。它与当前进程中实用户标识相符。
<b>TTY</b>	您登录的终端名称。它与进程的初始控制终端相符。不使用 <i>full login</i> 初始化的进程，就无法设置这个变量。完全登录是由 <b>getty</b> 命令初始化的登录。
<b>LOGNAME</b>	如果是从终端登录程序启动的话，该名称就是当前会话登录名称。如果会话不是从终端启动的话，则不能设置这个变量。

如果您输入的 **setsenv** 命令不包含任何已定义的变量，则它会显示出当前受保护的状态。**setsenv** 命令不能改变控制终端的安全性特征。

当您运行 **setsenv** 命令时，它会替换当前的 shell 并给出一个新的 shell。不管是否成功完成，该命令都会替换您的 shell。因此，命令不会返回错误代码。

## 标志

- 重新初始化环境，就象用户刚刚登录到系统那样。否则，不能更改环境。

## 安全性

访问控制: 此命令应该是一个标准的用户程序。该命令应该作为一个程序安装在“可信计算库”(TCB)上。带有 **setuid** (SUID) 位设置的 root 用户应该拥有此命令。

访问的文件:

方式	文件
r	<b>/etc/environment</b>
r	<b>/etc/security/envIRON</b>

审计事件:

事件	信息
<b>USER_SetEnv</b>	新的环境字符串

## 示例

1. 要显示当前环境变量，请输入:  

```
setsenv
```
2. 要添加 PSEUDO=tom 受保护的环境变量，请输入:  

```
setsenv PSEUDO=tom
```

本示例为 **PSEUDO** 受保护环境变量设置一个用户名称。

## 文件

<b>/usr/bin/setsenv</b>	为 <b>setsenv</b> 命令指定路径。
<b>/etc/environment</b>	包含每位用户的环境信息。
<b>/etc/security/envIRON</b>	包含每位用户的特权环境信息。



## 相关信息

`login` 命令、`setgroups` 命令、`su` 命令、`tsm` 命令。

`getuinfo` 子例程、`setpenv` 子例程、`usrinfo` 子例程。

关于用户的标识和认证、自主访问控制、可信计算库和审计的更多信息，请参阅《安全性》中的 `Securing the network`。

---

## settime 命令

### 用途

更新文件的访问和修改时间。

### 语法

**settime** [ [ *MMddhhmm*[*yy* ] ] ] | [ **-f** *ReferenceFile* ] ] *File* ...

### 描述

在缺省情况下，**settime** 用当前的访问和修改时间来更新参数文件。如果文件不存在，就不能创建它。如果文件不存在，则 **settime** 命令默默地继续它的操作。

注：对于 **settime** 命令来说，任何超出 2038 年（包括 2038 年）的日期都是无效的日期。

### 标志

**-f** *ReferenceFile*

使用 *ReferenceFile* 对应的时间而不是当前的时间。

### 参数

*MMddhhmm*[*yy*]

以 *MMddhhmm* 或 *MMddhhmm**yy* 格式为 **settime** 命令指定时间，其中 *MM* 是表示月份的两位数，*dd* 是表示月份中天的两位数，*hh* 是表示小时的两位数，*mm* 是表示分钟的两位数，以及 *yy* 是表示年份的两位数。

*File*

指定文件或由空格分隔的文件列表的名称。

### 退出状态

0 命令成功完成。

>0 发生错误。

从 **settime** 返回的代码是指定的文件数，不能成功修改它的时间。

### 示例

1. 要将“infile”文件的访问和修改时间更新为当前时间，请输入：

```
settime infile
```

2. 要将“infile”文件的访问和修改时间更新为和文件“reffile”的访问和修改时间一样，请输入：

```
settime -f reffile infile
```

3. 要更新多个文件的访问和修改时间，请输入：

```
settime file1 file2 file3
```

4. 要将文件的访问和修改时间更新为 2002 年 4 月 9 日 23: 59，请输入：

```
settime 0409235902 infile
```

## 文件

`/usr/bin/settime`

包含 **settime** 命令。

## 相关信息

**touch** 命令。

---

## setuname 命令

### 用途

设置系统的节点名称。

### 语法

```
setuname [-t ] -n Node
```

### 描述

**setuname** 命令用来设置系统的节点名称。必须指定 **-n** 选项。只有具有 root 用户权限的用户才能设置节点名称。使用 **-t** 选项可做临时更改。如果需要临时更改，那么只有在当前运行的内核上才能修改节点名称。临时设置的节点名称在重新引导之后不再存在。没有 **-t** 选项，ODM 数据库中的节点名称将被永久地更改。

### 标志

**-n** *Node*

指定要更改的节点名称。必须指定此选项。*Node* 是主机的主节点名称。这是系统的 UUCP 通信网络名。

**-t**

临时更改。不要试图做永久的更改。重新引导后，将恢复原始名称。

### 退出状态

**0** 命令成功完成。

**>0** 发生错误。

### 示例

1. 要临时地将节点名称更改为 “orion”，请输入：

```
setuname -t -n orion
```

2. 要永久地将节点名称更改为 “orion”，请输入：

```
setuname -n orion
```

## 文件

`/usr/bin/setuname`

包含 `setuname` 命令。

## 相关信息

`chdev` 命令、`hostname` 命令、`uname` 命令。

---

## sh 命令

### 用途

调用缺省 shell。

### 语法

请参阅 `ksh` 命令的语法。将 `/usr/bin/sh` 文件链接到 Korn shell。

### 描述

`sh` 命令调用缺省 shell 并使用它的语法和标志。链接到 `/usr/bin/sh` 路径的 shell 是缺省的 shell。操作系统的标准配置将 `/usr/bin/sh` 路径链接到 Korn shell。有关 Korn shell 程序功能的具体信息，请参阅《操作系统与设备管理》中的『Korn shell 或 POSIX shell 内置命令』。

### 标志

请参阅 Korn shell (`ksh` 命令) 的标志。

## 文件

`/usr/bin/sh`

包含 `sh` 命令。

## 相关信息

`ksh` 命令。

《操作系统与设备管理》中的『Korn shell 或 POSIX shell 内置命令』和『Shell』。

---

## shconf 命令

### 用途

管理系统挂起检测参数。

### 语法

`shconf -d`

`shconf -R -l Name`

`shconf {-D [-O ] | -E [-O ]} [-H] -l Name`

**shconf -l Name [-a Attribute=Value] ...**

## 描述

**shconf** 命令用于显示或者指定优先级问题检测和丢失的 I/O 检测的参数。

对于优先级问题，用户可以指定下列描述的五个操作，并且对于每个操作，用户可以指定检测的优先级别、在较低或同级优先级处没有执行进程或线程时的超时、警告操作的终端设备以及 **getty** 操作：

<b>pp_cmd</b>	启动路径参数指定的命令。
<b>pp_errlog</b>	记录错误日志中的错误。
<b>pp_login</b>	启动由终端设备参数（ <b>term</b> ）指定的串行线路上最高优先级的 <b>getty</b> 。
<b>pp_reboot</b>	重新引导系统。
<b>pp_warning</b>	显示终端设备参数（ <b>term</b> ）指定的控制台上的警告消息。

对于丢失的 I/O，用户可以指定下面列出的操作以及 **errlog**（它在启动丢失的 I/O 检测时是自动生成的）。存在适用于所有启用操作的唯一超时。

<b>lio_warning</b>	显示终端设备参数（ <b>term</b> ）指定的控制台上的警告消息。
<b>lio_reboot</b>	创建系统转储并重新引导系统

## 标志

<b>-d</b>	不管是否启用优先级问题检测和丢失的 I/O 检测，都要显示。
<b>-R</b>	恢复检测的指定名称的缺省值。
<b>-a Attribute=Value</b>	指定用于更改特定的属性值的属性值对。
<b>-D</b>	显示检测的指定名称的缺省值。
<b>-E</b>	显示检测的指定名称的有效值。
<b>-H</b>	显示输出列的头部分。它和 <b>-O</b> 标志同时使用时， <b>-O</b> 标志会覆盖 <b>-H</b> 标志。
<b>-l Name</b>	指定检测名称。
<b>-O</b>	显示由冒号分隔的所有属性名称，在第二行显示由冒号分隔的对应的属性值。也指定了 <b>-E</b> 标志时，属性值是当前值，指定了 <b>-D</b> 标志时，属性值是缺省值。此标志不能与 <b>-a</b> 标志一起使用。

## 文件

**/usr/sbin/shconf** 包含 **shconf** 命令。

---

## shell 命令

### 用途

使用用户的缺省凭证和环境执行 **shell**。

## 语法

### shell

## 描述

**shell** 命令重新初始化用户的登录会话。当给出该命令时，就会重新设置进程的控制终端的端口特征，并取消对端口的所有访问。然后 **shell** 命令为用户将进程凭证和环境重新设置为缺省值，并执行用户的初始程序。根据调用进程的登录用户标识建立所有的凭证和环境。

如果在可信路径上调用 **shell** 命令，并且 `/etc/security/user` 文件中用户的 `tpath` 属性没有 **always** 值，那么就不维护终端的可信环境。

注: **shell** 命令不会重新设置用户的登录标识。

## 安全性

访问控制: 该命令应该被 **setuid** 为 root 用户，以重新设置用户的进程凭证、授权执行 (x) 访问所有用户。该命令应该具有可信计算库属性。

访问的文件:

方式	文件
r	<code>/etc/passwd</code>
r	<code>/etc/group</code>
r	<code>/etc/security/audit/config</code>
r	<code>/etc/security/environ</code>
r	<code>/etc/security/limits</code>
r	<code>/etc/security/user</code>

审计事件:

事件	信息
<b>USER_Shell</b>	portname

## 示例

要在使用可信 shell (**tsh**) 之后将会话重新初始化为缺省凭证和环境，请输入:

```
shell
```

## 文件

<code>/usr/bin/shell</code>	包含 <b>shell</b> 命令。
<code>/etc/security/user</code>	包含用户的扩展属性。
<code>/etc/passwd</code>	包含用户标识。
<code>/etc/group</code>	包含组标识。
<code>/etc/security/audit/config</code>	包含审计配置信息。
<code>/etc/security/environ</code>	定义用户的环境属性。
<code>/etc/security/limits</code>	定义每个用户的进程资源限制。

## 相关信息

**getty** 命令、**init** 命令、**login** 命令、**logout** 命令、**setgroups** 命令、**su** 命令、**tsh** 命令、**tsm** 命令。

关于用户的标识和认证、自主访问控制、可信计算库和审计的更多信息，请参阅《安全性》中的 Securing the network。

---

## show 命令

### 用途

显示消息。

### 语法

```
show [ +Folder ] [ -draft | Messages ] [ -header | -noheader ] [ -showproc CommandString | -noshowproc ]
```

### 描述

**show** 命令显示消息的内容。如果标准输出不是显示器，则 **show** 命令用有一行的头和两个分隔行列出每一条信息。缺省情况下，**show** 命令显示当前文件夹中的当前信息。

**show** 命令调用列表程序来创建列表。缺省列表程序是 `/usr/bin/more`。可以用 `$HOME/.mh_profile` 文件中的 `showproc:` 条目来定义自己的缺省值。如果将 `showproc:` 条目设置为 `mh1`，那么 **show** 命令会代替 `mh1` 命令来调用一个内部 `mh1` 例程。也可以指定程序执行 **-showproc** 标志的 `CommandString` 参数的列表。

**show** 命令将其不能识别的任何标志传递给列表程序。因此，可以给列表程序指定标志，也可以给 **show** 命令指定标志。

如果在 `$HOME/.mh_profile` 文件中有 `Unseen-Sequence:` 条目并且它不是空的，那么 **show** 命令可以除去由概要文件条目指定的每个序列显示的每一条消息。如果指定了几条消息，那么最后显示的消息将成为当前消息。

### 标志

**-draft**

如果 `UserMhDirectory/draft` 文件存在，则显示它。

**+Folder**

指定文件夹。缺省值是当前文件夹。

**-header**

显示正被显示的消息的一行描述。描述包括文件夹名和消息号。如果显示多于一条的消息，则此标志不产生消息头。**-header** 标志是缺省值。

**-help**

列出命令语法、可用的参数选项（开关）和版本信息。

**注：**对于“消息处理程序”（MH），必须完整拼写此标志的名称。

## Messages

指定要显示的消息。您可以指定几条消息、一组消息或者一条消息。使用以下参考来指定消息：

*Number* 消息号。

### Sequence

用户指定的一组消息。识别值包括：

**all** 文件夹中的所有消息。

**cur** 或 **.** (句点)  
当前消息。这是缺省值。

**first** 文件夹中的第一条消息。

**last** 文件夹中的最后一条消息。

**next** 当前消息的下一条消息。

**prev** 当前消息的上一条消息。

**-noheader**

禁止显示每条消息的一行描述。

**-noshowproc**

使用 **/usr/bin/cat** 命令执行列表。这是缺省值。

**-showproc** *CommandString*

使用指定命令字符串执行列表。

## 概要文件条目

在 *UserMhDirectory/mh\_profile* 文件中输入以下条目：

Current-Folder:	设置缺省当前文件夹。
Path:	指定用户的“消息处理程序”目录。
showproc:	指定用于显示消息的程序。
Unseen-Sequence:	指定用于跟踪看不见的消息的序列。

## 示例

1. 要一次显示一屏当前文件夹中的当前消息的内容，请输入：

```
show
```

如果一屏不能完全显示消息，请按 **Enter** 键直到读取整条消息。

2. 要查看当前文件夹中所有消息的内容，请输入：

```
show all
```

如果一屏不能完全显示消息，请按 **Enter** 键直到读取所有的消息。

3. 要查看 *meetings* 文件中消息 5 的内容，请输入：

```
show +meetings 5
```

4. 要查看 *meeting* 文件夹中属于 *weekly* 序列的所有消息的内容，请输入：

```
show +meeting weekly
```

## 文件

**\$HOME/mh\_profile**  
*UserMhDirectory/draft*

指定“消息处理程序”用户概要文件。  
包含当前消息草稿。

`/usr/bin/show`

包含 `show` 命令。

## 相关信息

`mhl` 命令、`next` 命令、`pick` 命令、`prev` 命令、`scan` 命令、`sendmail` 命令。

`.mh_alias` 文件格式和 `.mh_profile` 文件格式。

《网络与通信管理》中的『邮件应用程序』。

---

## showmount 命令

### 用途

显示远程已安装文件系统的所有客户机的列表。

### 语法

```
/usr/bin/showmount [ -a ] [ -d ] [ -e ] [ Host ]
```

### 描述

`showmount` 命令显示 `Host` 参数指定的机器上已远程安装文件系统的所有客户机的列表。`Host` 参数上的 `mountd` 守护程序维护这些信息。`/etc/rmtab` 文件保存这些信息以防服务器崩溃。`Host` 参数的缺省值是由 `hostname` 命令返回的值。

注：如果客户机崩溃，在客户机重新引导并启动 `umount -a` 命令之前，将不会从列表中除去其条目。

注：`showmount` 命令返回 `mountd` 守护程序维护的信息。因为 NFS V4 不使用 `mount` 守护程序，所以 `showmount` 将不返回关于 V4 安装的信息。

### 标志

- `-a` 以 `HostName :Directory` 格式显示所有远程客户机安装，其中 `HostName` 是客户机的名称，`Directory` 是远程安装的目录路径名。
- `-d` 仅列出客户机远程安装的目录。
- `-e` 显示导出目录的列表。

### 示例

1. 要显示主机安装的所有远程目录的列表，请输入：

```
/usr/bin/showmount -a zeus
```

在本示例中，`showmount` 命令产生名为 `zeus` 的主机上客户机安装的所有远程目录的列表。

2. 要显示主机上的客户机安装的目录列表，请输入：

```
/usr/bin/showmount -d athena
```

在本示例中，`showmount` 命令产生名为 `athena` 的主机上客户机安装的所有远程目录的列表。

3. 要显示机器上导出的所有目录列表，请输入：



```
/usr/bin/showmount -e zeus
```

在本示例中，`showmount` 命令产生名为 `zeus` 的主机导出的所有远程目录的列表。

## 文件

`/etc/rmtab` 包含关于所有导出目录的当前状态的信息。  
`/etc/xtab` 列出当前导出目录。

## 相关信息

`hostname` 命令、`umount` 命令。

`mountd` 守护程序。

NFS 命令列表。

《网络与通信管理》中的『网络文件系统』和『NFS 故障诊断』。

---

## shutacct 命令

### 用途

关闭处理记帐。

### 语法

```
/usr/sbin/acct/shutacct [ "Reason" ]
```

### 描述

`shutacct` 命令关闭进程记帐并调用 `acctwtmp` 命令来添加一条说明原因的记录到 `/var/adm/wtmp` 文件中。  
`shutacct` 命令由 `shutdown` 命令调用。

注：必须在 `/var/adm/wtmp` 文件中的 `Reason` 值上加引号。

### 变量

`Reason` 指定记帐系统关闭的原因。这个值是可选的。

### 安全性

访问控制：此命令只授予 `adm` 组的成员以执行（x）权限。

### 文件

`/usr/sbin/acct` 记帐命令的路径。  
`/var/adm/wtmp` 登录与注销的历史记录文件。

## 相关信息

**turnacct** 命令。

有关“记帐系统”、每日和每月报告的准备以及记帐文件的更多信息，请参阅《操作系统与设备管理》中的『系统记帐』。

《操作系统与设备管理》中的『设置记帐子系统』说明了建立记帐系统所必须采取的步骤。

---

## shutdown 命令

### 用途

结束系统操作。

### 语法

```
shutdown [ -d ] [ -F ] [ -h ] [ -i ] [ -k ] [ -l ] [ -m ] [ -p ] [ -r ] [ -t mmdhHHMM [ yy ] ] [ -u ] [ -v ] [ +Time [ Message ] ]
```

### 描述

**shutdown** 命令终止操作系统。只有具有 root 用户权限的用户才能运行此命令。在默认关机状态下，用消息通知用户（通过 **wall** 命令）即将关闭系统。然而，直到用户接收到关闭完成的消息时关机才完成。在显示关闭完成消息前，请勿重新启动系统或者关闭系统；否则会导致文件系统损坏。

**注：**如果调用 **shutdown** 的 tty 通过多端口适配器连接到系统上，则在其上面不会显示 终止完成消息。

当接近关闭时间时，系统上所有用户的终端上都会显示警告消息。

指定的秒数后（缺省情况下是 60 秒），系统停止记帐和错误记录进程，并将一个条目斜到错误日志中。然后，**shutdown** 命令运行 **killall** 命令结束任何保留的进程并运行 **sync** 命令刷新所有内存驻留磁盘块。最后，它卸载文件系统并调用 **halt** 命令。

**注：**对于在运行 **shutdown** 命令的节点上具有打开的文件，但没有登录到此节点的用户，关机消息不会通知他们。

如果您要求完全终止操作系统，**shutdown** 命令将停止所有的进程、卸载所有的文件系统并调用 **halt** 命令。

系统管理员可以在 **/etc/rc.shutdown** shell 脚本中放置本地定制的关闭过程。如果该脚本存在，则在关闭开始时就运行。如果脚本运行但是失败（返回非零返回码），则关机停止。

**注意：**如果您正在将系统置于维护方式下，您必须在 /（根）目录下运行 **shutdown** 命令以确保它能干净地卸载文件系统。

**注：**在缺省情况下，如果在具有软件控制能力的电源的方式下发出该命令，则 **shutdown** 命令能关闭系统的电源。

### 标志

<b>-d</b>	将系统关闭从分布式方式转换到多用户方式。
<b>-F</b>	执行快速关闭，即忽略发送到其他用户的消息并尽可能快速地关闭系统。
<b>-h</b>	完全终止操作系统；和 <b>-v</b> 标志一样。

<b>-i</b>	指定交互方式。显示交互式消息以在关机过程中指导用户。
<b>-k</b>	允许管理员广播 <b>shutdown</b> 警告消息而不导致关闭系统。当使用 <b>-k</b> 标志时，除了发送消息之外不出现其他关闭活动。例如，如果指定了 <b>-1</b> ，则没有进程被杀死而且没有活动被记录到 <b>/etc/shutdown.log</b> 中，同时如果 <b>/etc/rc.shutdown</b> 脚本存在，则该脚本不运行。
<b>-l</b>	创建 / 附加包含有关文件系统、守护程序、用户登录、发放许可证服务以及关闭的网络接口的信息的 <b>/etc/shutdown.log</b> 文件。万一关机失败，此文件可以用来诊断并调试。 <b>注：</b> 使用此标志时，请确保 <b>shutdown</b> 命令有足够的磁盘空间来记录条目。
<b>-m</b>	将系统关闭置于维护（单用户）方式。
<b>-p</b>	不用关闭电源就可终止系统。不间断电源（UPS）使用它。此标志仅仅适用于 AIX 4.2 或以后的版本。 <b>注：</b> 如果 <b>-p</b> 标志和不需要永久停机的标志结合使用，则它将无效。如果其他操作数请求延迟开机和重新引导，则电源仍然关闭。
<b>-r</b>	使用 <b>reboot</b> 命令关闭后，重新启动系统。
<b>-t mdddHHMM [ yy ]</b>	立即关闭系统，然后按照 <b>mdddHHMM [ yy ]</b> 指定的日期重新启动系统，其中 <i>mm</i> 指定月份。 <i>dd</i> 指定天。 <i>HH</i> 指定小时。 <i>MM</i> 指定分钟。 <i>yy</i> 指定年份。 <b>shutdown -t</b> 标志不能和 <b>-v</b> 或 <b>-h</b> 选项一起使用。 <b>注：</b> 这个选项只在具有电源的系统上受支持，该系统关闭时自动关闭电源并发出警告允许稍后重新引导。没有这种能力的系统可能暂挂或在关闭后立即重新启动。
<b>-u</b>	诊断程序使用这个标志来更新闪存并重新引导。
<b>-v</b>	完全终止操作系统。

## 参数

<b>+Time</b>	指定 <b>shutdown</b> 命令停止系统的时间。屏幕上显示 <b>now</b> 表示立即关闭。可以用 <b>+number</b> 或者 <b>hour:minute</b> 两种格式的一种指定未来的时间。第一种格式在指定的分钟数里关闭系统，第二种格式在指定一天的时间（24 小时计时法）上关闭系统。如果指定了 <b>Message</b> 参数，那么也必须指定 <b>Time</b> 参数。
<b>Message</b>	指定消息。

## 示例

1. 要关闭机器，请输入：

```
shutdown
```

这样做会在停止用户进程和 **init** 进程之前等待一分钟，然后关闭系统。

2. 要给用户更多的时间完成正在做的事情并将系统置于维护方式下，请输入：

```
shutdown -m +2
```

两分钟以后会将系统从多用户方式转换到维护方式。

# 文件

`/usr/sbin/shutdown`

包含 `shutdown` 命令。

## 相关信息

`errpt` 命令、`init` 或 `telinit` 命令、`kill` 命令、`killall` 命令、`halt` 命令、`reboot` 命令和 `sync` 命令。

`sigaction` 子例程。

---

## size 命令

### 用途

显示“扩展公共对象文件格式”（XCOFF）对象文件的节大小。

### 语法

```
size [ -d | -o | -x ] [ -f ] [ -V ] [ -X {32|64|32_64} ] File [ File ... ]
```

### 描述

`size` 命令将所有节和每个 XCOFF 文件所需的字节数写入标准输出。如果指定 `-f` 标志，则节的名称遵循节的大小。

### 标志

输出是十进制符号表示法，除非您用下列标志更改输出：

`-d` 用十进制符号表示法写。  
`-f` 将节的名称写入节大小后面的圆括号里。  
`-o` 用八进制记数法写。  
`-x` 用十六进制记数法写。  
`-X mode` 指定 `size` 应该检测的对象文件的类型。 `mode` 必须是以下各项中的一个：

**32** 只处理 32 位对象文件  
**64** 只处理 64 位对象文件  
**32\_64** 既处理 32 位对象文件，又处理 64 位对象文件

缺省值是处理 32 位对象文件（忽略 64 位对象文件）。也可以用 `OBJECT_MODE` 环境变量设置 `mode`。例如，`OBJECT_MODE=64` 使 `size` 处理任意 64 位对象，而忽略 32 位对象。`-X` 标志覆盖 `OBJECT_MODE` 变量。

`-V` 显示 `size` 命令的版本号。

### 示例

1. 要使用十进制来显示 `a.out` 文件的大小，请输入：

```
size
```

这样就以字节为单位显示了可执行 `a.out` 文件的大小。给出了对象代码每一节的大小，总共是：

```
3720 + 1752 + 4152 = 9624
```

2. 要使用八进制来显示对象文件的大小，请输入：

```
size -o driver.o
```

这样就以八进制显示了 **driver.o** 对象文件的大小。

3. 要使用十六进制来显示多个对象代码的大小，请输入：

```
size -x *.o
```

这样就以十六进制显示了当前目录下以 **.o** 结尾的每一个文件的大小。

## 相关信息

**ar** 命令、**as** 命令、**dump** 命令、**ld** 命令、**nm** 命令、**strip** 命令。

---

## skulker 命令

### 用途

除去不需要的文件以清除文件系统。

### 语法

**skulker**

### 描述

注：由于由 **root** 用户运行 **skulker** 命令，它的目的是除去文件，因此可能出现意外结果。安装新的 **skulker** 命令之前，使用 **xargs -p** 命令来手工运行文件除去标准的例外情况来测试可能发生的例外。在您已验证了新的 **skulker** 命令只会除去您想要除去的文件后，您可以安装这个命令。

使用 **skulker** 命令用于定期地从文件系统中清除废弃的或不需要的文件。候选文件包含 **/tmp** 目录中的文件、比指定的寿命大的文件以及下列文件类型：**\*.bak**、**a.out**、**core**、**proof**、**galley**、**...\***、**ed.hup** 以及寿命超过一天的文件。

一般每天调用一次 **skulker** 命令，经常作为在非高峰期间由 **cron** 命令运行的记帐过程的一部分。遵循分布式版本所显示的模式来修改 **skulker** 命令以适合本地需要。本地用户应该知道文件自动除去的标准。

**find** 命令和 **xargs** 命令在 **skulker** 命令中构成强大的组合以供使用。通常大多数文件的选择标准可以使用 **find** 表达式来方便地表示。如果每一个文件都是用单独的命令删除的，那么可以使用 **xargs** 命令将结果文件分段并插入到 **rm** 命令中以减少可能导致的开销。

## 相关信息

**cron** 守护程序、**find** 命令、**rm** 命令、**xargs** 命令。

---

## slattach 命令

### 用途

连接串行线路作为网络接口。

### 语法

```
/usr/sbin/slattach TTYName [ BaudRate DialString [ DebugLevel ] ]
```

## 描述

`/usr/sbin/slattach` 命令给网络接口分配一根 TTY 线路。

在系统启动期间，`slattach` 文件由 `/etc/rc.net` 命令运行以自动配置“系统管理界面程序”（SMIT）所定义的任何“串行网络协议”（SLIP）网络接口。示例部分表明也可以手工配置 SLIP 接口。

对于直接连接的 SLIP 接口，如果没有人工的干预的话，会自动重试断开的连接。对于用调制解调器连接的 SLIP 接口，必须手工重新拨通断开的连接。如果用户在 `slattach` 命令行中提供了拨号字符串，则用户必须重新输入命令和拨号字符串以恢复断开的连接。

终止 `slattach` 命令后，如要拆离接口，请运行 `ifconfig Interface down` 命令。`Interface` 参数是 `netstat` 命令显示的名称。

如果从命令行配置一个 `slip` 接口，则必须使用适当的参数在该 `slip` 接口调用 `/usr/sbin/ifconfig` 命令，而且必须要有 `slip tty` 线路规则来使命令成功执行。要检查 `slip tty` 线路规程是否已经装入，请运行命令 `strinfo -m | grep slip`。如果没有显示输出的话，则表示这个模块还没有装入。发出命令 `strload -m /usr/lib/drivers/slip` 来装入此模块。

注:

1. 用 `ifconfig` 配置好 SLIP 接口后，任何拥有 TTY 许可权的用户都可以发出 `slattach` 命令。
2. 在建立连接之前，您必须配置 `slattach` 命令所使用的 `tty` 设备。您可能需要在 `BNU /usr/lib/uucp/Devices` 文件中为 `tty` 设备设置一个条目。
3. shell 脚本样本 - `/usr/sbin/slipcall`，为调用 `slattach` 并连接到远程系统提供了简化的接口。`slipcall` 对于连接到在激活 SLIP `tty` 线路规程之前需要用户登录的拨入 SLIP 网络是有用的。`slipcall` 的基本配置是使用 `sliplogin` 配置连接到其他的操作系统上，并派生本地、远程网际网地址以及调用系统指定的网络掩码。然后，用远程系统指定的值配置本地接口。

## 参数

<code>BaudRate</code>	设置连接速度。缺省值是 9600。
<code>DebugLevel</code>	设置所需的调试信息级别。可以指定从 0 到 9 的数字。0 值指定没有调试信息；9 指定最多的调试信息。缺省值是 0。
<code>DialString</code>	使用基本联网实用程序（BNU）/UNIX 至 UNIX 复制程序（UUCP）的 <code>chat</code> 语法指定期望/响应序列的字符串。
<code>TTYName</code>	指定 TTY 线路。此字符串是 <code>ttyxx</code> 或 <code>/dev/ttyxx</code> 的格式。

## 示例

1. 要使用直接连接将 SLIP 网络接口连接到 `tty1` 端口上，发出以下命令:

```
slattach /dev/tty1
```

该命令将 `tty1` 连接到 SLIP 所使用的网络接口上。

2. 要使用调制解调器连接将 SLIP 网络接口连接到 `tty1` 上，请发出下列命令:

```
slattach /dev/tty1 9600 ""AT OK \pATF1 OK \pATDT34335 CONNECT""
```

## 文件

`/etc/uucp/Devices` 列出远程连接使用的设备定义。

## 相关信息

**ifconfig** 命令、**netstat** 命令、**sliplogin** 命令。

《网络与通信管理》中的『TCP/IP 网络接口』。

---

## sleep 命令

### 用途

执行暂挂一段时间。

### 语法

**sleep** *Seconds*

### 描述

**sleep** 命令至少在 *Seconds* 参数指定的时间间隔内暂挂进程的执行。*Seconds* 参数指定的时间值可以从 1 到最大整数 (2,147,483,647) 秒。

### 退出状态

该命令返回以下退出值:

- 0 暂挂执行成功至少是 *Seconds* 秒, 或者收到 **SIGALRM** 信号。
- >0 发生错误。

### 示例

1. 要在一段时间以后运行命令, 请输入:

```
(
echo "SYSTEM SHUTDOWN IN 10 MINUTES!" | wall
sleep 300; echo "SYSTEM SHUTDOWN IN 5 MINUTES!" | wall
sleep 240; echo "SYSTEM SHUTDOWN IN 1 MINUTE!" | wall
sleep 60; shutdown
)&
```

系统关闭之前, 命令序列在还剩下的 10 分钟、5 分钟、1 分钟时警告所有的用户。

2. 要在规定的时间间隔内运行命令, 请输入:

```
while true
do
date
sleep 60
done
```

该 shell 过程每分钟显示一次日期和时间。要停止它, 请按下 **Interrupt** 按键序列。

## 相关信息

**shutdown** 命令、**wall** 命令。

**alarm** 子例程、**pause** 子例程、**sigaction** 子例程、**sleep** 子例程。

《操作系统与设备管理》中的『Shell』。

---

## slibclean 命令

### 用途

除去内核和库中任何当前不用的模块。

### 语法

**slibclean**

### 描述

**slibclean** 命令用装载和 0 计数来卸载所有的对象代码。该命令也可以用于通过除去不再需要的对象代码，从共享库区域以及共享库和内核文本区域中除去不再使用的对象代码。

### 文件

**/usr/sbin/slibclean** 包含 **slibclean** 命令。

### 相关信息

**unload** 子例程。

*AIX 5L Version 5.3 Kernel Extensions and Device Support Programming Concepts* 中的 Using Kernel Processes。

---

## sliplogin 命令

### 用途

将标准输入终端线路转变到远程主机的“串行线路网际协议”（SLIP）链路。

### 语法

**sliplogin** [*LoginName*]

### 描述

**sliplogin** 命令将标准输入终端线路配置入连接到远程主机的“串行线路网际协议”（SLIP）；即，该命令附加串行线路网络接口。

注：用户需要使用 root 用户权限才能连接网络接口。

**sliplogin** 命令在 **/etc/slip.hosts** 文件中搜索一个与 *LoginName* 参数值相匹配的登录名条目。如果发现匹配的条目，则 **sliplogin** 为 SLIP（即，8 位透明的输入/输出）配置相应的线路，并将它转换成 SLIP 线路规程。然后，**sliplogin** 调用合适的登录 shell 脚本，此脚本使用 **/etc/slip.hosts** 文件中与登录名条目相关的本地和远程网际协议（IP）地址、网络掩码以及可选的参数来初始化 SLIP 接口。

一般初始化脚本文件是 **/etc/slip.login**。然而，为了满足特殊主机的特定初始化需求，可以创建名为 **/etc/slip.login.userlogin** 的脚本文件（其中 *userlogin* 和 **/etc/slip.hosts** 文件中的登录名相匹配）。当 **sliplogin** 命令存在时，它使用 **/etc/slip.login.userlogin** 脚本文件，而不是 **/etc/slip.login** 脚本文件。



为了取消 SLIP 接口的初始化, **sliplogin** 命令使用 `/etc/slip.logout` 脚本文件或者 `/etc/slip.logout.userlogin` 脚本文件, 如果有其中一个文件存在的话, 优先使用后者。`/etc/slip.logout` 脚本文件和 `/etc/slip.login` 脚本文件有相同的参数; `/etc/slip.logout.userlogin` 脚本文件和 `/etc/slip.login.userlogin` 脚本文件有相同的参数。缺省状况下, `/etc/slip.logout` 脚本文件为指定的 SLIP 部件删除所有的通过网络接口的路由。当 SLIP 接口断开时需要做的额外进程可以添加到其中一个登出脚本文件中。

注:

1. 当远程连接终止或 **sliplogin** 命令坏掉时, 接口自动取消激活。
2. 使用 **slattach** 命令访问有 SLIP 链路配置的远程系统。使用样本 shell 脚本文件 `/usr/sbin/slipcall` 来激活带适当参数的 **slattach** 命令, 需要这些参数来调用远程系统并用远程系统指定的适当的值来配置当地接口。
3. 当在 tty 设备上使用 **sliplogin** 作为用户的登录 shell 时, 为了登录需要启用这个 tty 端口。当使用 **slattach** 而不是 **sliplogin** 作为 SLIP 服务器进程时, 这和配置不同。

## `/etc/slip.hosts` 文件

`/etc/slip.hosts` 文件是配置文件, 它包含预配置 **sliplogin** 用户名和用户登录时分配给本地和远程接口的 IP 地址。**sliplogin** 搜索这个文件以匹配 `LoginName` 条目。该文件有下列格式:

- 忽略注释 (以 # 开始的行) 和空白行。
- 其他的行必须以 `loginname` 参数开始, 并且字段应该包括任何适合那个名称所执行的 **slip.login** 文件的内容。
- 空格用来分隔参数, 并且符合正常的 sh(1) 引用约定。然而, `loginname` 参数不能引用。通常情况下, 行有下列格式:

```
loginname local_address remote_address netmask opt_args
```

其中, `local_address` 与 `remote_address` 是 IP 主机名称或者是 SLIP 线路的本地和远程结点的地址, `netmask` 是相应的 IP 网络掩码。这些参数直接传递给 **ifconfig** 命令。`Opt_args` 是用来配置线路的可选参数。

- **sliplogin** 的实现允许 `/etc/slip.hosts` 文件包含具有不同地址的单一 SLIP 用户的多个条目。这使得相同用户名称的 **sliplogin** 命令能够激活多个 SLIP 接口。当从 `/etc/slip.hosts` 文件中检索用户条目时, 只选择满足下列标准的条目地址。

如果 `slip.hosts` 条目指定了本地系统上在其他非 SLIP 接口中已经使用的本地地址, 就忽略该条目。

如果 `/etc/slip.hosts` 中条目中指定的远程地址已经在任何其他接口上使用, 则忽略该条目。

## `/etc/slip.login` 文件

`/etc/slip.login` 或 `/etc/slip.login.userlogin` 文件是由 **sliplogin** 命令调用以初始化用户网络接口的设置脚本。

如果 `/etc/slip.login.userlogin` 文件存在, 就调用它, 其中 **sliplogin** 命令的 `LoginName` 参数值与 `/etc/slip.hosts` 文件中登录名的条目相对应。如果不能访问这个文件, 就调用 `/etc/sip.login` 文件。登录脚本文件包括下列参数:

<code>slipunit</code>	指定分配到该线路的 SLIP 接口的单元数目。例如, 0 对应 sl0 (sl0 是 s, 小写字母 L, 零。)
<code>speed</code>	指定线路的速度。
<code>args</code>	按顺序指定以 <code>loginname</code> 开头的 <code>/etc/slip.hosts</code> 文件条目的参数。

## `/etc/slip.logout` 文件

`/etc/slip.logout` 或 `/etc/slip.logout.userlogin` 文件是 **sliplogin** 命令调用用来取消用户网络接口初始化的设置脚本。如果 `/etc/slip.logout.userlogin` 文件存在, 就调用它, 其中 **sliplogin** 的 `LoginName` 参数值和 `/etc/slip.hosts` 文件中登录名的条目相对应。如果不能访问该文件, 就调用 `/etc/slip.logout` 文件。

## 标志

`</dev/ttyx` 如果用户已经登录到 **tty** 设备上，并且想将他们的终端配置为 SLIP 线路，则向 **ttyx** 设备重新定向命令。

## 参数

*LoginName* 指定期望的登录名称。缺省值是当前的登录名称。

## 示例

**sliplogin** 命令的正常使用是为每个合法的、远程的 SLIP 的站点创建将 **sliplogin** 作为 shell 的 **/etc/passwd** 条目。例如，

```
foo:!:2010:1:slip line to foo:/tmp:/usr/sbin/sliplogin
```

必须在 **/etc/slip.hosts** 文件中添加条目。条目应该与下列示例相似：

```
foo 1.1.1.1 1.1.1.2 0xffffffff normal
```

其中，*loginname* = *foo*，*local\_address* = 1.1.1.1，*remote\_address* = 1.1.1.2，*netmask* = 0xffffffff，以及 *opt\_args* = *normal*。（可选参数 *normal* 表示激活哪种 SLIP 模式。对于 AIX 4.1，只支持正常模式。）

## 诊断

**sliplogin** 命令将各种信息记录到系统日志守护程序（**syslogd**）里。此处列出的消息按照严重性级别分组。

错误严重性

消息	描述
ioctl (TCGETS): <i>reason</i>	<b>ioctl</b> 子例程不能得到 <i>reason</i> 指定的线路参数。
ioctl (TCSETS): <i>reason</i>	<b>ioctl</b> 子例程不能设置 <i>reason</i> 表示的线路参数。
ioctl (TIOCGTD): <i>reason</i>	<b>ioctl</b> 子例程不能得到 <i>reason</i> 表示的当前 <b>tty</b> 规程。
<b>/etc/slip.hosts</b> : <i>reason</i>	不能打开 <i>reason</i> 表示的 <b>/etc/slip.hosts</b> 文件。
检查接口 <i>xxx</i> 标志失败。原因是 <i>Errno</i> 。	由于 <b>errno</b> 全局变量中所指示的原因，试图检查指示的接口状态以避免可能的地址冲突失败。
用户访问拒绝 - 无 <b>/etc/slip.login</b> [ <i>.userlogin</i> ] 文件。	找不到 <b>/etc/slip.login</b> 或 <b>/etc/slip.login.userlogin</b> 脚本文件。
用户访问拒绝 - 无 <b>/etc/slip.hosts</b> 条目可用。	<b>/etc/slip.hosts</b> 文件中没有登录名条目和命令指定的 <i>LoginName</i> 值相匹配。
访问拒绝 - <i>getlogin</i> 返回 0。	发出 <b>sliplogin</b> 命令的用户没有 <b>/etc/passwd</b> 文件中的密码条目。
注销脚本失败：从 <b>/etc/slip.logout</b> [ <i>.userlogin</i> ] 中退出状态 <i>xxx</i> 。	由于指定的退出状态，试图运行 <b>/etc/slip.logout</b> 或者 <b>/etc/slip.logout.userlogin</b> 脚本文件失败。
<b>ttyx</b> 没有 SLIP 接口。原因是 <i>Errno</i> 。	由于 <b>errno</b> 全局变量中所指示的原因，不能为 <b>ttyx</b> 设备指定 SLIP 接口。尝试运行 <b>ifconfig slx up</b> 命令或者使用 <b>SMIT</b> 来为 <b>tty</b> 设备添加网络接口。
打开 <b>/dev/null</b> : <i>reason</i>	由于指示的原因，试图打开 <b>/dev/null</b> 设备失败。
找不到 <b>/etc/slip.logout</b> 文件	不能找到 <b>/etc/slip.logout</b> 文件。

### 错误严重性

消息	描述
sliplogin: 不能将 SLIP 规程添加到 ttyx	<b>ttyx</b> 设备不存在 SLIP 接口。尝试运行 <b>ifconfig slx up</b> 命令或者使用 <b>SMIT</b> 来为 <b>tty</b> 设备添加网络接口。
从 tty 除去 SLIP 规程失败。原因是 Errno。	由于 <b>errno</b> 全局变量中所指示的原因，试图从 <b>tty</b> 设备中除去 SLIP 规程失败。
tcgetattr: <i>reason</i>	由于 <b>reason</b> 指示的原因，试图读取 <b>tty</b> 设备的当前属性失败。
<i>userlogin</i> 登录失败: /etc/slip.login[ <i>userlogin</i> ] 的退出状态 <i>xxx</i>	由于指定的退出状态，系统调用执行 <b>/etc/slip.login</b> 或者 <b>/etc/slip.login.userlogin</b> 脚本文件失败。

### 信息严重性

消息	描述
为 ttyx 上的 <i>userlogin</i> 附加 SLIP 单元 <i>xxx</i> 。	<b>sliplogin</b> 命令在 <b>/etc/slip.hosts</b> 文件中找到了登录名条目，该条目和命令中指定的 <i>LoginName</i> 值相匹配，调用适用的 <b>/etc/slip.login</b> 或者 <b>/etc/slip.login.userlogin</b> 文件，并且现在附加指示的网络接口。
关闭 <i>userlogin</i> SLIP 单元 <i>xxx</i> (信号)	由于信号的原因， <b>sliplogin</b> 命令终止，从而关闭指定的 <i>userlogin</i> 所指示的 SLIP 单元。

### 通知严重性

消息	描述
为 <i>userlogin</i> 附加 SLIP 单元 <i>xxx</i> 。	已经成功地为所指示的 <i>userlogin</i> 附加了指示的 SLIP 单元。

## 文件

**/etc/slip.hosts**

**/etc/slip.login** 或者 **/etc/slip.login.userlogin**  
**/etc/slip.logout** 或者 **/etc/slip.logout.userlogin**

当用户登录时，为本地和远程接口分配的包含预配置 sliplogin 用户名称和 IP 地址名称的配置文件。

**sliplogin** 命令调用设置脚本来初始化用户的网络接口。  
**sliplogin** 命令调用用来取消初始化用户网络接口的设置脚本。

## 相关信息

**slattach** 命令。

---

## slocal 命令

### 用途

处理接受的邮件。

### 语法

**slocal** [ **-verbose** | **-noverbose** ] [ **-debug** ]

## 描述

每当消息发送到用户时，**slocal** 命令会执行一系列的操作。用户不能启动 **slocal** 命令。**sendmail** 命令调用 **slocal** 命令。

当遇到 **\$HOME/.forward** 文件中的下列行时，**sendmail** 命令启动 **slocal** 命令：

```
/usr/lib/mh/slocal
```

对于每一条接收的消息，**slocal** 命令执行 **.maildelivery** 文件中指定的操作。如果 **slocal** 命令找不到 **\$HOME/.maildelivery** 文件，则 **slocal** 命令使用 **/etc/mh/maildelivery** 缺省文件。如果传送请求失败，**slocal** 命令会将消息传送给 **/usr/mail/\$USER** 文件。

## 标志

<b>-debug</b>	为调试提供信息。
<b>-help</b>	列出命令语法、可用的参数选项（开关）和版本信息。 注：对于消息处理程序（MH），必须完整拼写此标志的名称。
<b>-noverbose</b>	当系统执行 <b>.maildelivery</b> 文件中的命令时，不显示信息。此标志是缺省值。
<b>-verbose</b>	当系统执行 <b>.maildelivery</b> 文件中的命令时，显示信息。

## 文件

<b>/usr/lib/mh/mtstailor</b>	包含消息处理程序（MH）命令定义。
<b>/etc/mh/.maildelivery</b>	包含用于本地邮件传送的缺省 MH 指令。
<b>\$HOME/.maildelivery</b>	为用户提供用于本地邮件传送的 MH 指令。
<b>\$HOME/.forward</b>	包含启动 <b>slocal</b> 命令的行或者转发邮件的路径。
<b>/etc/mh/mh_profile</b>	包含定制 MH 软件包的参数。

## 相关信息

**rcvdist** 命令、**rcvpack** 命令、**rcvstore** 命令、**rcvttty** 命令和 **sendmail** 命令。

《网络与通信管理》中的『邮件应用程序』。

---

## smcaprop 命令

### 用途

提供有关“认证中心”的只读信息。

### 语法

```
smcaprop
```

### 描述

**smcaprop** 命令可在定义为内部“认证中心”（CA）的机器上运行。命令提示输入 CA 专用密钥环密码，然后提供有关 CA 的只读信息（CA 名称、最近发行的认证、CA 认证截止日期等）。有关 CA 执行的所有操作的详细信息（密钥环生成、证书签署等）可在 CA 日志文件 **/usr/webasm/security/SMCa.log** 中找到。

您可以使用基于 Web 的系统管理器（wsm）命令访问图形界面。

## 示例

smcaprop

## 文件

**/usr/websm/security/SMCa.log**  
**/usr/websm/security/SM.caprivkr**

列出 CA 执行的所有操作的详细信息。  
证书专用密钥环文件。

## 相关信息

**smdefca**、**smexpcacert**、**smimpcacert**、**smlicerts**、**smisigncert** 以及 **smundefca** 命令。

有关安装基于 Web 的系统管理器的信息，请参阅《AIX 5L V5.3 基于 web 的系统管理器管理指南》中的『第二章：安装与系统需求』。

---

## smdefca 命令

### 用途

定义内部认证中心。

### 语法

**smdefca** *ca\_name* **-o** *organization* **-c** *country\_code* **-d** *pub\_dir* [ **-e** *mm/dd/yyyy* ]

### 描述

使用 **smdefca** 命令为当前机器上的“基于 Web 的系统管理器”服务器和客户机定义内部 CA（认证中心）。当您定义“基于 Web 的系统管理器”（CA）时，会生成下列文件：

**/usr/websm/security/SM.caprivkr**

它是 CA 专用密钥环，包括 CA 专用密钥和 CA 证书。这是关于“基于 Web 的系统管理器”安全性方面的最敏感的文件。它创建 **root** 保护和加密的密码。

**SMpubkr.class**（在指定的 *pub\_dir* 中创建）

公用密钥环文件。必须将该文件分发给每个“基于 Web 的系统管理器”客户机（针对应用程序方式）和服务器（针对 applet 方式），并且放入 **/usr/websm/codebase** 中。

如果在当前机器上已经定义了 CA，必须首先使用 **smundefca** 命令取消配置。

使用 **/usr/websm/bin/wsm** 命令访问图形界面。快速路径是 **wsm** 系统。

## 标志

*ca\_name*

一个唯一定义您的“基于 Web 的系统管理器”（CA）的名称。带有一些额外的序列号的机器的 TCP/IP 全称可能是一个好的选项。如果您曾经重新定义一个 CA，建议您使用不同的名称以便按照名称识别每一个服务器和客户机使用的 CA。

**注：**不要将 CA 名称正好设置成机器的 TCP/IP 全名（这将中断 SMGate 实用程序，以防您想使用它从远程浏览器上管理这台机器）。

**-o** *organization*

组织名称（CA 证书所必需的）。

**-c** *country\_code*

两个字母的 ISO 国家或地区代码（CA 证书所必需的）。

**-d** *pub\_dir*

公用密钥环文件 **SMpubkr.class** 的输出目录。

**-e** mm/dd/yyyy

CA 证书的失效日期。缺省失效日期是自发出命令日起四年以后。

## 示例

```
smdefca IBMCA1 -o IBM -c US -d /usr/websm/security/tmp -e 12/31/1999
```

## 文件

**/usr/websm/security/SMpubkr.class**  
**/usr/websm/security/SMCa.log**  
**/usr/websm/security/SMCa.sn**  
**/usr/websm/security/SM.caprivkr**

CA 公用密钥环文件。  
列出 CA 执行的所有操作的详细信息。  
证书号文件。  
证书专用密钥环文件。

## 相关信息

**smcaprop**、**smexpcacert**、**smimpcacert**、**smlistcerts**、**smsigncert** 以及 **smundefca** 命令。

有关安装基于 Web 的系统管理器的信息，请参阅《AIX 5L V5.3 基于 web 的系统管理器管理指南》中的『第二章：安装与系统需求』。

---

## smdemon.cleanu 命令

### 用途

为定期的内务处理清除 **sendmail** 队列。

### 语法

**/usr/lib/smdemon.cleanu**

### 描述

**smdemon.cleanu** 命令是一个 shell 过程，用来清除 **sendmail** 命令队列并维护 **/var/spool/mqueue/log** 文件。

要启用 **smdemon.cleanu** 命令，您必须通过从 **/var/spool/cron/crontabs/root** 文件中 **smdemon.cleanu** 行的开头删除 **#** 字符来除去注释语句。如果 **/var/spool/mqueue** 目录不存在，那么就不要再更改 **/var/spool/cron/crontabs/root** 文件。

注意每一个 **smdemon.cleanu** 会话的日志文件的平均大小乘以日志文件数不能超过您所需要的空间。您可以调整日志文件数以便满足您的需要。

**注：****smdemon.cleanu** 命令一般不在命令行输入。**cron** 守护程序执行该命令。

### 示例

为了自动运行 **smdemon.cleanu** 过程，编辑 **/var/spool/cron/crontabs/root** 文件，从如下 **smdemon.cleanu** 行的开始部分删除 **#**（注释字符）：

```
# ulimit 5000; /usr/lib/smdemon.cleanu > /dev/null
```

## 文件

`/var/spool/cron/crontabs/root`  
`/var/spool/mqueue`

调度何时运行 `smdemon.cleanu` 命令。  
包含与邮件队列中消息相关的日志文件和临时文件。

## 相关信息

`cron` 守护程序。

`sendmail` 命令。

《网络与通信管理》中的『邮件记录』和『日志管理』。

---

## smexpcacert 命令

### 用途

导出认证中心（CA）证书。

### 语法

`smexpcacert cert_file`

### 描述

`smexpcacert` 命令可在定义为内部认证中心（CA）的机器上运行。命令提示 CA 专用密钥环密码，然后将内部 CA 的 CA 证书写入文件 `cert_file` 中。`cert_file` 为 CA 证书指定了输出文件的全路径名。

您可以使用基于 Web 的系统管理器命令访问图形界面。

### 示例

```
smexpcacert /tmp/CA1.cert
```

## 文件

`/usr/websm/security/SMCa.log`

列出 CA 执行的所有操作的详细信息。

## 相关信息

`smcaprop`、`smdefca`、`smimpcacert`、`smlistcerts`、`smsigncert` 以及 `smundefca` 命令。

有关安装基于 Web 的系统的信息，请参阅《AIX 5L V5.3 基于 web 的系统管理器管理指南》中的『第二章：安装与系统需求』。

---

## smgenkeycr 命令

### 用途

生成服务器专用密钥和证书请求。

## 语法

```
smgenkeycr { server_name | -f list_file } -o organization -c country_code -d out_dir [ -k 512 ]
```

## 描述

**smgenkeycr** 命令为“基于 Web 的系统管理器”服务器生成专用密钥和证书请求。为输入服务器列表文件中的每一个服务器或已指定名称的服务器生成专用密钥和证书请求。

您可以使用基于 Web 的系统管理器命令访问图形界面。

## 标志

<i>server_name</i>	服务器的 TCP/IP 全名。如果指定的名称是 <b>S</b> ，在输出目录中就会产生一个专用密钥环文件 <b>S.privkr</b> 。
<b>-f</b> <i>list_file</i>	带有服务器名称列表的文件的全路径名称（每行对应一个服务器的 TCP/IP 全称）。对于列表中每一个服务器 <b>S</b> ，在输出目录中会产生专用密钥环文件 <b>S.privkr</b> 。
<b>-o</b> <i>organization</i>	组织名称（服务器证书所需要的）。
<b>-c</b> <i>country_code</i>	两个字母的 ISO 国家或地区代码（服务器证书所需要的）。
<b>-d</b> <i>out_dir</i>	服务器专用密钥环文件的输出目录。
<b>-k</b> 512	导出版本中不存在该选项。服务器专用密钥长度是 512。美国版本中缺省值是 1024，导出版本缺省值是 512。

## 示例

```
smgenprivkr S101.IBM.COM -o IBM -c US -d /usr/websm/security/tmp  
smgenprivkr -f /usr/websm/security/tmp/server.list -o IBM -c US -d /usr/websm/security/tmp
```

## 相关信息

**smgenprivkr**、**smimpservercert**、**sminstkey**、**smlistcerts** 以及 **smserverprop** 命令。

有关安装基于 Web 的系统管理器的信息，请参阅《AIX 5L V5.3 基于 web 的系统管理器管理指南》中的『第二章：安装与系统需求』。

---

## smgenprivkr 命令

### 用途

生成服务器专用密钥环文件。

### 语法

```
smgenprivkr { server_name | -f list_file } -o organization -c country_code -d out_dir [ -k 512 ] [ -e mm/dd/yyyy ]
```

### 描述

**smgenprivkr** 命令可在定义为内部认证中心（CA）的机器上运行。**smgenprivkr** 命令生成“基于 Web 的系统管理器”服务器的“就绪执行”专用密钥环文件。为输入服务器列表文件中的每一个服务器或服务器的名称已指定的服务器生成专用密钥环文件。

您可以使用基于 Web 的系统管理器命令访问图形界面。



## 标志

<i>server_name</i>	服务器的 TCP/IP 全名。如果指定的名称是 <b>S</b> ，在输出目录中就会产生一个专用密钥环文件 <b>S.privkr</b> 。
<b>-f</b> <i>list_file</i>	带有服务器名称列表的文件的全路径名称（每行对应一个服务器的 TCP/IP 全称）。对于列表中每一个服务器 <b>S</b> ，在输出目录中会产生专用密钥环文件 <b>S.privkr</b> 。
<b>-o</b> <i>organization</i>	组织名称（服务器证书所需要的）。
<b>-c</b> <i>country_code</i>	两个字母的 ISO 国家或地区代码（服务器证书所需要的）。
<b>-d</b> <i>out_dir</i>	服务器专用密钥环文件的输出目录。
<b>-k</b> <b>512</b>	导出版本中不存在该选项。服务器专用密钥长度是 512。美国版本中缺省值是 1024，导出版本缺省值是 512。
<b>-e</b> <i>mm/dd/yyyy</i>	服务器证书的失效日期。缺省失效日期是自发出命令日两年以后。

## 示例

```
smgenprivkr S101.IBM.COM -o IBM -c US -d /usr/websm/security/tmp -e 12/31/1999
smgenprivkr -f /usr/websm/security/tmp/server.list -o IBM -c US -d /usr/websm/security/tmp
```

## 文件

**/usr/websm/security/SMCa.log**                      列出 CA 执行的所有操作的详细信息。

## 相关信息

**smgenkeycr**、**smimpservercert**、**sminstkey**、**smlistcerts** 以及 **smisigncert** 命令。

有关安装基于 Web 的系统管理器的信息，请参阅《AIX 5L V5.3 基于 web 的系统管理器管理指南》中的『第二章：安装与系统需求』。

---

## smimpcacert 命令

### 用途

导入认证中心证书。

### 语法

```
smimpcacert dir cert_file
```

### 描述

**smimpcacert** 命令将认证中心（CA）的证书 *cert\_file* 导入到存在于指定 *dir* 目录下的公用密钥环文件 **SMpubkr.class** 中。如果在 *dir* 中没有 **SMpubkr.class** 文件，就会产生一个只包含 *cert\_file* 证书的新 **SMpubkr.class**。

您可以使用基于 Web 的系统管理器命令访问图形界面。

### 参数

*dir*                      **SMpubkr.class** 所在的目录。  
*cert\_file*                CA 证书文件的全路径名。

## 示例

```
smimpcacert /usr/websm/security/tmp CA1.cert
```

## 相关信息

**smcaprop** 命令、**smdefca** 命令、**smexpcacert** 命令、**smlistcerts** 命令、**smsigncert** 命令以及 **smundefca** 命令。

有关安装基于 Web 的系统管理器的信息，请参阅《AIX 5L V5.3 基于 web 的系统管理器管理指南》中的『第二章：安装与系统需求』。

---

## smimpservercert 命令

### 用途

导入服务器证书。

### 语法

```
smimpservercert dir { -a | -s server_name }
```

### 描述

**smimpservercert** 命令将一个服务器证书 (**\*.cert** 文件) 导入到服务器专用密钥文件 (**\*.privk**) 中，生成一个专用密钥环文件 (**\*.privkr**)。您可以导入一个服务器的证书，或所有和指定目录 *dir* 中的专用密钥文件相匹配的证书。

使用 **/usr/websm/bin/wsm** 命令访问图形界面。快速路径是 **wsm** 系统。

### 标志

<i>dir</i>	证书申请 ( <b>*.certreq</b> 文件) 和专用密钥 ( <b>*.privk</b> 文件) 所在的目录，以及写入专用密钥环 ( <b>*.privkr</b> 文件) 的目录。
<b>-a</b>	处理所有和指定目录 <i>dir</i> 中专用密钥文件 ( <b>*.privk</b> ) 相匹配的证书 ( <b>*.certreq</b> 文件)。每一个证书 <b>S.cert</b> 都要导入到专用密钥文件 <b>S.privk</b> 中，生成指定 <i>dir</i> 目录中的专用密钥环文件 <b>S.privkr</b> 。
<b>-s server_name</b>	服务器的 TCP/IP 全名，该服务器的证书 <i>server_name.cert</i> 将导入其专用密钥文件 <i>server_name.privk</i> 中，这将在指定的 <i>dir</i> 目录中生成专用密钥环文件 <i>server_name.privkr</i> 。

## 示例

```
smimpservercert /usr/websm/security/tmp S101.IBM.COM
```

```
smimpservercert /usr/websm/security/tmp -a
```

## 相关信息

**smgenkeycr**、**smgenprivkr**、**sminstkey**、**smlistcerts** 以及 **smserverprop** 命令。

有关安装基于 Web 的系统管理器的信息，请参阅《AIX 5L V5.3 基于 web 的系统管理器管理指南》中的『第二章：安装与系统需求』。

---

## sminstkey 命令

### 用途

安装专用密钥环。

### 语法

**sminstkey** [ *inpdir* | **-d** | *tarfile* ]

### 描述

**sminstkey** 命令期望目录中或者由 TAR 创建的软盘归档中或在 TAR 文件中的当前机器的专用密钥环文件（**S.privkr** 其中 **S** 是 TCP/IP 机器全名），并且作为 **/usr/websm/security/SM.privkr** 安装它。

注：如果是 TAR 文件或软盘 TAR，则专用密钥环出现时没有路径。

如果源专用密钥环文件有加密的密码，则命令会提示输入密码。

您可以使用基于 Web 的系统管理器（wsm）命令访问图形界面。

### 标志

<i>inpdir</i>	源 <b>S.privkr</b> 在 <i>inpdir</i> 目录中。
<b>-d</b>	源 <b>S.privkr</b> 在 TAR 创建的软盘归档文档中。
<i>tarfile</i>	源 <b>S.privkr</b> 在 TAR 文件 <i>tarfile</i> 中。

### 示例

```
sminstkey /usr/websm/security/tmp
```

```
sminstkey -d
```

```
sminstkey /afs/security/privkrs.tar
```

### 文件

**/usr/websm/security/SM.privkr** 服务器专用密钥环文件。

### 相关信息

**smgenkeycr**、**smgenprivkr**、**smimpservercert**、**smlistcerts** 以及 **smserverprop** 命令。

有关安装基于 Web 的系统的信息，请参阅《AIX 5L V5.3 基于 web 的系统管理器管理指南》中的『第二章：安装与系统需求』。

---

## smit 命令

### 用途

执行系统管理。

## 语法

```
smit [ -C | -M ] [ -D ] [ -f ] [ -h ] [ -l File ] [ -o PathName ] [ -p Entity/ValueString ] [ -r RunMode ] [ -s File ] [ -t ] [ -v ] [ [ -m | -n | -d ] FastPath ] [ -X ] [ -x ]
```

## 描述

**smit** 命令调用“系统管理界面工具”（SMIT）。SMIT 是一交互式界面应用程序用来简化系统管理任务。**smit** 命令显示导致交互式对话的菜单层次结构。SMIT 构建并运行用户指定的命令。由于 SMIT 运行命令，您必须有权限执行 SMIT 运行的命令。

SMIT 创建两个文件：**smit.script** 文件和 **smit.log** 文件。调用带有 **-s** *PathName* 标志的 **smit** 命令会将 **smit.script** 文件保存在 *PathName* 参数指定的文件中。如果没有指定 **-s** 标志，则将脚本信息保存在 **\$HOME/smit.script** 文件中。调用带有 **-l** *PathName* 标志的 **smit** 命令会将 **smit.log** 文件保存在 *PathName* 参数指定的文件中。如果没有指定 **-l** 标志，则将日志信息记录在 **\$HOME/smit.log** 文件中。您必须对您要求写入 **smit** 文件的目录具有写许可权，否则不会创建 **smit.script** 文件和 **smit.log** 文件。SMIT 不覆盖 **smit.log** 文件或 **smit.script** 文件。可能时会附加文件。

**smit.script** 文件自动记录命令和命令使用的标志及参数。**smit.script** 文件可作为一个可执行 shell 脚本使用以复制系统配置。SMIT 创建 **smit.log** 文件，它包含可由程序员用来扩展 SMIT 系统所使用的附加的详细信息。**-D**、**-l**、**-t** 以及 **-v** 标志会影响 **smit.log** 文件。

如果您未使用 *FastPath* 参数，则 **smit** 命令将您置于菜单层次结构的顶级。要进入低级菜单，请使用 *FastPath* 参数。SMIT 运行的所有命令都可以用做 *FastPaths*。当您熟悉命令时，*FastPath* 参数能帮助您。例如，您可以输入：**smit chuser** 直接进入您能改变用户特征的对话框。要了解关于 *FastPaths* 的更多信息，请参阅《安全性》中的 Setting up and maintaining roles using SMIT。

注：可以使用 **smitacl.user** 或 **smitacl.group** 命令来控制用户对 SMIT 面板的访问。

SMIT 需要访问下列文件：

<b>sm_menu_opt</b>	SMIT 数据库
<b>sm_name_hdr</b>	SMIT 数据库
<b>sm_cmd_hdr</b>	SMIT 数据库
<b>sm_cmd_opt</b>	SMIT 数据库
<b>smit.log</b>	SMIT 日志文件
<b>smit.script</b>	SMIT 脚本文件
<b>/usr/lpp/msg/.../smit.cat</b>	消息目录

注：如果这些文件中的任何一个损坏，或者存在于当机的 NFS 服务器上，那么 SMIT 可能无法响应。

## 标志

<b>-C</b>	使用 ASCII（也称 Curses）界面启动 SMIT。
<b>-D</b>	设置调试模式；设置 <b>-t</b> 与 <b>-v</b> 标志。
<b>-d</b> <i>FastPath</i>	标识 <i>FastPath</i> 是对话框的名称。
<b>-f</b>	允许 SMIT 的标准输入和输出可以被重定向。
<b>-h</b>	显示命令用法的消息。
<b>-l</b> <i>File</i>	重新将 <b>smit.log</b> 文件重定向到指定的 <i>File</i> 。
<b>-M</b>	使用 windows（也称为 Motif）界面启动 SMIT。
<b>-m</b> <i>FastPath</i>	标识 <i>FastPath</i> 是菜单的名称。
<b>-n</b> <i>FastPath</i>	标识 <i>FastPath</i> 是选择程序的名称。
<b>-o</b> <i>PathName</i>	为 SMIT 对象指定备用库的目录 <i>PathName</i> 。缺省目录是 <b>/etc/objrepos</b> 。

<b>-p</b> <i>Entity/ValueString</i>	<p>此标志仅适用于 SMIT Windows® 版本。允许从命令行填写名称选择和对话。还允许您同时在多个实体上操作。您可以设置环境变量 <code>ENTITY_SEP</code> 和 <code>VALUE_SEP</code> 覆盖缺省的逗号和分号分隔符。</p> <p>您可以以下列任何一种格式输入 <i>Entity/ValueString</i>:</p> <p>“<i>Entity1:Val1,Val2... ; Entity2:Val1,Val2... ; ...</i>”</p> <p>或</p> <p>“<i>Val1,Val2... ; Val1,Val2... ; ...</i>”</p>
<b>-r</b> <i>RunMode</i>	<p>此标志仅适用于 SMIT Windows 版本。指定运行 <code>msmit</code> 的方式。</p> <p>您可以为 <i>RunMode</i> 输入下列值:</p> <ol style="list-style-type: none"> <li><b>1</b> 当在输出窗口中单击 <b>done</b> 时，退出 <code>msmit</code>。</li> <li><b>2</b> 当在对话框中单击 <b>ok</b> 时，退出 <code>msmit</code>。退出时，显示对话框选项。不运行命令。</li> <li><b>3</b> 静默运行 <code>msmit</code>，显示对话框选项。不运行命令。</li> <li><b>4</b> 当在对话框中单击 <b>ok</b> 时，退出 <code>msmit</code>。退出时，显示命令。不运行命令。</li> </ol>
<b>-s</b> <i>File</i>	重新将 <b>smit.script</b> 文件重定向到指定的 <i>File</i> 。
<b>-t</b>	将详细跟踪信息记录在 <b>smit.log</b> 文件中。
<b>-v</b>	记录 SMIT 运行的中间和目标任务命令的命令字符串，同时也将输出记录在 <b>smit.log</b> 文件中。
<b>-x</b>	不运行任何 <b>command_to_execute</b> ，但是仍然为以后的执行记录它们。
<b>-X</b>	不运行任何 <b>command_to_discover</b> 、 <b>command_to_list</b> 、 <b>command_to classify</b> 或 <b>command_to_execute</b> 。

## 示例

1. 要在总体系统管理分层中显示主菜单，输入:

```
smit
```

2. 要更改用户的特征，输入:

```
smit chuser
```

**chuser** 命令是 *FastPath* 参数的一个示例。**smit** 命令和 *FastPath* 参数 **chuser** 让您直接进入对话框，Change User Attributes，指导您完成更改用户的特征。

3. 为了复制配置，将 **smit.script** 文件变为可执行，输入:

```
chmod +x smit.script
```

然后，要复制您的配置，输入:

```
smit.script
```

在配置命令中可以编辑 **smit.script** 文件来创建微小的变动，或仅仅使用命令的子集。应该重新命名或复制 **smit.script** 文件以防止 SMIT 修改它。

**注:** SMIT 在 Korn shell (`/usr/bin/ksh`) 下运行命令。**smit.script** 文件中的一些命令字符串可能需要该环境才能正确运行。

## 文件

<code>/usr/bin/smit</code>	包含 <b>smit</b> 命令。
<code>/etc/objrepos</code>	为 SMIT 数据库指定缺省目录。
<code>smit.log</code>	用时间戳记指定您的会话的详细信息。
<code>smit.script</code>	用时间戳记指定 SMIT 运行的目标任务命令。

## 相关信息

`chmod`、`chsec` 和 `lssec` 命令。

`smitacl.group` 文件和 `smitacl.user` 文件。

《操作系统与设备管理》中的『系统管理界面工具 (SMIT)』。

《AIX 5L V5.3 通用编程概念: 编写并调试程序》中的『编程 - 系统管理界面工具 (SMIT) 概述』。

---

## smitty 命令

### 用途

提供“基于 Curses”的文本界面来执行系统管理。

### 语法

```
smitty [ -C ] [ -D ] [ -f ] [ -h ] [ -l File ] [ -o PathName ] [ -s File ] [ -t ] [ -v ] [ [ -m | -n | -d ] FastPath ] [ -X ] [ -x ]
```

### 描述

**smitty** 命令调用“系统管理界面工具” (SMIT)。SMIT 是一交互式界面应用程序用来简化系统管理任务。**smitty** 命令显示了导致交互式对话的菜单层次结构。SMIT 构建并运行用户指定的命令。由于 SMIT 运行命令, 您必须有权限执行 SMIT 运行的命令。

注: **smitty** 命令和 **smit -C** 一样。

SMIT 创建两个文件: **smit.script** 文件和 **smit.log** 文件。调用带有 **-s PathName** 标志的 **smitty** 命令会将 **smit.script** 文件保存在 *PathName* 参数指定的文件中。如果没有指定 **-s** 标志, 则将脚本信息保存在 **\$HOME/smit.script** 文件中。调用带有 **-l PathName** 标志的 **smitty** 命令会将 **smit.log** 文件保存在 *PathName* 参数指定的文件中。如果没有指定 **-l** 标志, 则将日志信息记录在 **\$HOME/smit.log** 文件中。您必须对您要求写入 **smit** 文件的目录具有写权限, 否则不会创建 **smit.script** 文件和 **smit.log** 文件。SMIT 不覆盖 **smit.log** 文件或 **smit.script** 文件。可能时会附加文件。

**smit.script** 文件自动记录命令和命令使用的标志及参数。**smit.script** 文件可作为一个可执行 shell 脚本使用以复制系统配置。SMIT 创建 **smit.log** 文件, 它包含可由程序员用来扩展 SMIT 系统所使用的附加的详细信息。**-D**、**-l**、**-t** 以及 **-v** 标志会影响 **smit.log** 文件。

如果您未使用 *FastPath* 参数, 则 **smitty** 命令将您置于菜单结构的顶级。要进入低级菜单, 请使用 *FastPath* 参数。SMIT 运行的所有命令都可以用做 *FastPaths*。当您熟悉命令时, *FastPath* 参数能帮助您。例如, 您可以输入: `smitty chuser` 直接进入您能改变用户特征的对话框。要了解关于 *FastPaths* 的更多信息, 请参阅《安全性》中的 Setting up and maintaining roles using SMIT。

SMIT 需要访问下列文件:

<b>sm_menu_opt</b>	SMIT 数据库
<b>sm_name_hdr</b>	SMIT 数据库
<b>sm_cmd_hdr</b>	SMIT 数据库
<b>sm_cmd_opt</b>	SMIT 数据库
<b>smit.log</b>	SMIT 日志文件
<b>smit.script</b>	SMIT 脚本文件
<b>/usr/lpp/msg/.../smit.cat</b>	消息目录

注: 如果这些文件中的任何一个损坏, 或者存在一个崩溃的 NFS 服务器上, SMIT 不能响应。

## 标志

<b>-C</b>	使用“基于 Curses”的文本界面启动 SMIT。它是 <b>smitty</b> 命令的缺省值。
<b>-D</b>	设置调试模式; 设置 <b>-t</b> 与 <b>-v</b> 标志。
<b>-d <i>FastPath</i></b>	标识 <i>FastPath</i> 是对话框的名称。
<b>-f</b>	允许 SMIT 的标准输入和输出可以被重定向。
<b>-h</b>	显示命令用法的消息。
<b>-l <i>File</i></b>	重新将 <b>smit.log</b> 文件重定向到指定的 <i>File</i> 。
<b>-m <i>FastPath</i></b>	标识 <i>FastPath</i> 是菜单的名称。
<b>-n <i>FastPath</i></b>	标识 <i>FastPath</i> 是选择程序的名称。
<b>-o <i>PathName</i></b>	为 SMIT 对象指定备用库的目录 <i>PathName</i> 。缺省目录是 <b>/etc/objrepos</b> 。
<b>-s <i>File</i></b>	重新将 <b>smit.script</b> 文件重定向到指定的 <i>File</i> 。
<b>-t</b>	将详细跟踪信息记录在 <b>smit.log</b> 文件中。
<b>-v</b>	记录 SMIT 运行的中间和目标任务命令的命令字符串, 同时也将输出记录在 <b>smit.log</b> 文件中。
<b>-x</b>	不运行任何 <b>command_to_execute</b> , 但是仍然为以后的执行记录它们。
<b>-X</b>	不运行任何 <b>command_to_discover</b> 、 <b>command_to_list</b> 、 <b>command_to_classify</b> 或 <b>command_to_execute</b> 。

## 示例

1. 要在总体系统管理分层中显示主菜单, 输入:

```
smitty
```

2. 要更改用户的特征, 输入:

```
smitty chuser
```

**chuser** 命令是 *FastPath* 参数的一个示例。 **smit** 命令和 *FastPath* 参数 **chuser** 让您直接进入对话框, Change User Attributes, 指导您完成更改用户的特征。

3. 为了复制配置, 将 **smit.script** 文件变为可执行, 输入:

```
chmod +x smit.script
```

然后, 要复制您的配置, 输入:

```
smit.script
```

在配置命令中编辑 **smit.script** 文件可以创建微小的变动, 或仅仅使用命令的子集。应该重新命名或复制 **smit.script** 文件以防止 SMIT 修改它。

注: SMIT 在 Korn shell ( **/usr/bin/ksh** ) 下运行命令。 **smit.script** 文件中的一些命令字符串可能需要该环境才能正确运行。

## 文件

<code>/usr/bin/smitty</code>	包含 <b>smitty</b> 命令。
<code>/etc/objrepos</code>	为 SMIT 数据库指定缺省目录。
<code>smit.log</code>	用时间戳记指定您的会话的详细信息。
<code>smit.script</code>	用时间戳记指定 SMIT 运行的目标任务命令。

## 相关信息

**chmod** 命令。

《操作系统与设备管理》中的『系统管理界面工具 (SMIT)』。

《AIX 5L V5.3 通用编程概念: 编写并调试程序》中的『编程 - 系统管理界面工具 (SMIT) 概述』。

---

## smlistcerts 命令

### 用途

列出 CA 证书。

### 语法

**smlistcerts** *dir*

### 描述

**smlistcerts** 命令列出存在于指定目录 *dir* 中的公用密钥环文件 **SMpubkr.class** 所包含的 CA 证书。由 *dir* 指定 **SMpubkr.class** 所在的目录。

您可以使用基于 Web 的系统管理器 (wsm) 命令访问图形界面。

### 示例

```
smlistcerts /usr/websm/codebase
```

## 相关信息

**smcaprop**、**smdefca**、**smexpcacert**、**smimpcacert**、**smsigncert** 以及 **smundefca** 命令。

有关安装基于 Web 的系统管理器的信息，请参阅《AIX 5L V5.3 基于 web 的系统管理器管理指南》中的『第二章: 安装与系统需求』。

---

## smrsh 命令

### 用途

sendmail 的受限 shell 程序。

### 语法

**smrsh -c** 命令



## 描述

**smrsh** 命令计划用于替换 **sendmail** 配置文件中 `prog` 邮件程序的 **sh** 命令。**smrsh** 命令使用 **sendmail** 命令语法限制可以运行的程序。这提高了整体的系统安全性。**smrsh** 限制程序员可以执行的一组程序，即使 **sendmail** 运行程序时不使用别名或转发文件。

**smrsh** 命令需要 `/var/adm/sm.bin` 目录中的程序。这允许系统管理员选择可以使用 **smrsh** 命令运行的程序。

**smrsh** 命令还拒绝任何在命令行中带有以下字符的命令，以防止运行结束时受到攻击：`,`、`<`、`>`、`|`、`;`、`&`、`$`、`[r]` (`<` 回车 `>`) 或 `\n` (`<` 新行 `>`)。

- `,`
- `<`
- `>`
- `|`
- `;`
- `&`
- `$`
- `\r` (`<` 回车 `>`)
- 或 `\n` (`<` 新行 `>`)

除去程序中初始的路径名，转发至 `/usr/ucb/vacation`、`/usr/bin/vacation` 和 `/home/server/mydir/bin/vacation`，其中所有 `vacation` 实际上都转发至 `/var/adm/sm.bin/vacation`。系统管理员填充 `/var/adm/sm.bin` 时应小心。合理的附加程序是如 **vacation(1)** 和 **procmail** 之类的实用程序。不要在 `sm.bin` 目录中包含任何 shell 或类似 shell 的程序（例如：**perl**）。这不允许执行任意程序，但是不限制在 `sm.bin` 目录中使用 shell 或 perl 脚本（使用 `#!` 语法）。

## 标志

**-c** 命令

运行 `command` 指定的程序。

## 位置

`/usr/sbin/smrsh`

`smrsh` 命令的缺省位置。

## 文件

`/var/adm/sm.bin`

受限程序的目录。

## 相关信息

**bellmail** 命令、**kill** 命令、**mail** / **Mail** 命令、**mailq** 命令、**mailstats** 命令、**newaliases** 命令、**refresh** 命令、**sendmail** 命令和 **uux** 命令。

**srcmstr** 守护程序。

《网络与通信管理》中的『基本联网实用程序』、『邮件管理』和『传输控制协议 / 因特网协议』

《网络与通信管理》中的『邮件应用程序』。

---

## smserverprop 命令

### 用途

列出服务器属性。

### 语法

**smserverprop**

### 描述

**smserverprop** 命令提供有关本地“基于 Web 的系统管理器”服务器的只读信息（名称、密钥长度、证书截止日期、认证中心名称等）。

您可以使用基于 Web 的系统管理器（wsm）命令访问图形界面。

### 示例

```
smserverprop
```

### 文件

**/usr/websm/security/SM.privkr** 服务器专用密钥环文件。

### 相关信息

**smgenkeycr**、**smgenprivkr**、**smimpservercert**、**sminstkey** 以及 **smlistcerts** 命令。

有关安装基于 Web 的系统管理器的信息，请参阅《AIX 5L V5.3 基于 web 的系统管理器管理指南》中的『第二章：安装与系统需求』。

---

## smcertgen 命令

### 用途

处理证书请求，生成证书。

### 语法

**smcertgen** *dir* { **-a** | **-s** *server\_name* } [ **-e** *mm/dd/yyyy* ]

### 描述

**smcertgen** 命令可在定义为内部认证中心（CA）的机器上运行。命令使用 CA 专用密钥处理认证请求（\*.certreq 文件）并生成证书（\*.cert 文件）。您可以处理一个服务器的请求，或指定目录 *dir* 中的所有服务器请求。

您可以使用基于 Web 的系统管理器（wsm）命令访问图形界面。

## 标志

*dir* 证书请求 (\*.certreq 文件) 所在的目录, 以及证书 (\*.cert 文件) 写入的目录。  
**-a** 处理指定 *dir* 目录中所有的证书请求 (\*.certreq 文件)。对于每个证书请求 (**S.certreq**), 将在指定的 *dir* 目录中生成证书 **S.cert**。  
**-s server\_name** 服务器的 TCP/IP 全名, 将处理它的证书请求 (*server\_name.certreq* 在指定的 *dir* 目录中)。  
**-e mm/dd/yyyy** 服务器证书的失效日期。缺省失效日期是自发出命令日两年以后。

## 示例

```
smsigncert /usr/websm/security/tmp S101.IBM.COM -e 12/31/1999  
smsigncert /usr/websm/security/tmp -a
```

## 文件

**/usr/websm/security/SMCa.log** 列出 CA 执行的所有操作的详细信息。

## 相关信息

**smcaprop**、**smdefca**、**smexpcacert**、**smlistcerts** 以及 **smundefca** 命令。

有关安装基于 Web 的系统管理器的信息, 请参阅《AIX 5L V5.3 基于 web 的系统管理器管理指南》中的『第二章: 安装与系统需求』。

---

## smtctl 命令

### 用途

**smtctl** 命令控制启用和禁用处理器同时多线程方式。

### 语法

```
smtctl [ -m off | on [ -w boot | now ] ]
```

### 描述

为特权用户和应用程序提供此命令来控制具有同时多线程支持的处理器的使用率。同时多线程方式允许处理器在指令级别上拥有线程级别并行性。可以为所有处理器立即或在后继引导系统时启用或禁用此方式。此命令控制同时多线程选项。

## 标志

**-m off** 此选项会将同时多线程方式设置为禁用。  
**-m on** 此选项会将同时多线程方式设置为启用。  
**-w boot** 如果您在下次重新引导系统之前运行 **bosboot** 命令, 那么此选项将使同时多线程方式更改在下次以及后续重新引导时有效。  
**-w now** 此选项使同时多线程方式立即更改, 但这种更改在重新引导时不再保持。如果既没有指定 **-w boot** 选项也没有指定 **-w now** 选项, 那么将立即更改此方式。如果您在下次重新引导系统之前运行 **bosboot** 命令, 那么该方式更改将在后续重新引导时保持有效。

如果不指定选项，那么将报告以下同时多线程设置：

SMT 功能	物理处理器或虚拟处理器具有同时多线程功能的指示符。
SMT 方式	禁用或启用的当前运行时同时多线程方式。
SMT 引导方式	禁用或启用的当前引导时同时多线程方式。
SMT 线程	每个物理处理器或虚拟处理器的同时多线程的线程数。
SMT 绑定	同时多线程线程在相同物理处理器或虚拟处理器上绑定的指示符。

## 退出状态

0	成功完成请求的操作。
>0	发生错误。

## 示例

1. 要启用当前引导循环的同时多线程，请输入：

```
smtctl -m on -w now
```

系统显示与以下内容类似的消息：

```
smtctl: SMT is now enabled.
```

2. 要查看当前同时多线程方式设置和处理器信息，请输入：

```
smtctl
```

系统显示与以下内容类似的消息：

```
This system is SMT capable.
```

```
SMT is currently enabled.
```

```
SMT boot mode is set to disabled.
```

```
proc0 has 2 SMT threads  
Bind processor 0 is bound with proc0  
Bind processor 1 is bound with proc0
```

```
proc2 has 2 SMT threads  
Bind processor 2 is bound with proc2  
Bind processor 3 is bound with proc2
```

3. 要禁用当前引导循环和所有后继引导的同时多线程，请输入：

```
smtctl -m off
```

系统显示与以下内容类似的消息：

```
smtctl: SMT is now disabled. It will persist across reboots if  
you run the bosboot command before the next reboot.
```

注：在下次重新引导之前必须用 **bosboot** 命令重新进行引导映像。

## 位置

**/usr/sbin/smtctl**

## 文件

`/usr/sbin/smtctl` 包含 `smtctl` 命令。

## 相关信息

《AIX 5L V5.3 命令参考大全, 卷 1》中的 `bosboot` 和 `bindprocessor` 命令。

---

## smundefca 命令

### 用途

取消配置内部认证中心。

### 语法

`smundefca`

### 描述

`smundefca` 命令用于取消配置当前机器上先前定义的内部认证中心 (CA)。

`smundefca` 命令除去下列文件:

<code>/usr/websm/security/SM.caprivkr</code>	包含 CA 专用密钥和 CA 证书的 CA 专用密钥环。
<code>/usr/websm/security/SMCa.sn</code>	证书号文件。

不能删除日志文件 `/usr/websm/security/SMCa.log`。

您可以使用基于 Web 的系统管理器 (wsm) 命令访问图形界面。

### 示例

```
smundefca
```

## 文件

<code>/usr/websm/security/SMCa.log</code>	列出 CA 执行的所有操作的详细信息。
<code>/usr/websm/security/SM.caprivkr</code>	证书专用密钥环文件。
<code>/usr/websm/security/SMCa.sn</code>	证书号文件。

## 相关信息

`smcaprop`、`smdefca`、`smexpcacert`、`smimpcacert`、`smlistcerts` 以及 `smsigncert` 命令。

有关安装基于 Web 的系统管理器的信息, 请参阅《AIX 5L V5.3 基于 web 的系统管理器管理指南》中的『第二章: 安装与系统需求』。

---

## snap 命令

### 用途

收集系统配置信息。

### 语法

```
snap [ -a ] [ -A ] [ -b ] [ -B ] [ -c ] [ -C ] [ -D ] [ -f ] [ -g ] [ -G ] [ -i ] [ -k ] [ -l ] [ -L ] [ -n ] [ -N ] [ -p ] [ -r ] [ -R ] [ -s ] [ -S ] [ -t ] [ -T Filename ] [ -w ] [ -o OutputDevice ] [ -d Dir ] [ -v Component ] [ -O FileSplitSize ] [ -P Files ] [ script1 script2 ... | All | file:filepath ]
```

```
snap [ -a ] [ -A ] [ -b ] [ -B ] [ -c ] [ -C ] [ -D ] [ -f ] [ -g ] [ -G ] [ -i ] [ -k ] [ -l ] [ -L ] [ -n ] [ -N ] [ -p ] [ -r ] [ -R ] [ -s ] [ -S ] [ -t ] [ -T Filename ] [ -o OutputDevice ] [ -d Dir ] [ -v Component ] [ -O FileSplitSize ] [ -P Files ] [ script1 script2 ... | All | file:filepath ]
```

```
snap -e [ -m Nodelist ] [ -d Dir ]
```

### 描述

**snap** 命令收集系统配置信息并将信息压缩到 **pax** 文件中。此文件可能会写入设备（例如，磁带或 DVD），或传送到远程系统。也许需要用 **snap** 命令收集的信息来识别并解决系统问题。

注：需要具有 root 用户权限来执行 **snap** 命令。使用 **snap -o /dev/cd0** 命令将压缩图像复制到 DVD。使用 **snap -o /dev/rmt0** 命令将图像复制到磁带上。

使用 **snap -o /dev/rfd0** 命令将压缩图像复制到软盘上。使用 **snap -o /dev/rmt0** 命令将图像复制到磁带上。

需要大约 8 MB 的临时磁盘空间来收集所有的系统信息，包含错误日志的内容。如果不使用 **snap -a** 命令收集所有的系统信息，可能只需要较少的磁盘空间（取决于选择的选项）。

注：如果您打算使用磁带将快照映像发送给 IBM 以获取软件支持，则磁带必须为以下格式中的一种：

- 8mm, 2.3 Gb 容量
- 8mm, 5.0 Gb 容量
- 4mm, 4.0 Gb 容量

使用其他格式会妨碍或延迟 IBM 软件支持检查其内容。

**snap -g** 命令收集常规系统信息，包括以下的信息：

- 错误报告
- 定制“对象数据管理器”（ODM）数据库的副本
- 跟踪文件
- 用户环境
- 物理内存和调页空间的数量
- 设备和属性信息
- 安全性用户信息

将 **snap -g** 命令的输出写到 **/tmp/ibmsupt/general/general.snap** 文件中。

**snap** 命令检查 **/tmp/ibmsupt** 目录中的可用的空间，此目录是 **snap** 命令输出的缺省目录。可使用 **-d** 标志将输出写到另一个目录中。如果没有足够的空间来存放 **snap** 命令的输出，那么您必须扩展文件系统。

**snap** 命令的每一次执行都会将信息附加到以前创建的文件中。使用 **-r** 标志除去以前收集并保存过的信息。

## 标志

<b>-a</b>	收集所有的系统配置信息。该选项需要大约 8 MB 的临时磁盘空间。
<b>-A</b>	收集异步 (TTY) 信息。
<b>-b</b>	收集 SSA 信息。
<b>-B</b>	绕过 SSA 适配器转储集合。 <b>-B</b> 标志仅在同时指定了 <b>-b</b> 标志的情况下才起作用，否则请忽略 <b>-B</b> 标志。
<b>-c</b>	创建 <b>/tmp/ibmsupt</b> 目录树或其他指定输出目录中的所有文件的压缩 <b>pax</b> 映像 ( <b>snap.pax.Z</b> 文件)。 <b>注:</b> 在使用 <b>-c</b> 标志之前，应将未使用该选项收集的信息复制到 <b>snap</b> 目录树中。如果需要一 个测试用例来演示系统问题，那么在压缩 <b>pax</b> 之前，将该测试用例复制到 <b>/tmp/ibmsupt/testcase</b> 目录中。
<b>-C</b>	检索 <b>fwddump_dir</b> 目录中的所有文件。这些文件位于“general”子目录中。 <b>-C</b> <b>snap</b> 选项的行为 与 <b>-P*</b> 相同。
<b>-D</b>	收集转储和 <b>/unix</b> 信息。使用主要的转储设备。 <b>注:</b> <ol style="list-style-type: none"><li>1. 如果使用 <b>bosboot -k</b> 指定非 <b>/unix</b> 的运行内核，就会收集到不正确的内核。当转储时，请 确保 <b>/unix</b> 是正在使用的内核或者连接到正在使用的内核。</li><li>2. 如果将转储文件复制到主机机器上，那么 <b>snap</b> 命令不能收集 <b>/tmp/ibmsupt/dump</b> 目录中 的转储图像。相反，它创建转储目录和实际转储图像的链接。</li></ol>
<b>-d AbsolutePath</b>	标识可选的 <b>snap</b> 命令输出目录 ( <b>/tmp/ibmsupt</b> 是缺省目录)。必须指定绝对路径。
<b>-e</b>	收集 HACMP™ 特定信息。 <b>注:</b> 从属于集群的所有节点收集 HACMP 特定数据。该标志不能与除 <b>-m</b> 和 <b>-d</b> 外的任何其他 标志一起使用。
<b>-f</b>	收集文件系统信息。
<b>-g</b>	收集 <b>lspp -hac</b> 命令的输出，该命令对于重新创建确切的操作系统环境是必需的。将输出写入 <b>/tmp/ibmsupt/general/lspp.hBc</b> 文件。也收集常规系统信息并将输出写到 <b>/tmp/ibmsupt/general/general.snap</b> 文件中。
<b>-G</b>	用 <b>-g</b> 标志将预定义的对象数据管理器 (ODM) 文件包含到收集到的常规信息中。
<b>-i</b>	收集安装调试关键产品数据 (VPD) 信息。
<b>-k</b>	收集内核信息
<b>-l</b>	收集编程语言信息。
<b>-L</b>	收集 LVM 信息。
<b>-m Nodelist</b>	用于收集 HACMP 信息的节点名列表 (用逗号分隔)。 <b>注:</b> 当前该标志仅在与 <b>-e</b> 标志一起使用时有效。
<b>-n</b>	收集网络文件系统 (NFS) 信息。
<b>-N</b>	禁止检查必需的可用空间。
<b>-o OutputDevice</b>	将压缩图像复制到指定设备上。
<b>-O FileSplitSize</b>	用来启用分割，以将 <b>snap</b> 输出文件分割为更小文件。将这些文件的大小作为一个参数指定到 <b>-O</b> 选项，并且必须以兆字节为单位进行指定。只有指定了 <b>-c</b> 标志，才能使用此标志。
<b>-p</b>	收集打印机信息。
<b>-P Files</b>	从 <b>fwddump_dir</b> 目录检索指定的 <b>Files</b> 。如果指定了 <b>-P *</b> ，将收集该目录中的所有文件。这些文 件放置在 <b>general</b> 子目录中。 <b>-C</b> <b>snap</b> 选项的作用与 <b>-P*</b> 相同。
<b>-r</b>	从 <b>/tmp/ibmsupt</b> 目录中除去 <b>snap</b> 命令的输出。
<b>-R</b>	收集 SCSI RAID 信息。
<b>-s</b>	收集“系统网络体系结构” (SNA) 信息。
<b>-S</b>	用 <b>-g</b> 标志将安全性文件包含到收集到的一般信息中。
<b>-t</b>	收集传输控制协议信息。

**-T** *Filename* 为多个 CPU 跟踪收集所有日志文件。 **-g** 标志只收集库文件 **trcfile**。  
**-v** *Component* 显示 **snap** 命令所执行的命令的输出。使用该标志查看指定的文件名称或文件组。  
**注**: 按下 Ctrl-C 键序列来中断 **snap** 命令。返回带有下列选项的提示: 按下 Enter 键返回到当前操作; 按下 S 键停止当前操作; 按下 Q 键完全退出 **snap** 命令。  
**-w** 收集 WLM 信息。

## 参数

### 自变量

要执行的第三方脚本的名称将作为参数指定到 **snap**。参数可以是单字, 也可以是加引号的字的列表。参数被括在引号中时, 列表中第一个参数表示脚本名称, 后续的字表示传递到脚本的参数。

**All** 指定为参数时, 执行脚本资源库中的所有脚本。在此情况下不会传递任何脚本参数。

如果使用了 **file:** 关键字, 并且它随后紧跟文件路径, 则读取此文件来获取要执行的脚本。文件的每一行都表示一个脚本和脚本的可选参数。

### snap 脚本

第三方脚本在 **/usr/lib/ras/snapscripts** 中必须可执行, 并且必须遵循以下描述的准则。脚本在传递 1 过程中被调用时, 必须将其大小估计值返回给 **snap**。在传递 2 中, 它将按照 **snap** 指定那样收集数据并将其保存。

脚本必须读取并利用以下环境变量: **SNAPDIR**、**PASSNO**、**SCRIPTSIZE** 和 **SCRIPTLOG**。

必须将所有输出文件写入 **\$SNAPDIR**。它是脚本应该保存其输出的目录。**PASSNO** 变量包含调用脚本过程中的 **snap** 阶段。在第一个传递过程中, 脚本应该计算出在第二个传递过程中它将撰写的数据的大小估计值。然后, 它将此数字估计值写入 **\$SCRIPTSIZE** 指向的文件。保存在该文件中的值应该为十进制。**snap** 将路径传递到应该保存脚本的所有调试数据的日志文件。脚本不应重定向标准输出和标准错误, 因为 **Snap** 将标准输出和标准错误分别保存到 **\$SNAPDIR/ScriptName.out** 和 **\$SNAPDIR/ScriptName.err**。

示例:

```
#!/usr/bin/ksh

if [ "$PASSNO" = 1 ]
then
    (( size=99999 ))
    ....
    # this is where code to do the size estimation should go.
    ....
    echo $size > $SCRIPTSIZE

else if [ "$PASSNO" = 2 ]
then
    # debug information should go to $SCRIPTLOG
    echo "Debug Data" >> $SCRIPTLOG

    # .....where the work to collect the data takes place
    # ...

    # The data collected should be written to $SNAPDIR
    touch $SNAPDIR/foo_output1
    touch $SNAPDIR/foo_output2

fi
fi
```



注: 要收集有关虚拟 SCSI 设备的信息, 请运行命令 `snap client_collect,all`。

**Snap 输出分割:** 如果被分割了, `snap` 输出可能类似于以下内容:

```
% ls -l
total 112048
-rw-r--r--  1 lmhc   adm      6291456 Nov 26 09:56 snap.hastings.112603095649.pax.Z.xaa
-rw-r--r--  1 lmhc   adm      6291456 Nov 26 09:56 snap.hastings.112603095649.pax.Z.xab
-rw-r--r--  1 lmhc   adm      6291456 Nov 26 09:56 snap.hastings.112603095649.pax.Z.xac
-rw-r--r--  1 lmhc   adm      6291456 Nov 26 09:56 snap.hastings.112603095649.pax.Z.xad
-rw-r--r--  1 lmhc   adm      6291456 Nov 26 09:56 snap.hastings.112603095649.pax.Z.xae
-rw-r--r--  1 lmhc   adm      6291456 Nov 26 09:56 snap.hastings.112603095649.pax.Z.xaf
-rw-r--r--  1 lmhc   adm      6291456 Nov 26 09:56 snap.hastings.112603095649.pax.Z.xag
-rw-r--r--  1 lmhc   adm      6291456 Nov 26 09:56 snap.hastings.112603095649.pax.Z.xah
-rw-r--r--  1 lmhc   adm      6291456 Nov 26 09:56 snap.hastings.112603095649.pax.Z.xai
-rw-r--r--  1 lmhc   adm      744518  Nov 26 09:56 snap.hastings.112603095649.pax.Z.xaj
```

## 示例

1. 输入下列命令收集所有的系统配置信息:

```
snap -a
```

该命令的输出写入 `/tmp/ibmsupt` 目录中。

2. 输入以下命令创建 `/tmp/ibmsupt` 目录中包含的所有文件的 `pax` 映像:

```
snap -c
```

3. 输入以下命令收集常规系统配置信息, 包括 `lspp -hBc` 命令的输出:

```
snap -g -o /dev/rfd0
```

输出将写入 `/tmp/ibmsupt/general/lspp.hBc` 和 `/tmp/ibmsupt/general/general.snap` 文件。该命令也将系统信息写入可拆卸的软盘中。

4. 输入以下命令, 从属于单个集群的节点 `node1` 和 `node2` 收集 HACMP 特定信息。

```
snap -e -m node1,node2
```

输出写入 `/tmp/ibmsupt/hacmp` 目录中。

5. 要运行脚本 `foo1`、`foo2` 和 `foo3` (其中 `foo1` 不带自变量, `foo2` 带有三个自变量, `foo3` 带有一个自变量), 请输入以下内容:

```
snap foo1 "foo2 -x -y 3" "foo3 6"
```

假定目的地目录是一个缺省目录

`/tmp/ibmsupt`, 输出将写入 `/tmp/ibmsupt/snapscripts/foo1`、`/tmp/ibmsupt/snapscripts/foo2` 和 `/tmp/ibmsupt/snapscripts/foo3`。

6. 要指定 **All** 参数来运行所有脚本, 请输入:

```
snap All
```

注: 在此情况下不传递任何参数。

7. 要指定文件 (其包含将执行的脚本的名称和可选参数列表) 的路径, 请输入:

```
snap file:/tmp/scriptnames
```

一个从示例 5 执行脚本的样本输入文件:

```
foo1
foo2 -x -y 3
foo6
```

8. 如果希望将 `snap` 输出分割成 4MB 的文件, 则请输入:

```
snap -a -c -O 4
```

## 文件

<code>/usr/sbin/snap</code>	包含 <b>snap</b> 命令。
<code>/tmp/ibmsupt</code>	包含 <b>snap</b> 命令的输出。
<code>/tmp/ibmsupt/general/lslpp.hBc</code>	包含 <b>lslpp -hBc</b> 命令的输出，需要该命令来重新创建精确的操作系统环境。
<code>/tmp/ibmsupt/general/general.snap</code>	包含用 <b>snap -g</b> 命令收集到的常规系统信息。
<code>/tmp/ibmsupt/testcase</code>	包含演示您的系统问题的测试用例。

## 相关信息

**errpt** 命令、**gettrc** 命令、**lslpp** 命令、**snapsplit** 命令、**sysdumpdev** 命令和 **sysdumpstart** 命令。

---

## snapcore 命令

### 用途

收集 **core** 文件。

### 语法

```
snapcore[ -d Dir] [-r] core [program]
```

### 描述

**snapcore** 命令搜集 **core** 文件、程序和程序使用的库，并将信息压缩到 **pax** 文件中。然后将文件下载到磁盘或磁带，或者发送到远程系统。需要用 **snapcore** 命令收集的信息来识别并解决应用程序的问题。

**snapcore** 命令检查 `/tmp/ibmsupt` 目录中的可用的空间，此目录是 **snapcore** 命令输出的缺省目录。您可以使用 **-d** 标志将输出写到另一个目录中。如果没有足够的空间来存放 **snapcore** 命令的输出，那么您必须扩展文件系统。

每次执行 **snapcore** 命令都会创建一个新的归档文件。归档文件命名为 **snapcore\_\$(pid).pax**。使用 **-r** 标志除去以前创建的归档文件。该命令使用 **\$(pid)** (**snapcore** 命令的 **pid**) 创建唯一的文件名称并保存任何以前创建的归档。

指定核心和程序的全路径名。如果没有指定程序名，则 **snapcore** 会从 **core** 文件读取程序名并在 **PATH** 变量中所包含的目录中搜索其位置。

### 标志

**-dDir** 标识可选择的 **snapcore** 命令的输出目录 (`/tmp/ibmsupt` 是缺省值)。  
**-r** 从 `/tmp/ibmsupt` 目录中除去 **snapcore** 命令的输出。

### 示例

- 要搜集 **core** 文件，输入以下命令：
  - `snapcore <core file name> <program name>`
  - `snapcore <core file name>`

搜索 *PATH* 变量所包含的目录以查找程序文件。在 `/tmp/snapcore` 目录中创建 `pax` 文件。

2. 为了清除以前创建的核心归档并创建一个新的归档，输入以下命令：

```
snapcore -r<core file name> <program name>
```

在 `/tmp/snapcore` 目录中创建 `pax` 文件。

3. 要在备用目录中创建 `core` 文件归档，输入下列命令：

```
snapcore -d<dir name> <core file name> <program name>
```

在 `<dirname>/tmp/snapcore` 目录中创建 `pax` 文件。

4. 为了清除 `/tmp/snapcore` 目录，输入下列命令：

```
snapcore -r
```

## 文件

<code>/usr/sbin/snapcore</code>	包含 <code>snapcore</code> 命令。
<code>/tmp/snapcore</code>	包含 <code>core</code> 文件归档。

## 相关信息

`dbx` 命令、`pax` 命令。

---

## snapshot 命令

### 用途

修改、创建或者查看快照的属性。

### 语法

```
snapshot { -o snapfrom=snappedFS -o size=Size | {-d [-s] | -q [-cfieldSeparator] | -o snapfrom=snappedFS | -o size=Size} Object}
```

### 描述

给 JFS2 快照提供界面。

### 标志

<code>-d</code>	删除快照和包含快照的逻辑卷以及任何以前的快照。指定的 <i>Object</i> 是一个快照存储对象。
<code>-s</code>	当删除快照时，保留指定快照的指定逻辑卷。

- q** 如果指定的 *Object* 是一个快照存储对象，则显示有关指定快照的信息。  
  
显示快照属于的文件系统、获得快照的时间、快照存储对象的大小以及剩余的可用空间。如果指定的对象是 `snappedFS`，则显示有关 `snappedFS` 的所有快照的信息。显示每一个快照和它们的存储对象以及获得快照的时间。
- cfieldSeparator** 显示快照存储对象的大小和剩余的可用空间。  
指定以冒号的格式显示快照查询的输出。*fieldSeparator* 是字符，用来分隔显示的字段。
- o snapfrom=snappedFS** 创建指定的 *snappedFS* 的快照。如果没有指定 *Object*，就会创建一个新的逻辑卷以供快照使用。如果指定的 *Object* 是一个逻辑卷，则它必须存在并且必须和 *snappedFS* 在同一个卷组中。如果指定的对象已经作为一个快照使用或作为 `/etc/filesystems` 文件所知的文件系统使用，那么发出错误消息并失败。
- o size=Size** 如果用 `-o snapfrom=snappedFS` 标志指定，那么指定新逻辑卷的大小。否则，该标志将 *Object* 字段指定的快照的大小增加到 *Size* 的值。如果给出任意其他的标志，那么就将忽略该标志。如果 *Size* 后面跟随一个 `M`，则认为该值是兆字节。如果 *Size* 后面跟随一个 `G`，则认为该值是千兆字节。如果 `M` 或者 `G` 都不使用，则认为该值是 512 字节块。

## 参数

- Object* 如何使用取决于其他指定的标志，参阅对象解释的标志描述。它可能是设备或文件系统。

## 示例

1. 为了给 `/dev/snapsb` 逻辑卷上的 `/home/janet/sb` 文件系统创建快照，输入下列命令：  

```
snapshot -o snapfrom=/home/janet/sb /dev/snapsb
```

该命令给已经存在的 `/dev/snapsb` 逻辑卷上的 `/home/janet/sb` 文件系统创建快照。
2. 为 `/home/janet/sb` 文件系统创建快照，输入下列命令：  

```
snapshot -o snapfrom=/home/janet/sb -o size=16M
```

该命令创建一个 16 兆字节大小的逻辑卷并为新建逻辑卷上的 `/home/janet/sb` 文件系统创建一个快照。
3. 为了查看有关 `/home/janet/sb` 文件系统上所有快照的信息，输入下列命令：  

```
snapshot -q /home/janet/sb
```

该命令显示 `/home/janet/sb` 文件系统上的每一个快照、获取快照的时间、每一个快照存储对象的大小以及剩余的可用空间。
4. 为了增加 `/dev/snapsb` 设备上快照的大小，输入下列命令：  

```
snapshot -o size=64M /dev/snapsb
```

该命令将 `/dev/snapsb` 设备增加到 64 兆字节并且包含设备上的快照。
5. 要删除 `/dev/snapsb` 设备上的快照，输入下列命令：

```
snapshot -d /dev/snapsb
```

该命令删除 `/dev/snapsb` 设备上包含的快照并除去 `/dev/snapsb` 逻辑卷。

---

## snapsplit 命令

### 用途

要将 **snap** 输出文件分割为任意或指定大小的多个更小文件。

### 语法

```
snapsplit [ -s size ] [ -H machinename ] [ -f filename ]
```

```
snapsplit -u -T timestamp [ -H machinename ]
```

### 描述

使用 **snapsplit** 命令将 **snap** 输出文件分割为更小文件。此命令有助于处理非常大的 **snap** 文件。它将文件分割为指定大小（以 1 兆字节的倍数计）的文件。而且，当使用 **-u** 选项调用它时，它还会将这些文件组合为原始文件。

以下是输出文件的名称：**snap.machinename.timestamp.pax.Z.xxx**。*Machinename* 是主机名，*timestamp* 的格式为 MMDDYYHHMMSS。另外，*xxx* 表示 **split** 文件的扩展名，将这些文件组合为原始文件时，该扩展名很关键。这些文件开头的扩展名按以下顺序列出：**xaa**、**xab**、**xac**、**xad**、**xae**，……，**xaz**、**xba**、**xbb**、**xbc**、**xbd**，……，**xbz**、**xca**、**xcb**、**xcc**，……。

对这些文件执行 **ls** 时，列出的第一个文件将表示原始文件的开头，而最后一个文件则表示原始文件的末尾。

请注意，只能对 **snap** 文件（这些文件为 **paxed** 并受到压缩）使用该命令。在收集 **snap** 输出的本地系统上执行时，则不需要使用 **-H** 选项。如果用户已经将完整的 **snap** 文件移动到远程系统并希望将其分割，在这种情况下需要提供此标志。可以选择任意机器名，但是建议使用收集数据的机器名。

### 标志

<b>-f filename</b>	输入 <b>snapsplit</b> 文件。它应该是压缩的 <b>pax</b> 文件。缺省情况下为 <b>snap.pax.Z</b> 。
<b>-H machinename</b>	主机名称。如果未作指定，缺省情况下为当前主机。必须小心命名相应系统的 <b>snap</b> 文件。
<b>-s size</b>	指定 <b>snap</b> 输出的大小（以 1 MB 的倍数计）。最后一个文件将小于等于此大小。应该以兆字节为单位输入 <i>Size</i> 。缺省大小为 1 MB。
<b>-T timestamp</b>	用来存储原始 <b>snap</b> 输出的 <b>snapsplit</b> 文件时间戳记。其格式为 MMDDYYHHMMSS，其中 MM 代表月，DD 代表天，YY 代表年，HH 代表小时，MM 代表分钟，SS 代表秒。
<b>-u</b>	用来重新加入 <b>snapsplit</b> 文件的标志。结合 <b>-T</b> 标志使用。

### 示例

1. 要分割缺省 **snap** 文件（**snap.pax.Z** 应该位于当前目录），请输入以下内容：

```
snapsplit
```

该命令的输出写入当前目录。

2. 要从系统 **doe** 分割文件 **snap.somefile.pax.Z**，请输入以下内容：

```
snapsplit -H doe -f snap.somefile.pax.Z
```

注：所生成的文件名为 **snap.doe.MMDDYYHHMMSS.pax.Z**。

3. 要恢复用于系统 **sue** 的 **snap** 文件 (**snap.sue.102303141211.xxx**) 的一个文件和时间戳记 102303141211, 请输入:

```
snapsplit -u -T 102303141211 -H sue
```

警告：如果丢失或重命名了 **snap** 文件中的任何一个文件，创建的 **snap** 文件将损坏。

4. 要从带有时间戳记 102603084512 的文件中恢复 **snap** 文件，并且这些文件用于当前系统，请输入:

```
snapsplit -u -T 102603084512
```

5. 要收集常规系统配置信息（包括 **lspp -hBc** 命令的输出），请输入以下内容:

```
snap -g -o /dev/rfd0
```

输出写入 **/tmp/ibmsupt/general/lspp.hBc** 和 **/tmp/ibmsupt/general/general.snap** 文件。该命令也将系统信息写入可拆卸的软盘中。

## 文件

**/usr/sbin/snapsplit** 包含 **snapsplit** 命令。

## 相关信息

**snap**、**split** 和 **cat** 命令。

---

## snmpevent 命令

### 用途

发送 **ERRM** 事件至 **SNMP** 管理器。

### 语法

```
snmpevent [-a host-name] [-c community] [-h]
```

### 描述

**snmpevent** 脚本向运行简单网络管理协议(SNMP)代理程序的主机发送事件响应资源管理器 (ERRM) 事件的 SNMP 陷阱。代理程序将陷阱信息格式化为 SNMP 陷阱并将其发送到在其配置文件中定义的 SNMP 管理器。预定义的 **ERRM** 响应生成 **SNMP** 陷阱调用该脚本。捕获事件或重整事件信息并由该 **ERRM** 事件或重整事件发生时所生成的环境变量传递。

**snmpevent** 脚本也可用作创建其他用户定义操作的模板。请参阅 *RSCT Administration Guide* 以了解事件响应资源如何运行操作命令。

当发生事件或重整事件时，以下消息模板作为陷阱发送，而 **snmpevent** 是定义的响应。

```
[ERRM_COND_SEVERITY] [ERRM_TYPE] occurred:  
Condition: [ERRM_COND_NAME]  
Node: [ERRM_NODE_NAME]  
Resource: [ERRM_RSRC_NAME]  
Resource Class: [ERRM_RSRC_CLASS_NAME]  
Resource Attribute: [ERRM_ATTR_NAME]  
Attribute Type: [ERRM_DATA_TYPE]  
Attribute Value: [ERRM_VALUE]
```

环境变量具有以下定义:

#### **ERRM\_COND\_SEVERITY**

指定引起事件或重整事件的条件资源的重要性。有效值为: 关键、警告或信息性。

#### **ERRM\_TYPE**

指定发生的事件类型。有效值为: 事件或重整事件。

#### **ERRM\_COND\_NAME**

用属性值指定条件资源名 (该属性值的更改引起事件或重整事件)。

#### **ERRM\_NODE\_NAME**

指定发生事件或重整事件的主机名。

#### **ERRM\_RSRC\_NAME**

用属性指定资源名 (该属性值的更改引起事件或重整事件)。

#### **ERRM\_RSRC\_CLASS\_NAME**

指定引起事件或重整事件的资源所属的资源类名称。

#### **ERRM\_ATTR\_NAME**

指定资源属性名称, 其更改引起事件或重整事件。

#### **ERRM\_DATA\_TYPE**

指定资源属性的数据类型。

#### **ERRM\_VALUE**

指定资源属性值, 其更改引起事件或重整事件。

**snmpevent** 命令捕获这些环境变量值, 并格式化通过调用 **snmptrap** 命令而作为陷阱发送的类属消息。

## 标志

#### **-a** *host-name*

指定 AIX 子代理将要连接的 SNMP 代理的主机名。缺省情况下, 该子代理将连接到在本地节点上运行的 SNMP 代理上。

**-c** 指定要使用的 SNMP 团体。这可以是 SNMP 代理可接受的任何字符串。缺省为 **public**。

**-h** 将该脚本的用法语句写到标准输出。

## 参数

#### *log\_file*

指定记录事件信息的文件名。需要指定 *log\_file* 参数的绝对路径。

*log\_file* 可看作循环日志, 具有 64 KB 的固定大小。当写满 *log\_file* 时, 新条目将改写在现有的最早的条目上。

如果 *log\_file* 已存在, 那么向它附加事件信息。如果 *log\_file* 不存在, 就创建一个以便写入事件信息。

## 退出状态

**0** 脚本已成功运行。

**1** 脚本运行时发生错误。

## 限制

脚本必须在正在运行 ERRM 的节点上运行。

## 标准输出

指定 **-h** 标志时，该脚本的用法语句将写到标准输出。

## 示例

1. 假设命令 `/usr/sbin/rsct/bin/snmpevent` 是紧急通知响应中的操作，该响应与 CSM 预定义的条件 **NodeChanged** 有关。这可通过运行 `mkcondresp` 命令然后运行 `startcondresp` 命令来完成。`/etc/snmpdv3.conf` 文件应当配置为陷阱将要发送到的地方。在本示例中，如果您想让陷阱发送到 **9.117.16.246**，请按如下方式写 `/etc/snmpdv3.conf` 文件：

```
VACM_GROUP group1 SNMPv1 public -
VACM_VIEW defaultView internet - included
-VACM_ACCESS group1 - - noAuthNoPriv SNMPv1 defaultView - defaultView -

NOTIFY notify1 traptag trap -

#TARGET_ADDRESS Target1 UDP 127.0.0.1 traptag trapparms1 - - -
TARGET_ADDRESS Target1 UDP 9.117.16.246 traptag trapparms1 - - -

TARGET_PARAMETERS trapparms1 SNMPv1 SNMPv1 public noAuthNoPriv -

COMMUNITY public public noAuthNoPriv 0.0.0.0 0.0.0.0 -

DEFAULT_SECURITY no-access - -

logging file=/usr/tmp/snmpdv3.log enabled
logging size=0 level=0

smux 1.3.6.1.4.1.2.3.1.2.1.2 gated_password # gated

snmpd smuxtimeout=200 #muxatmd
smux 1.3.6.1.4.1.2.3.1.2.3.1.1 muxatmd_password #muxatmd
```

然后，通过先杀死当前正在运行的 `snmpd` 守护程序来重新启动 `snmpd` 守护程序，然后重新启动：

```
# ps -ef | grep snmpd
  root 4570 12956 1 08:24:32 pts/0 0:00 grep snmpd
  root 13810 1 0 08:11:04 - 0:00 snmpd
# kill -9 13810
# snmpd
```

下一步，将节点 `c175n08` 的 `LParID` 属性改为 12：

```
# chnode c175n08 LParID=12
```

现在，在具有 `/etc/snmpdv3.conf` 文件所指定的 SNMP 管理器的节点 **9.117.16.158** 上，SNMP 管理器应该记录类似如下的内容：

```
2002-07-15 09:09:25 c174tr1.ppd.pok.ibm.com [9.114.78.17] TRAP, SNMP v1,
community public
    enterprises.ibm Enterprise Specific Trap (1) Uptime: 0:01:45.00
    enterprises.ibm.ibmProd.191.1.6.1.0 = "Informational Event
occurred. Condition=NodeChanged Node=c174tr1.ppd.pok.ibm.com
Resource=c175n08.ppd.pok.ibm.com Resource Class=Node Resource
Attribute=Changed Attributes Attribute Type=CT_CHAR_PTR_ARRAY Attribute
Val={LParID} "
```

输出内容根据 SNMP 管理器而变化。



## 位置

`/usr/sbin/rsct/bin/snmpevent` 包含 `snmpevent` 脚本

## 相关信息

命令: `lscondition`、`lscondresp`、`lsresponse`、`mkcondition`、`mkcondresp`、`mkresponse`、`snmptrap`、`startcondresp`、`stopcondresp`

书籍: 请参阅 *RSCT Administration Guide* 来获取有关事件响应资源管理器 (ERRM) 以及有关如何使用 ERRM 环境变量的信息

---

## snmpd 守护程序

### 用途

作为后台进程启动简单网络管理协议 (SNMP) 代理程序。

### 语法

参照 `snmpdv1` 或 `snmpdv3` 守护程序的语法。

### 描述

`/usr/sbin/snmpd` 是符号链接, 可链接到仅支持 SNMP V1 的 `snmpdv1` 守护程序, 也可链接到支持 SNMP V3 的加密或不加密的 `snmpdv3` 守护程序版本。关于 `snmpd` 守护程序的信息, 请参考 SNMP 代理相应版本的文档。在 AIX 5.2 上, 系统引导期间运行的缺省 SNMP 代理程序是 SNMP V3 代理程序的非加密版本。

注: AIX 扩展包中提供了 SNMP V3 代理程序的加密版本。

### 文件

<code>/usr/sbin/snmpd</code>	包含可链接到 <code>/usr/sbin/snmpdv1</code> 、 <code>/usr/sbin/snmpdv3e</code> 或 <code>/usr/sbin/snmpdv3ne</code> 的符号链接。
<code>/usr/sbin/snmpdv1</code>	包含 SNMP V1 代理程序。
<code>/usr/sbin/snmpdv3e</code>	包含 SNMP V3 代理程序的加密版。
<code>/usr/sbin/snmpdv3ne</code>	包含 SNMP V3 代理程序的非加密版。

## 相关信息

`snmpv3_ssw` 命令。

《网络与通信管理》中的『网络管理的 SNMP』章节。

---

## snmpdv1 守护程序

### 用途

作为后台进程启动简单网络管理协议 (SNMP) V1 代理程序

## 语法

**snmpd** [ **-c** *ConfigFile* ] [ **-d** *Level* ] [ **-f** *LogFile* ] [ **-S** ]

## 描述

**snmpd** 命令启动 SNMP 守护程序。该命令只能由具有 root 特权的用户或系统组成员发出。

SNMP 守护程序是一个支持 RFC 定义的标准简单网络管理协议 (SNMP) 和 RFC 1155 和 RFC 1213 定义的管理信息库 (MIB) 的服务器。SNMP 守护程序提供以下三种功能:

- 接收与认证来自网络监视器的 SNMP 请求。
- 处理请求并将结果返回到发出请求的监视器。
- 向配置文件列出的所有主机发送陷阱通知。

SNMP 守护程序服务器将日志消息保存在 *LogFile* 变量指定的文件中 (如果使用 **-f** 标志), 或保存在配置文件指定的日志文件中。当日志文件的大小超出了日志文件预定义的最大大小时, **snmpd** 命令就会通过以下步骤将旧的日志文件移动到另一个文件中来循环该日志文件:

- 删除 *LogFile.3*。
- *LogFile.2* 移动到 *LogFile.3*。
- *LogFile.1* 移动到 *LogFile.2*。
- *LogFile.0* 移动到 *LogFile.1*。
- *LogFile* 移动到 *LogFile.0*。
- 继续在 *LogFile* 中记录日志。

如果记录不是从带有 **-f** 的 **snmpd** 命令行指示, 记录可使用配置文件指示。

受支持的设置变量有:

- **sysContact**
- **sysName**
- **sysLocation**
- **ifAdminStatus**
- **atPhysAddress**
- **atNetAddress**
- **ipForwarding**
- **ipDefaultTTL**
- **ipRouteDest**
- **ipRouteNextHop**
- **ipRouteType**
- **ipNetToMediaPhysAddress**
- **ipNetToMediaNetAddress**
- **ipNetToMediaType**
- **snmpEnableAuthenTraps**
- **smuxPstatus**
- **smuxTstatus**

关于受支持的设置变量的更多信息，请参阅 *AIX 5L Version 5.3 Communications Programming Concepts* 中的“Understanding SNMP Daemon Support for SET Request Processing”。

下列命令应当在 SNMP 守护程序启动前发出：

- **ifconfig loopback**
- **startsrc -s inetd**

这些命令通常将在系统启动期间，调用 **/etc/rc.net** 和 **/etc/rc.tcpip** shell 脚本时执行。（**snmpd** 命令可放在 **/etc/rc.tcpip** shell 脚本中。）

应当使用“系统资源控制器”（SRC）控制 **snmpd** 守护程序。建议不要在命令行输入 **snmpd**。

## 使用“系统资源控制器”操作 **snmpd** 守护程序

**snmpd** 守护程序是“系统资源控制器”（SRC）控制的子系统。**snmpd** 守护程序是 **tcpip** 系统组的成员。缺省情况下启用 **snmpd** 守护程序并可使用 SRC 命令操作。

请使用以下 SRC 命令操作 **snmpd** 守护程序：

<b>startsrc</b>	启动子系统、子系统组或子服务器。发出 <b>startsrc</b> 命令使 <b>snmpd</b> 命令生成 <i>coldStart</i> 陷阱。
<b>stopsrc</b>	停止子系统、子系统组或子服务器。
<b>refresh</b>	使子系统或子系统组重新读取相应的配置文件。发出 <b>refresh</b> 命令使 <b>snmpd</b> 守护程序生成 <i>warmStart</i> 陷阱。
<b>traceson</b>	启用子系统、子系统组或服务器的跟踪。如果发出 <b>traceson</b> 命令的用户不是 <b>root</b> 用户，则调试级别将不会超出级别 2。
<b>tracesoff</b>	禁用子系统、子系统组或服务器的跟踪。
<b>lssrc</b>	获取子系统、子系统组或服务器的状态。如果发出长状态格式 <b>lssrc</b> 命令的用户不是 <b>root</b> 用户，则不会显示团体名信息。

## 标志

<b>-c ConfigFile</b>	指定 <b>snmpd</b> 守护程序的配置文件的全路径和文件名。当 <b>snmpd</b> 守护程序启动及 <b>refresh</b> 或 <b>kill-1</b> 信号发出时，读取该文件。如果不指定 <b>-c</b> 标志，缺省配置文件是 <b>/etc/snmpd.conf</b> 。有关该文件格式的信息，请参阅 <b>snmpd.conf</b> 文件。								
<b>-d Level</b>	指定跟踪 <b>snmpd</b> 命令生成的级别。该 <i>Level</i> 值可为下列其中之一： <table><tr><td><b>0</b></td><td>所有注意事项、异常和致命错误消息</td></tr><tr><td><b>1</b></td><td>级别 0 加上调试消息</td></tr><tr><td><b>2</b></td><td>级别 1 加上接收和发送信息包的十六进制转储</td></tr><tr><td><b>3</b></td><td>级别 2 加上一个请求与响应信息包的英语版本。</td></tr></table>	<b>0</b>	所有注意事项、异常和致命错误消息	<b>1</b>	级别 0 加上调试消息	<b>2</b>	级别 1 加上接收和发送信息包的十六进制转储	<b>3</b>	级别 2 加上一个请求与响应信息包的英语版本。
<b>0</b>	所有注意事项、异常和致命错误消息								
<b>1</b>	级别 0 加上调试消息								
<b>2</b>	级别 1 加上接收和发送信息包的十六进制转储								
<b>3</b>	级别 2 加上一个请求与响应信息包的英语版本。								
<b>-f LogFile</b>	如果不指定 <b>-d</b> 标志，调试级别设置为 0。 指定记录 <b>snmpd</b> 跟踪信息的全路径和文件名。如果不指定 <b>-f</b> 标志，就不记录任何信息。关于设置记录参数的更多信息，请参阅 <b>snmpd.conf</b> 文件。								
<b>-S</b>	如果指定，启用安全性选项。它可防止本地非 <b>root</b> 用户更改本地主机上的 MIB 变量的值。								

## 示例

1. 要启动 **snmpd** 守护程序，请输入类似下列命令：

```
startsrc -s snmpd -a "-f /tmp/snmpd.log"
```

该命令启动 **snmpd** 守护程序并以调试级别 0 将信息记录在 **/tmp/snmpd.log** 文件中。

2. 要正常停止 **snmpd** 守护程序，请输入：

```
stopsrc -s snmpd
```

该命令停止守护程序。**-s** 标志指定随后停止的子系统。

3. 要从 **snmpd** 守护程序获取短状态，请输入：

```
lssrc -s snmpd
```

该命令返回守护程序的名称、进程标识和守护程序的状态（活动或不活动）。

4. 要从 **snmpd** 守护程序获取长状态，请输入：

```
lssrc -ls snmpd
```

如果您是 **root** 用户，该长格式状态报告列出配置的团体名和相关访问权限，并可查看 **snmp** 请求。该长格式报告也列出与主机陷阱通知相关的团体名，记录配置参数、**snmpd** 特定配置参数和 **smux** 配置参数。

5. 要启用 **snmpd** 守护程序的跟踪，请输入以下命令：

```
traceson -s snmpd
```

如果配置 **snmpd** 守护程序用于记录，该命令启用 **snmpd** 调试。

6. 要查看 DHCP 服务器数据库文件 **/etc/dhcpd.ar** 和 **/etc/dhcpd.cr** 的内容，请输入：

```
lssrc -l -s dhcpd
```

## 文件

### **/etc/services**

包含所需服务的端口指定。如果下列条目还不存在，它们必须存在于 **/etc/services** 文件中：

```
snmp    161/udp
```

#### **snmp-trap**

```
162/udp
```

```
smux    199/tcp
```

#### 注：

1. **snmp** 端口必须是 RFC 1157 要求的 161。
2. **snmp** 陷阱端口必须是 RFC 1157 要求的 162。
3. **smux** 端口必须是 199。
4. **/etc/services** 文件已在适当的位置上包含这些条目。
5. 如果 **/etc/services** 文件正从服务器服务，这些条目必须存在于服务器的 **/etc/services** 文件中。

### **/etc/snmpd.conf**

指定 **snmpd** 代理程序的配置参数。

### **/etc/mib.defs**

定义 SNMP 代理应当识别与处理的管理信息库（MIB）变量。

## 相关信息

**gated** 守护程序、**snmpd** 守护程序、**snmpv3\_ssw** 命令。

---

## snmpdv3 守护程序

### 用途

作为后台进程启动简单网络管理协议（SNMP）V3 代理程序。

### 语法

```
snmpd [ -d Level ] [ -i Interval ] [ -p Port ] [ -S ]
```

### 描述

**snmpd** 命令启动 SNMP 守护程序。该命令只能由具有 root 特权的用户或系统组成员发出。

SNMP 守护程序是一个支持 RFC 1157、RFC 1905 和 RFC 2572 记录的所有 SNMPv1、SNMPv2c 和 SNMPv3 协议的服务器。它也可用作 RFC 1227 定义的 SMUX 服务器和 RFC 1592 定义的 DPI2 代理程序来工作。SNMP 守护程序提供以下三种功能：

- 接收与认证来自网络监视器的 SNMP 请求。
- 处理请求并将结果返回到发出请求的监视器。
- 向配置文件列出的所有主机发送陷阱通知。

SNMP 守护程序服务器将日志消息保存在 *LogFile* 变量指定的文件中（如果使用 **-f** 标志），或保存在配置文件指定的日志文件中。当日志文件的大小超出了日志文件预定义的最大大小时，**snmpd** 命令就会通过以下步骤将旧的日志文件移动到另一个文件中来循环该日志文件：

- 删除 LogFile.3。
- LogFile.2 移动到 LogFile.3。
- LogFile.1 移动到 LogFile.2。
- LogFile.0 移动到 LogFile.1。
- LogFile 移动到 LogFile.0。
- 继续在 LogFile 中记录。

下列命令应当在 SNMP 守护程序启动前发出：

- **ifconfig loopback**
- **startsrc -s inetd**

这些命令通常将在系统启动期间，调用 */etc/rc.net* 和 */etc/rc.tcpip* shell 脚本时执行。（**snmpd** 命令可放在 */etc/rc.tcpip* shell 脚本中。）

应当使用系统资源控制器（SRC）控制 **snmpdv3** 守护程序。建议不要在命令行输入 **snmpd**。

### 使用“系统资源控制器”操作 snmpd 守护程序

**snmpdv3** 守护程序是“系统资源控制器”（SRC）控制的子系统。**snmpdv3** 守护程序是 **tcpip** 系统组的成员。缺省情况下启用 **snmpdv3** 守护程序并可使用 SRC 命令操作。

请使用以下 SRC 命令操作 **snmpd** 守护程序：

<b>startsrc</b>	启动子系统、子系统组或子服务器。发出 <b>startsrc</b> 命令使 <b>snmpdv3</b> 命令生成 <i>coldStart</i> 陷阱。
<b>stopsrc</b>	停止子系统、子系统组或子服务器。
<b>lssrc</b>	获取子系统、子系统组或子服务器的状态。

## 标志

**-d** *Level*

指定要启动的跟踪级别。级别的有效值为 0-255。如果不指定 **-d** 参数，那么使用缺省级别 0，即不进行跟踪。如果指定 **-d** 参数时没有指定级别，则使用 31 这一级别，这意味着所有 SNMP 请求 / 响应 / 陷阱和 DPI® 活动都会被跟踪。

共提供 8 个跟踪级别。选定的每个级别都有相应的数字。与每个选定的跟踪级别相关的数字的和就是应该指定的级别值。跟踪级别的数字为：

- 0 不跟踪。这是缺省值。
- 1 跟踪 SNMP 响应、请求和陷阱。
- 2 跟踪 DPI 级别 1 和 DPI 级别 2。
- 3 等同于级别 1 加上级别 2 加上内部跟踪。
- 4 等同于跟踪级别 3 加上扩展跟踪。

**-i** *Interval*

指定时间间隔（以分钟为单位），在此时间间隔内应将 SNMP 代理的动态配置更改写出到 **/etc/snmpdv3.conf** 配置文件。有效值是 0-10。缺省值是 5。只有 **/etc/snmpdv3.conf** 文件用于 SNMPv3 配置时才与该参数相关。

**-p** *Port*

在此端口侦听 SNMP 信息包。缺省值是端口 161。

**-S**

防止非 root 用户更改 MIB 值。

## 示例

1. 要启动 **snmpd** 守护程序，请输入类似下列命令：

```
startsrc -s snmpd
```

该命令以调试级别 0 启动 **snmpd** 守护程序。

2. 要正常停止 **snmpd** 守护程序，请输入：

```
stopsrc -s snmpd
```

该命令停止守护程序。**-s** 标志指定随后停止的子系统。

3. 要从 **snmpd** 守护程序获取状态，请输入：

```
lssrc -s snmpd
```

该命令返回守护程序的名称、进程标识和守护程序的状态（活动或不活动）。

## 文件

**/etc/services**

包含所需服务的端口指定。如果下列条目还不存在，它们必须存在于 **/etc/services** 文件中：

```
snmp 161/udp
```

```
snmp-trap  
162/udp
```

```
smux 199/tcp
```

**/etc/snmpdv3.conf**

为 **snmpdv3** 代理程序指定配置参数。

**/etc/snmpd.boots**

为 **snmpdv3** 代理程序指定 engineID 和 engineBoots。

/etc/mib.defs

定义 SNMP 代理应当识别与处理的管理信息库（MIB）变量。

## 相关信息

**clsnmp** 命令、**pwchange** 命令、**pwtkey** 命令、**snmpd** 守护程序、**snmpv3\_ssw** 命令。

/etc/clsnmp.conf 文件。

《网络与通信管理》中的『网络管理的 SNMP』章节。

---

## snmpinfo 命令

### 用途

请求或修改简单网络管理协议（SNMP）代理程序管理的管理信息库（MIB）变量的值。

### 语法

#### get 或 next 选项

```
snmpinfo [ -m get | next ] [ -v ] [ -c Community ] [ -d Level ] [ -h HostName ] [ -o ObjectsFile ] ... [ -t Tries ] [ -w Waittime ] Variable. Instance ...
```

#### set 选项

```
snmpinfo -m set [ -v ] [ -c Community ] [ -d Level ] [ -h HostName ] [ -o ObjectsFile ] ... [ -t Tries ] [ -w Waittime ] Variable . Instance= Value ...
```

#### dump 选项

```
snmpinfo -m dump [ -v ] [ -c Community ] [ -d Level ] [ -h HostName ] [ -o ObjectsFile ] ... [ -t Tries ] [ -w Waittime ] [ Variable. Instance ] ...
```

### 描述

**snmpinfo** 命令请求或修改 SNMP 代理程序的一个或多个 MIB 变量的值。该命令只能由具有 root 特权的用户或系统组成员发出。

如果您指定 **get** 选项，**snmpinfo** 命令从 SNMP 代理请求 MIB 的一个或多个变量的信息。

如果您指定 **next** 选项，**snmpinfo** 命令从 SNMP 代理请求关于指定实例后的实例信息。**next** 选项使得在不知道实例限定符的前提下获取 MIB 值成为可能。

如果您指定 **set** 选项，**snmpinfo** 命令修改 SNMP 代理的一个或多个 MIB 变量的值。只有几个 MIB 变量被指定成可读写。作为修改 MIB 变量的副作用，管理 MIB 数据库的代理程序可能进行各种不同的操作对更改 MIB 变量产生副作用。例如，设置 **ifAdminStatus** MIB 变量为 2 通常将关闭网络接口。通过实现管理数据库的 SNMP 代理程序来确定要实施的操作。

如果您指定 **dump** 选项，可使用 **snmpinfo** 命令遍历指定代理程序的整个 MIB 树。如果组作为 *Variable* 参数传递，**snmpinfo** 将遍历 MIB 树的指定路径。

**snmpinfo** 命令具有一个调试工具，该工具可以转储发送和接收信息包的调试信息。该工具使用 **-d** 标志启用。

## 参数

<i>Value</i>	指定 MIB <i>Variable</i> 参数要设置的值。必须为每个变量指定一个值。如果有一个值没有指定，则请求信息包无效。
<i>Variable</i>	以文本格式或数字格式指定在 <i>/etc/mib.defs</i> 文件中定义的指定 MIB 变量的名称。如果 <b>-m</b> 标志的选项是 <b>next</b> 或 <b>dump</b> ，则 <i>Variable</i> 参数可作为 MIB 组指定。
<i>Instance</i>	指定 MIB <i>Variable</i> 参数的实例限定符。如果 <b>-m</b> 标志的选项是 <b>get</b> 或 <b>set</b> ，就需要 <i>Instance</i> 参数。如果 <b>-m</b> 标志的选项是 <b>next</b> 或 <b>dump</b> ，则 <i>Instance</i> 参数是可选的。

### 注:

1. *Variable.Instance* 参数序列中不应有空格。
2. 如果不指定 *Instance* 参数，请勿将一个 . (点) 放置在 *Variable* 参数的后面。

有关更多信息，请查阅定义网络管理的管理信息库 (MIB) 的 RFC 1213 和定义创建 MIB 信息及格式化响应的 SNMP 协议的 RFC 1157。

## 标志

<b>-c</b> <i>Community</i>	指定用于查询 SNMP 代理的团体名。如果不指定 <b>-c</b> 标志，缺省团体名为 <b>public</b> 。								
<b>-d</b> <i>Level</i>	指定 I/O 调试信息的级别。 <i>Level</i> 值可为下列其中之一：  <table><tr><td><b>0</b></td><td>无调试信息。</td></tr><tr><td><b>1</b></td><td>端口绑定和发送及接收的字节数。</td></tr><tr><td><b>2</b></td><td>级别 1 加上接受和发送信息包的十六进制转储。</td></tr><tr><td><b>3</b></td><td>级别 2 加上一个请求与响应信息包的英语版本。</td></tr></table> 如果不指定 <b>-d</b> 标志，缺省调试级别为 0。	<b>0</b>	无调试信息。	<b>1</b>	端口绑定和发送及接收的字节数。	<b>2</b>	级别 1 加上接受和发送信息包的十六进制转储。	<b>3</b>	级别 2 加上一个请求与响应信息包的英语版本。
<b>0</b>	无调试信息。								
<b>1</b>	端口绑定和发送及接收的字节数。								
<b>2</b>	级别 1 加上接受和发送信息包的十六进制转储。								
<b>3</b>	级别 2 加上一个请求与响应信息包的英语版本。								
<b>-h</b> <i>HostName</i>	指定要查询的 SNMP 代理的主机名。如果不指定 <b>-h</b> 标志，缺省的主机名就是用户当前登录的机器的主机名。								
<b>-m</b> <i>Option</i>	指定存取 MIB 变量的方式。  <i>Option</i> 值可为下列其中之一： <b>get</b> 请求关于指定 MIB 变量的信息。 <b>next</b> 请求指定实例之后的实例。 <b>set</b> 修改指定的写访问 MIB 变量。 <b>dump</b> 转储 MIB 树的指定区域。  注: <ol style="list-style-type: none"><li>1. 选项名称可由使其唯一的最小字符数指定。</li><li>2. 如果不指定 <b>-m</b> 标志，缺省方式是 <b>get</b>。</li></ol>								
<b>-o</b> <i>ObjectsFile</i>	指定对象定义文件名，该文件定义了 <b>snmpinfo</b> 命令可以请求的 MIB 对象。如果不指定 <b>-o</b> 标志，缺省对象定义文件名是 <i>/etc/mib.defs</i> 。关于创建该文件的更多信息，请参阅 <b>mosy</b> 命令。可以引用多个 <i>ObjectsFile</i> ，限制条件为包含父定义的文件应当在包含子定义的文件前指定。								
<b>-t</b> <i>Tries</i>	指定终止 (显示 no SNMP response 消息) 前， <b>snmpinfo</b> 命令向 SNMP 代理发送 SNMP 请求的次数。如果不指定 <b>-t</b> 标志，缺省的尝试次数是 3。								
<b>-v</b>	指定以详细方式显示 <b>snmpinfo</b> 命令的输出。如果不指定 <b>-v</b> 标志，则不能以详细方式显示信息。								



**-w** 指定等待从 **snmpd** 代理程序响应的的时间（以秒为单位）。如果没有指定 **-w** 标志，则缺省等待时间是 15 秒。

## 示例

1. 要获取 MIB 变量 **ifDescr.1**（与 **ifIndex.1** 相关的接口）以及 **SysDescr** 的值，请输入：

```
snmpinfo -m get -v sysDescr.0 ifDescr.1
```

在本示例中，**-m get** 标志指定 **snmpinfo** 命令应当检索 MIB 变量 **ifDescr.1**（与 **ifIndex.1** 相关的接口描述）和 **sysDescr.0**（本地主机的系统描述）的值。

2. 要获取 **ipAdEntIfIndex** MIB 变量后面的 MIB 变量（来自 IP 地址 192.100.154.1 指定的主机）的值，请输入：

```
snmpinfo -m next -v 1.3.6.1.2.1.4.20.1.2.192.100.154.1
```

在本示例中，**-m next** 标志指定 **snmpinfo** 命令应当检索 MIB 变量 **ifAdEntIfIndex.192.100.154.1** 的有关信息。

3. 要获取系统组中第一个 MIB 变量的值，请输入：

```
snmpinfo -m next -v -h giants system
```

在本示例中，**-m next** 标志指定 **snmpinfo** 命令应当检索系统组（即 **sysDescr.0**）后的 MIB 变量信息；**-v** 标志指示详细方式；**-h** 标志指示要查询的代理程序是 **giants**；从中检索信息的组是 **system**。

4. 要设置 MIB 变量的值，请输入类似下列命令：

```
snmpinfo -m set -v -h giants -c monitor -t 2 ifAdminStatus.1=2
```

在本示例中，MIB **ifAdminStatus** 变量（与已知的 **giants** 主机上的 **ifIndex.1** 相关的接口）设置为 2，或小于 2。**-c** 标志为主机指定团体。**-t 2** 标志指定在终止 **snmpinfo** 命令前，若没有从 SNMP 代理接收到响应，则向 SNMP 代理程序发送两次 SNMP 请求。

5. 要以详细方式转储 MIB 树的组，请输入类似下列命令：

```
snmpinfo -m dump -v interfaces
```

在本示例中，**interfaces** 组以详细方式转储。

6. 要转储整个 MIB 树，请输入：

```
snmpinfo -m dump
```

## 文件

**/etc/mib.defs** 定义 SNMP 代理应当识别与处理的管理信息库（MIB）变量。

## 相关信息

**mosy** 命令。

*AIX 5L Version 5.3 Communications Programming Concepts* 中的 Understanding the Simple Network Management Protocol (SNMP)、Using the Management Information Base (MIB) Database 和 Understanding the Management Information Base (MIB)。

---

## snmpmibd 守护程序

### 用途

作为后台进程启动 **snmpmibd** dpi2 子代理守护程序。

### 语法

```
snmpmibd [ -f File ] [ -d [Level] ] [ -h HostName ] [ -c Community ]
```

### 描述

**snmpmibd** 命令启动 **snmpmibd** dpi2 子代理程序。该命令只能由具有 root 特权的用户或系统组成员发出。

**snmpmibd** 守护程序符合 RFC 1592 定义的标准“简单网络管理协议分布式协议接口” V2.0。它作为一个 dpi2 子代理通过 RFC1592 3.1 节定义的 dpiPortForTCP.0 (1.3.6.1.4.1.2.2.1.1.1.0) 与 dpi2 代理程序通信。

管理信息库 (MIB) 由 RFC 1155 定义。正在管理的特定 MIB 变量 **snmpmibd** 由下面的 RFC 定义:

#### RFC 1213

MIB-II

#### RFC 1229

类属接口 MIB 的扩展

#### RFC 1231

IEEE 802.5 令牌环 MIB

#### RFC 1398

类似以太网接口类型 MIB

#### RFC 1512

FDDI MIB

注: RFC 1213 中定义的“**system**”和“**snmp**”组不是使用 **snmpdmibd** 守护程序实现的。而是用 **snmpdv3** 代理程序实现的。

系统启动期间调用 **/etc/rc.tcpip** shell 脚本时, 通常执行 **snmpmibd** 守护程序。

应当使用系统资源控制器 (SRC) 控制 **snmpmibd** 守护程序。建议不要在命令行输入 **snmpmibd**。

请使用下列 SRC 命令操作 **snmpmibd** 守护程序:

#### **startsrc**

启动子系统、子系统组或子服务器。

#### **stopsrc**

停止子系统、子系统组或子服务器。

#### **refresh**

使子系统或子系统组重新读取相应的配置文件。

**lssrc** 获取子系统、子系统组或子服务器的状态。如果发出长状态格式 **lssrc** 命令的用户不是 root 用户, 则不会显示团体名信息。

## 标志

**-c** *Community*

使用指定的共用名。如果不指定 **-c** 标志，缺省的团体名是 **public**。

**-d** [*Level*]

指定跟踪 / 调试级别。级别有：

**8**        DPI 级别 1

**16**        DPI 级别 2

**32**        内部 1 级

**64**        内部 2 级

**128**       内部 3 级

给多个跟踪级别添加数值。

如果指定 **-d** 标志而不指定 *Level*，则缺省级别为 56。

**-f** *File*

如果不指定 **-d** 标志，则缺省级别为 0。

非缺省配置文件。如果不指定 **-f** 标志，缺省配置文件为 **/etc/snmpmibd.conf**。有关该文件格式的信息，请参阅 **/etc/snmpmibd.conf** 文件。

**-h** *HostName*

向指定主机发送请求。如果不指定 **-h** 标志，缺省的目的地主机是回送 (127.0.0.1)。

## 示例

1. 要启动 **snmpmibd** 守护程序，请输入类似下列命令：

```
startsrc -s snmpmibd -a "-f /tmp/snmpmibd.conf"
```

该命令启动 **snmpmibd** 守护程序并从 **/tmp/snmpmibd.conf** 中读取配置文件。

2. 要正常停止 **snmpmibd** 守护程序，请输入：

```
stopsrc -s snmpmibd
```

该命令返回守护程序的名称、进程标识和守护程序的状态（活动或不活动）。

3. 要从 **snmpmibd** 守护程序中获取长状态信息，请输入：

```
lssrc -ls snmpmibd
```

如果您是 **root** 用户，该长状态报告列出 **/etc/snmpmibd.conf** 中的配置参数。

## 文件

**/etc/snmpmibd.conf**

定义 **snmpmibd** 命令的配置参数。

**/etc/mib.defs**

定义 SNMP 代理程序应当识别和处理的管理信息库 (MIB) 变量。

## 相关信息

**hostmibd** 命令、**snmpdv3** 守护程序。

---

## snmptrap 命令

### 用途

生成一个通知（陷阱），用指定的消息向 SNMP 管理器报告事件。

### 语法

```
snmptrap [ -a Host ] [ -h TargetHost ] [ -c Community ] [-d ] -m Message
```

### 描述

生成一个通知（陷阱），用指定的消息向 SNMP 管理器报告事件。

### 标志

<b>-a</b> <i>Host</i>	指定连接至指定的主机上的 SNMP 代理程序。如果不指定 <b>-a</b> 标志，缺省主机为本地主机。 <i>Host</i> 可以是一个 IP 地址或一个主机名。
<b>-c</b> <i>Community</i>	指定要使用的团体名。必须在 SNMP V3 的 <code>/etc/snmpdv3.conf</code> 或 SNMP V1 的 <code>/etc/snmpd.conf</code> 中设置该团体，并且该团体至少具有对运行在指定主机或本地主机上的 SNMP 代理的读访问权限。如果不指定 <b>-c</b> 标志，缺省团体名是“public”。
<b>-d</b>	启用调试工具
<b>-h</b> <i>TargetHost</i>	指定陷阱消息将发送至的目标网络管理器主机。它不同于 <b>-a</b> 标志。 <b>-a</b> 标志指定了主机，其中 AIX SNMP 代理（ <code>snmp</code> ）必须正在运行，并且 SNMP 代理将此陷阱转发到网络管理器中。然而， <b>-h</b> 标志不需要 AIX SNMP 代理将陷阱消息转发到网络管理器，它将陷阱直接发送到管理器。如果没有 <b>-h</b> 和 <b>-a</b> 标志，陷阱将发送到本地主机上的 AIX SNMP 代理。
<b>-m</b> <i>Message</i>	定义 <code>snmptrap</code> 命令将发送的消息。 <i>Message</i> 指定陷阱将保留的信息。该信息是文本格式。 <b>-m</b> 标志必须是最后指定的标志。

### 退出状态

- 0 陷阱信息正确发出。
- 1 这表明处理中发生问题。

### 示例

- 要向本地主机上运行的 SNMP 代理发送带有消息“hello world”的陷阱，请输入以下内容：

```
snmptrap -m hello world
```

**注：**团体“public”必须具有对本地主机上运行的 SNMP 代理的读访问权。有关详细信息，请参考 SNMP 配置文档。

- 要向远程主机 `blah` 上运行的 SNMP 代理发送带有团体名称“community1”和消息“hello world”的陷阱，请输入以下内容：

```
snmptrap -c community1 -h blah -m hello world
```

**注：**团体“community1”必须具有对主机“blah”上运行的 SNMP 代理的读访问权。有关详细信息，请参考 SNMP 配置文档。

- 要将陷阱发送到在 Linux® 平台上运行的网络管理器，其中主机名为 `nehcyg`，请输入以下内容：

```
snmptrap -h nehcyg -m hello world
```

## 文件

`/etc/snmpdv3.conf`  
`/etc/snmpd.conf`

包含 SNMP V3 代理程序的配置文件。  
包含 SNMP V1 代理程序的配置文件。

## 相关信息

`snmpdv3` 守护程序、`snmpdv1` 守护程序。

《网络与通信管理》中的『网络管理的 SNMP』章节

---

## snmpv3\_ssw 命令

### 用途

在非加密 `snmpdv3` 代理程序、加密 `snmpdv3` 代理程序和 `snmpdv1` 代理程序间切换符号链接。

### 语法

`snmpv3_ssw [ -e | -n | -1 ]`

### 描述

在非加密 `snmpdv3` 代理程序、加密的 `snmpdv3` 代理程序和 `snmpdv1` 代理程序间切换符号链接，然后启动新选择的 SNMP 代理。用户可选择运行哪个版本的 SNMP 代理。

例如，如果当前运行的 SNMP 代理程序是加密的 `snmpdv3` 代理程序，那么在机器上运行的实际可执行的 SNMP 代理是 “`/usr/sbin/snmpdv3e`”。机器上的符号链接是：

- `/usr/sbin/snmpd --> /usr/sbin/snmpdv3e`
- `/usr/sbin/clsnpmp --> /usr/sbin/clsnmpe`

用户使用 `-n` 选项运行 `/usr/sbin/snmpv3_ssw` 命令后，如果用户选择切换至非加密的 `snmpdv3` 代理程序，那么在机器上运行的实际可执行的 `snmp` 代理是 “`/usr/sbin/snmpdv3ne`”。机器上的符号链接将更改为：

- `/usr/sbin/snmpd --> /usr/sbin/snmpdv3ne`
- `/usr/sbin/clsnpmp --> /usr/sbin/clsnmpne`

### 标志

- |                 |                                      |
|-----------------|--------------------------------------|
| <code>-e</code> | 切换至 <code>snmpdv3</code> 代理程序的加密版本。  |
| <code>-n</code> | 切换至 <code>snmpdv3</code> 代理程序的非加密版本。 |
| <code>-1</code> | 切换至 <code>snmpdv1</code> 代理程序。       |

### 示例

1. 要切换至 `snmpdv3` 代理程序的加密版本，请输入：

```
/usr/sbin/snmp3_ssw -e
```

## 相关信息

`clsnmp` 命令、`hostmibd` 命令、`snmpdv1` 守护程序、`snmpdv3` 守护程序。

`/etc/clsnmp.conf` 文件格式、`/etc/snmpd.conf` 文件格式和 `/etc/snmpdv3.conf` 文件格式。

---

## sno 命令

### 用途

提供 SNOBOL 解释器。

### 语法

`sno` [*File* ...]

### 描述

`sno` 命令提供不同于标准 SNOBOL 的 SNOBOL 编译器和解释器。它读取指定的文件和标准输入，并通过包含 `end` 标签的语句编译全部输入。剩余部分对 `syspit` 伪变量可用。

`sno` 命令在以下方面不同于标准 SNOBOL:

- 没有无锚定搜索。为获得同样的效果，请使用下列类似行:

```
a ** b           为 b 生成无锚定搜索。  
a **x* b = x c   生成无锚定指定。
```

- 没有向后引用。

```
x = "abc"
```

a \*\*x\* x 为 `abc` 生成无锚定搜索。

- 使用（非唯一的）`define` 标签在编译时完成函数声明。从 `define` 标签后的语句开始执行函数调用。函数不能在运行时定义，并且要预先试用所使用的 `define` 名称。不提供参数以外的自动变量。例如:

```
define f()  
define f(a, b, c)
```

- 除 `define` 外的所有标签（甚至 `end`）必须有非空语句。
- 标签、函数和变量必须具有唯一的名称。特别是 `end` 的非空语句不能只命名成标签。
- 如果 `start` 是程序中的标签，就从这里开始执行程序。如果不是，就从第一个可执行语句开始执行。`define` 标签不是可执行语句。
- 没有内置函数。
- 不需要进行圆括号计算。使用正常优先级。因此，算术操作符 `\`（反斜杠）和 `*`（星号）必须使用空格分开。
- 赋值的右边必须非空。
- 可以使用 `'`（单引号）或 `"`（双引号）作为文字引号。
- `pseudo` 变量 `syspnt` 不可用。

## 示例

要使用 **sno** 命令运行文件 `test.s`，并将输出指向文件输出，请输入：

```
sno < test.s > output
```

## 文件

`/usr/bin/sno` 包含 **sno** 命令。

## 相关信息

**awk** 命令。

---

## sodebug 命令

### 用途

设置或取消设置套接字的套接字调试标志（**SO\_DEBUG** 套接字选项）和跟踪级别。

### 语法

```
sodebug [ -h ] [ -l [ level ] ] [ -p pid | -s sockaddr [ -t type ] ]
```

### 描述

**sodebug** 命令设置、取消设置或列出活动套接字的套接字调试标志和跟踪级别

如果为某个套接字设置了套接字调试标志（也称为 **SO\_DEBUG** 套接字选项），就可以使用 **trace** 命令来跟踪该套接字上的事件。

您可以使用 **-l** 选项来对已存在于系统中的套接字设置套接字调试标志。**-l** 选项还设置了给定套接字的跟踪级别。

如果 **sodebug** 命令运行时不带有任何选项，则将显示每个活动套接字的套接字调试标志状态和跟踪级别。

**trace** 和 **trpt** 命令根据跟踪级别来收集信息。

下表描述了基于跟踪挂钩标识 25 A（TCPDBG）的跟踪级别所收集的信息：

表 1.

	min	normal	detail
tcp_debug 数据 (td_time、td_act、td_ostate、 td_tcb、family 和 td_req)		X	X
tcip 头		X	X
tcpcb 的地址		X	X
所有 tcpcb 字段			X
套接字的地址		X	X
所有套接字字段			X

您还可以设置或取消设置套接字调试标志和跟踪级别，如下所述：

1. 以下命令对随后在系统上创建的所有套接字启用套接字调试标志：

```
no -o sodebug=1
```

2. 您可以在 `inetd.conf` 中某个服务的 `wait/nowait` 字段中指定 `IDEBUG[=level]`，以打开对特定服务的套接字调试。您可以将跟踪级别设置为 **min**、**normal** 或 **detail**。如果没有指定级别，则缺省级别为 **normal**。
3. 您可以使用 `no` 命令的 `sodebug_env` 参数并在进程环境中指定 `export SODEBUG=level` 来将某个进程创建的所有后续套接字的套接字调试设置为打开或关闭。您可以将跟踪级别设置为 **min**、**normal** 或 **detail**。

## 标志

<b>-h</b>	显示 <code>sodebug</code> 命令的帮助。
<b>-l</b> [ <i>level</i> ]	指定跟踪级别。级别的有效值为 <b>none</b> 、 <b>min</b> 、 <b>normal</b> 和 <b>detail</b> 。如果没有指定级别，则缺省跟踪级别为 <b>normal</b> 。
<b>-p</b> <i>pid</i>	指定进程的进程标识。
<b>-s</b> <i>sockaddr</i>	通过套接字地址、套接字的 <code>inpcb</code> 地址或套接字的 <code>tcpcb</code> 地址指定套接字。
<b>-t</b> <i>type</i>	指定由 <code>-s sockaddr</code> 选项指定的地址的类型。有效值为 <b>socket</b> 、 <b>inpcb</b> 和 <b>tcpcb</b> 。缺省值为 <b>socket</b> 。

## 安全性

您必须具有引导权限才能运行 `sodebug` 命令。

## 示例

1. 要列出套接字 `f100090002d0a800` 的调试标志和套接字跟踪级别，请输入：

```
sodebug -s f100090002d0a800
```

输出与以下示例类似：

```
socket address : f100090002d0a800 , sodebug flag : 0 , trace level : none(0)
```

2. 要将跟踪级别设置为 `normal` 并将调试标志设置为 `1`，请输入：

```
sodebug -s f100090002d0a800 -l normal
```

输出与以下示例类似：

```
Setting new values for trace level and debug flag  
socket address : f100090002d0a800 , sodebug flag : 1 , trace level : normal(3)
```

## 相关信息

`trace` 守护程序、`trpt` 命令。

---

## soelim 命令

### 用途

在 `nroff` 命令文件中处理 `.so` 请求。

### 语法

```
soelim [ File ... | - ]
```



## 描述

当请求出现在输入行开始处时，**soelim** 命令读取指定文件或标准输入，并执行 **nroff** 命令和 **troff** 命令请求所指定的包含（以 `.so filename` 格式）。任何 ASCII 空格和 ASCII 制表符的组合可跟在 `.so` 请求后文件名前。文件名后不应跟任何字符。

**soelim** 命令很有用，这是因为一些命令，比如 **tbl** 命令在处理中不能正常执行文件包含。

指定 `-`（减号）标志时，包含与标准输入相符的文件名。

## 标志

`-` 指示与标准输入相符的文件名。

注：可使用 `'`（单引号）代替 `.`（句点）来取消文件包含，如下所示：

## 参数

*File* 指定命令执行包含的文件。缺省值是标准输入。

```
'so /usr/share/lib/tmac/tmac.s
```

## 示例

下面是使用 **soelim** 命令的样本：

```
soelim exum?.n | tbl | nroff -ms -Tlp | col -Tlp | pg
```

在本示例中，您使用 **soelim** 命令预处理文件包含（`.so`）请求。然后将输出传递至 **tbl** 命令。这易于将可包含的单独文件中的表格置入形成较大的文档。

## 相关信息

**colcrt** 命令、**nroff** 命令、**tbl** 命令和 **troff** 命令。

---

## sort 命令

### 用途

排序文件、对已排序的文件进行合并，并检查文件以确定它们是否已排序。

### 语法

```
sort [ -A ] [ -b ] [ -c ] [ -d ] [ -f ] [ -i ] [ -m ] [ -n ] [ -r ] [ -u ] [ -o OutFile ] [ -t Character ] [ -T Directory ] [ -y [ Kilobytes ] ] [ -z RecordSize ] [ + [ FSkip ] [ .CSkip ] [ b ] [ d ] [ f ] [ i ] [ n ] [ r ] ] [ - [ FSkip ] [ .CSkip ] [ b ] [ d ] [ f ] [ i ] [ n ] [ r ] ] ... [ -k KeyDefinition ] ... [ File ... ]
```

### 描述

**sort** 命令对 *File* 参数指定的文件中的行排序，并将结果写到标准输出。如果 *File* 参数指定多个文件，那么 **sort** 命令将这些文件连接起来，并当作一个文件进行排序。`-`（减号）代替文件名指定标准输入。如果您不指定任何文件名，那么该命令对标准输入排序。可以使用 `-o` 标志指定输出文件。

如果不指定任何标志, **sort** 命令基于当前语言环境的整理顺序对输入文件的所有行排序。

## 排序关键字

排序关键字是输入行的一部分, 由字段号和列号指定。字段是输入行的组成部分, 由字段分隔符分隔。缺省字段分隔符是由一个或多个连续空格字符组成的序列。然而, 这些空白字符被看作以下用于排序的字段的一部分。您可以指定 **-b** 选项来忽略这些开头的空白字符。使用 **-t** 标志可指定不同的字段分隔符。在 C 语言和英语语言环境下, 制表符和空格字符都是空格符。

使用排序关键字时, **sort** 命令首先根据第一个排序关键字的内容对所有行排序。然后, 根据第二个排序关键字的内容, 对所有第一个排序关键字相同的行排序, 如此进行下去。按照排序关键字在命令行中出现的顺序给它们编号。如果两行对所有排序关键字的排序都相同, 则对全部行依据当前语言环境的整理顺序进行比较。

对字段中的列进行编号时, 缺省字段分隔符中的空格符将作为后继字段计数。前导空格不计作第一字段的一部分, **-t** 标志指定的字段分隔符将不作为字段的一部分计数。可使用 **-b** 标志忽略前导空格符。

可使用下列两种方法定义排序关键字:

- **-k** *KeyDefinition*
- *FSkip.CSkip* (废弃版本)。

## 使用 **-k** 标志定义排序关键字

**-k** *KeyDefinition* 标志采用下列形式:

**-k** [ *FStart* [ *.CStart* ] ] [ *Modifier* ] [ , [ *FEnd* [ *.CEnd* ] ] [ *Modifier* ] ]

排序关键字包括所有以 *FStart* 变量指定的字段和 *CStart* 变量指定的列开头的字符及以 *FEnd* 变量指定的字段和 *CEnd* 变量指定的列结束的字符。如果不指定 *Fend*, 就假定行的最后一个字符。如果不指定 *CEnd*, 就假定 *FEnd* 字段的最后一个字符。*KeyDefinition* 变量中的任何字段号或列号都可以省略。缺省值为:

<i>FStart</i>	行开头
<i>CStart</i>	字段第一列
<i>FEnd</i>	行结束
<i>CEnd</i>	字段最后一列

如果字段间有任意空格, **sort** 就将它们看作分隔的字段。

*Modifier* 变量的值可以是字母 **b**、**d**、**f**、**i**、**n** 或 **r** 中的一个或多个。修饰符仅应用于它们连接的字段定义, 与同一字母的标志有同样的效果。修饰符字母 **b** 仅应用于其连接的字段定义的末尾。例如:

**-k 3.2b,3r**

指定排序关键字, 从第三字段的第二非空格列开始并扩展至第三字段结束, 对这个关键字的排序以逆向整理顺序完成。如果 *FStart* 变量和 *CStart* 变量在命令行末尾以外或在 *FEnd* 变量和 *CEnd* 变量之后, 那么该排序关键字被忽略。

排序关键字也可用下列方式指定:

**[+*FSkip1*] [*.CSkip1*] [*Modifier*] ] [-*FSkip2*] [*.CSkip2*] [*Modifier*]**

**+*FSkip1*** 变量指定跳过的字段数以到达排序关键字第一字段, **+*CSkip*** 变量指定在该字段中跳过的列数以到达排序关键字第一个字符。**-*FSkip*** 变量指定跳过的字段数以到达排序关键字后的第一个字符, **-*CSkip*** 变量指定在该字段中跳过的列数。可以省略任何要跳过的字段和列。缺省值为:

<i>FSkip1</i>	行开头
<i>CSkip1</i>	零
<i>FSkip2</i>	行结束
<i>CSkip2</i>	零

*Modifier* 变量指定的修改量与 **-k** 标志关键字排序定义中的相同。

因为 **+FSkip1.CSkip1** 变量指定到达排序关键字前要跳过多少字段和列，所以这些变量指定的字段号和列号通常比排序关键字本身的字段号和列号小 1。例如：

```
+2.1b -3r
```

指定排序关键字，从第三字段的第二非空格列开始并扩展至第三字段结束，对这个关键字的排序以逆向整理顺序完成。语句 **+2.1b** 指定跳过两个字段，然后跳过后导空格和另一列。如果 **+FSkip1.CSkip1** 变量在命令行末尾以外或在 **-FSkip2.CSkip2** 变量之后，则忽略该排序关键字。

注：一行的最大字段数为 32。

## 标志

注：在任何排序关键字定义前出现的 **-b**、**-d**、**-f**、**-i**、**-n** 或 **-r** 标志应用于所有排序关键字。**-b**、**-d**、**-f**、**-i**、**-n** 或 **-r** 标志都不能单独出现在 **-k** *KeyDefinition* 之后；如果它们作为修饰符连接 *KeyDefinition* 变量，那么就只应用于连接排序关键字。如果这些标志之一跟随在 **+Fskip.Cskip** 或 **-Fskip.Cskip** 排序关键字定义后，那么该标志只能用于此排序关键字。

<b>-A</b>	使用 ASCII 整理顺序代替当前语言环境的整理顺序在逐字节的基础上排序。
<b>-b</b>	忽略前导空格和制表符，找出字段的第一或最后列。
<b>-c</b>	检查输入是否已按照标志中指定的排序规则进行排序。如果输入文件排序不正确，就返回一个非零值。
<b>-d</b>	使用字典顺序排序。比较中仅考虑字母、数字和空格。
<b>-f</b>	比较前将所有小写字母改成大写字母。
<b>-i</b>	比较中忽略所有非显示字符。
<b>-k</b> <i>KeyDefinition</i>	指定排序关键字。 <i>KeyDefinition</i> 选项的格式为：

```
[ FStart [ .CStart ] ] [ Modifier ] [ , [ FEnd [ .CEnd ] ] [ Modifier ] ]
```

排序关键字包括所有以 *FStart* 变量指定的字段和 *CStart* 变量指定的列开头的字符及以 *FEnd* 变量指定的字段和 *CEnd* 变量指定的列结束的字符。*Modifier* 变量的值可以是 **b**、**d**、**f**、**i**、**n** 或 **r**。修饰符与同一字母的标志等价。

<b>-m</b>	只合并多个输入文件；假设输入文件已经排序。
<b>-n</b>	按算术值对数字字段排序。数字字段可包含前导空格、可选减号、十进制数字、千分位分隔符和可选基数符。对包含任何非数字字符的字段进行数字排序会出现无法预知的结果。
<b>-o</b> <i>OutFile</i>	将输出指向 <i>OutFile</i> 参数指定的文件，而不是标准输出。 <i>OutFile</i> 参数值可以与 <i>File</i> 参数值相同。
<b>-r</b>	颠倒指定排序的顺序。
<b>-t</b> <i>Character</i>	指定 <i>Character</i> 为单一的字段分隔符。
<b>-u</b>	禁止按照排序关键字和选项的所有等同排序（每一组行中一行除外）。
<b>-T</b> <i>Directory</i>	将创建的所有临时文件放入 <i>Directory</i> 参数指定的目录中。
<b>-y</b> [ <i>Kilobytes</i> ]	用 <i>Kilobytes</i> 参数指定的主存储的千字节数启动 <b>sort</b> 命令，并根据需要增加存储量。（如果 <i>Kilobytes</i> 参数指定的值小于最小存储站点或大于最大存储站点，就以这个最小存储站点或最大存储站点取代）。如果省略 <b>-y</b> 标志， <b>sort</b> 命令以缺省的存储大小启动。 <b>-y0</b> 标志用最小存储启动，而 <b>-y</b> 标志（不带 <i>Kilobytes</i> 值）用最大存储启动。 <b>sort</b> 命令使用的存储量显著地影响性能。以大存储量对小文件排序将很浪费。

**-z RecordSize**

如果正在排序的任一行大于缺省的缓冲区大小，要防止出现异常终止。指定 **-c** 或 **-m** 标志时，省略排序阶段，使用系统的缺省缓冲大小。如果已排序行超出这一大小，排序异常终止。**-z** 选项指定排序阶段最长行的记录，因而可在合并阶段分配足够的缓冲区。**RecordSize** 必须指明等于或大于要合并的最长行的字节值。

## 退出状态

该命令返回以下退出值：

- 0 所有输入文件成功输出，或指定了 **-c** 且正确排序了输入文件。
- 1 在 **-c** 选项下，文件没有按指定排序，或如果指定 **-c** 和 **-u** 选项，找到了两个具有相同关键字的输入行。
- >1 发生错误。

## 示例

1. 要在 **LC\_ALL**、**LC\_COLLATE** 或 **LANG** 环境变量设置为 **En\_US** 的情况下排序 **fruits** 文件，请输入：

```
LANG=En_US sort fruits
```

此命令序列显示以升序词典顺序排序的 **fruits** 文件的内容。每一列的字符，包括空格、数字和特殊字符都经一一比较。例如，如果 **fruits** 文件包含文本：

```
banana
orange
Persimmon
apple
%%banana
apple
ORANGE
```

**sort** 命令显示：

```
%%banana
ORANGE
Persimmon
apple
apple
banana
orange
```

在 ASCII 整理序列中，%（百分号）在大写字母前，大写字母在小写字母前。如果您当前的语言环境指定 ASCII 之外的字符集，结果可能不同。

2. 要以字典顺序排序，请输入：

```
sort -d fruits
```

此命令序列排序和显示 **fruits** 文件的内容，并且只比较字母、数字和空格。如果 **fruits** 文件与示例 1 相同，那么 **sort** 命令显示：

```
ORANGE
Persimmon
apple
apple
%%banana
banana
orange
```

**-d** 标志忽略 %（百分号）字符，因为它不是个字母、数字或空格。（即 %%banana 被 banana 取代）。

3. 要将包含大写字母和具有类似小写行的特殊字符行分组，请输入：

```
sort -d -f fruits
```

**-d** 标志忽略特殊字符，**-f** 标志忽略大小写差异。将 **LC\_ALL**、**LC\_COLLATE** 或 **LANG** 环境变量设置为 C 的情况下，`fruits` 文件的输出结果变为：

```
apple
apple
%%banana
banana
ORANGE
orange
Persimmon
```

4. 要除去重复行排序，请输入：

```
sort -d -f -u fruits
```

**-u** 标志告诉 **sort** 命令除去重复的行，使文件中的每一行唯一。此命令序列显示：

```
apple
%%banana
ORANGE
Persimmon
```

不仅除去重复的 `apple`，而且也除去了 `banana` 和 `ORANGE`。除去这些是因为 **-d** 标志忽略 `%%` 这个特殊字符，**-f** 标志忽略大小写差异。

5. 要如例 4 那样排序，除去重复的实例（除非是大写字母或标点不同），请输入：

```
sort -u +0 -d -f +0 fruits
```

输入 `+0 -d -f` 完成的排序与示例 3 中 `-d -f` 的排序类型相同，`+0` 进行另一项比较以区分不一样的行。这防止 **-u** 标志将它们除去。

示例 1 所示的 `fruits` 文件中，添加的 `+0` 将 `%%banana` 与 `banana` 及 `ORANGE` 与 `orange` 区分开来。然而，`apple` 的两个实例是相同的，所以其中之一被删除。

```
apple
%%banana
banana
ORANGE
orange
Persimmon
```

6. 要指定分隔字段的字符，请输入：

```
sort -t: +1 vegetables
```

此命令序列排序 `vegetables` 文件，对每一行上第一个冒号后的文本进行比较。`+1` 告诉 **sort** 命令忽略第一字段，从第二字段的开始到该行的结束进行比较。**-t:** 标志告诉 **sort** 命令冒号分隔字段。如果 `vegetables` 包含：

```
yams:104
turnips:8
potatoes:15
carrots:104
green beans:32
radishes:5
lettuce:15
```

那么，将 **LC\_ALL**、**LC\_COLLATE** 或 **LANG** 环境变量设置为 C 的情况下，**sort** 命令将显示：

```
carrots:104
yams:104
lettuce:15
potatoes:15
green beans:32
radishes:5
turnips:8
```

注意数字没有按照数字排序。当用字典式分类从左至右比较每一个字符时出现这种情况。换句话说，3 在 5 之前，所以 32 在 5 之前。

7. 要排序数字，请输入：

```
sort -t: +1 -n vegetables
```

此命令序列按照第二个字段对 `vegetables` 文件进行数字排序。如果 `vegetables` 文件与示例 6 中的相同，那么 **sort** 命令将显示：

```
radishes:5
turnips:8
lettuce:15
potatoes:15
green beans:32
carrots:104
yams:104
```

8. 要对多个字段排序，请输入：

```
sort -t: +1 -2 -n +0 -1 -r vegetables
```

或

```
sort -t: -k2,2 n -k1,1 r vegetables
```

此命令序列对第二字段（+1 -2 -n）进行数字排序。在这个顺序中，它以逆字母顺序（+0 -1 -r）对第一字段排序。将 **LC\_ALL**、**LC\_COLLATE** 或 **LANG** 环境变量设置为 C 的情况下，输出将类似于：

```
radishes:5
turnips:8
potatoes:15
lettuce:15
green beans:32
yams:104
carrots:104
```

此命令按数字顺序对行排序。当两行数字相同时，它们以逆字母顺序出现。

9. 要使用排序的文本替换原始文件，请输入：

```
sort -o vegetables vegetables
```

此命令序列将排序输出存入 `vegetables` 文件（`-o vegetables`）。

## 文件

**/usr/bin/sort**            包含 **sort** 命令。

**/var/tmp**                **sort** 命令处理期间的临时空间。

**/usr/tmp**                如果不能在 **/var/tmp** 中创建文件，是 **sort** 命令处理期间的临时空间。

**/tmp**                     如果不能在 **/var/tmp** 或 **/usr/tmp** 中创建文件，是 **sort** 命令处理期间的临时空间。

## 相关信息

**comm** 命令、**join** 命令和 **uniq** 命令。

《操作系统与设备管理》中的『文件』和『输入和输出重定向』。

《AIX 5L V5.3 本地语言支持指南和参考大全》中的『本地语言支持』。

---

## sortbib 命令

### 用途

对参考文献资料数据库排序。

### 语法

**sortbib** [ **-sKeys** ] [ *Database ...* ]

### 描述

**sortbib** 命令按照用户指定的关键字对包含 **refer** 命令关键字的记录文件排序。这些记录可由空行分隔或由 . (句点、左方括号) 和 ] (句点、右方括号) 定界符包围，但两种方法不能混用。**sortbib** 命令通读 *Database* 参数指定的每个数据库，并选出的关键字段，对它们分别进行排序。排序的关键字段包含文件指针、字节偏移量和相应记录的长度。这些记录用磁盘寻道和读取来传送的，因而 **sortbib** 命令无法用于以管道方式读取标准输入。

缺省情况下，**sortbib** 命令首先对 %A 字段和 %D 字段按字母顺序排序，其中包含高级作者及日期。

**sortbib** 命令按 %A 字段中的最后一个字排序，该字假设是作者的姓。如果名字以逗号结束（比如 jr. 或 ed.），就忽略最后位置的字。通过使用 **nroff** 命令约定 \0 取代空格字符对具有两个字的姓的作者或具有非常规结构的名字的作者排序。除非从第一个而不是最后一个字开始排序，对 %Q 字段的指定类似于 %A 字段。

注：无作者字段的记录按照标题排序。

**sortbib** 命令按 %D 行的最后一个字排序，通常是年。当按 %T 或 %J 字段中的标题排序时，忽略前导文章。忽略的文章对语言环境来说是特定的，它在特定的语言环境 **refer message catalog** 中指定。在此目录中，文章包含在单一消息中。每篇文章由任意数量的 ASCII 空格或制表符分隔。如果记录中缺少排序意义的字段，那么 **sortbib** 命令就将此记录放在其他包含该字段的记录之前。

一次排序的数据库不能超过 16 个。超过 4096 个字符的记录会被截断。

*Database* 参数包含 **refer** 命令关键字，**sortbib** 命令按用户指定的关键字对这些关键字排序。

### 标志

**-sKeys** 指定排序依据的字段关键字

### 示例

1. 按作者、标题和日期排序:

```
sortbib -sATD Database
```

2. 按作者和日期排序:

## 文件

**/tmp/SbibXXXXX** 包含临时文件。  
**/usr/bin/sort** 包含 **sort** 命令。

## 相关信息

**addbib** 命令、**indxbib** 命令、**lookbib** 命令、**refer** 命令、**roffbib** 命令和 **sort** 命令。

《AIX 5L V5.3 本地语言支持指南和参考大全》中的 **refer** 消息目录。

---

## sortm 命令

### 用途

对消息排序。

### 语法

**sortm** [ **+Folder** ] [ *Messages* ] [ **-datefield** *Field* ] [ **-noverbose** | **-verbose** ]

### 描述

**sortm** 命令根据消息的 **Date:** 字段对它们排序并以数字 1 开始对它们重新连续编号。文件夹中未指定要排序的消息放在已排序消息后。如果不能分析日期字段，**sortm** 命令将显示一条消息。

要指定 **Date:** 字段以外的字段，请指定 **-datefield** 标志。如果指定一个文件夹，那么该文件夹就成为当前文件夹。当前消息仍然是指定文件夹的当前消息，即使排序过程中消息被移动。

### 标志

**-datefield** *Field* 指定排序中使用的头字段。缺省字段是 **Date:** 字段。  
**+Folder** 指定要排序其消息的文件夹。缺省值是当前文件夹。  
**-help** 列出命令语法、可用的参数选项（开关）和版本信息。  
**注：** 对于消息处理程序（MH），必须完整拼写此标志的名称。



*Messages*

指定要排序的消息。使用以下参考来指定消息:

*Number* 消息号。

*Sequence*

用户指定的一组消息。可识别值为:

**all** 文件夹中的所有消息。这是缺省值。

**cur** 或 **.** (句点)  
当前消息。

**first** 文件夹中的第一条消息。

**last** 文件夹中的最后一条消息。

**next** 当前消息的下一条消息。

**prev** 当前消息的上一条消息。

**-noverbose**

防止排序期间显示信息。此标志是缺省值。

**-verbose**

排序期间显示信息。此信息允许您监视涉及的步骤。

## 概要文件条目

可在 *UserMhDirectory/.mh\_profile* 文件中找到以下条目:

Current-Folder: 设置缺省当前文件夹。  
Path: 指定 *UserMhDirectory*。

## 示例

1. 要按日期对当前文件夹中的所有消息排序, 请输入:

```
sortm
```

2. 要按照日期对 *easter* 文件夹中的消息 5 至 10 进行排序, 请输入:

```
sortm +easter 5-10
```

## 文件

**\$HOME/.mh\_profile** 包含消息处理程序 (MH) 用户概要文件。  
**/usr/bin/sortm** 包含 **sortm** 命令。

## 相关信息

**folder** 命令。

**.mh\_alias** 文件格式和 **.mh\_profile** 文件格式。

《网络与通信管理》中的『邮件应用程序』。

---

## spell 命令

### 用途

查找英语语言拼写错误。

### 语法

```
spell [ -b ] [ -i ] [ -l ] [ -v ] [ -x ] [ -d HashList ] [ -h HistoryList ] [ -s HashStop ] [ + WordList ] [ File ... ]
```

### 描述

**spell** 命令读取 *File* 变量指示的文件中的词，并将它们与拼写列表中的词比较。与拼写列表不匹配的词或拼写列表中的词派生的词（通过应用某种词形变化、前缀和后缀而生成）将写到标准输出。如果不指定文件名，**spell** 命令从标准输入读取。

与 **deroff** 命令一样，**spell** 命令也忽略 **troff**、**tbl** 和 **eqn** 代码。

拼写列表的覆盖不均衡。您应当创建文件中使用的特殊词字典。您的字典是包含排序词汇列表的文件，一行一个词。要创建字典，请使用 **spellin** 命令。

包含备用拼写列表、历史记录列表和停止列表的文件可由 **-d**、**-f** 和 **-h** 标志后的文件名参数指定。所有输出的副本可累加到历史记录文件中。

下列三个程序帮助维护和检查 **spell** 命令使用的散列列表。

**/usr/sbin/spell/hashmake**

从标准输入读取词列表，并将相应的 9 位散列代码写到标准输出。

**/usr/bin/spellin** *Number*

从标准输入读取散列代码的指定 *Number*，并将压缩的拼写列表写到标准输出。

**/usr/sbin/spell/hashcheck** *SpellingList*

读取压缩的 *SpellingList*，对其中的所有词重新创建 9 位散列代码，并将这些代码写入标准输出。

*File* 参数指定由 **spell** 命令读取并与拼写列表比较的文件。如果不指定文件，此命令从标准输入读取。

### 标志

- b** 检查英国英语拼写。然而，此标志没有为英国英语拼写提供合理的对比原型。通过使用某种词形变化、前缀和后缀来与拼写字典进行匹配的算法是基于美式英语拼写。
- d** *HashList* 指定 *HashList* 文件为备用拼写列表。缺省为 **/usr/share/dict/hlist[ab]**。
- h** *HistoryList* 指定 *HistoryList* 文件为备用历史列表，它用于积累所有输出。缺省为 **/usr/sbin/spell/spellhist**。  
注: *HistoryList* 文件必须是现有的具有读写许可权的文件。
- i** 取消处理包含文件。
- l** 此标志跟在全部包含文件链 (**.so** 和 **.nx** 格式化命令) 之后。不使用该标志时，**spell** 命令将跟在全部包含文件链 (除了那些以 **/usr/lib** 开头的文件) 之后。
- s** *HashStop* 指定 *HashStop* 文件为备用停止列表，它用于过滤出错误拼写但未查出的词。缺省为 **/usr/share/dict/hstop**。
- v** 显示所有不在拼写列表中的词，并指出这些词的可能派生词。
- x** 用 = (等号) 显示每个可能的词根。

## + WordList

对于其他的词，检查 *WordList*。*WordList* 是您提供的包含排序词汇列表（一行一个词）的文件名。使用这个标志，您可为每个作业指定一组正确拼写的词汇集（除了 **spell** 命令自己的拼写列表外）。

## 退出状态

返回以下退出值：

- 0 指示成功完成。
- >0 指示发生错误。

## 示例

1. 要检查拼写，请输入：

```
spell chap1 >mistakes
```

这将创建名为 *mistakes* 的文件，该文件包含在 *chap1* 中找到的而不在系统拼写字典中的所有词汇。其中一些词汇可能是 **spell** 命令没有识别出来的拼写正确的词。由于词汇列表可能比较长，所以请将 **spell** 命令的输出保存在文件中。

2. 要检查英国英语拼写，请输入：

```
spell -b chap1 >mistakes
```

将根据英国英语字典检查 *chap1*，并将有疑问的词汇写入 *mistakes* 文件。

3. 要了解 **spell** 命令如何派生词汇，请输入：

```
spell -v chap1 >deriv
```

这将列出不是在字典中逐字对照找出的而是根据字典词汇的形式派生出来的词。对于每个词，显示出用于形成派生词的前缀和后缀。还将列出字典中根本没有的词汇。

4. 要根据额外的词汇表进行拼写检查，请输入：

```
spell +newwords chap1
```

将根据系统字典和 *newwords* 检查 *chap1* 中的词汇拼写。*newwords* 文件按字母顺序列出词汇，一行一个词。您可使用文本编辑器（比如 *ed* 编辑器）来创建该文件并用 **sort** 命令对它按字母顺序排列。

## 文件

**/usr/share/dict/hlist[ab]**

包含美式英语和英国英语两种散列拼写列表。

**/usr/share/dict/hstop**

包含散列停止列表。

**/usr/sbin/spell/spellhist**

包含历史记录文件。

**/usr/sbin/spell/compress**

包含用于压缩历史记录文件的可执行 shell 程序。

**/usr/sbin/spell/hashmake**

从拼写列表中创建散列码。

**/usr/bin/spellin** *Number*

从散列码创建拼写列表。

**/usr/sbin/spell/hashcheck** *SpellingList*

从压缩的拼写列表创建散列码。

**/usr/sbin/spell/spellinprg**

**spellin** 文件调用的主程序。

**/usr/sbin/spell/spellprog**

检查拼写。

## 相关信息

**deroff** 命令、**eqn** 命令、**neqn** 命令、**sed** 命令、**sort** 命令、**spellin** 命令、**spellout** 命令、**tbl** 命令、**tee** 命令和 **troff** 命令。

---

## spellin 命令

### 用途

创建拼写列表。

### 语法

**spellin** [ *List* | *Number* ]

### 描述

**spellin** 命令创建 **spell** 命令使用的拼写列表。**spellin** 命令的参数可以是文件名或数字。**spellin** 命令将来自标准输入的词与现有拼写列表文件中的词结合起来，并将新的拼写列表放在标准输出上。如果未指定列表文件，就新建一个列表。如果指定 *Number*，**spellin** 命令从标准输入读取指定的散列码编号并写成压缩拼写列表。

### 示例

要将词 `hookey` 加入名为 `myhlist` 的拼写列表中，请输入：

```
echo hookey | spellin /usr/share/dict/hlista > myhlist
```

## 相关信息

**spell** 命令。

---

## spellout 命令

### 用途

验证一个词不在拼写列表中。

### 语法

**spellout** [ **-d** ] *List*

### 描述

**spellout** 命令从标准输入中查找每个词，并在标准输出上显示那些不在 *List* 参数指定的散列列表文件中的词汇。散列列表文件类似于 **spell** 命令使用的字典文件。

### 标志

**-d** 显示那些出现在散列列表文件中的词汇。

## 示例

要验证词 `hookey` 不在缺省拼写列表中，请输入：

```
echo hookey | spellout /usr/share/dict/hlista
```

在本示例中，`spellout` 命令将词 `hookey` 显示到标准输出（如果它不在散列列表文件中）。使用 `-d` 标志，`spellout` 显示词 `hookey`（如果它在散列文件中）。

## 相关信息

`spell` 命令和 `spellin` 命令。

---

## split 命令

### 用途

将文件分割成几段。

### 语法

要将一个文件分割成包含指定行数的多个文件

```
split [ -l LineCount ] [ -a SuffixLength ] [ File [ Prefix ] ]
```

要将一个文件分割成包含指定字节数的多个文件

```
split -b Number [ k | m ] [ -a SuffixLength ] [ File [ Prefix ] ]
```

### 描述

`split` 命令读取指定文件，以 1000 行大小写在一组输出文件上。第一个输出文件名由指定前缀（缺省值 `x`）和 `aa` 后缀组合构成，第二个文件名由前缀和 `ab` 后缀组合构成，如此按字典顺序一直到 `zz`（最多 676 个文件）。后缀的字母数及因此的输出名称文件数可用 `-a` 标志增加。

您指定的 `Prefix` 不能长于 `PATH_MAX - 2` 个字节（如果指定了 `-a` 标志，则不能长于 `PATH_MAX - SuffixLength` 个字节）。`PATH_MAX` 变量指定系统的最大路径名的长度（在 `/usr/include/sys/limits.h` 文件中定义）。

如果您不指定输入文件或如果您指定 `-`（减号）文件名，那么 `split` 命令从标准输入读取文件。

`split` 命令可以用于常规文本或二进制文件。将一个文件分割后，可以使用 `cat` 命令将其恢复到原始格式，并且文件的片段将以合适的顺序放置。

### 标志

注：`-b` 和 `-l` 标志是互斥的。

- |                                     |  |
|-------------------------------------|--|
| <code>-a <i>SuffixLength</i></code> | 指定用于形成输出名称文件后缀部分的字母数。字母数确定可能的输出文件名组合数。缺省是两个字母。   |
| <code>-b <i>Number</i></code>       | 将文件分割成 <code>Number</code> 变量指定的字节数。将 <code>k</code> （千字节）或 <code>m</code> （兆字节）乘数加到 <code>Number</code> 值的末尾使文件分别分割成 <code>Number*1024</code> 字节或 <code>Number*1,048,576</code> 字节的几个段。 |
| <code>-l <i>LineCount</i></code>    | 指定每个输出文件的行数。缺省值是 1000 行。   |

## 退出状态

该命令返回以下退出值:

- 0 命令成功运行。
- >0 发生错误。

## 示例

1. 要将文件分割成 1000 行的段, 请输入:

```
split book
```

此示例将 `book` 分割成 1000 行的段, 命名为 `xaa`、`xab`、`xac` 等。

2. 要将文件分割成 50 行的段并指定文件名前缀, 请输入:

```
split -l 50 book sect
```

此示例将 `book` 分割成 50 行的段, 命名为 `sectaa`、`sectab`、`sectac` 等。

3. 要将文件分割成 2 KB 的段, 请输入:

```
split -b 2k book
```

此示例将 `book` 分割成 2\*1024 字节的段, 命名为 `xaa`、`xab`、`xac` 等。

4. 要将文件分割成 676 个以上的段, 请输入:

```
split -l 5 -a 3 book sect
```

此例将 `book` 分割成 5 行的段, 命名为 `sectaaa`、`sectaab`、`sectaac` 等, 直到 `sectzzz` (最多 17,576 个文件)。

## 文件

`/usr/bin/split` 包含 `split` 命令。

## 相关信息

`cat` 和 `csplit` 命令。

《操作系统与设备管理》中的『文件』。

《操作系统与设备管理》中的『输入和输出重定向』。

---

## splat 命令

### 用途

简单性能锁分析工具 (`splat`)。  
提供内核和 `pthread` 锁使用情况报告。

### 语法

```
splat -i file [ -n file ] [ -o file ] [ -d [ bfta ] ] [ -l address ] [ -c class ] [ -s [ acelmsS ] ] [ -C cpus ] [ -S count ] [ -t start ] [ -T stop ] [ -p ]
```

**splat -h** [*topic*]

**splat -j**

## 描述

**splat**（简单性能锁分析工具）是一种软件工具，它对 AIX 跟踪文件进行后处理以产生内核简单和复杂锁使用报告。它也生成 **pthread** 互斥读写锁和条件变量使用报告。

## 标志

**-i** *inputfile* AIX 跟踪文件（REQUIRED）。  
**-n** *namefile* 包含 **gensyms** 命令输出的文件。  
**-o** *outputfile* 写入报告的文件（缺省：**stdout**）。  
**-d** *detail* 详细信息可为以下各项中的一项：  
    **[b]** 基本：摘要和锁详细信息（缺省）  
    **[f]** 函数：基本 + 函数详细信息  
    **[t]** 线程：基本 + 线程详细信息  
    **[a]** 全部：基本 + 函数 + 线程详细信息  
**-c** *class* 如果用户提供十进制锁类索引，**splat** 将只报告该类中锁的活动。  
**-l** *address* 如果用户提供十六进制锁地址，**splat** 将只报告该地址中锁的活动。**splat** 将过滤包含该锁地址的锁 hook 的跟踪文件，并单独为该锁生成报告。  
**-s** *criteria* 按下列条件对锁、函数和线程报告排序：  
    **a** 获取  
    **c** CPU 占用时间百分比  
    **e** 已占用时间百分比  
    **l** 锁地址、函数地址或线程标识  
    **m** 失误率  
    **s** 轮转计数  
    **S** CPU 轮转占用时间百分比（缺省）  
    **w** 实际等待时间百分比  
    **W** 平均等待队列深度  
**-C** *cpus* 指定跟踪的现有的 CPU 数。  
**-S** *count* 每个报告中的最大条目数（缺省：10）。  
**-t** *starttime* 从跟踪开始起的时间偏移量（秒）。  
**-T** *stoptime* 从跟踪开始到停止分析跟踪数据的时间偏移量（秒）。（缺省：跟踪结束）。  
**-h** [*topic*] 关于使用或特定主题的帮助。有效主题为：  
    • 全部  
    • 概述  
    • 输入  
    • 名称  
    • 报告  
    • 排序  
**-j** 显示 **splat** 使用的跟踪 hook 的列表。  
**-p** 指定 PURR 寄存器的使用来计算 CPU 时间。

## 帮助

以下是可用帮助主题的列表及其简要总结:

概述	本文。
输入	需要 AIX 跟踪挂钩以从 <b>splat</b> 中获取有用的输出。
名称	可使用什么名称的实用程序来使 <b>splat</b> 将地址映射到人类可读的符号。
报告	描述 <b>splat</b> 可产生的每个报告及用于计算报告值的公式。
排序	所有可用的排序选项及其如何应用于 <b>splat</b> 的输出。

## Splat 跟踪

**Splat** 将 AIX 跟踪命令收集的 AIX 跟踪文件作为主要输入。用 **splat** 分析跟踪前，您需要确保跟踪是用一组适当的 **hook** 来收集的，包括以下内容:

```
106 DISPATCH
10C DISPATCH IDLE PROCESS
10E RELOCK
112 LOCK
113 UNLOCK
134 HKWD_SYSC_EXECVE
139 HKWD_SYSC_FORK
419 CPU PREEMPT
465 HKWD_SYSC_CRTHREAD
46D WAIT_LOCK
46E WAKEUP_LOCK
606 HKWD_PTHREAD_COND
607 HKWD_PTHREAD_MUTEX
608 HKWD_PTHREAD_RWLOCK
609 HKWD_PTHREAD_GENERAL
```

由于在多处理器环境中使用锁的频率，捕获这些锁和解锁跟踪事件可能导致严重的性能下降。因此，通常禁用锁跟踪事件报告。为了启用锁跟踪事件报告，在收集包含 **splat** 需要的（KornShell 语法）锁跟踪事件的跟踪前，必须采用以下步骤:

1. `bosboot -ad /dev/hdisk0 -L`
2. `shutdown -Fr`
3. (reboot the machine)
4. `locktrace -S`
5. `mkdir temp.lib; cd temp.lib`
6. `ln -s /usr/ccs/lib/perf/libpthreads.a`
7. `export LIBPATH=$PWD:$LIBPATH`

步骤 1 到 3 是可选的。它们启用显示内核锁类名而非地址。请参考 **bosboot(1)** 以获取有关 **bosboot** 及其标志的更多信息。步骤 5 到 7 对于激活用户 **pthread** 锁检测是必要的；**temp.lib** 子目录可以放在任何位置。为了完成报告，步骤 1 到步骤 7 是必需的。

## Splat 名称

**Splat** 可以将 **gensyms** 的输出当作可选输入，并使用它将锁和函数地址映射为人类可读的符号。

锁类和偏移量可以用来广泛地标识一个锁，但不像实际的符号那样特定地标识锁。



## Splat 报告

由 **splat** 生成的报告包含报告摘要、锁摘要报告部分和锁详细信息报告的列表，每个报告都可能与相关的函数详细信息报告和 / 或线程详细信息报告。

### 摘要报告

~~~~~

报告摘要由下列元素构成:

- 用于收集跟踪的跟踪命令。
- 执行跟踪的主机。
- 执行跟踪的日期。
- 跟踪持续的时间 (秒)。
- 估计的 CPU 数。
- 总共的已用跟踪持续时间 (秒);  
(跟踪持续时间乘以跟踪中标识的 CPU 数)。
- 开始时间, 是从跟踪开始起的时间偏移量 (秒), 此时开始收集跟踪统计信息。
- 停止时间, 是从跟踪开始起的时间偏移量 (秒), 此时停止收集跟踪统计信息。
- 跟踪期间的获取总数。
- 每秒获取数, 由锁获取总数除以实时跟踪持续时间算出。
- 总轮转时间的百分比 (%), 这是所有锁轮转占用时间的总和除以跟踪持续时间总和, 再除以 100。  
当前的目标是使此值小于总跟踪持续时间的 10%。

### 锁摘要

~~~~~

锁摘要报告有以下字段:

锁	名称, 锁类或锁的地址。
类型	锁的类型, 由以下字母之一标识: Q RunQ 锁 S 简单内核锁 D 禁用的简单内核锁 C 复杂内核锁 M PThread 互斥锁 V PThread 条件变量 L PThread 读/写锁
获取	此锁的锁定尝试成功的次数减去 占用此锁时线程被先占的 次数。
轮转	此锁的锁定尝试失败的次数减去 轮转时线程未分派的 次数。
等待 或者 转换 %Miss	导致尝试线程进入休眠状态 以等待该锁可用或者导致分配 <code>krlock</code> 的 锁尝试失败的次数。 轮转数除以获取数加上轮转数, 再乘以 100。
%Total	获取数除以所有锁获取总数, 再乘以 100。
Locks/CSec	获取数除以总共已用的 持续时间 (秒)。
实际 CPU 占用百分比	线程分派时占用有问题的锁所占 已用总跟踪时间的百分比。 已分派保留时间 (秒) 除以跟踪持续总时间。

再乘以 100。

实用时间	分派或睡眠时线程保留锁所占 已用总时间的百分比。 未分派和分派保留时间（秒）除以 跟踪持续总时间，再乘以 100。
梳状装配轮转	等待获得该锁时线程轮转所占 所用跟踪总时间的百分比。 SPIN_HOLDTIME_IN_SECONDS 除以总跟踪持续时间， 再乘以 100。

锁摘要报告缺省为十个锁的列表，按轮转占用时间百分比（第十个字段）的降序排序。摘要报告的长度可用 **-S** 开关调整。摘要报告（和所有其他报告）的排序顺序可用 **-s** 开关设置，其选项在 SORTING 帮助部分（**splat -h** 排序）中描述。

#### 锁详细信息

^^^^^^^^

锁详细信息报告由下列字段构成：

锁	锁的地址（十六进制）。	
名称	该地址（如果可用）的符号映射	
类	锁类名（如果可用）和十六进制偏移量， 用于分配该所（lock_alloc() 内核服务）。	
父线程	父线程的线程标识。该字段仅为互斥锁存在。 读 / 写锁和条件变量报告。	
创建时间	在跟踪中记录的第一个事件后所用的时间（秒） （如果可用）。该字段仅为互斥锁、读 / 写锁 和条件变量报告而存在。	
删除时间	在跟踪中记录的第一个事件后所用的时间（秒） （如果可用）。该字段仅为互斥锁、读 / 写锁 和条件变量报告而存在。	
Pid	与锁关联的 Pid 数（该字段仅为互斥锁、 读 / 写锁和条件变量报告而存在）。	
进程名称	与锁关联的进程名称（该字段仅为互斥锁、 读 / 写锁和条件变量报告而存在）。	
调用链	调用方法的堆栈（如果可能有调用方法的话，该字段仅为 互斥锁、读 / 写锁和条件报告而存在）。	
获取	此锁的锁定尝试成功的次数。	对于条件变量锁报告，该字段命名为 Passes。
失败率	锁定尝试失败的次数除以 获取数加上锁定尝试失败的次数， 再乘以 100。	
轮转计数	锁定尝试失败的次数。	
等待计数	导致线程进入睡眠状态 以等待锁可用的锁定尝试失败的 次数。	
转换计数	简单锁分配和取消分配 krlock 的次数。	
繁忙计数	simple_lock_try() 调用返回为繁忙的次数。	
CPU 占用秒数	锁被分派的线程所占用的 总时间（秒）。	

已用时间	锁被分派的和未分派的线程所占用的总时间（秒）。
注:	这两个值都不能超过跟踪实际已用的跟踪持续时间。
实际 CPU 占用百分比	线程分派时占用有问题的锁所占已用总跟踪时间的百分比。 $\text{DISPATCHED\_HOLDTIME\_IN\_SECONDS}$ 除以跟踪持续时间, 再乘以 100。
实用时间	分派或睡眠时线程保留锁所占已用总时间的百分比。 $\text{UNDISPATCHED\_AND\_DISPATCHED\_HOLDTIME\_IN\_SECONDS}$ 除以跟踪持续时间, 再乘以 100。
梳状装配轮转	等待获得该锁时线程轮转所占所用跟踪总时间的百分比。 轮转保留时间（秒）除以跟踪持续时间, 再乘以 100。
等待	线程尝试获得该锁失败所占已用总跟踪时间的百分比。
SpinQ	Splat 保持最小、最大和平均的轮转队列深度（线程正在轮转、等待锁可用）。
WaitQ	和队列深度一样, splat 也跟踪所等待的线程队列的最大和平均深度（等待锁可用）。
PROD	相关 krlocks prod 调用计数。
授予自身	简单锁和相关 krlock 的授予自身调用计数。
授予目标	简单锁和相关 krlock 的授予目标调用计数。w/ preemption 报告成功的调用计数, 并产生占先。
授予全部	简单锁和相关 krlock 的授予全部调用计数。w/ preemption 报告成功的调用计数, 并产生占先。
HANDOFF	相关 krlocks handoff 调用计数。

锁定活动 w / 中断启用（毫秒）

锁详细报告的这部分是 splat 收集的每个锁的原始数据的转储, 时间以毫秒表示。五种状态: 锁、轮转、等待、未分派 (atched) 和占先是 splat 的已启用的 **simple\_lock** 有限状态机的五种基本状态。每一种状态的计数是导致转换为该状态的线程的操作次数。以毫秒计的持续时间显示锁请求在此状态耗费的最小时间、最大时间及时间的总量。

- 锁: 此状态表示线程成功地获取锁。
- 轮转: 此状态表示线程尝试获取锁失败。
- 等待: 此状态表示轮转线程（处于轮转状态）在超出线程轮转的阈值后将要进入睡眠状态（自动地）。
- 未分派: 此状态表示轮转线程（处于轮转状态）在超出线程的轮转阈值前变为未分派（不自觉地）。
- 占先: 此状态表示何时不分派用占用锁的线程。

锁定活动 w / 中断禁用（毫秒）

锁详细报告的这部分是 splat 收集的每个锁的原始数据的转储, 时间以毫秒表示。六种状态: 锁、轮转、带 KRLOCK 的锁、KRLOCK 锁、KRLOCK 轮转和转换是 splat 的已禁用的 **simple\_lock** 有限状态机的六种基本状态。每一种状态的计数是导致转换为该状态的线程的操作次数。以毫秒计的持续时间显示锁请求在此状态耗费的最小时间、最大时间及时间的总量。

LOCK: 此状态表示线程成功地获取锁。

轮转: 此状态表示线程尝试获取锁失败。  
带有 KRLOCK 的锁: 线程已在保持相关 `krlock` 的同时成功获取锁, 并且当前正在执行。  
KRLOCK 锁: 线程已成功获取相关 `krlock`, 并且当前正在执行。  
KRLOCK 轮转: 线程正在执行并且尝试获取相关 `krlock` 失败。  
转换: 线程已成功分配与其相关的 `krlock`, 并且正在执行。

#### 函数详细信息

~~~~~

函数详细信息报告由以下字段组成:

|                  |                                                                                                                    |
|------------------|--------------------------------------------------------------------------------------------------------------------|
| 函数名              | 使用了该锁的<br>函数名和返回地址。                                                                                                |
| 获取               | 此锁的锁定尝试成功的次数。<br>对于复杂锁和读 / 写锁<br>在获取写 (Acquisition Write)<br>和获取读 (Acquisition Read)<br>之间有区别。                     |
| 失败率              | 锁定尝试失败的次数除以<br>获取数, 再乘以 100。                                                                                       |
| 轮转计数             | 锁定尝试失败的次数。<br>对于复杂锁和读 / 写锁<br>在轮转计数的写 (轮转计数写)<br>和读 (轮转计数读)<br>之间有区别。                                              |
| 等待计数             | 导致线程进入睡眠状态<br>以等待锁可用的锁定尝试失败的<br>次数。<br><br>对于复杂锁和读 / 写锁<br>在写 (等待计数写)<br>和读 (等待计数读)<br>的等待计数之间有区别。                 |
| 转换计数<br>繁忙计数     | 线程正在尝试获取简单锁时, 简单锁已分配 <code>krlock</code> 的次数。<br><code>simple_lock_try()</code> 调用返回为繁忙的次数。                        |
| CPU 占总时间的<br>百分比 | 线程分派时保留有问题的锁所占<br>已用总跟踪时间的百分比。<br><code>DISPATCHED_HOLDTIME_IN_SECONDS</code> 除以跟踪持续时间,<br>再乘以 100。                |
| 已用时间             | 分派或睡眠时线程保留锁所占<br>已用总时间的百分比。<br><code>UNDISPATCHED_AND_DISPATCHED_HOLDTIME_IN_SECONDS</code> 除以<br>跟踪持续时间, 再乘以 100。 |
| 轮转               | 线程在等待获取该锁时轮转<br>所用跟踪总时间的百分比。<br><code>SPIN_HOLDTIME_IN_SECONDS</code> 除以总跟踪持续时间,<br>再乘以 100。                       |
| 等待               | 线程尝试获得该锁失败所占<br>已用总跟踪时间的百分比。                                                                                       |
| 返回地址             | 调用函数返回地址 (十六进制)。                                                                                                   |
| 起始地址             | 调用函数的起始地址 (十六进制)。                                                                                                  |
| 偏移量              | 函数起始地址 (十六进制) 的偏移量。                                                                                                |

## 线程详细信息

线程详细信息报告由以下字段构成:

|              |                                                                                                            |
|--------------|------------------------------------------------------------------------------------------------------------|
| 线程标识         | 线程标识。                                                                                                      |
| 获取           | 此锁的锁定尝试成功的次数。                                                                                              |
| 失败率          | 锁定尝试失败的次数除以获取数, 再乘以 100。                                                                                   |
| 轮转计数         | 锁定尝试失败的次数。                                                                                                 |
| 等待计数         | 导致线程进入睡眠状态以等待锁可用的锁定尝试失败的次数。                                                                                |
| 转换计数         | 线程正在尝试获取简单锁时, 简单锁已分配 <code>krlock</code> 的次数。                                                              |
| 繁忙计数         | <code>simple_lock_try()</code> 调用返回为繁忙的次数。                                                                 |
| CPU 占总时间的百分比 | 线程分派时保留有问题的锁所占已用总跟踪时间的百分比。<br><code>DISPATCHED_HOLDTIME_IN_SECONDS</code> 除以跟踪持续时间, 再乘以 100。               |
| 已用时间         | 分派或睡眠时线程保留锁所占已用总时间的百分比。<br><code>UNDISPATCHED_AND_DISPATCHED_HOLDTIME_IN_SECONDS</code> 除以跟踪持续时间, 再乘以 100。 |
| 轮转           | 线程在等待获取该锁时轮转所用跟踪总时间的百分比。<br><code>SPIN_HOLDTIME_IN_SECONDS</code> 除以总跟踪持续时间, 再乘以 100。                      |
| 等待           | 线程尝试获得该锁所占用已用总跟踪时间的百分比。                                                                                    |
| 进程标识         | 进程标识 (仅对简单和复杂锁报告)。                                                                                         |
| 进程名          | 进程名 (仅对简单和复杂锁报告)。                                                                                          |

## Splat 排序

**splat** 允许用户指定使用哪个条件、使用 **-s** 选项对摘要和锁详细信息报告排序。缺省排序条件是按轮转占用时间百分比排序, 这是线程用于锁轮转的时间与总的跟踪持续时间的比率。使用 **-s**, 排序条件可变为以下值:

- a** 获取; 线程成功获取锁的次数。
- c** CPU 占用时间百分比; CPU 占用时间与总的跟踪持续时间的比率。
- e** 已用占用时间百分比; 已用占用时间与总的跟踪持续时间的比率。
- l** 位置; 锁或函数的地址, 或线程的标识。
- m** 错失率; 错失的锁定尝试次数与获取数的比率。
- s** 转数计数; 导致线程轮转等待该锁的锁定尝试失败的次数。
- S** CPU 轮转占用时间百分比 (缺省)。
- w** 已用等待时间百分比; 非零数量的线程等待锁的总时间百分比。
- W** 平均等待队列深度; 等待锁的线程平均数, 相当于每个等待线程在此状态耗用的平均时间。

**splat** 将用指定的条件对锁报告按降序排序。

## 限制

不分析其他类型的锁，如 VMM、XMAP 和一些特定于 Java 的锁。

## 文件

`/etc/bin/splat` 简单性能锁分析工具（**splat**）。提供内核和 pthread 锁使用报告。

## 相关信息

**simple\_lock(3)**、**simple\_unlock(3)**、**disable\_lock(3)**、**unlock\_enable(3)**、**trace(1)**、**trcrpt(1)**、**trcfmt(5)**、**gensyms(1)** 和 **bosboot(1)** 守护程序。

---

## splitvg 命令

### 用途

分割全镜像卷组的单一镜像副本。

### 语法

```
splitvg [ -y SnapVGname ] [ -c Copy ] [ -f ] [ -i ] VGname
```

### 描述

**splitvg** 命令将全镜像卷组的单一镜像副本分割成快照卷组。原始卷组 **VGname** 将停止使用现在是快照卷组 **SnapVGname** 组成部分的磁盘。两个卷组都将跟踪卷组中的写操作，以便快照卷组与原始卷重新组合时在合并的镜像副本中数据保持一致性。

#### 注:

1. 为了分割卷组，卷组中的所有逻辑卷必须具有目标镜像副本并且此镜像必须在磁盘或磁盘组上。只有目标镜像副本必须在目标磁盘或磁盘组上。
2. 如果分割的任何磁盘在原始卷组中都不活动，则 **splitvg** 命令将失败。
3. 万一发生系统崩溃或限额损失，则必须运行 **joinvg** 命令来将磁盘重新加入至原始卷组。
4. 创建快照卷组没有并发或增强的并发方式支持。
5. 快照卷组中将创建新逻辑卷和文件系统安装点。
6. 对于 **rootvg**，**splitvg** 命令不受支持。
7. 对于具有活动调页空间的卷组，**splitvg** 命令不受支持。
8. 当 **splitvg** 命令目标是一个可并发的卷组（以非并发方式联机）时，创建的新卷组将不会在 **splitvg** 命令完成时联机。新的卷组必须通过手工联机。

### 标志

- y SnapVGname** 允许指定卷组名称而不是自动生成卷组名称。卷组名称必须系统内是唯一的，且在 1 至 15 个字符的范围内。名称不能以已在其他设备的“设备配置数据库”的 **PdDv** 类中定义的前缀开头。新卷组名发送到标准输出。
- c Copy** 分割哪个镜像。有效值为 1、2 或 3。缺省是第二个副本。
- f** 即使指定创建快照卷组的镜像副本具有旧文件分区，也将强制分割。

**-i** 将卷组的镜像副本分割成不能重新加入原始卷组的新卷组。

## 安全性

访问控制: 您必须具有 `root` 权限才能运行此命令。

## 示例

1. 要分割卷组, 输入:

```
splitvg testvg
```

卷组 `testvg` 的第二个镜像副本分割成带有自动生成名称 (将要显示) 的新卷组。

2. 以 `snapvg` 名称分割卷组的第一个镜像副本, 输入:

```
splitvg -y snapvg -c 1 testvg
```

## 文件

`/usr/sbin` `splitvg` 命令驻留的目录。

## 相关信息

`joinvg` 和 `recreatevg` 命令。

---

## splitlvcopy 命令

### 用途

分割一个逻辑卷的副本并从它们创建新的逻辑卷。

### 语法

```
splitlvcopy [ -f ] [ -y NewLogicalVolumeName ] [ -Y Prefix ] LogicalVolume Copies [ PhysicalVolume ... ]
```

### 描述

注:

1. 要使用此命令, 您或者必须具有 `root` 用户权限或是系统组的成员。
2. `splitlvcopy` 命令不允许在快照卷组上或具有快照卷组的卷组上使用。

**警告:** 尽管 `splitlvcopy` 命令可分割打开的逻辑卷, 包括包含已安装文件系统的逻辑卷, 但建议不要这么做。如果多个进程同时访问逻辑卷, 您可能失去 `LogicalVolume` 和 `NewLogicalVolume` 之间的一致性。分割打开的逻辑卷时, 您默认接受了与此操作有关的潜在数据丢失和数据损坏风险。为避免潜在的损坏窗口, 分割前请关闭逻辑卷及卸载文件系统。

`splitlvcopy` 命令从 `LogicalVolume` 的每个逻辑分区除去副本, 并用它们创建 `NewLogicalVolume`。 `Copies` 参数确定分割后保留在 `LogicalVolume` 中的最大物理分区数。因此, 如果 `LogicalVolume` 分割前有 3 个副本, 并且 `Copies` 参数是 2, 那么分割后 `LogicalVolume` 将有 2 个副本, 而 `NewLogicalVolume` 将有 1 个副本。您不能分割逻辑卷, 使分割后 `LogicalVolume` 和 `NewLogicalVolume` 中的总副本数大于分割前 `LogicalVolume` 中的副本数。

*NewLogicalVolume* 将具有与 *LogicalVolume* 相同的所有逻辑卷特征。如果 *LogicalVolume* 没有逻辑卷控制块，命令将忽略警告消息成功执行并创建没有逻辑卷控制块的 *NewLogicalVolume*。

分割包含文件系统的逻辑卷时，还需要注意其他注意事项。分割后有两个逻辑卷，但 **/etc/filesystems** 文件中将只有一个指向 *LogicalVolume* 的条目。为了作为文件系统访问 *NewLogicalVolume*，您必须以不同的安装点在 **/etc/filesystems** 中创建指向 *NewLogicalVolume* 的另外一个条目。如果此安装点不存在，您必须在可以安装新文件系统前创建它。另外，如果在 *LogicalVolume* 处于打开状态时创建 *NewLogicalVolume*，您一定要运行以下命令：

```
fsck /dev/NewLogicalVolume
```

在可以安装新文件系统之前。

您不能使用系统管理界面工具（SMIT）运行此命令。此命令不支持消息编目，因此只能提供没有消息目录号的英文错误消息。`splitlvcopy` 的文档由此联机帮助页组成。

## 标志

- f** 指定不请求确认分割打开的逻辑卷。缺省情况下，**splitlvcopy** 在分割打开的逻辑卷前请求确认。这包括打开原始的逻辑卷及包含已安装的文件系统的逻辑卷。
- y *NewLogicalVolumeName*** 指定从 *LogicalVolume* 移入副本的新逻辑卷的名称。
- Y *Prefix*** 指定 *Prefix* 用于取代新逻辑卷的系统生成名的前缀。前缀必须少于或等于 13 个字符。名称不能以已在其他设备的“设备配置数据库”的 **PdDv** 类中定义的前缀开头，也不能使用已经由另一设备使用的名称。

## 参数

- Copies* 指定分割后保留在逻辑卷中的最大物理分区数。
- LogicalVolume* 指定要分割的逻辑卷名称或逻辑卷标识。
- PhysicalVolume* 指定要从中除去副本的物理卷名称或物理卷标识。

## 退出状态

该命令返回以下退出值：

- 0** 成功完成。
- >0** 发生错误。

## 安全性

访问控制：您必须具有 `root` 用户权限或是系统组的成员才能运行此命令。

审计事件：不适用

## 示例

要分割属于名为 **oldlv** 逻辑卷（当前每个逻辑分区有 3 个副本）的每个逻辑分区的一个副本，并创建逻辑卷 **newlv**，请输入：

```
splitlvcopy -y newlv oldlv 2
```

逻辑卷 **oldlv** 中的每个逻辑分区现在有两个物理分区。逻辑卷 **newlv** 中的每个逻辑分区现在有一个物理分区。



## 文件

`/etc/splittlvcopy`  
`/tmp`

包含 `splittlvcopy` 命令。  
包含运行 `splittlvcopy` 命令时创建的临时文件。

## 相关信息

命令: `rmlvcopy` 和 `mklv`。

---

## splp 命令

### 用途

更改或显示打印机驱动程序设置。

### 语法

```
splp [ -b Option ] [ -B Number ] [ -c Option ] [ -C Option ] [ -e Option ] [ -f Option ] [ -F! ]  
[ -i Number ] [ -l Number ] [ -n Option ] [ -N Option ] [ -p Option ] [ -P Option ] [ -r Option ]  
[ -s Number ] [ -S Option ] [ -t Option ] [ -T Number ] [ -w Number ] [ -W Option ] [ DevicePath ]
```

### 描述

`splp` 命令更改或显示打印机设备驱动程序设置。缺省设备路径为 `/dev/lp0`；所有标志都可选。如果设备路径不能以 `/`（反斜杠）字符开头，就假定为 `/dev` 目录。同样，如果未指定标志，`splp` 命令报告指定设备路径的当前设置。要更改当前设置，指定相应的标志。不需要进行其他处理，也没有其他输出。

`splp` 命令所做的更改在下次重新启动系统或重新运行 `splp` 命令前都保持有效。在您每次启动系统时，`splp` 命令可从 `/etc/inittab` 命令文件运行以配置打印机。

注: `-b`、`-c`、`-C`、`-f`、`-i`、`-l`、`-n`、`-p`、`-r`、`-t`、`-w` 和 `-W` 标志的 `splp` 命令设置仅在数据直接发送到打印机设备的时候（例如，将 `cat` 命令的输出直接重定向到指定设备路径）应用。当文件使用 `enq`、`qprt`、`lp` 或 `lpr` 命令排队等待打印时，忽略且不更改这些标志的设置。

### 标志

- `-b Option` 指定是否发送退格至打印机:
  - + 指定发送退格至打印机。
  - ! 指定废弃退格。
- `-B Number` 以指定的每秒的位数设置速度。 *Number* 变量的值为 50、75、110、134、150、300、600、1200、1800、2400、4800、9600、19,200 和 38,400。
- `-c Option` 指定是否发送回车符至打印机:
  - + 发送回车符至打印机。
  - ! 将回车符转换为换行符。
- `-C Option` 指定是否将所有的小写字符都转换成大写字符:
  - + 将小写字符转换为大写字符。
  - ! 不将小写字符转换为大写字符。

- e Option** 指定检测到错误时执行的处理:
- + 返回错误。
  - ! 等待直到错误清除。
- f Option** 指定是否向打印机发送换页或用换行或回车符模拟换页:
- + 发送换页至打印机。
  - ! 用换行或回车符模拟换页。
- F!** 对 3812 或 3816 页式打印机重新设置字体状态指示符。此标志通过下一个假脱机打印作业, 使字体从打印机字体软盘重新装入打印机内存。如果关闭然后再打开打印机或者打印机内存中的字体已损坏, 应当指定此标志。
- i Number** 缩进指定的列数, 其中 *Number* 变量值为整数。
- l Number** 每页打印指定行数, 其中 *Number* 变量值为整数。
- n Option** 指定是否将换行发送至打印机或将换行转换为回车符。
- + 向打印机发送换行。
  - ! 将换行转换成回车符。
- N Option** 指定是否启用奇偶性校验的生成和检测:
- + 启用奇偶性校验的生成和检测。
  - ! 禁用奇偶性校验的生成和检测。
- p Option** 指定系统是向未修改打印机发送所有的字符还是根据 **-b**、**-c**、**-C**、**-f**、**-i**、**-l**、**-n**、**-r**、**-t**、**-w** 和 **-W** 标志的设置转换字符:
- + 将所有的字符发送至未更改的打印机, 覆盖其他设置。
  - ! 根据设置转换字符。
- P Option** 指定奇偶校验:
- + 指定奇校验。
  - ! 指定偶校验。
- r Option** 指定换行后是否添加回车符:
- + 换行后发送回车符。
  - ! 换行后不发送回车符。
- s Number** 选择字符大小, 其中 *Number* 变量是位数。 *Number* 变量值可以是 5、6、7 或 8。关于字符大小的其他信息, 请参阅 **termio.h** 特殊文件。
- S Option** 指定每个字符的停止位数:
- + 每个字符 2 个停止位。
  - ! 每个字符 1 个停止位。
- t Option** 指定是否扩展制表符:
- + 不扩展制表符。
  - ! 在 8 个位置边界扩展制表符。
- T Number** 将超时周期设置为 *Number* 变量指定的秒数。 *Number* 变量值必须是整数。
- w Number** 显示 *Number* 变量指定的列数。 *Number* 变量值必须是整数。
- W Option** 指定是否将超出指定宽度的字符回绕至下一行, 并在换行符后显示 ... (3 个点):
- + 将超出指定宽度的字符回绕到下一行, 并在换行符后显示 ... (3 个点)。
  - ! 截断超出指定宽度的字符。

## 示例

1. 要显示 `/dev/lp0` 打印机的当前打印机设置，输入：

```
sp lp
```

2. 要更改打印机设置，输入：

```
sp lp -w 80 -W + -C +
```

这就改变了 80 列纸（`-w 80` 标志）的 `/dev/lp0` 打印机的设置。它也将大于 80 列宽度的每一行换行回绕到第二行（`-W+` 标志），并以大写方式打印所有的字母字符（`-C+` 标志）。

## 文件

`/dev/lp*` 包含打印机属性文件。  
`/etc/inittab` 包含打印机配置命令文件。

## 相关信息

`cat` 命令、`enq` 命令、`lp` 命令、`lpr` 命令和 `qprt` 命令。

`termio.h` 文件。

《打印机和打印指南》中的『打印机管理』。

《打印机和打印指南》中的『使用打印机冒号文件添加打印机』。

《打印机和打印指南》中的『虚拟打印机定义和属性』。

---

## spost 命令

### 用途

路由消息。

### 语法

```
spost [ -noalias | -alias File ... ] [ -format | -noformat ] [ -filter File | -nofilter ] [ -width Number ] [ -watch | -nowatch ] [ -remove | -noremove ] [ -backup | -nobackup ] [ -verbose | -noverbose ] File
```

### 描述

`spost` 命令将消息路由到正确目的地。`spost` 命令不是由用户启动。`spost` 命令只能被其他程序调用。

`spost` 命令搜索指定收件人地址的消息的所有组成部分，并分析每个地址以进行格式正确性检查。然后 `spost` 命令以标准格式放置地址并启动 `sendmail` 命令。`spost` 命令执行类似于 `post` 命令的功能，但它所做的地址格式化比 `post` 命令少。

`spost` 命令是缺省值（相对于 `post` 命令）。通过设置 `.mh_profile` 中的 `postproc` 变量改变缺省值。例如：  
postproc: /usr/lib/mh/post

*File* 参数是被发送的文件名。

## 标志

|                             |                                                                                        |
|-----------------------------|----------------------------------------------------------------------------------------|
| <b>-alias</b> <i>File</i>   | 搜索指定邮件别名文件以获取地址。您可重复此标志以指定多个邮件别名文件。 <b>spost</b> 命令自动搜索 <b>/etc/mh/MailAliases</b> 文件。 |
| <b>-backup</b>              | 在 <b>spost</b> 命令成功发送消息后，在文件名前放一个, (逗号) 来对消息文件重命名。                                     |
| <b>-filter</b> <i>File</i>  | 用指定文件中的消息头组件来复制发送到 Bcc: 字段收件人的消息。                                                      |
| <b>-format</b>              | 以标准格式将所有的收件人地址提交给交付传送系统。此标志是缺省值。                                                       |
| <b>-help</b>                | 列出命令语法、可用的参数选项 (开关) 和版本信息。<br><b>注:</b> 对于消息处理程序 (MH), 必须完整拼写此标志的名称。                   |
| <b>-noalias</b>             | 不要使用任何别名文件传送消息。                                                                        |
| <b>-nobackup</b>            | 发送文件后不对消息重命名。此标志是缺省值。                                                                  |
| <b>-nofilter</b>            | 从消息除去 Bcc: 字段头, 并将它发送到 Bcc: 组件中指定的收件人。此标志是缺省值。                                         |
| <b>-noformat</b>            | 不要改变收件人地址格式。                                                                           |
| <b>-noremove</b>            | 发送消息后不要除去临时消息文件。                                                                       |
| <b>-noverbose</b>           | 在消息传送到 <b>sendmail</b> 命令期间不显示信息。此标志是缺省值。                                              |
| <b>-nowatch</b>             | 在 <b>sendmail</b> 命令执行传递期间不要显示信息。此标志是缺省值。                                              |
| <b>-remove</b>              | 消息成功发送后, 除去临时消息文件。此标志是缺省值。                                                             |
| <b>-verbose</b>             | 在传递消息到 <b>sendmail</b> 命令的过程中显示信息。此信息允许您监视涉及的步骤。                                       |
| <b>-watch</b>               | <b>sendmail</b> 命令传送消息期间显示信息。此信息允许您监视涉及的步骤。                                            |
| <b>-width</b> <i>Number</i> | 设置包含地址的组件的宽度。缺省值是 72 列。                                                                |

## 文件

|                                |                       |
|--------------------------------|-----------------------|
| <b>\$HOME/mh_profile</b>       | 包含消息处理程序 (MH) 用户概要文件。 |
| <b>/tmp/pst</b> <i>Number</i>  | 包含临时消息文件。             |
| <b>/etc/mh/MailAliases</b>     | 包含缺省邮件别名。             |
| <b>/usr/lib/mh/.mh_profile</b> | 包含消息处理程序 (MH) 用户概要文件。 |

## 相关信息

**ali** 命令、**conflict** 命令、**mhmail** 命令、**post** 命令、**send** 命令、**sendmail** 命令和 **whom** 命令。

**.mh\_alias** 文件格式。

《网络与通信管理》中的『邮件应用程序』。

---

## spray 命令

### 用途

向主机发送指定数量的信息包并报告性能统计信息。

### 语法

```
/usr/sbin/spray Host [ -c Count ] [ -d Delay ] [ -i ] [ -l Length ]
```

### 描述

**spray** 命令使用远程过程调用 (RPC) 协议向指定主机发送单向流信息包。此命令报告接收到多少信息包以及传输率是多少。*Host* 参数可以是名称或因特网地址。主机仅在 **sprayed** 守护程序运行时响应。

**注:** **spray** 命令不支持 IPv6。

要了解影响 **spray** 命令性能的因素，请参阅 **rpc.sprayd** 守护程序文档。

## 标志

- c Count** 指定发送的信息包数。缺省值是组成总数据流大小为 100,000 字节所需的信息包数目。
- d Delay** 指定在发送各数据包间系统暂停时间(微秒)。缺省值是 0。
- i** 使用因特网控制报文协议 (ICMP) 而不是 RPC 协议来回送信息包。因为 ICMP 自动回送，所以它创建了双向数据流。您必须是 root 用户才能使用此选项。
- l Length** 指定保留 RPC 调用消息的数据包的字节数。Length 参数的缺省值是 86 字节，即 RPC 和 UDP 头部分的大小。

信息包中的数据用外部数据表示法 (XDR) 编码。因为 XDR 只处理 32 位数，所以 **spray** 命令将较小的值舍入到最近的可能值。

当 Length 参数在以太网中大于 1500 或在令牌环中大于 1568，RPC 调用就不再适合以太网信息包。因此，Length 字段与以太网信息包大小不再有简单的对应。

## 示例

1. 向工作站发送 **spray** 命令时，指定发送的信息包数及系统在发送每个数据包间将等待的时间长度，如下所示：

```
/usr/sbin/spray zorro -c 1200 -d 2
```

在本例中，**spray** 命令以 2 微秒的间隔向名为 zorro 的工作站发送 1200 个数据包。

2. 要更改发送的数据包中的字节数，输入：

```
/usr/sbin/spray zorro -l 1350
```

在本例中，**spray** 命令向名为 zorro 的工作站发送 1350 字节的信息包。

3. 要用 ICMP 协议而不是 RPC 协议发送回送信息包，输入：

```
/usr/sbin/spray zorro -i
```

在本例中，**spray** 命令向名为 zorro 的工作站发送回送信息包。

## 相关信息

**sprayd** 守护程序。

NFS 命令列表。

《网络与通信管理》中的『网络文件系统 (NFS)』。

《网络与通信管理》中的『NFS 故障诊断』。

---

## sprayd 守护程序

### 用途

接收 **spray** 命令发送的信息包。

## 语法

`/usr/lib/netsvc/spray/rpc.sprayd`

## 描述

**rpc.sprayd** 守护程序是一个记录 **spray** 命令发送的信息包的服务器。通常，**rpc.sprayd** 守护程序由 **inetd** 守护程序启动。

## UDP 性能

用户数据报协议（UDP）用 **spray** 命令执行，**rpc.sprayd** 守护程序可受下列因素影响：

- 如何为系统配置调整内存缓冲区（mbufs）。
- **spray** 命令的 UDP 信息包的接受脉冲串传输速率（即帧间间隔）。
- 其他系统活动。因为 **rpc.sprayd** 守护程序作为普通用户进程运行，其他活动(如 **init** 进程或 **syncd** 守护程序)就可以影响 **rpc.sprayd** 守护程序的操作。
- **rpc.sprayd** 守护程序进程的优先级。**rpc.sprayd** 守护程序有动态计算的浮动进程优先级。
- **rpc.sprayd** 守护程序使用的接收套接字缓冲区大小。由于各种操作使用不同的套接字缓冲区大小，所以难以用 **spray** 命令和 **rpc.sprayd** 守护程序测定 UDP 性能且测定结果是不确定的。

## 文件

`/etc/inetd.conf`                    启动 RPC 守护程序和其他 TCP/IP 守护程序的 TCP/IP 配置文件。

## 相关信息

**spray** 命令。

**inetd** 守护程序。

NFS 命令列表。

《网络与通信管理》中的『网络文件系统（NFS）』。

---

## srcmstr 守护程序

### 用途

启动系统资源控制器。

### 语法

**srcmstr** `/usr/sbin/srcmstr` [ **-r** ] [ **-B** ]

### 描述

**srcmstr** 守护程序是系统资源控制器（SRC）。**srcmstr** 守护程序生成并控制子系统、处理子系统短状态请求、向子系统传递请求并处理出错通知。

通常 **srcmstr** 守护程序通过使用 **inittab** 文件条目启动。

## 标志

- r** 如果用 **-r** 标志启动守护程序，则接收远程请求。如果不用 **-r** 标志启动 **srcmstr**，则忽略远程请求。
- B** 指定 **-B** 标志，它引起 **srcmstr** 守护程序以前发行版（AIX 4.3.1 和更早的版本）的方式运行。

### 注:

- **srcmstr** 守护程序通常是从 **inittab** 启动。要添加 **-r** 或 **-B** 标志，编辑 **/etc/inittab** 并运行 **init q** 或重新引导。
- 用户必须在远程系统上作为 **root** 用户运行。必须配置本地 **/etc/hosts.equiv** 文件或 **.rhosts** 文件以允许远程请求。

## 安全性

审计事件：如果审查子系统已正确配置并启用，则每次执行 **srcmstr** 命令时都将生成以下审查记录（事件）。

| 事件               | 信息                  |
|------------------|---------------------|
| <b>SRC_Start</b> | 在审计日志中列出正在启动的子系统名称。 |
| <b>SRC_Stop</b>  | 在审计日志中列出正在停止的子系统名称。 |

有关如何适当选择与分组审计事件及如何配置审计事件数据收集的更详细的信息，请参阅《安全性》中的 **Setting Up Auditing**。

## 错误恢复

缺省的 **/etc/inittab** 指定 **srcmstr** 守护程序的 **respawn** 标志。如果 **srcmstr** 守护程序异常终止并且 **/etc/inittab** 指定 **respawn** 标志，则重新启动 **srcmstr** 守护程序。然后它确定在前面的启用过程中，哪个 SRC 子系统是活动的。守护程序与这些子系统（如果它以前存在）重新建立通信并初始化专用的内核扩展和 **srcd** 守护程序以监视子系统进程。

如果对于前面调用的 **srcmstr** 已知的子系统终止，SRC 内核扩展通知 **srcd** 守护程序。**srcd** 守护程序向 **srcmstr** 发送套接字消息，并处理子系统终止（如同 **srcmstr** 启动子系统那样）。可指定 **-B** 标志禁用此功能，此时启动 **srcmstr** 守护程序。SRC 内核扩展在 **/usr/lib/drivers/SRC\_kex.ext** 中。**srcd** 的可执行文件是 **/usr/sbin/srcd**。

## 文件

|                                 |                                                    |
|---------------------------------|----------------------------------------------------|
| <b>/etc/inittab</b>             | 指定 <b>init</b> 命令读取的节。                             |
| <b>/etc/objrepos/SRCsubsys</b>  | 指定 SRC 子系统配置对象类。                                   |
| <b>/etc/objrepos/SRCnotify</b>  | 指定 SRC 通知方法对象类。                                    |
| <b>/etc/hosts.equiv</b>         | 如果指定的主机名不在 <b>/etc/hosts.equiv</b> 文件中，指定没有远程请求工作。 |
| <b>/etc/services</b>            | 定义用于因特网服务的套接字和协议。                                  |
| <b>/dev/SRC</b>                 | 指定 <b>AF_UNIX</b> 套接字文件。                           |
| <b>/dev/.SRC-unix</b>           | 指定临时套接字文件位置。                                       |
| <b>/dev/.SRC-unix/SRCD</b>      | 指定 <b>srcd</b> 守护程序的 <b>AF_UNIX</b> 套接字文件。         |
| <b>/var/adm/SRC/active_list</b> | 包含活动的子系统列表。                                        |

警告：此文件的结构对 SRC 是未公开的，可能被更改。

`/var/adm/SRC/watch_list`

包含在前面启用 **srcmstr** 守护程序期间活动的子系统进程列表。

`/.rhosts`

**警告：** 此文件的结构对 SRC 是未公开的，可能被更改。  
指定允许从此机器请求 SRC 功能的远程机器和用户（只有 root 用户）。

## 相关信息

**auditpr** 命令、**init** 命令。

《操作系统与设备管理》中的『系统资源控制器』提供了对子系统、子服务器和系统资源控制器的说明。

《安全性》中的 Auditing overview。

---

## startcondresp 命令

### 用途

开始对具有一个或多个已链接响应的条件进行监视。

### 语法

要开始监视一个条件：

```
startcondresp [-h] [-TV] condition[:node_name] [response [response...]]
```

要对条件 / 响应关联进行解锁或锁定：

```
startcondresp {-U | -L} [-h] [-TV] condition[:node_name] response
```

### 描述

**startcondresp** 命令启动对具有链接响应的条件的监视。条件和响应之间的链接被称为条件 / 响应关联。在集群环境下，条件和响应必须在相同节点中定义。监视启动后条件出现时，就运行响应。如果未指定响应，启动所有与此条件链接的响应的监视。当条件出现时，这引起所有的链接响应运行。如果指定多个响应，则只为那些链接响应启动监视。

如果指定一个或多个响应且响应未与条件链接，**startcondresp** 命令将指定响应与条件链接并启动监视。使用 **mkcondresp** 命令将响应与条件链接而不启动监视。

如果为了使系统软件能正常工作而需要特殊条件 / 响应关联，则它可能被锁定。锁定的条件 / 响应关联无法由 **startcondresp** 命令来启动。如果您对 **startcondresp** 命令指定的条件 / 响应关联被锁定，它不会启动；而会生成一个错误，通知您该条件 / 响应关联已锁定。要解锁条件 / 响应关联，您可以使用 **-U** 标志。然而，由于条件 / 响应关联通常是锁定的（因为它对于系统软件的正常工作是必需的），所以您在解锁它之前需谨慎。要锁定条件 / 响应关联以使它无法被启动、停止或除去，请使用者 **-L** 标志重新发出此命令。

### 标志

- h** 将命令的用法语句写到标准输出中。
- T** 将命令的跟踪消息写到标准错误中。仅供您的软件服务组织使用。
- V** 将命令的详细消息写到标准输出中。
- U** 解锁条件 / 响应关联以使它可以被启动、停止或除去。如果条件 / 响应关联被锁定，这通常是由于它



对于系统软件的正常工作不可或缺而造成的。由于此原因，您在解锁它之前应小心谨慎。当使用 **-U** 标志解锁条件 / 响应关联时，无法使用此命令执行任何其他操作。

**-L** 锁定条件 / 响应关联以使其无法被启动、停止或删除。当使用 **-L** 标志锁定条件 / 响应关联时，无法使用此命令执行任何其他操作。

## 参数

|                  |                                                                                                                     |
|------------------|---------------------------------------------------------------------------------------------------------------------|
| <i>condition</i> | 指定与响应链接的条件名称。条件总是最先指定。                                                                                              |
| <i>node_name</i> | 指定定义条件的域中的节点。如果未指定 <i>node_name</i> ，则使用本地节点。 <i>node_name</i> 是由 <code>CT_MANAGEMENT_SCOPE</code> 环境变量所确定的作用域内的节点。 |
| <i>response</i>  | 指定一个或多个响应的名称。如果响应未与条件链接，指定多个响应与条件链接并启动对指定响应的监视。                                                                     |

## 安全性

用户需要 **IBM.Association** 资源类的写权限以运行 **startcondresp**。许可权在联系系统的访问控制表（ACL）文件中指定。请参阅 *RSCT: Administration Guide* 以获取关于 ALC 文件以及如何修改该文件的详细信息。

## 退出状态

- 0 命令成功运行。
- 1 RMC 出错。
- 2 命令行界面脚本出错。
- 3 命令行中输入错误标志。
- 4 命令行中输入错误参数。
- 5 不正确的命令行输入导致出错。

## 环境变量

### CT\_CONTACT

确定与资源监视和控制（RMC）守护程序发生会话的系统。当 `CT_CONTACT` 设置为主机名或 IP 地址时，该命令会与指定主机上的 RMC 守护程序联系。如果未设置 `CT_CONTACT`，则该命令会与运行它的本地系统的 RMC 守护程序联系。RMC 守护程序会话的目标和管理作用域确定了处理的资源类或资源。

### CT\_MANAGEMENT\_SCOPE

确定处理事件响应资源管理器（ERRM）的资源时用于与 RMC 守护程序的会话的管理作用域。管理作用域确定了处理资源的可能目标节点设置。有效值为：

- 0 指定本地作用域。
- 1 指定本地作用域。
- 2 指定对等域作用域。
- 3 指定管理域作用域。

如果此环境变量未设置，则使用本地作用域。

## 实现细节

此命令是 AIX 的 Reliable Scalable Cluster Technology（RSCT）文件集的一部分。

## 标准输出

当指定了 **-h** 标志时，此命令的用法语句被写到标准输出。所有详细消息都写到标准输出中。

## 标准错误

所有跟踪消息都写到标准错误中。

## 示例

这些示例适用于单机系统:

1. 为了用响应 “Broadcast event on-shift” 启动监视 “FileSystem space used” 条件，不管响应是否与条件链接，运行此命令:

```
startcondresp "FileSystem space used" "Broadcast event on-shift"
```

2. 为了用所有与其链接的响应启动监视 “FileSystem space used” 条件，运行此命令:

```
startcondresp "FileSystem space used"
```

3. 为了用响应 “Broadcast event on-shift” 和 “E-mail root anytime” 启动监视 “FileSystem space used” 条件，不管它们是否与条件链接，运行此命令:

```
startcondresp "FileSystem space used" "Broadcast event on-shift" "E-mail root anytime"
```

这些示例适用于管理域:

1. 为了用响应 “Broadcast event on-shift” 在管理服务器上启动监视条件 “FileSystem space used” ,不管响应是否与条件链接，在管理服务器上运行此命令:

```
startcondresp "FileSystem space used" "Broadcast event on-shift"
```

2. 为了用响应 “Broadcast event on-shift” 在受管节点 **nodeB** 上启动监视条件 “FileSystem space used” ，不管响应是否与条件链接，在管理服务器上运行此命令:

```
startcondresp "FileSystem space used":nodeB "Broadcast event on-shift"
```

该示例适用于对等域:

1. 为了用响应 “Broadcast event on-shift” （也在域中的 **nodeA** 上）启动监视在域中 **nodeA** 上的条件 “FileSystem space used” ，不管此响应是否与条件链接，在域中的任何节点上运行此命令:

```
startcondresp "FileSystem space used":nodeA "Broadcast event on-shift"
```

## 位置

**/usr/sbin/rsct/bin/startcondresp**

包含 **startcondresp** 命令

## 相关信息

书籍: 请参阅 *RSCT: Administration Guide* 以获取关于 ERRM 操作的更多信息

命令: **lscondresp**、**mkcondition**、**mkcondresp**、**mkresponse**、**stopcondresp**

信息文件: **rmccli**

---

## start-secdapclntd 命令

### 用途

**start-secdapclntd** 脚本用于启动 **secdapclntd** LDAP 客户机守护程序。

## 语法

```
/usr/sbin/start-secdapclntd [ -C CacheSize ] [ -p NumOfThread ] [ -t CacheTimeOut ] [ -T HeartBeatIntv ] [ -o ldapTimeOut ]
```

## 描述

如果 **secdapclntd** 守护程序未运行，则 **start-secdapclntd** 脚本启动此程序。如果 **secdapclntd** 守护程序已经运行，则该脚本不进行任何操作。该脚本也在启动 **secdapclntd** 守护程序之前，从以前的 **secdapclntd** 守护程序中清理端口映射程序的注册（如果有的话）。这就防止了由于端口映射程序注册失败而引起的新建守护程序的启动故障。

## 标志

在缺省情况下，**secdapclntd** 守护程序在启动时读取 **/etc/security/ldap/ldap.cfg** 文件中指定的配置信息。如果当启动 **secdapclntd** 进程时，下列选项出现在命令行中，则这些来此命令行的选项将覆盖 **/etc/security/ldap/ldap.cfg** 文件中的值。

|                                |                                                                                                                       |
|--------------------------------|-----------------------------------------------------------------------------------------------------------------------|
| <b>-C</b> <i>CacheSize</i>     | 将 <b>secdapclntd</b> 守护程序使用的最大高速缓存项设置为 <i>CacheSize</i> 项数。用户高速缓存的有效范围是 100-10,000 个条目。缺省值是 1000。组高速缓存项是用户高速缓存项的 10%。 |
| <b>-o</b> <i>ldapTimeOut</i>   | LDAP 客户机向服务器请求的超时周期（以秒计）。该值确定客户机将等待 LDAP 服务器响应的的时间长度。有效范围为 0 - 3600（1 小时）。缺省值为 60 秒。将该值设置为 0 来禁用超时并迫使客户机无限期等待。        |
| <b>-p</b> <i>NumOfThread</i>   | 将 <b>secdapclntd</b> 守护程序使用的线程数设置为 <b>NumOfThread</b> 个线程。有效范围是 1-1000。缺省值是 10。                                       |
| <b>-t</b> <i>CacheTimeout</i>  | 将高速缓存设置为在 <i>CacheTimeout</i> 秒后失效。有效范围是 60-3600 秒。缺省值是 300 秒。                                                        |
| <b>-T</b> <i>HeartBeatIntv</i> | 设置此客户机与 LDAP 服务器之间脉动信号的时间间隔。有效值是 60-3,600 秒。缺省值是 300。                                                                 |

## 示例

1. 要启动 **secdapclntd** 守护程序，请输入：

```
/usr/sbin/start-secdapclntd
```

2. 要启动使用 20 个线程和高速缓存超时值为 600 秒的 **secdapclntd**，请输入：

```
/usr/sbin/start-secdapclntd -p 20 -t 600
```

建议您在 **/etc/security/ldap/ldap.cfg** 文件中指定这些值，这样您就可以在每次启动 **secdapclntd** 进程时使用这些值。

## 文件

**/usr/sbin/start-secdapclntd** 用来启动 **secdapclntd** LDAP 客户机守护程序。

## 相关信息

**secdapclntd** 守护程序

**mksecdap**、**stop-secdapclntd**、**restart-secdapclntd**、**ls-secdapclntd** 和 **flush-secdapclntd** 命令。

**/etc/security/ldap/ldap.cfg** 文件。

---

## stop-secdapclntd 命令

### 用途

**stop-secdapclntd** 脚本是用来终止 **secdapclntd** LDAP 客户守护程序的。

### 语法

**/usr/sbin/stop-secdapclntd**

### 描述

**stop-secdapclntd** 脚本终止运行的 **secdapclntd** 守护程序进程。如果 **secdapclntd** 守护程序未在运行，则脚本将返回错误。

### 示例

要停止运行的 **secdapclntd** 守护程序的进程，请输入：

```
/usr/sbin/stop-secdapclntd
```

### 文件

**/usr/sbin/stop-secdapclntd** 用来终止 **secdapclntd** LDAP 客户守护程序。

### 相关信息

**secdapclntd** 守护程序

**mksecdap**、**start-secdapclntd**、**restart-secdapclntd**、**ls-secdapclntd** 和 **flush-secdapclntd** 命令。

**/etc/security/ldap/ldap.cfg** 文件。

---

## startdomain 命令

### 用途

使已定义的对等域联机。

### 语法

```
startdomain [ -A | -L ] [-t timeout] [ -Q quorum_type | quorum_type_name ] [-m fanout] [-h] [-TV]  
peer_domain
```

### 描述

**startdomain** 命令通过启动属于此对等域的每个节点上的资源使已联机定义的对等域联机。

**startdomain** 命令必须在定义到对等域的节点上运行。每次在对等域中运行此命令时，它都请求定义到对等域中的所有脱机节点在对等域中联机。可以在对等域中多次运行此命令。如果在对等域中定义的所有节点已经联机，则不执行操作。

**startprdomain** 命令通过在定义到对等域的节点上检测对等域配置从而确定对等域配置以用来使对等域联机。找到最新版本的对等域配置信息用来使对等域联机。在缺省情况下，使用在至少一半的节点上找到的最新版本的对等域配置。指定 **-A** 标志会导致使用在对等域中定义的所有节点上发现的最新版本对等域配置。指定 **-L** 标志会导致使用本地节点上的配置。

确定最新版本对等域配置信息时，配置超时定义何时停止检查版本并开始将对等域联机。缺省超时值是 120 秒。可以使用 **-t** 标志来更改超时值。超时值应该至少足以能够从至少一半的节点上找到最新版本的对等域配置信息。

一个节点每次只能在一个对等域上联机。当某节点已与一个对等域联机时，**startprdomain** 命令不能运行在该节点上用于另一个对等域。

## 标志

**-A** 从对等域的所有节点中查找并使用最新版对等域配置信息。如果指定了 **-L** 标志，则不能指定该标志。如果既没有指定 **-A** 标志也没有指定 **-L** 标志，则会使用来自对等域中至少一半节点上的最新版本的对等域配置信息。

**-L** 使用在本地节点上的最新版本的对等域配置信息。如果指定了 **-A** 标志，则不能指定该标志。如果既没有指定 **-A** 标志也没有指定 **-L** 标志，则会使用来自对等域中至少一半节点上的最新版本的对等域配置信息。

**-t** *timeout*

指定超时值（秒）。该标志限制了用于查找最新版本的对等域配置所需的时间量。当超过超时值时，使用此时找到的最新版本的对等域配置信息。超时值应该足够长，以便能够从至少一半的节点上找到最新版本的对等域配置信息。缺省超时值是 120 秒。

**-Q** *quorum\_type* | *quorum\_type\_name*

使您能够覆盖启动限额方式。可以指定它作为整数限额类型或限额类型名称。如果您没有指定该标志，当您创建对等域时，将使用 **mkrpdomain** 命令的 **-Q** 标志（或您环境的缺省限额方式）来指定启动限额方式。仅当限额方式已定义为**常规**或**快速**时，您才可以覆盖限额启动方式。有效值为：

**0** | 常规

指定常规启动限额规则。将接触配置信息的一半节点。

**1** | 快速

指定快速启动限额规则。将接触配置信息的一个节点。

**-m** *fanout*

指定用于此启动操作的线程的最大数目。对于指定的对等域，**-m** 标志覆盖缺省 *fanout* 值。该值作为持久属性存储在对等域的 **IBM.PeerNode** 类中。*fanout* 可以是 16 至 2048 的整数。

**-h** 将命令的用法语句写到标准输出中。

**-T** 将命令的跟踪消息写到标准错误中。仅供您的软件服务组织使用。

**-V** 将命令的详细消息写到标准输出中。

## 参数

*peer\_domain* 指定要联机的预定义对等域的名称。

## 安全性

**startprdomain** 命令的用户需要对定义到对等域的每个节点上的 **IBM.PeerDomain** 资源类的具有写权限。缺省情况下，对等域中任何节点上的 **root** 用户通过配置资源管理器拥有对此资源类的读写权限。

## 退出状态

- 0 命令成功运行。
- 1 RMC 出错。
- 2 命令行界面脚本出错。
- 3 命令行中输入错误标志。
- 4 命令行中输入错误参数。
- 5 不正确的命令行输入导致出错。
- 6 对等域定义不存在。

## 环境变量

### CT\_CONTACT

确定与资源监视和控制 (RMC) 守护程序发生会话的系统。当 CT\_CONTACT 设置为主机名或 IP 地址时, 该命令会与指定主机上的 RMC 守护程序联系。如果未设置 CT\_CONTACT, 则该命令会与运行它的本地系统的 RMC 守护程序联系。RMC 守护程序会话的目标和管理作用域确定了处理的资源类或资源。

## 限制

此命令必须从定义到对等域的节点上运行。

## 实现细节

此命令是 AIX 的 Reliable Scalable Cluster Technology (RSCT) 文件集的一部分。

## 标准输入

当指定了 **-F** "-" 标志时, 该命令从标准输入中读取一个或多个节点名。

## 标准输出

当指定了 **-h** 标志时, 此命令的用法语句被写到标准输出。所有详细消息都写到标准输出中。

## 标准错误

所有跟踪消息都写到标准错误中。

## 示例

在这些示例中, **nodeA** 是在 **AppIDomain** 上定义的节点之一。

1. 为使 **AppIDomain** 联机, 在 **nodeA** 上运行此命令:

```
startdomain AppIDomain
```

2. 要使用对等域中的所有节点以实现 **AppIDomain** 联机来获取最新版本的对等域配置信息, 请在 **nodeA** 上运行该命令:

```
startdomain -A AppIDomain
```

3. 要使用 240 秒的对等域配置信息超时值以使 **AppIDomain** 联机 (确保使用对等域中至少一半的节点), 请在 **nodeA** 上运行该命令:

```
startdomain -t 240 AppIDomain
```

## 位置

`/usr/sbin/rsct/bin/startrpdomain`

包含 **startrpdomain** 命令

## 相关信息

书籍: 请参阅 *RSCT: Administration Guide* 以获取关于对等域操作的信息。

命令: **lsrpdomain**、**lsrpnnode**、**mkrpdomain**、**preprnode**、**stoprpdomain**

信息文件: **rmccli**, 关于与 RMC 相关的命令的一般信息

---

## startrpnnode 命令

### 用途

使一个或多个节点与对等域联机。

### 语法

**startrpnnode** [-h] [-TV] *node\_name1* [*node\_name2* ...]

**startrpnnode** -f | -F { *file\_name* | "-" } [-h] [-TV]

### 描述

**startrpnnode** 命令将一个或多个脱机节点联机到一个对等域上。对等域是由运行该命令的联机对等域来确定的。该命令必须在与期望的对等域联机的节点上运行。

要联机的节点必须已经使用 **addrpnnode** 命令或 **mkrpdomain** 命令定义到此对等域中。该节点必须不与任何其他对等域联机。

### 标志

**-f** | **-F** { *file\_name* | "-" }

从 *file\_name* 读取节点名的列表。扫描文件的每一行以获取一个节点名。磅符号 (#) 表明行的其余部分 (如果 # 位于列 1 中则为整行) 是一个注释。

使用 **-f "-"** 或 **-F "-"** 来指定 **STDIN** 作为输入文件。

**-h** 将命令的用法语句写到标准输出中。

**-T** 将命令的跟踪消息写到标准错误中。仅供您的软件服务组织使用。

**-V** 将命令的详细消息写到标准输出中。

### 参数

*node\_name1* [*node\_name2* ...] 指定将要与对等域联机的节点的对等域节点名。您可以使用 **startrpnnode** 命令使一个或多个节点联机。您指定节点名所用的格式必须与使用 **addrpnnode** 命令或 **mkrpdomain** 命令指定这些节点所用的格式完全相同。要列出对等域节点名,请运行 **lsrpnnode** 命令。

## 安全性

**startprnode** 命令的用户需要对要在对等域中启动的每个节点上的 **IBM.PeerNode** 资源类具有写权限。缺省情况下，对等域中任何节点上的 **root** 用户通过配置资源管理器拥有对此资源类的读写权限。

## 退出状态

- 0 命令成功运行。
- 1 RMC 出错。
- 2 命令行界面脚本出错。
- 3 命令行中输入错误标志。
- 4 命令行中输入错误参数。
- 5 不正确的命令行输入导致出错。

## 环境变量

### CT\_CONTACT

确定与资源监视和控制（RMC）守护程序发生会话的系统。当 **CT\_CONTACT** 设置为主机名或 IP 地址时，该命令会与指定主机上的 RMC 守护程序联系。如果未设置 **CT\_CONTACT**，则该命令会与运行它的本地系统的 RMC 守护程序联系。RMC 守护程序会话的目标和管理作用域确定了处理的资源类或资源。

## 限制

该命令必须在联机到对等域的节点上运行。要联机的节点必须与对等域脱机，也不能与其他对等域联机，并且必须从命令运行处可到达。

## 实现细节

此命令是 AIX 的 Reliable Scalable Cluster Technology (RSCT) 文件集的一部分。

## 标准输入

当指定了 **-f "-"** 或 **-F "-"** 标志时，该命令从标准输入中读取一个或多个节点名。

## 标准输出

当指定了 **-h** 标志时，此命令的用法语句被写到标准输出。所有详细消息都写到标准输出中。

## 标准错误

所有跟踪消息都写到标准错误中。

## 示例

在本示例中，**nodeA** 已定义并且联机到 **ApplDomain** 上，从 **nodeA** 可达到 **nodeB**，并且 **nodeB** 没有与 **ApplDomain** 或其他对等域联机。要使 **nodeB** 联机到 **ApplDomain** 上，请在 **nodeA** 上运行此命令：

```
startprnode nodeB
```



## 位置

`/usr/sbin/rsct/bin/startprnode`

包含 **startprnode** 命令

## 相关信息

书籍: 请参阅 *RSCT: Administration Guide* 以获取关于对等域操作的信息。

命令: **addrprnode**、**lsrprnode**、**preprprnode**、**rnrprnode**、**stopprnode**

信息文件: **rmccli**, 关于与 RMC 相关的命令的一般信息

---

## startsrc 命令

### 用途

启动子系统、子系统组或子服务器。

### 语法

#### 启动子系统

**startsrc** [ **-a** *Argument*] [ **-e** *Environment*] [ **-h** *Host*] { **-s** *Subsystem* | **-g** *Group*}

#### 启动子服务器

**startsrc** [ **-h** *Host*] **-t** *Type* [ **-o** *Object*] [ **-p** *SubsystemPID*]

### 描述

**startsrc** 命令向系统资源控制器 (System Resource Controller, SRC) 发送请求以启动子系统或子系统组, 或向启动子服务器的子系统发送一个信息包。

如果启动子服务器的请求发送到 SRC, 且该子服务器所属的子系统当前不活动, 则 SRC 会启动子系统, 并将启动子服务器的请求发送给子系统。

### 标志

- |                              |                                                                                                                                                                                                                                                                                                                                                               |
|------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-a</b> <i>Argument</i>    | 指定子系统执行时传递给子系统的自变量字符串。该字符串从命令行发送, 并附加在子系统对象类中的命令行自变量上。指定的 <i>Argument</i> 字符串最大长度为 1200 个字符, 否则命令就会失败。根据与 shell 使用的相同的规则, 该命令参数由 SRC 传递到子系统。引号括起的字符串作为单个参数传递, 并且以引号括起的字符串外的空格定界一个参数。可使用单引号和双引号。                                                                                                                                                               |
| <b>-e</b> <i>Environment</i> | 当子系统执行时, 指定一个放入子系统环境中的环境字符串。指定的 <i>Environment</i> 字符串的最大长度为 1200 个字符, 否则命令将失败。使用与 shell 所使用的相同的规则, SRC 为子系统设置环境。<br><br>引号括起的字符串指定给一个单一的环境变量, 并且引号括起的字符串外的空格定界了每个要设置的环境变量。例如: <code>-e "HOME=/tmp TERM=dumb MESSAGE=\"Multiple word message\""</code> 将 HOME=/tmp 设置为子系统第一个环境变量, 将 TERM=dumb 设置为子系统第二个环境变量, 将 MESSAGE="Multiple word message" 设置为子系统第三个环境变量。 |
| <b>-g</b> <i>Group</i>       | 指定一组要启动的子系统。如果 <i>Group</i> 名称未包含在子系统对象类中, 命令将失败。                                                                                                                                                                                                                                                                                                             |

|                               |                                                                                                                                                                                             |
|-------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-h</b> <i>Host</i>         | 指定在其上请求此启动操作的外部主机。本地用户必须作为“root”用户运行。必须配置远程系统以接受系统资源控制器的请求。也就是说， <b>srcmstr</b> 守护程序（请参阅 <b>/etc/inittab</b> ）必须用 <b>-r</b> 标志启动，并且必须配置 <b>/etc/hosts.equiv</b> 或 <b>.rhosts</b> 文件以允许远程请求。 |
| <b>-o</b> <i>Object</i>       | 指定子服务器对象作为字符串传递给子系统。确定 <i>Object</i> 字符串的有效性是子系统的责任。                                                                                                                                        |
| <b>-p</b> <i>SubsystemPID</i> | 指定启动子服务器请求要发送到的子系统的特定实例。                                                                                                                                                                    |
| <b>-s</b> <i>Subsystem</i>    | 指定要启动的子系统。 <i>Subsystem</i> 可以是实际的子系统名，或子系统的同义词名称。如果 <i>Subsystem</i> 名称没有包含在子系统对象类中，则该命令失败。                                                                                                |
| <b>-t</b> <i>Type</i>         | 指定要启动的子服务器。如果子服务器对象类中未包含 <i>Type</i> ，命令将失败。                                                                                                                                                |

## 示例

1. 要使用参数和环境变量启动一个子系统，请输入：

```
startsrc -s srctest -a "-D DEBUG" -e "TERM=dumb HOME=/tmp"
```

这将使用其环境中的“TERM=dumb”、“HOME=/tmp”并且将“-D DEBUG”作为该子系统的两个参数来启动 srctest 子系统。

2. 要启动外部主机上的子系统组，请输入：

```
startsrc -g tcpip -h zork
```

这将启动 zork 机器上 tcpip 子系统组中所有的子系统。

3. 要启动一个子服务器，请输入：

```
startsrc -t tester
```

这将向拥有 tester 子系统的子系统发送启动子服务器的请求。

4. 要使用命令参数启动一个子系统，请输入：

```
startsrc -s srctest -a "-a 123 -b \"4 5 6\""
```

这将使“-a”作为 srctest 子系统的第一个参数，“123”作为第二个参数，“-b”作为第三个参数，而“456”作为第四个参数。

## 文件

|                                |                          |
|--------------------------------|--------------------------|
| <b>/etc/objrepos/SRCsubsys</b> | 指定 SRC 子系统配置对象类。         |
| <b>/etc/objrepos/SRCsubsvr</b> | 指定 SRC 子服务器配置对象类。        |
| <b>/etc/services</b>           | 定义用于因特网服务的套接字和协议。        |
| <b>/dev/SRC</b>                | 指定 <b>AF_UNIX</b> 套接字文件。 |
| <b>/dev/.SRC-unix</b>          | 指定临时套接字文件位置。             |

## 相关信息

**topsrc** 命令、**refresh** 命令。

《操作系统与设备管理》中的『系统资源控制器』提供了对子系统、子服务器和系统资源控制器的说明。

---

## startup 命令

### 用途

在系统启动时打开记帐功能。

### 语法

**/usr/sbin/acct/startup**

### 描述

如果 **/etc/rc** 命令文件调用 **startup** 命令，则该命令会在系统启动时打开记帐功能。请参阅 **startup** 示例，了解如何将该命令加入 **/etc/rc** 文件。

### 安全性

访问控制：此命令只授予 **adm** 组的成员以执行（x）权限。

### 示例

要在系统启动时打开记帐功能，请将以下行加入 **/etc/rc** 文件中：

```
/usr/bin/su - adm -c /usr/sbin/acct/startup
```

**startup** shell 过程然后将记录时间，并清除前一天的记录。

### 文件

**/usr/sbin/acct**                    记帐命令的路径。

### 相关信息

**shutacct** 命令、**turnacct** 命令。

有关“记帐系统”、每日和每月报告的准备以及记帐文件的更多信息，请参阅《操作系统与设备管理》中的『系统记帐』。

《操作系统与设备管理》中的『设置记帐子系统』说明了建立记帐系统所必须采取的步骤。

---

## startvsd 命令

### 用途

**startvsd** - 使一个虚拟共享磁盘可用并激活它。

### 语法

**startvsd [-p | -b] [-a | vsd\_name ...]**

### 描述

**startvsd** 命令使指定的虚拟共享磁盘可用并激活它们。它在指定的虚拟共享磁盘上运行 **preparevsd** 命令之后运行 **resumevsd** 命令是等效的。

可以使用“系统管理界面工具”（SMIT）来运行该命令。要使用 SMIT，输入：

```
smit vsd_mgmt
```

并选择 **Start a Virtual Shared Disk** 选项。

在常规情况下，您不应发出该命令。“可恢复的”虚拟共享磁盘子系统使用该命令以受控制的方式来管理共享磁盘。如果您发出该命令，结果可能是不可预测的。

## 标志

- p** 指定为全局卷组定义的主服务器节点将是活动的服务器。  
有关更多信息，请参阅 *RSCT: Managing Shared Disks*。
- b** 指定为全局卷组定义的辅助服务器节点将是活动的服务器。
- a** 指定已定义的所有虚拟共享磁盘都要启动。

## 参数

*vsd\_name* 指定一个虚拟共享磁盘。

## 安全

您必须具有 root 用户权限以运行该命令。

## 退出状态

- 0** 说明命令已成功完成。
- 非零** 表明发生错误。

## 限制

您必须从对等域中联机的节点发出该命令。要使对等域联机，请使用 **startprdomain** 命令。要使现有对等域中一个特定的节点联机，请使用 **startprnode** 命令。有关创建和管理 RSCT 对等域的更多信息，请参阅 *RSCT: Administration Guide*。

在常规情况下，您不应发出该命令。“可恢复的”虚拟共享磁盘子系统使用该命令以受控制的方式来管理共享磁盘。如果您发出该命令，结果可能是不可预测的。

## 标准输出

当前 RVSD 子系统运行级别。

## 示例

要使虚拟共享磁盘 **vsd1vg1n1** 可用并激活它，请输入：

```
startvsd vsd1vg1n1
```

## 位置

**/opt/rsct/vsd/bin/startvsd**

## 相关信息

命令: **cfgvds**、**lsvds**、**preparevds**、**resumevds**、**stopvds**、**suspendvds**、**ucfgvds**

---

## startx 命令

### 用途

初始化一个 X 会话。

### 语法

**startx** [ **-d** *Display:0* ] [ **-t** | **-w** ] [ **-x** *Startup* | [ **-r** *Resources* ] [ **-m** *Window\_Manager* ] ] [ **-wait** ]

### 描述

**startx** 命令简化了启动 X 会话的过程。

该命令执行下列操作:

- 设置用户的 **DISPLAY** 环境变量以标识 X 服务器到 X 客户机
- 当从工作站运行时, 启动 X 服务器
- 启动 X 客户机。

**startx** 命令将 X 服务器和 X 客户机的错误消息重定向至用户的 **XERRORS** 环境变量指定的文件。该处理对调试很有用, 并且使 X 服务器在工作站上有一个清洁的启动和关机外观。

如果 *startup* 脚本文件名没有用 **-x** 选项在命令行中给出, 那么 **startx** 命令就会搜索一个由用户的 **XINITRC** 环境变量指定的文件。如果未设置 **XINITRC** 环境变量, **startx** 命令就会在用户的主目录中分别搜索名为 **.Xinit**、**.xinit**、**.Xinitrc**、**.xinitrc** 或 **.xsession** 的文件来开始 X 客户机程序。

如果未找到启动文件, **startx** 命令运行在命令行上由 **-m** 选项指定的窗口管理器, 或在找到相关的配置文件 (**.mwmrc**、**.twmrc**、**.awmrc** 或 **.uwmrc**, 分别地) 之后, 调用窗口管理器 **mwm**、**twm**、**awm** 或 **uwm**。如果没有在用户主目录中找到窗口管理器配置文件, **startx** 将启动 **Xterm** 客户机和 **mwm** 窗口管理器。

当启动文件找不到时, **startx** 命令也将在命令行上使用 **-r** 选项指示装入资源文件, 或分别装入用户主目录中的名为 **.Xdefaults**、**.xdefaults**、**.Xresources** 或 **.xresources** 的文件。如果未找到 X 资源文件, 将不个性化 X 会话。

如果工作站中存在一个启动文件并且用户未装入任何资源, 则 **startx** 命令中的 **xinit** 命令将尝试装入 **.Xdefaults** 文件。

当 X 会话从 **/dev/lft\*** 启动时, 就假定使用工作站。如果不是这种情况, 就必须使用 **-t** 或 **-w** 选项。

### 标志

|                                 |                                                 |
|---------------------------------|-------------------------------------------------|
| <b>-d</b> <i>Display :0</i>     | 指定在启动过程中传递给 X 客户机的 X 服务器的显示名称。                  |
| <b>-m</b> <i>Window_Manager</i> | 当未找到 <i>startup</i> 脚本时, 启动窗口管理器。               |
| <b>-r</b> <i>Resources</i>      | 当未找到 <i>startup</i> 脚本时, 装入资源文件。                |
| <b>-t</b>                       | 为一个 X 终端启动 X 客户机。                               |
| <b>-w</b>                       | 为工作站上的 X 窗口会话启动 X 服务器和 X 客户机。                   |
| <b>-wait</b>                    | 当 <b>xdm</b> 命令调用 <b>startx</b> 时, 防止重新启动 X 会话。 |
| <b>-x</b> <i>Startup</i>        | 使用 <i>startup</i> 脚本启动 X 窗口会话。                  |

注：您可以单独或同时使用 **-m** 和 **-r** 选项，或者使用 **-x** 选项，但是您不能在使用 **-x** 选项的同时使用 **-m** 和 **-r** 选项。在启动脚本中，启动窗口管理器会话、装入 X 资源以及产生 X 客户机是用户的责任。

## 示例

1. 要在工作站上或 X 终端上启动 X 会话，请输入：

```
startx
```

2. 要在工作站上强制启动 X 会话，请输入：

```
startx -w
```

3. 要为 X 终端启动 X 会话，并注销用户的 Telnet 会话，请输入：

```
startx; kill -9 $$
```

4. 要使用 **.xinitrc** 脚本启动 X 会话，请输入：

```
startx -x .xinitrc
```

5. 要使用 **mwm** 窗口管理器启动 X 会话，请输入：

```
startx -m mwm
```

但是，如果找到启动脚本文件，则忽略 **-w** 选项。

6. 在启动脚本中，启动窗口管理器、装入 X 资源以及产生 X 客户机是用户的责任。以下是 **.xsession** 脚本的一个示例。

```
#!/bin/csh
(mwm &)
xrdb -load .Xdefaults
(xclock -g 75x75+0+0 &)
(xbiff -g 75x75+101-0 &)
if ("/dev/lft*" == "`tty`") then
    aixterm -g 80x24+0+0 +ut -C -T `hostname`
else
    aixterm -g 80x24+0+0 +ut -T `hostname`
endif
```

对于工作站，**startup** 脚本中的最后一行应该是前台 **aixterm** 命令，该命令带有 **-C** 选项表示控制台信息。

对于 X 终端，**startup** 脚本中的最后一行应该是不带有 **-C** 选项的前台 **aixterm** 命令。另外，由于某些 X 终端在关闭时不终止 **telnet** 会话，所以用户必须在使用热键切换至 X 会话前退出当前的 **telnet** 会话。

**/usr/lib/X11/xdm/Xsession** 文件中的 **xdm** 命令也可以使用 **startx** 命令。这为 **xdm** 命令提供了 **startx** 命令的功能。

## 文件

以下是启动 X 会话一贯使用的文件名。

**\$HOME/.xerrors**

其中，**startx** 用来重定向错误消息。在缺省情况下，**startx** 将错误重定向至用户主目录中的 **.xerrors** 文件中。

**\$HOME/.Xinit,**

**\$HOME/.xinit,**

**\$HOME/.Xinitrc,**

**\$HOME/.xinitrc,**

**\$HOME/.xsession**

作为包含 shell 命令的“启动文件”来启动窗口管理器、装入 X 资源并产生 X 客户机。

|                                 |                                             |
|---------------------------------|---------------------------------------------|
| <code>\$HOME/.Xdefaults,</code> | 作为装入的 X 资源文件来设置 X 客户机的用户首选项。                |
| <code>\$HOME/.xresources</code> |                                             |
| <code>\$HOME/.mwmrc</code>      | <code>mwm</code> 配置文件。                      |
| <code>\$HOME/.twmrc</code>      | <code>twm</code> 配置文件。                      |
| <code>\$HOME/.awmrc</code>      | <code>awm</code> 配置文件。                      |
| <code>\$HOME/.uwmrc</code>      | <code>uwm</code> 配置文件。                      |
| <code>/dev/lft*</code>          | 终端或 tty、工作站初始 <code>login shell</code> 的界面。 |

## 相关信息

`mwm` 命令、`xinit` 命令、`xdm` 命令、`aixterm` 命令、`telnet`、`tn` 或 `tn3270` 命令、`X` 命令以及 `xrdb` 命令。

---

## statd 守护程序

### 用途

为 NFS 上的锁定服务提供崩溃和恢复功能。

### 语法

```
/usr/sbin/rpc.statd [-t threads]
```

### 描述

`statd` 守护程序与 `lockd` 守护程序相互作用，为网络文件系统（Network File System, NFS）上的锁定服务提供崩溃和恢复功能。`statd` 守护程序应总是在 `lockd` 守护程序之前启动。

`statd` 守护程序由下列 SRC 命令启动和停止：

```
startsrc -s rpc.statd
```

```
stopsrc -s rpc.statd
```

状态监视器保持有关连接位置的信息以及 `/var/statmon/sm` 目录、`/var/statmon/sm.bak` 目录和 `/var/statmon/state` 文件中的状态。当重新启动时，`statd` 守护程序查询这些文件，并尝试重建它在终止前所拥有的连接。要在事先不了解现有的锁或状态的情况下重新启动 `statd` 守护程序随后启动 `lockd` 守护程序，请在重新启动 `statd` 守护程序前将这些文件删除。

### 标志

`-t threads` 指定允许的 `rpc.statd` 线程最大数目。缺省值为 50。

## 相关信息

`lockd` 守护程序。

NFS 命令列表。

《操作系统与设备管理》中的『网络文件系统（NFS）』。

---

## statvsd 命令

### 用途

**statvsd** - 显示一个节点的虚拟共享磁盘驱动程序统计信息。

### 语法

**statvsd**

### 描述

**statvsd** 命令显示一个节点的虚拟共享磁盘统计信息。例如，在一台繁忙的服务器上正在增加的“排队等待伙伴缓冲区的请求”数目是正常的且并不一定意味着有问题。更大的值是“平均伙伴缓冲区 wait\_queue 大小”，这是当发出 **statvsd** 命令时，排队获取伙伴缓冲区的请求的数目。有关输出行的含义，请参阅『示例』一节。

### 标志

无

### 参数

无

### 安全

您必须在 AIX **bin** 组中才能运行此命令。

### 退出状态

**0**                说明命令已成功完成。  
**非零**            表明发生错误。

### 限制

您必须从对等域中联机的节点发出该命令。要使对等域联机，请使用 **startprdomain** 命令。要使现有对等域中一个特定的节点联机，请使用 **startprnode** 命令。有关创建和管理 RSCT 对等域的更多信息，请参阅 *RSCT: Administration Guide*。

### 标准输出

当前 RVSD 子系统运行级别。

### 示例

以下示例显示虚拟共享磁盘的驱动程序统计信息。

1. 标题行指示了代码的版本和发行版。例如：  
VSD driver (vsdd): IP/SMP V4 R1
2. 虚拟共享磁盘并行性级别的缺省值为 9 并且是设备驱动程序在内核中进行的 **uphysio** 调用上的 **buf\_cnt** 参数。例如：  
9 vsd 并行性
3. 最大 IP 消息大小以字节为单位。例如：



- 61440 vsd 最大 IP 消息大小
4. 必须等待请求程序段的请求的数目。例如:  
61440 vsd 最大 IP 消息大小
  5. 必须等待 pbuf (用于提交到磁盘的实际 I/O 请求的缓冲区) 的请求的数目。例如:  
0 个排队等待 pbuf 的请求
  6. 必须等待伙伴缓冲区的请求的数目。伙伴缓冲区是在服务器上用来暂时存储于客户机节点处开始的 I/O 操作的日期。例如:  
2689 个排队等待伙伴缓冲区的请求
  7. 当发出 **statvsd** 命令时排队获取伙伴缓冲区的请求的数目。例如:  
0 平均伙伴缓冲区 wait\_queue 大小
  8. 通常因为超出范围的序号或内部问题, 服务器拒绝的请求的数目。例如:  
4 个拒绝的请求
  9. 客户机拒绝的响应的数目。通常因为在重试已经发送到服务器之后响应到达。例如:  
0 个拒绝的响应
  10. 放置在重新工作队列上的请求的数目。例如:  
0 个请求重新工作
  11. 不在 64 字节边界上的读请求的数目。例如:  
0 个 64 字节未对齐读请求
  12. 获取 DMA 不足信息的请求的数目。此条件将需要以非零复制方式来执行 I/O 操作。例如:  
0 DMA 空间不足
  13. 已超时的请求的数目。当前超时周期大约为 15 分钟。例如:  
0 次超时
  14. 重试有固定的次数。重试计数器显示对特定“重试存储区”已重试的请求的数目。再往右显示的数字代表要求了更多重试的请求的数目。当一个请求用完了它的重试次数时, 该请求将被记录为超时。例如:  
重试次数: 0 0 0 0 0 0 0 0 0  
0 总重试次数
  15. 序号由设备驱动程序在内部使用。这些序号由设备驱动程序和“可恢复的”虚拟共享磁盘子系统来管理。  
例如:  
非零序号  

| 节点号 | 预期号    | 发送号 | 排除? | 代号: 0 |
|-----|--------|-----|-----|-------|
| 11  | 125092 | 0   |     |       |

带有零序号的 11 个节点正常工作: 1 3 5 7 9 11 12 13 14 15 16

## 位置

**/opt/rsct/vsd/bin/statvsd**

## 相关信息

命令: **ctlvsd**、**vsdnode**

有关调整虚拟共享磁盘性能的信息, 请参考 *RSCT: Managing Shared Disks*。

---

## stopcondresp 命令

### 用途

停止监视具有一个或多个链接响应的条件。

### 语法

要停止监视一个条件:

```
stopcondresp [-q] [-h] [-TV] condition[:node_name] [response [response...]]
```

要对条件 / 响应关联进行解锁或锁定:

```
stopcondresp {-U | -L} [-h] [-TV] condition[:node_name] response
```

### 描述

**stopcondresp** 命令停止对具有一个或多个链接响应的条件的监视。如果没有指定响应，则停止条件的所有链接响应。如果指定一个或多个响应，则只停止那些链接到条件的响应。如果条件发生，则不运行响应。如果条件没有活动的响应，则该条件不再受监视。

如果为了使系统软件能正常工作而需要特殊条件 / 响应关联，则它可能被锁定。锁定的条件 / 响应关联无法由 **stopcondresp** 命令来停止。如果您对 **stopcondresp** 命令指定的条件 / 响应关联被锁定，它不会停止；而会生成一个错误，通知您该条件 / 响应关联已锁定。要解锁条件 / 响应关联，您可以使用 **-U** 标志。条件 / 响应关联通常是锁定的（因为它对于系统软件的正常工作是必需的），所以您在解锁它之前需谨慎。

### 标志

- q** 当 *condition* 或 *response* 不存在时，或与 *response* 链接的 *condition* 不受监视时，不返回错误。
- h** 将命令的用法语句写到标准输出中。
- T** 将命令的跟踪消息写到标准错误中。仅供您的软件服务组织使用。
- V** 将命令的详细消息写到标准输出中。
- U** 解锁条件 / 响应关联以使它可以被启动、停止或删除。如果条件 / 响应关联被锁定，这通常是由于它对于系统软件的正常工作不可或缺而造成的。由于此原因，您在解锁它之前应小心谨慎。当使用 **-U** 标志解锁条件 / 响应关联时，无法使用此命令执行任何其他操作。
- L** 锁定条件 / 响应关联以使它无法被启动、停止或删除。当使用 **-L** 标志锁定条件 / 响应关联时，无法使用此命令执行任何其他操作。

### 参数

- condition* 指定与响应链接的条件名称。条件总是最先指定。
- node\_name* 指定定义条件的域中的节点。如果未指定 *node\_name*，则使用本地节点。*node\_name* 是由 `CT_MANAGEMENT_SCOPE` 环境变量所确定的作用域内的节点。
- response* 指定一个或多个响应的名称。停止对指定响应的监视。（如果指定的响应未链接到该条件，则忽略该响应。）

## 安全性

用户需要 **IBM.Association** 资源类的写权限以运行 **stopcondresp**。许可权在联系系统的访问控制表（ACL）文件中指定。请参阅 *RSCT: Administration Guide* 以获取关于 ALC 文件以及如何修改该文件的详细信息。

## 退出状态

- 0 命令成功运行。
- 1 RMC 出错。
- 2 命令行界面脚本出错。
- 3 命令行中输入错误标志。
- 4 命令行中输入错误参数。
- 5 不正确的命令行输入导致出错。

## 环境变量

### CT\_CONTACT

确定与资源监视和控制（RMC）守护程序发生会话的系统。当 CT\_CONTACT 设置为主机名或 IP 地址时，该命令会与指定主机上的 RMC 守护程序联系。如果未设置 CT\_CONTACT，则该命令会与运行它的本地系统的 RMC 守护程序联系。RMC 守护程序会话的目标和管理作用域确定了处理的资源类或资源。

### CT\_MANAGEMENT\_SCOPE

确定处理事件响应资源管理器（ERRM）的资源时用于与 RMC 守护程序的会话的管理作用域。管理作用域确定了处理资源的可能目标节点设置。有效值为：

- 0 指定本地作用域。
- 1 指定本地作用域。
- 2 指定对等域作用域。
- 3 指定管理域作用域。

如果此环境变量未设置，则使用本地作用域。

## 实现细节

此命令是 AIX 的 Reliable Scalable Cluster Technology（RSCT）文件集的一部分。

## 标准输出

当指定了 **-h** 标志时，此命令的用法语句被写到标准输出。所有详细消息都写到标准输出中。

## 标准错误

所有跟踪消息都写到标准错误中。

## 示例

这些示例适用于单机系统：

1. 要对与“Broadcast event on-shift”响应相链接的“FileSystem space used”条件停止监视，请运行此命令：  
`stopcondresp "FileSystem space used" "Broadcast event on-shift"`
2. 要对使用其所有链接响应的“FileSystem space used”条件停止监视，请运行此命令：

```
stopcondresp "FileSystem space used"
```

该示例适用于管理域:

1. 要对在受管节点 **nodeB** 上且与“Broadcast event on-shift”响应相链接的“FileSystem space used”条件停止监视, 请在管理服务器上运行此命令:

```
stopcondresp "FileSystem space used:nodeB" "Broadcast event on-shift"
```

该示例适用于对等域:

1. 要对在节点 **nodeA** 上且与“Broadcast event on-shift”响应相链接的“FileSystem space used”条件停止监视, 请在域中的任何节点上运行此命令:

```
stopcondresp "FileSystem space used:nodeA" "Broadcast event on-shift"
```

## 位置

**/usr/sbin/rsct/bin/stopcondresp**

包含 **stopcondresp** 命令

## 相关信息

书籍: 请参阅 *RSCT: Administration Guide* 以获取关于 ERRM 操作的更多信息

命令: **lscondresp**、**mkcondition**、**mkcondresp**、**mkresponse**、**startcondresp**

信息文件: **rmcccli**

---

## stoprpdomain 命令

### 用途

使联机的对等域脱机。

### 语法

```
stoprpdomain [-f] [-h] [-TV] peer_domain
```

### 描述

**stoprpdomain** 命令使当前在对等域中联机的所有节点脱机。不从节点上除去对等域定义。

该命令必须在联入对等域的节点上运行。如果该命令在与对等域脱机的节点上运行, 将不执行任何操作。

必须使用 **-f** 标志来忽略子系统对对等域脱机请求的拒绝。如果对等域资源正忙(例如共享磁盘), 则子系统可能会拒绝这种请求。在这种情况下指定 **-f** 标志向子系统表明, 无论资源状态如何都必须使对等域脱机。

### 标志

- f** 强制子系统接受停止请求, 否则子系统将不会接受。
- h** 将命令的用法语句写到标准输出中。
- T** 将命令的跟踪消息写到标准错误中。仅供您的软件服务组织使用。
- V** 将命令的详细消息写到标准输出中。

## 参数

*peer\_domain* 指定要脱机的联机对等域的名称。

## 安全性

**stoprpdomain** 命令的用户需要将 **IBM.PeerDomain** 资源类的写权限写到每个定义在对等域的节点中。缺省情况下，对等域中任何节点上的 **root** 用户通过配置资源管理器拥有对此资源类的读写权限。

## 退出状态

- 0 命令成功运行。
- 1 RMC 出错。
- 2 命令行界面脚本出错。
- 3 命令行中输入错误标志。
- 4 命令行中输入错误参数。
- 5 不正确的命令行输入导致出错。
- 6 对等域定义不存在。

## 环境变量

### CT\_CONTACT

确定与资源监视和控制 (RMC) 守护程序发生会话的系统。当 **CT\_CONTACT** 设置为主机名或 IP 地址时，该命令会与指定主机上的 RMC 守护程序联系。如果未设置 **CT\_CONTACT**，则该命令会与运行它的本地系统的 RMC 守护程序联系。RMC 守护程序会话的目标和管理作用域确定了处理的资源类或资源。

## 限制

该命令必须在联入对等域的节点上运行。

## 实现细节

此命令是 AIX 的 Reliable Scalable Cluster Technology (RSCT) 文件集的一部分。

## 标准输入

当指定了 **-f "-"** 或 **-F "-"** 标志时，该命令从标准输入中读取一个或多个节点名。

## 标准输出

当指定了 **-h** 标志时，此命令的用法语句被写到标准输出。所有详细消息都写到标准输出中。

## 标准错误

所有跟踪消息都写到标准错误中。

## 示例

在这些示例中，**nodeA** 是定义的节点之一，并且联在 **AppIDomain** 上。

1. 要使 **AppIDomain** 脱机，请在 **nodeA** 上运行此命令：

```
stoprpdomain App1Domain
```

2. 要使 **App1Domain** 脱机，同时确保任何子系统都不会拒绝停止请求，请在 **nodeA** 上运行此命令：

```
stoprpdomain -f App1Domain
```

## 位置

**/usr/sbin/rsct/bin/stoprpdomain**

包含 **stoprpdomain** 命令

## 相关信息

书籍：请参阅 *RSCT: Administration Guide* 以获取关于对等域操作的信息。

命令：**lsrpdomain**、**lsrpnnode**、**mkrpdomain**、**preprpnnode**、**startrpdomain**

信息文件：**rmccli**，关于与 RMC 相关的命令的一般信息

---

## stoprpnnode 命令

### 用途

使一个或多个节点从对等域脱机。

### 语法

```
stoprpnnode [-f] [-h] [-TV] node_name1 [node_name2...]
```

```
stoprpnnode -F { file_name | "-" } [-f] [-h] [-TV]
```

### 描述

**stoprpnnode** 命令使联机节点从对等域脱机。对等域是由运行该命令的联机对等域来确定的。该命令必须在与期望的对等域联机的节点上运行。

必须使用 **-f** 标志来忽略子系统对于节点脱机请求的拒绝。如果节点资源正忙（例如共享磁盘的情况），子系统可能会拒绝这种请求。在这种情况下指定 **-f** 标志向子系统表明，无论资源状态如何都必须使节点脱机。

如果该命令被用来通过指定多个 *node\_name* 参数使多个节点脱机，并且运行该命令的节点在列表中，那么这个节点最后才会脱机。

### 标志

**-f** 强制子系统接受停止请求，否则子系统将不会接受。

**-F { *file\_name* | "-" }**

从 *file\_name* 读取节点名的列表。扫描文件的每一行以获取一个节点名。磅符号（#）表明行的其余部分（如果 # 位于列 1 中则为整行）是一个注释。

使用 **-F "-"** 来指定 **STDIN** 作为输入文件。

**-h** 将命令的用法语句写到标准输出中。

**-T** 将命令的跟踪消息写到标准错误中。仅供您的软件服务组织使用。

**-V** 将命令的详细消息写到标准输出中。

## 参数

`node_name1`[`node_name2...`] 指定要从对等域脱机的节点的对等域节点名。您指定节点名所用的格式必须与使用 **addrpnode** 命令或 **mkrpdomain** 命令指定这些节点所用的格式完全相同。要列出对等域节点名,请运行 **lsrpnnode** 命令。

## 安全性

**stoprpnnode** 命令的用户需要将在对等域中停止的每个节点上的 **IBM.PeerNode** 资源类的写许可权。缺省情况下,对等域中任何节点上的 **root** 用户通过配置资源管理器拥有对此资源类的读写权限。

## 退出状态

- 0 命令成功运行。
- 1 RMC 出错。
- 2 命令行界面脚本出错。
- 3 命令行中输入错误标志。
- 4 命令行中输入错误参数。
- 5 不正确的命令行输入导致出错。

## 环境变量

### CT\_CONTACT

确定与资源监视和控制 (RMC) 守护程序发生会话的系统。当 **CT\_CONTACT** 设置为主机名或 IP 地址时,该命令会与指定主机上的 RMC 守护程序联系。如果未设置 **CT\_CONTACT**,则该命令会与运行它的本地系统的 RMC 守护程序联系。RMC 守护程序会话的目标和管理作用域确定了处理的资源类或资源。

## 限制

该命令必须在联入对等域的节点上运行。该节点 (命令在该节点上运行) 必须达到要脱机的节点。

## 实现细节

此命令是 AIX 的 Reliable Scalable Cluster Technology (RSCT) 文件集的一部分。

## 标准输入

当指定了 **-F "-"** 标志时,该命令从标准输入中读取一个或多个节点名。

## 标准输出

当指定了 **-h** 标志时,此命令的用法语句被写到标准输出。所有详细消息都写到标准输出中。

## 标准错误

所有跟踪消息都写到标准错误中。

## 示例

在这些示例中, **nodeA** 和 **nodeB** 联在 **AppIDomain** 上。

1. 要使 **nodeB** 脱机,请在 **nodeA** 上运行此命令:

```
stoprnode nodeB
```

2. 要使 **nodeB** 脱机并强制执行脱机请求，请在 **nodeA** 上运行此命令：

```
stoprnode -f nodeB
```

## 位置

**/usr/sbin/rsct/bin/stoprnode**

包含 **stoprnode** 命令

## 相关信息

书籍：请参阅 *RSCT: Administration Guide* 以获取关于对等域操作的信息。

命令：**addrpnode**、**lsrnode**、**preprnode**、**rmpnode**、**startpnode**

信息文件：**rmccli**，关于与 RMC 相关的命令的一般信息

---

## stopsrc 命令

### 用途

停止子系统、子系统组或子服务器。

### 语法

#### 停止子系统

```
stopsrc [ -h Host ] [ -f | -c ] { -a | -g Group | -p SubsystemPID | -s Subsystem }
```

#### 停止子服务器

```
stopsrc [ -h Host ] [ -f ] -t Type [ -p SubsystemPID ] [ -P SubserverPID | -o Object ]
```

### 描述

**stopsrc** 命令向系统资源控制器（SRC）发送要求停止一个子系统、一组子系统或所有子系统的请求。**stopsrc** 命令向系统资源控制器发送子系统请求包，该请求包转发至子系统用于停止子服务器的请求。

如果没有 **-f**（强制停止）标志，则假设为正常的停止操作。正常停止要求子系统或子服务器完成所有当前处理，在所有应用程序活动完成时释放资源，然后结束。子系统不再接受任何新的工作请求。

强制停止要求子系统或子服务器迅速结束，释放所有资源，但不等待应用程序活动完成。

取消操作是在子系统的资源释放后以及一段宽延时间后才停止子系统。此宽延时间在子系统的对象类中指定。取消停止只用于子系统的停止，并且总是以 **SIGTERM** 信号发送给子系统。子系统抓取这个信号，执行子系统清除操作，然后结束。如果子系统没有在等待时间内结束（等待时间在子系统对象类中指定），则将 **SIGKILL** 信号发送给子系统以确保子系统的停止。

如果子系统使用套接字或消息队列进行通信，则构造一个数据包并发送给子系统。如果子系统使用信号进行通信，则将子系统对象类中的适当的信号发送给子系统。



## 标志

|                        |                                                                                                                                                                                                   |
|------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-a</b>              | 指定停止所有子系统。                                                                                                                                                                                        |
| <b>-c</b>              | 指定停止请求是已取消的停止请求。对于取消停止请求，将 <b>SIGTERM</b> 信号发送至子系统。在子系统对象类中指定的等待时间过去后，如果子系统仍未停止，则将 <b>SIGKILL</b> 信号发送至子系统。                                                                                       |
| <b>-f</b>              | 指定强制停止请求。                                                                                                                                                                                         |
| <b>-g Group</b>        | 指定停止一组子服务器。如果 <i>Group</i> 名称未包含在子系统对象类中，该命令将失败。                                                                                                                                                  |
| <b>-h Host</b>         | 指定请求该停止操作的外部 <i>Host</i> 机器。本地用户必须作为“root”用户运行。必须配置远程系统以接受系统资源控制器的请求。即 <b>srcmstr</b> 守护程序（请参阅 <b>/etc/inittab</b> ）必须用 <b>-r</b> 标志启动，并且必须配置 <b>/etc/hosts.equiv</b> 或 <b>.rhosts</b> 文件以允许远程请求。 |
| <b>-o Object</b>       | 指定子服务器 <i>Object</i> 值是作为字符串发送给子系统的。                                                                                                                                                              |
| <b>-p SubsystemPID</b> | 指定要停止的子系统的特定实例，或子服务器（停止子服务器请求要传递到该子系统的特定实例）。                                                                                                                                                      |
| <b>-P SubserverPID</b> | 指定子服务器 <i>PID</i> 是作为字符串传递给子系统的。                                                                                                                                                                  |
| <b>-s Subsystem</b>    | 指定要停止的子系统。 <i>Subsystem</i> 参数可以是实际的子系统名称或子系统的同义词名称。 <b>stopsrc</b> 命令停止所有当前活动的子系统实例。如果 <i>Subsystem</i> 名称没有包含在子系统对象类中，则该命令失败。                                                                   |
| <b>-t Type</b>         | 指定要停止的子服务器。如果子服务器对象类中未包含指定的 <i>Type</i> ， <b>stopsrc</b> 命令将失败。                                                                                                                                   |

## 示例

1. 要在外部主机上强制停止一个子系统，请输入：

```
stopsrc -h zork -s srctest -f
```

这将强制停止 *zork* 机器上的 *srctest* 子系统的所有实例。

2. 要取消停止一个子系统组，请输入：

```
stopsrc -g tcpip -c
```

这将在 *tcpip* 组的所有子系统上激活停止取消。

3. 要停止一个子服务器，请输入：

```
stopsrc -t tester -p 1234
```

这将停止 *tester* 子服务器，它属于子系统 *PID* 为 1234 的 *srctest* 子系统。

4. 要停止所有子系统，请输入：

```
stopsrc -a
```

这将停止本地机器上所有活动的子系统。

## 文件

|                                |                          |
|--------------------------------|--------------------------|
| <b>/etc/objrepos/SRCsubsys</b> | 指定 SRC 子系统配置对象类。         |
| <b>/etc/objrepos/SRCsubsvr</b> | 指定 SRC 子服务器配置对象类。        |
| <b>/etc/services</b>           | 定义用于因特网服务的套接字和协议。        |
| <b>/dev/SRC</b>                | 指定 <b>AF_UNIX</b> 套接字文件。 |
| <b>/dev/.SRC-unix</b>          | 指定临时套接字文件位置。             |

## 相关信息

**startsrc** 命令、**refresh** 命令。

《操作系统与设备管理》中的『系统资源控制器』提供了对子系统、子服务器和系统资源控制器的说明。

---

## stopvdsd 命令

### 用途

**stopvdsd** - 使虚拟共享磁盘不可用。

### 语法

**stopvdsd** {-a | vdsd\_name ...}

### 描述

**stopvdsd** 命令使指定的虚拟共享磁盘从暂挂状态变为停止状态。该命令使虚拟共享磁盘不可用。对虚拟共享磁盘具有未完成的请求的所有应用程序将看到这些请求以错误终止。读写请求通过将 **errno** 设置为 **ENODEV** 来返回错误。如果虚拟共享磁盘处于停止状态，则该命令将其保留在停止状态。

可以使用“系统管理界面工具”（SMIT）来运行该命令。要使用 SMIT，输入：

```
smit vdsd_mgmt
```

并选择 **Stop a Virtual Shared Disk** 选项。

在常规情况下，您不应发出该命令。“可恢复的”虚拟共享磁盘子系统使用该命令以受控制的方式来管理共享磁盘。如果您发出该命令，结果可能是不可预测的。

### 标志

**-a** 指定处于暂挂状态的所有虚拟共享磁盘都要停止。

### 参数

*vdsd\_name* 指定一个虚拟共享磁盘。如果虚拟共享磁盘未处于暂挂状态，则您将得到一条错误消息。

### 安全

您必须具有 **root** 用户权限以运行该命令。

### 退出状态

**0** 说明命令已成功完成。

非零 表明发生错误。

### 限制

您必须从对等域中联机的节点发出该命令。要使对等域联机，请使用 **startprdomain** 命令。要使现有对等域中一个特定的节点联机，请使用 **startprnode** 命令。有关创建和管理 RSCT 对等域的更多信息，请参阅 *RSCT: Administration Guide*。

在常规情况下，您不应发出该命令。“可恢复的”虚拟共享磁盘子系统使用该命令以受控制的方式来管理共享磁盘。如果您发出该命令，结果可能是不可预测的。

## 标准输出

当前 RVSD 子系统运行级别。

## 示例

要使虚拟共享磁盘 **vsd1vg1n1** 从暂挂状态变为停止状态，请输入：

```
stopvsd vsd1vg1n1
```

## 位置

```
/opt/rsct/vsd/bin/stopvsd
```

## 相关信息

命令：**cfgvsd**、**lsvsd**、**preparevsd**、**resumevsd**、**startvsd**、**suspendvsd**、**ucfgvsd**

---

## stpinet 方法

### 用途

禁用 inet 实例。

### 语法

```
stpinet [ -l "Interface ..." ] [ -t Time ]
```

### 描述

如果 **stpinet** 是以由 **-l** 选项指定的网络接口列表来启动的，则该方法只停止那些 IF。否则，**stpinet** 将使用 **wall** 命令通知用户关于 TCP/IP 的迫近终止，并调用 **ifconfig** 命令将每个已配置的 IF 标记为 **down**。如果未指定网络接口，则 inet 实例的状态标志设置为 DEFINED。

### 标志

**-l** "Interface ..."

指定禁用的接口名称。

**-t** Time

指定以分钟计数的时间，直到停止 inet 实例。

### 示例

以下示例在该方法执行 5 分钟后禁用 inet 实例 tr0:

```
stpinet -l "tr0" -t 5
```

### 相关信息

**ifconfig** 命令、**rmdev** 命令、**wall** 命令。

**odm\_run\_method** 子例程。

AIX 5L Version 5.3 Kernel Extensions and Device Support Programming Concepts 中的 Writing a Device Method。

《AIX 5L V5.3 通用编程概念：编写并调试程序》中的『程序员对象数据管理器（ODM）概述』。

《网络与通信管理》中的『TCP/IP 网络接口』。

---

## strace 命令

### 用途

显示 STREAMS 跟踪消息。

### 语法

**strace** [ *mid sid level* ] ...

### 描述

没有参数的 **strace** 命令将所有的驱动程序和模块中的所有 STREAMS 事件跟踪消息写到它的标准输出中。这些消息是从 STREAMS 日志驱动程序中获取的。如果提供参数，它们必须是在三元组中。每个三元组表明跟踪消息要从给定的模块或驱动程序、子标识（通常表明次要设备）以及优先级等于或小于给定级别的模块或驱动程序中接收。all 标记可由任何成员使用，以表明对该属性没有限制。

### 参数

*mid*        指定 STREAMS 模块的标识号。  
*sid*        指定子标识号。  
*level*      指定跟踪优先级别。

### 输出格式

每个跟踪消息输出的格式是：

<seq> <time> <ticks> <level> <flags> <mid> <sid> <text>

<seq>            跟踪序列号  
<time>           消息时间（格式为 *hh:mm:ss*）  
<ticks>          系统启动后，以机器滴答信号表示消息的时间  
<level>          跟踪优先级别  
<flags>          有以下值之一：

**E**            消息也在错误日志中  
**F**            表示致命错误  
**N**            邮件已发送给系统管理员

<mid>            源的模块标识号  
<sid>            源的子标识号  
<text>           跟踪消息的格式化文本

在多处理器系统上，<text> 由两部分组成：

- 消息所有者发送处的处理器号码，
- 格式化文本本身。

一旦启动，**strace** 命令将继续执行，直到用户终止该命令。

注：由于性能的考虑，所以一次只允许一个 **strace** 命令来打开 STREAMS 日志驱动程序。日志驱动程序有一个三元组的列表（该列表在命令调用中指定），并且程序会根据该列表比较每个潜在的跟踪消息，以决定是否要格式化和发送这个信息到 **strace** 进程中。因此，长的三元组列表会对 STREAMS 的总体性能的影响更大。运行 **strace** 命令对于某些模块和驱动程序（生成要发送给 **strace** 进程的跟踪消息的模块和驱动程序）的定时的影响最大。如果跟踪消息生成过快，以至 **strace** 进程无法处理，那么就会丢失一些消息。最后的情况可以通过检查跟踪消息输出上的序列号来确定。

## 示例

1. 要输出模块标识为 41 的模块或驱动程序中的所有跟踪消息，请输入：

```
strace 41 all all
```

2. 要输出模块标识为 41，子标识为 0、1 或 2 的模块或驱动程序中的跟踪消息：

```
strace 41 0 1 41 1 1 41 2 0
```

子标识为 0 和 1 的模块或驱动程序中的消息必须具有小于或等于 1 的跟踪级别。子标识为 2 的模块或驱动程序中的消息必须具有跟踪级别 0。

## 相关信息

*AIX 5L Version 5.3 Communications Programming Concepts* 中的 List of Streams Commands、STREAMS Overview 和 Understanding the log Device Driver。

---

## strchg 命令

### 用途

更改流配置。

### 语法

要推动模块到流上：

```
strchg -h Module1 [ , Module2 ... ]
```

要将模块从流上弹出：

```
strchg -p [ -a | -u Module ]
```

要推动和弹出模块以符合配置文件：

```
strchg -f File
```

### 描述

**strchg** 命令是用来改变与用户的标准输入关联的流配置。**strchg** 命令推动模块到流上，将模块从流上弹出，或者两者都进行。只有 root 用户或 STREAMS 设备的所有者可以更改该流的配置。如果另一位用户试图更改该配置，**strchg** 命令将失败。

注：如果模块是以错误的顺序推动，则流可能不会按预期运行。

### 标志

**-a** 将顶端驱动程序上的所有模块从流上弹出。**-p** 标志必须在 **-a** 标志前使用。

- f** *File*                    推动和弹出必要的模块以使流符合指定文件中给出的配置。
  
- h** *Module1*                将模块推动到流上。模块按照被推动的顺序排列在命令行中。
- p**                            将模块从流上弹出。单独使用时，**-p** 标志将最顶端的模块从流上弹出。
- u** *Module*                将指定模块上的所有模块从流上弹出。**-p** 标志必须在 **-u** 标志前使用。
  
- a** 和 **-u** 标志是互斥的。

## 参数

- Module1*                    指定要推动到流上的模块。（由 **-h** 标志使用。）
- Module*                    指定保留在流上的顶端模块。将该模块上的所有模块从流上弹出。（由 **-u** 标志使用。）
- File*                        包含一个模块列表，它表示期望的流配置。每个模块名必须出现在单独的行上，其中第一个模块名表示最顶端的模块，最后一个模块名表示离驱动程序最近的模块。

## 返回值

成功完成后，**strchg** 命令返回 0 值。否则，返回一个非 0 值并显示错误消息表示使用情况错误、模块名错误、要推动的模块太多、流上的 **ioctl** 操作故障或未能打开 *File* 参数指定的文件。

## 示例

1. 要推动流上的 **ldterm** 模块，请输入：

```
strchg -h ldterm
```

2. 要将顶端模块从与 **/dev/term/24** 设备关联的流上弹出，请输入：

```
strchg -p < /dev/term/24
```

用户必须是该设备的所有者或 **root** 用户。

3. 如果 **fileconf** 文件包含以下几项：

```
compat
ldterm
ptem
```

则以下命令会配置流，以便将 **ptem** 模块推到驱动程序上，**ldterm** 模块跟随在该模块的后面，而 **compat** 模块被推到最靠近流头。

```
strchg -f fileconf
```

## 相关信息

**strconf** 命令。

*AIX 5L Version 5.3 Communications Programming Concepts* 中的 List of Streams Commands 和 STREAMS Overview。

*AIX 5L Version 5.3 Technical Reference: Communications Volume 2* 中的 **streamio** 操作。

---

## strclean 命令

### 用途

清除 STREAMS 错误记录器。

### 语法

```
strclean [ -d ] [ -a Age ]
```

### 描述

**strclean** 命令是用来定期清除 STREAMS 错误记录器目录的：例如，通过使用 **cron** 守护程序。缺省情况下，除去所有 **/var/adm/streams** 目录中的与 **error.\*** 名称匹配、并且最近三天未修改的文件。

注：**strclean** 命令通常使用 **cron** 守护程序来每天或每星期运行一次。

### 标志

**-a Age** 指定记录文件的最长寿命（以天数计算）。  
**-d** 指定一个不同于缺省目录的目录。

### 示例

以下示例与运行不带参数的 **strclean** 命令具有相同的结果。

```
strclean -d /var/adm/streams -a 3
```

### 文件

**/var/adm/streams/error.\*** 包含 STREAMS 错误日志。

### 相关信息

**cron** 守护程序、**strerr** 守护程序。

*AIX 5L Version 5.3 Communications Programming Concepts* 中的 List of Streams Commands 和 STREAMS Overview。

---

## strconf 命令

### 用途

查询流配置。

### 语法

```
strconf [ -t | -m module ]
```

## 描述

**strconf** 命令用来查询流的配置。当不使用任何标志时，它显示该流中的所有模块的列表以及顶端驱动程序。该列表每行显示一个名称，其中显示的第一个名称是流上的最顶端模块，显示的最后一项是该驱动程序的名称。

注：**strconf** 命令只从标准输入中读取。

## 标志

**-m** *Module* 确定指定的模块是否在流上。如果该模块在流上，则 **strconf** 命令显示 **yes** 消息并返回 0 值。如果该模块不在流上，则 **strconf** 命令显示 **no** 消息并返回非零值。

**-t** 和 **-m** 标志是互斥的。

**-t** 只显示该流的最顶端模块（如果存在）。

## 参数

*Module* 指定要查看的模块。

## 示例

1. 对于只有 **ldterm** 模块在 **ports** 驱动程序上推动的流，**strconf** 命令（无任何标志）将产生以下输出：

```
ldterm ports
```

2. 输入以下命令来询问 **ldterm** 模块是否在该流上：

```
strconf -m ldterm
```

该命令返回退出状态 0，同时产生以下输出：

```
yes
```

## 相关信息

**strchg** 命令。

*AIX 5L Version 5.3 Technical Reference: Communications Volume 1* 中的 **streamio** 操作。

*AIX 5L Version 5.3 Communications Programming Concepts* 中的 List of Streams Commands 和 STREAMS Overview。

---

## strerr 守护程序

### 用途

从 STREAMS 日志驱动程序接收错误日志消息。

### 语法

**strerr**



## 描述

**strerr** 守护程序从 STREAMS 日志驱动程序接收错误日志消息，并将它们附加到一个日志文件中。产生的错误日志文件驻留在 `/var/adm/streams` 目录中，并命名为 `error.mm-dd`，其中 `mm` 是包含在每个日志文件中的信息的月份，`dd` 则是信息的日期。

错误日志消息的格式是：

```
<seq> <time> <ticks> <flags> <mid> <sud> <text>
```

这些字段的定义如下：

|                            |                                                                                          |
|----------------------------|------------------------------------------------------------------------------------------|
| <code>&lt;seq&gt;</code>   | 错误序列号                                                                                    |
| <code>&lt;time&gt;</code>  | 消息时间（格式为 <code>hh:mm:ss</code> ）                                                         |
| <code>&lt;ticks&gt;</code> | 引导优先级别之后的消息时间（以机器滴答信号计算）                                                                 |
| <code>&lt;flags&gt;</code> | 有以下值之一：<br><br><b>T</b> 消息也被发送到跟踪进程中<br><br><b>F</b> 表示致命错误<br><br><b>N</b> 发送邮件至您的系统管理员 |
| <code>&lt;mid&gt;</code>   | 源的模块标识号                                                                                  |
| <code>&lt;sid&gt;</code>   | 源的子标识号                                                                                   |
| <code>&lt;text&gt;</code>  | 错误消息的格式化文本                                                                               |

在多处理器系统上，`<text>` 由两部分组成：

- 消息所有者发送处的处理器号码，
- 格式化文本本身。

错误日志中出现的消息是为了报告异常情况，需要您的系统管理员注意。标明 STREAMS 驱动程序或模块的完全失败的消息应设置 **F** 标志。需要管理员立即注意的消息应设置 **N** 标志，以便使错误记录器通过 `mail` 命令发送消息给管理员。优先级别在错误日志中通常没有含义，但如果消息也发送给跟踪进程，则该级别就有了含义。

一旦启动，**strerr** 守护程序将继续执行，直到用户终止该守护程序。通常，**strerr** 守护程序是异步执行的。

**注：**每次只允许一个 **strerr** 守护程序打开 STREAMS 日志驱动程序。如果一个模块或驱动程序生成大量错误消息，则运行错误记录器会导致 STREAMS 性能退化。如果大量消息在短时间内生成，则日志驱动程序可能无法传递某些消息。这种情况会通过日志文件中消息序列号的间距显示。

## 文件

`/var/adm/streams/error.mm-dd`

错误日志文件。

## 相关信息

*AIX 5L Version 5.3 Communications Programming Concepts* 中的 List of Streams Commands、STREAMS Overview 和 Understanding the log Device Driver。

---

## strinfo 命令

### 用途

显示关于 STREAMS 活动的管理信息。

### 语法

**strinfo -m | -q**

### 描述

**strinfo** 命令显示关于 STREAMS、驱动程序和模块或流头和 STREAMS 运行队列的信息，以便用于调试。

### 标志

- m** 显示出现在 STREAMS 中的驱动程序和模块的信息。
- q** 显示活动的流头信息，以及保留 STREAMS 模块和驱动程序服务过程的运行队列的信息。

### 示例

1. 要显示正在使用的 STREAMS 驱动程序和模块的信息，请输入：

```
strinfo -m
```

这将产生与以下内容类似的清单：

```
Device: 'sad', dcookie 0xf, flags:0x4, str 0x19a69e8
Device: 'slog', dcookie 0x10, flags:0x4, str 0x19a6c18
Device: 'rs', dcookie 0x11, flags:0x2, str 0x19bcb00
Module: 'bufcall', flags:0x1, str 0x19a5c00
Module: 'ldterm', flags:0x0, str 0x19cc858
```

在这个示例中，dcookie 表示主号码，flags 表示标志配置，而 str 是 STREAMS 表地址。

2. 要显示关于活动的流头和 STREAMS 运行队列的信息，请输入：

```
strinfo -q
```

这将产生与以下内容类似的清单：

```
Active Stream Heads
sth      sth_dev  sth_rq   sth_wq   sth_flag rq->q_first
05a7ee00 00110001 05ad7000 05ad7074 00000818 00000000

STREAMS Service Queue
Queue 0x5ad7000 Flags 0x10
```

### 文件

**/usr/sbin/strinfo** 包含 **strinfo** 命令。

## 相关信息

AIX 5L Version 5.3 Communications Programming Concepts 中的 List of Streams Commands。

AIX 5L Version 5.3 Communications Programming Concepts 中的 STREAMS Overview。

---

## strings 命令

### 用途

在对象文件或二进制文件中查找可显示的字符串。

### 语法

```
strings [ -a ] [ - ] [ -o ] [ -t Format ] [ -n Number ] [ -Number ] [ File ... ]
```

### 描述

**strings** 命令在对象文件或二进制文件中查找可显示的字符串。字符串是 4 个或更多可显示字符的任意序列，以换行符或空字符结束。**strings** 命令对识别随机对象文件很有用。

### 标志

|                      |                                                                                                                                                                                                                          |
|----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-a</b> 或 <b>-</b> | 搜索整个文件，而不仅仅是数据段，以寻找可显示的字符串。如果省略这个标志，则 <b>strings</b> 命令只在对象文件的初始化数据空间内寻找。                                                                                                                                                |
| <b>-n Number</b>     | 指定最小的字符串长度（除了缺省的 4 个字符以外）。字符串长度的最大值是 4096。这个标志与 <b>-Number</b> 标志相同。                                                                                                                                                     |
| <b>-o</b>            | 列出文件中每个跟随在其八进制偏移量之后的字符串。这个标志与 <b>-t o</b> 标志相同。                                                                                                                                                                          |
| <b>-t Format</b>     | 列出从文件最开始起，每个跟随在其偏移量之后的字符串。该格式取决于用作 <i>Format</i> 变量的字符。<br><b>d</b> 以十进制写下偏移量。<br><b>o</b> 以八进制写下偏移量。<br><b>x</b> 以十六进制写下偏移量。<br><b>注：</b> 当 <b>-o</b> 和 <b>-t Format</b> 标志在一个命令行上多次定义，则最后指定的标志控制 <b>strings</b> 命令的行为。 |
| <b>-Number</b>       | 指定最小的字符串长度（除了缺省的 4 个字符以外）。字符串长度的最大值是 4096。这个标志与 <b>-n Number</b> 标志相同。                                                                                                                                                   |
| <i>File</i>          | 要搜索的二进制文件或对象文件。                                                                                                                                                                                                          |

### 退出状态

该命令返回以下退出值：

- 0** 表示命令成功运行。
- >0** 表示出错。

### 示例

- 要搜索文件，请输入：

```
strings strings
```

**strings** 命令显示：

```

@(#)56
1.17 com/cmd/scan/strings.c, cmdscan, bos320 5/7/92 10:21:20
标准输入
strings.cat
/usr/sbin/strings
Usage: strings [-a [-o] [-t format] [-n] [-#] [file...]]
Usage: strings [-a [-o] [-t format] [-n] [-#] [file...]]
Usage: strings [-a [-o] [-t format] [-n] [-#] [file...]]
Usage: strings [-a [-o] [-t format] [-n] [-#] [file...]]
Usage: strings [-a [-o] [-t format] [-n] [-#] [file...]]
%7o
%7d
%7x
%7o
%7d

```

2. 要搜索至少有 12 个字符的字符串，请输入：

```
strings -l2 strings
```

**strings** 命令显示：

```

1.17 com/cmd/scan/strings.c, cmdscan, bos320 5/7/92 10:21:20
标准输入
/usr/sbin/strings
Usage: strings [-a [-o] [-t format] [-n] [-#] [file...]]
Usage: strings [-a [-o] [-t format] [-n] [-#] [file...]]
Usage: strings [-a [-o] [-t format] [-n] [-#] [file...]]
Usage: strings [-a [-o] [-t format] [-n] [-#] [file...]]
Usage: strings [-a [-o] [-t format] [-n] [-#] [file...]]

```

3. 要搜索至少有 20 个字符的字符串，并以十六进制来显示偏移量，请输入：

```
strings -t x -n 20 strings
```

**strings** 命令显示：

```

1017 1.17 com/cmd/scan/strings.c, cmdscan, bos320 5/7/92 10:21:20
108c Usage: strings [-a [-o] [-t format] [-n] [-#] [file...]]
10d8 Usage: strings [-a [-o] [-t format] [-n] [-#] [file...]]
1124 Usage: strings [-a [-o] [-t format] [-n] [-#] [file...]]
1170 Usage: strings [-a [-o] [-t format] [-n] [-#] [file...]]
11bc Usage: strings [-a [-o] [-t format] [-n] [-#] [file...]]

```

## 相关信息

**od** 命令。

## strip 命令

### 用途

通过除去绑定程序和符号调试程序使用的信息，减少扩展公共对象文件格式（XCOFF）的对象文件的大小。

### 语法

```
strip [-V] [ -r [ -l ] | -x [ -l ] | -t | -H | -e | -E ] [ -X {32|64|32_64} ] [ - ] File ...
```

## 描述

**strip** 命令减少 XCOFF 对象文件的大小。**strip** 命令从 XCOFF 对象文件中有选择地除去行号信息、重定位信息、调试段、typchk 段、注释段、文件头以及所有或部分符号表。一旦您使用该命令，则很难调试文件的符号；因此，通常应该只在已经调试和测试过的生成模块上使用 **strip** 命令。使用 **strip** 命令减少对象文件所需的存储量开销。

对于每个对象模块，**strip** 命令除去给出的选项所指定的信息。对于每个归档文件，**strip** 命令从归档中除去全局符号表。

可以使用 **ar -s** 命令将除去的符号表恢复到归档文件或库文件中。

没有选项的 **strip** 命令除去行号信息、重定位信息、符号表、调试段、typchk 段和注释段。

## 标志

- e** 在对象文件的可选头中设置 **F\_LOADONLY** 标志。如果对象文件放置在归档中，则该标志指示绑定程序 (**ld** 命令)，在与此归档链接时应忽略该对象文件中的符号。
- E** 复位 (关闭) 对象文件的可选头中的 **F\_LOADONLY** 位。(请参阅 **-e** 标志。)
- H** 除去对象文件头、任何可选的头以及所有段的头部分。  
注: 不除去符号表信息。
- l** (小写 **L**) 从对象文件中除去行号信息。

- r** 除了外部符号和静态符号条目，将全部符号表信息除去。不除去重定位信息。同时除去调试段和 typchk 段。这个选项产生一个对象文件，该对象文件仍可以用作输入到链接编辑器 (**ld** 命令) 中。
- t** 除去大多数符号表信息，但并不除去函数符号或行号信息。
- V** 显示 **strip** 命令的版本号。
- x** 除去符号表信息，但并不除去静态或外部符号信息。**-x** 标志同时除去重定位信息，因此将不可能链接到该文件。
- X mode** 指定应检查 **strip** 的对象文件的类型。 *mode* 必须是以下各项中的一个:

- 32** 只处理 32 位对象文件
- 64** 只处理 64 位对象文件
- 32\_64** 既处理 32 位对象文件，又处理 64 位对象文件

缺省值是处理 32 位对象文件 (忽略 64 位对象文件)。也可以用 **OBJECT\_MODE** 环境变量设置 *mode*。例如，**OBJECT\_MODE=64** 使 **strip** 处理任何 64 位对象文件，并忽略 32 位对象文件。**-X** 标志覆盖 **OBJECT\_MODE** 变量。

- (双连字符) 将跟随在该标志后的所有参数解释为文件名。这就允许除去名称是以连字符开始的文件。

## 退出状态

该命令返回以下退出值:

- 0** 成功完成。
- >0** 发生错误。

## 示例

1. 要除去 **a.out** 文件中的符号表和行号信息，请输入:

```
strip a.out
```

2. 要除去 **a.out** 文件中的对象文件头，请输入：

```
strip -H a.out
```

3. 要除去 **lib.a** 中的 32 位和 64 位符号表，请输入：

```
strip -X 32_64 lib.a
```

## 文件

**/usr/ccs/bin/strip** 包含 **strip** 命令。

## 相关信息

**ar** 命令、**as** 命令、**dump** 命令、**ld** 命令、**size** 命令。

**ar** 文件、**a.out** 文件。

---

## stripnm 命令

### 用途

显示指定对象文件的符号信息。

### 语法

```
stripnm [ -x | -d ] [ -s ] [ -z ] File
```

### 描述

**stripnm** 命令（当不带 **-s** 标志运行时）将指定对象文件的符号表显示到标准输出中。由 *File* 参数指定的文件可以是单独的对象文件或对象文件的归档库。如果由 *File* 参数指定的文件是归档文件，将生成归档中每个对象文件的列表。如果符号表已从对象文件中除去，则 **stripnm** 命令从回溯表抽取符号名（即使未指定 **-s** 标志），并从对象文件的装入程序段抽取符号名。如果回溯表不存在，则显示错误消息。

每个符号名称前都有它的地址和代表符号类型的一个字符（类似于 **nm** 输出）。当和 **-z** 一起使用时，输出格式与 AIX 5.2 之前的输出格式相同，即每个符号名后跟着它的地址（如果地址未定义，则是一串空格）、类的类型和段类型。地址字段可以用十进制显示（使用 **-z** 或 **-d** 时的缺省值），或用十六进制显示（不使用 **-z** 或如果已使用 **-x** 标志时的缺省值）。

**stripnm** 命令也收集和报告源文件的名称。所有跟在源文件名之后的符号都属于该源文件，直到遇到下一个源文件名行。对于已除去的文件，源文件名将作为对象文件名报告。

当使用 **-s** 标志运行时，**stripnm** 命令将忽略符号表（如果存在），并且始终从对象文件的回溯表和装入程序段中抽取例程名称。

如果符号表不存在，或使用 **-s** 标志，则 **stripnm** 命令也将搜索 **glue** 代码和指针 **glue** 信息。两者都是该对象文件的文本区域中的指示信息序列。

对于 32 位应用程序，**glue** 代码由以下指示信息序列组成：

```
8182xxxx # lwz r12,xxxx(r12) (xxxx 是 TOC 条目索引)
90410014 # stw r2,14(r1)
800c0000 # lwz r0,0(r12)
```

```
804c0004 # lwx r2,4(r12)
7c0903a6 # mtctr r0
7c0903a6 # mtctr r0
4e800420 # bctr
```

装入程序段条目（其地址与 `xxxx` 指向的 TOC 条目相匹配）给出了此 `glue` 代码序列的函数名称。

对于 64 位可执行程序，`glue` 代码序列如下：

```
982xxxx # ld r12,xxxx(r2) (xxxx 是 TOC 条目索引)
8410028 # std r2,28(r1)
80c0000 # ld r0,0(r12)
84c0008 # ld r2,8(r12)
c0903a6 # mtctr r0
c0903a6 # mtctr r0
e800420 # bctr
```

对于 32 位应用程序，指针 `glue` 代码由以下序列组成：

```
800b0000 # lwx r0,0(r11)
90410014 # stw r2,20(r1)
7c0903a6 # mtctr r0
804b0004 # lwx r2,4(r11)
816b0008 # lwx r11,8(r11)
4e80xx20 # bctr
```

对于 64 位可执行程序，指针 `glue` 代码序列如下：

```
e80b0000 # ld r0,0(r11)
f8410028 # std r2,20(r1)
7c0903a6 # mtctr r0
e84b0008 # ld r2,8(r11)
e96b0010 # ld r11,16(r11)
4e80xx20 # bctr
```

指针 `glue` 只存在于一个副本中，并且始终以 `._prtg1` 符号报告。

`stripnm` 命令在文本区域从头到尾搜索这些序列。如果命令发现一个匹配的指示信息序列，它将被报告为 `glue` 代码或指针 `glue`。

**stripnm** 为 `glue` 代码和指针 `glue` 人工生成源文件符号。对于 32 位可执行程序，所有 `glue` 代码条目的源文件是 `glink.s`，而指针 `glue` 的源文件是 `ptrg1.s`。对于 64 位可执行程序，源文件分别是 `glink64.s` 和 `ptrg1_64.s`。

**stripnm** 命令也可用来在 `/unix` 文件中搜索符号信息。如果 `/unix` 文件不符合目前运行的内核，将显示警告消息。

## 标志

- d** 以十进制格式显示符号地址值。
- 这是使用 **-z** 时的缺省值。
- s** 强制忽略符号表。
- x** 以十六进制格式显示符号地址值。
- 这是没有使用 **-z** 时的缺省值。
- z** 使用旧格式。

## 示例

1. 要列出 **a.out** 对象文件的符号，请输入：

```
stripnm a.out
```

2. 要以十进制列出 **a.out** 对象文件的符号地址值，请输入：

```
stripnm -d a.out
```

3. 要以旧的格式列出 **libc.a** 中对象文件中的符号，但使用十六进制地址，请输入：

```
stripnm -xz libc.a
```

## 相关信息

**strip** 命令。

---

## strload 命令

### 用途

装入和配置可移植流环境（PSE）。

### 语法

```
strload [ -u | -q ] [ -f File ] [ -d List ] [ -m List ]
```

### 描述

**strload** 命令使系统管理员能够装入和卸装驱动程序和模块，并且查询 PSE 和它的从属的装入状态。

缺省情况下，**strload** 命令根据 **/etc/pse.conf** 文件装入 PSE。**-f** 标志允许管理员使用备用的配置文件。**-d** 和 **-m** 标志是用来指定不在配置文件中（比如正在开发新的驱动程序时）的驱动程序和模块。**-q** 标志报告引用的驱动程序和模块的系统装入状态（内核的存在）。

### 配置文件

配置文件是一个平面 ASCII、面向行的数据库。注释由 #（镑符号）引入，并继续到行尾。忽略空白行。每个记录的格式为：

```
attributes filename [argument [node [minor ...] ] ]
```

字段由空格、制表符或空格和制表符隔开。破折号（-）可以用来指定字段值，表示使用缺省值。字段定义如下：

|          |                                  |
|----------|----------------------------------|
| 属性       | 描述要装入的扩展名。可接受的值有：                |
| <b>d</b> | 指定驱动程序。                          |
| <b>m</b> | 指定模块。                            |
| <b>s</b> | 创建节点作为标准（非克隆）的设备。                |
| <b>+</b> | 指定可以多次配置扩展名。所有包含扩展文件名的行都必须指定这个值。 |

文件名 指定包含该扩展名的对象文件。如果发出的命令在要装入、卸装或查询的驱动程序或模块的文件名中有“/”（斜杠），则 **strload** 命令在文件名字段中明确地使用该值。如果文件名条目中没有“/”，则 **strload** 命令首先在当前目录中寻找该驱动程序或模块的副本。如果该驱动程序或模块不在当前目录中，**strload** 将在 **/usr/lib/drivers/pse** 目录中寻找该驱动程序或模块。



注: 建议从根目录 (/) 发出 **strload** 命令。进行装入、卸装和查询的 **strload** 命令应该始终从相同的目录发出。

内核扩展名装入程序“要求”在装入、卸装和查询时必须使用相同的路径名。这个要求，加上文件名是由 **strload** 确定，可能会带来问题。因为内核在寻找匹配时在文件名上执行 **strcmp()**，所以 **strload** 命令使用的路径名中的每个字节，必须与内核扩展名装入程序使用的路径名中的每个对应位置上的字节“严格”匹配。如果 **strload** 命令是从一个不同的目录发出以卸装该模块或驱动程序，则发生以下情况之一：

- 如果 **strload** 命令在新的当前目录中未发现该驱动程序或模块的副本，**strload** 将试图卸装 **/usr/lib/drivers/pse** 目录中的驱动程序或模块。然而，此路径名也许与装入程序为该驱动程序或模块所记录的路径名不同。如果路径名不同，**strload** 命令将失败。
- 如果 **strload** 命令在新的当前目录中发现该模块或驱动程序的另一个副本，那么路径名就是相同的，并且装入程序将正确卸装已装入的驱动程序或模块。这样，**strload** 命令会成功运行，但结果未必是用户所期望的。

例如:

以下方案（“不”推荐）将导致卸装“spx”（也称为“A”）。这可能并不是用户期望的效果。

```
mkdir /tmp/foo /tmp/bar
cp /usr/lib/drivers/pse/spx /tmp/foo/A
cp /bin/ls /tmp/bar/A
cd /tmp/foo
strload -d A      # The loader knows the path and filename as
                  # "A" because "A" is found in the current
                  # directory

cd /tmp/bar
strload -q -d A  # Reports "yes" because there is "A" in the
                  # current directory. Note that the file "A"
                  # in /tmp/bar is NOT the same file "A" in
                  # /tmp/foo, but the loader does not care
                  # because it identifies the file by
                  # pathname.
strload -u -d A  # Unloads spx (also known as "A")!
```

以下内容是一个错误方案:

```
mkdir /tmp/foo2 /tmp/bar2
cp /usr/lib/drivers/pse/spx /tmp/foo2/A
cd /tmp/foo2
strload -d A      # The loader knows the path and filename as
                  # "A" because "A" is found in the current
                  # directory.

cd /tmp/bar2
strload -q -d A  # Answers "no". There is no filename
                  # in /tmp/bar2 that matches "A", so strload
                  # prepends pathname "/usr/lib/drivers/pse" to
                  # "A". "/usr/lib/drivers/pse/A" is not found,
                  # so strload answers "no".
strload -u -d A  # Fails - "A" does not exist.
```

以下内容是一个错误方案:

```
cd /usr/lib/drivers/pse
strload -d spx   # The loader knows the path and filename as
                  # "spx" because "spx" is found in the
                  # current directory.

cd /
strload -q -d spx # Answers "no". There is no filename in /
                  # that matches "spx", so strload prepends
                  # the pathname "/usr/lib/drivers/pse" to
                  # "spx". "/usr/lib/drivers/pse/spx" is found
```

```

# since it exists, so strload gives
# "/usr/lib/drivers/pse/spx" to the loader.
# The strcmp() fails since
# "/usr/lib/drivers/pse/spx" and "spx" do
# not match exactly.
strload -u -d spx # Fails - "spx" does not exist.

```

- 参数** 对于 **strload** 命令，该参数没有含义。这个字段是可选的。在调用它的配置例程时，将它传递到扩展名。其解释是特定于该扩展名的。缺省参数是 `filename` 字段中的值。
- 节点** 指定要创建的节点名称。这个字段是可选的。它只适用于驱动程序，并且只在装入驱动程序时作为创建的节点名使用。缺省情况下，创建的节点是 `/dev/filename`。
- 次要** 指定为此驱动程序创建的附加的、非克隆的节点。这个字段是可选的。通过将次要数字附加到克隆的驱动程序节点名上，创建该节点名。给出的次要数字不超过 5（从 0 到 4），并且对每个数字都创建一个节点。

如果命令行中明确地给出了配置文件的名称，则 **-d** 和 **-m** 标志不会使配置文件被忽略，情况如下：

```
strload -f /tmp/my.conf -d newdriver
```

**注：****-d** 和 **-m** 标志不会覆盖配置文件。即如果使用该配置文件装入驱动程序 **dgb**，**-d** 标志将试图重新装入它，但会失败。在 **-d** 和 **-m** 标志前处理该配置文件。

对 **-d** 和 **-m** 标志，*List* 变量是以逗号分隔的文件名的列表，其中每个文件名包含一个单独的 PSE 驱动程序或模块。就像在该配置文件中找到以下表单之一的某行，进行配置处理：

```

d filename
m filename

```

## 标志

- d** *List* 列出要装入或卸装的 PSE 设备驱动程序。*List* 变量指定一个以逗号分隔的驱动程序对象名的列表。
- f** *File* 根据 *File* 变量显示的文件中所包含的配置信息，配置 PSE。缺省的配置文件是 `/etc/pse.conf`。
- m** *List* 列出要装入或卸装的 PSE 模块。*List* 变量指定一个以逗号分隔的模块对象名的列表。
- q** 报告扩展名的装入状态。
- u** 卸装扩展名。

## 示例

1. 输入以下命令来装入 PSE（如果还未装入）、`/usr/lib/drivers/pse/` 目录中的 `dgb` 和 `ssb` 驱动程序以及当前目录中的 `aoot` 模块，但不使用配置文件：

```
root# strload -d dgb,ssb -m ./aoot
```

2. 要只卸装 `aoot` 模块，请输入：

```
root# strload -u -m ./aoot
```

3. 输入以下命令询问 `spx` 驱动程序是否存在：

```
root# strload -q -d
spx
```

如果不存在，将产生以下输出：

```
spx
: no
```

4. 以下内容是一个示例配置文件：

```

#example configuration file
d      dgb                               #line 1
d      mux      -      -      0         #line 2
ds     foo                               #line 3
d+     xtiso    tcp     /dev/xti/tcp     #line 4
d+     xtiso    udp     /dev/xti/udp     #line 5
m      aoot                               #line 6

```

行 1 装入 dgb 驱动程序扩展名，作为名为 /dev/dgb 的克隆设备。传递到 dgb 配置例程的参数是 dgb。

行 2 装入 mux 驱动程序扩展名，作为名为 /dev/mux 的克隆设备，同时创建一个带有次要数字 0（零）的标准设备名 /dev/mux0。（用次要数字 0 到 4 创建的设备名不超过 5 个。）

行 3 装入 foo 驱动程序扩展名，作为一个名为 /dev/foo 的标准设备（非克隆）。次要数字是 0。

行 4 和行 5 装入 xtiso 驱动程序扩展名，并且将其配置两次：一次作为 tcp，而另一次作为 udp。创建的克隆节点是 /dev/xti/tcp 和 /dev/xti/udp。调用 xtiso 的配置例程两次：一次使用参数 tcp，而另一次使用参数 udp。

行 6 装入 aoot 模块扩展名。不创建节点，并且将值 aoot 传递给该配置例程。

## 文件

|                               |                       |
|-------------------------------|-----------------------|
| <b>/usr/lib/drivers/pse/*</b> | 包含 PSE 内核扩展名。         |
| <b>/etc/pse.conf</b>          | 缺省的 PSE 配置文件。         |
| <b>/usr/sbin/strload</b>      | 包含 <b>strload</b> 命令。 |

## 相关信息

**slibclean** 命令、**strerr** 命令。

*AIX 5L Version 5.3 Communications Programming Concepts* 中的 Configuring Drivers and Modules in the Portable Streams Environment (PSE)、List of Streams Commands 和 STREAMS Overview。

---

## strreset 命令

### 用途

复位一个流。

### 语法

```
strreset [ -M Major ] [ -m Minor ]
```

### 描述

**strreset** 命令通过在流头生成 M\_FLUSH 信息来复位开放的流。它主要是用来复位阻塞的流。当无法重新打开流时，发出 I\_FLUSH ioctl() 或等价命令。休眠在模块关闭例程中的进程可能会发生这种情况，这时信号无法发送到该进程（例如，正在退出的 zombie 进程）。

### 标志

**-M** *Major*                    指定与要复位的流相关联的特殊文件的主号码。

**-m** *Minor* 指定与要复位的流相关联的特殊文件的次要号码。

## 退出状态

该命令返回以下退出值:

**0** 成功完成。  
**>0** 发生错误。

## 安全性

访问控制: 您必须具有 `root` 用户权限以运行该命令。

审计事件: 不适用

## 文件

`/usr/sbin/streset` 包含 `streset` 命令。

---

## strtune 命令

### 用途

该命令具有几个相关功能:

- 获取或设置流可调参数。
- 使用组件跟踪定义对象跟踪。
- 列出流模块的可调值。
- 列出活动队列的可调值。

### 语法

**strtune** {**-n** *name* | **-q** *addr*} **-o** *tunable\_name*[=*value*] **-o** *tunable\_name*[=*value*] ...

**strtune** [**-n** *name* | **-q** *addr* [**-a**]] **-o** *trclevel*[=*value*]

**strtune** [**-M** ]

**strtune** [**-Q** ]

**strtune** [**-f** *tunefile* ]

### 描述

使用该命令显示或列出值时对该命令的使用没有限制, 但是在使用该命令修改可调值或定义对象跟踪时, 您必须具有 `root` 用户权限。

### 标志

**-n** *name* 定义流模块名称或设备名。

|                               |                                                                                                                                                                                                                                            |
|-------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-q</b> <i>addr</i>         | 定义活动队列地址。<br><br>如果命令设置可调参数，则它根据队列的同步级别修改队列对或唯一的队列。如果同步级别不是 <code>SQLVL_QUEUE</code> ，则同步级别还将传播到所有队列对。                                                                                                                                     |
| <b>-o</b> <i>tunable_name</i> | 定义可调参数的名称。可能的值有： <ul style="list-style-type: none"> <li>• <b>hiwat</b>，定义队列中流量控制的高水位标记。</li> <li>• <b>lowat</b>，定义队列中流量控制的低水位标记。</li> <li>• <b>minpsz</b>，定义最小包大小。</li> <li>• <b>maxpsz</b>，定义最大包大小。值 <code>-1</code> 表示不限制包大小。</li> </ul> |
| <i>value</i>                  | <b>strttune</b> 命令可以通过几次列出 <b>-o</b> 选项来初始化几个可调参数。如果没有给定新值，则命令显示可调参数的值。只有具有 <code>root</code> 权限的用户才能修改可调参数值。                                                                                                                              |
| <b>-n</b> <i>name</i>         | 定义流模块名称。如果命令中没有显示 <b>-n</b> 或 <b>-q</b> 标志，则命令将显示或修改包含 <code>pse</code> 全局跟踪级别 ( <code>pse_trclevel</code> ) 的全局变量。                                                                                                                        |
| <b>-q</b> <i>addr</i>         | 定义活动队列地址。如果命令中没有显示 <b>-n</b> 或 <b>-q</b> 标志，则命令将显示或修改包含 <code>pse</code> 全局跟踪级别 ( <code>pse_trclevel</code> ) 的全局变量。<br><br>如果命令设置跟踪级别，则它会根据队列的同步级别来修改队列对或唯一的队列。如果同步级别不是 <code>SQLVL_QUEUE</code> ，则同步级别还将传播到所有队列对。                        |
| <b>-o</b> <i>trclevel</i>     | 显示或修改跟踪级别。 <b>-o</b> 标志不能多次列出。                                                                                                                                                                                                             |
| <i>value</i>                  | 如果没有给定新值，则命令显示可调参数的值。只有具有 <code>root</code> 权限的用户才能修改可调参数值。                                                                                                                                                                                |
| <b>-a</b>                     | 使用该标志强制 <b>strttune</b> 命令将新值传播到流中的所有队列（从流头部到驱动程序）。如果同步级别不是 <code>SQLVL_QUEUE</code> ，则同步级别还将传播到所有队列对。                                                                                                                                     |
| <b>-M</b>                     | 显示每个模块的 <code>name</code> 、 <code>idname</code> 和关联的可调参数 ( <b>minpsz</b> 、 <b>maxpsz</b> 、 <b>lowat</b> 、 <b>hiwat</b> 和 <b>trclevel</b> ) 的值。                                                                                             |
| <b>-Q</b>                     | 显示每个活动队列的 <code>name</code> 、 <code>idname</code> 和关联的可调参数 ( <b>minpsz</b> 、 <b>maxpsz</b> 、 <b>lowat</b> 、 <b>hiwat</b> 和 <b>trclevel</b> ) 的值。                                                                                           |
| <b>-f</b> <i>tunefile</i>     | <i>tunefile</i> 变量将文件路径保存在包含可调参数设置的文件中。 <i>tunefile</i> 文件的每一行作为一条命令管理；如果 <i>tunefile</i> 中有任何修改命令，则用户必须具有 <code>root</code> 权限才能实现这些修改。                                                                                                   |

## 退出状态

|    |       |
|----|-------|
| 0  | 成功完成。 |
| >0 | 发生错误。 |

## 示例

1. 要显示 **ldterm** 模块的 **hiwat** 可调参数值：

```
strttune -n ldterm -o hiwat
```

2. 在具有 `root` 权限的情况下，要将 **ldterm** 模块的 **hiwat** 值设置为 8192，请输入：

```
strttune -n ldterm -o hiwat=8192
```

3. 要运行以下行：

```
-n udp6 -o lowat=256  
-n dlpi -o hiwat=4096 -o lowat=128 -o minpsz=128
```

在 `/tmp/ff` 文件中列出：

```
strttune -f /tmp/ff
```

这将导致运行以下命令:

```
strtune -n udp6 -o lowat=256
strtune -n dlpi -o hiwat=4096 -o lowat=128 -o minpsz=128
```

## 文件

`src/bos/usr/sbin/strtune/strtune.c` 包含 `strtune` 命令。

## 相关信息

---

## struct 命令

### 用途

将 FORTRAN 程序转换成 RATFOR 程序。

### 语法

```
struct [ -s ] [ -i ] [ -a ] [ -b ] [ -n ] [ -tNumber ] [ -cNumber ] [ -eNumber ] [ File ]
```

### 描述

`struct` 命令将 *File* (标准输入缺省值) 指定的 FORTRAN 程序转换为 RATFOR 程序。在任何可能之处, RATFOR 控制构造替换原始的 FORTRAN。语句号仅出现在仍为必要之处。进行装饰性的更改, 包括将 Hollerith 字符串更改为引用的字符串, 以及将关系运算符更改为符号 (例如, 将 `.GT.` 更改为 `>`)。输出被适当缩格。

`struct` 命令知道 FORTRAN 66 语法, 但不知道全部的 FORTRAN 77 语法。如果输入的 FORTRAN 程序包含 RATFOR 中保留字的标识符, 则该程序的结构版本将不是有效的 RATFOR 程序。生成的标号不能超过 32767。如果您遇到一个没有目标的 `goto` 语句, 请尝试使用 `-e` 标志。

### 标志

`-a` 将 `else-if` 语句的序列变为非 RATFOR `switch` 的格式:

```
switch
{
  case pred1: code
  case pred2: code
  case pred3: code
  default: code
}
```

按顺序测试 `case` 谓词。只执行一个与 `case` 对应的代码。RATFOR 中不出现此通用的 `switch` 语句格式。

`-b` 生成 `goto` 语句, 而不是多级别的 `break` 语句。

`-c Number` 通过非零整数 *Number* 来递增输出程序中的连续标号。缺省值是 1。请不要在 `-c` 和 *Number* 之间插入空格。

`-e Number` 如果 *Number* 为 0 (缺省值), 请仅在循环中放入能导致循环迭代的代码。请不要在 `-e` 和 *Number* 之间插入空格。

`-i` 请不要将计算过的 `goto` 语句变为 `switch` 语句。(RATFOR 不会将 `switch` 语句变回计算过的 `goto` 语句。)

`-n` 生成 `goto` 语句, 而不是多级别的 `next` 语句。

`-s` 接受标准格式的输入。注释由列 1 中的 `c`、`C` 或 `*` 指定, 而连续行由列 6 中的非零、非空格字符指定。输入的格式是 `f77` 命令可接受的格式。

**-t** *Number* 将非零整数 *Number* 作为输出程序中的最低值标号。缺省值是 10。请不要在 **-t** 和 *Number* 之间插入空格。

如果 *Number* 非零，则将小代码段引进循环，否则该循环会退出到几处地方（包括该段），并且该段只能从循环处到达。在这种情况下，“小”接近但不等于代码段中的语句数目。建议使用小于 10 的 *Number* 值。

## 示例

要将 test.f FORTRAN 程序转换为 newtest.ratfor RATFOR 程序，请输入：

```
struct -s -i -n -t2 test.f > newtest.ratfor
```

## 文件

|                           |                              |
|---------------------------|------------------------------|
| /tmp/struct*              | 处理 <b>struct</b> 命令时使用的临时文件。 |
| /usr/lib/struct/structure | 处理 <b>struct</b> 命令进程的文件。    |
| /usr/lib/struct/beautify  | 处理 <b>struct</b> 命令进程的文件。    |
| /usr/ucb/struct           | 包含 <b>struct</b> 命令。         |

## 相关信息

**asa** 或 **fpr** 命令、**fsplit** 命令。

《操作系统与设备管理》中的『命令概述』。

---

## sttinet 方法

### 用途

启用 inet 实例。

### 语法

```
sttinet [ -l Interface ... ]
```

### 描述

**sttinet** 方法调用 **ifconfig** 命令以及将 inet 实例的状态标志设置为 AVAILABLE（可用的），从而启用 inet 实例。

注：**sttinet** 方法是编程的工具，不应在命令行中执行。

### 标志

**-l** *Interface* ... 指定要启用的特定接口。如果未指定任何接口，则启动所有配置的接口。

### 示例

用以下方法启用 inet 实例：

```
sttinet -l tr0 -l tr1
```

## 相关信息

**ifconfig** 命令、**mkdev** 命令。

**odm\_run\_method** 子例程。

*AIX 5L Version 5.3 Kernel Extensions and Device Support Programming Concepts* 中的 Writing a Device Method。

《*AIX 5L V5.3 通用编程概念: 编写并调试程序*》中的『程序员对象数据管理器 (ODM) 概述』。

《*网络与通信管理*》中的『TCP/IP 网络接口』。

---

## stty 命令

### 用途

设置、复位和报告工作站操作参数。

### 语法

```
stty [ -a ] [ -g ] [ Options ]
```

### 描述

**stty** 命令对当前为标准输入的设备设置某些 I/O 选项。该命令将输出写到当前为标准输出的设备中。

这个操作系统的版本使用标准 X/Open Portability Guide Issue 4 接口来控制终端，保持与 POSIX 和 BSD 接口的兼容性。**stty** 命令支持 POSIX 和 BSD 的适应选项，但强烈推荐使用 POSIX 选项。同样提供废弃的 BSD 选项列表（带有对应的 POSIX 选项）。

输入以下命令时，可以将 tty 设备的标准输入重定向：

```
stty -a </dev/ttyx
```

**stty** 命令 (POSIX) 将挂起并等待该 tty 的 **open()**，直到确定 RS-232 载波检测信号。如果设置了 **local** 或 **forcedcd** (仅对 128 端口) 选项，这个规则将不适用。

### 标志

- a** 将所有选项设置的当前状态写到标准输出中。
- g** 将选项设置写到标准输出中，其格式可以由另一个 **stty** 命令使用。

### 选项

**stty** 命令支持以下选项类别：

- 控制模式
- 输入方式
- 输出方式
- 本地模式
- 硬件流量控制模式
- 控制字符指定



- 组合模式
- 窗口大小

#### 控制模式

|                           |                                                                                                                                                                                                                                                     |
|---------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>cllocal</b>            | 假定一行没有调制解调器控制。                                                                                                                                                                                                                                      |
| <b>-cllocal</b>           | 假定一行带有调制解调器控制。                                                                                                                                                                                                                                      |
| <b>cread</b>              | 启用接收器。                                                                                                                                                                                                                                              |
| <b>-cread</b>             | 禁用接收器。                                                                                                                                                                                                                                              |
| <b>cstopb</b>             | 每个字符选择两个停止位。                                                                                                                                                                                                                                        |
| <b>-cstopb</b>            | 每个字符选择一个停止位。                                                                                                                                                                                                                                        |
| <b>cs5, cs6, cs7, cs8</b> | 选择字符大小。                                                                                                                                                                                                                                             |
| <b>hup, hupcl</b>         | 最后关闭时挂起拨号连接。                                                                                                                                                                                                                                        |
| <b>-hup, -hupcl</b>       | 最后关闭时不挂起拨号连接。                                                                                                                                                                                                                                       |
| <b>parenb</b>             | 启用奇偶性校验的生成和检测。                                                                                                                                                                                                                                      |
| <b>-parenb</b>            | 禁用奇偶性校验的生成和检测。                                                                                                                                                                                                                                      |
| <b>parodd</b>             | 选择奇校验。                                                                                                                                                                                                                                              |
| <b>-parodd</b>            | 选择偶校验。                                                                                                                                                                                                                                              |
| <b>0</b>                  | 立即挂起电话线路。                                                                                                                                                                                                                                           |
| <b>speed</b>              | 将工作站输入和输出速度设置为指定的 <i>speed</i> 数（以位/秒为单位）。并不是所有的硬件接口都支持所有的速度。 <i>speed</i> 的可能值有: <b>50、75、110、134、200、300、600、1200、1800、2400、4800、9600、19200、19.2、38400、38.4、exta</b> 和 <b>extb</b> 。<br><b>注: exta、19200 和 19.2 是同义词; extb、38400 和 38.4 是同义词。</b> |
| <b>ispeed speed</b>       | 将工作站输入速度设置为指定的 <i>speed</i> 数（以位/秒为单位）。并不是所有的硬件接口都支持所有的速度，而且并不是所有的硬件接口都支持该选项。 <i>speed</i> 的可能值与 <i>speed</i> 选项相同。                                                                                                                                 |
| <b>ospeed speed</b>       | 将工作站输出速度设置为指定的 <i>speed</i> 数（以位/秒为单位）。并不是所有的硬件接口都支持所有的速度，而且并不是所有的硬件接口都支持该选项。 <i>speed</i> 的可能值与 <i>speed</i> 选项相同。                                                                                                                                 |

#### 输入模式

|                |                                 |
|----------------|---------------------------------|
| <b>brkint</b>  | 中断时发出 INTR 信号。                  |
| <b>-brkint</b> | 中断时不发出 INTR 信号。                 |
| <b>icrnl</b>   | 输入时将 CR 映射为 NL。                 |
| <b>-icrnl</b>  | 输入时不将 CR 映射为 NL。                |
| <b>ignbrk</b>  | 输入时忽略 BREAK。                    |
| <b>-ignbrk</b> | 输入时不忽略 BREAK。                   |
| <b>igncr</b>   | 输入时忽略 CR。                       |
| <b>-igncr</b>  | 输入时不忽略 CR。                      |
| <b>ignpar</b>  | 忽略奇偶错误。                         |
| <b>-ignpar</b> | 不忽略奇偶错误。                        |
| <b>inlcr</b>   | 输入时将 NL 映射为 CR。                 |
| <b>-inlcr</b>  | 输入时不将 NL 映射为 CR。                |
| <b>inpck</b>   | 启用奇偶校验。                         |
| <b>-inpck</b>  | 禁用奇偶校验。                         |
| <b>istrip</b>  | 将输入字符剥离到 7 位。                   |
| <b>-istrip</b> | 不将输入字符剥离到 7 位。                  |
| <b>iuclc</b>   | 将大写字母字符映射为小写。                   |
| <b>-iuclc</b>  | 不将大写字母字符映射为小写。                  |
| <b>ixany</b>   | 允许任何字符重新启动输出。                   |
| <b>-ixany</b>  | 只允许 START (Ctrl-Q 按键顺序) 重新启动输出。 |
| <b>ixoff</b>   | 当输入队列接近空或满时，发送 START/STOP 字符。   |
| <b>-ixoff</b>  | 不发送 START/STOP 字符。              |

|                           |                                                                                                       |
|---------------------------|-------------------------------------------------------------------------------------------------------|
| <b>ixon</b>               | 启用 START/STOP 输出控制。一旦启用 START/STOP 输出控制，您可以按下 Ctrl-S 按键顺序暂停向工作站的输出，也可按下 Ctrl-Q 按键顺序恢复输出。              |
| <b>-ixon</b>              | 禁用 START/STOP 输出控制。                                                                                   |
| <b>imaxbel</b>            | 当输入溢出时，回送 BEL 字符并且废弃最后的输入字符。                                                                          |
| <b>-imaxbel</b>           | 当输入溢出时，废弃所有输入。                                                                                        |
| <b>parmrk</b>             | 标记奇偶错误。                                                                                               |
| <b>-parmrk</b>            | 不标记奇偶错误。                                                                                              |
| <br>                      |                                                                                                       |
| 输出方式                      |                                                                                                       |
| <b>bs0, bs1</b>           | 为退格符选择延迟样式 ( <b>bs0</b> 表示没有延迟)。                                                                      |
| <b>cr0, cr1, cr2, cr3</b> | 为 CR 字符选择延迟样式 ( <b>cr0</b> 表示没有延迟)。                                                                   |
| <b>ff0, ff1</b>           | 为换页选择延迟样式 ( <b>ff0</b> 表示没有延迟)。                                                                       |
| <b>nl0, nl1</b>           | 为 NL 字符选择延迟样式 ( <b>nl0</b> 表示没有延迟)。                                                                   |
| <b>ofill</b>              | 使用延迟填充字符。                                                                                             |
| <b>-ofill</b>             | 使用延迟定时。                                                                                               |
| <b>ocrnl</b>              | 将 CR 字符映射为 NL 字符。                                                                                     |
| <b>-ocrnl</b>             | 不将 CR 字符映射为 NL 字符。                                                                                    |
| <b>olcuc</b>              | 输出时将小写字母字符映射为大写。                                                                                      |
| <b>-olcuc</b>             | 输出时不将小写字母字符映射为大写。                                                                                     |
| <b>onlcr</b>              | 将 NL 字符映射为 CR-NL 字符。                                                                                  |
| <b>-onlcr</b>             | 不将 NL 字符映射为 CR-NL 字符。                                                                                 |
| <b>onlret</b>             | 在终端 NL 执行 CR 功能。                                                                                      |
| <b>-onlret</b>            | 在终端 NL 不执行 CR 功能。                                                                                     |
| <b>onocr</b>              | 不在零列输出 CR 字符。                                                                                         |
| <b>-onocr</b>             | 在零列输出 CR 字符。                                                                                          |
| <b>opost</b>              | 处理输出。                                                                                                 |
| <b>-opost</b>             | 不处理输出；即忽略所有其他输出选项。                                                                                    |
| <b>ofdel</b>              | 使用 DEL 字符作为填充字符。                                                                                      |
| <b>-ofdel</b>             | 使用 NUL 字符作为填充字符。                                                                                      |
| <b>tab0, tab1, tab2</b>   | 为水平制表符选择延迟样式 ( <b>tab0</b> 表示没有延迟)。                                                                   |
| <b>tab3</b>               | 扩展制表符至多个空格。                                                                                           |
| <b>vt0, vt1</b>           | 为垂直制表符选择延迟样式 ( <b>vt0</b> 表示没有延迟)。                                                                    |
| <br>                      |                                                                                                       |
| 本地模式                      |                                                                                                       |
| <b>echo</b>               | 回送每个输入的字符。                                                                                            |
| <b>-echo</b>              | 不回送字符。                                                                                                |
| <b>echoctl</b>            | 以 ^X (Ctrl-X) 回送控制字符，X 是将 100 八进制加到控制字符代码中给出的字符。                                                      |
| <b>-echoctl</b>           | 不以 ^X (Ctrl-X) 回送控制字符。                                                                                |
| <b>echoe</b>              | 以 “backspace space backspace” 字符串回送 ERASE 字符。<br><b>注：</b> 该模式不保持对列位置的跟踪，因此您可能在擦除制表符和转义序列等符号时得到意外的结果。 |
| <b>-echoe</b>             | 不回送 ERASE 字符，只回送退格符。                                                                                  |
| <b>echok</b>              | 在 KILL 字符后回送 NL 字符。                                                                                   |
| <b>-echok</b>             | 在 KILL 字符后不回送 NL 字符。                                                                                  |
| <b>echoke</b>             | 通过擦除输出行上的每个字符，回送 KILL 字符。                                                                             |
| <b>-echoke</b>            | 只回送 KILL 字符。                                                                                          |
| <b>echonl</b>             | 回送 NL 字符。                                                                                             |
| <b>-echonl</b>            | 不回送 NL 字符。                                                                                            |
| <b>echoprt</b>            | 以 / (斜杠) 和 \ (反斜杠) 向后回送擦除的字符。                                                                         |
| <b>-echoprt</b>           | 不以 / (斜杠) 和 \ (反斜杠) 向后回送擦除的字符。                                                                        |

|                 |                                                                                                                                                                             |
|-----------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>icanon</b>   | 启用规范输入（规范输入允许使用 ERASE 和 KILL 字符进行输入行的编辑）。请参阅 <i>AIX 5L Version 5.3 Communications Programming Concepts</i> 中的 Line Discipline Module (ldterm) 中关于 canonical mode input 的讨论。 |
| <b>-icanon</b>  | 禁用规范输入。                                                                                                                                                                     |
| <b>ixexten</b>  | 指定从输入数据中无法识别实现性定义的功能。要识别以下控制字符，需要设置 <b>ixexten: eol2、dsusp、reprint、discard、werase、lnext</b> 。与这些模式关联的功能也需要设置 <b>ixexten: imaxbel、echoke、echoprnt、echoctl</b> 。              |
| <b>-ixexten</b> | 指定从输入数据中无法识别实现性定义的功能。                                                                                                                                                       |
| <b>isig</b>     | 启用对特殊控制字符（INTR、SUSP 和 QUIT）的字符检查。                                                                                                                                           |
| <b>-isig</b>    | 禁用对特殊控制字符（INTR、SUSP 和 QUIT）的字符检查。                                                                                                                                           |
| <b>noflsh</b>   | 不清除 INTR、SUSP 或 QUIT 控制字符之后的缓冲区。                                                                                                                                            |
| <b>-noflsh</b>  | 清除 INTR、SUSP 或 QUIT 控制字符之后的缓冲区。                                                                                                                                             |
| <b>pending</b>  | 下次读操作暂挂或输入到达时，要重新输入从原始模式转换为规范模式后被暂挂的输入。暂挂是一个内部状态位。                                                                                                                          |
| <b>-pending</b> | 没有文本暂挂。                                                                                                                                                                     |
| <b>tostop</b>   | 为背景输出发出 <b>SIGTOU</b> 信号。                                                                                                                                                   |
| <b>-tostop</b>  | 不为背景输出发出 <b>SIGTOU</b> 信号。                                                                                                                                                  |
| <b>xcase</b>    | 在输入中回送大写字符，并在输出显示的大写字符之前加上 \ (反斜杠)。                                                                                                                                         |
| <b>-xcase</b>   | 不在输入时回送大写字符。                                                                                                                                                                |

## 硬件流量控制模式

这些选项是对 X/Open Portability Guide Issue 4 标准的扩展。

|                 |                     |
|-----------------|---------------------|
| <b>cdxon</b>    | 输出时启用 CD 硬件流量控制模式。  |
| <b>-cdxon</b>   | 输出时禁用 CD 硬件流量控制模式。  |
| <b>ctsxon</b>   | 输出时启用 CTS 硬件流量控制模式。 |
| <b>-ctsxon</b>  | 输出时禁用 CTS 硬件流量控制模式。 |
| <b>dtrxoff</b>  | 输入时启用 DTR 硬件流量控制模式。 |
| <b>-dtrxoff</b> | 输入时禁用 DTR 硬件流量控制模式。 |
| <b>rtsxoff</b>  | 输入时启用 RTS 硬件流量控制模式。 |
| <b>-rtsxoff</b> | 输入时禁用 RTS 硬件流量控制模式。 |

## 控制指定

要将一个控制字符指定到某字符串中，请输入：

```
stty Control String
```

其中，*Control* 参数可以是 INTR、QUIT、ERASE、KILL、EOF、EOL、EOL2、START、STOP、SUSP、DSUSP、REPRINT、DISCARD、WERASE、LNEXT、MIN 或 TIME 参数。（使用字符 MIN 和 TIME 时，请加上 **-icanon** 选项。）

注：MIN 和 TIME 的值解释为整数值，而不是字符值。

*String* 参数可以是任何单一的字符，比如 c。以下内容为控制赋值的示例：

```
stty STOP c
```

另一种指定控制字符的方法可以是：输入一个字符序列，它是由一个 \^ (反斜杠，插入记号) 后面跟着一个单一字符组成的。如果跟在 ^ (插入记号) 后的单一字符是下表的 ^c (插入记号 c) 栏中列出的字符之一，将设置相应的控制字符值。例如，要使用 ? (问号) 字符指定 DEL 控制字符，请输入字符串 \^? (反斜杠，插入记号，问号)，如下：

stty ERASE \^?

stty 中的插入记号控制字符

| ^c   | 替换值   |
|------|-------|
| a, A | <SOH> |
| b, B | <STX> |
| c, C | <ETX> |
| d, D | <EOT> |
| e, E | <ENQ> |
| f, F | <ACK> |
| g, G | <BEL> |
| h, H | <BS>  |
| i, I | <HT>  |
| j, J | <LF>  |
| k, K | <VT>  |
| l, L | <FF>  |
| m, M | <CR>  |
| n, N | <SO>  |
| o, O | <SI>  |
| p, P | <DLE> |
| q, Q | <DC1> |
| r, R | <DC2> |
| s, S | <DC3> |
| t, T | <DC4> |
| u, U | <NAK> |
| v, V | <SYN> |
| w, W | <ETB> |
| x, X | <CAN> |
| y, Y | <EM>  |
| z, Z | <SUB> |
| [    | <ESC> |
| \    | <FS>  |
| ]    | <GS>  |
| ^    | <RS>  |
| _    | <US>  |
| ?    | <DEL> |
| @    | <NUL> |

#### 组合模式

**cooked**

**ek**

**evenp**

**-evenp**

**lcase, LCASE**

请参阅 **-raw** 选项。

分别将 ERASE 和 KILL 字符设置为 Ctrl-H 和 Ctrl-U 按键顺序。

启用 **parenb** 和 **cs7**。

禁用 **parenb** 并设置 **cs8**。

设置 **xcase**, **iuclc** 和 **olcuc**。在工作站只以大写字符使用。

|                       |                                                                                                   |
|-----------------------|---------------------------------------------------------------------------------------------------|
| <b>-lcase, -LCASE</b> | 设置 <b>-xcase</b> 、 <b>-iuclc</b> 和 <b>-olcuc</b> 。                                                |
| <b>nl</b>             | 设置 <b>-icrnl</b> 和 <b>-onlcr</b> 。                                                                |
| <b>-nl</b>            | 设置 <b>icrnl</b> 、 <b>onlcr</b> 、 <b>-inlcr</b> 、 <b>-igncr</b> 、 <b>-ocrnl</b> 和 <b>-onlret</b> 。 |
| <b>oddp</b>           | 启用 <b>parenb</b> 、 <b>cs7</b> 和 <b>parodd</b> 。                                                   |
| <b>-oddp</b>          | 禁用 <b>parenb</b> 并设置 <b>cs8</b> 。                                                                 |
| <b>parity</b>         | 请参阅 <b>evenp</b> 选项。                                                                              |
| <b>-parity</b>        | 请参阅 <b>-evenp</b> 选项。                                                                             |
| <b>sane</b>           | 将参数重新设置为合理的值。                                                                                     |
| <b>raw</b>            | 允许原始模式输入（不包括输入处理，例如 <b>erase</b> 、 <b>kill</b> 或 <b>interrupt</b> ）；传回奇偶（校验）位。                    |
| <b>-raw</b>           | 允许规范输入方式。                                                                                         |
| <b>tabs</b>           | 保留制表符。                                                                                            |
| <b>-tabs, tab3</b>    | 显示时将制表符替换为空格。                                                                                     |

### 窗口大小

|                                        |                             |
|----------------------------------------|-----------------------------|
| <b>cols <i>n</i>, columns <i>n</i></b> | 将终端（窗口）大小记录为有 <i>n</i> 列。   |
| <b>rows <i>n</i></b>                   | 将终端（窗口）大小记录为有 <i>n</i> 行。   |
| <b>size</b>                            | 将终端（窗口）大小显示到标准输出（先是行，再是列）中。 |

## 废弃选项

**stty** 命令支持以下 BSD 选项。对于每个选项，都给出了建议的 POSIX 选项。

|                   |                                      |
|-------------------|--------------------------------------|
| <b>all</b>        | 使用 <b>stty -a</b> 命令显示所有当前设置。        |
| <b>crt</b>        | 使用 <b>sane</b> 选项将参数重新设置为合理的值。       |
| <b>crtbs</b>      | 使用 <b>-echoe</b> 选项。                 |
| <b>crterase</b>   | 使用 <b>echoe</b> 选项。                  |
| <b>-crterase</b>  | 使用 <b>-echoe</b> 选项。                 |
| <b>crtkill</b>    | 使用 <b>echoke</b> 选项。                 |
| <b>-crtkill</b>   | 使用 <b>echok</b> 和 <b>-echoke</b> 选项。 |
| <b>ctlecho</b>    | 使用 <b>echoctl</b> 选项。                |
| <b>-ctlecho</b>   | 使用 <b>-echoctl</b> 选项。               |
| <b>decctlq</b>    | 使用 <b>-ixany</b> 选项。                 |
| <b>-decctlq</b>   | 使用 <b>ixany</b> 选项。                  |
| <b>even</b>       | 使用 <b>evenp</b> 选项。                  |
| <b>-even</b>      | 使用 <b>-evenp</b> 选项。                 |
| <b>everything</b> | 使用 <b>stty -a</b> 命令显示所有当前设置。        |
| <b>litout</b>     | 使用 <b>-opost</b> 选项。                 |
| <b>-litout</b>    | 使用 <b>opost</b> 选项。                  |
| <b>odd</b>        | 使用 <b>oddp</b> 选项。                   |
| <b>-odd</b>       | 使用 <b>-oddp</b> 选项。                  |
| <b>pass8</b>      | 使用 <b>-istrip</b> 选项。                |
| <b>-pass8</b>     | 使用 <b>istrip</b> 选项。                 |
| <b>prterase</b>   | 使用 <b>echoprt</b> 选项。                |
| <b>speed</b>      | 使用 <b>stty</b> 命令显示当前设置。             |
| <b>tandem</b>     | 使用 <b>ixoff</b> 选项。                  |
| <b>-tandem</b>    | 使用 <b>-ixoff</b> 选项。                 |

## 示例

1. 要显示您的工作站配置的简短列表，请输入：

```
stty
```

这将列出与缺省值不同的设置。

2. 要显示您的工作站配置的完整列表，请输入：

```
stty -a
```

3. 要启用停止列表滚动出屏幕的按键顺序，请输入：

```
stty ixon ixany
```

这将设置 **ixon** 模式，从而可以通过按下 **Ctrl-S** 按键顺序来停止列表的滚动。**ixany** 标志允许按任意键来恢复列表的滚动。正常的工作站配置包含 **ixon** 和 **ixany** 标志，使您可以用 **Ctrl-S** 按键顺序停止列表的滚动，而只有 **Ctrl-Q** 按键顺序才能使列表重新滚动。

4. 要重新设置搞乱的配置，请输入：

```
Ctrl-J stty sane Ctrl-J
```

在命令前后按 **Ctrl-J** 按键顺序，而不是按 **Enter** 键。当控制 **Enter** 键处理的参数搞乱时，系统通常能识别 **Ctrl-J** 按键顺序。

有时屏幕上显示的信息可能看上去很奇怪，或当您按 **Enter** 键时系统不反应。当您使用 **stty** 命令，而它带有不兼容的参数或参数的操作难以理解时，可能会发生这种情况。当面向屏幕的应用程序异常结束并且无法复位工作站配置时，也可能发生这种情况。

输入 **stty sane** 命令设置合理的配置，但它可能与您的正常配置略有不同。

5. 要保存和恢复终端的配置：

```
OLDCONFIG=`stty -g`          # save configuration
stty -echo                  # do not display password
echo "Enter password: \c"
read PASSWD                 # get the password
stty $OLDCONFIG             # restore configuration
```

该命令保存工作站的配置、关闭回送信号、读取密码并恢复原始配置。

输入 **stty - echo** 命令关闭回送信号，这意味着当您在键盘上输入密码时，密码并不出现在屏幕上。此操作与 **echo** 命令（该命令在屏幕上显示消息）毫无关系。

## 文件

**/usr/bin/stty** 包含 **stty** 命令。

## 相关信息

**terminfo** 文件、**tty** 特殊文件 **termios.h** 头文件。

《AIX 5L V5.3 通用编程概念：编写并调试程序》中的『TTY 子系统概述』。

《AIX 5L V5.3 通用编程概念：编写并调试程序》中的 **ldterm Line Discipline**。

《AIX 5L V5.3 本地语言支持指南和参考大全》中的『本地语言支持』。

---

## stty-cxma 命令

### 用途

设置和报告 128 端口异步子系统的 TTY 配置的终端选项。

### 语法

```
stty-cxma [ -a ] [ -g ] [ Option(s) ] [ ttyName ]
```

### 描述

如果没有指定标志或选项，则 **stty-cxma** 命令报告所有 128 端口特殊驱动程序的设置和调制解调器信号，以及由 **stty** 命令报告的关于 tty 设备（当前的标准输入）的所有标准参数。

可指定 *ttyName* 参数来设置或报告不作为标准输入的 tty 设备的选项。*ttyName* 参数可以是简单的 tty 名称（比如 **tty0**），或加上前缀 **/dev/**（比如 **/dev/tty0**）。没有载波时，此选项可用于调制解调器控制线路。

可指定更多的选项来更改流量控制设置、设置透明显示选项、强制调制解调器控制线路以及显示所有 tty 设置。将无法识别的选项传递给 **stty** 命令以寻求解释。

### 标志

**-a** 将所有唯一的 128 端口设置，以及所有由 **stty -a** 报告的标准 tty 设置写到标准输出中。

**-g** 将选项设置写到标准输出中，其格式可以由另一条 **stty** 命令使用。

### 选项

以下选项指定要立即执行的瞬态操作：

|                 |                             |
|-----------------|-----------------------------|
| <b>break</b>    | 将 250 MS 中断信号向外发送到 tty 线路上。 |
| <b>flush</b>    | 立即废弃 tty 输入和输出。             |
| <b>flushin</b>  | 只废弃 tty 输入。                 |
| <b>flushout</b> | 只废弃 tty 输出。                 |

以下选项所指定的操作在设备关闭前有效。下次打开该设备时，使用缺省值。

|                 |                                      |
|-----------------|--------------------------------------|
| <b>dtr</b>      | 如果没有选定 DTR 硬件流量控制，则产生 DTR 调制解调器控制线路。 |
| <b>-dtr</b>     | 如果没有选定 DTR 硬件流量控制，则删除 DTR 调制解调器控制线路。 |
| <b>rts</b>      | 如果没有选定 RTS 硬件流量控制，则产生 RTS 调制解调器控制线路。 |
| <b>-rts</b>     | 如果没有选定 RTS 硬件流量控制，则删除 RTS 调制解调器控制线路。 |
| <b>startin</b>  | 释放流量控制以恢复停止的输入。                      |
| <b>startout</b> | 重新启动停止的输出，正如接收到一个 XON 字符。            |
| <b>stopin</b>   | 激活流量控制以停止输入。                         |
| <b>stopout</b>  | 停止输出，正如接收到一个 XOFF 字符。                |

|                   |                                                                                                                                             |
|-------------------|---------------------------------------------------------------------------------------------------------------------------------------------|
| <b>2200flow</b>   | 在端口启用 2200 样式流量控制。2200 终端支持附加打印机，并使用以下四种流量控制字符：<br><b>0xF8</b> 终端 XON<br><b>0xF9</b> 打印机 XON<br><b>0xFA</b> 终端 XOFF<br><b>0xFB</b> 打印机 XOFF |
| <b>-2200flow</b>  | 在端口禁用 2200 样式流量控制。                                                                                                                          |
| <b>2200print</b>  | 分别运行终端的流量控制和透明打印设备的流量控制（由 <b>2200flow</b> 选项设置）。                                                                                            |
| <b>-2200print</b> | 同时运行终端和打印机的流量控制（由 <b>2200flow</b> 选项设置）。因此，如果接收到终端或打印机 XOFF 字符，则暂停所有输出，直到接收到匹配的 XON 字符。                                                     |
| <b>altpin</b>     | 在模块化的接口上切换 DSR 和 DCD 输入的位置，从而在使用 8-pin RJ45 接口（而非 10-pin RJ45 接口）时可以使用 DCD。                                                                 |
| <b>-altpin</b>    | 使用 10-pin RJ45 接口时恢复 DSR 的可用性。                                                                                                              |
| <b>aixon</b>      | 为了对 XON 和 XOFF 使用两个唯一的字符，启用辅助的流量控制。如果已接收到两个 XOFF 字符，则只有再接收到两个 XON 字符，传输才会得到恢复。                                                              |
| <b>-aixon</b>     | 禁用辅助的流量控制。                                                                                                                                  |
| <b>astartc c</b>  | 设置辅助的 XON 流量控制字符。这个字符可能是十进制、八进制或十六进制数字。                                                                                                     |
| <b>astopc c</b>   | 设置辅助的 XOFF 流量控制字符。这个字符可能是十进制、八进制或十六进制数字。                                                                                                    |
| <b>bufsize n</b>  | 设置驱动程序对透明打印机输入缓冲区大小的估计。一段休止状态之后，在减少到由上述选定的 <b>maxcps</b> 选项速率所指定的最大 CPS 速率前，驱动程序将这些许多字符以脉冲串传输到透明打印机上。缺省值是 100 个字符。                          |
| <b>ctspace</b>    | 启用 CTS 硬件输出流量控制，从而在 CTS 删除时，本地传输暂停。                                                                                                         |
| <b>-ctspace</b>   | 禁用 CTS 硬件输出流量控制。                                                                                                                            |
| <b>dcdspace</b>   | 启用 DCD 硬件输出流量控制，从而在 DCD 删除时，本地传输暂停。                                                                                                         |
| <b>-dcdspace</b>  | 禁用 DCD 硬件输出流量控制。                                                                                                                            |
| <b>dsrpace</b>    | 启用 DSR 硬件输出流量控制，从而在 DSR 删除时，本地传输暂停。                                                                                                         |
| <b>-dsrpace</b>   | 禁用 DSR 硬件输出流量控制。                                                                                                                            |
| <b>dtrpace</b>    | 启用 DTR 硬件输入流量控制，这样在 DTR 删除时，远程传输暂停。                                                                                                         |
| <b>-dtrpace</b>   | 禁用 DTR 硬件输入流量控制。                                                                                                                            |
| <b>edelay n</b>   | 设置 128 端口异步适配器唤醒输入驱动程序的速率。适配器以每 <i>n</i> 毫秒的速率唤醒驱动程序。缺省值是 100 毫秒。                                                                           |
| <b>fastbaud</b>   | 更改波特率表，将 50 波特率变为 57600 波特率。                                                                                                                |
| <b>-fastbaud</b>  | 恢复波特率表，将 57500 波特率变为 50 波特率。                                                                                                                |
| <b>fastcook</b>   | 在 128 端口异步适配器上执行已加工的输出处理，以减少主机 CPU 的使用和增加原始模式输入的性能。                                                                                         |
| <b>-fastcook</b>  | 禁用已加工的输出处理。                                                                                                                                 |
| <b>forcedcd</b>   | 禁用载波监听，这样即使载波不存在，也可以打开和使用 tty。                                                                                                              |
| <b>-forcedcd</b>  | 再次启动载波监听。                                                                                                                                   |
| <b>maxchar n</b>  | 设置由驱动程序放在输出队列中的透明打印字符的最大数值。减小这个数值将增加系统开销；增大这个数值将延迟透明打印机使用时操作员击键的回送时间。缺省值是 50 个字符。                                                           |
| <b>maxcps n</b>   | 设置字符输出到透明打印设备的最大 CPS（字符/秒）速率。选择的速率应该略低于平均打印速率。如果这个数值太低，则打印机速度将变慢。如果这个数值太高，打印机将求助于流量控制，而 CRT 上的用户条目也将相应受损。缺省值是 100 CPS。                      |
| <b>offstr s</b>   | 设置关闭透明打印的 CRT 转义序列。一个任意的八进制字符 <i>xxx</i> 可能以 <code>\xxx</code> 的形式给出。                                                                       |
| <b>onstr s</b>    | 设置打开透明打印的 CRT 转义序列。一个任意的八进制字符 <i>xxx</i> 可能以 <code>\xxx</code> 的形式给出。                                                                       |
| <b>rtspace</b>    | 启用 RTS 硬件输入流量控制，这样 RTS 的删除会使远程传输暂停。                                                                                                         |
| <b>-rtspace</b>   | 禁用 RTS 硬件输入流量控制。                                                                                                                            |
| <b>startc c</b>   | 设置 XON 流量控制字符。这个字符可能是十进制、八进制或十六进制数字。                                                                                                        |
| <b>stopc c</b>    | 设置 XOFF 流量控制字符。这个字符可能是十进制、八进制或十六进制数字。                                                                                                       |



**term** *t* 将透明打印机的开关字符串设置为内部缺省表中指定的值。以下终端使用内部缺省值：**adm31**、**ansi**、**dg200**、**dg210**、**hz1500**、**mc5**、**microterm**、**multiterm**、**pcterm**、**tvi**、**vp-a2**、**vp-60**、**vt52**、**vt100**、**vt220**、**wyse30**、**wyse50**、**wyse60** 或 **wyse75**。如果在内部缺省值表中找不到该终端类型，则透明打印开关字符串将设置为 **termcap** 文件中由 **po** 和 **pf** 属性指定的值。

## 示例

1. 要显示在 128 端口异步控制器上配置为 `/dev/tty0` 的 tty 端口的所有唯一的 128 端口设置以及所有的标准 tty 设置，请输入：

```
stty-cxma -a tty0
```

2. 使用 128 端口异步控制器上配置为 `/dev/tty3` 的 tty 端口的 8-pin RJ45 接口时，要使 DCD 可用，请输入：

```
stty-cxma altpin tty3
```

该命令相互交换模块化接口上的 DSR 和 DCD 输入的位置。

## 文件

`/usr/sbin/tty/stty-cxma` 包含 **stty-cxma** 命令。

## 相关信息

**stty** 命令。

---

## style 命令

### 用途

分析文档的表面特征。

### 语法

```
style [ -a ] [ -e ] [ -lNumber ] [ -ml ] [ -mm ] [ -p ] [ -P ] [ -rNumber ] File ...
```

### 描述

**style** 命令分析英语文档的写入样式的表面特征。它会报告可读性、语句长度和结构、字长度和用法、动词类型以及语句开头。因为 **style** 命令在查看文本前先运行 **deroff** 命令，则应该将包含相应的格式化信息的头文件作为输入的一部分包括。

注：使用非标准格式化的宏可能会导致不正确的语句中断。

### 标志

|                 |                                                         |
|-----------------|---------------------------------------------------------|
| <b>-a</b>       | 显示所有语句以及它们的长度和可读性索引。                                    |
| <b>-e</b>       | 显示所有以虚词（比如“ <b>There are</b> ”）开头的语句。                   |
| <b>-lNumber</b> | 显示所有多于 <i>Number</i> 参数所指定的数字的语句。                       |
| <b>-ml</b>      | 使 <b>deroff</b> 命令跳过列表；如果文档包含许多语句片断的列表，则使用 <b>-ml</b> 。 |
| <b>-mm</b>      | 覆盖缺省的 <b>ms</b> 宏数据包。                                   |
| <b>-p</b>       | 显示所有包含被动动词的语句。                                          |

**-P** 显示文档中的字的语音部分。  
**-rNumber** 显示所有可读性索引大于 *Number* 的语句。

## 相关信息

**diction** 命令、**deroff** 命令。

**ms** 宏数据包。

---

## su 命令

### 用途

更改与会话关联的用户标识。

### 语法

**su** [ - ] [ *Name* [ *Argument* ... ] ]

### 描述

**su** 命令将用户凭证更改为 **root** 用户凭证或由 *Name* 参数指定的用户，然后开始一个新的会话。用户名可能包含 DCE 单元规范。

**注：**当转换为 DCE 用户时，**root** 用户不需要满足分布式计算环境（DCE）认证的要求。在这种情况下，该用户不会获得 DCE 凭证。

任何由 *Arguments* 参数指定的变量（例如标志或参数），必须与 *Name* 参数指定的用户所定义的登录 shell 相关联。将这些变量传递到指定用户的登录 shell。例如，如果用户 Fred 的登录 shell 是 **/usr/bin/csh**，则您可以包含 **csh** 命令的任何标志，例如 **-f** 标志。当 **su** 命令运行时，它将 **-f** 标志传递到 **csh** 命令。当 **csh** 命令运行时，**-f** 标志省略 **.cshrc** 启动脚本。

以下功能是由 **su** 命令执行的：

|      |                                                                                                                      |
|------|----------------------------------------------------------------------------------------------------------------------|
| 帐户检查 | 验证用户帐户以确定：该用户帐户存在，已经为 <b>su</b> 命令启用，当前用户是属于被允许使用 <b>su</b> 命令以切换至这个帐户的组，且可以从当前控制终端使用它。                              |
| 用户认证 | 使用系统定义的主要用户认证方法确认用户身份。如果密码过期，那么用户必须提供新密码。                                                                            |
| 凭证建立 | 使用用户数据库中的值建立初始用户凭证。这些凭证定义用户对系统的访问权和可记帐性。                                                                             |
| 会话启动 | 如果指定 <b>-</b> 标志，则 <b>su</b> 命令使用用户数据库和 <b>/etc/environment</b> 文件中的值来初始化用户环境。如果未使用 <b>-</b> 标志，则 <b>su</b> 命令不更改目录。 |

这些功能按照显示的顺序执行。如果某个功能失败，将不执行以后的功能。要获取这些功能的语义，请参阅 **ckuseracct**、**ckuserID**、**authenticate**、**setpcred** 和 **setpenv** 子例程。

要恢复以前的会话，请输入 **exit** 或按下 **Ctrl-D** 按键顺序。这项操作结束由 **su** 命令调用的 shell，并使您返回以前的 shell、用户标识和环境。

如果 **su** 命令是从 **/usr/bin/tsh** shell（可信 shell）运行的，您将从那个 shell 退出。**su** 命令不更改控制终端的安全性特征。

每次执行 **su** 命令时，将在 **/var/adm/sulog** 文件中产生一个条目。**/var/adm/sulog** 文件记录以下信息：日期、时间、系统名称以及登录名。**/var/adm/sulog** 文件也记录登录尝试是否成功：+（加号）表示登录成功，-（减号）表示登录失败。

注：仅当用户的 **rlogin** 和 **login** 属性在 **/etc/security/user** 中都设置为 **false** 时，**su** 命令的成功使用才会复位 **/etc/security/lastlog** 文件中的 **unsuccessful\_login\_count** 属性。否则，**su** 命令不会复位 **unsuccessful\_login\_count**，因为管理员经常使用 **su** 命令来修订用户帐户问题。用户能通过本地或远程登录来复位属性。

## 标志

- 指定要设置的进程环境，就好像用户已经使用 **login** 命令登录系统。当前环境中无任何东西传播到新的 shell 中。

## 安全性

**su** 命令是支持 PAM 的带有服务名称 **su** 的应用程序。在 **/etc/security/login.cfg** 的 **usw** 节中，通过修改 **auth\_type** 属性值来将要使用 PAM 进行认证的系统范围配置设置为作为 **root** 用户的 **PAM\_AUTH**。

启用 PAM 时使用的认证机制取决于 **/etc/pam.conf** 中 **su** 服务的配置。**su** 命令需要用于 **auth**、**account**、**password** 和 **session** 模块类型的 **/etc/pam.conf** 条目。要使 **su** 命令在整个 PAM 认证过程中表现的行为与标准 AIX 认证中所见到的行为相类似，必须在 **auth** 和 **account** **su** 服务堆栈中充分使用 **pam\_allowroot** 模块并在 **pam\_aix** 之前调用它。以下列出了在 **/etc/pam.conf** 中 **su** 服务的建议配置：

```
#
# AIX su 配置
#
su auth sufficient /usr/lib/security/pam_allowroot
su auth required /usr/lib/security/pam_aix

su account sufficient /usr/lib/security/pam_allowroot
su account required /usr/lib/security/pam_aix

su session required /usr/lib/security/pam_aix

su password required /usr/lib/security/pam_aix
```

## 示例

1. 要获取 **root** 用户权限，请输入：

```
su
```

该命令使用有效用户标识和 **root** 用户特权来运行子 shell。将要求您输入 **root** 用户密码。按下 End-of-File（文件结束符）、**Ctrl+D** 按键顺序来结束子 shell，并且返回到原始 shell 会话和特权。

2. 要获得 **jim** 用户的特权，请输入：

```
su jim
```

该命令使用有效用户标识和 **jim** 特权来运行子 shell。

3. 要设置环境，就好像您已作为 **jim** 用户登录，请输入：

```
su - jim
```

这将使用 jim 的登录环境来启动子 shell。

4. 要使用 root 用户权限运行 backup 命令，然后返回到原始 shell，请输入：

```
su root "-c /usr/sbin/backup -9 -u"
```

这将在 root 用户的缺省 shell 内使用 root 用户权限来运行 **backup** 命令。在查询要执行的命令时，必须给出正确的 root 用户密码。

## 文件

|                                    |                  |
|------------------------------------|------------------|
| <code>/usr/bin/su</code>           | 包含 <b>su</b> 命令。 |
| <code>/etc/environment</code>      | 包含用户环境值。         |
| <code>/etc/group</code>            | 包含组的基本属性。        |
| <code>/etc/passwd</code>           | 包含基本用户属性。        |
| <code>/etc/security/user</code>    | 包含用户的扩展属性。       |
| <code>/etc/security/environ</code> | 包含用户的环境属性。       |
| <code>/etc/security/limits</code>  | 包含用户进程资源限制。      |
| <code>/etc/security/passwd</code>  | 包含密码信息。          |
| <code>/var/adm/sulog</code>        | 包含登录尝试的信息。       |

## 相关信息

**bsh** 命令、**csch** 命令、**getty** 命令、**ksh** 命令、**login** 命令、**setgroups** 命令、**setsenv** 命令、**tsh** 命令和 **tsm** 命令。

**authenticate** 子例程、**ckuseracct** 子例程、**ckuserID** 子例程、**setpcred** 子例程、**setpenv** 子例程。

关于用户的标识和认证、自主访问控制、可信计算库和审计的更多信息，请参阅《安全性》中的 *Securing the network*。

---

## subj 命令

### 用途

生成文档中主题列表。

### 语法

```
subj [ File ... ]
```

### 描述

**subj** 命令在一个或多个英文文件中搜索那些可能与主题页索引相应的主题，并在标准输出上显示主题列表。文档必须包含格式化命令（比如 **nroff**、**troff** 以及 **mm** 等命令）来最好地使用 **subj** 命令。

除了每个语句的头一个单词，**subj** 命令还选择大写单词的序列作为主题。这样，如果语句以正确的名词开始，大写规则将不选择这个单词作为主题。然而，由于每个语句通常都是从新的行开始，故可能会错误地选择从一行的中间开始的语句中的第一个单词。同样，**subj** 命令从摘要、标题和主题句（每段的首句）中选择修订名词序列。因此，有时会将一个单词错误地分类为名词或形容词。

**subj** 命令的输出可能不适应您的需要，因而必须对它进行编辑。

## 参数

*File* 指定 **subj** 命令要搜索的英文文件，在这些文件中寻找适当的主题以建立索引。

## 相关信息

**mm** 命令、**ndx** 命令、**nroff** 命令、**troff** 命令。

---

## sum 命令

### 用途

显示文件的校验和以及块计数。

### 语法

```
sum [ -i ] [ -r ] [ -o ] [ File ... ]
```

### 描述

**sum** 命令读取由 *File* 参数指定的文件，并计算该文件中的校验和以及 1024 字节块的数目。如果未指定选项，则使用逐字节算法，比如 BSD 4.3 缺省算法。如果未指定文件，则读取标准输入。将校验和以及 1024 字节块的数目写到标准输出中。**sum** 命令通常用来确定通过传输线路复制或通信的文件是否是原始文件的严格副本。

### 标志

- i** 如果输入文件是二进制文件，则允许用户计算校验和而不必包含头信息。如果输入文件不是二进制文件，则校验和包含头信息。
- o** 使用逐字算法计算校验和。带有 **-o** 标志的 **sum** 命令与版本 2 的 **sum** 命令在校验和上是兼容的，但在块数目上不兼容。
- r** 使用逐字节算法计算校验和。使用 **-r** 标志和不使用选项是相同的。

注：缺省是 BSD 4.3 缺省算法，而不再是逐字计算算法。

### 退出状态

该命令返回以下退出值：

- 0** 成功完成。
- >0** 发生错误。

### 示例

要显示 **file1** 和 **file2** 文件中的校验和以及 1024 字节块数目，请输入：

```
sum file1 file2
```

如果 **file1** 文件中的校验和是 32830，**file2** 文件中的校验和是 32481，并且 **file1** 文件包含一个块，而 **file2** 包含四个块，则 **sum** 命令显示：

```
32830          1      file1
32481          4      file2
```

# 文件

`/usr/bin/sum` 包含 `sum` 命令。

## 相关信息

`cksum` 命令和 `wc` 命令。

《操作系统与设备管理》中的『文件系统』说明了文件系统的含义以及使用文件系统的原因。

---

## suma 命令

### 用途

创建任务以从修订服务器自动下载维护和技术级别。

### 语法

要创建、编辑或调度 **SUMA** 任务:

```
suma { { [ -x ] [-w ] } | -s CronSched } [ -a Field=Value ]... [ TaskID ]
```

要列出 **SUMA** 任务:

```
suma -l [ TaskID ]...
```

要列出或编辑缺省 **SUMA** 任务:

```
suma -D [ -a Field=Value ]...
```

要列出或编辑 **SUMA** 全局配置设置:

```
suma -c [ -a Field=Value ]...
```

要取消调度 **SUMA** 任务:

```
suma -u TaskID
```

要删除 **SUMA** 任务:

```
suma -d TaskID
```

### 描述

`suma` 命令可以用来执行对 **SUMA** 任务或策略的以下操作:

- 创建
- 编辑
- 显示列表
- 调度
- 取消调度
- 删除

将对由唯一任务标识表示的任务执行指定操作。对于创建或编辑 SUMA 任务的情况，如果不指定 *TaskID*，则采用创建操作，并且将生成唯一的 *TaskID*。对于 **-l** 标志，如果不指定 *TaskID*，将显示所有 SUMA 任务的列表。对于 **-c** 标志，如果不指定 **-a** 标志，将列出 SUMA 全局配置设置。

## 标志

**-c** 列出或编辑 SUMA 全局配置设置。 **-a** 标志允许一个或多个配置设置更新为指定值。如果使用时不带有 **-a** 标志，将列出所有的 SUMA 配置设置。

可以用 **-a** 标志编辑的配置设置如下：

### **FIXSERVER\_PROTOCOL**

与修订服务器进行通信时，它指定传送是否将使用 http（非安全）或 https（安全）。

缺省值：http

允许值：http 和 https

### **DOWNLOAD\_PROTOCOL**

下载文件集时，它指定传送是否将利用 http、ftp 或 https（安全）传送。

缺省值：ftp

允许值：ftp、http 和 https

### **DL\_TIMEOUT\_SEC**

指定在下载操作过程中等待修订服务器响应的的时间（以秒计）。

缺省值：180

允许值：大于零的整数。

### **MAX\_CONCURRENT\_DOWNLOADS**

同时下载的文件最大数目。

缺省值：5

允许值：大于零的整数和“无限制”（或 -1）。

### **HTTP\_PROXY**

用于 HTTP 传送的代理服务器和端口。

缺省值：空（禁用）

允许值：

- 空（不使用 HTTP 代理）
- 使用以下格式的 URL：
  - *server.name:port*；例如：httpproxy.customer.com:8080/
  - *protocol://server.name:port*；例如：http://httpproxy.customer.com:8080/
  - *protocol://username:password@server.name:port*；例如：  
http://username:password@httpproxy.customer.com:8080/

### **HTTPS\_PROXY**

用于 HTTPS 传送的代理服务器和端口。

缺省值：空（禁用）

允许值：

- 空（不使用 HTTPS 代理）
- 使用以下格式的 URL：
  - *server.name:port*；例如：httpsproxy.customer.com:8080/
  - *protocol://server.name:port*；例如：http://httpsproxy.customer.com:8080/
  - *protocol://username:password@server.name:port*；例如：  
http://username:password@httpsproxy.customer.com:8080/



-c (续)

### FTP\_PROXY

用于 FTP 传送的代理服务器和端口。

缺省值: 空 (禁用)

允许值:

- 空 (不使用 FTP 代理)
- 使用以下格式的 URL:
  - *server.name:port*; 例如: `ftpproxy.customer.com:8080/`
  - *protocol://server.name:port*; 例如: `http://ftpproxy.customer.com:8080/`
  - *protocol://username:password@server.name:port*; 例如:  
`http://username:password@ftpproxy.customer.com:8080/`

### SCREEN\_VERBOSE

将记录信息的冗余级别指定到标准输出和标准错误。在 **suma** 命令从命令行或 SMIT 界面运行时使用。它不适用于从 cron 运行的调度任务。

缺省值: LVL\_INFO

允许值:

- LVL\_OFF: 不显示或记录信息。
- LVL\_ERROR: 显示错误消息和其他非常重要的消息。
- LVL\_WARNING: 除 LVL\_ERROR 消息之外还显示警告消息。
- LVL\_INFO: 除 LVL\_WARNING 消息之外还显示参考消息。
- LVL\_VERBOSE: 除 LVL\_INFO 消息之外还显示详细参考消息。
- LVL\_DEBUG: 显示调试输出。此设置用于调试用途并且不应该用于常规操作。

### NOTIFY\_VERBOSE

指定以电子邮件通知形式发送的信息的冗余级别。仅适用于从 cron 运行的调度任务。

缺省值: LVL\_INFO

允许值: LVL\_OFF、LVL\_ERROR、LVL\_WARNING、LVL\_INFO、LVL\_VERBOSE 和 LVL\_DEBUG (有关值的描述, 请参考 **SCREEN\_VERBOSE** 设置)

### LOGFILE\_VERBOSE

指定记录到日志文件 (`/var/adm/ras/suma.log`) 的信息的冗余级别。注: LVL\_OFF 设置仍然会将信息记录到下载日志文件 (`/var/adm/ras/suma_dl.log`)。

缺省值: LVL\_VERBOSE

允许值: LVL\_OFF、LVL\_ERROR、LVL\_WARNING、LVL\_INFO、LVL\_VERBOSE 和 LVL\_DEBUG (有关值的描述, 请参考 **SCREEN\_VERBOSE** 设置)

### MAXLOGSIZE\_MB

允许日志文件达到的最大大小 (以 MB 计数)。

缺省值: 1

允许值: 大于零的整数。

### REMOVE\_CONFLICTING\_UPDATES

指定 **lppmgr** 是否应该在清洁操作过程中运行时除去与基本图像 (**lppmgr -u** 标志) 具有相同级别的有冲突的更新。

缺省值: yes

允许值: yes 和 no

### REMOVE\_DUP\_BASE\_LEVELS

指定 **lppmgr** 是否应该在清洁操作过程中运行时除去重复基本级别 (**lppmgr -b** 标志)。

缺省值: yes

允许值: yes 和 no

-c (续)

#### REMOVE\_SUPERSEDE

指定 **lppmgr** 是否应该在清洁操作过程中运行时除去被取代的文件集更新 (**lppmgr -x** 标志)。

缺省值: **yes**

允许值: **yes** 和 **no**

#### TMPDIR

指定存储临时文件的目录。

缺省值: **/var/suma/tmp**

允许值: 当前存在的任何目录。

- d** 删除与给定 *TaskID* 相关联的 SUMA 任务以及使用 **-s** 标志创建并用于此任务的所有调度。
- D** 列出或编辑缺省的 SUMA 任务。**-a** 标志允许缺省任务的一个或多个字段更新为指定值。如果使用时不带有 **-a** 标志, 将列出缺省 SUMA 任务。
- l** 列出 SUMA 任务。如果使用时不带有 *TaskID*, 将列出所有 SUMA 任务。*TaskID* 可用于指定一个或多个要列出的任务标识。
- s CronSched** 调度 SUMA 任务。如果在创建新任务时指定, 则意味着保存 (**-w** 标志功能)。*CronSched* 是空格分隔的加引号的五个条目 (分钟、小时、日、月和工作日) 的列表。这些条目的有效值如下 (有关其他详细信息, 请参阅 **crontab** 手册页):
- 分钟: 0 到 59
  - 小时: 0 到 23
  - 日: 1 到 31
  - 月: 1 到 12
  - 工作日: 0 到 6 (星期天到星期六)
- u** 取消调度 SUMA 任务。它将除去指定 *TaskID* 的任何调度信息。
- w** 撰写或保存 SUMA 任务。如果使用时不带有 **-s** 标志, 将保存此任务, 同时允许稍后添加调度信息。如果结合 **-x** 标志使用, 将立即运行此任务, 同时还将其保存。
- x** 指定是否应该立即运行 SUMA 任务并且不调度它。如果使用时不带有 **-w** 标志, 将不为以后使用保存此任务。

**-a** *Field=Value ...*

将指定的值分配到指定的字段。要对 SUMA 任务进行创建或编辑操作，受支持的字段和值如下。

### RqType

如果运行 `suma` 时带有 **Security**、**Critical** 或 **Latest** 的 **RqType**，则 **RqType** 是唯一的必填字段。有关在这种情况下将使用的缺省值，请参阅示例 1（下面）。其他 **RqType** 值（**APAR**、**PTF**、**ML**、**TL**、**SP** 和 **Fileset**）将需要其他 *Field=Value* 信息的规范。

**APAR** 指定下载 APAR 的请求。例如：IY12345。

**PTF** 指定下载 PTF 的请求。例如：U123456。

**ML** 指定下载特定维护或技术级别的请求。例如：5300-03。

**TL** 指定下载特定维护或技术级别的请求。例如：5300-04。

**SP** 指定下载特定 service pack 或最终 service pack 的请求。例如：5300-04-02 或 5300-04-CSP。

**Fileset** 指定下载特定文件集的请求。例如 `bos.rte.lvm`。

### Security

指定下载最新安全性修订包的请求。

**Critical** 指定下载最新关键修订包的请求。这些修订包解决了很多普遍的（HIPER）问题。

**Latest** 指定下载所有最新修订包的请求。

### RqName

请求的项目的特定名称（例如：IY12345、U123456、5200-07、5200-08、5200-08-02、5200-08-CSP 或 `bos.rte.lvm`）。在 **RqType** 等于 **Security**、**Critical** 或 **Latest** 时，**RqName** 字段应该保留为空。

### RqLevel

只有在请求的项目类型为 **Fileset**、**Latest**、**Security**、**Critical** 或 **APAR** 时才能指定请求级别。当类型为 **Fileset** 时，这将包含请求的文件集的 VRMF；例如：5.3.0.40。如果保留为空，则将请求文件集的最新级别。

当类型为 **APAR**、**Security**、**Critical** 或 **Latest** 时，可以指定一个可选的维护或技术级别（例如 5300-04）以请求指定维护或技术级别的修订，而无需改为下一个维护或技术级别；例如：5300-05。

### Repeats

指定是否只执行一次该任务且不将其保留在系统上，并在找到项目之前重复，还是永久性重复。**Repeats** 字段仅适用于从 `cron` 运行的调度任务（带有 **Download** 或 **Clean** 的 **Action**）。如果从命令行运行，或者如果 **Action=Preview**，则忽略此字段，并且不会除去任务。

**y** 设置重复任务，并需要已经为该任务指定了带有 **-s** 标志的 *CronSched*。当 **RqType** 等于 **APAR**、**PTF**、**ML**、**TL**、**SP** 或 **Fileset**（带有包含 VRMF 的 **RqLevel**）时，一旦找到项目就立即除去任务。当 **RqType** 等于 **Security**、**Critical**、**Latest** 或 **Fileset**（**RqLevel** 包含意味着“最新”的空白）时，任务将设置为永久重复。

**n** 指定是否只执行一次该任务并且不将其保留在系统上。

-a (续)

### **DisplayName**

表明此 SUMA 任务的显示名称 (例如, “可用时下载 APAR IY12345”)。在 SMIT 中查看现有的 SUMA 任务时可以使用它。

### **Action**

#### **Preview**

指定将执行的下载预览。不下载文件集。

#### **Download**

指定将文件集下载到基于策略的 **DLTarget**。

**Clean** 指定将文件集下载到基于策略的 **DLTarget**, 随后进行清洁操作。**lppmgr** 命令将用来清洁 **DLTarget** 中不需要的文件集。在 SUMA 全局配置设置中列出的三个可配置的 **lppmgr** 标志选项如下:

- REMOVE\_CONFLICTING\_UPDATES
- REMOVE\_DUP\_BASE\_LEVELS
- REMOVE\_SUPERSEDE

### **PreCoreqs**

表明是否应该下载 prerequisite 和 corequisite 文件集 (y 或 n)。

**Ifreqs** 表明是否应该下载 iferequisite 文件集 (y 或 n)。

### **Supersedes**

表明是否应该下载取代请求文件集的文件集 (而不是请求文件集) (y 或 n)。

### **ResolvePE**

表明对指定为回归或 PE 的修订包采取何种操作。应该指定三个有效值中的一个:

**Never** 指定从不解析回归并始终特定地下下载请求的内容。

#### **Always**

指定始终解析回归。如果解析的文件集可用, 将下载它。否则将不下载任何内容, 并且这会成为下载预览操作。

#### **IfAvailable**

如果解析的文件集可用, 则指定解析回归。如果解析的文件集可用, 将下载解析的文件集, 而不是回归。否则, 将下载指定为回归的文件集。

### **DLTarget**

包含将存储下载文件的目录位置。如果不指定该字段, 则为其指定值 **/usr/sys/inst.images**, 并且根据映像类型将文件存储在某个目录中; 例如: **/usr/sys/inst.images/install/ppc** 或 **/usr/sys/inst.images/RPMS/ppc**。

### **NotifyEmail**

包含在文件集下载或预览之后将通知电子邮件发送到的一个或多个电子邮件地址 (多个地址应该以逗号分隔)。如果将任务调度为以后执行 (已经指定 *CronSched*), 将仅发送通知。

-a (续)

#### FilterDir

指定要过滤的修订包资源库目录的名称，以便不下载重复的修订包。它允许过滤目录（**DLTarget** 除外）。例如，您可以过滤 NIM lpp\_source，而不需要下载到此目录。如果保留为空，将使用 **DLTarget**。

#### FilterML

指定要过滤的维护或技术级别，例如：5300-04。如果不指定，将使用本地系统上的 **oslevel -r** 返回的值。

#### FilterSysFile

指定要过滤的包含已安装软件列表的文件名称（用 **lslpp -Lc** 来生成）或者要过滤的系统名称。如果要过滤本地系统，应该使用“localhost”。如果不希望过滤已安装软件，应该使用 **/dev/null**。在 NIM 环境中，可以指定 NIM 客户机。

#### MaxDLSize

要通过任何单个策略执行下载的数据的最大允许数量（以 MB 计）。如果确定下载将超出此大小，将不下载。可以指定“unlimited”或 -1 值来表明要下载的数据数量没有上限。

**Extend** 指定 y 将自动扩展 **DLTarget** 驻留的文件系统。如果指定 n 并且需要额外空间进行下载，将不下载。

#### MaxFSSize

**DLTarget** 文件系统可扩展到的最大允许大小以 MB 计。如果确定下载将超出此限制，将不下载。可以指定“unlimited”或 -1 值来表明文件系统大小没有上限（即，可以扩展文件系统，直到物理磁盘空间耗尽）。

## 参数

*TaskID* 指定与任务相关的唯一数字标识。创建任务时将指定此标识。

## 退出状态

0 命令成功完成。  
>0 发生错误。

## 示例

1. 要列出 SUMA 全局配置设置，请输入以下内容：

```
suma -c
```

将显示类似于以下内容的输出：

```
FIXSERVER_PROTOCOL=http
DOWNLOAD_PROTOCOL=ftp
DL_TIMEOUT_SEC=180
MAX_CONCURRENT_DOWNLOADS=5
HTTP_PROXY=
HTTPS_PROXY=
FTP_PROXY=
SCREEN_VERBOSE=LVL_INFO
NOTIFY_VERBOSE=LVL_INFO
LOGFILE_VERBOSE=LVL_VERBOSE
MAXLOGSIZE_MB=1
```

```
REMOVE_CONFLICTING_UPDATES=yes
REMOVE_DUP_BASE_LEVELS=yes
REMOVE_SUPERSEDE=yes
TMPDIR=/var/suma/tmp
```

2. 要编辑 SUMA 全局配置设置以将日志文件的最大大小更改为 2 MB，请输入以下内容：

```
suma -c -a MAXLOGSIZE_MB=2
```

3. 要列出 SUMA 任务缺省值，请输入以下内容：

```
suma -D
```

将显示类似于以下内容的输出：

```
DisplayName=
Action=Download
RqType=Security
RqName=
RqLevel=
PreCoreqs=y
Ifreqs=y
Supersedes=n
ResolvePE=IfAvailable
Repeats=y
DLTarget=/usr/sys/inst.images
NotifyEmail=root
FilterDir=/usr/sys/inst.images
FilterML=
FilterSysFile=localhost
MaxDLSize=-1
Extend=y
MaxFSSize=-1
```

4. 要创建并调度每月（例如，每月 15 号凌晨 2:30）将下载最新关键修订包的任务，请输入以下内容：

```
suma -s "30 2 15 * *" -a RqType=Critical \
-a DisplayName="Critical fixes - 15th Monthly"
```

**注：**将返回此新建任务的任务标识。此示例假定将利用某些 SUMA 任务缺省值（如 **suma -D** 示例中所显示）。例如，当任务缺省值为 **DLTarget=/usr/sys/inst.images**，installp 图像将下载到 **/usr/sys/inst.images/installp/ppc** 目录。

5. 要查看通过运行 **suma -s CronSched** 命令设置的 SUMA 调度信息，请输入以下内容：

```
crontab -l root
```

6. 要创建和调度每周（例如，每星期四凌晨 3 点）将检查一次特定 APAR 的任务，当它可用时则下载它，并将电子邮件通知发送到远程系统用户，请输入以下内容：

```
suma -s "0 3 * * 4" -a RqType=APAR -a RqName=IY12345 \
-a NotifyEmail="bob.smith@host2,ann@host2"
```

**注：**针对每周将检查 APAR 的此任务，**Repeats** 字段需要设置为 **y**。在这种情况下，找到 APAR 之后将删除此任务。如果 **Repeats=n**，删除此任务之前只检查一次。

7. 要创建并调度每月（例如，每月 20 号凌晨 4:30）将检查 **bos.rte.install** 文件集最新级别的任务，请输入以下内容：

```
suma -s "30 4 20 * *" -a RqType=Fileset -a RqName=bos.rte.install \
-a RqLevel=latest -a Repeats=y
```

**注：**通过设置 **Repeats=y**，此任务将“永久性重复”并在成功下载之后不会删除。

8. 要创建并调度每周（例如，每星期四凌晨 3 点）将检查 **bos.rte.install** 文件集特定级别的任务，请输入以下内容：

```
suma -s "0 3 * * 4" -a RqType=Fileset -a RqName=bos.rte.install \
-a RqLevel=5.2.0.30 -a Repeats=y
```

注: 通过设置 **Repeats=y**, 此任务将“重复直到找到”并且在成功下载特定文件集之后删除。

9. 要创建并调度在星期一晚上 11:00 将整个带有 5300-02 推荐维护包的 AIX 5L V5.3 下载到 **/lppsrc/5302** 目录的任务, 同时关闭过滤安装软件, 并在下载之后运行 **lppmgr** 清洁操作来除去任何被取代的更新、重复的基本级别和有冲突的更新, 请输入以下内容:

```
suma -s "0 23 * * 1" -a Action=Clean -a RqType=ML -a RqName=5300-02 \  
-a DLTarget=/lppsrc/5302 -a FilterSysFile=/dev/null
```

注: 在运行指定 **Action=Clean** 的任务之前, 可以运行 **suma -c** 来验证运行 **lppmgr** 时将使用的 SUMA 全局配置设置。在这种情况下, 将 **REMOVE\_SUPERSEDE**、**REMOVE\_DUP\_BASE\_LEVELS** 和 **REMOVE\_CONFLICTING\_UPDATES** 全部设置为 **yes** 将产生上述操作。

10. 要创建并调度在星期一晚上 11:00 将整个带有 5300-03 推荐维护包的 AIX 5L V5.3 下载到 **/tmp/lppsrc/5303** 目录的任务, 同时关闭过滤安装软件 (允许下载已安装在系统上的更新), 但是过滤已包含在 **/lppsrc** 中的任何更新, 请输入以下内容:

```
suma -s "0 23 * * 1" -a RqType=ML -a RqName=5300-03 \  
-a DLTarget=/tmp/lppsrc/5303 -a FilterDir=/lppsrc -a FilterSysFile=/dev/null
```

注: 成功完成此任务之后, 将除去此任务, 因为 **RqType=ML** 是一个“重复直到找到”任务。然而, 如果 **Repeats=n**, 将只检查 5300-01 ML 一次, 如果在修订服务器上找不到 ML, 将删除任务, 因为它已经设置为不重复。

11. 要立即执行将下载任何关键修订包 (自 5300-00 以来已变得可用, 并且尚未安装在系统上) 的任务, 请输入以下内容:

```
suma -x -a RqType=Critical -a FilterML=5300-00 -a FilterSysFile=localhost
```

注: 此示例假定将利用某些 SUMA 任务缺省值 (如 **suma -D** 示例中所显示)。例如, 当任务缺省值为 **DLTarget=/usr/sys/inst.images**, **installp** 图像将下载到 **/usr/sys/inst.images/installp/ppc** 目录。

12. 要立即执行一个任务 (此任务将执行预览来检查 APAR 是否存在于修订服务器上, 如果 APAR 尚不存在, 则创建并保存此任务以便以后调用), 请输入以下内容:

```
suma -x -w -a Action=Preview -a RqType=APAR -a RqName=IY12345
```

注: 将返回此新建任务的任务标识。

13. 要立即执行以上示例的新建任务 (假定已返回任务标识 23), 尝试下载 APAR 并保存任务标识 23 的 **Action=Download** 设置, 请输入以下内容:

```
suma -x -w -a Action=Download 23
```

注: 因为正在从命令行运行此任务而且未通过 **cron** 调度它, 所以将忽略 **Repeats** 字段, 并且无论是否找到 APAR 都不删除此任务。

14. 要调度任务标识 23 以每周 (例如, 每星期四凌晨 3 点) 重复检查一次特定 APAR, 并在它可用时下载它, 请输入以下内容:

```
suma -s "0 3 * * 4" -a Repeats=y 23
```

注: 找到 APAR 时将删除此任务。

15. 要取消调度任务 (此操作将从 **/var/spool/cron/crontabs** 目录中的 **crontab** 文件除去其调度信息), 请输入以下内容:

```
suma -u 23
```

16. 要删除一个任务 (如果存在其调度信息, 此操作还将除去此信息), 请输入以下内容:

```
suma -d 23
```

17. 要列出多个 SUMA 任务, 其中 4 和 23 表示任务标识, 请输入以下内容:

```
suma -l 4 23
```

18. 要列出所有 SUMA 任务，请输入以下内容：

```
suma -l
```

19. 要创建并调度每月（例如，每月 15 号凌晨 2:30）将检查所有最新更新的任务，并下载在 **/tmp/latest** 资源库中尚不存在的任何更新，请输入以下内容：

```
suma -s "30 2 15 * *" -a RqType=Latest \  
-a DLTarget=/tmp/latest -a FilterDir=/tmp/latest
```

注：将返回此新建任务的任务标识。

## 位置

**/usr/suma/bin/suma**

## 文件

**/usr/suma/bin/suma**

包含 **suma** 命令。

**/usr/sbin/suma**

链接到 **/usr/suma/bin/suma**。

**/var/adm/ras/suma.log**

包含运行 **suma** 命令的详细结果。

**/var/adm/ras/suma\_dl.log**

包含已下载文件的列表。

**/var/spool/cron/crontabs**

包含要调度的 crontab 文件的目录。

## 相关信息

《AIX 5L V5.3 命令参考大全，卷 1》中的 crontab 命令。

《AIX 5L V5.3 命令参考大全，卷 3》中的 lppmgr 命令。

---

## survd 守护程序

### 用途

控制 surveillance 守护程序。

### 语法

#### 控制 Surveillance 参数

```
survd [ -d Delay ] [ -h ]
```

#### 重新设置 Surveillance

```
survd -r
```

### 描述

**survd** 命令使具有 root 用户权限的用户能够控制 surveillance 守护程序。您可以使用 **-d Delay** 标志，来选择操作系统发送至 bring-up 微处理器（BUMP）信号的频率。如果在给定的延迟内 BUMP 未收到来自操作系统的信号，您还可以选择 BUMP 重新引导系统的方式（硬件或软件重新引导）。（**-h** 标志表示需要硬件重新引导）。最后，您可以使用 **-r** 标志来决定关闭 surveillance。



**survd** 守护程序只在具有 Micro Channel® I/O for AIX 5.1 及更早版本的多处理器系统上工作。对于 IBM 系统，这包括 IBM 7012 型 G 系列、IBM 7013 型 J 系列 和 IBM 7015 型 R 系列 for AIX 5.1 及更早版本。

**注：** 请不要使用 **kill** 命令来停止 **surveillance**：在这种情况下，**BUMP** 将不再接收从守护程序传来的信号，并将重新引导系统。

## 标志

- d Delay** 指定操作系统发送至 **BUMP** 信号的周期。 *Delay* 参数表示以秒计数的周期。授权给 *Delay* 参数的最小值是 10 秒。如果未指定这个标志，频率将是 60 秒。
- h** 这表示如果 **BUMP** 没有在定义的延迟内从操作系统接收到任何信号，它将执行硬件重新引导。如果未指定这个标志，将执行软件重新引导。
- r** 关闭 **surveillance** 并删除 **surveillance** 守护程序。这个标志不能和 **-d** 或 **-h** 标志一起使用，并且如果没有运行守护程序，则这个标志没有作用。

## 安全性

访问控制：只有 root 用户才能运行该命令。

## 示例

1. 要将信号周期设置为 70 秒，请输入：

```
survd -d 70
```

请注意，由于未指定 **-h** 标志，所以如果在 70 秒内 **BUMP** 未接收到任何信号，**BUMP** 将执行软件重新引导。

2. 要关闭 **surveillance**，请输入：

```
survd -r
```

---

## suspendvsd 命令

### 用途

**suspendvsd** - 取消激活可用的虚拟共享磁盘。

### 语法

```
suspendvsd {-a | vsd_name...}
```

### 描述

**suspendvsd** 命令使指定的虚拟共享磁盘从活动状态变为暂挂状态。它们保持可用。在虚拟共享磁盘是活动的时候读写请求是活动的，现在被暂挂和停止。后继的读写操作也将停止。如果虚拟共享磁盘处于暂挂状态，则该命令将其保留在暂挂状态。

可以使用“系统管理界面工具”（SMIT）来运行该命令。要使用 SMIT，输入：

```
smit vsd_mgmt
```

并选择 **Suspend a Virtual Shared Disk** 选项。

在常规情况下，您不应发出该命令。“可恢复的”虚拟共享磁盘子系统使用该命令以受控制的方式来管理共享磁盘。如果您发出该命令，结果可能是不可预测的。

## 标志

**-a** 指定处于活动状态的所有虚拟共享磁盘都要进行暂挂。

## 参数

*vsd\_name* 指定一个虚拟共享磁盘。如果虚拟共享磁盘未处于活动状态，则您将得到一条错误消息。

## 安全

您必须具有 `root` 用户权限以运行该命令。

## 退出状态

**0** 说明命令已成功完成。

**非零** 表明发生错误。

## 限制

您必须从对等域中联机的节点发出该命令。要使对等域联机，请使用 `startprdomain` 命令。要使现有对等域中一个特定的节点联机，请使用 `startprnode` 命令。有关创建和管理 RSCT 对等域的更多信息，请参阅 *RSCT: Administration Guide*。

在常规情况下，您不应发出该命令。“可恢复的”虚拟共享磁盘子系统使用该命令以受控制的方式来管理共享磁盘。如果您发出该命令，结果可能是不可预测的。

## 标准输出

当前 RVSD 子系统运行级别。

## 示例

要指定的虚拟共享磁盘 `vsd1vg1n1` 从活动状态变为暂挂状态，请输入：

```
suspendvsd vsd1vg1n1
```

## 位置

`/opt/rsct/vsd/bin/suspendvsd`

## 相关信息

命令：`cfgvsd`、`lsvsd`、`preparevsd`、`resumevsd`、`startvsd`、`stopvsd`、`ucfgvsd`

---

## svmon 命令

### 用途

捕获并分析虚拟内存快照

## 语法

### 全局报告

```
svmon -G [ -i Interval [ NumIntervals ] ] [ -z ]
```

### 用户报告

```
svmon -U [ LogName1...LogNameN ] [ -r ] [ -n | -s ] [ -w | -f | -c ] [ -t Count ] [ -u | -p | -g | -v ] [ -i Interval [ NumIntervals ] ] [ -l ] [ -j ] [ -d ] [ -z ] [ -m ]
```

### 命令报告

```
svmon -C Command1...CommandN [ -r ] [ -n | -s ] [ -w | -f | -c ] [ -t Count ] [ -u | -p | -g | -v ] [ -i Interval [ NumIntervals ] ] [ -l ] [ -j ] [ -d ] [ -z ] [ -m ] [ -q [ s | L ] ]
```

### 工作负载管理类报告

```
svmon -W [ ClassName1...ClassNameN ] [ -e ] [ -r ] [ -n | -s ] [ -w | -f | -c ] [ -t Count ] [ -u | -p | -g | -v ] [ -i Interval [ NumIntervals ] ] [ -l ] [ -d ] [ -z ] [ -m ] [ -q [ s | L ] ]
```

### 工作负载管理层报告

```
svmon -T [ Tier1...TierN ] [ -a SupClassName ] [ -x ] [ -e ] [ -r ] [ -u | -p | -g | -v ] [ -n | -s ] [ -w | -f | -c ] [ -q [ s | L ] ] [ -t Count ] [ -i Interval [ NumIntervals ] ] [ -l ] [ -z ] [ -m ]
```

### 进程报告

```
svmon -P [ PID1...PIDN ] [ -r ] [ -n | -s ] [ -w | -f | -c ] [ -t Count ] [ -u | -p | -g | -v ] [ -i Interval [ NumIntervals ] ] [ -l ] [ -j ] [ -z ] [ -m ] [ -q [ s | L ] ]
```

### 段报告

```
svmon -S [ SID1...SIDN ] [ -r ] [ -n | -s ] [ -w | -f | -c ] [ -t Count ] [ -u | -p | -g | -v ] [ -i Interval [ NumIntervals ] ] [ -l ] [ -j ] [ -z ] [ -m ] [ -q [ s | L ] ]
```

### 详细报告

```
svmon -D SID1...SIDN [ -b ] [ -i Interval [ NumIntervals ] ] [ -z ] [ -q [ s | L ] ]
```

### 分帧报告

```
svmon -F [ Frame1...FrameN ] [ -i Interval [ NumIntervals ] ] [ -z ] [ -q [ s | L ] ]
```

## 描述

**svmon** 命令显示当前内存状态的信息。因为 **svmon** 命令在启用中断的用户级别运行，所以显示的信息不组成真正的内存快照。

段是一组页面，用来报告内存消耗的基本对象。因此 **svmon** 所报告的统计信息是以页面表示的。

页面是虚拟内存块，而帧是实内存块。帧的大小总是为 4 KB，而页面可以有不同的大小。基本页面大小为 4 KB。段内所有页面的大小都相同。

由 **svmon** 报告的值以 4KB 为单位来表示，除非它们明显属于特定页面大小，在这种情况下，它们以此特定大小的页面表示。例如，在以下报告中，第一部分以 4KB 为单位来表示。第二和第三部分相应地以 **PageSize** 大小的页面和 **Psize** 大小的页面表示。

```
# svmon -P 1
```

```
-----
```

| Pid     | 命令          | 正在使用  | 固定内存 | 调页空间 | 虚拟    | 64 位 | 多线程 | 16 MB |
|---------|-------------|-------|------|------|-------|------|-----|-------|
| 1486924 | ptxtstshmlp | 14161 | 7876 | 0    | 14160 | N    | N   | Y     |

  

| 页大小     | 正在使用  | 固定内存 | 调页空间 | 虚拟    |
|---------|-------|------|------|-------|
| s 4 KB  | 10065 | 3780 | 0    | 10064 |
| L 16 MB | 1     | 1    | 0    | 1     |

  

| Vsid  | Esid | 类型   | 描述                           | 页大小 | 正在使用 | 固定内存 | 调页空间 | 虚拟   |
|-------|------|------|------------------------------|-----|------|------|------|------|
| 0     | 0    | 工作   | 内核段                          | s   | 6083 | 3777 | 0    | 6083 |
| 1640  | 3    | 工作   | 共享内存段<br>(lgpg_vsid=ffc1fff) | L   | 1    | 1    | 0    | 1    |
| 148aa | d    | 工作   | 装入器段                         | s   | 3961 | 0    | 0    | 3961 |
| b685  | 2    | 工作   | 进程专用                         | s   | 13   | 3    | 0    | 13   |
| 13689 | f    | 工作   | 共享库数据                        | s   | 7    | 0    | 0    | 7    |
| b605  | 1    | 持久代码 | /dev/hd1:413                 | s   | 1    | 0    | -    | -    |

使用正在使用、空闲、固定内存、虚拟以及调页空间计数器报告内存消耗。

- 正在使用计数器表示使用的帧数。
- 空闲计数器表示所有内存池中的空闲帧数。
- 固定内存计数器表示固定内存的帧数，即无法交换的帧数。
- 虚拟计数器表示分配在系统虚拟空间中的页数。
- 调页空间计数器表示在调页空间上保留或使用的页数。

在一些报告中 (**-P**、**-U**、**-C**、**-W**、**-T**)，标题行显示了实体的全局内存消耗，它总结了该实体所用的段的内存消耗。在支持可选择页面大小的系统上会显示内存消耗情况，后面跟有不同的受支持页面大小的分布情况。

| 页大小     | 正在使用 | 固定内存 | 调页空间 | 虚拟   |
|---------|------|------|------|------|
| s 4 KB  | 6086 | 2000 | 1000 | 6080 |
| L 16 MB | 1    | 1    | 0    | 1    |

列标题为:

**页大小** 所考虑的页面的大小。有一个字母与页面大小相关联并显示在大小的前面。大小以千字节 (KB)、兆字节 (MB) 或吉字节 (GB) 为单位来表示。与页面大小相关联的字母作为提示显示，在其他报告中用来引用页面大小。

#### 正在使用

表示用户使用的段中的实内存的**页大小**的页面总数。

#### 固定内存

表示用户所用的段中固定内存的**页大小**的页面总数。

#### 调页空间

表示用户所用的段中的调页空间中保留或使用的**页大小**的页面总数。

#### 虚拟

表示在进程虚拟空间中分配的**页大小**的页面总数。

不同的值以**页大小**页面单元来表示。

一个段可被多个进程使用。这样一个段的每个页面在使用该段的每个进程的正在使用、固定内存、虚拟或调页空间字段中得以说明。因此，所有活动的进程的正在使用、固定内存、虚拟以及调页空间字段的总数可能会超过内存或调页空间的页面总数。

虚拟内存管理器管理虚拟页面计数器，将其仅用于统计，这意味着它们不会始终最新的，并且它们的值可能小于相应的正在使用计数器。

段属于以下五种类型之一：

|       |                                          |
|-------|------------------------------------------|
| 持久    | 用于处理文件和目录的段。                             |
| 工作    | 用于实现进程的数据区域的段和共享内存段。                     |
| 客户    | 用于实现诸如网络文件系统（NFS）和 CD-ROM 文件系统等虚拟文件系统的段。 |
| 映射    | 用于实现内存文件的映射的段。                           |
| 实内存映射 | 用于从虚拟地址空间存取 IO 空间的段。                     |

## 实现细节

该命令只在 PowerPC® 平台上有效。

## 报告

svmon 命令创建九种类型的报告：

- 全局
- 用户
- 命令
- 类
- 层
- 进程
- 段
- 详细的段
- 帧

此处对每种报告类型一一描述。

### 全局报告

只有指定了 **-G** 标志，才显示全局报告。全局报告中的列标题是：

**内存** 指定描述实内存使用的统计信息，包含：

**大小** 实内存帧数（实内存的大小）

**注：** 它包含被内存缩放工具 **rmss** 命令设置为不可用的空闲帧。

**正在使用**

包含页面的帧数

**注：** 在定义了保留池（例如 16 MB 页池）的系统上，该值包括为这些保留池中的任何一个池所保留的帧。

**空闲** 所有内存池中的空闲帧数

**固定内存**

包含固定内存的页面的帧数

**注:** 在定义了保留池（例如 16 MB 页池）的系统上，该值包括为这些保留池中的任何一个池所保留的帧。

**虚拟** 分配于系统虚拟空间的页数

**窃取** 被 **rmss** 窃取并且被 VMM 标记为不可使用的帧的数目

#### 正在使用

指定正在使用的实内存子集的统计信息，包含：

**工作** 包含工作段页面的帧数

**持久** 包含持久段页面的帧数

**客户** 包含客户段页面的帧数

#### 固定内存

指定包含固定内存的页面的实内存的子集上的统计信息，包含：

**工作** 包含工作段固定内存页面的帧数

**持久** 包含持久段固定内存页面的帧数

**客户** 包含客户段固定内存页面的帧数

#### 调页空间

指定描述调页空间使用的统计信息。

**大小** 调页空间的大小

#### 正在使用

使用的调页空间页面的数量

**页大小** 只在系统上可以使用可选择页面大小（不是 4 KB）的情况下显示。指定可以在系统上使用的每种页面大小单独的统计信息。

**页大小** 指以下统计信息中的页大小

**池大小** 对于使用保留池的页面大小（例如 16 MB 页池），池中的页面数。

#### 正在使用

已使用的具有该大小的页面的数目

#### 页面空间

在调页空间中已分配的具有该大小的页面的数目

#### 固定内存

具有该大小的固定页面的数目

**虚拟** 在系统虚拟空间中已分配的具有该大小的页面数目

## 用户报告

只有指定了 **-U** 标志，才显示用户报告。用户登录报告中的列标题是：

**用户** 表示用户名。

#### 正在使用

表示用户所用的段中实内存中的页面总数。

#### 固定内存

表示用户使用的段中固定内存的页面总数。

## 调页空间

表示用户使用的段在调页空间保留或使用的页面总数。

**虚拟** 表示在进程虚拟空间中分配的页面总数。

如果该用户所拥有的进程使用的页面大小不是基本的 4KB 页面大小，那么这些统计信息后就跟有在不同页面大小之间的分布情况。

如果指定了 **-d** 标志，那么关于指定的登录用户名运行的所有进程的信息会跟随这些统计信息。该信息与**进程报告**中所描述的信息相似。

然后 **svmon** 命令显示关于那些进程使用的段的信息。该组段分为三个类别：

- 由虚拟内存管理器标志为 **system** 的段
- 仅用于属于用户登录名的该组进程的段
- 在几个用户间共享的段

对于每个类别，都有一个关于相关的段的页面消耗的报告。如果相关（请参阅上述内容），则还会有一个关于不同页面大小的使用情况的报告。然后列出段的特定度量值。

如果指定了 **-l** 标志，那么最后一个类别中的每个段后都跟有使用该段的进程标识的列表。除了进程标识之外，也显示启动该进程的登录用户名。请参阅关于特殊段处理的 **-l** 标志的描述。

如果指定了 **-r** 标志，那么每个段后都跟有已分配了页面的范围（在该段内）。

## 命令报告

只有指定了 **-C** 标志，才显示命令报告。命令报告中的列标题是：

**命令** 表示命令名称

### 正在使用

表示该命令使用的段的实内存的页面总数（运行该命令的所有进程）。

### 固定内存

表示该命令使用的段的固定内存页面总数（运行该命令的所有进程）。

### 调页空间

表示该命令使用的段在调页空间上保留或使用的页面总数。

**虚拟** 表示在该命令的虚拟空间分配的页面总数。

如果运行该命令的进程使用的页面大小不是基本的 4 KB 页面大小，那么这些统计信息后就跟有不同页面大小的分布情况。

如果指定了 **-d** 标志,那么关于运行该指定的命令的所有进程的信息跟随这些统计。该信息与**进程报告**中所描述的信息相似。

下一条 **svmon** 显示了关于那些进程使用的段的信息。该组段分为三个类别：

- 被虚拟内存管理器标志为 **system** 的段
- 只有运行该命令的组进程才能使用的段
- 几个命令名称共享的段

对于每个类别，都有一个关于相关的段的页面消耗的报告。如果相关（请参阅上述内容），则还会有一个关于不同页面大小的使用情况的报告。然后列出段的特定度量值。

如果指定了 **-l** 标志, 那么最后一个类别中的每个段后都跟有使用该段的进程标识的列表。除了进程标识符之外, 也显示该进程所运行的命令名称。请参阅关于特殊段处理的 **-l** 标志的描述。

如果指定了 **-r** 标志, 那么每个段后都跟有已分配了页面的范围 (在该段内)。

如果指定了 **-q** 标志, 那么只报告与具有请求大小的段相关的信息。

## 工作负载管理报告

通过以下两种报告类型, **svmon** 能够报告与工作负载管理相关的活动。

- 类报告
- 层报告

在运行工作负载管理器时, 这些报告可用。否则, 会显示必须启动工作负载管理器的消息, 并且不报告统计信息。工作负载管理器以被动的模式运行时, 在显示统计信息之前 **svmon** 会显示以下信息: 工作负载管理器以被动模式运行。

请参考《操作系统与设备管理》中的『工作负载管理』。

### 类报告

只有指定了 **-W** 标志, 才显示该类报告。类报告中的列标题是:

#### 类或超类

表示类或超类名称。

#### 正在使用

表示属于该类的段中实内存的页面总数。

#### 固定内存

表示属于该类的段中固定内存的页面总数。

#### 调页空间

表示属于该类的段在调页空间保留或使用的页面总数。

**虚拟** 表示在该类的虚拟空间分配的页面总数。

如果类使用的页大小不是基本的 4 KB 页大小, 那么这些统计信息后就跟有不同页大小的分布情况。

显示这些统计信息后, **svmon** 显示关于属于该类的段的信息。

如果指定了 **-e** 标志, 那么显示该类的子类的统计信息, 并且每个子类报告段的统计信息。在这种情况下, 类参数必须是超类名称。

如果指定了 **-l** 标志, 那么每个段后都跟有正在使用该段的进程标识的列表。除了进程标识符外, 进程所属的工作负载类也会显示出来。请参阅关于特殊段处理的 **-l** 标志的描述。

如果指定了 **-r** 标志, 那么每个段后都跟有已分配了页面的范围 (在该段内)。

如果指定了 **-q** 标志, 那么只报告具有请求大小的段。在这种情况下, 全局度量值只与这些段相关。

### 层报告

只有指定了 **-T** 标志, 才显示层报告。层报告中的列标题是:

**层** 表示层数



**超类** 可选的列标题。表示当层适用于超类时的超类名称（使用 **-a** 标志时）。

#### 正在使用

表示属于层的段的实内存中的页面总数。

#### 固定内存

表示属于层的段的固定内存页面总数。

#### 调页空间

表示属于层的段在调页空间上保留或使用的页面总数。

**虚拟** 表示在层的虚拟空间中分配的页面总数。

如果层使用的页大小不是基本的 4 KB 页大小，那么这些统计信息后就跟有不同页面大小的分布情况。

显示统计信息之后，**svmon** 显示关于属于层的类的信息。

如果指定了 **-e** 标志，那么就报告属于层的每个超类的子类的统计信息。

如果指定了 **-x** 标志，那么 **svmon** 显示关于属于每个类的段的信息。

**-r** 和 **-l** 标志只能与 **-x** 结合使用。

如果指定了 **-l** 标志，那么每个段后都跟有正在使用该段的进程标识的列表。除了进程标识符外，进程所属的层数和类也会显示出来。请参阅关于特殊段处理的 **-l** 标志的描述。

如果指定了 **-r** 标志，那么每个段后都跟有已分配了页面的范围（在该段内）。

如果指定了 **-q** 标志，那么只报告具有请求大小的段。在这种情况下，全局度量值只与这些段相关。

## 进程报告

只有指定了 **-P** 标志，才显示进程报告。进程报告中的列标题是：

**Pid** 表示进程标识符。

**命令** 表示进程在运行的命令。

#### 正在使用

表示进程使用的段中的实内存的页面总数。

#### 固定内存

表示进程使用的段中固定内存的页面总数。

#### 调页空间

表示进程使用的段在调页空间上保留或使用的页面总数。

**虚拟** 表示在进程虚拟空间中分配的页面总数。

**64 位** 表示该进程是否是 64 位进程（Y）或 32 位进程（N）。

**多线程** 表示该进程是否是多线程的（Y）或不是多线程的（N）。

#### 16 MB

表示进程的 16 MB 页面状态。“M”表示强制性，“Y”表示进程使用或已使用 16 MB 页段，而“N”表示进程不使用 16 MB 页面。

如果进程使用的页面大小不是基本的 4 KB 页面大小，那么这些统计信息后就跟有不同页面大小的分布情况。

显示进程信息之后，**svmon** 显示关于进程使用的所有段的信息。段落**段报告**描述了段的信息。

如果指定了 **-q** 标志，那么只报告使用请求大小的段的进程。注意要报告这些进程中所有的段。如果指定了进程列表，则对于每个未使用请求大小的段的进程，都会报告错误。

## 段报告

只有指定了 **-S** 标志，才显示段报告。段报告中的列标题是：

**Vsid** 表示虚拟段标识符。表示在虚拟内存管理器中的唯一段。

**Esid** 表示有效的段标识符。Esid 只有在段属于进程的地址空间时才有效。提供时，它表示进程如何使用段。如果 vsid 段被几个进程映射，但具有不同的 esid 值，那么该字段包含“-”。如果是那样的话，那么通过应用于使用该段的每个进程标识符的 **-P** 标志，可以获得精确的 esid 值。因为用于管理打开的文件或多线程结构的段不是进程的用户地址空间的一部分，所以也对这些段显示“-”。

**类型** 表示段的类型：

- 持久表示持久的段
- 工作表示工作段
- 客户表示客户段
- 映射表示映射的段
- 实映射表示实内存映射段

**描述** 给出段的文本描述。列的内容取决于段的类型和用途（请参阅段表）。

如果该段是持久的段，并且与日志无关，那么就显示设备名和相关文件的 i-node 号，两者用冒号分隔。可以通过 **ncheck** 命令或使用 **-j** 标志将设备名和 i-node 转换为文件名。如果该段是某个大文件的主段，则要在描述前预先考虑词语大文件。

**注：**将设备名和 i-node 号映射为文件名会成为大量嵌套文件系统的冗长操作。因此，应该谨慎使用 **-j** 选项。

如果该段是持久的段，并且与日志有关，那么就显示字符串 log。如果该段是工作段，那么 **svmon** 命令会试图确定该段的角色。例如，特殊工作段，如内核和共享库是由 **svmon** 命令识别。如果该段是进程的专用的数据段，那么会显示专用。如果该段是进程的代码段，那么响应 **-P** 标志的段报告便显示出来，然后要在描述前预先考虑字符串 code。

如果该段被几个进程映射，并且以不同方式使用（例如，被另一个进程映射为共享内存的进程专用段），那么描述是空的。通过应用于每个使用段的进程标识符的 **-P** 标志，可以获得精确的描述。

如果段的描述太大而不符合描述空间，那么可以截短描述。截短的部分可以通过给定的段的 **-S** 标志（不带 **-l**）而获得。

## 段表

### 描述表

| 段类型 | 段用途             | 描述                       |
|-----|-----------------|--------------------------|
| 持久  | 日志文件            | 日志                       |
| 持久  | 文件和目录           | 设备名: i-node 号            |
| 持久  | 大文件             | 大文件设备名: i-node 号         |
| 映射  | 文件映射            | 映射到段标识符源 sid             |
|     |                 | 不再映射                     |
| 工作  | 进程和共享内存段的数据区域   | 依赖于 VSID 和 ESID 基础上的段的角色 |
| 客户  | NFS 和 CD-ROM 文件 | 依赖于 VSID 和 ESID 基础上的段的角色 |

| 段类型   | 段用途     | 描述                       |
|-------|---------|--------------------------|
| 实内存映射 | IO 空间映射 | 依赖于 VSID 和 ESID 基础上的段的角色 |

### 基于 32 位内核 vsid 的描述表

| 段类型       | 描述       |
|-----------|----------|
| 段表        | 段表       |
| 内核段       | 内核段      |
| 内核扩展段     | 内核扩展段    |
| 软件页帧表     | 页帧表      |
| ram 磁盘段   | ram 磁盘   |
| 内核段的用户盲区  | 内核盲区     |
| 调页空间磁盘映射段 | 页面空间磁盘映射 |
| 基础配置段     | 基础配置段    |
| 软件随机编码段   | 软件随机编码   |
| 文件结构段     | 杂项内核表    |
| 内核堆段      | 内核堆      |
| mbuf 池段   | mbuf 池   |
| 地址映射条目段   | 地址映射条目段  |
| 页面表区域段    | 页面表区域    |

### 基于 64 位内核 vsid 的描述表

| 段类型               | 描述              |
|-------------------|-----------------|
| 内核段               | 内核段             |
| 虚拟内存管理器专用段        | 虚拟内存管理器数据段      |
| 段表段               | 虚拟内存管理器 scb 段   |
| 页面表区域段            | 页面表区域           |
| 调页空间磁盘映射段         | 页面空间磁盘映射        |
| 软件随机编码段           | vmm 软件随机编码      |
| 实内存堆段             | vmm rmalloc 堆   |
| 地址映射条目段           | vmm ame 段       |
| vmm vmintervals 段 | vmm vmintervals |
| 其他 vmm 段          | 其他 vmm 段        |
| 过程和线程表            | 过程和线程表          |
| mbuf 池段           | mbuf 池          |
| 装入程序段             | 装入程序段           |
| jfs 段             | jfs 段           |
| lfs 段             | lfs 段           |
| 锁检测段              | 锁检测             |
| 内核堆段              | 内核堆             |
| 堆调试段              | 堆调试             |
| mpdata 调试段        | mpdata 调试       |
| 其他内核段             | 其他内核段           |

### 基于 32 位内核上 32 位进程的 esid 的描述表

| ESID 值或范围 | 描述   |
|-----------|------|
| 0x0       | 内核   |
| 0x1       | 代码   |
| 0x2       | 进程专用 |

| ESID 值或范围 | 描述                                                                                                                                                                                                                                                                        |
|-----------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 0x3-0xC   | <ul style="list-style-type: none"> <li>• 未使用段</li> <li>• 共享内存段</li> <li>• 映射文件</li> <li>• 映射的文件读写</li> <li>• 延迟的更新</li> <li>• mmap 段</li> <li>• 工作存储器</li> <li>• 实内存映射区域段</li> <li>• 其他段</li> <li>• 扩展的共享内存段</li> <li>• 文本或共享库代码段</li> <li>• 溢出堆</li> <li>• 未知</li> </ul> |
| 0xD       | 共享库文本                                                                                                                                                                                                                                                                     |
| 0xE       | 共享库 ovfl                                                                                                                                                                                                                                                                  |
| 0xF       | 共享库数据                                                                                                                                                                                                                                                                     |

### 基于 32 位内核上 64 位进程的 esid 的描述表

| ESID 值或范围               | 描述                                                                                                                                                                                                                                                                        |
|-------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 0x0                     | 内核                                                                                                                                                                                                                                                                        |
| 0xD                     | 共享库文本                                                                                                                                                                                                                                                                     |
| 0x10 - 0x6FFFFFFF       | 文本数据 BSS 堆                                                                                                                                                                                                                                                                |
| 0x70000000 - 0x7FFFFFFF | <ul style="list-style-type: none"> <li>• 未使用段</li> <li>• 共享内存段</li> <li>• 映射文件</li> <li>• 映射的文件读写</li> <li>• 延迟的更新</li> <li>• mmap 段</li> <li>• 工作存储器</li> <li>• 实内存映射区域段</li> <li>• 其他段</li> <li>• 扩展的共享内存段</li> <li>• 文本或共享库代码段</li> <li>• 溢出堆</li> <li>• 未知</li> </ul> |
| 0x80000000 - 0x8FFFFFFF | 专用负载                                                                                                                                                                                                                                                                      |
| 0x90000000 - 0x90010009 | 共享库文本                                                                                                                                                                                                                                                                     |
| 0x9001000A - 0x90020013 | 共享库数据                                                                                                                                                                                                                                                                     |
| 0x90020014 - 0x9FFFFFFE | 共享库                                                                                                                                                                                                                                                                       |
| 0x9FFFFFFF              | USLA                                                                                                                                                                                                                                                                      |
| 0xA0000000 - 0xEFFFFFFF | 保留供系统使用                                                                                                                                                                                                                                                                   |
| 0xF0000000 - 0xFFFFFFFF | 应用程序堆栈                                                                                                                                                                                                                                                                    |
| 0xF0000002              | 进程专用                                                                                                                                                                                                                                                                      |

## 基于 64 位内核上 32 位进程的 esid 的描述表

| ESID 值或范围               | 描述                                                                                                                                                                                                                                                          |
|-------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 0x0                     | 内核                                                                                                                                                                                                                                                          |
| 0x1                     | 代码                                                                                                                                                                                                                                                          |
| 0x2                     | 进程专用                                                                                                                                                                                                                                                        |
| 0x3-0xC                 | <ul style="list-style-type: none"><li>• 未使用段</li><li>• 共享内存段</li><li>• 映射文件</li><li>• 映射的文件读写</li><li>• 延迟的更新</li><li>• mmap 段</li><li>• 工作存储器</li><li>• 实内存映射区域段</li><li>• 其他段</li><li>• 扩展的共享内存段</li><li>• 文本或共享库代码段</li><li>• 溢出堆</li><li>• 未知</li></ul> |
| 0xD                     | 共享库文本                                                                                                                                                                                                                                                       |
| 0xE                     | <ul style="list-style-type: none"><li>• 未使用段</li><li>• 共享内存段</li><li>• 映射文件</li><li>• 映射的文件读写</li><li>• 延迟的更新</li><li>• mmap 段</li><li>• 工作存储器</li><li>• 实内存映射区域段</li><li>• 其他段</li><li>• 扩展的共享内存段</li><li>• 文本或共享库代码段</li><li>• 溢出堆</li><li>• 未知</li></ul> |
| 0xF                     | 共享库数据                                                                                                                                                                                                                                                       |
| 0x9FFFFFFD0-0x9FFFFFFDF | 32 位用户别名                                                                                                                                                                                                                                                    |
| 0x9FFFFFFF0-0x9FFFFFFF  | 32 位装入程序别名                                                                                                                                                                                                                                                  |

## 基于 64 位内核上 64 位进程的 esid 的描述表

| ESID 值或范围 | 描述   |
|-----------|------|
| 0x0       | 内核   |
| 0x2       | 进程专用 |

## ESID 值或范围

0x3-0xC

## 描述

- 未使用段
- 共享内存段
- 映射文件
- 映射的文件读写
- 延迟的更新
- mmap 段
- 工作存储器
- 实内存映射区域段
- 其他段
- 扩展的共享内存段
- 文本或共享库代码段
- 溢出堆

0xE

- 未知
- 未使用段
- 共享内存段
- 映射文件
- 映射的文件读写
- 延迟的更新
- mmap 段
- 工作存储器
- 实内存映射区域段
- 其他段
- 扩展的共享内存段
- 文本或共享库代码段
- 溢出堆
- 未知

0x10-0x6FFFFFFF

文本数据 BSS 堆

0x70000000-0x7FFFFFFF

缺省 shmat/mmap

0x80000000-0x8FFFFFFF

专用负载

0x90000000-0x9FFFFFFF

共享库文本

0xF0000000-0xFFFFFFFF

应用程序堆栈

**页大小** 表示段内页面的大小。页面大小以字母表示（例如，**s** 表示 4 KB 的页面，**L** 表示 16 MB 的页面）。

## 正在使用

表示段中的实内存中的页面数。

## 固定内存

表示该段中固定内存的页面数。

## 调页空间

表示该段的调页空间上页面数。该字段只与工作段相关。

**虚拟** 表示分配给段的虚拟空间的页面数。（只分配给工作段。）

如果指定了 **-r** 标志，那么显示已分配段页面内的范围。因为页面可从两端开始然后向中间移动而得到分配，所以工作段可能具有两个范围。

如果指定了 **-l** 标志，那么显示使用段的进程标识符列表。请参阅关于特殊段处理的 **-l** 标志的描述。

如果指定了 **-q** 标志，那么只报告具有请求大小的段。如果指定了段列表，则对于具有不同大小的每个段，都会报告错误。

注：只能为主段生成段报告。

## 详细报告

只有指定了 **-D** 标志，才显示详细报告。

在所使用的页面列表之前会显示几个字段：

**Segid** 段标识

**类型** 段的类型

**页大小** 表示段内页面的大小。

**地址寻址**

段所使用的帧的范围。

对于工作段，还会显示以下附加的字段：

**页面空间分配的大小**

调出的页面数。

**虚拟** 段使用的页面数。

**正在使用**

段使用的帧数。

详细报告中的列标题。

段报告描述了相同的段信息，此外还对每个帧进行了描述：

**页面** 相对于虚拟空间的页码。如果虚拟空间比单个段（大文件）更大，那么此页面数可能比段内的帧数（65535）更多。

**帧** 实内存中的帧号。由于无论段的页面大小是多少，帧的大小总是被视为 4 KB，那么对于任何大于 4 KB 的页面大小，都表示与页面相关的是帧的范围而不是单个帧。此范围被注明为 `XXXXXXXX..YYYYYYYY`，这表示用于给定页面的帧的连续范围，标识是从 `XXXXXXXX` 至 `YYYYYYYY`。

**固定内存**

表示帧是否带固定内存

**引用** 表示进程是否已经引用帧（只有 **-b** 标志）。

**修改** 表示进程是否已经修改帧（只有 **-b** 标志）。

**ExtSegid**

扩展的段标识符。页码比段中最大的帧号高时，才会设置该字段。

**ExtPage**

扩展的页码。页码比段中的最大帧号高时才会设置该字段，并且表示扩展的段内的页码。

注：只能为主段生成详细报告。

## 帧报告

只有指定了 **-F** 标志，才显示帧报告。

没有指定实参时，如果已定义，帧报告会返回保留池的使用百分比（例如 16 MB 页面池）。报告的数目仅考虑设置了访问位的帧。在此处理过程中，会重新设置所有的访问位。这样，第二次使用此命令时，它会报告自上次使用此命令以来所访问的实内存的百分比。这有助于确定机器的活动强度。例如：

```
# svmon -F -i 10 3
Processing.. 100%
percentage of memory used: 73.85%
percentage of 16MB page memory used: 0.00%
Processing.. 100%
percentage of memory used: 12.84%
percentage of 16MB page memory used: 0.00%
Processing.. 100%
percentage of memory used: 11.25%
percentage of 16MB page memory used: 0.00%
```

注：此处理过程会妨碍页面窃取进程（LRU）活动（它也使用访问位来确定要调出的页面）。

指定帧号时，报告中的列标题是：

帧 实内存中帧号。

#### 段标识符

表示帧所属的虚拟段标识符（如果是扩展段，则为主段标识符）。

引用 表示进程是否已经引用帧。

修改 表示进程是否已经修改帧。

#### 固定内存数

表示帧的长期固定内存数和短期固定内存数。

状态 表示帧的状态（损坏、正在使用、空闲、I/O、PgAhead、隐藏）。

软件位 表示软件页帧表的帧状态。

#### ExtSegid

扩展的段标识符。只有在帧属于扩展段时才设置该字段。

页大小 表示帧所属的页面的大小。页面大小以字母表示（例如，**s** 表示 4 KB 的页面，**L** 表示 16 MB 的页面）。

注：为了生成帧所属的段的段报告或详细报告，会始终使用段标识符，这是因为扩展段不能生成那些报告。

## 标志

如果没有给出命令行标志，那么 **-G** 标志是缺省值。

|                                      |                                                                                                           |
|--------------------------------------|-----------------------------------------------------------------------------------------------------------|
| <b>-a</b> <i>SupClassName</i>        | 将范围限于 <i>SupClassName</i> 类参数的子类（在层报告 <b>-T</b> 中）。参数是超类名。不支持类列表。                                         |
| <b>-b</b>                            | 显示所有显示帧的引用和修改位的状态（详细报告 <b>-D</b> ）。显示时，重新设置帧的引用位。使用 <b>-i</b> 标志时，会检测在每个间隔间存取的帧。<br>注：因为该标志会影响性能，所以请谨慎使用。 |
| <b>-c</b>                            | 表示统计信息中只包含客户段。缺省情况下会分析所有段。                                                                                |
| <b>-C</b> <i>Command1...CommandN</i> | 显示运行命令名 <i>Command1...CommandN</i> 的进程的内存使用统计信息。 <i>Commandn</i> 是字符串。它是可执行文件精确的基本名。                      |
| <b>-d</b>                            | 对于给定实体，显示属于该实体（用户名或命令名）的进程的内存统计信息。                                                                        |
| <b>-D</b> <i>SID1...SIDN</i>         | 显示段 <i>SID1...SIDN</i> 的内存使用统计信息和每个段所有帧的详细状态。指定的段标识符必须关于主段。                                               |



|                                                   |                                                                                                                                                                                                    |
|---------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-e</b>                                         | 显示 <i>Class</i> 参数的子类的内存使用统计信息（该参数在工作负载类报告 <b>-W</b> 和层报告 <b>-T</b> 中）。 <b>-W</b> 或 <b>-a</b> 类参数需要是超类名称。                                                                                          |
| <b>-f</b>                                         | 表示统计信息中只包含持久段（文件）。缺省情况下会分析所有段。                                                                                                                                                                     |
| <b>-F</b> [ <i>Frame1...FrameN</i> ]              | 显示帧 <i>Frame1...FrameN</i> 状态，包含其所属的段。如果没有供应帧列表，那么会显示所用内存的百分比。                                                                                                                                     |
| <b>-g</b>                                         | 表示要显示的信息按照在调页空间上保留或使用的页面总数的降序排列。这个标志结合段报告改变了排序列表末尾的非工作段。                                                                                                                                           |
| <b>-G</b>                                         | 显示全局报告。                                                                                                                                                                                            |
| <b>-i</b> <i>Interval</i> [ <i>NumIntervals</i> ] | 指示 <b>svmon</b> 命令重复显示统计信息。每个 <i>Interval</i> 秒就收集和显示统计信息。 <i>NumIntervals</i> 是重复的次数；如果未指定，那么 <b>svmon</b> 一直运行到用户使用 <b>Ctrl-C</b> 中断为止。<br><b>注：</b> 因为需要花费几秒来收集某些选项的统计信息，所以观察到的时间间隔可能比指定的时间间隔大。 |
| <b>-j</b>                                         | 显示每个持久段引用的文件路径。<br><b>注：</b> 因为该标志会对性能产生潜在的影响（特别是 <b>svmon -S</b> ），所以请谨慎使用。                                                                                                                       |
| <b>-l</b>                                         | 每个显示的段都会显示使用段的进程标识符列表，并且根据报告类型显示进程所属的实体名（登录、命令、层或类）。对于特别的段，会显示标号而不是进程标识符列表。<br><b>系统段</b> 显示标志为 <i>system</i> 的段的标号。<br><b>未使用段</b><br>此标号将显示所有的现有进程都未使用的段。例如，与文件相关的持久段不再使用。                       |
|                                                   | <b>共享库文本</b><br>显示包含共享库文本的段的标号，可能大部分进程都使用该共享库文本（例如， <b>libc.a</b> ）。这将阻止显示进程的长列表。                                                                                                                  |
| <b>-m</b>                                         | 段映射源段时，显示关于源段的信息并映射段。缺省值是只显示关于映射段的信息。                                                                                                                                                              |
| <b>-n</b>                                         | 表示统计信息中只包含非系统段。缺省情况下会分析所有段。                                                                                                                                                                        |
| <b>-p</b>                                         | 表示将要显示的信息按照带固定内存的页面总数降序排列。                                                                                                                                                                         |
| <b>-P</b> [ <i>PID1... PIDN</i> ]                 | 显示进程 <i>PID1...PIDN</i> 的内存使用统计信息。 <i>PID</i> 是一个十进制的值。如果没有供应进程标识符（ <i>PID</i> ）列表，那么就会显示所有活动进程的内存使用统计信息。                                                                                          |
| <b>-q</b> [ <i>s   L</i> ]                        | 无论它们处理的页面是否为请求大小，都会过滤结果。通过子实参选项来指定请求的页面大小。使用 <b>s</b> 指定 4 KB 的页面，使用 <b>L</b> 指定 16 MB 的页面。如果没有指定子实参，则缺省值是 16 MB 的页面（ <b>L</b> ）。此时还会显示每个页面大小的度量值。                                                 |
| <b>-r</b>                                         | 显示已分配的段页面内的范围。因为页面可从两端开始然后向中间移动而得到分配，所以工作段可能具有两个范围。                                                                                                                                                |
| <b>-s</b>                                         | 表示统计信息中只包含系统段。缺省情况下会分析所有段。                                                                                                                                                                         |
| <b>-S</b> [ <i>SID1...SIDN</i> ]                  | 显示段 <i>SID1...SIDN</i> 的内存使用统计信息。 <i>SID</i> 是一个十六进制的值。指定的段标识符必须关于主段。如果没有供应段标识符（ <i>SID</i> ）列表，那么就会显示所有定义的段的内存使用统计信息。                                                                             |
| <b>-t</b> <i>Count</i>                            | 显示要显示的最高 <i>Count</i> 对象的内存使用统计信息                                                                                                                                                                  |
| <b>-T</b> [ <i>Tier1...TierN</i> ]                | 显示所有层号 <i>Tier1...TierN</i> 的类的内存使用统计信息。如果没有供应层列表，那么就会显示所有定义的层的内存使用统计信息。                                                                                                                           |
| <b>-u</b>                                         | 表示要显示的信息按照实内存的页面总数降序排列。如果无下列标志出现，这就是缺省的排序标准： <b>-p</b> 、 <b>-g</b> 和 <b>-v</b> 。                                                                                                                   |
| <b>-U</b> [ <i>LogName1...LogNameN</i> ]          | 显示登录名 <i>LogName1...LogNameN</i> 的内存使用统计信息。 <i>Logname</i> 是一个字符串，它是精确的登录名。如果没有提供登录标识符列表，那么就会显示所有定义的登录标识符的内存使用统计信息。                                                                                |

|                                    |                                                                                                                                                                   |
|------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-v</b>                          | 表示要显示的信息按照虚拟空间的页面总数降序排列。这个标志结合段报告改变了排序列表末尾的非工作段。                                                                                                                  |
| <b>-w</b>                          | 表示统计信息中只包含工作段。缺省情况下会分析所有段。                                                                                                                                        |
| <b>-W</b> [ <i>Clmn1...ClmnN</i> ] | 显示工作负载管理类 <i>Clmn1...ClmnN</i> 的内存使用统计信息。 <i>Clmn</i> 是一个字符串。它是一个类的精确名称。对于一个子类来说，名称应该具有 <i>superclassname.subclassname</i> 的格式。如果没有提供类名列表，那么就会显示所有定义的类名的内存使用统计信息。 |
| <b>-x</b>                          | 显示层报告 <b>-T</b> 中对某个层的每个类的段的内存使用统计信息。                                                                                                                             |
| <b>-z</b>                          | 显示在执行 <b>svmon</b> 过程中动态分配的最大内存大小。                                                                                                                                |

## 安全性

访问控制: 您必须具有 root 用户权限以运行该命令。

## 示例

1. 要显示全局统计信息，请输入：

```
svmon -G
svmon -G
```

|         |        |       |        |       |       |
|---------|--------|-------|--------|-------|-------|
|         | 页大小    | 正在使用  | 空闲     | 固定内存  | 虚拟    |
| 内存      | 786432 | 99980 | 686452 | 61545 | 86822 |
| 页空间     | 16384  | 1084  |        |       |       |
|         | 工作     | 持久    | 客户机    |       |       |
| 固定内存    | 53353  | 0     | 0      |       |       |
| 正在使用    | 70261  | 21527 | 0      |       |       |
| 页大小     | 池大小    | 正在使用  | 调页空间   | 固定内存  | 虚拟    |
| s 4 KB  | -      | 83596 | 1084   | 45161 | 78630 |
| L 16 MB | 4      | 2     | 0      | 4     | 2     |

系统的内存大小是 786432 个帧（即  $54524277 \times 4 \times 1024$  字节，或 208 GB）。这个大小分成正在使用的帧（99980 帧）和空闲帧（686452 帧）。

61545 帧固定内存固定于内存中。

在系统虚拟空间中分配的页面数是 86822。

正在使用的值和固定内存值包括为 16 MB 页面内存池保留的页面。

调页空间的大小是 16384 个页面（即  $16384 \times 4 \times 1024$  字节，或 64 MB）。

所用的调页空间页面数是 1084。

固定帧（61545 个帧）由工作段固定页面（53353 个帧）组成，但未计入由 16 MB 页池固定但未使用的内存。

包含页面的帧的数目（99980 个帧）由工作段页面（70261 个帧）和持久段页面（21527 个帧）组成，但未计入仅保留在 16 MB 池中但已计为正在使用的内存。

然后会显示可以在系统上使用的每个页面大小的统计信息。例如，16 MB 页池由 4 个 16 MB 的页面组成。使用了两个这样的页面，这些页面都不在调页空间中（因为它们全都是固定的），所有这些页面都是固定的，且有两个页面在系统的虚拟空间中。

其值以页大小为单位来表示。16 MB 的页池相当于 16384 个 4 KB 的页面。

2. 要显示 1/2 小时中每分钟的全局统计信息，请输入：

```
svmon -G -i 60 30
```

3. 要显示用户 *steve* 的内存统计信息，请输入：

```
svmon -U steve
```

```
=====
用户
steve                正在使用    固定内存    调页空间    虚拟
                    16309      11269      1083        16304
  页大小    正在使用    固定内存    调页空间    虚拟
  s    4 KB    8117      3077      1083      8112
  L    16 MB    2         2         0         2
-----
系统段                正在使用    固定内存    调页空间    虚拟
                    5828      3074      1083        5823
  页大小    正在使用    固定内存    调页空间    虚拟
  s    4 KB    5828      3074      1083      5823
  L    16 MB    0         0         0         0
  Vsid    Esid 类型 描述    页大小 正在使用 固定内存 调页空间 虚拟
  0       0 工作 内核段    s      5823    3074    1083    5823
-----
互斥段                正在使用    固定内存    调页空间    虚拟
                    8686      8195      0           8686
  页大小    正在使用    固定内存    调页空间    虚拟
  s    4 KB    494       3         0         494
  L    16 MB    2         2         0         2
  Vsid    Esid 类型 描述    页大小 正在使用 固定内存 调页空间 虚拟
  4d093  70000000 工作 缺省 shmat/mmap    L      2       2       0       2
  5ceb7    2 工作 进程专用    s      494    3       0       494
  44ed1    - 持久 /dev/hd9var:4228    s       0     0       -       -
  64e79    - 持久 /dev/hd9var:37     s       0     0       -       -
-----
共享段                正在使用    固定内存    调页空间    虚拟
                    1795      0         0           1795
  页大小    正在使用    固定内存    调页空间    虚拟
  s    4 KB    1795      0         0         1795
  L    16 MB    0         0         0         0
  Vsid    Esid 类型 描述    页大小 正在使用 固定内存 调页空间 虚拟
  289aa    d 工作 共享库文本    s      1795    0       0       1795
-----
```

用户 *steve* 具有 16309 页的实内存。11269 帧有固定内存。

在调页空间上保留或使用的页面数是 1083。虚拟空间中已分配的页面数是 16304。

在用户 *steve* 使用的页面中，有 2 个页面属于 16 MB 的页池。

用户 *steve* 运行的进程所使用的段分成三个类别：

- 系统：所有进程共享的段
- 互斥：属于 *steve* 的进程集所使用的段
- 共享：几个用户共享的段。

用户 *steve* 的全局统计信息是这三个类别（系统、互斥和共享）中的每个字段（正在使用、固定内存、调页空间以及虚拟）的和。

4. 要显示出只考虑工作段的用户 *root* 和 *steve* 的内存使用统计信息，请输入：

```
svmon -U root steve -w
```

5. 要显示出调页空间的最高的 10 个用户，请输入：

```
svmon -U -g -t 10
```

6. 要显示出用户 *steve* 的内存使用统计信息，包含进程标识符列表，请输入：

```
svmon -U steve -l
```

```

=====
用户                               正在使用  固定内存  调页空间  虚拟
steve                               16309    11269    1083     16244
  页大小  正在使用  固定内存  调页空间  虚拟
  s  4 KB    8117    3077    1083    8052
  L 16 MB    2        2        0        2
-----
系统段                               正在使用  固定内存  调页空间  虚拟
                               5828    3074    1083    5823
  页大小  正在使用  固定内存  调页空间  虚拟
  s  4 KB    5828    3074    1083    5823
  L 16 MB    0        0        0        0
  Vsid   Esid  类型  描述  页大小  正在使用  固定内存  调页空间  虚拟
  0      0    工作  内核段  s      5823    3074    1083    5823
-----
互斥段                               正在使用  固定内存  调页空间  虚拟
                               8686    8195    0     8686
  页大小  正在使用  固定内存  调页空间  虚拟
  s  4 KB    494      3        0     494
  L 16 MB    2        2        0        2
  Vsid   Esid  类型  描述  页大小  正在使用  固定内存  调页空间  虚拟
  4d093 70000000 工作  缺省 shmat/mmap  L      2        2        0        2
  5ceb7 2        工作  进程专用  s     494      3        0     494
  44ed1 - 持久  /dev/hd9var:4228 s      0        0        -        -
  64e79 - 持久  /dev/hd9var:37  s      0        0        -        -
-----
共享段                               正在使用  固定内存  调页空间  虚拟
                               1795    0        0     1735
  页大小  正在使用  固定内存  调页空间  虚拟
  s  4 KB    1795    0        0     1735
  L 16 MB    0        0        0        0
  Vsid   Esid  类型  描述  页大小  正在使用  固定内存  调页空间  虚拟
  289aa d 工作  共享库文本  s     1735    0        0     1735
  8050  1 持久  代码, /dev/hd2:4204 s      58        0        -        -
      pid:12374  用户: steve
      pid:11194  用户: root
      pid:10870  用户: root
      pid:10574  用户: root
      pid:10000  用户: root
      pid:6544   用户: root
  1863 - 持久  /dev/hd2:12837 s      2        0        -        -
      pid:12374  用户: steve
      pid:11194  用户: root
      pid:10870  用户: root
      pid:10574  用户: root
      pid:10000  用户: root
      pid:6544   用户: root
-----

```

既然指定了 **-l** 标志，则进程标识符和使用进程标识符的用户名的列表跟随“共享”段类别的每个段。

- 要显示出用户 *steve* 的内存使用统计信息，包含属于 *steve* 的进程，请输入：

```
svmon -U steve -d
```

```

=====
用户                               正在使用  固定内存  调页空间  虚拟
steve                               16309    11269    1083    16304
  页大小  正在使用  固定内存  调页空间  虚拟
  s  4 KB    8117    3077    1083    8112
  L 16 MB    2        2        0        2
-----
  Pid 命令 正在使用  固定内存  调页空间  虚拟  64 位  多线程  16 MB
  1032304 shm_lgpg64 16309    11269    1083 16304  Y    N    Y
-----
系统段                               正在使用  固定内存  调页空间  虚拟
                               5828    3074    1083    5823
  页大小  正在使用  固定内存  调页空间  虚拟
  s  4 KB    5828    3074    1083    5823

```

| L     | 16 MB    | 0    | 0                | 0    | 0    | 0    | 0    | 0    | 0 |
|-------|----------|------|------------------|------|------|------|------|------|---|
| Vsid  | Esid     | 类型   | 描述               | 页大小  | 正在使用 | 固定内存 | 调页空间 | 虚拟   |   |
| 0     | 0        | 工作   | 内核段              | s    | 5823 | 3074 | 1083 | 5823 |   |
| 互斥段   |          |      |                  |      | 正在使用 | 固定内存 | 调页空间 | 虚拟   |   |
|       |          |      |                  |      | 8686 | 8195 | 0    | 8686 |   |
| 页大小   |          | 正在使用 | 固定内存             | 调页空间 | 虚拟   |      |      |      |   |
| s     | 4 KB     | 494  | 3                | 0    | 494  |      |      |      |   |
| L     | 16 MB    | 2    | 2                | 0    | 2    |      |      |      |   |
| Vsid  | Esid     | 类型   | 描述               | 页大小  | 正在使用 | 固定内存 | 调页空间 | 虚拟   |   |
| 4d093 | 70000000 | 工作   | 缺省 shmat/mmap    | L    | 2    | 2    | 0    | 2    |   |
| 5ceb7 | 2        | 工作   | 进程专用             | s    | 494  | 3    | 0    | 494  |   |
| 44ed1 | -        | 持久   | /dev/hd9var:4228 | s    | 0    | 0    | -    | -    |   |
| 64e79 | -        | 持久   | /dev/hd9var:37   | s    | 0    | 0    | -    | -    |   |
| 共享段   |          |      |                  |      | 正在使用 | 固定内存 | 调页空间 | 虚拟   |   |
|       |          |      |                  |      | 1795 | 0    | 0    | 1795 |   |
| 页大小   |          | 正在使用 | 固定内存             | 调页空间 | 虚拟   |      |      |      |   |
| s     | 4 KB     | 1795 | 0                | 0    | 1795 |      |      |      |   |
| L     | 16 MB    | 0    | 0                | 0    | 0    |      |      |      |   |
| Vsid  | Esid     | 类型   | 描述               | 页大小  | 正在使用 | 固定内存 | 调页空间 | 虚拟   |   |
| 289aa | d        | 工作   | 共享库文本            | s    | 1795 | 0    | 0    | 1795 |   |

因为指定了 **-d** 标志，全局统计信息后跟有关于 **steve** 运行的所有进程的信息（在这种情况下，仅针对 **shm\_lgpg64** 命令）。

8. 要显示 **login** 命令的内存使用情况统计信息（包括地址范围），请输入：

```
svmon -C login -r
```

```
=====
```

| 命令    | 正在使用 | 固定内存 | 调页空间            | 虚拟   |                            |      |      |      |  |
|-------|------|------|-----------------|------|----------------------------|------|------|------|--|
| login | 7092 | 3130 | 1083            | 6844 |                            |      |      |      |  |
| ===== |      |      |                 |      |                            |      |      |      |  |
| 页大小   |      | 正在使用 | 固定内存            | 调页空间 | 虚拟                         |      |      |      |  |
| s     | 4 KB | 7092 | 3130            | 1083 | 6844                       |      |      |      |  |
| ..... |      |      |                 |      |                            |      |      |      |  |
| 系统段   |      |      |                 |      | 正在使用                       | 固定内存 | 调页空间 | 虚拟   |  |
|       |      |      |                 |      | 5025                       | 3127 | 1083 | 5025 |  |
| 页大小   |      | 正在使用 | 固定内存            | 调页空间 | 虚拟                         |      |      |      |  |
| s     | 4 KB | 5025 | 3127            | 1083 | 5025                       |      |      |      |  |
| Vsid  | Esid | 类型   | 描述              | 页大小  | 正在使用                       | 固定内存 | 调页空间 | 虚拟   |  |
| 0     | 0    | 工作   | 内核段             | s    | 5025                       | 3127 | 1083 | 5025 |  |
|       |      |      |                 |      | 地址范围: 0..23941             |      |      |      |  |
| ..... |      |      |                 |      |                            |      |      |      |  |
| 互斥段   |      |      |                 |      | 正在使用                       | 固定内存 | 调页空间 | 虚拟   |  |
|       |      |      |                 |      | 103                        | 3    | 0    | 103  |  |
| 页大小   |      | 正在使用 | 固定内存            | 调页空间 | 虚拟                         |      |      |      |  |
| s     | 4 KB | 103  | 3               | 0    | 103                        |      |      |      |  |
| Vsid  | Esid | 类型   | 描述              | 页大小  | 正在使用                       | 固定内存 | 调页空间 | 虚拟   |  |
| 3ad9d | 2    | 工作   | 进程专用            | s    | 54                         | 3    | 0    | 54   |  |
|       |      |      |                 |      | 地址范围: 0..35 : 65310..65535 |      |      |      |  |
| 3cd9e | f    | 工作   | 共享库数据           | s    | 49                         | 0    | 0    | 49   |  |
|       |      |      |                 |      | 地址范围: 0..1080              |      |      |      |  |
| ..... |      |      |                 |      |                            |      |      |      |  |
| 共享段   |      |      |                 |      | 正在使用                       | 固定内存 | 调页空间 | 虚拟   |  |
|       |      |      |                 |      | 1964                       | 0    | 0    | 1716 |  |
| 页大小   |      | 正在使用 | 固定内存            | 调页空间 | 虚拟                         |      |      |      |  |
| s     | 4 KB | 1964 | 0               | 0    | 1716                       |      |      |      |  |
| Vsid  | Esid | 类型   | 描述              | 页大小  | 正在使用                       | 固定内存 | 调页空间 | 虚拟   |  |
| 149aa | d    | 工作   | 共享库文本           | s    | 1716                       | 0    | 0    | 1716 |  |
|       |      |      |                 |      | 地址范围: 0..2800              |      |      |      |  |
| 3eb9f | -    | 持久   | /dev/hd2:202789 | s    | 194                        | 0    | -    | -    |  |
|       |      |      |                 |      | 地址范围: 0..947               |      |      |      |  |
| 10ba8 | -    | 持久   | /dev/hd2:202790 | s    | 36                         | 0    | -    | -    |  |
|       |      |      |                 |      | 地址范围: 0..35                |      |      |      |  |

```

d80      1 持久 代码, /dev/hd2:297192 s          15          0          -          -
          地址范围: 0..14
24c32    - 持久 /dev/hd4:8200          s           3           0          -          -
          地址范围: 0..2

```

login 命令使用了实内存中的 7092 个页面。3130 帧带有固定内存。

在调页空间上保留或使用的页面数是 1083。

分配在虚拟空间中的页面数是 6844。login 命令不使用 4 KB 页面以外的其他页面。

运行 login 命令的进程所使用的段分为三个类别:

- 系统: 所有进程共享的段
- 互斥: 运行 login 命令的进程集所使用的段
- 共享: 几个用户共享的段

login 命令的全局统计信息是这三种类别 (系统、互斥和共享) 的每个字段 (正在使用、固定内存、调页空间和虚拟) 之和。

每个段的地址范围在其统计信息后显示。

9. 要显示命令 oracle、xemacs 以及 cc 的内存使用统计信息, 请输入:

```
svmon -C oracle xemacs cc
```

10. 要显示出超类 System 的内存使用统计信息, 请输入:

```
svmon -W System
```

```

=====
超类
System          正在使用  固定内存  调页空间  虚拟
                2111      465      1128     2394
VsId   Esid  类型  描述          页大小  正在使用  固定内存  调页空间  虚拟
700e   - 工作  内核固定堆    s         1235     460     1128    1638
af90   - 工作                s          391         1         0       391
9f76   - 工作                s          223         0         0       214
a771   - 工作                s           67         1         0       67
  80    - 持久  /dev/hd2:3    s           61         0         -         -
cf7c   - 工作                s           47         0         0        35
f79b   - 工作                s           21         0         0        14
9052   - 持久  /dev/hd3:4    s           17         0         -         -
37a3   - 工作                s           15         2         0       15
  7a5   - 工作                s           15         0         0         8
9797   - 工作                s           12         1         0        12
52aa   - 持久  /dev/hd2:4196 s            3         0         -         -
786f   - 持久  日志          s            2         0         -         -
ca1a   - 持久  /dev/hd2:12817 s            1         0         -         -
89b1   - 持久  /dev/hd4:761  s            1         0         -         -

```

类 System 使用 2111 页实内存。465 帧带有固定内存。

在调页空间上保留或使用的页面数是 1128。

分配在虚拟空间中的页面数是 2394。

然后会显示属于类 System 的每个段的信息。

由于类只使用了 4 KB 页面, 所以不会显示不同页面大小的分布情况。

11. 要显示出超类 default 和 developer 的内存使用统计信息, 请输入:

```
svmon -W default developer
```

12. 要显示出超类 Mysupclass 及其子类的内存使用统计信息, 请输入:

```
svmon -W Mysupclass -e
```

```

=====
超类                                正在使用    固定内存    调页空间    虚拟
Mysupclass                          1128        14          0          596
  页大小    正在使用    固定内存    调页空间    虚拟
  s    4 KB    1128        14          0          596
=====
类                                正在使用    固定内存    调页空间    虚拟
Mysupclass.Default                    1128        14          0          596
  页大小    正在使用    固定内存    调页空间    虚拟
  s    4 KB    1128        14          0          596
  Vsid    Esid    类型    描述    页大小    正在使用    固定内存    调页空间    虚拟
  f9bf    -    持久    /dev/hd2:6386    s    123        0          -          -
  6fcd    -    工作                    s    81        2          0          79
  d1da    -    工作                    s    79        2          0          77
  260    -    工作                    s    77        2          0          75
  ea5d    -    工作                    s    77        2          0          75
  3fe7    -    工作                    s    66        2          0          64
  d7da    -    工作                    s    66        2          0          64
  87d0    -    工作                    s    66        2          0          64
  3867    -    持久    /dev/hd2:4205    s    57        0          -          -
  522a    -    持久    /dev/hd2:4392    s    47        0          -          -
  1a23    -    持久    /dev/hd2:24764    s    42        0          -          -
=====
类                                正在使用    固定内存    调页空间    虚拟
Mysupclass.Shared                      0          0          0          0
  页大小    正在使用    固定内存    调页空间    虚拟
  s    4 KB    0          0          0          0
=====
类                                正在使用    固定内存    调页空间    虚拟
Mysupclass.subclass                    0          0          0          0
  页大小    正在使用    固定内存    调页空间    虚拟
  s    4 KB    0          0          0          0

```

当指定 **-e** 标志时，就会报告超类 **Mysupclass** 及其子类的统计信息。

13. 要显示所有定义的层的内存使用情况，请输入：

```

svmon -T
=====
层                                正在使用    固定内存    调页空间    虚拟
0                                58740        28140        1083        46875
  页大小    正在使用    固定内存    调页空间    虚拟
  s    4 KB    50548        19948        1083        38683
  L    16 MB    2          2          0          2
=====
超类                                正在使用    固定内存    调页空间    虚拟
System                          52565        28095        1083        43705
Shared                          3097          0          0          3097
Unclassified                    3078          45          0          73
Unmanaged                        0            0          0          0
Default                          0            0          0          0
=====
层                                正在使用    固定内存    调页空间    虚拟
1                                1128          14          0          596
  页大小    正在使用    固定内存    调页空间    虚拟
  s    4 KB    1128          14          0          596
  L    16 MB    0            0          0          0
=====
超类                                正在使用    固定内存    调页空间    虚拟
Mysupclass                        1128          14          0          596

```

报告所有定义的层的所有超类。

14. 要显示出层 0 的内存使用情况，请输入：

```
svmon -T 0
```

```

=====
层                正在使用      固定内存      调页空间      虚拟
0                58740        28140        1083         46875
  页大小          正在使用      固定内存      调页空间      虚拟
s  4 KB          50548        19948        1083         38683
L  16 MB         2            2            0            2
=====

```

```

=====
超类                正在使用      固定内存      调页空间      虚拟
System             52565        28095        1083         43705
Shared             3097         0            0            3097
Unclassified       3078         45           0            73
Unmanaged           0            0            0            0
Default             0            0            0            0
=====

```

15. 要显示出超类 *Mysupclass* 的层 0 子类的内存使用情况，请输入：

```
svmon -T 0 -a Mysupclass
```

```

=====
层  超类                正在使用      固定内存      调页空间      虚拟
0  Mysupclass          1129         14           0            596
  页大小          正在使用      固定内存      调页空间      虚拟
s  4 KB          1129         14           0            596
=====

```

```

=====
类                正在使用      固定内存      调页空间      虚拟
Mysupclass.Default  1129         14           0            596
Mysupclass.Shared   0            0            0            0
Mysupclass.subclass 0            0            0            0
=====

```

报告超类 *Mysupclass* 中所有属于层 0 的子类。

16. 要显示出层 0 和层 1 的内存使用情况，包括段统计信息，请输入：

```
svmon -T 0 1 -x
```

17. 要显示出层 1 的内存使用情况，包括子类统计信息，请输入：

```
svmon -T 1 -e
```

```

=====
层                正在使用      固定内存      调页空间      虚拟
1                1129         14           0            596
  页大小          正在使用      固定内存      调页空间      虚拟
s  4 KB          1129         14           0            596
=====

```

```

=====
超类                正在使用      固定内存      调页空间      虚拟
Mysupclass          1129         14           0            596
=====

```

```

=====
类                正在使用      固定内存      调页空间      虚拟
Mysupclass.Default  1129         14           0            596
Mysupclass.Shared   0            0            0            0
Mysupclass.subclass 0            0            0            0
=====

```

报告每个属于层 1 的超类及其子类的统计信息。

18. 要显示出超类 *Mysupclass* 在层 0 中的子类的内存使用情况，包括段统计信息和进程标识符列表，请输入：

```
svmon -T 0 -a Mysupclass -x -l
```

```

=====
层  超类                正在使用      固定内存      调页空间      虚拟
0  Mysupclass          4152         4118         0            4152
  页大小          正在使用      固定内存      调页空间      虚拟
s  4 KB           56           22           0            56
L  16 MB          1            1            0            1
=====

```

```

=====
类                正在使用      固定内存      调页空间      虚拟
Mysupclass.Default  4152         4118         0            4152
  Vsid  Esid  类型  描述          页大小  正在使用  固定内存  调页空间  虚拟
  1140  70000000 工作  缺省 shmat/mmap  L        1        1        0        1
      pid:827620 层: 0 类: Mysupclass.Default
=====

```



```

a9135      - 工作          s          27          22          0          27
            未使用段
6910d 9001000a 工作 共享库文本 s          16          0          0          16
            共享库文本段
99133 8fffffff 工作 专用负载 s          5          0          0          5
            pid:827620 层: 0 类: Mysupclass.Default
7112e 80020014 工作 专用负载 s          4          0          0          4
            pid:827620 层: 0 类: Mysupclass.Default
7912f ffffffff 工作 应用程序堆栈 s          3          0          0          3
            pid:827620 层: 0 类: Mysupclass.Default
89131      11 工作 文本数据 BSS 堆 s          1          0          0          1
            pid:827620 层: 0 类: Mysupclass.Default
=====
类          正在使用      固定内存      调页空间      虚拟
Mysupclass.Shared          0          0          0          0
=====
类          正在使用      固定内存      调页空间      虚拟
Mysupclass.subclass        0          0          0          0

```

报告超类 Mysupclass 在层 0 中的所有子类的统计信息。

然后，由于在机器上定义了 16 MB 的页池，所以会显示不同页面大小的分布情况。

然后，当指定了 **-x** 时，子类的段的统计信息会跟随子类的统计信息。

最后，当指定 **-l** 时，对于每个段，就会显示使用该段的进程列表。

在此列表中，给出进程标识符的层号及其类名跟随的进程标识符。

19. 要显示出进程 278620 的内存使用统计信息，请输入：

```
svmon -P 278620
```

```

-----
Pid 命令          正在使用      固定内存      调页空间      虚拟      64 位      多线程      16 MB
278620 shm_lgpg64 13781        11319        1083        13769      Y          N          Y
 页大小          正在使用      固定内存      调页空间      虚拟
s 4 KB          5589        3127        1083        5577
L 16 MB         2          2          0          2
Vsid  Esid  类型  描述          页大小      正在使用      固定内存      调页空间      虚拟
1cf8e 70000000 工作 缺省 shmat/mmap L          2          2          0          2
0 0          工作 内核段 s          5025        3127        1083        5025
1a9ad 90000000 工作 共享库文本 s          420         0          0          420
20cf0 90020014 工作 共享库文本 s          96          0          0          96
18f8c 9001000a 工作 共享库文本 s          16          0          0          16
16d0b 9fff ffff 持久, /dev/hd2:309263 s          10         0          -          -
30af8 9fffffff 工作 s          7          0          0          7
ef87 8fffffff 工作 专用负载 s          5          0          0          5
1af8d 80020014 工作 专用负载 s          4          0          0          4
10f88 ffffffff 工作 应用程序堆栈 s          3          0          0          3
16f8b 10 持久 文本数据 BSS 堆 s          2          0          -          -
/dev/hd1:2053
14f8a 11 工作 文本数据 BSS 堆 s          1          0          0          1

```

进程 278620 正在运行 **shm\_lgpg64** 命令。它是使用 16 MB 页面的 64 位进程，不是多线程进程。

该进程使用了 13781 页的实内存。11319 帧带有固定内存。

在调页空间上保留或使用的页面数是 1083。

分配在虚拟空间中的页面数是 13769。

在这些页面内，该进程使用 2 个 16 MB 的页面（这相当于 8192 个 4 KB 的页面）。

然后会显示属于进程的每个段的信息。

段 1cf8e 使用 16 MB 页面。

20. 要显示只考虑工作和持久段的进程 6746 和 10078 的内存使用统计信息，请输入：

```
svmon -P 6746 10078 -wf
```

21. 要根据固定页面的数目打印出前 10 个进程的内存使用统计信息，请输入：

```
svmon -P -t 10 -p
```

22. 要仅考虑非系统工作段并根据虚拟页面的数目对进程进行排序来打印出所有进程的内存使用情况统计信息，请输入：

```
svmon -P -n -w -v
```

23. 要显示出段 700e 8811 和 bed7 的内存使用统计信息，请输入：

```
svmon -S 700e 8811 bed7 1cf8e
```

| Vsid  | Esid | 类型 | 描述             | 页大小 | 正在使用 | 固定内存 | 调页空间 | 虚拟   |
|-------|------|----|----------------|-----|------|------|------|------|
| 700e  | -    | 工作 | 内核固定堆          | s   | 1236 | 460  | 1128 | 1638 |
| 8811  | -    | 工作 |                | s   | 379  | 0    | 26   | 1757 |
| bed7  | -    | 持久 | /dev/hd2:41146 | s   | 0    | 0    | -    | -    |
| 1cf8e | -    | 工作 |                | L   | 2    | 2    | 0    | 2    |

显示关于列表中每个段的信息。

不提供 **Esid**，这是因为只有当该段属于进程的地址空间时该字段才是有效的。

段 700e 是内核固定堆。

段 bed7 与一个文件有关，该文件的设备是 /dev/hd2，i-node 号是 41146。

段 8811 没有特殊的特征。

段 bed7 的调页空间和虚拟字段是没有意义的（持久段）。

段 1cf8e 是包含 2 个 16 MB 页面（相当于 8192 个 4 KB 的页面）的页段。

24. 要显示段 e00e 和 15015 的内存使用统计信息，根据保留的页面空间块的数量对结果进行排序，请输入：

```
svmon -S e00e 15015 -g
```

25. 根据虚拟页面数显示最高的 5 个工作段的内存使用统计信息，请输入：

```
svmon -S -t 5 -w -v
```

26. 根据实内存的页数将系统段排序，并且显示出结果列表中最高 10 个系统段，请输入：

```
svmon -S -s -u -t 10
```

27. 要仅打印 16 MB 页面的段，请输入：

```
svmon -S -q L
```

28. 要显示段 30138 的地址范围，请输入：

```
svmon -S 30138 -r
```

| Vsid  | Esid | 类型 | 描述  | 页大小 | 正在使用  | 固定内存  | 调页空间 | 虚拟    |
|-------|------|----|-----|-----|-------|-------|------|-------|
| 30138 | -    | 工作 | 页帧表 | s   | 10752 | 10752 | 0    | 10752 |

地址范围: 0..10751 : 0..65535

29. 要显示持久段 60218 引用的文件路径，请输入：

```
svmon -S 60218 -j
```

| Vsid  | Esid | 类型  | 描述                                       | 页大小 | 正在使用 | 固定内存 | 调页空间 | 虚拟 |
|-------|------|-----|------------------------------------------|-----|------|------|------|----|
| 60218 | -    | 大文件 | /dev/hd9var:214<br>/var/tmp/hostmibd.log | s   | 328  | 0    | -    | -  |

30. 要显示出属于段 36cfb 的帧，请输入：

```
svmon -D 36cfb
段标识符: 36cfb
类型: 持久
页大小: s (4 KB)
地址范围: 0..4
```

| 页面     | 帧     | 固定内存 | 扩展段标识符 | 扩展页面 |
|--------|-------|------|--------|------|
| 0      | 62861 | N    | -      | -    |
| 1      | 70594 | N    | -      | -    |
| 2      | 24204 | N    | -      | -    |
| 3      | 46339 | N    | -      | -    |
| 4      | 23868 | N    | -      | -    |
| 122070 | 78191 | N    | 208831 | dcd6 |

段 36cfb 是具有 5 个页面的持久段。没有一个页面是固定内存的。

页面 122070 物理上是在扩展段 208831 中的页面 dcd6。

31. 要显示出属于段 36cfb 的帧和每个帧的状态位，请输入：

```
svmon -D 36cfb -b
段标识符: 36cfb
类型: 持久
页大小: s (4 KB)
地址范围: 0..5
```

| 页面 | 帧      | 固定内存 | 引用 | 修改 | 扩展段标识符 | 扩展页面 |
|----|--------|------|----|----|--------|------|
| 0  | 392948 | N    | Y  | N  | -      | -    |
| 1  | 393187 | N    | Y  | N  | -      | -    |
| 2  | 392968 | N    | Y  | N  | -      | -    |
| 3  | 392825 | N    | Y  | N  | -      | -    |
| 4  | 392890 | N    | N  | N  | -      | -    |
| 5  | 23651  | N    | Y  | N  | -      | -    |

当指定 **-b** 标志时，会显示段 36cfb 的每个帧的引用和修改位。

32. 要显示属于段 36cfb 的帧，包括执行 svmon 过程中被 svmon 动态分配的最大内存大小，请输入：

```
svmon -D 36cfb -z
段标识符: 36cfb
类型: 持久
页大小: s (4 KB)
地址范围: 0..5
```

| 页面 | 帧      | 固定内存 | 扩展段标识符 | 扩展页面 |
|----|--------|------|--------|------|
| 0  | 392948 | N    | -      | -    |
| 1  | 393187 | N    | -      | -    |
| 2  | 392968 | N    | -      | -    |
| 3  | 392825 | N    | -      | -    |
| 4  | 392890 | N    | -      | -    |
| 5  | 23651  | N    | -      | -    |

分配的最大内存 = 916

33. 要打印出在 10 秒的时间间隔中访问的内存的百分比，请输入：

```
svmon -F -i 10 2
Processing.. 100%
percentage of memory used: 73.85%
percentage of 16 MB page memory used: 50.00%
Processing.. 100%
percentage of memory used: 22.83%
percentage of 16 MB page memory used: 50.00%
```

提供分析的内存百分比以便使用户知道正在处理。

在指定时间间隔内访问的内存（即具有引用标志设置的帧）的百分比是 22.83%。

已使用的 16 MB 页面内存（即段所用的页面）的百分比是 50.00%。

34. 要显示出帧 94072 和 672914 的状态，请输入：

```
svmon -F 94072 672914
   帧      段标识  引用  修改  固定内存计数  状态      软件位  扩展段标识  页大小
   94072   e6fd      N    N      0/0      正在使用  88000004  -          s
   672914  784de     Y    Y      0/0      正在使用  88100004  250054    s
```

帧 94072 属于段 e6fd。

不能引用和修改该帧，因为它处于“正在使用”状态，它既不是关于扩展段的也不是关于大页面段的。

帧 672914 属于链接 784de 主段的扩展段 250054。

## 相关信息

**nccheck** 命令、**rmss** 命令。

《操作系统与设备管理》中的『逻辑卷存储』和『调页空间』。

《操作系统与设备管理》中的『工作负载管理』。

---

## swap 命令

### 用途

提供调页空间管理界面。

### 语法

```
swap [ -a device ] | [ -d device ] | [ -s ] | [ -l ]
```

### 描述

swap 命令提供特征显示、调页空间增加和调页空间除去的功能。

### 标志

**-a device**

激活调页空间。执行与 **swapon** 命令相同的功能。

**-d device**

取消激活调页空间。执行与 **swapoff** 命令相同的功能。

**-l**

在列表中列出调页空间区域的状态。输出有 4 列，包含以下信息：

**设备** 页面空间的路径名称。

**maj/min** (**major/minor**: 主/次)

设备的主要和次要设备号码。

**总的** 以兆字节表示的区域总大小。

**可用的** 可用空间的数量。

**-s**

显示关于总调页空间使用情况和可用性的摘要信息。输出中显示了以下信息（调页空间量以 4K 字节块为单位列出）。

**分配的** 目前分配的调页空间区域的总数。

**使用的** 目前正在使用的调页空间区域的总数。

**可用的** 可用的调页空间总数。

这些数字包括在活动的调页空间上的 **-l** 选项所列出的所有配置区域的调页空间。

**注：** 每个设备的调页空间限制为 64 GB。

## 退出状态

**0** 命令成功完成。

**>0** 发生错误。

## 示例

1. 要显示关于总调页空间的摘要信息，请输入：

```
swap -s
```

2. 要在列表中列出调页空间区域的状态，请输入：

```
swap -l
```

3. 要激活一个特定的调页空间设备 `paging01`，请输入：

```
swap -a /dev/paging01
```

## 文件

`/usr/sbin/swap`

包含 System V **swap** 命令。

## 相关信息

**chps** 命令、**lsps** 命令、**swapoff** 命令、**swapon** 命令。

---

## swapoff 命令

### 用途

取消激活一个或多个调页空间。

### 语法

```
swapoff DeviceName { DeviceName ... }
```

### 描述

**swapoff** 命令取消激活一个或多个调页空间。 *DeviceName* 指定调页空间。

**注：** 每个设备的调页空间限制为 64 GB。

要取消激活：

- 调页空间必须是先前通过 **swapon** 命令激活的。
- 在剩余的调页空间里必须具有足够的空间。剩余的页面调度设备应具有足够的空间，可容纳当前系统范围内的调页空间使用和 **npswarn** 值。

## 退出状态

| 值 | 描述                                                                                                                                                                                                                                                 |
|---|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 0 | 取消激活成功，调页状态设置为“不活动”状态。                                                                                                                                                                                                                             |
| 1 | 会显示以下消息：<br>swapoff: 不能取消激活调页空间 <i>DeviceName</i>                                                                                                                                                                                                  |
| 2 | 在剩余的调页空间里没有足够的空间，取消激活没有完成，会显示以下消息：<br>“swapoff: 不能取消激活调页空间 <i>DeviceName</i> : 文件系统没有足够的空间。”                                                                                                                                                       |
| 3 | 调页空间的用户页面出现 I/O 错误，会显示以下消息：<br>swapoff: 对调页空间 <i>DeviceName</i> 的取消激活操作被暂挂：<br>在用户支持页面遇到 I/O 错误。<br><br>建议操作： <ul style="list-style-type: none"> <li>• 检查错误日志。</li> <li>• 使用 <b>chps</b> 命令取消激活调页空间以便在下一步重新引导。</li> <li>• 重新引导系统。</li> </ul>       |
| 4 | 调页空间的系统页面发生 I/O 错误，会显示以下消息：<br>swapoff: 对调页空间 <i>DeviceName</i> 的取消激活操作被暂挂：<br>系统支持页面遇到 I/O 错误。系统可能崩溃。<br><br>建议操作： <ul style="list-style-type: none"> <li>• 检查错误日志。</li> <li>• 使用 <b>chps</b> 命令取消激活调页空间以便在下一步重新引导。</li> <li>• 重新引导系统。</li> </ul> |

## 相关信息

**lsps**、**swap**、**swapon** 和 **vmo** 命令。

《操作系统与设备管理》中的『调页空间』。

---

## swapon 命令

### 用途

激活调页空间。

### 语法

**swapon -a** | *DeviceName*

## 描述

**swapon** 命令激活调页空间。它用于早期的系统初始化过程以使初始调页空间可用。在系统初始化的后期阶段，**swapon -a** 命令用于使其他设备可用，这样页面调度和交换活动就可以在几个设备之间交错。**swapon -a** 命令使所有在 **/etc/swapspaces** 文件中指定的设备可用。调用 **swapon** 命令通常出现在系统多用户初始化 **/etc/rc** 文件中。

*DeviceName* 参数指定一个特定的设备以使其可用。如在系统交换配置表中给出的那样，第二个表格给出单独块设备。调用使这个空间和其他已定义的空间可用于系统页面调度和交换资源的分配。系统交换配置表是所有在 **/etc/swapspaces** 文件指定的设备的设置。

您可以使用基于 Web 的系统管理器 (wsm) 中的设备应用程序来运行该命令。

注：最大的活动调页空间数目为 16。此外，每个设备的调页空间限制为 64 GB。

## 标志

**-a** 使所有出现在 **/etc/swapspaces** 文件中的设备可用。

## 示例

1. 要使所有出现在 **/etc/swapspaces** 文件中的设备可用，请输入：

```
swapon -a
```

现在所有出现在 **/etc/swapspaces** 文件的设备都可用了。

2. 要使 **/dev/paging03** 和 **/dev/paging04** 设备可用于页面调度和交换，请输入：

```
swapon /dev/paging03 /dev/paging04
```

**/dev/paging03** 和 **/dev/paging04** 设备现在可用了。

## 文件

|                        |              |
|------------------------|--------------|
| <b>/etc/rc</b>         | 系统多用户初始化     |
| <b>/dev/paging</b>     | 调页和交换空间的设备条目 |
| <b>/etc/swapspaces</b> | 包含交换设备的列表。   |

## 相关信息

**mkps** 命令、**chps** 命令。

**swapon** 子例程。

《操作系统与设备管理》中的『调页空间』说明了调页空间及其分配策略。

有关安装基于 Web 的系统管理器的信息，请参阅《AIX 5L V5.3 基于 web 的系统管理器管理指南》中的『第二章：安装与系统需求』。

《操作系统与设备管理》中的『系统管理界面工具』说明了结构、主菜单和 SMIT 完成的任务。

---

## swcons 命令

### 用途

暂时将系统控制台输出重定向到一个指定的设备或文件。

### 语法

```
swcons [ -p Log_File ] [ -s Log_Size ] [ -t Tag_Verbosity ] [ -v Log_Verbosity ] PathName
```

### 描述

**swcons** 命令在系统运作的过程中暂时将系统控制台输出转换到不同的目标。该命令只将需要的系统信息、错误和干预的消息输出转换到指定的目标。**swcons** 命令不会影响系统控制台设备的操作，该设备经由 **getty** 命令提供登录。

使用该命令时指定的设备或文件保留控制台输出的目标，直到被另一个 **swcons** 命令更改，或者直到下一次启动系统，或者直到控制台驱动程序在存取指定的设备或文件时检测到错误。如果在 **swcons** 命令指定的设备或文件上检测到打开错误或写错误，那么当最后启动系统时，控制台设备驱动程序会将所有的输出转换回提供控制台支持的设备或文件。

*PathName* 参数必须是接收系统控制台消息输出的设备或文件的全限定路径名。如果 *PathName* 参数指定的文件不存在，那么 **swcons** 命令会创建此文件。如果此文件存在，那么 **swcons** 命令会将任何新的控制台消息输出附加到此文件的内容中。

注：使用 **swcons** 命令将控制台输出转换到已安装 NFS 的文件系统会导致操作系统挂起。

### 标志

|                                |                                                                                                                |
|--------------------------------|----------------------------------------------------------------------------------------------------------------|
| <b>-p</b> <i>Log_File</i>      | 指定控制台输出日志文件的全路径名。                                                                                              |
| <b>-s</b> <i>Log_Size</i>      | 指定控制台输出日志文件的大小（以字节为单位）。                                                                                        |
| <b>-t</b> <i>Tag_Verbosity</i> | 指定控制台输出标记的详细程度级别。0 禁用标记；1 至 9 启用标记。关于控制台输出记录和标记的附加信息，请参阅 <i>AIX 5L Version 5.3 Files Reference</i> 中的 控制台特殊文件。 |
| <b>-v</b> <i>Log_Verbosity</i> | 指定控制台输出记录的详细程度级别。0 禁用记录；1 至 9 启用记录。                                                                            |

### 示例

1. 要将系统控制台消息输出更改到 /tmp 目录内的名为 console.out 的文件中，请输入：  

```
swcons /tmp/console.out
```
2. 要将系统控制台消息输出更改到逻辑名为 tty3 的终端，请输入：  

```
swcons /dev/tty3
```
3. 要将系统控制台消息输出更改回在系统启动时支持控制台输出的设备或文件，请输入：  

```
swcons
```

### 文件

|                         |                        |
|-------------------------|------------------------|
| <b>/dev/console</b>     | 指定系统控制台存取的特殊文件。        |
| <b>/usr/sbin/swcons</b> | 包含 <b>swcons</b> 命令文件。 |



## 相关信息

**chcons** 命令、**lscons** 命令。

**console** 特殊文件。

---

## swts 命令

### 用途

将瘦服务器切换到不同的 COSI。

### 语法

```
swts -c Image [-t cron_time] [-v] ThinServer
```

### 描述

**swts** 命令将瘦服务器切换到不同的公共操作系统映像（COSI）。如果使用 **-t** 标志指定，则瘦服务器在被 *Time* 参数指定时将切换到新的公共映像。*Time* 的值必须是有效的 crontab 条目。请参阅 **crontab** 命令，获取有关创建有效的 cron 时间条目的信息。

**swts** 命令可以在 NIM 主服务器或瘦服务器上运行。当将瘦服务器切换到新的公共映像时，瘦服务器的 **/inst\_root** 目录中的文件将与新的公共映像同步。

### 标志

|                            |                              |
|----------------------------|------------------------------|
| <b>-c</b> <i>Image</i>     | 指定要将瘦服务器切换到的公共映像。            |
| <b>-t</b> <i>cron_time</i> | 指定 cron 条目，允许在更方便的时间切换瘦服务器。  |
| <b>-v</b>                  | 运行 <b>swts</b> 命令时启用详细的调试输出。 |

### 退出状态

|    |         |
|----|---------|
| 0  | 命令成功完成。 |
| >0 | 发生错误。   |

### 安全性

访问控制：必须具有 root 权限才能运行 **swts** 命令。

### 示例

1. 要将名为 lobo 的瘦服务器的 cosi1 公共映像切换到名为 cosi2 的公共映像，请输入：

```
swts -c cosi2 lobo
```

lobo 瘦服务器被重新初始化并且 cosi2 是其新的操作系统。

2. 要在 12 月 25 日星期天的午夜将名为 lobo 的瘦服务器的 cosi1 公共映像切换到名为 cosi2 的公共映像，请输入：

```
swts -c cosi2 -t "0 0 25 12 0" lobo
```

lobo 瘦服务器将继续使用 cosi1 公共映像，直到 12 月 25 日星期天午夜它被切换到 cosi2。

## 位置

`/usr/sbin/swts`

## 文件

`/etc/niminfo`

包含 NIM 使用的变量。

## 相关信息

`crontab` 命令、`dbts` 命令、`lsts` 命令、`mkcosi` 命令、`mkts` 命令、`nim` 命令、`nim_clients_setup` 命令、`nim_master_setup` 命令、`nimconfig` 命令和 `rmts` 命令。

---

## sync 命令

### 用途

更新 i-node 表，并将缓冲文件写到硬盘中。

### 语法

`sync`

### 描述

`sync` 命令运行 `sync` 子例程。如果必须停止系统，则运行 `sync` 命令以确保文件系统的完整性。`sync` 命令将所有未写的系统缓冲区写到磁盘中，包含已修改的 i-node、已延迟的块 I/O 和读写映射文件。

注：虽然已调度写，但没有必要在其从 `sync` 子例程返回时就完成。

### 相关信息

`sync` 子例程。

---

## synclvodm 命令

### 用途

同步或重建逻辑卷控制块、设备配置数据库和物理卷上的卷组描述符区域。

### 语法

`synclvodm -v -P VolumeGroup LogicalVolume ...`

### 描述

在正常的操作过程中，设备配置数据库与逻辑卷控制块以及物理卷上的卷组描述符区域中的逻辑卷管理器信息保持一致。如果由于某种原因，设备配置数据库跟逻辑卷管理器信息不一致，那么可以使用 `synclvodm` 命令来重新同步数据库。卷组必须是活动的以便再同步（请参阅 `varyonvg`）。如果指定了逻辑卷名，那么只有与那些逻辑卷相关的信息才会得到更新。

**提示:** 不要除去卷组或逻辑卷的 `/dev` 条目。不要使用对象数据管理器来更改卷组或逻辑卷上的设备配置数据库条目。

**注:** 要使用该命令，必须具有 `root` 用户权限或者是 `system` 组的成员。

## 标志

- P** 保持逻辑卷特殊文件的所有权和许可权。如果未设置此标志，逻辑卷特殊文件所有权将设置为 `root`，而组将设置为 `system`。
- v** 指定来自 `synclvodm` 命令的输出以详细方式显示。

## 示例

要使设备配置数据库与 `rootvg` 的逻辑卷管理器信息同步，请输入：

```
synclvodm rootvg
```

## 文件

`/usr/sbin/synclvodm`

包含 `synclvodm` 命令。

## 相关信息

`varyonvg` 命令、`varyoffvg` 命令。

---

## syncvg 命令

### 用途

同步非当前的逻辑卷副本。

### 语法

```
syncvg [ -f ] [ -i ] [ -H ] [ -P NumParallelLps ] { -l | -p | -v } Name ...
```

### 描述

`syncvg` 命令同步非当前的物理分区，即原始物理分区的副本。`syncvg` 命令可用于逻辑卷、物理卷或卷组，带有 `Name` 参数表示逻辑卷名、物理卷名或卷组名。同步进程可能是耗时的，这取决于硬件特征和数据量。

使用 `-f` 标志时，会选择一个好的物理副本传播到逻辑分区的所有其他副本，不管其是否为旧文件。在逻辑卷不具有镜像写一致性恢复时，使用该标志是必要的。

除非禁用，否则当 `varyonvg` 命令激活卷组时，卷组内的副本会自动同步。

**注:** 要成功执行 `syncvg` 命令，至少应该有逻辑卷的一个良好副本可以访问，并且包含该副本的物理卷应该在“活动”的状态。如果使用了 `-f` 选项，以上的条件适用于所有的镜像副本。

如果不指定 `-P` 选项，`syncvg` 会检查 `NUM_PARALLEL_LPS` 环境变量。`NUM_PARALLEL_LPS` 的值会用于设置将要并行同步的逻辑分区数量。

## 标志

|                          |                                                                                                              |
|--------------------------|--------------------------------------------------------------------------------------------------------------|
| <b>-f</b>                | 指定已选择一个良好的物理副本，并将其传播到逻辑分区的所有其他副本，而不管其是否是旧文件。                                                                 |
| <b>-H</b>                | 延迟写其他并发的活动集群节点上的卷组，直到完成同步操作。当使用 <b>-H</b> 标志时， <b>-P</b> 标志不要求集群上所有的节点都支持 <b>-P</b> 标志。如果卷组不以并发方式联机，则会忽略该标志。 |
| <b>-i</b>                | 从标准输入读取名称。                                                                                                   |
| <b>-l</b>                | 指定 <i>Name</i> 参数表示逻辑卷设备名。                                                                                   |
| <b>-p</b>                | 指定 <i>Name</i> 参数代表物理卷设备名。                                                                                   |
| <b>-P NumParallelLps</b> | 将以并行方式同步的逻辑分区的数量。 <i>NumParallelLps</i> 的有效范围是 1 至 32。 <i>NumParallelLps</i> 必须按机器、卷组中的磁盘、系统资源和卷组方式定制。       |
|                          | 如果卷组以并发方式联机，那么使卷组变化的所有其他的集群节点必须至少是 AIX 4.3.0，否则 <b>syncvg</b> 将忽略该选项，并且继续。                                   |
|                          | <b>注：</b> 有关更多的信息，请参阅以上的描述。                                                                                  |
| <b>-v</b>                | 指定 <i>Name</i> 参数表示卷组设备名。                                                                                    |

## 示例

1. 要同步物理卷 `hdisk04` 和 `hdisk05` 的副本，输入：

```
syncvg -p hdisk04 hdisk05
```

2. 要同步卷组 `vg04` 和 `vg05` 的副本，输入：

```
syncvg -v vg04 vg05
```

## 文件

|                               |                      |
|-------------------------------|----------------------|
| <code>/usr/sbin/syncvg</code> | 包含 <b>syncvg</b> 命令。 |
| <code>/tmp</code>             | 该命令运行时，存储临时文件的目录。    |

## 相关信息

**varyonvg** 命令。

《操作系统与设备管理》中的『逻辑卷存储』说明了逻辑卷管理器、物理卷、逻辑卷、卷组、组织、确保数据完整性以及分配特征。

《操作系统与设备管理》中的『系统管理界面工具』说明了结构、主菜单和 SMIT 完成的任务。

---

## syscall 命令

### 用途

执行指定的子例程调用。

### 语法

```
syscall [ -n ] Name [ Argument1 ... ArgumentN ] [ ; Name [ Argument1 ... ArgumentN ] ] ...
```

## 描述

**syscall** 命令执行系统调用接口程序，该程序执行 *Name* 参数指定的子例程调用。如果您指定了 **-n** 标志，那么 **syscall** 命令会执行调用 **n** 次。由 *Argument* 参数指定的变量不经过误差检测就传递到子例程。*Argument* 参数可以通过下列格式表示：

|                   |                                             |
|-------------------|---------------------------------------------|
| 0x <i>nnn</i>     | 十六进制常量 <i>nnn</i> 。                         |
| 0 <i>nnn</i>      | 八进制常量 <i>nnn</i> 。                          |
| <i>nnn</i>        | 十进制常量 <i>nnn</i> 。                          |
| + <i>nnn</i>      | 十进制常量 <i>nnn</i> 。                          |
| - <i>nnn</i>      | 十进制常量 <i>nnn</i> 。                          |
| " <i>string</i> " | 字符串 " <i>string</i> "。                      |
| ' <i>string</i> ' | 字符串 " <i>string</i> "。                      |
| \ <i>string</i>   | 字符串 " <i>string</i> "。                      |
| # <i>string</i>   | 字符串 " <i>string</i> " 的长度。                  |
| && <i>n</i>       | 子例程第 <i>n</i> 个变量的地址。（ <i>n</i> =0 即为子例程名。） |
| & <i>n</i>        | 内部 10KB 缓冲区的第 <i>n</i> 个字节的地址。              |
| \$ <i>n</i>       | 第 <i>n</i> 个子例程的结果。（ <i>n</i> =0 为第一个子例程。）  |
| <i>string</i>     | 任何其他文字字符串。                                  |

**syscall** 命令对于未知子例程和返回值为 -1 的子例程会显示出一条消息并且退出。

注：**syscall** 命令将 **sleep** 子例程理解为特例子例程。

## 标志

**-n** 指定 **syscall** 命令执行指定的子例程的次数。  
； 分离 **syscall** 命令的同一调用发出的多个子例程（最长达 20 个）。

## 示例

要模拟 C 程序段：

```
output=open("x", 401, 0755);  
write(output, "hello", strlen("hello"));
```

输入：

```
syscall open x 401 0755 \; write \0 hello \#hello
```

注：特殊 shell 字符必须转义。

## 文件

**/usr/bin/syscall** 包含 **syscall** 命令。

## 相关信息

**bsh** 命令、**Rsh** 命令、**cs** 命令、**ksh** 命令以及 **sh** 命令。

**open** 子例程、**sleep** 子例程。

《操作系统与设备管理》中的『Shell』。

---

## sysck 命令

### 用途

在安装和更新过程中检查清单信息。

### 语法

```
sysck { -i | -u } [ -R RootPath ] [ -N ] [ -v ] [ -s SaveFile ] [ -O { r | s | u } ] -f File ProductName  
{ tcbck Flags }
```

所有的 **tcbck** 命令标志对该命令都是有效的。

### 描述

注: **sysck** 命令不支持检查大于 2 吉字节的文件。如果一个产品需要提供大于 2 吉字节的文件, 请将它们在 *Fileset* 清单中的大小和校验值设置为 **VOLATILE**, 这样 **sysck** 命令就不会尝试访问这个文件了。

注: 所有 **tcbck** 命令标志对 **sysck** 命令都是有效的。此特征提供了与 V3.1 的兼容性。有关 **tcbck** 命令的更多信息及其标志的完全列表, 请参阅《AIX 5L V5.3 命令参考大全》。

**sysck** 命令对照从安装和更新介质中抽取的文件来检查文件定义, 并更新软件重要产品数据 (SWVPD) 数据库。**sysck** 命令无法识别文件名中的以下特殊字符: \, ', \, ", ^, ( ), |, {}, [], <> 和 :。如果文件名中包含这些字符的其中之一, **sysck** 命令就会失败。

**sysck** 命令主要在软件产品的安装和更新过程中使用。

在使用 **-i** 标志调用时, **sysck** 命令根据文件定义检查抽取的文件的属性, 更新 SWPCD, 并试图修订文件中可能存在的错误。

*File* 参数是包含文件定义的节文件名。此类文件的一个示例是 */etc/security/sysck.cfg* 文件, 尽管 **syschk** 命令不使用这个文件。**sysck** 命令检查类型属性设为 **FILE** 的文件的大小、链接、符号链接、所有者、组和方式属性。当使用 **-v** 标志和 **-i** 标志调用时, **sysck** 也会检查文件校验和的值。

**sysck** 命令会更新 SWVPD 数据库中每个文件的文件名、产品名、类型、校验和以及大小。

为了修订错误, **sysck** 命令会将已安装或更新的文件的属性重新设置为在 *File* 节文件中定义的值, 在“修订错误”中描述的某些属性除外。

当使用 **-u** 标志调用时, **sysck** 命令将每个属于软件产品 *ProductName* 组成部分的文件的条目从 SWVPD 数据库中除去。**sysck** 命令也按照 SWVPD 数据库中的定义删除每个文件的任何硬链接和符号链接。

### 标志

|                           |                                               |
|---------------------------|-----------------------------------------------|
| <b>-f</b> <i>File</i>     | 指定包含文件定义的节文件名。                                |
| <b>-i</b>                 | 检查软件产品文件的正确安装。根据文件定义更新 SWVPD 数据库, 并尝试修订发现的错误。 |
| <b>-N</b>                 | 指定不应更新 SWVPD 数据库。                             |
| <b>-O</b> { <i>rsiu</i> } | 指定要更新 SWVPD 的哪一部分, 如下:                        |
| <b>r</b>                  | 指定 SWVPD 的根部分。                                |
| <b>s</b>                  | 指定 SWVPD 的 <b>/usr/share</b> 部分。              |
| <b>u</b>                  | 指定 SWVPD 的 <b>/usr</b> 部分 (缺省)。               |

|                           |                                                                                                                                                                                                           |
|---------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-R</b> <i>RootPath</i> | 使用 <i>RootPath</i> 作为根以代替 “/”。                                                                                                                                                                            |
| <b>-s</b> <i>SaveFile</i> | 制作 VPD (Vital Product Data : 重要产品数据) 中当前内容的快照, 并以节格式将其保存到 <i>SaveFile</i> 指定的文件。用 <b>-u</b> 选项调用。用这个标志不会对数据库进行操作。必须与 <b>-f</b> 选项一起使用。例如:<br><br>sysck -i -s /tmp/save.inv -f /tmp/real.inv bos.rte.shell |
| <b>-u</b>                 | 从 SWVPD 中删除文件条目, 并删除硬链接和符号链接。                                                                                                                                                                             |
| <b>-v</b>                 | 验证校验和是否正确。                                                                                                                                                                                                |
| <i>ProductName</i>        | 指定正在检查的可安装的软件产品或选项。                                                                                                                                                                                       |

## 环境变量

|                  |                                                                                                                                                                                                                                              |
|------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>INUTREE</b>   | 环境变量 <b>INUTREE</b> 只有下列四种有效值:<br><br><b>NULL</b> 与未设置 <b>INUTREE</b> 相同。<br><b>M</b> 指定 SWVPD 的根部分。<br><b>S</b> 指定 SWVPD 的 <b>/usr/share</b> 部分。<br><b>U</b> 指定 SWVPD 的 <b>/usr</b> 部分 (缺省)。<br><br><b>INUTREE</b> 可以用来代替 <b>-OTree</b> 标志。 |
| <b>INUNOVPD</b>  | 环境变量 <b>INUNOVPD</b> 可以是空的或设置为 1。如果设置为 1, 则 <b>sysck</b> 不会更新 SWVPD。 <b>INUNOVPD</b> 可用来代替 <b>-N</b> 标志。                                                                                                                                     |
| <b>INUVERIFY</b> | 如果将环境变量 <b>INUVERIFY</b> 设置为 1, 那么 <b>sysck</b> 会验证节文件中的校验和属性是否正确。 <b>INUVERIFY</b> 可以用来代替 <b>-v</b> 标志。                                                                                                                                     |

## 文件定义

|                 |                                                                                                                                                                          |
|-----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>acl</b>      | 文件的访问控制表。如果该值为空, 则会除去 <b>acl</b> 属性。如果没有指定值, 则命令会根据访问控制表中描述的格式计算值。<br><br>属性会将 x (执行) 访问权仅授权给 root 用户和安全组的成员。该命令应该 <b>setuid</b> 为 root 用户, 并且具有可信计算库属性。                 |
| <b>class</b>    | 文件的逻辑组。必须指定一个值, 因为不能计算得到该值。该值是 <i>ClassName</i> [ <i>ClassName</i> ]。                                                                                                    |
| <b>checksum</b> | 文件的校验和。如果该值为空, 则会除去 <b>checksum</b> 属性。如果没有指定值, 则该命令根据 <b>sum</b> 命令指定的格式计算值。该值是 <b>sum -r</b> 命令的输出, 包含空格。                                                              |
| <b>group</b>    | 文件组。如果该值为空, 则会除去 <b>group</b> 属性。如果未指定值, 则该命令计算一个值, 该值可以是组标识, 也可以是组名。                                                                                                    |
| <b>mode</b>     | 文件方式。如果该值为空, 则会除去 <b>mode</b> 属性。如果没有指定值, 则该命令计算值, 它可以是八进制数字或字符串 ( <b>rwX</b> ), 并且具有 <b>TCB</b> 、 <b>SUID</b> 、 <b>SGID</b> 以及 <b>SVTX</b> 属性。                          |
| <b>owner</b>    | 文件所有者。如果该值为空, 则会除去 <b>owner</b> 属性。如果没有指定值, 则该命令会计算值, 它可以是用户标识或用户名。                                                                                                      |
| <b>size</b>     | 文件的大小, 以字节表示。如果该值为空, 则会除去 <b>size</b> 属性。在 <b>size</b> 字段中的 <b>VOLATILE</b> 值表示文件大小会更改 (因此不能给出校验和值)。 <b>NOSIZE</b> 值表示文件为 0 长度。如果没有指定值, 则该命令会计算值, 该值为十进制数字。              |
| <b>target</b>   | 允许符号链接和硬链接在清单中以单独节存在。 <b>target</b> 文件定义指向链接源的全路径名称, 例如:<br><br>/etc/foo --> /usr/bar<br><br><b>target</b> 为 /usr/bar。                                                   |
| <b>type</b>     | 文件类型。该值不能为空。如果没有指定值, 则该命令会计算值, 它可以是 <b>FILE</b> 、 <b>DIRECTORY</b> 、 <b>FIFO</b> 、 <b>BLK_DEV</b> 、 <b>CHAR_DEV</b> 、 <b>LINK</b> 、 <b>MPX_DEV</b> 和 <b>SYMLINK</b> 关键字。 |
| <b>xacl</b>     | 对扩展访问控制表的添加。因为不能计算该值, 所以值必须指定为扩展访问控制表中的单个条目。该属性仅在使用 <b>-i</b> 标志时有效。关于格式的更多信息, 请参阅上述的 <b>acl</b> 文件定义。                                                                   |

## 修订错误

为了修订错误，**sysck** 命令会将已安装或已更新的文件的属性重新设置为在 *File* 节文件中定义的值，下列的属性例外。对于这些属性，**sysck** 命令的操作如下所述：

|                 |                                                                           |
|-----------------|---------------------------------------------------------------------------|
| <b>links</b>    | 创建任何缺失的硬链接。如果存在链接指向本定义中未列出的文件，则删除该链接。                                     |
| <b>program</b>  | 如果该属性包含在 <i>File</i> 节文件中， <b>sysck</b> 会调用该程序。如果发生错误，会显示出一条消息，但不采取其他的操作。 |
| <b>symlinks</b> | 创建缺失的符号链接。如果存在链接指向本定义中未列出的文件，则删除该链接。                                      |

## 安全性

特权控制：只有 root 用户才能运行该命令。

## 示例

1. 使用 **installp** 命令安装的产品在其映像中提供一个清单文件。要将定义添加到清单数据库并检查许可权、链接、校验和等，输入：

```
sysck -i -f dude.rte.inventory dude.rte
```

其中 `dude.rte.inventory` 与下列格式相似：

```
/usr/bin/dude.exec:  
class = apply,inventory,dude.rte  
owner = bin  
group = bin  
mode = 555  
type = FILE  
size = 2744  
checksum = "04720      3"
```

2. 要除去指向已从系统除去的产品文件的链接并将这些文件从库存数据库中除去，输入：

```
sysck -u -f dude.rte.inventory dude.rte
```

## 文件

**/etc/objrepos/inventory**

**/usr/lib/objrepos/inventory**

**/usr/share/lib/objrepos/inventory**

在根上指定软件产品文件的名称和位置。

在 **/usr** 文件系统里指定软件产品文件的名称和位置。

在 **/usr/share** 文件系统里指定软件产品文件的名称和位置。

## 相关信息

**installp** 命令、**sum** 命令以及 **tcback** 命令。

---

## syscorepath 命令

### 用途

指定一个系统范围的目录，在该目录中，将转储任何进程的所有核心文件。



## 语法

```
syscorepath [ -p DirectoryName ] [ -g ] [ -c ]
```

## 描述

**syscorepath** 命令使系统管理员能够建立一个系统范围的目录，在该目录中从任何进程转储核心文件。这样可以缓解管理文件系统空间中的管理任务，并提供单一的已知目录来查找核心文件。缺省情况下在正进行核心转储的进程的工作目录中创建核心文件。

系统上的所有用户应对此目录都具有读和写特权。如果用户不具有在该目录中的写许可权，则将不能创建核心文件。将为核心文件提供基于进程标识和时间的唯一名称，这样，核心文件将被命名为 **core.pid.ddhhmmss**，其中 *pid* 是进程标识，*dd* 是几号，*hh* 是以 24 小时制表示的小时，*mm* 是分钟，而 *ss* 是秒。

## 标志

|                                |                                               |
|--------------------------------|-----------------------------------------------|
| <b>-c</b>                      | 取消指定为核心文件库的当前目录的设置。将在进程的工作目录中创建后续的核心文件。       |
| <b>-g</b>                      | 显示指定为核心文件库的当前目录。                              |
| <b>-p <i>DirectoryName</i></b> | 指定该目录以用作核心文件库。 <i>DirectoryName</i> 必须是有效目录名。 |

## 退出状态

|    |         |
|----|---------|
| 0  | 命令成功完成。 |
| >0 | 发生错误。   |

## 标准错误

### EPERM

用户不具有许可权。

### ENOTDIR

指定的 *DirectoryName* 不是目录。

### ENAMETOOLONG

指定的 *DirectoryName* 过长。

## 安全性

只有 root 用户可以运行该命令。

## 示例

- 要将 **/core** 设置为核心文件库，请输入：

```
syscorepath -p /core
```
- 要显示当前核心文件库，请输入：

```
syscorepath -g
```
- 要取消设置用作核心文件库的目录，请输入：

```
syscorepath -c
```

# 文件

`/usr/bin/syscorepath`

包含 `syscorepath` 命令。

## 相关信息

`dbx` 命令、`gencore` 命令。

`a.out` 文件格式、`core` 文件格式。

---

## sysdumpdev 命令

### 用途

更改运行中系统的主要或辅助转储设备的指定。

### 语法

```
sysdumpdev -P { -p Device | -s Device } [ -q ] [ -i ]
```

```
sysdumpdev [ -p Device | -s Device ] [ -q ]
```

```
sysdumpdev [ -d Directory | -D Directory | -e | -l | [ -k | -K ] | -l | -p Device | -q | -r Host: Path | -s Device | -z ]
```

```
sysdumpdev [ -c | -C ] [ -i ]
```

```
sysdumpdev -L { -v | -S Device }
```

### 描述

`sysdumpdev` 命令更改运行中系统的主要或辅助转储设备的指定。主要和辅助转储设备在一个系统配置对象中指定。新的设备指定直到 `sysdumpdev` 命令重新运行或系统重新启动时才有效。

如果 `sysdumpdev` 命令没有使用标志，则会使用在 `SWservAt` ODM 对象类中定义的转储设备。缺省的主要转储设备是 `/dev/hd6`。缺省的辅助转储设备是 `/dev/sysdumpnull`。

#### 注:

1. 镜像调页空间可用作转储设备。
2. 不要使用软盘驱动器作为您的转储设备。
3. 如果您使用页面调度设备，那么只使用主要页面调度设备 `hd6`。AIX 4.2.1 或以后的版本支持使用根卷组 (`root volume group: rootvg`) 中的任何页面调度设备作为辅助转储设备。
4. 如果使用可拆卸设备（例如，磁带或 DVD），要注意转储不跨越卷。因而，转储必须安装在单个卷上。

您可以使用基于 Web 的系统管理器 (`wsm`) 中的设备应用程序来运行该命令。

您也可以使用 `sysdumpdev` 命令指定在将转储写入转储设备前是否将转储压缩。压缩转储会减少转储设备需要的大小，但会引起转储进程占用更长的时间。

#### 注:

1. **savecore** 命令用于将已压缩的转储从转储设备复制到文件。
2. 转储压缩功能只适用于 AIX 4.3.2 和以后的版本。
3. 在 AIX 5.3 中, 在缺省情况下将打开转储压缩。

## 在非根卷组运行 sysdumpdev

如果不是永久转储设备, 您可在根卷组之外使用转储逻辑卷。例如, 如果没有指定 **-P** 标志。不过, 如果您选择了调页空间, 就不能复制转储设备了, 除非它在根卷组中。必须复制转储设备时, 启动调页前只有根卷组是活动的。

主要转储设备必须总是在根卷组中以作为永久转储设备。辅助设备可以在根卷组外, 除非它是调页空间。

## 用 sysdumpdev 配置远程转储设备

**sysdumpdev** 命令也可用来配置远程转储设备。在配置远程转储设备前必须满足下列条件:

- 本地主机必须是单处理器机器。
- 本地和远程主机必须安装和配置了传输控制协议 / 网际协议 (TCP/IP)。
- 本地主机必须安装了网络文件系统 (Network File System: NFS)。
- 远程主机必须支持 NFS。
- 远程主机必须是在网络上并且是可操作的。可以发出 **ping** 命令来测试这个条件。
- 远程主机必须定义 NFS 导出目录以使本地主机对远程主机上的转储文件具有读写许可权以及 root 访问权。
- 远程主机不能与本地主机相同。

网络设备驱动程序必须支持远程转储。支持远程转储的驱动程序包含这些网络设备的驱动程序:

- Integrated Ethernet MCA Adapter through AIX 5.1 only
- IBM 10/100 Mbps Ethernet TX MCA Adapter (8f62) through AIX 5.1 only
- IBM PCI Ethernet Adapter (22100020)
- IBM 10/100 Mbps Ethernet PCI Adapter (23100020)
- FDDI MCA Adapter through AIX 5.1 only
- ISA Token Ring Adapter
- Token-Ring High-Performance Adapter (8fa2)
- Token-Ring High-Performance Adapter (8fc8)
- IBM PCI Token-Ring Adapter (14101800)
- IBM PCI Token-Ring Adapter (14103e00)

不支持远程转储的驱动程序包含这些网络设备的驱动程序:

- ISA Ethernet Adapter
- Ethernet High-Performance LAN Adapter (8ef5)
- Ethernet High-Performance LAN Adapter (8f95)
- Gigabit Ethernet-SX PCI Adapter (14100401)

## 标志

- c 指定不压缩转储。**-c** 标志只适用于 AIX 4.3.2 和以后的版本。
- C 指定所有将来的转储在其写入转储设备之前将其压缩。**-C** 标志只适用于 AIX 4.3.2 和以后的版本。

|                            |                                                                                                                                                                                                                                                                     |
|----------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-d</b> <i>Directory</i> | 指定系统引导时转储所复制到的目录。如果引导时复制失败，那么 <b>-d</b> 标志会忽略系统转储。                                                                                                                                                                                                                  |
| <b>-D</b> <i>Directory</i> | 指定系统引导时转储所复制到的目录。如果引导时复制失败，那么使用 <b>-D</b> 标志允许您将转储复制到外部的介质。<br>注：使用 <b>-d</b> <i>Directory</i> 或 <b>-D</b> <i>Directory</i> 标志时，会检测下列的错误情况： <ul style="list-style-type: none"> <li>• 目录不存在。</li> <li>• 目录不在本地日志文件系统里。</li> <li>• 目录不在 <b>rootvg</b> 卷组中。</li> </ul> |
| <b>-e</b>                  | 估计当前运行的系统的转储大小（以字节表示）。如果压缩转储，那么所显示的大小是压缩之后大小的估计值。                                                                                                                                                                                                                   |
| <b>-i</b>                  | 表示从系统函数调用 <b>sysdumpdev</b> 命令。只有系统实用程序才使用该标志。如果不是自动 IBM 函数的函数已修改了受影响的值，则 <b>-i</b> 标志不会作出请求的更改；也就是说， <b>-i</b> 标志不会覆盖之前的更改。                                                                                                                                        |
| <b>-l</b>                  | 重新设置先前更改的指示信息。指定 <b>-l</b> 标志后，用 <b>-i</b> 标志就允许更改。                                                                                                                                                                                                                 |
| <b>-k</b>                  | 如果您的机器具有密钥方式开关，那么在能够用转储键序列强制转储前，它需要处于服务位置。在 AIX 5.3 之前，此限制也适用于复位按钮。                                                                                                                                                                                                 |
| <b>-K</b>                  | 如果您的机器有密钥方式开关，那么密钥处于正常位置时使用复位按钮或转储键序列将强制转储，或在没有密钥方式开关的机器上也强制转储。<br>注：在没有密钥方式开关的机器上，不能未设置此值就用键序列强制转储。AIX 5.3 之前的复位按钮情况也是如此。                                                                                                                                          |
| <b>-l</b>                  | 列出主要和辅助转储设备、复制目录和 <b>forcecopy</b> 属性的当前值。                                                                                                                                                                                                                          |
| <b>-L</b>                  | 显示最近的系统转储的统计信息。这包含最近的转储的日期和时间、所写的字节数和完成状态。如果压缩了转储，那么该标志既显示转储的原始的未压缩的大小，也显示了转储的压缩后的大小。压缩后的大小是实际写入转储设备的大小。如果先前没有将转储记录在非易失性的内存中，则该标志在转储设备中扫描现有的转储。<br>注：显示的转储大小可能不会反映介质上的转储的精确大小。由于磁盘和复制块大小，因此会有一些差别。                                                                  |
| <b>-P</b>                  | 使 <b>-p</b> 或 <b>-s</b> 标志指定的转储设备成为永久设备。 <b>-P</b> 标志只能与 <b>-p</b> 或 <b>-s</b> 标志一起使用。                                                                                                                                                                              |
| <b>-p</b> <i>Device</i>    | <i>Device</i> 暂时将主要转储设备更改为指定的设备。此设备可以为逻辑卷、可写 DVD 或磁带设备。对于网络转储，该设备可以是主机名和路径名。                                                                                                                                                                                        |
| <b>-q</b>                  | 禁止输出到标准输出的所有消息。如果该标志与 <b>-l</b> 、 <b>-r</b> 、 <b>-z</b> 或 <b>-L</b> 标志一起使用，则会忽略 <b>-q</b> 命令。                                                                                                                                                                       |
| <b>-r</b> <i>Host:Path</i> | 释放服务器 <i>Host</i> 上远程转储文件所使用的空间。 <i>Path</i> 指定转储文件的位置。                                                                                                                                                                                                             |
| <b>-s</b> <i>Device</i>    | <i>Device</i> 暂时将辅助转储设备更改为指定的设备。如果相同设备对于 <b>-p</b> 有效，那么在这里也有效。                                                                                                                                                                                                     |
| <b>-S</b> <i>Device</i>    | 扫描特定的转储设备以进行有效的压缩转储。该转储必须来自支持并行转储的 AIX 发行版。该标志只能与 <b>-L</b> 标志一起使用。                                                                                                                                                                                                 |
| <b>-v</b>                  | 转储状态为非 0 时，此选项将显示可用的转储调试信息。当调试数据可用时，服务将用它来诊断转储故障。此标志只能与 <b>-L</b> 标志一起使用。                                                                                                                                                                                           |
| <b>-z</b>                  | 确定是否出现新的系统转储。如果出现，则会包含转储大小（以字节表示）和转储设备名的字符串写到标准输出。如果不存在新的系统转储，则不返回任何东西。在现有的系统转储上运行 <b>sysdumpdev -z</b> 命令后，就不认为转储是最新了。                                                                                                                                             |

如果没有标志和 **sysdumpdev** 命令一起使用，就会使用缺省的转储设备。

## 安全性

访问控制：只有 root 用户才能运行该命令。

## 示例

1. 要显示当前的转储设备设置，输入：

```
sysdumpdev -l
```

2. 要将逻辑卷 hd7 指派为主要转储设备，输入：

```
sysdumpdev -p /dev/hd7
```

3. 要将磁带设备 rmt0 指派为辅助转储设备，输入：

```
sysdumpdev -s /dev/rmt0
```

4. 要显示先前的转储调用的信息，输入：

```
sysdumpdev -L
```

注：非零转储状态指示转储失败。可能的转储状态值和它们对应的 LED 值如下：

| 转储状态 | 描述        | LED 值 |
|------|-----------|-------|
| 0    | 转储已成功完成   | 0C0   |
| -1   | 未定义转储设备   | 0C8   |
| -2   | 转储设备太小    | 0C4   |
| -3   | 转储已崩溃或未启动 | 0C5   |
| -4   | I/O 错误    | 0C1   |

5. 要永久地将主要转储设备的数据库对象更改到 /dev/newdisk1，输入：

```
sysdumpdev -P -p /dev/newdisk1
```

6. 要确定是否存在新的系统转储，输入：

```
sysdumpdev -z
```

如果最近发生了系统转储，则会出现与下列相似的输出：

```
4537344 /dev/hd7
```

7. 将远程转储文件 /var/adm/ras/systemdump（在主机 mercury 上）指派给主要转储设备，输入：

```
sysdumpdev -p mercury:/var/adm/ras/systemdump
```

在主机名和文件名之间必须输入冒号：。

8. 要指定系统崩溃后转储要复制到其上的目录（如果转储设备是 /dev/hd6），输入：

```
sysdumpdev -d /tmp/dump
```

这会在系统崩溃后试图将转储从 /dev/hd6 复制到 /tmp/dump。如果在复制过程中出现了错误，那么系统继续引导，但是丢失了转储。

9. 要指定系统崩溃后转储要复制到其上的目录（如果转储设备是 /dev/hd6），输入：

```
sysdumpdev -D /tmp/dump
```

这会在崩溃后尝试将转储从 /dev/hd6 复制到 /tmp/dump 目录。如果复制失败，那么会提示您一个菜单以允许手工将转储复制到某个外部介质。

10. 要扫描转储设备以进行转储，请输入：

```
sysdumpdev -L -S /dev/hd6
```

## 相关信息

**mount** 命令、**ping** 命令和 **savecore** 命令、**sysdumpstart** 命令。

*AIX 5L Version 5.3 Kernel Extensions and Device Support Programming Concepts* 中的 System Dump Facility。

有关安装基于 Web 的系统管理器的信息，请参阅《*AIX 5L V5.3 基于 web 的系统管理器管理指南*》中的『第二章：安装与系统需求』。

---

## sysdumpstart 命令

### 用途

提供命令行界面以便开始内核转储至主要或辅助转储设备。

### 语法

```
sysdumpstart { -p | -s [ -f ] }
```

### 描述

**sysdumpstart** 命令提供命令行界面以便开始内核转储至主要或辅助转储设备。转储完成时，系统停机。使用 **crash** 命令检查内核转储。使用 **sysdumpdev** 命令重新分配转储设备。

内核转储过程中，在三位数字终端显示屏上可显示下列值：

- 0c0** 表示转储成功完成。
- 0c1** 表示在转储过程中发生 I/O。该值只适用于 AIX 4.2.1 或更新的版本。
- 0c2** 表示有一个转储在进行中。
- 0c4** 表示转储太小。
- 0c5** 表示转储内部错误。
- 0c6** 提示您准备好辅助转储设备。该值不适用 AIX 4.2.1 或更新的版本。
- 0c7** 表示转储进程正在等待远程主机响应。
- 0c8** 表示禁用转储。在这种情况下，转储设备系统配置对象中就不指定转储设备。**sysdumpstart** 命令中断，但系统继续运行。
- 0c9** 表示该转储在进行中。
- 0cc** 表示试图转储到主要设备后系统切换至辅助转储设备。该值只适用于 AIX 4.2.1 或更新的版本。

您可以使用基于 Web 的系统管理器设备应用程序（**wsm devices** 快速路径）来运行该命令。您也可以使用系统管理界面工具（SMIT）**smit sysdumpstart** 快速路径来运行该命令。

### 标志

- f** 禁止使辅助转储设备就绪的提示。该标志不适用于 AIX 4.2.1 或更新的版本。
- p** 启动系统转储，并将结果写到主要转储设备。
- s** 启动系统转储并将结果写到辅助转储设备。

### 安全性

访问控制：只有 root 用户才能运行该命令。

## 示例

1. 要开始内核转储至主要转储设备, 输入:  
`sysdumpstart -p`
2. 要开始内核转储至辅助转储设备, 输入:  
`sysdumpstart -s`

## 相关信息

*AIX 5L Version 5.3 Kernel Extensions and Device Support Programming Concepts* 中的 System Dump Facility。

**sysdumpdev** 命令。

有关安装基于 Web 的系统管理器的信息, 请参阅《AIX 5L V5.3 基于 web 的系统管理器管理指南》中的『第二章: 安装与系统需求』。

---

## sysline 命令

### 用途

在终端的状态行显示系统状态。

### 语法

```
/usr/bin/sysline [ -b ] [ -c ] [ -d ] [ -e ] [ -h ] [ -i ] [ -j ] [ -l ] [ -m ] [ -p ] [ -q ] [ -r ] [ -s ] [ -w ] [ -D ] [ -H Remote ] [ +N ]
```

### 描述

**sysline** 命令在后台运行, 并在终端的状态行定期显示系统状态信息。并非所有的终端都包含状态行。如果没有指定标志, 那么 **sysline** 命令就会显示下列的状态项:

- 一天中的时间
- 当前可能运行的进程数
- 用户数 (后跟一个 **u**)
- 可执行进程数 (后跟一个 **r**)
- 暂挂进程数 (后跟一个 **s**)
- 自上次状态报告以来登录和注销的用户数

最后, 如果有新的邮件到达, 就会显示出新邮件的摘要。如果在您的邮箱里有未读邮件, 那么在显示了用户数后会出现星号。通常以反相显示方式显示 (如果您的终端在状态行支持这种显示方式的话), 并且右对齐以减少干扰。正常视频每隔四次显示一次, 以给屏幕休息的机会。

如果您在主目录有一个叫 **.who** 的文件, 那么首先会显示出该文件的内容。该功能的一个通常的用处是在它更改新目录后给 **chdir**、**pushd** 和 **popd** 命令取别名以将当前目录堆栈放置在 **/.who** 中。

如果您的主目录上有一个叫 **.syslinelock** 的文件, 则 **sysline** 命令不会更新其统计信息并写在屏幕上, 而只是进入休眠状态一小会儿。如果想暂时禁用 **sysline**, 这是很有用的。注意从创建锁定文件到确保 **sysline** 不会在屏幕上写信息可能要花几秒钟。

## 标志

|                  |                                                                                                                                                       |
|------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-b</b>        | 每半个小时一次，每小时两次发出蜂鸣声。                                                                                                                                   |
| <b>-c</b>        | 每次重新显示前五秒钟清除状态行。                                                                                                                                      |
| <b>-D</b>        | 在时间之前显示当前的星期 / 日期。                                                                                                                                    |
| <b>-d</b>        | 以用户可读格式和调试模式显示状态行数据。                                                                                                                                  |
| <b>-e</b>        | 只显示信息。禁止信息写入末行所必需的控制命令。该选项对于将 <b>sysline</b> 命令的输出写入 <b>emacs</b> 窗口的方式是有用的。                                                                          |
| <b>-H Remote</b> | 显示远程主机 <i>Remote</i> 的负载平均值。如果主机已关闭，或者不发送 <i>rwhod</i> 信息包，则改为显示关闭时间。如果出现前缀 <b>ucb</b> ，则将其除去。                                                        |
| <b>-h</b>        | 在时间之后显示出主机名。                                                                                                                                          |
| <b>-i</b>        | 启动时将 <b>sysline</b> 命令进程的进程标识显示到标准输出。用该信息您可以发送警报信号到 <b>sysline</b> 进程以使其立刻更新。 <b>sysline</b> 命令写到标准错误，因此您可以将标准输出重定向到文件以捕获进程标识。                        |
| <b>-j</b>        | 将状态行上能够移动光标的终端上的 <b>sysline</b> 命令输出左对齐。                                                                                                              |
| <b>-l</b>        | 禁止显示登录和注销的用户名。                                                                                                                                        |
| <b>-m</b>        | 禁止邮件检查。                                                                                                                                               |
| <b>+N</b>        | 每 <i>N</i> 秒更新状态行。缺省值为 60 秒。                                                                                                                          |
| <b>-p</b>        | 禁止可执行和暂挂进程数报告。                                                                                                                                        |
| <b>-q</b>        | 如果在启动时出现问题则禁止显示输出诊断消息。                                                                                                                                |
| <b>-r</b>        | 禁止反相显示。                                                                                                                                               |
| <b>-s</b>        | 当且仅当 ( <i>iff</i> ) 状态行中不允许转义时，以左对齐方式显示简短格式行。某些终端（例如：Televidios 和 Freedom 100）不允许在状态行移动光标（或其他“智能”操作）。对于这些终端， <b>sysline</b> 命令通常使用空格来实现右对齐。该标志禁止添加空格。 |
| <b>-w</b>        | 在终端当前行显示状态，适合在单行窗口（窗口方式）中使用。                                                                                                                          |

## 示例

要显示星期和日期、可能运行的进程数、用户数并在更新之前五秒钟清除屏幕，请输入：

```
sysline -Dcr
```

注：仅在具有状态行功能的屏幕上起作用。

## 文件

|                               |                              |
|-------------------------------|------------------------------|
| <b>/etc/utmp</b>              | 包含已登录的用户名。                   |
| <b>/dev/kmem</b>              | 包含进程表。                       |
| <b>/var/spool/rwho/whod.*</b> | 包含远程主机的 who/Uptime 信息。       |
| <b>\${HOME}/.who</b>          | 指定要在末行显示的信息。                 |
| <b>\${HOME}/.syslinelock</b>  | 指定当其存在时， <b>sysline</b> 不显示。 |

## 相关信息

**pstat** 命令、**vmstat** 命令。

---

## syslogd 守护程序

### 用途

记录系统消息。



## 语法

**syslogd** [ **-d** ] [ **-s** ] [ **-f** *ConfigurationFile* ] [ **-m** *MarkInterval* ] [ **-r** ] [ **-R** ] [ **-n** ] [ **-N** ] [ **-p** *LogName* ] [ **-M** *all* ]

## 描述

**syslogd** 守护程序读取数据报套接字，并将每个消息行发送到 **/etc/syslog.conf** 配置文件所描述的目的地。

**syslogd** 守护程序在激活和收到挂断信号时读取配置文件。

**syslogd** 守护程序创建 **/etc/syslog.pid** 文件，它包含一个单行，内容是用于结束或重新配置 **syslogd** 守护程序的命令进程标识。

发送到 **syslogd** 守护程序的终止信号结束该守护程序。**syslogd** 守护程序记录结束信号信息并立刻终止。

每条消息占一行。消息可能包含优先级代码，该代码用行首 < >（尖括号）中所包含的数字来标记。可能会截短长度超过 900 字节的消息。

**/usr/include/sys/syslog.h** 包含文件定义了配置文件使用的程序和优先级代码。本地写的应用程序使用包含在 **syslog.h** 文件中的定义来通过 **syslogd** 守护程序记录消息。

## 标志

|                                    |                                                                                                                                             |
|------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-d</b>                          | 打开调试。                                                                                                                                       |
| <b>-f</b> <i>ConfigurationFile</i> | 指定备用的配置文件。                                                                                                                                  |
| <b>-m</b> <i>MarkInterval</i>      | 指定在 <b>mark</b> 命令消息之间的分钟数。如果不使用这个标志，则 <b>mark</b> 命令每 20 分钟发送一个具有 <b>LOG_INFO</b> 优先级的消息。包含 *（星号）的 <b>selector</b> 字段不能启用该程序，而是将选择所有其他的程序。 |
| <b>-M</b> <i>all</i>               | 指定不要禁止日志文件中的重复消息。该标志仅在与 <b>all</b> 参数一起使用时才有效。                                                                                              |
| <b>-s</b>                          | 指定对本地系统上生成的所有转发系统日志消息都转发一条“缩短的”消息到另一个系统（如果它被配置成这样做）。                                                                                        |
| <b>-r</b>                          | 禁止记录从远程主机接收的消息。                                                                                                                             |
| <b>-R</b>                          | 指定时，使用因特网域套接字禁用设施从网络接收消息。                                                                                                                   |
| <b>-n</b>                          | 禁止“从 <log_host_name> 转发的消息：”字符串，该字符串添加在转发到远程日志主机的 <b>syslog</b> 消息的开头。                                                                      |
| <b>-N</b>                          | 禁止在每个日志消息中记录优先级和程序信息。                                                                                                                       |
| <b>-p</b>                          | 指定 UNIX 数据报套接字的备用路径名。                                                                                                                       |

## 配置文件

配置文件通知 **syslogd** 守护程序将系统消息发送到哪里，这取决于消息的优先级和生成该消息的程序。

如果不使用 **-f** 标志，则 **syslogd** 守护程序将读取缺省配置文件 **/etc/syslog.conf**。

**syslogd** 守护程序忽略空行和以 #（镑符号）开头的行。

## 格式

**syslogd** 守护程序的配置文件中的行包含 **selector** 字段、**action** 字段和可选的 **rotation** 字段，以一个或多个制表符分隔。

**selector** 字段名指定程序和优先级。用一个 ,（半角逗号）分隔程序名称。用一个 .（句点）分隔 **selector** 字段的程序和优先级部分。用 ;（半角分号）分隔同一 **selector** 字段的多个条目。使用 \*（星号）选择所有的程序。

action 字段标识接收消息的目的地（文件、主机或用户）。如果路由到远程主机，则远程系统会处理在其自己的配置文件中表示的消息。要在用户的终端显示消息，destination 字段必须包含有效的登录系统的用户名。

rotation 字段标识怎样使用轮换。如果 action 字段是文件，则可以根据大小和 / 或时间进行轮换。也可以压缩和 / 或归档轮换的文件。

## 程序

在 selector 字段使用下列系统程序名:

| 程序              | 描述                  |
|-----------------|---------------------|
| kern            | 内核                  |
| user            | 用户级别                |
| mail            | 邮件子系统               |
| daemon          | 系统守护程序              |
| auth            | 安全性或授权              |
| syslog          | <b>syslogd</b> 守护程序 |
| lpr             | 行式打印机子系统            |
| news            | 新闻子系统               |
| uucp            | uucp 子系统            |
| local0 到 local7 | 本地使用                |
| *               | 所有程序                |

## 优先级

在 selector 字段中使用下列的消息优先级。指定等级及所有更高等级的消息会按指示发送。

| 优先级     | 描述                                                                             |
|---------|--------------------------------------------------------------------------------|
| emerg   | 指定紧急消息 ( <b>LOG_EMERG</b> )。这些消息并非分发给所有用户。可以将 <b>LOG_EMERG</b> 优先级消息记录到单独文件备查。 |
| alert   | 指定重要的消息 ( <b>LOG_ALERT</b> )，如严重的硬件错误。这些消息分发给所有用户。                             |
| crit    | 指定不列为错误的消息 ( <b>LOG_CRIT</b> )，如不适当的登录尝试。 <b>LOG_CRIT</b> 和较高优先级消息会发送到系统控制台。   |
| err     | 指定表示错误情况的消息 ( <b>LOG_ERR</b> )，例如失败的磁盘写入。                                      |
| warning | 指定反常但可恢复的情况的消息 ( <b>LOG_WARNING</b> )。                                         |
| notice  | 指定重要的信息性消息 ( <b>LOG_NOTICE</b> )。没有指定优先级的消息会映射为此优先级的消息。                        |
| info    | 指定信息性消息 ( <b>LOG_INFO</b> )。这些消息可以废弃，但它们在分析系统是很有用。                             |
| debug   | 指定调试消息 ( <b>LOG_DEBUG</b> )。这些消息可以废弃。                                          |
| none    | 排除选定的程序。只有在同一 selector 字段里跟在带有 * (星号) 的条目后时，该优先级级别才有用。                         |

## 目的地

使用 action 字段中的下列消息目的地。

| 目的地               | 描述                  |
|-------------------|---------------------|
| 文件名               | 以附加方式打开的文件的全路径名     |
| @Host             | 主机名，在其前面有 @ (at 符号) |
| User[, User][...] | 用户名                 |
| *                 | 所有用户                |

## Rotation

使用下列在 `rotation` 字段里的轮换关键字。

| 关键字                   | 描述                                                                                                |
|-----------------------|---------------------------------------------------------------------------------------------------|
| <code>rotate</code>   | 该关键字必须在 <code>action</code> 字段之后指定。                                                               |
| <code>size</code>     | 该关键字指定旋转基于的大小。其后接数字和一个 <b>k</b> (千字节) 或 <b>m</b> (兆字节)。                                           |
| <code>time</code>     | 该关键字指定旋转基于的时间。其后接数字和一个 <b>h</b> (小时) 或 <b>d</b> (日) 或 <b>w</b> (周) 或 <b>m</b> (月) 或 <b>y</b> (年)。 |
| <code>files</code>    | 该关键字指定旋转的文件的总数。其后接数字。如果未指定, 则有无限数量的轮换文件。                                                          |
| <code>compress</code> | 该关键字指定要压缩的已保存轮换文件。                                                                                |
| <code>archive</code>  | 该关键字指定将保存的轮换文件复制到目录。其后接目录名。                                                                       |

## 示例

- 要在调试级别或更高的级别将所有的邮件程序消息记录到文件 `/tmp/mailsyslog`, 输入:  
`mail.debug /tmp/mailsyslog`
- 要将除来自邮件程序之外的所有系统消息发送到叫 `rigil` 的主机, 输入:  
`*.debug;mail.none @rigil`
- 要将所有程序的 **emerg** 优先级的消息和来自邮件和守护程序程序的 **crit** 优先级及以上级别的消息发送到用户 `nick` 和 `jam`, 输入:  
`*.emerg;mail,daemon.crit nick, jam`
- 要将所有的邮件程序消息发送到所有用户的终端屏幕, 输入:  
`mail.debug *`
- 要将所有调试或以上级别的程序消息记录到文件 `/tmp/syslog.out`, 并在收到超过 500 千字节的文件或者经过一周时轮换文件, 将已轮换的文件数限制在 10 个, 使用压缩同时使用 `/syslogfiles` 作为压缩文档目录, 输入:  
`*.debug /tmp/syslog.out rotate size 500k time 1w files 10 compress archive /syslogfiles`

## 文件

|                               |                        |
|-------------------------------|------------------------|
| <code>/etc/syslog.conf</code> | 控制 <b>syslogd</b> 的输出。 |
| <code>/etc/syslog.pid</code>  | 包含进程标识。                |

## 相关信息

**syslog** 子例程。

---

## tab 命令

### 用途

将空格更改为制表符。

### 语法

**tab** [ **-e** ] [ *File ...* ]

## 描述

**tab** 命令读取由 *File* 参数指定的文件或标准输入，并在 **tab** 命令可以消除一个或多个空格的地方用制表符代替输入中的空格。如果用 *File* 参数指定一个文件，则 **tab** 命令将结果文件写回到原始文件中。如果输入是标准输入，则 **tab** 命令写到标准输出中。**tab** 命令假定每八列设置制表符停止位，从第九列开始。*File* 参数指定的文件名长度不能超出 **PATH\_MAX**-9 字节。

## 标志

**-e** 只替换每行开头第一个非空格字符前的那些空格。

## 示例

要用制表符替换 *File* 文件中的空格字符，输入：

```
tab File
```

## 文件

**/usr/bin/tab** 包含 **tab** 命令。

## 相关信息

**expand** 命令、**newform** 命令、**unexpand** 命令和 **untab** 命令。

《操作系统与设备管理》中的『文件』。

《操作系统与设备管理》中的『输入和输出重定向』。

---

## tabs 命令

### 用途

在终端上设置制表符停止位。

### 语法

```
tabs [ TabSpec ... ] [ +m [ Number ] ] [ -TTerminal ... ]
```

### 描述

**tabs** 命令在支持可远程设置硬件制表符的终端上指定制表符停止位。根据 *TabSpec* 参数设置制表符停止位，并擦除先前的设置。

使用 **tabs** 命令时，总是将最左列号引用为 1，即使工作站引用它为 0。

如果不指定 *TabSpec* 参数，那么缺省值为 **-8**。

*TabSpec* 参数可以指定下列预置的格式：

- a** 将制表符设置为 1、10、16、36 和 72（IBM System/370™ 汇编程序第一格式）。
- a2** 将制表符设置为 1、10、16、40 和 72（IBM System/370 汇编程序第二格式）。
- c** 将制表符设置为 1、8、12、16、20 以及 55（COBOL 正常格式）。

- c2** 将制表符设置为 1、6、10、14 以及 49 (COBOL 压缩格式, 省略 1-6 列)。用该代码, 第一列位置相当于卡片列第 7 列。一个空格使您到达第 8 列, 而一个制表符使您到达第 12 列。使用该代码的文件应该包括这样的格式规范:  

```
<:t-c2 m6 s66 d:>
```
- c3** 将制表符设置为 1、6、10、14、18、22、26、30、34、38、42、46、50、54、58、62 以及 67 (比 **-c2** 更多制表符的 COBOL 压缩格式)。这些制表符提供推荐的 COBOL 格式。使用该代码的文件应该包括这样的格式规范:  

```
<:t-c3 m6 s66 d:>
```
- f** 将制表符设置为 1、7、11、15、19 以及 23 (FORTRAN)。
- p** 将制表符设置为 1、5、9、13、17、21、25、29、33、37、41、45、49、53、57 以及 61 (PL/I)。
- s** 将制表符设置为 1、10 以及 55 (SNOBOL)。
- u** 将制表符设置为 1、12、20 以及 44。

除了预置的格式, *TabSpec* 参数可包含:

- Number** 每 *Number* 列规则地重复设置制表符。(操作系统制表符标准设置是 **-8**。使用带有 **-h** 标志的 **nroff** 命令时需要 **-8** 设置。)另一种特殊情况是 **-0** 设置, 这意味着根本就没有制表符。如果设置的制表符超过了 20 个, 您必须运行两次 **tabs** 命令来将其清除。  
  
*Number* [ ,*Number* ] ...  
  
在指定列号处设置制表符 (逗号分隔并以升序排列的列表)。您最多可指定 40 个列号。如果除第一个列号之外的任何其他列号有一个加号前缀, 那么该有前缀的列号就会加到前一列号中以形成下一设置的列号。因此, **1、10、20、30** 指定的制表符列表提供了与 **1、10、+10、+10** 指定的制表符列表相同的制表符设置。
- Filep** 读取 *Filep* 文件的第一行以找出格式规范。如果 **tabs** 命令找出格式规格, 那么 **tabs** 命令如指定的那样设置制表符。如果 **tabs** 命令没有找到格式规范, 那么将制表符设置为系统缺省值 (**-8**)。

用非标准制表符停止位设置 (不在每个第八列设置的制表符停止位) 有时可方便地维护文本文件。这种文件必须转换为标准格式。通常在其可被任何命令处理之前用适当数目的空格字符代替所有的制表符而完成的。在文本文件第一行出现的格式规范指定在该文件剩余部分如何扩展制表符。

格式规范由一个被空格分离以及被 **<:** 和 **:>** 包围的参数序列所构成。每个参数包含字母键, 可能在其后还紧跟值。可识别下列参数:

*ttabs*

指定一个文件的制表符停止位设置。 *tabs* 的值必须是下列的其中之一：

- 以逗号分隔的列号的列表，其表示在指定列设置了制表符停止位。
- 一个 —（破折号）紧跟一个整数 *n*，表示以 *n* 列的间隔设置制表符停止位，也就是说，在  $1+n$ ， $1+2*n$ ，等位置。
- 一个 —（破折号）后跟预置的制表符停止位规范的名称。

逗号分隔的制表符停止位设置列表中最多允许 40 个列号。如果数字（第一个除外）的前面有加号，那么以递增方式加到前面的值上。因此，格式 **t1、10、20、30** 和 **t1、10、+10、+10** 视为相同的。

标准制表符停止位由 **t-8** 或与此相当的 **t1、9、17、25** 来指定。这是大多数系统实用程序假定的制表符停止位设置，并且在终端上最有可能的设置。规范 **t-0** 根本不指定任何制表符停止位。

可识别的预置制表符停止位的规范如下：

- a** 1、10、16、36、72  
汇编程序，IBM System/370，第一格式
- a2** 1、10、16、40、72  
汇编程序，IBM System/370，第二格式
- c** 1、8、12、16、20、55  
COBOL，正常格式
- c2** 1、6、10、14、49  
COBOL 压缩格式（省略 1-6 列）。使用该代码，第一个输入的字符对应于第 7 卡片列；一个空格使您到达第 8 列；而一个制表符使您到达第 12 列。使用制表符停止位设置的文件应该包含如下的格式规范：  
<:t-c2 m6 s66 d:>
- c3** 1、6、10、14、18、22、26、30、34、38、42、46、50、54、58、62、67  
具有比 **c2** 更多制表符停止位的 COBOL 压缩格式（省略 1-6 列）。这是 COBOL 的推荐格式。适当的格式规格是：  
<:t-c3 m6 s66 d:>
- f** 1、7、11、15、19、23  
FORTRAN
- p** 1、5、9、13、17、21、25、29、33、37、41、45、49、53、57、61  
PL/I
- s** 1、10、55  
SNOBOL
- u** 1、12、20、44  
UNIVAC 1100 汇编程序

*ssize*

指定最大的行大小。大小的值必须是一个整数。在扩展制表符之后，调整页边距之前执行大小检查。

*mmargin*

指定添加到每行开头的空格字符数。页边距值必须是一个整数。

*d*

表示包含格式规范的行将从已转换的文件中删除。*d* 参数没有值。

*e*

表示在文件中遇到另一种格式规范时，当前的格式才失效。*e* 参数没有值。

缺省值为 **t-8** 和 **m0**，在未提供参数时假定为该值。如果不指定 *s* 参数，就不会执行大小检查。如果文件的第一行不包含格式规范，那么整个文件就会采用上述的缺省值。以下是包含格式规范的行的一个示例：

<:t5,10,15 s72:>

如果格式规范能够伪装成注释，就没有必要对 *d* 参数编码。

## 标志

### **-T***Terminal*

标识终端以使 **tabs** 命令能够正确设置制表符和页边距。*Terminal* 变量是 **greek** 命令指定的终端之一。*Terminal* 变量支持的值包括：

**ANSI** 任何 ANSI 终端，如 VT100 终端。

**hp** Hewlett-Packard 硬拷贝终端。

**2621** Hewlett-Packard 2621。

**2640** Hewlett-Packard 2640。

**2645** Hewlett-Packard 2645。

**tabs** 命令支持的其他硬拷贝终端包括：

- 1620
- 1620-12
- 1620-12-8
- 1700
- 1700-12
- 1700-12-8
- 300
- 300-12
- 300s
- 300s-12
- 40-2
- 4000a
- 4000a-12
- 43
- 450
- 450-12
- 450-12-8
- tn1200
- tn300
- oki

如果不提供 **-T** 标志，则会使用环境变量 **TERM** 的值。如果提供了没有值的 **-T** 标志或 **-T** 和 **TERM** 具有无效值，则会显示错误消息 **unknown terminal**，并终止命令。

### **+m** *Number*

将所有的制表符移到 *Number* 变量指定的列数右边。该标志也设置 *Number* 变量指定的列的左页边距。如果指定的 **m** 没有值，那么 *Number* 变量的缺省值是 10。大多数工作站上最左边的页边距由 **+m0** 定义。制表符的第一列定义为列 0，而不是列 1。

**注：**如果相同的标志出现不止一次，那么只有最后的标志生效。

## 退出状态

该命令返回以下退出值:

- 0 成功完成。
- >0 发生错误。

## 示例

1. 要每四个空格设置制表符, 输入:

```
tabs -4
```

2. 要在 VT100 终端上每 10 个空格设置制表符, 输入:

```
tabs -10 -TANSI
```

## 文件

`/usr/bin/tabs` 包含 `tabs` 命令。

## 相关信息

`greek` 命令、`nroff` 命令以及 `troff` 命令。

---

## tail 命令

### 用途

显示文件的最后几行。

### 语法

#### 标准语法

```
tail [ -f ] [ -c Number | -n Number | -m Number | -b Number | -k Number ] [ File ]
```

#### 要以逆序显示行

```
tail [ -r ] [ -n Number ] [ File ]
```

### 描述

`tail` 命令从指定点开始将 *File* 参数指定的文件写到标准输出。如果没有指定文件, 则会使用标准输入。 *Number* 变量指定将多少单元写入标准输出。 *Number* 变量的值可以是正的或负的整数。如果值的前面有 + (加号), 从文件开头指定的单元数开始将文件写到标准输出。如果值的前面有 - (减号), 则从文件末尾指定的单元数开始将文件写到标准输出。如果值前面没有 + (加号) 或 - (减号), 那么从文件末尾指定的单元号开始读取文件。

*Number* 变量用于确定计数的起点的单元类型由 `-b`、`-c`、`-k`、`-m` 以及 `-n` 标志确定。如果没有指定其中的任何一个标志, 那么 `tail` 命令就会读取指定文件的最后十行, 并将其写到标准输出。这与在命令行输入 `-n 10` 是相同的。



**-m** 标志的单字节和双字节字符环境中提供了统一的结果。当输入是包含多字节字符的文本文件时应谨慎使用 **-c** 标志，因为产生的输出可能不从字符边界开始。

## 标志

- b** *Number* 从 *Number* 变量表示的 512 字节块位置开始读取指定文件。
- c** *Number* 从 *Number* 变量表示的字节位置开始读取指定文件。
- f** 如果输入文件是常规文件或如果 *File* 参数指定 FIFO（先进先出），那么 **tail** 命令不会在复制了输入文件的最后的指定单元后终止，而是继续从输入文件读取和复制额外的单元（当这些单元可用时）。如果没有指定 *File* 参数，并且标准输入是管道，则会忽略 **-f** 标志。**tail -f** 命令可用于监视另一个进程正在写入的文件的生长。
- k** *Number* 从 *Number* 变量表示的 1KB 块位置开始读取指定文件。
- m** *Number* 从 *Number* 变量表示的多字节字符位置开始读取指定文件。使用该标志提供在单字节和双字节字符代码集环境中的一致结果。
- n** *Number* 从首行或末行位置来读取指定文件，位置由 *Number* 变量的符号（+ 或 - 或无）表示，并通过行号 *Number* 进行位移。
- r** 从文件末尾以逆序方式显示输出。**-r** 标志的缺省值是以逆序方式显示整个文件。如果文件大于 20,480 字节，那么 **-r** 标志只显示最后的 20,480 字节。  
  
**-r** 标志只有与 **-n** 标志一起时才有效。否则，就会将其忽略。

## 退出状态

该命令返回以下退出值：

- 0** 成功完成。
- >0** 发生错误。

## 示例

1. 要显示 `notes` 文件的最后十行，输入：

```
tail notes
```

2. 要指定从 `notes` 文件末尾开始读取的行数，输入：

```
tail -n 20 notes
```

3. 要从第 200 字节开始，每次显示一页 `notes` 文件，输入：

```
tail -c +200 notes | pg
```

4. 要跟踪文件的生长，输入：

```
tail -f accounts
```

这显示 `accounts` 文件的最后十行。**tail** 命令继续显示添加到 `accounts` 文件中的行。显示会一直继续，直到您按下 **Ctrl-C** 按键顺序来停止。

## 文件

**/usr/bin/tail** 包含 **tail** 命令。

## 相关信息

**dd** 命令、**head** 命令、**more** 命令和 **pg** 命令。

《操作系统与设备管理》中的『文件』描述了文件、文件类型以及如何命名文件。

《操作系统与设备管理》中的『输入和输出重定向』描述了操作系统如何处理输入和输出。

---

## talk 命令

### 用途

与另一个用户对话。

### 语法

```
talk { User | User@Host | Host!User | Host.User | Host:User } [ Tty ] [ Pty ]
```

### 描述

**/usr/bin/talk** 命令允许同一主机或不同主机上的两个用户进行交互式对话。**talk** 命令在每个用户的显示器上打开发送窗口和接收窗口。这样每个用户在 **talk** 命令显示另一个用户正在输入的内容时能够在发送窗口输入。

为了启动对话，本地用户执行 **talk** 命令，并且指定远程用户的登录标识。远程用户的登录标识可包含 NLS 字符。如果远程用户在远程主机上，还必须以下列方式之一指定主机名：

```
User@Host  
Host!User  
Host.User  
Host:User
```

使用完整域名时，指定用户和主机的唯一有效格式是 *User @Host*。例如，`michael@host17.dev.ibm.com` 启动与 `dev.ibm.com` 域中的主机 `host17` 上的用户 `michael` 的对话。

当本地用户启动对话时，会向远程用户发送消息，邀请其参加对话。如果本地用户也指定了 `tty`，则只会向指定的终端发送邀请消息。否则，邀请会发送到远程用户的登录终端。这通常是控制台，但也可以是另一个终端。一旦接收到邀请，**talk** 命令会在本地用户的终端上显示两个窗口，并且显示进度信息，直到远程用户回应邀请。

**注：**如果远程用户正在运行 AIXwindows，并且没有打开其他终端，则 **talk** 命令无法发送邀请。

为开始对话，远程用户也必须在任一终端上执行 **talk** 命令，并且指定本地用户的帐户名称和主机名，如果适当的话。远程用户接受邀请时，**talk** 命令在每个用户的终端上显示两个窗口。其中一个窗口显示本地用户输入的内容；另一个窗口显示远程用户输入的内容。要结束对话，每个用户可按下中断（Ctrl-C）按键顺序，同时关闭连接。可以使用 **stty** 命令来显示和修改中断按键顺序。

如果对话中的用户使用本地语言支持（NLS）功能的话，那么他们的终端必须支持显示 NLS 字符。使用日语汉字功能的对话也是这样；所用的终端必须支持显示日语汉字字符。

**talk** 命令需要绑定到一个有效地址。远程机器的主机名必须绑定到工作网络接口，该接口可以被其他的网络命令，如 **ping** 命令使用。如果机器没有网络接口，即单机，那么必须将其主机名绑定到回送地址（127.0.0.1）以使 **talk** 命令可以运行。例如，在单机上名为 `local` 和 `remote` 的两个用户能够使用 **talk** 命令启动对话，输入：

```
talk remote@loopback
```

用户 remote 对此作出反应:

```
talk local@loopback
```

要禁止 **talk** 命令邀请, 远程用户可发出 **mesg** 命令。

注: **talk** 命令使用 Talk 4.3 协议, 它与 **talk** 命令的 AIX 4.2 版本不兼容。

## 示例

1. 要与登录到远程主机的用户交谈, 输入:

```
talk dale@host2
```

在该示例中, 本地用户想与登录到 host2 上的用户 dale 交谈。

2. 要只与在远程主机控制台上登录的用户交谈, 输入:

```
talk dale@host2 console
```

用户 dale 只有在 host2 的控制台上登录才能接收该消息。

## 相关信息

**mesg** 命令、**stty** 命令。

**named** 守护程序、**talkd** 守护程序。

《操作系统与设备管理》中的『通信与网络』。

《网络与通信管理》中的『与远程用户对话』。

---

## talkd 守护程序

### 用途

提供 **talk** 命令的服务器功能。

### 语法

```
/usr/sbin/talkd [ -s ]
```

### 描述

注: **talkd** 守护程序通常由 **inetd** 守护程序启动。也可使用 **SRC** 命令从命令行控制。

**/usr/sbin/talkd** 守护程序是通知一个用户 (接收方) 另一个用户 (呼叫方) 想启动对话的服务器。如果接收方接受邀请, 则守护程序就会建立对话。呼叫方通过执行指定接收方的 **talk** 命令启动对话。接收方通过执行指定呼叫方的 **talk** 命令接受邀请。

**talkd** 守护程序在 **/etc/services** 文件中定义的套接字上侦听。**talkd** 守护程序收到来自本地或远程 **talk** 进程的 **LOOK\_UP** 请求时, **talkd** 守护程序会扫描其内部邀请表以查找一个条目, 该条目将客户机进程 (本地或远程 **talk** 进程) 与呼叫方配对。

如果在邀请表里不存在条目, 那么 **talkd** 守护程序就会假定客户机进程是呼叫方。然后 **talkd** 守护程序收到客户机进程的 **ANNOUNCE** 请求。**talkd** 守护程序会在接收方首次登录的远程计算机上广播邀请 (除非呼叫方指定特别的 **tty** 设备)。该终端通常是控制台, 但也可以是另一个终端。

否则，就会将邀请发送到第二个用户首次登录的终端。这通常是控制台，但也可以是另一个终端。

如果在 **talkd** 守护程序的内部邀请表里不存在条目，那么 **talkd** 守护程序就会假定客户机是接收方。**talkd** 守护程序将适当的集合地址返回给接收方的 **talk** 进程。然后接收方进程与呼叫方进程建立流连接。

**注：****talkd** 守护程序使用 Talk 4.3 协议，这与 **talk** 进程的 AIX 4.2 版本不兼容。AIX 4.3 协议的子服务器名为 **ntalk**。

可以使用系统管理界面工具（SMIT）或系统资源控制器（SRC）编辑 **/etc/inetd.conf** 或 **/etc/services** 文件更改 **talkd** 守护程序。建议不要在命令行输入 **talkd**。**talkd** 守护程序在缺省情况下（在 **/etc/inetd.conf** 文件中没有注释掉该守护程序）会启动。

**inetd** 守护程序从 **/etc/inetd.conf** 文件和 **/etc/services** 文件中获取信息。

更改 **/etc/inetd.conf** 或 **/etc/services** 文件后，运行 **refresh -s inetd** 或 **kill -1 InetdPID** 命令通知 **inetd** 守护程序已更改其配置文件。

调试消息会发送到 **syslogd** 守护程序。

**注：****talkd** 守护程序应该使用系统管理界面工具（SMIT）或更改 **/etc/inetd.conf** 文件来控制。

## 用系统资源控制器操作 **talkd** 守护程序

**talkd** 守护程序是 **inetd** 守护程序的子服务器，后一守护程序是系统资源控制器（SRC）的子系统。**talkd** 守护程序是 **tcPIP** SRC 子系统组的成员。缺省情况下在 **/etc/inetd.conf** 文件中启用该守护程序，并可用下列的 SRC 命令对进其行操作：

|                 |                     |
|-----------------|---------------------|
| <b>startsrc</b> | 启动子系统、子系统组或子服务器。    |
| <b>stopsrc</b>  | 停止子系统、子系统组或子服务器。    |
| <b>lssrc</b>    | 获取子系统、子系统组或子服务器的状态。 |

## 标志

**-s** 打开套接字级别的调试。

## 示例

1. 要启动 **talkd** 守护程序，输入如下：

```
startsrc -t ntalk
```

该命令启动 **talkd** 子服务器。

2. 要正常停止 **talkd** 守护程序，输入如下：

```
stopsrc -t ntalk
```

该命令允许启动所有暂挂的连接并完成现有的连接，但会阻止启动新的连接。

3. 要强制停止 **talkd** 守护程序和所有 **talkd** 连接，输入如下：

```
stopsrc -t -f ntalk
```

该命令会立刻终止所有暂挂的连接和现有的连接。

4. 要显示关于 **talkd** 守护程序的简短状态报告，输入如下：

```
lssrc -t ntalk
```

该命令返回守护程序名、进程标识以及状态（活动的或不活动的）。

## 文件

**/etc/utmp** 包含关于当前登录的用户的数据。

## 相关信息

**kill** 命令、**lssrc** 命令、**refresh** 命令、**startsrc** 命令、**stopsrc** 命令以及 **talk** 命令。

**inetd** 守护程序、**syslogd** 守护程序。

**/etc/inetd.conf** 文件格式。

《网络与通信管理》中的『TCP/IP 守护程序』。

---

## tapechk 命令

### 用途

在流式磁带设备上执行一致性校验。

### 语法

```
tapechk [ -? ] Number1 Number2
```

### 描述

**tapechk** 命令在连接的流式磁带设备上执行基本的一致性校验。仅仅通过读取磁带就能够检测流式磁带机的某些硬件故障。**tapechk** 提供了在文件级别执行磁带读取的方法。

因为流式磁带机不能在物理数据块或文件上退格，所以 **tapechk** 命令会在每次检查前将磁带倒带到其起始位置。该命令要么检查 *Number1* 参数指定的下一文件号的数据，要么跳过 *Number2* 参数指定的下一文件号。如果您不指定任何参数，那么 **tapechk** 命令会将磁带倒带，并且只检查第一个物理块。

**tapechk** 命令使用 **TAPE** 环境变量（如果定义了该环境变量）中的设备。否则，缺省的磁带设备是 **/dev/rmt0**。

**注：****backup** 命令允许您选择性地压缩文件或压缩整个文件系统。采用以文件标记终止的持续流的形式写入数据，而不管指定的文件数。**tapechk** 命令将每个数据流理解为单一的文件，这一点在您指定数字参数时很重要。

虽然您能够在任何流式盒式磁带上使用 **tapechk** 命令，但它主要是设计用来检查 **backup** 命令所写的磁带。

## 标志

**-?** 说明 **tapechk** 命令的格式。

注：如果指定了 **-?** 标志，它必须在 *Number1* 和 *Number2* 参数之前指定。

## 退出状态

该命令返回以下退出值：

**0** 成功完成。  
**>0** 发生错误。

## 示例

要在流式磁带设备上检查前三个文件，输入：

```
tapechk 3
```

## 文件

`/usr/sbin/tapechk` 包含 `tapechk` 命令。

## 相关信息

`backup` 命令。

`rmt` 特殊文件。

《操作系统与设备管理》中的『磁带机』。

---

## tar 命令

### 用途

处理归档。

### 语法

**X/Open** 标准：

```
tar {-cl-ri-tl-ul-x} [-B] [-d] [-E] [-F] [-h] [-i] [-l] [-m] [-o] [-p] [-s] [-U] [-v] [-w]
    [-Number] [-f Archive] [-b Blocks]
    [-S [Feet] [Feet @Density] [Blocksb]] [-L InputList] [-X ExcludeList]
    [-N Blocks] [-R] [-D] [-C Directory] File | Directory ...
```

**Berkeley** 标准：

```
tar {c |r|t|u|l|x} [bBdDEfFh|LXmNopRsSUvw[0-9] ]
    [ Blocks ] [ Archive ] [ InputList ] [ ExcludeFile ]
    [ [ Feet ] | [ Feet@Density ] | [ Blocksb ] ] Directory | File ...
```

### 描述

**警告：** 由于在 `tar` 命令中对头块空间的限制，当恢复到某些系统时，大于 65,535 的用户号（UID）和组标识号（GID）将毁坏。大小约束只影响所有权和许可权而不对数据造成损害。所有权的毁坏发生在下列系统中：

- 不使用 `uname` 和 `gname` 字段来检查所有权的系统。
- 与归档系统没有相同的用户和组标识的系统。

注:

1. **ustar** 头格式允许文件大到 8 GB。因此, **tar** 命令支持归档最大 8 GB 的文件。
2. **tar** 不保留任何稀疏分配的文件稀疏属性。任何在恢复前原是稀疏的文件, 将按文件大小在文件系统内分配其所有空间。

**tar** 命令通过将文件写入归档存储介质或将文件从介质中取回, 来处理归档。由 **tar** 命令使用的文件通过 *File* 参数来表示。如果 *File* 参数引用一个目录, 则引用该目录并递归引用该目录下全部的文件和目录。

**tar** 命令在缺省设备(通常是磁带)上寻找归档, 除非使用 **-f Archive** 标志指定另一个设备。当为美国磁带归档程序(USTAR)格式指定大于 100 个字符的路径名称时, 请记住路径名称是由一个前缀缓冲区、一个 / (斜杠) 和一个名称缓冲区组成的。

前缀缓冲区最大可以有 155 字节, 而名称缓冲区最大可以容纳 100 字节。如果路径名称不能用斜杠分割为这两部分, 则它不能被归档。这个限制是考虑到 **tar** 归档头的结构, 而且必须为了符合标准和向下兼容而维护该限制。另外, 硬链接或符号链接(“链接名称”)的目标长度不能超出 100 字节。

当写到归档时, **tar** 命令使用一个临时文件(/tmp/tar\* 文件), 并在内存中维护一个包含多链接文件的表。如果 **tar** 命令不能创建临时文件, 或者没有足够的可用内存来放置链接表, 您会接收到错误消息。

**tar** 命令有两组标志: 必需标志和可选标志。必需标志控制 **tar** 命令的操作, 包括 **-c**、**-r**、**-t**、**-u** 和 **-x** 标志。要执行 **tar** 命令至少要选择一个必需标志。选择了必需标志后, 您可以选择一个可选标志, 但它们对控制 **tar** 命令都不是必要的。

注:

1. 当存储设备是普通文件或块特殊文件时, **-u** 和 **-r** 标志将退格。然而, 原始磁带设备不支持退格。因此当存储设备是原始磁带时, **-u** 和 **-r** 标志进行倒带、打开磁带并再次读取磁带。
2. 在块磁带上记录的长度是一块, 但它们通常比在原始磁带上的密度的一半还少。因此, 尽管按块记录原始磁带必须被读取两遍, 但磁带运动总量比从块式磁带读取一遍单块记录还少。
3. 流式磁带设备的结构不支持在磁带最后增加信息。因此, 当存储设备是流式磁带时, **-u** 和 **-r** 标志不是有效的选项。试图使用这些标志将导致下列错误消息:

`tar: 更新和替换选项对流式磁带机无效。`

4. 磁带错误无法恢复。
5. 可以通过更改缺省块大小来提高 **tar** 命令在 IBM 9348 Magnetic Tape Unit Model 12 上的性能。要更改块大小, 请在命令行输入下列命令:

```
chdev -l <device_name> -a block_size=32k
```

要了解更多关于使用磁带设备的信息, 请参阅 **rmt** 特殊文件。

## 标志

**tar** 命令的标志有两组, 必需的和可选的。必须提供至少一个必需标志来控制 **tar** 命令。

### 必需标志

- c** 创建新的归档, 并将由一个或多个 *File* 参数所指定的文件写入归档的开头。
- r** 将由一个或多个 *File* 参数所指定的文件写入归档的最后。这个标志不是对任何磁带设备都有效, 因为有些设备不支持在磁带的最后添加信息。
- t** 按照文件在归档中的顺序列出文件。文件可能会多次列出。

- u** 只有当文件不在归档中，或写入归档后被修改过，才将这些文件（由一个或多个 *File* 参数所指定）写入归档的最后。**-u** 标志不是对任何磁带设备都有效，因为有些磁带设备不支持在磁带的最后添加信息。
  - U** 允许对扩展属性进行归档和抽取。扩展属性还包括访问控制列表（ACL）。
  - x** 从归档中抽取一个或多个 *File* 参数指定的文件。如果 *File* 参数是指一个目录，**tar** 命令递归地从归档中抽取那个目录。如果您未指定 *File* 参数，**tar** 命令从归档中抽取所有的文件。如果归档中包含同一文件的多个副本，最后抽取的副本覆盖所有以前抽取的副本。如果被抽取的文件在系统中不存在，则创建该文件。如果您有适当的许可权，**tar** 命令用与磁带上的相同所有者和组标识恢复所有文件和目录。如果您没有适当的许可权，将以您的所有者标识和组标识恢复文件和目录。只能请求文件最后出现的版本。
- 可选标志
- B** 输入分块和输出分块强制为每个记录 20 块。使用这个选项，**tar** 命令可在那些可能未维护分块的通信信道间执行。
  - b Blocks** 指定每个记录的 512 字节块数量。缺省值和最大值都是 20，这是对磁带记录适当的数目。由于记录间隙大小的原因，用大的分块因子写的磁带可以比每个记录只有一块的磁带保存更多的数据。
- 当读取磁带时，块大小自动确定（**-x** 或 **-t** 功能标志）。归档被 **-u** 和 **-r** 功能更新时，使用现有的记录大小。只有当使用 **-c** 标志创建新的归档时，**tar** 命令才使用 *Blocks* 参数指定的值写入归档。
- C Directory** 当用 **-f** 标志输出到普通文件时，您可以使用与磁盘块大小相匹配的分块因子来节省磁盘空间（例如，对于 2048 字节的磁盘块，使用 **-b4** 标志）。  
使 **tar** 命令向由 *Directory* 变量指定的目录执行 **chdir** 子例程。使用 **-C** 标志允许归档不用相近公共父目录来关联的多个目录，使用简短的相对路径名。例如，要从 **/usr/include** 和 **/etc** 目录中归档文件，可以使用下列命令：
- ```
tar c -C /usr/include File1 File2 -C /etc File3 File4
```
- C Directory** 标志必须在所有其他标志后出现，也可以在给定的文件名列表中出现。
  - D** 指定目录时，禁止递归处理。
  - d** 为块文件、特殊字符文件以及先进先出（FIFO）管道进程创建单独条目。通常，**tar** 命令不会归档这些特殊文件。当使用 **-d** 标志写入归档时，**tar** 命令使归档可以用 **-x** 标志恢复空目录、特殊文件以及先进先出（FIFO）管道进程。  
**注：**虽然任何人都可以归档特殊文件，但只有拥有 root 用户权限的用户可以从归档中抽取特殊文件（FIFO 还可以由非 root 用户抽取）。
  - E** 在将文件添加到新的或现有的归档的过程中，避免截断较长的用户和组名。
  - F** 在归档前检查文件类型。源代码控制系统（SCCS），修订控制系统（RCS），名为 **core**、**errs** 和 **a.out** 的文件，以及以 **.o**（点 o）结束的文件不归档。
  - f Archive** 使用 *Archive* 变量表示要读或写的归档。当未指定这个标志，**tar** 命令使用一个针对特定系统的缺省文件名（格式为 **/dev/rmt0**）。如果指定的 *Archive* 变量是 -（减号），**tar** 命令写入标准输出或从标准输入中读取。如果写入标准输出，您必须使用 **-c** 标志。
  - h** 强制 **tar** 命令监视符号链接，就好像它们是常规文件或目录。通常，**tar** 命令不监视符号链接。
  - i** 忽略头校验和的错误。**tar** 命令在归档中写入一个文件头，它包含每个文件的校验和。如果未指定这个标志，系统通过重新计算校验和来验证头块的内容，并在发生不匹配时，停止并报目录校验和错误。如果指定这个标志，**tar** 命令将记录错误，然后向前扫描直到查找到一个有效的头块。这就允许从多卷归档的后面的卷中恢复文件，而不需读取前面的卷。
  - L InputList** **-L** 选项的 *Inputlist* 自变量应该始终是文件的名称，该文件列出需要归档或抽取的文件和目录。
  - l** 对于每个带有链接数大于 1 的文件，而其相应的链接却未归档，写错误消息到标准输出。例如，如果 **file1** 和 **file2** 是用硬链接链接在一起，而只有 **file1** 存放在归档中，则 **-l** 标志将发出错误消息。如果未指定 **-l** 标志，将不显示错误消息。
  - m** 使用抽取时间作为修改时间。缺省是保留文件的修改时间。



|                                                                                          |                                                                                                                                                                                                 |
|------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-N</b> <i>Blocks</i>                                                                  | 当 <b>tar</b> 命令处理流式磁带归档时，允许它使用很大的块的簇。然而请注意，在输入时，如果用这个标志创建了非常长的块大小， <b>tar</b> 命令将不能自动确定磁带的块大小。没有 <b>-N</b> <i>Blocks</i> 标志时， <b>tar</b> 命令可以自动确定的最大块大小为 20 块。                                  |
| <b>-o</b>                                                                                | 提供与旧版本（非 AIX） <b>tar</b> 命令的向下兼容。当使用这个标志进行读取时，被抽取的文件使用运行程序的用户的用户标识和组标识（UID 和 GID），而不是归档中的用户标识和组标识。这是对于普通用户的缺省行为。                                                                                |
| <b>-p</b>                                                                                | 恢复字段到它们的原始方式，忽略现有的用户权限屏蔽位（umask）。 <b>setuid</b> 、 <b>setgid</b> 和 tacky 位许可权也恢复给拥有 root 用户权限的用户。这个标志恢复文件到其原始方式，但不恢复目录到其原始方式。                                                                    |
| <b>-R</b>                                                                                | 指定目录时，使用递归。与 <b>-D</b> 选项一起使用时则忽略。                                                                                                                                                              |
| <b>-s</b>                                                                                | 如果 <b>tar</b> 命令未能使用 <b>-s</b> 标志成功尝试链接（常规链接）两个文件，请试着创建符号链接。                                                                                                                                    |
| <b>-S</b> <i>Blocks</i> <b>b</b> , <b>-S</b> <i>Feet</i> , <b>-S</b> <i>Feet@Density</i> | 指定每卷中的 512 KB 块的数目（第一种格式），不依赖于磁带分块因子。您也可以使用第二种格式，用英尺来指定磁带大小，在这种情况下， <b>tar</b> 命令假定一个缺省的 <i>Density</i> 变量。第三种格式允许您指定磁带长度和密度。英尺被保守地假定为 11 英寸长。这个标志让您更容易处理多卷磁带归档，这时 <b>tar</b> 命令必须能够确定每卷能放入多少块。 |

注:

1. 磁带机在密度能力上有差异。*Density* 变量计算系统在磁带上可放入的数据量。
2. 如果使用 1/4 英寸磁带设备，在指定 *Feet* 变量的值时，一定要考虑磁带设备上的磁道数目。例如，可以使用 **-S** *Feet@Density* 标志来指定一个有 600 英尺磁带和 8000 字节/英寸密度的 4 磁道、1/4 英寸磁带机，如下：  

```
-S 2400@8000
```

这里，600 英尺乘以 4 磁道等于 2400 英尺。

|                              |                                                                                                                                                                               |
|------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-U</b>                    | 对命名的扩展属性和 ACL 进行归档或恢复。列出时，此选项将显示任何命名的扩展属性的名称和任何与每个文件相关的 ACL 类型，它们是归档图像的一部分。                                                                                                   |
| <b>-v</b>                    | 在处理每个文件时，列出它们的名称。使用 <b>-t</b> 标志， <b>-v</b> 给出磁带条目的更多信息，包括文件大小、上次修改的时间、用户标识（UID）、组标识（GID）和许可权。                                                                                |
| <b>-w</b>                    | 显示要进行的操作，跟着是文件名，然后等候用户确认。如果回答是肯定的，则执行操作。如果回答不是肯定的，则忽略文件。                                                                                                                      |
| <b>-Number</b>               | 使用 <b>/dev/rmtNumber</b> 文件，而不是缺省值。例如， <b>-2</b> 标志和 <b>-f/dev/rmt2</b> 文件是相同的。                                                                                               |
| <b>-X</b> <i>ExcludeList</i> | 从正在创建、抽取或列出的 <b>tar</b> 归档排除 <i>ExcludeList</i> 给定的文件名或目录。 <i>ExcludeList</i> 的每一行应该只包含一个文件名或目录，这些文件名或目录将从正在创建、抽取或列出的 <b>tar</b> 归档排除出去。可以多次指定 <b>-X</b> 选项，并且它的优先顺序高于所有其他选项。 |

## 退出状态

该命令返回以下退出值:

- 0 成功完成。
- >0 发生错误。

## 示例

1. 要将 *file1* 和 *file2* 文件写入缺省磁带机上新的归档中，请输入:

```
tar -c file1 file2
```

2. 要从 **/dev/rmt2** 磁带设备上的归档文件中抽取 **/tmp** 目录里的全部文件，并以抽取时间作为修改时间，请输入:

```
tar -xm -f/dev/rmt2 /tmp
```

3. 要新建一个包含 `file1` 文件的归档文件，并将此归档文件传递到将写入 `/dev/rmt1` 设备的 `dd` 命令，请输入：

```
tar -cvf - file1 | dd of=/dev/rmt1 conv=sync
```

4. 要显示当前目录下 `out.tar` 磁盘归档文件中文件的名称，请输入：

```
tar -vtf out.tar
```

5. 要扩展压缩的 `tar` 归档文件 `fil.tar.z`，将文件传递到 `tar` 命令，并从扩展的 `tar` 归档文件中抽取所有文件，请输入：

```
zcat fil.tar.Z | tar -xvf -
```

6. 要使用简短的相对路径名归档 `/usr/include` 和 `/usr/bin` 文件的内容，请输入：

```
cd /usr
tar -cvf/dev/rmt0 -C./include . -C ../bin .
```

注：当用相对路径名指定 `-C` 标志的多个实例时，用户必须考虑到先前的 `-C` 标志请求。

7. 要在使用 `-S` 标志时归档到 8 毫米磁带设备，请输入：

```
tar -cvf /dev/rmt0 -S 4800000b /usr
```

注：归档到 8 毫米磁带设备时，建议不要使用 `-S Feet` 和 `-S Feet@Density` 标志，因为 8 毫米磁带设备在写入磁带时不使用密度概念。

8. 要归档所有 C 文件的一个列表（这些文件通过 `-L` 选项的 `InputList` 参数被列在文件中），请输入：

```
tar -cvf fl.tar -L fl_list
```

这里，`fl_list` 是一个文件，它包含一个所有 `.c` 文件的列表。这可以通过以下命令获得：

```
ls *.c > fl_list
```

9. 要使用 `-L` 选项设置变量来归档所有 C 文件的一个列表，请输入：

```
ls *.c > fl_list
fl=fl_list
tar -cvf var.tar -L $fl
```

10. 要在创建归档过程中避免截断较长的用户或组名，请输入：

```
tar -cvEf file.tar file
```

11. 要新建包含带有 ACL 和 EA 的 `file1` 文件的归档文件，请输入：

```
tar -cvUf /tmp/tar.ar file1
```

## Berkeley 选项

以下是使用 `tar` 命令的 Berkeley 选项的示例：

注：带有 Berkeley 选项时，标志的自变量应该以下面标志排列的顺序进行准确排列。例如：

```
tar cvfbL test.tar 20 infile
```

其中 `test.tar` 是归档 `tar` 文件，`20` 是块的数目，`infile` 是此归档的 `Inputlist`。

1. 要将在输入列表文件 `infile` 中列出的所有目录和完整文件名归档到 `ar.tar`，请输入：

```
tar cvfL ar.tar infile
```

其中 `infile` 包含了要归档的文件的名称。

2. 要将在输入列表文件 `infile` 中列出的目录中的文件归档到 `ar.tar`，请输入：

```
tar cvRfL ar.tar infile
```

3. 要从名为 `ar.tar` 的归档中抽取在输入列表文件 `infile` 中指定的目录和完整文件，请输入：

```
tar xvfl ar.tar infile
```

4. 要从名为 **ar.tar** 的归档中抽取在输入列表文件 **infile** 中指定的目录和全部文件中的文件，请输入：

```
tar xvRfL ar.tar infile
```

## 文件

|                     |                        |
|---------------------|------------------------|
| <b>/dev/rmt0</b>    | 指定缺省磁带设备。              |
| <b>/bin/tar</b>     | 指定 <b>tar</b> 命令的符号链接。 |
| <b>/usr/bin/tar</b> | 包含 <b>tar</b> 命令。      |
| <b>/tmp/tar*</b>    | 指定临时文件。                |

注：在 AIX 3.2 中，整个 **/bin** 目录是对 **/usr/bin** 的符号链接。

## 相关信息

**cat** 命令、**dd** 命令、**pax** 命令。

**rmt** 特殊文件。

《操作系统与设备管理》中的『文件系统』说明了文件系统类型、管理、结构以及维护。

《操作系统与设备管理》中的『目录概述』说明了如何使用目录和路径名称。

《操作系统与设备管理》中的『文件』提供了使用文件的信息。

---

## tbl 命令

### 用途

为 **nroff** 和 **troff** 命令格式化表。

### 语法

```
tbl [ -TX ] [ - ] [ File... | - ]
```

### 描述

**tbl** 命令是个预处理器，它为 **nroff** 和 **troff** 命令格式化表。它读取一个或多个文件。如果没有指定 *File* 参数或 **-**（减号）被指定为最后参数，缺省情况下，命令读取标准输入。除了包含有 **.TS** 和 **.TE** 的行之间的文本外，命令将输入原封不动地复制到标准输出中。**tbl** 命令重新格式化此类描述表的文本，而不改变 **.TS** 和 **.TE** 行。

根据目标输出设备，由 **nroff** 命令格式化了了的输出可能需要用 **col** 命令再处理以产生正确的输出。

注：要使通过管道的数据量最小化，则在与 **eqn** 或 **neqn** 命令一起使用时，请首先输入 **tbl** 命令。

### 输入格式

**tbl** 命令处理以下列格式显示的文本：

[ {**.DS** **.DF** } ]  
**.TS**  
 选项;  
 格式。  
 数据  
**.TE** [**.DE**]

要想将短表包括在 **mm** 宏文件中, 请将它们包括在 **.DS** (或 **.DF**) 和 **.DE** 宏对中。

## 选项

下面是输入格式的可用全局选项:

| 选项                                                                    | 用途                                            |
|-----------------------------------------------------------------------|-----------------------------------------------|
| <b>center</b> 或 <b>CENTER</b>                                         | 使行居中。                                         |
| <b>expand</b> 或 <b>EXPAND</b>                                         | 扩展行的长度。                                       |
| <b>box</b> 或 <b>BOX</b>                                               | 围在框中。                                         |
| <b>allbox</b> 或 <b>ALLBOX</b>                                         | 框入所有条目。                                       |
| <b>doublebox</b> 或 <b>DOUBLEBOX</b>                                   | 围在两个框中。                                       |
| <b>tab</b> ( <i>Character</i> ) 或 <b>TAB</b> ( <i>Character</i> )     | 将制表符改为 <i>Character</i> 值。                    |
| <b>linesize</b> ( <i>Number</i> ) 或 <b>LINESIZE</b> ( <i>Number</i> ) | 将所有行高变为 <i>Number</i> 值指定的点数。                 |
| <b>delim</b> ( <i>XY</i> ) 或 <b>DELIM</b> ( <i>XY</i> )               | 将 <i>X</i> 和 <i>Y</i> 的变量标为 <b>eqn</b> 命令定界符。 |
| ;                                                                     | 指示选项结束。                                       |

## 格式

输入格式中 *Format* 的变量描述了文本的格式。每个格式行 (该行的最后必须以句点结束) 描述了表中所有的剩余行。单键字母描述表中每行的每列。这个键字母后有说明符, 它确定对应项的字体和大小, 指示竖条显示在列间的什么地方并确定一些诸如宽度和列间距等事项。下面就是可用的键字母:

**l** 或 **L**  
**r** 或 **R**  
**c** 或 **C**  
**n** 或 **N**

列左对齐。  
 列右对齐。  
 列居中。  
 按数字对齐列。

**注:** 按数字对齐数据、**n** 或 **N** 格式规格, 要根据为 **RADIXCHAR** 指定的语言环境。语言环境被假定为单字符。对齐也可以用 \& (反斜杠、& 符号) 字符序列来决定, 这种字符序列独立于任何 **RADIXCHAR** 字符存在。如果在数字对齐区显示了几个 **RADIXCHAR** 字符, 用最后一个对齐。如果有特定的列中没有显示 **RADIXCHAR** 字符, 请按最后一个 ASCII 阿拉伯数字对齐。如果列中没有 ASCII 数字和 **RADIXCHAR** 字符, 则数据居中。

**a** 或 **A**  
**s** 或 **S**  
**t** 或 **T**  
**v** 或 **V**  
<sup>^</sup>  
**u** 或 **U**  
**z** 或 **Z**  
 -  
 =  
 |  
 ||  
**b** 或 **B**

子列左对齐。  
 水平横跨项。  
 将垂直范围推到顶端。  
 调整行间距。  
 垂直跨项。  
 将项上移半行  
 显示宽度为 0 的项。  
 显示水平行。  
 显示双倍水平行。  
 显示垂直行。  
 显示双倍垂直行。  
 显示加粗项。

|                                                         |                                  |
|---------------------------------------------------------|----------------------------------|
| <b>i</b> 或 <b>I</b>                                     | 显示斜体项。                           |
| <b>f</b> <i>Character</i> 或 <b>F</b> <i>Character</i>   | 将字体更改为 <i>Character</i> 变量指定的字体。 |
| <b>p</b> <i>Number</i> 或 <b>P</b> <i>Number</i>         | 将大小变为 <i>Number</i> 变量指定的大小。     |
| <b>w</b> ( <i>Number</i> ) 或 <b>W</b> ( <i>Number</i> ) | 将最小列宽的值设为与 <i>Number</i> 变量值相等。  |
| <i>NumberNumber</i>                                     | 列间距。                             |
| <b>e</b> 或 <b>E</b>                                     | 等分列宽。                            |
| .                                                       | 结束格式化。                           |

## 数据

要想用输入格式处理数据，尤其是表，请用下列行命令：

|                            |                                     |
|----------------------------|-------------------------------------|
| <b>T{...T}</b>             | 如下显示文本块：                            |
|                            | <i>Data</i> <TAB> <b>T</b> {        |
|                            | <i>Text Block</i>                   |
|                            | <b>T</b> }<TAB> <i>Data</i>         |
| <b>\</b>                   | 写短水平行。                              |
| <b>\RX</b>                 | 跨列重复 <i>X</i> 参数值。                  |
| <b>\^</b>                  | 指示前面所列的项向下延伸到该行。                    |
| <b>.T&amp;</b>             | 启动新格式。                              |
| <b>.TS H, .TH, and .TE</b> | 允许多页表在每页上重复列标题。（这是 <b>mm</b> 宏的特征。） |

## 参数

*File* 指定 **tbl** 命令要处理的文件。

## 标志

|            |                                   |
|------------|-----------------------------------|
| <b>-TX</b> | 只用整行垂直移动使输出适合行打印机和其他没有部分行垂直移动的设备。 |
| <b>—</b>   | （双破折号）表示标志结束。                     |
| <b>-</b>   | 强制从标准输入中读取输入。                     |

## 示例

下列示例显示了 **tbl** 命令的编码输入及其相关的表输出。 @ (at 符号) 用于在输入中代表输入制表符。

### 输入

```
.TS
center box ;
cB s s
cI | cI s
^ | c c
l | n n .
Household Population

Town@Households
@Number@Size
=
Bedminster@789@3.26
```

Bernards Twp.@3087@3.74  
Bernardsville@2018@3.30  
Bound Brook@3425@3.04  
Bridgewater@7897@3.81  
Far Hills@240@3.19  
.TE

## 相关信息

**col** 命令、**eqn** 命令、**mm** 命令、**mmt** 命令、**mvt** 命令、**neqn** 命令、**nroff** 命令、**soelim** 命令和 **troff** 命令。

**mm** 宏信息包、**mv** 宏信息包。

---

## tc 命令

### 用途

为 Tektronix 4015 系统解释文本到 **troff** 命令输出。

### 语法

```
tc [ -t ] [ -e ] [ -a Number ] [ -o List | -s Number ] [ - ] [ File | - ]
```

### 描述

**tc** 命令将输入解释为 **troff** 命令的输出。**tc** 命令读取一个或多个英语文件。如果未指定文件或 **-**（减号）标志被指定为最后一个参数，则缺省读取标准输入。**tc** 命令的标准输出将给 Tektronix 4015（一个带 ASCII 码和 APL 字符集的 Tektronix 4014 终端）使用。各种排版机的大小都映射到 4014 的四个大小。整个 **troff** 命令字符集可以使用 4014 字符生成器来生成，并在必要处使用叠印合并。

在每页的结束，**tc** 命令都会在继续下页之前等待键盘输入一个换行字符。在等待时，会识别下列命令：

|                 |                               |
|-----------------|-------------------------------|
| <b>!Command</b> | 发送 <i>Command</i> 变量值到 shell。 |
| <b>-e</b>       | 在每页之前不要擦除。                    |
| <b>-Number</b>  | 往回跳过指定页数。                     |
| <b>-aNumber</b> | 设置宽高比为 <i>Number</i> 变量值。     |
| <b>?</b>        | 显示可用的选项列表。                    |

注：**tc** 命令不区分字体。

### 参数

*File* 指定要解释为 **troff** 命令输出的英文文本文件。

### 标志

|                         |                                                                                                                                                               |
|-------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-a</b> <i>Number</i> | 将宽高比设置为指定数字。缺省值为 1.5。                                                                                                                                         |
| <b>-e</b>               | 在每页之前不要擦除。                                                                                                                                                    |
| <b>-o</b> <i>List</i>   | 只显示在 <i>List</i> 变量中列举的页数。列表包含显示页面以及页范围（例如，5-17），以逗号隔开。 <i>Number-</i> 的页码范围为从 <i>Number</i> 变量值一直到结束； <i>-Number</i> 的页码范围为从开始一直到 <i>Number</i> 变量指定的页并包含该页。 |

**-s** *Number*        跳过第一个指定的页数。  
**-t**                在定向输出到一个文件中时，页之间无需等待。  
**-**                从标准输入读取。  
**—**                (双破折号)表示标志结束。

## 示例

要在管道中与 **troff** 命令一起使用 **tc** 命令，输入：

```
troff [Flag...] [File...] | tc
```

## 相关信息

**nroff** 命令、**troff** 命令。

---

## tcbck 命令

### 用途

审计系统的安全状态。

### 语法

#### 检测方式

```
tcbck { -n | -p | -tl -y } [ -i ] [-o] { ALL | tree | { Name ... Class ... } }
```

#### 更新方式

```
tcbck -a -f File | PathName Attribute = Value ...
```

或

```
tcbck -d -fFile | { PathName ... | Class ... }
```

或

```
tcbck -l /dev/filename /dev/filename
```

### 描述

**tcbck** 命令通过检测在 **/etc/security/sysck.cfg** ( **sysck** 数据库) 文件中定义的文件安装来审计系统的安全状态。**/etc/security/sysck.cfg** 文件之中的每个文件定义可以包含一个或多个描述正确安装的属性。当不带标记和参数调用时，**tcbck** 命令显示其语法摘要。

**tcbck** 数据库通常定义作为可信计算库的一部分的所有文件和程序，但是 **root** 用户或安全组的成员可以选择只定义那些认为与安全相关的文件。

注：该命令将其消息写入 **stderr**。

### 标志

**-a**                添加或更新 **sysck** 数据库中的文件定义。  
**-d**                从 **sysck** 数据库中删除文件定义。

|                       |                                                                           |
|-----------------------|---------------------------------------------------------------------------|
| <b>-f</b> <i>File</i> | 指定文件定义从 <i>File</i> 读入。                                                   |
| <b>-i</b>             | 当指定 <b>tree</b> 选项后，排除列在 <b>treeck_nodir</b> 属性中的目录下的文件系统。                |
| <b>-l</b>             | (小写的 <b>L</b> ) 为管理员希望注册到可信计算库的 <b>/dev/</b> 文件增加条目到 <b>sysck.cfg</b> 文件。 |
| <b>-n</b>             | 指定检测方式，并指明将报告但不修订错误。                                                      |
| <b>-o</b>             | 写输出到 <b>syslog</b> 。                                                      |
| <b>-p</b>             | 指定检测方式，并指明将修订但不报告错误。                                                      |
| <b>-t</b>             | 指定检测方式，指明将报告错误并提示询问是否应该修订错误。                                              |
| <b>-y</b>             | 指定检测方式，并指明将修订并报告错误。                                                       |

## 操作方式

**tcbck** 命令有两种操作方式：检测方式和更新方式。下面是它们各自的描述。

### 检测方式

在检测方式中，**tcbck** 命令检测已安装文件的定义。可以通过指定 **ALL** 值来检测 **sysck** 数据库 (**/etc/security/sysck.cfg** 文件) 中的所有文件定义，或通过指定 **tree** 值检测文件系统树中的所有文件。如果想要检测特定文件，可以使用 **Name** 参数来给出单个文件的路径名或给出 **Class** 参数将若干文件打成一个由类名定义的逻辑包，例如 **audit**。必须从以下选择其一：**ALL** 或 **tree** 值，或由 **Class** 或 **Name** 参数标识的一个或多个文件。

如果 **tree** 值为选择条件，则检测所有在文件系统树中的文件来确保所有相关文件已在 **sysck** 数据库中定义。对定义在 **tcbck** 数据库中的文件则按其定义检测。不在 **tcbck** 数据库中的文件一定不可：

- 设置可信计算库属性。
- **setuid** 或 **setgid** 到管理标识。
- 链接到 **tcbck** 数据库中的文件。
- 作为设备特别文件。

如果 **tcbck** 命令以检测方式运行并带有 **tree** 值和 **-t** 标记时出现错误，该命令会提供一个出错消息，并提示您确定如何或是否应纠正该错误。如果决定不删除该文件或关闭非法许可，则将提示您确定更新该数据库。如果请求更新，系统提供缺少的信息，诸如文件名、链接或未注册的设备名。

还必须包括一个标记 (**-n**、**-p**、**-t**、**-y**) 来指定检测方式并标识错误处理方法。如果文件 **/etc/security/sysck.cfg** 中有一个重复节，将报告一个错误，但并不修订。

更新关键产品数据库 (VPD) 包括定义 VPD 管理器每一个文件的 **type**、**checksum** 和 **size** 属性。该信息用于验证正确的安装。如果这些属性未在 **-f File** 中定义，则将在程序安装或更新时计算它们。**checksum** 属性将用一个专门为 VPD 管理器定义的方法来计算。有关文件属性的更多信息，请参阅第 316 页的『修订错误』。

在更新时修改的文件定义仅是指示文件为可信计算库 (TCB) 的一部分的新定义。**File** 参数是一个节文件，包括 **tcbck** 格式的文件定义，在文件 **/etc/security/sysck.cfg** 中定义。当更新完成时，按节文件中的文件定义检测文件，并修订和报告错误。

要求 **setuid** 或 **setgid** 特权的程序必须在 **tcbck** 数据库里，否则当 **tcbck** 命令以检测方式运行时这些特权会被清除。

### 更新方式

在更新方式中，**tcbck** 命令为 **File**、**PathName** 或 **Class** 参数所指定的文件添加 (**-a**)、删除 (**-d**) 或修改 **/etc/security/sysck.cfg** 文件中的文件定义。**Class** 参数允许您将若干文件打成一个由类名定义的逻辑包，例如 **audit**。**tcbck** 命令还删除 **/etc/security/sysck.cfg** 文件中指定节。



在更新方式中, **tcback** 命令 (**-l**) 为指定的 **/dev** 条目增加或修改在 **/etc/security/sysck.cfg** 文件中定义的 **/dev/** 条目。该标记应由管理员运行, 用来添加新创建的并为 **sysck.cfg** 文件所信任的设备。在新设备未添加到 **sysck.cfg** 文件时, **tree** 选项生成未注册设备的警告。

**-l** 标记为命令行上列出的每一 **/dev/** 条目创建一个节。节的信息来自 **/dev** 条目的当前状态。节包括:

|       |                                                                                                                   |
|-------|-------------------------------------------------------------------------------------------------------------------|
| 设备名   | <b>/dev/</b> 条目名                                                                                                  |
| 文件类型  | <b>FILE</b> 、 <b>DIRECTORY</b> 、 <b>FIFO</b> 、 <b>SYMLINK</b> 、 <b>BLK_DEV</b> 、 <b>CHAR_DEV</b> 或 <b>MPX_DEV</b> |
| 所有者标识 | 所有者名                                                                                                              |
| 组标识   | 组名                                                                                                                |
| 许可权   | 所有者、组和其他人的读 / 写 / 执行许可权。 <b>SUID</b> 、 <b>SGID</b> 、 <b>SVTX</b> 和 <b>TCB</b> 属性位                                 |
| 目标    | 如果文件为符号链接, 将列出目标文件。                                                                                               |

要以 **-a** 标记添加或修改的文件定义可以在命令行上指定, 或在文件中以 **Attribute=Value** 语句指定。可使用下列属性:

|                 |                                                                                                                                                 |
|-----------------|-------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>acl</b>      | 文件的访问控制表。如果该值为空, 则除去 <b>acl</b> 属性。如果没有指定值, 则命令会根据访问控制表中描述的格式计算值。                                                                               |
| <b>class</b>    | 文件的逻辑组。必须指定一个值, 因为无法计算出来。如果该值为空, 则从指定的文件节除去 <b>class</b> 属性。该值是 <b>ClassName [ClassName ]</b> 。                                                 |
| <b>checksum</b> | 文件的校验和。如果该值为空, 则除去 <b>checksum</b> 属性。如果未指定值, 则该命令根据 <b>sum</b> 命令指定的格式计算值。该值为 <b>sum -r</b> 命令的输出, 包含空格。                                       |
| <b>group</b>    | 文件组。如果该值为格, 则除去 <b>group</b> 属性。如果未指定值, 则该命令计算一个值, 该值可以是组标识, 也可以是组名。                                                                            |
| <b>links</b>    | 指向该文件的硬链接。如果该值为空, 则除去 <b>links</b> 属性。必须指定一个值, 因为无法计算出来。该值必须为绝对路径名, 表示为 <b>Path [, Path ...]</b> 。                                              |
| <b>mode</b>     | 文件方式。如果该值为空, 则除去 <b>mode</b> 属性。如果没有指定值, 则该命令计算值, 它可以是八进制数字或字符串 ( <b>rxw</b> ), 并具有属性 <b>tcback</b> 、 <b>SUID</b> 、 <b>SGID</b> 和 <b>SVTX</b> 。 |
| <b>owner</b>    | 文件所有者。如果该值为空, 则除去 <b>owner</b> 属性。如果没有指定值, 则该命令会计算值, 它可以是用户标识或用户名。                                                                              |
| <b>program</b>  | 文件关联的检测程序。如果该值为空, 则除去 <b>program</b> 属性。必须指定一个值, 因为无法计算出来。该值必须为绝对路径名。如果指定了标志, 则该值应表示为 <b>Path, Flag</b> 。                                       |
| <b>symlinks</b> | 指向文件的符号链接。如果该值为空, 则除去 <b>symlinks</b> 属性。必须指定一个值, 因为无法计算出来。该值必须为绝对路径名, 表示为 <b>Path [, Path...]</b> 。                                            |
| <b>size</b>     | 文件的大小, 以字节表示。如果该值为空, 则除去 <b>size</b> 属性。如果未指定值, 则该命令会计算值。该值为一个十进制值。                                                                             |
| <b>source</b>   | 文件源。如果该值为空, 则除去 <b>source</b> 属性。如果未指定值, 则创建一个适当类型的空文件。该值必须为绝对路径名。                                                                              |
| <b>type</b>     | 文件类型。该值不可为空。如果未指定值, 则该命令会计算值, 它可以是 <b>FILE</b> 、 <b>DIRECTORY</b> 、 <b>FIFO</b> 、 <b>BLK_DEV</b> 、 <b>CHAR_DEV</b> 或 <b>MPX_DEV</b> 关键字。        |

可通过创建或修改 **/etc/security/sysck.cfg** 文件中的 **sysck** 节来添加、删除或修改 **tcback** 命令的属性。可使用下列属性:

|                 |                                                                                                                                       |
|-----------------|---------------------------------------------------------------------------------------------------------------------------------------|
| <b>checksum</b> | 计算文件校验和值的备用 <b>checksum</b> 命令。系统将每个文件名附加到该命令。如果该值为空, 则去除该备用 <b>checksum</b> 属性。该值为对每一文件运行的命令字符串。缺省字符串为 <b>/usr/bin/sum -r &lt;</b> 。 |
| <b>setgids</b>  | 要用 <b>setgid</b> 程序检测是否为无效的管理组的附加列表 (ID 号大于 200 的组)。如果该值为空, 则除去 <b>setgids</b> 属性。该值为一个由逗号分隔的组名列表。                                    |
| <b>setuids</b>  | 要用 <b>setuid</b> 程序检测是否为无效的管理用户的附加列表 (ID 号大于 200 的用户)。如果该值为空, 则除去 <b>setuids</b> 属性。该值为一个由逗号分隔的用户名列表。                                 |

|                     |                                                                                                                   |
|---------------------|-------------------------------------------------------------------------------------------------------------------|
| <b>treeck_nodir</b> | <b>tcbck</b> 命令不验证的目录列表。如果该值为空，则除去 <b>treeck_nodir</b> 属性。该值为由逗号分隔的目录列表。不排除该属性中的目录下的文件系统。用 <b>-i</b> 标记来排除这些文件系统。 |
| <b>treeck_novfs</b> | 为在检测已安装的文件系统树时 <b>tcbck</b> 命令不验证的文件系统列表。如果该值为空，则除去 <b>treeck_novfs</b> 属性。该值为由逗号分隔的文件系统列表。                       |

有关这些属性的更多信息，请参阅 `/etc/security/sysck.cfg` 文件，有关典型节的信息，请参阅第 317 页的『示例』。

如果包含了无值的属性，则该命令试图从要改变的文件中计算该值。**type** 属性是强制性的，但其他不必指定。

## 修订错误

要修订错误，**tcbck** 命令通常将属性复位为已定义值。对于下列属性，命令按如下所述修改其操作：

|                 |                                                                                          |
|-----------------|------------------------------------------------------------------------------------------|
| <b>checksum</b> | 通过清除文件的访问控制表来禁用文件，但不停止任何进一步的检测。                                                          |
| <b>links</b>    | 创建任何缺失的硬链接。如果存在到另一文件的链接，则删除该链接。                                                          |
| <b>program</b>  | 调用该程序，该程序必须存在并有绝对路径名。如果发生错误，会显示出一条消息，但不采取其他的操作。                                          |
| <b>size</b>     | 通过清除文件的访问控制表来禁用文件，但不停止任何进一步的检测。                                                          |
| <b>source</b>   | 将源文件复制到由 <i>File</i> 参数标识的文件。如果源为空（ <code>null</code> ），则删除任何现有文件，并创建正确 <b>type</b> 的文件。 |
| <b>symlinks</b> | 创建缺失的符号链接。如果存在到另一文件的链接，则删除该链接。                                                           |
| <b>type</b>     | 通过清除文件的访问控制表来禁用文件，并停止任何进一步的检测。                                                           |

如果使用 **-t** 标志和 **tcbck** 命令，会提示请您决定修订错误。如果回答是，则修订错误。如果给出其他任何答复，则不修订错误。

## 安全

访问控制：该命令仅授予 `root` 用户和安全组的成员执行（`x`）访问权。该命令应对 `root` 用户设置用户标识并具有可信计算库属性。

存取的文件：

| 方式        | 文件                                   |
|-----------|--------------------------------------|
| <b>r</b>  | <code>/etc/passwd</code>             |
| <b>r</b>  | <code>/etc/group</code>              |
| <b>r</b>  | <code>/etc/security/user</code>      |
| <b>rw</b> | <code>/etc/security/sysck.cfg</code> |
| <b>x</b>  | <code>/usr/bin/aclget</code>         |
| <b>x</b>  | <code>/usr/bin/aclput</code>         |
| <b>x</b>  | <code>/usr/bin/sum</code>            |

审计事件：

| 事件                  | 信息       |
|---------------------|----------|
| <b>TCBCK_Check</b>  | 文件、错误、状态 |
| <b>TCBCK_Update</b> | 文件、功能    |

## 示例

1. 将带有 **acl**、**checksum**、**class**、**group**、**owner** 和 **program** 属性的 **/bin/boo** 文件添加到 **tcback** 数据库，输入：

```
tcback -a /bin/boo acl checksum class=audit group owner\  
program=/bin/boock
```

产生的节将包含之前所给出的属性，未定义的属性都会插入计算值。数据库将包含如下的节：

```
/bin/boo:  
acl =  
checksum = 48235  
class = audit  
group = system  
owner = root  
program = /bin/boock  
type = FILE
```

属性值添加到安装定义，但不检测正确性。**program** 属性值来自命令行，**checksum** 属性值用 **checksum** 程序计算，所有其他除 **acl** 之外的值，从文件 **i-node** 计算而来。

2. 要指示应检测的文件大小，但不添加到数据库中（因为会在安装时扩展），请按 **/etc/passwd** 文件的下列示例使用 **VOLATILE** 关键字：

```
/etc/passwd:  
type = FILE  
owner = root  
group = system  
size = 1234,VOLATILE
```

3. 要从 **tcback** 数据库删除 **/bin/boo** 文件定义，输入：

```
tcback -d /bin/boo
```

4. 要从 **tcback** 数据库删除所有带有 **audit** 类的定义，输入：

```
tcback -d audit
```

5. 要检测 **tcback** 数据库中的所有文件，修订并报告所有错误，输入：

```
tcback -y ALL
```

6. 要在已安装文件系统树的安全审计时不验证 **/calvin** 和 **/hobbes** 文件系统，输入：

```
tcback -a sysck treeck_novfs=/calvin,/hobbes
```

7. 要在安全审计时不验证目录，输入：

```
tcback -a sysck treeck_nodir=/home/john
```

8. 要将 **jfh** 和 **jsl** 作为管理用户、将 **developers** 作为管理组添加到已安装文件的安全审计时的验证，输入：

```
tcback -a sysck setuids=jfh,jsl setgids=developers
```

9. 为新创建的 **/dev** 条目 **foo** 和 **bar** 创建/修改 **sysck.cfg** 节条目，输入：

```
tcback -l /dev/foo /dev/bar
```

注：添加这些条目，即将它们注册为可信计算库的一部分。

警告： 尽管特殊字符 “\$” 和 “?” 在该例程中是允许的，在文件名中使用它们可能会引起潜在的问题，诸如引起歧义的文件。

## 文件

**/usr/bin/tcback**  
**/etc/security/sysck.cfg**

指定到 **tcback** 命令的路径。  
指定到系统配置数据库的路径。

## 相关信息

**aclget** 命令、**grpck** 命令、**installp** 命令、**pwdck** 命令、**sum** 命令、**usrck** 命令。

《AIX 5L V5.3 通用编程概念：编写并调试程序》中的『软件重要产品数据（SWVPD）』。

《操作系统与设备管理》中的『访问控制表』讨论访问控制表的格式并提供一个示例。

关于用户的标识和认证、自主访问控制、可信计算库和审计的更多信息，请参阅《安全性》中的 Securing the network。

---

## tcopy 命令

### 用途

复制磁带。

### 语法

**tcopy** *Source* [ *Destination* ]

### 描述

**tcopy** 命令复制磁带。源文件和目标文件名由 *Source* 和 *Destination* 参数指定。**tcopy** 命令假定在磁带的末尾有两处磁带标记，当命令找到这两处文件标记时就结束。如果仅指定了源磁带，**tcopy** 命令显示记录和磁带文件大小的信息。

### 示例

要从流磁带复制到 9 磁道的磁带上，输入：

```
tcopy /dev/rmt0 /dev/rmt8
```

### 文件

**/usr/bin/tcopy** 包含 **tcopy** 命令。

## 相关信息

《操作系统与设备管理》中的『备份文件和存储介质』。

**rmt** 特殊文件。

---

## tcpdump 命令

### 用途

在网络上转储流量

### 语法

```
tcpdump [ -a ] [ -A ] [ -d ] [ -D ] [ -e ] [ -f ] [ -l ] [ -L ] [ -n ] [ -N ] [ -O ] [ -p ] [ -q ] [ -R ]  
[ -S ] [ -t ] [ -u ] [ -U ] [ -v ] [ -x ] [ -X ] [ -c count ]
```

```
[ -C file_size ] [ -F file ]  
[ -i interface ] [ -m module ] [ -r file ]  
[ -s snaplen ] [ -w file ]  
[ -E addr ] [ -y datalinktype ]  
[ expression ]
```

## 描述

**tcpdump** 命令显示与布尔表达式相匹配的网络接口上信息包的头部分。运行它时还可以带有 **-w** 标志，此标志使它将信息包的数据保存到文件中以备日后分析。运行它时还可以带有 **-r** 标志，此标志使它从保存的信息包文件读取信息包，而不是从网络接口读取信息包。在所有情况下，**tcpdump** 将只处理与表达式相匹配的信息包。

如果运行时不带有 **-c** 标志，**tcpdump** 将继续捕获信息包，直到 SIGINT 信号（通常为 control-C）或 SIGTERM 信号（通常为 **kill(1)** 命令）中断它。如果 **tcpdump** 运行时带有 **-c** 标志，它将捕获信息包，直到 SIGINT 或 SIGTERM 信号中断它，或者直到已经处理信息包的指定数字。

当 **tcpdump** 完成捕获信息包时，它将报告以下内容的计数：

### “过滤器接收”的信息包

对所有信息包进行计数，而不管是否通过过滤器表达式对其进行匹配。

### “内核删除”的信息包

由于缺乏缓冲区空间而删除的信息包的数目。

## 允许的原语

### dst host host

如果信息包的 IPv4/v6 目标字段为 host（它可以为地址或者名称），则为真。

### src host host

如果信息包的 IPv4/v6 源字段为 host，则为真。

### host host

如果信息包的 IPv4/v6 源或目标为 host，则为真。以上任何 host 表达式可前置关键字 ip、arp、rarp 或 ip6，如在 ip host host 中所示，ip host host 等同于以下内容：

```
ether proto \ip and host host
```

如果 host 是带有多个 IP 地址的名称，则检查每个地址是否匹配。

### ether dst ehost

如果以太网目标地址为 ehost，则为真。Ehost 可以为 **/etc/ethers** 的一个名称，也可以为一个数字（请参阅 ethers(3N) 以了解数字格式）。

### ether src ehost

如果以太网源地址为 ehost，则为真。

### ether host ehost

如果以太网源地址或目标地址为 ehost，则为真。

### gateway host

如果信息包使用 host 作为网关，则为真。例如，以太网源地址或目标地址为 host，但是 IP 源和 IP 目标都不是 host。host 必须为名称，并且机器的 host-name-to-IP-address 解决方案机制（主机名文件、DNS 和 NIS 等）和机器的 host-name-to-Ethernet-address 解决方案机制（/etc/ethers 等）都必须找到它。等效的表达式为 ether host ehost，而不是可以结合 host / ehost 的名称或数字使用的 host host。此时在支持 IPv6 的配置中语法不起作用。

**dst net net**

如果信息包的 IPv4/v6 目标地址有一个网络号 net, 则为真。

**src net net**

如果信息包的 IPv4/v6 源地址有一个网络号 net, 则为真。

**net net**

如果信息包的 IPv4/v6 源地址或目标地址有一个网络号 net, 则为真。

**net net mask netmask**

如果 IP 地址与带有特定网络掩码的 net 相匹配, 则为真。可能受到 src 或 dst 限定。请注意, 此语法对 IPv6 网络无效。

**net net/len**

如果 IPv4/v6 地址与带有网络掩码长度位范围的 net 相匹配, 则为真。可能受到 src 或 dst 限定。

**dst port port**

如果信息包为 ip/tcp、ip/udp、ip6/tcp 或 ip6/udp, 并具有目标端口值 port, 则为真。port 可以为 **/etc/services** 中使用的数字或名称 (请参阅 tcp(4P) 和 udp(4P))。如果使用了名称, 则检查端口号和协议。如果使用了数字或有歧义的名称, 则只检查端口号 (例如, dst port 513 将显示 tcp/login 流量和 udp/who 流量, 并且端口域将显示 tcp/domain 和 udp/domain 流量)。

**src port port**

如果信息包具有源端口值 port, 则为真。

**port port**

如果信息包的源或目标端口为 port, 则为真。以上任何 port 表达式都可以前置关键字 tcp 或 udp, 如 tcp src port port 中所示, tcp src port port 仅与源端口为 port 的 tcp 信息包相匹配。

**less length**

如果信息包的长度小于或等于 length, 则为真。它等同于: len <= length。

**greater length**

如果信息包的长度大于或等于 length, 则为真。它等同于: len >= length。

**ip proto protocol**

如果信息包为 protocol 类型协议的 IP 信息包, 则为真。protocol 可以为一个数字或以下名称之一: icmp、icmp6、igmp、igrp、pim、ah、esp、vrrp、udp 或 tcp。请注意, 标识 tcp、udp 和 icmp 还是关键字, 必须通过反斜杠 (\) (在 C-shell 中为 \) 进行转义。请注意, 此原语不追踪协议标题链。

**ip6 proto protocol**

如果信息包为 protocol 类型协议的 IPv6 信息包, 则为真。请注意, 此原语不追踪协议标题链。

**ip6 protochain protocol**

如果信息包为 IPv6 信息包, 并包含协议标题 (在其协议标题链中带有类型协议), 则为真。例如, ip6 protochain 6 与任何 IPv6 信息包 (在协议标题链中带有 TCP 协议标题) 都相匹配。在 IPv6 标题和 TCP 标题之间, 信息包可能包含这几个标题, 例如, 认证标题、路由标题或逐跳点选项标题。此原语发射的 BPF 代码很复杂, tcpdump 中的 BPF 优化器代码无法对其进行优化, 所以它速度有些慢。

**ip protochain protocol**

等同于 ip6 protochain protocol, 但是它针对于 IPv4。

**ether broadcast**

如果信息包是以太网广播信息包, 则为真。ether 关键字为可选。

**ip broadcast**

如果信息包是 IPv4 广播信息包, 则为真。它检查全 0 和全 1 的广播约定, 并查找已在其上完成捕获的接口上的子网掩码。

如果已在其上完成捕获的接口的子网掩码不可用，例如，因为已在其上完成捕获的接口没有网络掩码，则此检查将无法正确执行。

#### **ether multicast**

如果信息包是以太网多点广播信息包，则为真。**ether** 关键字为可选。它是 `ether[0] & 1 != 0` 的缩略语。

#### **ip multicast**

如果信息包是 IP 多点广播信息包，则为真。

#### **ip6 multicast**

如果信息包是 IPv6 多点广播信息包，则为真。

#### **ether proto protocol**

如果信息包为 **ether** 类型协议，则为真。**protocol** 可以为数字或以下名称之一：`ip`、`ip6`、`arp`、`rarp`、`atalk`、`aarp`、`decnet`、`sca`、`lat`、`mopdl`、`moprc`、`iso`、`stp`、`ipx` 或 `netbeui`。请注意，这些标识也是关键字，并且必须通过反斜杠 (\) 进行转义。

[ 在 FDDI (例如，“`fddi protocol arp`”)、令牌环 (例如，“`tr protocol arp`”) 和 IEEE 802.11 无线 LAN (例如，“`wlan protocol arp`”) 的情况下，对于大部分这些协议，协议标识来源于 802.2 逻辑链路控制 (LLC) 标题，此标题通常位于 FDDI、令牌环或 802.11 标题的顶层。为 FDDI、令牌环或 802.11 上的大多数协议标识进行过滤时，对于封装的以太网，`tcpdump` 仅检查 LLC 标题的协议标识字段是否是带有组织单元标识 (OUI) `0x000000` 的所谓 SNAP 格式；它不检查信息包是否是带有 OUI `0x000000` 的 SNAP 格式。异常如下：

**iso** **tcpdump** 检查 LLC 标题的 DSAP (目标服务访问点) 和 SSAP (源服务访问点) 字段。

#### **stp and netbeui**

**tcpdump** 检查 LLC 标题的 DSAP。

**atalk** **tcpdump** 检查带有 OUI `0x080007` 和 AppleTalk etype 的 SNAP 格式的信息包。

在以太网的情况下，**tcpdump** 检查大多数这些协议的以太网类型字段。异常如下：

#### **iso、sap 和 netbeui**

**tcpdump** 将检查 802.3 框架，然后检查 LLC 标题，这些操作与对 FDDI、令牌环和 802.11 的操作相同。

**atalk** **tcpdump** 将检查以太网框架中的 AppleTalk etype 和 SNAP 格式的信息包，这些操作与对 FDDI、令牌环和 802.11 的操作相同。

**aarp** **tcpdump** 将要么检查以太网框架中的 AppleTalk ARP etype，要么检查带有 OUI `0x000000` 的 802.2 SNAP 框架中的 AppleTalk ARP etype；

**ipx** **tcpdump** 将检查以太网框架中的 IPX etype、LLC 标题中的 IPX DSAP、IPX 的不带 LLC 标题的 802.3 封装以及 SNAP 框架中的 IPX etype。

#### **decnet src host**

如果 DECNET 源地址为 `host`，它可能是格式 `10.123` 的地址，也可能是 DECNET 主机名。[ DECNET 主机名支持只在经过配置以运行 DECNET 的 Ultrix 系统上可用。 ]

#### **decnet dst host**

如果 DECNET 目标地址为 `host`，则为真。

#### **decnet host host**

如果 DECNET 源地址或目标地址为 `host`，则为真。

#### **ifname interface**

如果信息包记录为来自指定接口，则为真。

### **on interface**

与 `iface` 修饰符同义。

### **rn timer**

如果信息包记录为与指定 PF 规则号码相匹配，则为真（仅适用于 OpenBSD 的 `pf(4)` 记录的信息包）。

### **rule number**

与 `nr` 修饰符同义。

### **reason code**

如果用指定 PF 原因码来记录信息包，则为真。已知的代码为：`match`、`bad-offset`、`fragment`、`short`、`normalize` 和 `memory`（仅适用于 OpenBSD 的 `pf(4)` 记录的信息包）。

### **action act**

如果在记录信息包时 PF 采取指定操作，则为真。已知的操作为：传送和阻塞（仅适用于 OpenBSD 的 `pf(4)` 记录的信息包）。

### **netbeui**

`ip`、`ip6`、`arp`、`rarp`、`atalk`、`aarp`、`decnet`、`iso`、`stp` 和 `ipx`。

缩略语：

`ether proto p`

其中 `p` 是上述协议之一。

`lat`、`moprc` 和 `mopdl`

缩略语：

`ether proto p`

其中 `p` 是上述协议之一。请注意，`tcpdump` 当前不知道如何对这些协议进行语法分析。

### **vlan [*vlan\_id*]**

如果信息包是 IEEE 802.1Q VLAN 信息包，则为真。如果指定了 [*vlan\_id*]，仅在信息包拥有指定的 *vlan\_id* 时才为真。请注意，如果假定信息包是 VLAN 信息包，在表达式中遇到的第一个 `vlan` 关键字更改了表达式其余部分的译码偏移量。

### **tcp、udp 和 icmp**

缩略语：

`ip proto p or ip6 proto p`

其中 `p` 是上述协议之一。

### **iso proto protocol**

如果信息包是 protocol 类型协议的 OSI 信息包，则为真。protocol 可以为数字或以下名称之一：`clnp`、`esis` 或 `isis`。

### **clnp、esis 和 isis**

缩略语：

• `iso proto p`

其中 `p` 是上述协议之一。

### **I1、I2、Iih、Isp、snp、csnp 和 psnp**

IS-IS PDU 类型的缩略语。



- vpi *n*** 如果信息包是用于 Solaris 上 SunATM 的 ATM 信息包，并带有虚路径标识 *n*，则为真。
- vci *n*** 如果信息包是用于 Solaris 上 SunATM 的 ATM 信息包，并带有虚拟信道标识 *n*，则为真。
- lane** 如果信息包是用于 Solaris 上 SunATM 的 ATM 信息包，并且是 ATM LANE 信息包，则为真。请注意，如果假定信息包是 LANE 模拟以太网信息包，或者是 LANE LE Control 信息包，则在表达式中遇到的第一个 lane 关键字更改了在表达式其余部分中完成的测试。如果不指定 lane，则在假定信息包是封装 LLC 的信息包的情况下完成测试。
- llc** 如果信息包是用于 Solaris 上 SunATM 的 ATM 信息包，并且是封装 LLC 的信息包，则为真。
- oamf4s**  
如果信息包是用于 Solaris 上 SunATM 的 ATM 信息包，并且是段 OAM F4 流单元（VPI=0 和 VCI=3），则为真。
- oamf4e**  
如果信息包是用于 Solaris 上 SunATM 的 ATM 信息包，并且是端到端 OAM F4 流单元（VPI=0 和 VCI=4），则为真。
- oamf4** 如果信息包是用于 Solaris 上 SunATM 的 ATM 信息包，并且是段或端到端 OAM F4 流单元（VPI=0 和（VCI=3 | VCI=4）），则为真。
- oam** 如果信息包是用于 Solaris 上 SunATM 的 ATM 信息包，并且是段或端到端 OAM F4 流单元（VPI=0 和（VCI=3 | VCI=4）），则为真。
- metac** 如果信息包是用于 Solaris 上 SunATM 的 ATM 信息包，并且位于元信令环路（VPI=0 和 VCI=1）上，则为真。
- bcc** 如果信息包是用于 Solaris 上 SunATM 的 ATM 信息包，并且位于广播信令环路（VPI=0 和 VCI=2）上，则为真。
- sc** 如果信息包是用于 Solaris 上 SunATM 的 ATM 信息包，并且位于信令环路（VPI=0 和 VCI=5）上，则为真。
- ilmic** 如果信息包是用于 Solaris 上 SunATM 的 ATM 信息包，并且位于 ILMI 环路（VPI=0 和 VCI=16）上，则为真。
- connectmsg**  
如果信息包是用于 Solaris 上 SunATM 的 ATM 信息包，并位于信令环路，而且是 Q.2931 设置、呼叫进程、连接、连接 ACK、发布或发布完成消息，则为真。
- metaconnect**  
如果信息包是用于 Solaris 上 SunATM 的 ATM 信息包，并位于元信令环路，而且是 Q.2931 设置、呼叫进程、连接、发布或发布完成消息，则为真。
- expr relop expr**  
如果保留关系，其中 relop 是 >、<、>=、<=、= 和 != 之一，expr 是算术表达式，此算术表达式由整数常量（用标准 C 语法表达）、常规二目运算符 [+、-、\*、/ 和 ()]、长度运算符和特殊信息包数据存储器组成，则为真。要存取信息包里的数据，请使用以下语法：  
proto [ expr : size ]
- Proto 是 ether、fddi、tr、wlan、ppp、slip、link、ip、arp、rarp、tcp、udp、icmp 或 ip6 其中之一，并且表明索引操作的协议层。（ether、fddi、wlan、tr、ppp、slip 和 link 都称为链接层。）请注意，tcp、udp 和其他上层协议类型仅适用于 IPv4，而不是 IPv6（以后将对其进行修订）。与表明协议层相关的字节偏移量由 expr 给定。size 为可选，它表明兴趣字段里的字节数；它可以为 1、2 或 4，缺省值为 1。由关键字 len 表示的长度运算符给定信息包的长度。

例如, `ether[0] & 1 != 0` 捕获所有多点广播流量。表达式 `ip[0] & 0xf != 5` 捕获带有选项的所有 IP 信息包。表达式 `ip[6:2] & 0x1fff = 0` 仅捕获未分段的信息包和分段信息包的片段零。此检查隐式适用于 `tcp` 和 `udp` 索引操作。例如, `tcp[0]` 始终表示 TCP 标题的第一个字节, 而从不表示插入片段的第一个字节。

某些偏移量和字段值可以表达为名称而不是数字值。以下协议标题字段偏移量可用: `icmp`type (ICMP 类型字段)、`icmp`code (ICMP 代码字段) 和 `tcp`flags (TCP 标志字段)。

以下 ICMP 类型字段值可用:

`icmp-echoreply`、`icmp-unreach`、`icmp-sourcequench`、`icmp-redirect`、`icmp-echo`、`icmp-routeradvert`、`icmp-routersolicit`、`icmp-timxceed`、`icmp-paramprob`、`icmp-tstamp`、`icmp-tstampreply`、`icmp-ireq`、`icmp-ireqreply`、`icmp-maskreq` 和 `icmp-maskreply`。

以下 TCP 标志字段值可用: `tcp-fin`、`tcp-syn`、`tcp-rst`、`tcp-push`、`tcp-ack` 和 `tcp-urg`。

## 组合原语

用括号括起的原语组和运算符组 (括号特定于 Shell 并且必须转义)。

非 (“!” 或 “not” )。

并置 (“&&” 或 “and” )。

或 (“||” 或 “or” )。

“非” 具有最高优先顺序。“或” 和 “并置” 具有相同的优先顺序, 并按由左向右的顺序相连。请注意, 现在需要 “并置” 的 “显式” 和 “标记”, 而不需要 “并列”。

如果给出了不带关键字的标识, 则假定使用最新的关键字。例如, `not host vs and ace` 是 `not host vs and host ace` 的缩略语, 不应与 `not ( host vs or ace )` 混淆。

表达式自变量可作为单自变量或多自变量 (选择两者中较方便的一种) 传递给 `tcpdump`。通常, 如果表达式包含 Shell 元字符, 将其作为单个的、加引号的自变量来传递会更容易。多自变量在语法分析之前以空格并置。

## 标志

- a** 尝试将网络和广播地址转换为名称。
- A** 以 ASCII 显示每个信息包 (减去其链接级别标题)。方便捕获 Web 页面。
- c** 在接收 *Count* 信息包后退出。
- C *file\_size*** 在将原始信息包写入 *savefile* 之前, 检查文件当前是否大于 *file\_size*, 如果确实如此, 则关闭当前的 *savefile* 并打开新的 *savefile*。第一个 *savefile* 之后的 *savefile* 将拥有用 **-w** 标志指定的名称, 名称后带有以 2 开头并往上继续的数字。*file\_size* 的单元以百万字节计 (1,000,000 字节, 而不是 1,048,576 字节)。
- d** 将编译过的信息包匹配代码转储至标准输出, 然后停止。
- D** 显示网络接口的列表, 这些网络接口在系统上和在网络接口上可捕获信息包的 `tcpdump` 上可用。显示每个网络接口的数字和接口名称 (可能带有此接口的文本描述)。接口名称或数字可以提供给 **-i** 标志以指定在其上进行捕获的接口。
- dd** 将信息包匹配代码作为 C 程序片段进行转储。
- ddd** 将信息包匹配代码作为十进制数字 (以某个计数开始) 进行转储。
- e** 在每一转储行上显示链接级别标题。

- E *addr*** 使用 `spi@ipaddr algo:secret` 来解密 IPsec ESP 信息包，该信息包的地址将写到 *addr*，且该信息包将包含安全性参数索引值 `spi`。可以用逗号或换行分隔来重复此组合。  
注：现在支持设置 IPv4 ESP 信息包的 `secret`。  
  
算法可以为 `des-cbc`、`3des-cbc`、`blowfish-cbc`、`rc3-cbc`、`cast128-cbc` 或没有算法。缺省值为 `des-cbc`。如果安装了 **libcrypto** 并且它在 `LIBPATH` 中，将只显示解密信息包的功能。  
  
`secret` 为 ESP 密钥的 ASCII 文本。如果前置 `0x`，将读取十六进制值。  
  
此选项采用 RFC2406 ESP，而不是 RFC1827 ESP。此选项仅用于调试用途，使用此选项时不鼓励带有真密钥。通过 **ps(1)** 和其他手段在命令行上显示 IPsec 密钥，您可以使他人看见密钥。  
  
除以上语法之外，语法文件名称可以用来使 **tcpdump** 读取提供的文件。因为在接收到第一个 ESP 信息包时打开文件，因此还应该放弃可能已提供给 **tcpdump** 的任何特殊权限。
- f** 以数字而非符号方式显示外来 IPv4 地址。  
  
通过使用在其上执行捕获的接口的 IPv4 地址和网络掩码来完成外来 IPv4 地址的测试。如果此地址或网络掩码不可用，则此选项将不会正确地起作用。
- F *file*** 使用 *file* 作为过滤器表达式的输入。忽略了命令行上给出的其他表达式。
- i *interface*** 在 *interface* 上侦听。如果未指定，**tcpdump** 将搜索系统 *interface* 列表以查找最低编号和配置的 *interface*（回送除外）。通过选择最早的匹配来中断间距。  
  
通过 **-D** 标志显示的 *interface* 数字可以用作 *interface* 自变量。
- l** 缓存标准输出行。如果您要在捕获数据时查看数据，它很有用。例如：  

```
tcpdump -l | tee dat
或 tcpdump -l > dat & tail -f dat
```
- L** 列出接口和出口的已知的数据链路类型。
- m *module*** 从 *module* 文件装入 SMI MIB 模块定义。可多次使用此选项来将多个 MIB 模块装入 **tcpdump**。
- n** 块将地址（即，主机地址和端口号等）转换为名称。
- N** 省略显示主机名的域名限定。例如，**tcpdump** 将显示 `nic`，而不显示 `nic.ddn.mil`。
- O** 使 **tcpdump** 不运行信息包匹配代码优化器。这仅在怀疑优化器中存在错误时有用。
- p** 不要将接口处于混合方式。请注意，由于其他某种原因，接口可能处于混合方式；因此，**-p** 无法用作 `ether host {local-hw-addr}` 或 `ether broadcast` 的缩略语。
- q** 快速输出。因为显示更少协议信息，所以输出行更短。
- r *file*** 从 *file*（用 **-w** 选项创建）中读取信息包。如果 *file* 是 “-”，则使用标准输入。
- R** 假定 ESP/AH 信息包基于原有的规范。（RFC1825 到 RFC1829）。如果指定，**tcpdump** 将不显示重放禁止字段。因为在 ESP/AH 规范中没有协议版本字段，所以 **tcpdump** 无法演绎 ESP/AH 协议版本。
- S** 显示绝对而非相对的 TCP 序号。
- s *snaplen*** 从每一信息包捕获 *snaplen* 数据字节，而不是缺省值 68。68 字节对 IP、ICMP、TCP 和 UDP 而言已足够，但有可能截断名称服务器和 NFS 信息包的协议信息（请参阅以下内容）。因为有限快照而被截断的信息包在输出中显示时带有 `[proto]`，其中 *proto* 是发生截断的协议级别的名称。请注意，采用更大的快照将增加它处理信息包的时间，并有效减少信息包缓冲的数量。这可能会导致丢失信息包。应该将 *snaplen* 限制为将捕获您感兴趣的协议信息的最小数目。将 *snaplen* 设置为 0 则意味着使用必需长度来捕获所有信息包。
- T** 强制使用指定类型解释 *expression* 选择的信息包。当前已知的类型有 `cnfp`（Cisco 网络流量协议）、`rpc`（远程过程调用）、`rtp`（实时应用协议）、`rtcp`（实时应用控制协议）、`snmp`（简单网络管理协议）、`tftp`（次要文件传输协议）、`vat`（可视音频工具）、和 `wb`（分布式白板）。
- t** 在每一转储行上省略时间戳显示。
- tt** 在每一转储行上显示未格式化的时间戳。
- ttt** 在每一转储行的当前行和前行之间显示变化量（以微秒计数）。
- tttt** 在每一转储行上显示前面带有日期的缺省格式的时间戳记。
- u** 显示未译码的 NFS 句柄。
- U** 通过 **-w** 选项（例如，“`packet-buffered`”）保存输出。因为保存了每一个信息包，所以它将写入输出文件，而不是仅在输出缓冲区填满时撰写它。

- v** 指定稍微详细些的输出。例如，显示 IP 信息包中的生存时间、标识、总长度和选项。还启用其他信息包完整性检查，例如，验证 IP 和 ICMP 标题校验和。
  - vv** 甚至比 **-v** 更详细的输出。例如，从 NFS 显示其他字段，并对应答包进行完全译码。
  - vvv** 甚至比 **-vv** 更详细的输出。例如，完整显示 Telnet SB 到 SE 选项。带有 **-X** 的 Telnet 选项也以十六进制显示。
  - w file** 将原始的信息包写入 *file*，而不对其进行语法分析，也不将其显示出来。以后可以用 **-r** 标志显示它们。如果 *file* 为 “-”，则使用标准输出。
  - x** 以十六进制显示每个信息包（减去其链接级别标题）。将显示整个信息包或 `snaplen` 字节中的较小者。请注意，它是整个链接层信息包，因此对于填充器的链接层（例如，以太网），当更高层信息包短于必需的填充时，还将显示填充字节。
  - xx** 以十六进制显示每个信息包，包括其链接级别标题。
  - X** 以十六进制和 ASCII 显示每个信息包（减去其链接级别标题）。用于分析新协议时它非常方便。
  - y** 设置数据链路类型以在将信息包捕获到 *datalinktype* 时使用。
- datalinktype*

## 参数

**表达式** 选择要转储的信息包。如果未给定表达式，将转储网络上的所有信息包。否则仅转储表达式为 `true` 的信息包。

表达式由一个或多个原语构成。原语通常由前面带有一个或多个限定符的标识（名称或数字）构成。存在三种不同的限定符：

**type** 限定符显示标识名称或数字所指的原语类型。可能的类型为 `host`、`net` 和 `port`。例如，“`host foo`”、“`net 128.3`”和“`port 20`”。如果没有类型限定符，则采用 `host`。

**dir** 限定符指定到标识和 / 或来自标识的特定传送方向。可能的方向为 `src`、`dst`、`src` 或 `dst` 和 `dst`。如果没有 `dir` 限定符，则采用 `src` 或 `dst`。对于某些链接层（例如，SLIP）和某些其他设备类型，入站和出站限定符可以用来指定希望的方向。

**proto** 限定符限制与特定协议相匹配。可能的协议为 `fddi`、`tr`、`wlan`、`ip`、`ip6`、`arp`、`rarp`、`decnet`、`tcp` 和 `udp`。如果没有 `proto` 限定符，则采用与该类型一致的所有协议。

**fddi** 是 `ether` 的别名。解析器将其理解为：“在指定网络接口上使用的数据链路级”。FDDI 头部分包含类似以太网的源地址和目标地址，并经常包含类似以太网的信息包类型，因此可以在这些 FDDI 字段上进行过滤，此操作与对类似以太网字段的操作相同。FDDI 头部分还包含其他字段，但是无法在过滤器表达式中对其进行命名。

与 **fddi** 类似，**tr** 和 **wlan** 是 `ether` 的别名。上一段有关 FDDI 头部分的论述还适用于令牌环和 802.11 无线 LAN 头部分。对于 802.11 头部分，目标地址为 `DA` 字段，源地址为 `SA` 字段；不测试 `BSSID`、`RA` 和 `TA` 字段。

除以上所述之外，还有一些特殊的“原语”关键字不遵循这种模式：网关、广播、小于、大于和算术表达式。所有这些关键字描述如下。

通过使用单词 `and`、`or` 和 `not` 组合原语来构建更复杂的过滤器表达式。

## 环境变量

必须设置 `LIBPATH` 环境变量，或者 `libcrypto` 库应该位于 **-E** 标志的 `/usr/lib` 以起作用。例如：

```
ksh$ LIBPATH=/opt/freeware/lib tcpdump -E"algo:secret"
```

## 退出状态

|    |     |
|----|-----|
| 0  | 成功。 |
| 非零 | 错误。 |

## 安全

从网络接口读取信息包需要对 `/dev/bpf*` 的读访问权，通常仅限于 `root` 用户。从文件读取信息包除了文件读许可权之外，不需要任何特殊权限。

## 示例

1. 要显示到达或离开 `sundown` 的所有信息包，请输入以下内容：

```
tcpdump host sundown
```

2. 要显示 `helios` 和 `hot` 或 `ace` 之间的流量，请输入以下内容：

```
tcpdump host helios and \( hot or ace \)
```

3. 要显示 `ace` 和任何主机（`helios` 除外）之间的所有 IP 信息包，请输入以下内容：

```
tcpdump ip host ace and not helios
```

4. 要显示本地主机和 Berkeley 上的主机之间的所有流量，请输入：

```
tcpdump net ucb-ether
```

5. 要显示整个因特网网关 `snup` 的所有 `ftp` 流量，请输入以下内容：

```
tcpdump 'gateway snup and (port ftp or ftp-data)'
```

注：给表达式加引号以防止 shell（误）解释括号）

6. 要显示既非来源于本地主机也非发送到本地主机的流量（如果将网关通往其他一个网络，此网关不会将流量发送到本地网络），请输入以下内容。

```
tcpdump ip and not net localnet
```

7. 要显示涉及非本地主机的每个 TCP 对话的开始和结束信息包（SYN 和 FIN 信息包），请输入：

```
tcpdump 'tcp[tcpflags] & (tcp-syn|tcp-fin) != 0 and not src and d dst net localnet'
```

8. 要显示通过网关 `snup` 发送的长于 576 字节的 IP 信息包，请输入以下内容：

```
tcpdump 'gateway snup and ip[2:2] > 576'
```

9. 要显示未通过以太网广播或多点广播发送的 IP 广播或多点广播信息包，请输入：

```
tcpdump 'ether[0] & 1 = 0 and ip[16] >= 224'
```

10. 要显示不是回传请求 / 答复（例如，不是 ping 信息包）的所有 ICMP 信息包，请输入：

```
tcpdump 'icmp[icmptype] != icmp-echo and icmp[icmptype] != icmp-e choreply'
```

## 标准错误

所有的错误和警告都发送到 `stderr`。

## 限制

跨越夏令时更改的信息包跟踪将提供有偏差的时间戳记（忽略时间更改）。

令牌环头部分中那些字段以外的字段上的过滤器表达式将不会正确处理源路由令牌环信息包。

802.11 头部分中那些字段以外的字段上的过滤器表达式将不会正确处理带有 To DS 和 From DS 集合的 802.11 数据包。

ip6 proto 应该追踪标题链，但在此时它未进行此操作。为此工作情况提供了 ip6 protochain。

传输层头部分的算术表达式类似于 tcp[0]，对 IPv6 信息包不起作用。它仅查看 IPv4 信息包。

## 文件

|                                                          |                             |
|----------------------------------------------------------|-----------------------------|
| <code>/usr/sbin/tcpdump</code>                           | <code>tcpdump</code> 命令的位置。 |
| <code>/usr/lib/libpcap.a</code>                          |                             |
| <code>/dev/bpf*</code>                                   |                             |
| <code>/opt/freeware/lib/libcrypto.a(libcrypto.so)</code> | 可选                          |

## 相关信息

pcap 库、iptrace 和 Berkeley 信息包过滤器。

---

## tctl 命令

### 用途

将子命令提供给一个流磁带设备。

### 语法

```
tctl [ -f Device ] [ eof | weof | fsf | bsf | fsr | bsr | rewind | offline | rewoffl | erase | retention  
| reset | status ] [ Count ]
```

```
tctl [ -b BlockSize ] [ -f Device ] [ -p BufferSize ] [ -v ] [ -n ] [ -B ] { read | write }
```

### 描述

`tctl` 命令将子命令提供给一个流磁带设备。如果您没有用 `-f` 标志指定 `Device` 变量，则使用环境变量 `TAPE`。如果环境变量不存在，则 `tctl` 命令使用 `/dev/rmt0.1` 设备。（当 `tctl` 命令提供 `status` 子命令时，缺省设备是 `/dev/rmt0`。）该 `Device` 变量必须指定一个原始（非块）磁带设备。`Count` 参数指定文件结束标记数、文件标记数、或记录数。如果 `Count` 参数没指定，则默认值为 1。

### 子命令

#### `eof` 或 `weof`

将由 `Count` 参数指定的文件结束符数目写在磁带当前位置上。在 8 毫米磁带设备上，文件结束符可写在三个地方：

- 空带前
- 扩展文件标记前
- 在磁带开始标记处

在 9 磁道磁带上，磁带结束符可写在磁带任何位置上。但是，该子命令不支持重写单个块数据。

#### `fsf`

将磁带向前移动由 `Count` 参数指定的文件标记数目，位置标记放在文件标记的磁带结束符（EOT）侧。

|                                 |                                                                                                                                           |
|---------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------|
| <b>bsf</b>                      | 将磁带向后移动由 <i>Count</i> 参数指定的文件标记数目，位置标记放在文件标记的磁带开始符（BOT）侧。<br>如果 <b>bsf</b> 子命令移动磁带到开始处，则磁带倒带，且 <b>tctl</b> 命令返回 <b>EIO</b> 。              |
| <b>fsr</b>                      | 将磁带向前移动由 <i>Count</i> 参数指定的记录数。                                                                                                           |
| <b>bsr</b>                      | 将磁带向后移动由 <i>Count</i> 参数指定的记录数目。                                                                                                          |
| <b>rewind</b>                   | 将磁带倒带。忽略 <i>Count</i> 参数。                                                                                                                 |
| <b>offline</b> 或 <b>rewoffl</b> | 将磁带倒带，并使磁带机脱机。这将在适当的时候卸装磁带。在设备再次使用前，磁带必须重新插入。                                                                                             |
| <b>erase</b>                    | 删除磁带上的所有内容，并将它倒带。                                                                                                                         |
| <b>read</b>                     | 从指定磁带设备读数据（使用指定的块大小）直到内部缓冲器满为止，然后向标准输出写数据，继续用这种方式读写，直到遇到文件结束符为止。                                                                          |
| <b>reset</b>                    | 发一个总线设备复位信号（BDR）给磁带设备。只有设备不能打开并且不忙时才发送 BDR。                                                                                               |
| <b>retension</b>                | 移动磁带到开始处，然后到结束处，再回到磁带开始处。如果在恢复期间，有额外的读错误时，您应该运行 <b>retension</b> 子命令。如果您将磁带暴露在恶劣环境中，则在向磁带写数据时，您应该运行 <b>retension</b> 子命令。8 毫米磁带机不会响应这个命令。 |
| <b>status</b>                   | 显示关于指定的磁带设备的状态信息。                                                                                                                         |
| <b>write</b>                    | 打开磁带机，从标准输入中读数据，或向磁带机写数据。                                                                                                                 |

#### 注:

1. 当指定 **read** 或 **write** 子命令时，**tctl** 命令打开磁带设备并设置由 **-b** 或 **-n** 标志指定的磁带块大小。如果两个都未指定，则 **tctl** 命令使用缺省块大小，其大小为 512 个字节。
2. 仅当使用 **read** 和 **write** 子命令时才适用 **-b**、**-n**、**-p** 和 **-v** 标志。
3. 仅当使用 **read** 子命令时才适用 **-B** 标志。

## 标志

|                             |                                                                                                                                          |
|-----------------------------|------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-b</b> <i>BlockSize</i>  | 指定读写磁带设备缓冲区的大小（字节），在没有指定 <b>-n</b> 标志时，也要指定磁带块的大小。如果块大小为 0 字节，则使用可变长度块，且磁带缓冲区的大小为 32,768 字节。如果 <b>-b</b> 标志没指定，则缺省块大小和磁带缓冲区的大小均为 512 字节。 |
| <b>-B</b>                   | 每次读磁带时，向缓冲器写数据。当读可变长度记录(该记录大小不定，也没有规律)时，设定这个标志。                                                                                          |
| <b>-f</b> <i>Device</i>     | 指定磁带设备。                                                                                                                                  |
| <b>-p</b> <i>BufferSize</i> | 指定在标准输入输出上所使用的缓冲区大小。缺省缓冲区大小是 32,768 字节。 <i>BufferSize</i> 值必须是磁带块大小的乘积。                                                                  |
| <b>-v</b>                   | 详细。向标准错误显示每个读和写的大小。                                                                                                                      |
| <b>-n</b>                   | 当用 <b>read</b> 或 <b>write</b> 子命令来读写磁带时，指定可变长度记录。                                                                                        |

## 退出状态

该命令返回以下退出值:

|              |       |
|--------------|-------|
| <b>0</b>     | 成功完成。 |
| <b>&gt;0</b> | 发生错误。 |

## 示例

1. 要将 **rmt1** 磁带设备倒带，请输入:

```
tctl -f /dev/rmt1 rewind
```

2. 要在缺省磁带设备上向前移动两个文件标记，请输入：

```
tctl fsf 2
```

3. 要在 `/dev/rmt0.6` 中的磁带上写两个文件结束符，键入：

```
tctl -f /dev/rmt0.6 weof 2
```

4. 要读取用 80 字节的块格式化了的磁带设备并将结果放入一个文件中，请输入：

```
tctl -b 80 read > file
```

5. 要从用 80 字节的块格式化了的磁带设备上读取变长记录将结果放入一个文件中，请输入：

```
tctl -b 80 -n read > file
```

6. 要用 1024 字节的缓冲区向磁带设备写变长记录，请输入：

```
cat file | tctl -b 1024 -n -f/dev/rmt1 write
```

7. 要用 512 字节的块向磁带设备写数据，并为标准输出使用 5120 字节缓冲区，请输入：

```
cat file | tctl -v -f /dev/rmt1 -p 5120 -b 512 write
```

**注：**1/4 英寸（QIC）磁带设备的有效块大小只有 0 字节和 512 字节。

8. 要在 8 毫米磁带上覆盖几个备份之一，将磁带定位到备份文件的开始处，并发出这些命令：

```
tctl bsf 1
```

```
tctl eof 1
```

第一个命令移动磁带到文件标记的磁带开始端。第二个命令重写文件标记，因为在扩展文件标记前允许写。在写磁头到达前，驱动器的删除磁头删除数据，于是 **write** 子例程可以在磁带上覆盖已有数据。但是，所有旧的数据丢失，因为文件标记无意义。

**注：****write** 子例程不能覆盖短文件标记，除非短文件标记后面是空带。如在该例中，覆盖已存在的数据时，磁带必须用扩展文件标记写数据（这一点通过 **SMIT** 接口指定）。

## 文件

**/dev/rmt $n$**  指定原始流磁带接口。  
**/usr/bin/tctl** 包含 **tctl** 命令。

## 相关信息

**dd** 命令，**mt** 命令。

**environment** 文件，**rmt** 特殊文件。

**ioctl** 子程序。

《操作系统与设备管理》中的『备份文件和存储介质』。



---

## tee 命令

### 用途

显示程序的输出并将其复制到一个文件中。

### 语法

```
tee [ -a ] [ -i ] [ File ... ]
```

### 描述

**tee** 命令读取标准输入，然后将程序的输出写到标准输出，并同时将其复制到指定的一个或多个文件。

### 标志

- a** 将输出添加到 *File* 的末尾而不是覆盖写入。
- i** 忽略中断。

### 退出状态

该命令返回以下退出值：

- 0** 标准输入被成功地复制到所有输出文件中。
- >0** 发生错误。

**注：**如果向任意成功打开的 *File* 操作数的写入不成功，写入其他成功打开的 *File* 操作数，并且标准输出会继续，但是退出值将会是 **>0**。

### 示例

- 要同时查看和保存一个命令的输出：

```
lint program.c | tee program.lint
```

它在工作站上显示命令 **lint program.c** 的标准输出，同时在文件 `program.lint` 中保存输出的一个副本。如果 `program.lint` 文件早已存在，它将被删除并替换。

- 要同时查看一个命令的输出并保存到一个现有文件：

```
lint program.c | tee -a program.lint
```

它将在工作站上显示 **lint program.c** 命令的标准输出，同时在 `program.lint` 文件尾部添加输出的一个副本。如果 `program.lint` 文件不存在，它将被创建。

### 文件

`/usr/bin/tee` 包含 **tee** 命令。

### 相关信息

**script** 命令。

《操作系统与设备管理》中的『输入和输出重定向』描述了操作系统如何处理输入和输出，以及如何使用重定向和管道符号。

---

## telinit 或 init 命令

### 用途

初始化并控制进程

### 语法

```
{ telinit | init } { 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | a | b | c | h | Q | q | S | s | M | m | N }
```

### 描述

**init** 命令初始化并控制进程。其首要任务是基于从文件 **/etc/inittab** 读入的记录启动进程。**/etc/inittab** 文件通常请求 **init** 命令在用户可登录的每一行上运行 **getty** 命令。**init** 命令控制系统所需的独立进程。

组成 **init** 命令的进程分派操作的主要进程是 **/usr/sbin/getty**。**/usr/sbin/getty** 进程启动单独终端行。其他典型地由 **init** 命令分派的进程为守护程序和 shell。

链接到 **init** 命令的 **telinit** 命令，指引 **init** 命令的操作。**telinit** 命令引用单字符参数并通过 **kill** 子例程发出 **init** 命令从而执行适当的操作。

**telinit** 命令将系统设置在一特定的运行级别上。运行级别是允许仅选中的进程组存在的软件配置。系统可以在下列的运行级别之一上：

- 0-9** 指示 **init** 命令将系统放置在 **0-9** 的其中一个运行级别上。当 **init** 命令请求运行级别在 **0-9** 中的更改时，它中止所有当前运行级别上的进程然后重新启动与新运行级别关联的任何进程。
- 0-1** 预留给操作系统供将来使用。
- 2** 包含运行在多用户环境之中的所有终端进程和守护程序。在多用户环境中，**/etc/inittab** 文件被设置，所以 **init** 命令为系统的每一个终端创建一个进程。控制台设备驱动程序也被设置在所有运行级别上运行，因此系统仅能在活动的控制台上运行。
- 3-9** 可根据用户的偏好定义。
- S、s、M、m** 告诉 **init** 命令进入维护方式。当系统从另一运行级别进入维护方式，只有系统控制台被用作终端。

下列参数也作为 **init** 命令的伪指令：

**a、b、c、h** 告诉 **init** 命令只处理那些位于 **/etc/inittab** 文件中，在运行级别字段中为 **a、b、c** 或 **h** 的记录。这四个参数 **a、b、c** 和 **h** 不是真正的运行级别。它们与运行级别的不同之处在于 **init** 命令无法要求整个系统进入运行级别 **a、b、c** 或 **h**。

当 **init** 命令在 **/etc/inittab** 文件中找到运行级别字段的值为 **a、b、c** 或 **h** 的记录时，它会启动此进程。然而，它不会杀死处于当前运行级别上的任何进程；除了已经运行在当前的系统运行级别的进程外，还会启动在运行级别字段中值为 **a、b、c** 或 **h** 的进程。真正的运行级别与 **a、b、c** 或 **h** 的另一个区别在于以 **a、b、c** 或 **h** 开始的进程不会在 **init** 命令更改运行级别时停止。有三种方式停止 **a、b、c** 或 **h** 进程：

- 在 *Action* 字段中输入 **off**。
- 删除整个对象。
- 用 **init** 命令进入维护状态。

**Q,q** 告诉 **init** 命令重新检查 **/etc/inittab** 文件。

**N** 发送信号阻止进程被重新派生。

当系统启动时，当根文件系统已在预初始化过程之中安装，则发生下列事件序列：

1. **init** 命令作为启动过程的最后一步运行。
2. **init** 试图读 **/etc/inittab** 文件。
3. 如果 **/etc/inittab** 文件存在，**init** 命令试图定位 **/etc/inittab** 文件中的一个 **initdefault** 条目。
  - a. 如果 **initdefault** 条目存在，**init** 命令用指定的运行级别作为初始系统运行级别。
  - b. 如果 **initdefault** 条目不存在，**init** 命令请求用户从系统控制台 (**/dev/console**) 输入一个运行级别。
  - c. 如果用户输入了 **S**、**s**、**M** 或 **m** 运行级别，**init** 命令进入维护运行级别。这些是仅有的不要求正确格式的 **/etc/inittab** 文件的运行级别。
4. 如果 **/etc/inittab** 文件不存在，**init** 命令缺省将系统置于维护运行级别。
5. **init** 命令每 60 秒重读 **/etc/inittab** 文件。如果 **/etc/inittab** 自从 **init** 命令上次读取它时发生了改变，**/etc/inittab** 文件中的新命令在系统启动时执行。

如请求 **init** 命令改变运行级别，**init** 命令读 **/etc/inittab** 文件以识别应当存在于新运行级别的进程。然后，**init** 命令取消所有不应该运行在新级别上的进程，并启动所有应该运行在新级别上的进程。

在这些运行级别的每一级别上由 **init** 命令运行的进程定义在文件 **/etc/inittab** 之中。通过让 **root** 用户运行 **telinit** 命令来改变运行级别，该命令链接到 **init** 命令。该用户运行的 **init** 命令向在系统启动时初始化的原始 **init** 命令发送适当的信号。缺省运行级别可通过修改 **/etc/inittab** 文件中 **initdefault** 项的运行级别来更改。

在维护运行级别，**/dev/console** 控制台终端对读写开放。提示输入 **root** 密码。当成功输入 **root** 用户密码时，**su** 被调用。有两种方法可以退出维护运行级别：

- 如果 **shell** 终止了，**init** 命令要求一个新的运行级别。

或

- **init** (或 **telinit**) 命令可以用信号通知 **init** 命令并强制其改变系统的运行级别。

在系统试图启动时，**init** 命令提示输入一个新运行级别 (**initdefault** 为维护时) 的明显失败的原因可能是终端控制台设备 (**/dev/console**) 被转换到一个并非物理控制台的设备上。倘若如此，而您希望在物理控制台而非 **/dev/console** 上工作，可以通过按物理控制台设备上的 **DEL** (删除) 键强制 **init** 命令转换到物理控制台。

当 **init** 命令提示输入一个新的运行级别，输入 **0** 到 **9** 中的任一位或以下任一字母 **S**、**s**、**M** 或 **m**。如果输入的是 **S**、**s**、**M** 或 **m**，**init** 命令以维护方式操作，其额外结果是：如果此前控制被强制转换到物理控制台，**/dev/console** 文件也被转换到该设备。**init** 命令生成一条消息到 **/dev/console** 文件事先已连接到的设备上。

如果您输入了 **0** 到 **9** 运行级别，**init** 命令进入相应的运行级别。**init** 命令拒绝任何其他输入并重新提示您输入正确值。如果这是 **init** 命令第一次进入任意运行级别而非维护级别，它搜索 **/etc/inittab** 文件中的带 **boot** 或 **bootwait** 关键字的条目。如果 **init** 命令找到了这些关键字，它执行相应的任务，假如输入的运行级别与条目的运行级别相匹配的话。例如，如果 **init** 命令找到 **boot** 关键字，它引导机器。系统的任意特定的初始化，诸如检测和安装文件系统，都发生在系统允许任何用户操作之前。然后 **init** 命令扫描 **/etc/inittab** 文件查找为该运行级别处理的所有条目。接着它继续 **/etc/inittab** 文件的正常处理。

运行级别 **2** 缺省定义为包含运行在多用户环境的所有终端进程和守护程序。在多用户环境中，**/etc/inittab** 文件被设置以使得 **init** 命令为系统的每一终端创建进程。

对终端进程而言，**shell** 终止是因为一明确输入的文件结束符，或断开连接。当 **init** 命令收到一个进程已中止的信号，它在 **/etc/utmp** 文件和 **/var/adm/wtmp** 文件之中记录该事实及其原因。**/var/adm/wtmp** 文件记录已启动的进程的历史。

要在 **/etc/inittab** 文件之中启动每一进程，**init** 命令等待其一后继过程停止，等待一电源故障信号 **SIGPWR**，或直到 **init** 命令由 **init** 或 **telinit** 命令发出以改变系统的运行级别。当上述三个条件之一发生时，**init** 命令重新检查 **/etc/inittab** 文件。即使已经有新的条目添加到 **/etc/inittab** 文件，**init** 命令仍在等待三个条件之一的发生。为提供瞬时响应，请运行 **telinit -q** 命令来重新检查 **/etc/inittab** 文件。

如果 **init** 命令发现其连续在文件 **/etc/inittab** 中运行一个条目（225 秒内超过 5 次），它假定在条目命令字符串中存在一个错误。然后它显示一个出错信息到控制台，并在系统出错日志中记录一个错误。发送该消息后，60 秒内该条目不运行。如果该错误继续发生，该命令将每 240 秒仅重新生成该条目 5 次。**init** 命令继续假定出错，直到该命令在时间间隔内 5 次不响应，或直到其收到一个用户的信号。**init** 命令仅在错误第一次出现时记录该错误。

当 **telinit** 命令请求 **init** 命令改变运行级别，**init** 命令向当前运行级别内未定义的所有进程发送一个 **SIGTERM** 信号。在以信号 **SIGKILL** 中止这些进程前，**init** 命令等待 20 秒。

如果 **init** 命令收到一个 **SIGPWR** 信号，且不在维护方式下，它扫描 **/etc/inittab** 文件查找特定的电源故障条目。在任何其他进一步的进程运行之前，**init** 命令调用与这些条目关联的任务（如果运行级别允许的话）。按这个方法，无论何时系统遇到电源故障，**init** 命令都可以执行清除和记录功能。需要注意的是，这些电源故障条目不应该使用最先初始化的设备。

## 环境

因为 **init** 命令是系统每个进程的终极祖先，系统的每一其他进程继承该 **init** 命令的环境变量。作为其初始化序列的一部分，**init** 命令读 **/etc/environment** 文件，并将该文件中发现的任一赋值复制到传递给其所有子进程的环境中。因为 **init** 子进程在注册会话中不运行，它们不继承 **init** 命令的 **umask** 设置。这些进程可以设置 **umask** 为任意它们所需要的值。**/etc/inittab** 文件中的由 **init** 执行的命令使用 **init** 的 **ulimit** 值而非 **/etc/security/limits** 中所给出的缺省值。结果为由命令行成功执行的命令在由 **init** 调用时可能执行不正确。任何有特殊 **ulimit** 需要的命令应包含特定的操作以设置为所需要的 **ulimit** 值。

## 示例

1. 要请求 **init** 命令重新检查 **/etc/inittab** 文件，输入：

```
telinit q
```

2. 要请求 **init** 命令进入维护方式，输入：

```
telinit s
```

## 文件

|                         |                        |
|-------------------------|------------------------|
| <b>/etc/inittab</b>     | 指定 <b>init</b> 命令控制文件。 |
| <b>/etc/utmp</b>        | 指定登录用户的记录。             |
| <b>/var/adm/wtmp</b>    | 指定永久登录帐户文件。            |
| <b>/sbin/rc.boot</b>    | 指定预初始化命令文件。            |
| <b>/etc/rc</b>          | 指定初始化命令文件。             |
| <b>/etc/environment</b> | 指定系统环境变量。              |
| <b>/dev/console</b>     | 指定控制台设备驱动程序。           |

## 相关信息

**chitab** 命令、**lsitab** 命令、**mkitab** 命令、**rmitab** 命令、**getty** 命令、**rc** 命令。

**inittab** 文件，**rc.boot** 文件。

**reboot** 子例程、 **umask** 子例程、 **ulimit** 子例程、 **kill** 子例程。

---

## telnet、tn 或 tn3270 命令

### 用途

使用 Telnet 接口连接本地主机到远程主机。

### 语法

```
{ telnet | tn | tn3270 } [ -d ] [ -p ] [ -n TraceFile ] [ -e TerminalType ] [ -f | -F ] [ -k realm ] [ -l user ] [ Host [ Port ] ]
```

### 描述

**telnet** 命令，也称作 **tn** 或 **tn3270** 命令，以命令方式和输入方式两种不同的方式操作。

### 命令方式

如果不带参数发出 **telnet** 命令，它将进入命令方式，如 **telnet>**、**tn>**或 **tn3270>** 提示所示。用户也可以通过针对 **telnet** 命令按 **Ctrl-]**，针对 **tn** 命令按 **Ctrl-T** 或针对 **tn3270** 命令按 **Ctrl-C** 从输入方式进入命令方式。在命令方式下，可以输入子命令来管理远程系统。部分子命令会在完成时返回到远程会话。对于那些不能返回到远程会话的子命令，按 **Enter** 键可以返回到远程会话。

注：该命令的缺省转义序列是 **telnet** 命令的 **Ctrl-]**，**tn**命令的 **Ctrl-T** 或 **tn3270** 命令的 **Ctrl-C**。该缺省值可通过改变 **TNESC** 环境变量来覆盖。

要在连接到远程主机后进入 **telnet** 命令方式，输入 Telnet 转义键序列。当在命令方式下时，标准操作系统编辑约定，例如退格键，是可用的。

### 输入方式

当带参数发出 **telnet** 命令时，它执行包含这些参数的 **open** 子命令，然后进入输入方式。输入方式的类型为逐个字母或逐行，取决于远程系统所支持的是什么。在逐个字母方式下，大部分输入的文本会立即送到远程主机处理。在逐行方式下，所有文本在本地屏幕显示，然后将完整的行发送到远程主机。

在任一输入方式下，如果 **toggle localchars** 子命令值为 **True**，用户的 **QUIT**，**INTR** 和 **FLUSH** 字符会在本地被捕获并送到远程主机的 Telnet 协议序列。**toggle autoflush** 和 **toggle autosynch** 子命令使得该操作刷新随后的输出到终端，直到远程主机确认 Telnet 序列并刷新以前的终端输入（当为 **QUIT** 和 **INTR** 字符时）。

### 阿拉伯 / 希伯来语支持

**telnet**、**tn** 和 **tn3270** 命令支持阿拉伯和希伯来文本，允许用户在仿真会话中输入阿拉伯或希伯来字符。**Ar\_AA** 语言环境显示阿拉伯字符的正确形状。以下功能支持双向阿拉伯和希伯来文本：

**语言选择：** 该功能允许切换语言层。通过下列键组合来激活阿拉伯 / 希伯来语言选择：

|                             |              |
|-----------------------------|--------------|
| <b>Alt+N</b>                | 从 AIX 终端     |
| <b>Esc+N</b>                | 从 ASCII 终端   |
| <b>Alt+N</b> 或 <b>Esc+N</b> | 从一拉丁语 AIX 终端 |

通过下列键组合激活拉丁语言层:

|                             |                     |
|-----------------------------|---------------------|
| <b>Alt+L</b>                | 从一个阿拉伯语或希伯来语 AIX 终端 |
| <b>Esc+L</b>                | 从 ASCII 终端          |
| <b>Alt+L</b> 或 <b>Esc+L</b> | 从 AIX 终端            |

**屏幕反转:** 该功能反转显示屏幕图像, 并调用新屏幕方向的缺省语言。这样, 如果屏幕反转为从右到左, 语言会变为阿拉伯 / 希伯来语。如果屏幕反转为从左到右, 语言会变为拉丁语。

如果启用对称字符交换, 反转屏幕会将双向的字符替换为其对应字。例如, 如果启用数字字符交换, 屏幕反转会将印地数字替换为其对应的阿拉伯数字, 而将阿拉伯数字替换为其对应的印地数字。

通过下列键组合激活屏幕反转:

|                             |                     |
|-----------------------------|---------------------|
| <b>Alt+S</b>                | 从一个阿拉伯语或希伯来语 AIX 终端 |
| <b>Esc+S</b>                | 从 ASCII 终端          |
| <b>Alt+S</b> 或 <b>Esc+S</b> | 从一拉丁语 AIX 终端        |

**Push/End Push:** Push 功能允许编辑方向与屏幕方向相反的文本。当激活该功能时, 光标方向反转, 语言层也相应改变, 并生成一个 Push 段。

Push 功能有两个次级方式:

|             |                                                                     |
|-------------|---------------------------------------------------------------------|
| <b>边界方式</b> | 当进入 Push 方式时该方式被激活。在该方式下, 当您输入额外的字符时光标保持在其位置不动。文本被 push 为屏幕方向的相反方向。 |
| <b>编辑方式</b> | 当光标从其边界位置移动到 Push 段区域时该方式被激活。在该方式下, 可以在 Push 段编辑文本, 同时以字段的本来方向输入。   |

通过下列键组合激活该功能:

|                             |                     |
|-----------------------------|---------------------|
| <b>Alt+P</b>                | 从一个阿拉伯语或希伯来语 AIX 终端 |
| <b>Esc+P</b>                | 从 ASCII 终端          |
| <b>Alt+P</b> 或 <b>Esc+P</b> | 从一拉丁语 AIX 终端        |

End Push 功能终止 Push 功能。光标跳至 Push 段的末尾, 并且变回原来的方向。可能通过按任意字段退出键如光标向上、光标向下或任何注意标识符 (AID) 键, 如 Enter 键来激活 End Push。也可以通过下列键组合激活该功能:

|                             |                     |
|-----------------------------|---------------------|
| <b>Alt+E</b>                | 从一个阿拉伯语或希伯来语 AIX 终端 |
| <b>Esc+E</b>                | 从 ASCII 终端          |
| <b>Alt+E</b> 或 <b>Esc+E</b> | 从一拉丁语 AIX 终端        |

**字段逆向:** 该功能将字段方向切换为屏幕方向的相反方向或相同方向。该功能不反转字段中的文本。光标方向设为新字段方向, 并选择了相应语言层。

例如, 当激活字段逆向功能时, 如果光标在字段或行的第一逻辑位置, 光标跳至该字段或行的相反一侧。这时该位置为第一逻辑位置。如果当激活字段逆向功能时光标不在字段或行的第一位置, 光标保持在原位置, 并允许自然而正确的编辑现有文本。通过下列键组合激活该功能:

|              |                     |
|--------------|---------------------|
| <b>Alt+R</b> | 从一个阿拉伯语或希伯来语 AIX 终端 |
| <b>Esc+R</b> | 从 ASCII 终端          |

**Alt+R** 或 **Esc+R**                    从一拉丁语 AIX 终端

**Autopush:**    该功能允许输入混合的从左到右和从右到左的文本。在启用后，逆向段根据输入的字符或选择的语言层自动初始化和终止。因而，该方式自动调用 Push 方式并帮您调用 Push 功能。

当向一从右到左的字段中输入一个数字字符或一个拉丁字符时，Autopush 功能自动启动 Push 功能而不改变语言。当输入额外的数字或拉丁字符时，Push 功能继续执行；否则，Push 功能自动中止。这样就可以输入嵌有数字或拉丁字符的阿拉伯语 / 希伯来语文本而不必调用 Push/End Push 功能。

当向一从左到右的字段中输入一阿拉伯 / 希伯来字符时，Autopush 功能自动启动 Push 功能而不改变语言。如果输入一个数字或拉丁字符，Autopush 功能自动终止。这样，就可以通过语言选择功能输入嵌有阿拉伯语 / 希伯来语文本的拉丁文本，而不是 Push/End 功能。

通过下列键组合激活该功能:

**Alt+A**                                    从一个阿拉伯语或希伯来语 AIX 终端  
**Esc+A**                                    从 ASCII 终端  
**Alt+A** 或 **Esc+A**                    从一拉丁语 AIX 终端

**字段形状:**    该功能给当前字段或行的阿拉伯字符定形。通过下列键组合激活该功能:

**Alt+H**                                    从一阿拉伯语 AIX 终端  
**Esc+H**                                    从 ASCII 终端  
**Alt+H** 或 **Esc+H**                    从一拉丁语 AIX 终端

**字段除形:**    该功能解除当前字段或行中的阿拉伯文本形状。通过下列键组合激活该功能:

**Alt+B**                                    从一阿拉伯语 AIX 终端  
**Esc+B**                                    从 ASCII 终端  
**Alt+B** 或 **Esc+B**                    从一拉丁语 AIX 终端

**上下文形状确定:**    该功能基于周围文本决定了阿拉伯字符的形状。当且仅当输入或编辑从右到左文本时使用上下文形状确定功能。当按下任意指定的形状选择键时该功能终止。它是缺省功能。通过下列键组合激活该功能:

**Alt+C**                                    从一阿拉伯语 AIX 终端  
**Esc+C**                                    从 ASCII 终端  
**Alt+C** 或 **Esc+C**                    从一拉丁语 AIX 终端

**初始形状确定:**    该功能确定阿拉伯字符的初始形状。通过下列键组合激活该功能:

**Alt+I**                                    从一阿拉伯语 AIX 终端  
**Esc+I**                                    从 ASCII 终端  
**Alt+I** 或 **Esc+I**                    从一拉丁语 AIX 终端

**中间形状确定:**    该功能确定阿拉伯字符的中间形状。通过下列键组合激活该功能:

**Alt+M**                                    从一阿拉伯语 AIX 终端  
**Esc+M**                                    从 ASCII 终端  
**Alt+M** 或 **Esc+M**                    从一拉丁语 AIX 终端

**孤立的形状确定:** 该功能确定阿拉伯字符的孤立形状。通过下列键组合激活该功能:

**Alt+O** 从一阿拉伯语 AIX 终端  
**Esc+O** 从 ASCII 终端  
**Alt+O** 或 **Esc+O** 从一拉丁语 AIX 终端

**最终形状确定:** 该功能确定阿拉伯字符的最终形状。通过下列键组合激活该功能:

**Alt+Y** 从一阿拉伯语 AIX 终端  
**Esc+Y** 从 ASCII 终端  
**Alt+Y** 或 **Esc+Y** 从一 AIX 终端

**杂项功能:** 要激活数字交换, 在命令行输入下列行:

```
export ARB_NUM_SWAP=1
```

要激活对称交换, 即, 交换双向字符, 如花括号和方括号等, 在命令行输入下列行:

```
export ARB_SYM_SWAP=1
```

指定主机使用的代码页, 在命令行输入下列一行:

```
export RM_HOST_LANG=IBM-420
```

## 终端类型协商

**telnet** 命令使用 Telnet 协议协商终端类型, 它根据所协商的结果设置 **TERM** 环境变量。

要从控制台覆盖终端协商, 使用 **EMULATE** 环境变量或 **-e** 标志; 如果需要 3270 模拟则调用 **tn3270** 命令。要确定是否执行了终端类型协商, 以下的列表描述了 **telnet** 命令处理的顺序:

1. **-e** 命令行标志。(无协商。)
2. **EMULATE** 环境变量。(无协商。)
3. **tn3270** 命令。(无协商。)
4. 如果不存在第 1, 2 和 3 步, 终端类型协商自动发生。

如果客户机和服务器协商使用 3270 数据流, 键盘映射由下列优先权决定:

**\$HOME/.3270keys** 当调用 **tn** 或 **telnet** 命令时指定用户的 3270 键盘映射。如果使用的是彩色显示器, 还可以改变该文件来定制 3270 显示器的色彩。

**/etc/map3270** 当调用 **tn3270** 命令时指定用户的 3270 键盘映射。文件 **/etc/map3270** 定义了键盘映射和 **tn3270** 命令的颜色。

**/etc/3270.keys** 指定基本的 3270 键盘映射以用于有限功能的终端。

## 安全注意键 (SAK) 选项

除终端协商外, **telnet** 命令允许对安全注意键 (SAK) 协商选项。当支持该选项时, 它为本地用户诸如改变用户标识或密码等任务提供到远程主机安全通信路径。如果远程主机支持 **SAK** 功能, 当 **telnet send sak** 子命令发出时, 一可信 shell 在远程主机上打开。**SAK** 功能也可被分配到 **telnet** 输入方式的一个单个关键变量, 使用 **set sak** 子命令。

## 尾行约定

Telnet 协议定义回车换行 (CR-LF) 序列来代表行结束 (“end-of-line”)。对终端输入而言, 它对应应在用户终端按下的命令完成或行结束键。在 ASCII 终端上, 这是一个 CR 键, 但它也有可能被标记为 “Return” 或 “Enter”。



当 Telnet 服务器收到 Telnet 行结束序列，即从一远程终端上输入的 CR-LF，效果与用户在本地终端上按下行结束键相同。

在 ASCII 服务器上，接收 Telnet 序列 CR-LF 与本地用户在本地终端上按 CR 键的效果一样。在 ASCII 服务器上，当 CR-LF 和 CR-NUL 作为从 Telnet 连接上接收的输入时效果相同。

注：Telnet 用户必须能够发送 CR-LF、CR-NULL 或 LF。ASCII 用户必须能够发送 CR-LF 或 CR-NULL。

在 ACSII 主机上，当用户按行结束键时，Telnet 用户应该有一个用户可控制的方式来发送 CR-LF 或 CR-NULL。CR-LF 应是缺省值。Telnet 行结束序列，CR-LF，必须用于发送不是从终端到计算机的 Telnet 数据。例如，当 Telnet 服务器发送输出或当 Telnet 协议合并另一应用协议时它就会发生。

**telnet** 命令“执行”（用 **exec** 命令）**/usr/sbin/login**命令来验证用户。它 1) 允许所有用户和设备属性在 telnet 连接上生效，2) 使telnet连接计数注册会话一次可允许的最大数（由 **maxlogins** 属性决定）。属性定义在文件 **/etc/security/user** 和 **/etc/security/login.cfg** 文件之中。

## 限制

- 在发送仿真高级功能终端（HFT）的转义时，早期版本的 **telnet** 命令与 **telnet** 命令的 AIX V4 以及更高版本不兼容。**telnet** 命令的 AIX V4 以及更改版本在隐藏转义密钥时仅发送一个转义，而以前版本发送两个转义字符。
- **telnet** 命令必须允许传输不是以二进制方式来实现 ISO 8859 拉丁语代码页的 8 位字符。这在 TCP/IP 命令的国际化上是必须的。
- 为了支持新字符集合，下列内容被加入到了 **terminfo** 文件的 **hft-m**、**ibm5081**、**hft**、**hft-nam**、**hft-c**、**aixterm-m** 和 **aixterm** 条目中：

```
box1=\154\161\153\170\152\155\167\165\166\164\156,      batt1=f1,
box2=\154\161\153\170\152\155\167\165\166\164\156,      batt2=f1md,
font0=\E(B,        font1=\E(0,
```
- **rlogind** 和 **telnetd** 守护程序使用 POSIX 线路规程来改变本地 tty 上的线路规程。如果 POSIX 线路规程没有用于本地 tty，回显其他线路规程可能会导致不正确的行为。AIX TCP/IP 必须有 POSIX 线路规程来正确工作。
- 使用 **telnet** 命令时不能将鼠标用作输入设备。
- **telnet** 命令不支持 APL 数据流。

## 环境变量

下列环境变量可与 **telnet** 命令一起使用：

|                |                                                                                                                                                                                                                                                                                                         |
|----------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>EMULATE</b> | 与 <b>-e</b> 标记一样覆盖终端类型协商。如果 <b>EMULATE</b> 环境变量定义为 <b>vt100</b> 或 <b>3270</b> ， <b>telnet</b> 命令分别模拟 DEC VT100 终端或 3270 终端。如果 <b>EMULATE</b> 变量未定义或值为 <b>none</b> ， <b>telnet</b> 命令正常操作。如果 <b>EMULATE</b> 变量设为 <b>vt100</b> 或 <b>3270</b> ，远程连接注册的 <b>TERM</b> 环境变量也应设为相同值。可以在打开连接后用 <b>env</b> 命令来检测。 |
| <b>TNESC</b>   | 指定除缺省值以外的备用 TELNET 转义字符，Ctrl-J 用于 <b>telnet</b> 命令，Ctrl-T 用于 <b>tn</b> 命令或 Ctrl-C 用于 <b>tn3270</b> 命令。要改变 <b>telnet</b> 转义序列，设置 <b>TNESC</b> 为您所希望用的字符的八进制值。然后导出 <b>TNESC</b> 。例如，设置 <b>TNESC</b> 为 35 以改变 TELNET 转义序列为 Ctrl-]。                                                                         |
| <b>MAP3270</b> | 指定一包含用户的 3270 键盘映射的备用文件。 <b>MAP3270</b> 变量必须包含备用文件的全路径名。用与缺省文件 <b>/etc/map3270</b> 相同的格式创建备用文件。                                                                                                                                                                                                         |

**RM\_HOST\_LANG** 指定在远程主机 3270 上使用的 EBCDIC 代码页。在 telnet (用 **telnet** 命令) 到一非英语的 3270 主机上之前, 设置环境变量 **RM\_HOST\_LANG** 到正确的代码页。缺省为英语。关于可用的代码页, 参阅《AIX 5L V5.3 本地语言支持指南和参考大全》中的『转换器编程概述』。通过指定所期望的代码页来格式化 **RM\_HOST\_LANG** 环境变量。

**注:** **tn3270** 命令不支持 DBCS, 因为它不支持 DBCS 的终端类型。

**telnet** 命令通过 **iconv** 命令转换字符。用户可通过使用 **genxlt** 命令来改变缺省转换表。

## 标志

|                        |                                                                                                           |
|------------------------|-----------------------------------------------------------------------------------------------------------|
| <b>-d</b>              | 打开调试模式。                                                                                                   |
| <b>-e TerminalType</b> | 覆盖终端类型协商。可能的值为 <b>vt100</b> 、 <b>3270</b> 或 <b>none</b> 。                                                 |
| <b>-n TraceFile</b>    | 将网络跟踪信息记录到由 <i>TraceFile</i> 变量指定的文件中。                                                                    |
| <b>-p</b>              | 保存当前 <b>tty</b> 属性。                                                                                       |
| <b>-f</b>              | 转发凭证。如果 Kerberos 5 不是当前的认证方法, 该标记将被忽略。如果当前 DCE 凭证未被标记为可转发, 认证将失败。                                         |
| <b>-F</b>              | 转发凭证。此外, 远程系统上的凭证将被标记为可转发的 (允许它们被传输到另一个远程系统)。如果 Kerberos 5 不是当前的认证方法, 该标记将被忽略。如果当前 DCE 凭证未被标记为可转发, 认证将失败。 |
| <b>-k realm</b>        | 如果与本地系统域不同, 允许用户指定远程站的域。为此, <i>realm</i> 与 DCE 单元同义。如果 Kerberos 5 不是当前的认证方法, 该标记将被忽略。                     |
| <b>-l user</b>         | 指定 telnet 希望注册为的远程用户。如果 Kerberos 5 不是当前的认证方法, 该选项将被忽略。                                                    |

## 子命令

在输入每个子命令前, 按转义键序列。转义序列指示程序没有文本信息跟随。否则, 程序将子命令解释为文本。

对下面列表中的每个子命令, 只需输入足够的字母来唯一识别该子命令。(例如, **q** 对 **quit** 子命令而言就已足够)。这同样适用于 **display**、**emulate**、**mode**、**set** 和 **toggle** 子命令的参数。

**telnet** 子命令为:

|                             |                                                                                                                                                                     |
|-----------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>? [Subcommand]</b>       | 请求关于 <b>telnet</b> 子命令的帮助。没有参数时, <b>?</b> 子命令显示一个帮助摘要。如果指定了 <i>Subcommand</i> 变量, 则该子变量的帮助信息显示出来。                                                                   |
| <b>close</b>                | 当使用 <b>open</b> 子命令来建立连接时, 关闭 TELNET 连接并返回到 <b>telnet</b> 命令方式。当调用了 <b>telnet</b> 命令并指定了主机, <b>close</b> 子命令关闭 TELNET 连接并退出 <b>telnet</b> 程序 (与 <b>quit</b> 子命令一致)。 |
| <b>display [Argument ]</b>  | 如果没有指定 <i>Argument</i> 变量, 显示所有的 <b>set</b> 和 <b>toggle</b> 值; 否则, 只列出与 <i>Argument</i> 变量相匹配的值。                                                                    |
| <b>emulate TerminalType</b> | 用指定的终端类型覆盖终端类型协商。可选项有:<br><b>?</b> 显示帮助信息。<br><b>3270</b> 模拟一个 3270 终端。<br><b>无</b> 不指定模拟。                                                                          |
| <b>vt100</b>                | 模拟一个 DEC VT100 终端。                                                                                                                                                  |

所有来自远程主机的输出由指定的仿真器处理。要模拟的初始终端类型可通过 **EMULATE** 环境变量或 **telnet** 命令的 **-e** 标志来指定。

注：在仿真方式中只允许标准 ASCII 字符。

|                                |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|--------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>mode</b> <i>Type</i>        | 指定当前输入方式。当 <i>Type</i> 变量值为 <b>line</b> 时，方式为逐行。当 <i>Type</i> 变量值为 <b>character</b> 时，方式为每次一个字符。在进入一个请求方式之前，远程主机请求许可，如果远程主机支持它，则进入新的方式。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| <b>open</b> <i>Host [Port]</i> | 打开到指定主机的连接。 <i>Host</i> 可以指定为一个主机名或一个以点分十进制数形式的因特网地址。如果未指定 <i>Port</i> 变量，则 <b>telnet</b> 子命令试图联系缺省端口上的 TELNET 服务器。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| <b>quit</b>                    | 关闭 TELNET 连接并退出 <b>telnet</b> 程序。在命令方式下 Ctrl-D 也可关闭连接并退出。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| <b>send</b> <i>Arguments</i>   | 发送一个或多个参数（特定的字符序列）到远程主机。多个参数由空格分隔。以下是可用的参数：<br><b>?</b> 显示 <b>send</b> 子命令的帮助信息。<br><b>ao</b> 发送 TELNET AO（异常输出）序列，使得远程主机清除从远程系统到本地终端的所有输出。<br><b>ayt</b> 发送 TELNET AYT（您在吗）序列，远程系统可对此作出响应。<br><b>brk</b> 发送 TELNET BRK（中断）序列，使得远程系统执行一个 kill 操作。<br><b>ec</b> 发送 TELNET EC（擦除字符）序列，致使远程主机擦除最后一个输入的字符。<br><b>el</b> 发送 TELNET EL（擦除行）序列，致使远程主机擦除当前正在输入的行。<br><b>escape</b> 发送当前 <b>telnet</b> 转字符。 <b>telnet</b> 命令的缺省转义序列为 Ctrl-], <b>tn</b> 命令的为 Ctrl-T，或者 <b>tn3270</b> 命令的为 Ctrl-C。<br><b>ga</b> 发送 TELNET GA（前进）序列，它提供远程系统一个指示本地系统返回给用户控制的机制。<br><b>ip</b> 发送 TELNET IP（中断过程）序列，它致使远程系统取消当前正在运行的过程。<br><b>nop</b> 发送 TELNET NOP（无操作）序列。<br><b>sak</b> 发送 TELNET SAK（安全注意键）序列，它致使远程系统调用可信的 shell。如果不支持 SAK，则显示一个出错信息：Remote side does not support SAK。<br><b>synch</b> 发送 TELNET SYNC 序列，它致使远程系统废弃所有已经敲入但尚未读的输入。该序列被当作 TCP/IP 紧急数据发送。 |

- set VariableValue** 设置指定的 TELNET 变量为指定值。特殊值 **off** 关闭与输入的变量关联的功能。**display** 子命令可用于查询每个变量的当前设置。可指定的变量有:
- echo** 在输入字符的本地屏幕显示与禁止本地屏幕显示间切换。本地屏幕显示用于正常处理，而禁止屏幕显示便于输入不宜显示在屏幕上的文本，如密码。该变量仅可用于逐行方式。
  - eof** 为 **telnet** 命令定义该字符。当 **telnet** 命令处于逐行方式，输入 **eof** 字符以作为一行的第一个字符发送该字符到远程主机。字符 **eof** 的初始值为本地终端文件结束符。
  - erase** 为 **telnet** 命令定义擦除字符。当 **telnet** 命令以每次一个字符方式执行，而且 **localchars** 值为 **True** 时，输入擦除字符将发送 TELNET EC 序列到远程主机。擦除字符的初始值为本地终端 ERASE 字符。
- escape**  
指定 **telnet** 转义字符，当连接到远程主机时它将 **telnet** 命令置入命令方式。该字符还可以以八进制方式指定为 **TNESC** 环境变量的值。
- flushoutput**  
为 **telnet** 命令定义刷新字符。当 **localchars** 值为 **True** 时，输入 **flushoutput** 字符将 TELNET AO 序列发送到远程主机。刷新字符的初始值为 **Ctrl-O**。如果远程主机正在运行 AIX，**flushoutput** 变量，与 **set** 子命令定义的其他特定字符不同，仅在 **localchars** 方式下工作，因为它没有 **termio** 等效命令。
- interrupt**  
为 **telnet** 命令定义中断字符。当 **localchars** 值为 **True** 时，输入中断字符将 TELNET IP 序列发送到远程主机。中断字符的初始值为本地终端中断 (**INTR**) 字符。
- kill** 为 **telnet** 命令定义 **kill** 字符。当 **telnet** 命令按每次一个字符方式执行，而且 **localchars** 值为 **True** 时，输入 **kill** 字符将发送 TELNET EL 序列到远程主机。**kill** 字符的初始值为本地终端 **KILL** 字符。
- quit** 定义 **quit** 字符给 **telnet** 命令。当 **localchars** 值为 **True** 时，输入 **quit** 字符将发送 TELNET BRK 序列到远程主机。**quit** 字符的初始值为本地终端 **QUIT** 字符。
- sak** 定义安全注意键 (**SAK**) 给 **telnet** 命令。当输入 **sak** 字符后，将要求远程系统创建一个可信的 **shell**。如果远程主机不支持 **SAK**，该序列不起作用。
- status** 显示 **telnet** 命令的状态，包括当前方式和当前连接的远程主机。

**toggle Arguments** 切换一个或多个参数控制 **telnet** 命令如何响应事件。可能的值为 **True** 和 **False**。多个参数由空格分隔。**display** 子命令可用于查询每个参数变量的当前设置。以下是可用的参数:

? 显示有效的参数给 **toggle**。

#### **autoflush**

如果 **autoflush** 和 **localchars** 皆为 **True** 而且 AO, INTR 和 QUIT 字符被认出并转换成 TELNET 序列, **telnet** 命令在用户的终端上不显示任何数据, 直到远程系统确认 (用 TELNET **timing mark** 选项) 其已经处理了这些 TELNET 序列。**autoflush** 的初始值为 **True**, 如果终端尚未设置 **stty noflush**; 如果已设置, 则为 **False**。

#### **autosynch**

如果 **autosynch** 和 **localchars** 皆为 **True**, 则输入 INTR 或 QUIT 字符将发送该字符的 TELNET 序列, 其后为 ELNET SYNC 序列。该过程使得远程主机废弃所有先前的输入, 直到这两种 TELNET 序列都已读入并工作。此切换的初始值为 **False**。

**crmod** 切换回车方式。当设置为 **True** 时, 大多数从远程主机接收到的回车字符被映射为一个回车, 其后跟随一个换行。该方式不影响用户输入的字符, 仅影响那些从远程主机上接收到的字符。当远程主机仅发送一个回车而非换行时此方式有用。此切换的初始值为 **False**。

**debug** 在套接字级别切换调试。此切换的初始值为 **False**。

#### **localchars**

决定 TELNET 特殊字符的处理。当该值为 **true** 时, ERASE、FLUSH、INTERRUPT、KILL 和 QUIT 字符被在本地识别并转换为适当的 TELNET 控制序列 (分别为 EC, AO, IP, BRK 和 EL)。当该值为 **false** 时, 这些特殊字符作为文字字符发送到远程主机。**localchars** 的初始值在逐行方式为 **true**, 在每次一字母方式下为 **false**。

#### **netdata**

切换所有网络数据 (十六进制格式) 的显示。数据被写入到标准输出, 除非 *TraceFile* 值以 **-n** 标志的形式在 **telnet** 命令行上指定。此切换的初始值为 **False**。

#### **options**

切换内部 TELNET 协议处理选项的显示, 诸如终端协商和本地或远程字符的屏幕显示。该切换的初始值为 **false**, 表明当前选项不应该显示。

#### **lineterm**

切换缺省行结束终止器为 CR-LF (ASCII 回车换行)。当用户按行结束键时, 运行在 ASCII 主机上的 **telnet** 客户机应该有用户可配置的选项来发送 CR-NUL 或 CR-LF 终止器。此切换的初始值为 **False**。

**z** 暂挂 TELNET 进程。要返回 TELNET 进程, 用 **fg** 内置命令 **cs**h 或 **ksh** 命令。

注: 对任意其他进程, **z** 子命令有着与 Ctrl-Z 键序列相同的效果。它暂挂 **Telnet** 执行并将您返回到原来的登录 shell。

## 认证

如果系统是为 Kerberos 5 认证配置的, **telnet** 客户机将尝试认证协商。**telnet** 所用的认证协商以及选项和子选项的定义在 rfc 1416 中作了定义。

如果客户机与服务器同意某一种认证, 它们将交换认证信息, 包括客户机期望访问的帐户。这将会是本地用户, 除非设置了 **-l** 标志。

如果它们不能就认证信息达成同意, 或者失败了, **telnet** 连接将继续标准连接 (假设配置了标准 AIX)。

仅当下列所有条件都得到满足时远程主机允许存取:

- 本地用户具有当前 DCE 凭证。
- 远程系统接受 DCE 凭证作为访问远程帐户足够的凭证。关于附加信息, 请参阅 **kvalid\_user** 函数。

## 示例

在以下示例中，如果输入 **tn** 命令，而非 **telnet** 命令，命令方式提示符将显示为 **tn>**。

1. 要登录到远程主机 **host1** 并执行终端协商，输入：

```
telnet host1
```

2. 要作为终端 **vt100** 登录到 **host1** 上（无终端类型协商），选择下面方法中的一种：

- a. 用下列命令为该登录会话设置 **EMULATE** 环境变量，然后输入 **telnet** 命令：

```
EMULATE=vt100; export EMULATE
telnet host1
```

- b. 用 **-e** 标志仅为该 **telnet** 会话设置终端类型：

```
telnet -e vt100 host1
```

3. 要登录到远程主机上，然后检查 **telnet** 程序的状态，请输入：

```
telnet host3
```

当登录提示符出现时，输入您的登录标识和密码。按 **Ctrl-T** 按键顺序接收 **telnet>** 提示符。在 **telnet>** 提示符后输入以下内容：

```
status
```

与下列内容类似的信息显示在屏幕上：

```
Connected to host3.
Operating in character-at-a-time mode.
Escape character is '^['.
```

**status** 子命令完成后，按 **Enter** 键返回到远程提示符。

一旦完成了登录，就可以发出命令。要注销系统并关闭连接，按 **Ctrl-D** 键序列，或输入 **exit**。

4. 要用 **tn3270** 命令登录到远程主机上，输入：

```
tn3270 hostname
```

主机的登录屏幕应显示出来。这时就可以输入登录标识和密码。一旦完成了登录，就可以发出命令。按 **Ctrl-D** 或输入 **exit** 退出系统并断开连接。

## 文件

**/etc/3270.keys** 定义基本 3270 键盘映射以用于有限功能的终端。

## 相关信息

**env** 命令，**ftp** 命令，**login** 命令，**rcp** 命令，**rexec** 命令，**rlogin** 命令，**rsh** 命令。

**telnetd** 守护程序。

**kvalid\_user** 函数。

**map3270** 文件格式，**.3270keys** 文件格式。

《网络与通信管理》中的『通信与网络』。

《网络与通信管理》中的『与远程用户对话』。

《网络与通信管理》中的『认证和安全 rcmds』。

---

## telnetd 守护程序

### 用途

为 TELNET 协议提供服务器功能。

### 语法

```
/usr/sbin/telnetd [ -a ] [ -n ] [ -s ]
```

### 描述

注: **telnetd** 守护程序通常由 **inetd** 守护程序启动。也可使用 **SRC** 命令从命令行控制。

**/usr/sbin/telnetd** 守护程序为一服务器, 它支持 Defense Advanced Research Product Agency (DARPA) 标准 Telnet 协议 (TELNET)。**telnetd** 守护程序的变动要由系统管理界面工具 (SMIT) 来完成。

对 **telnetd** 守护程序的改动可通过系统管理界面工具 (SMIT) 或系统资源控制器 (SRC) 来完成, 即通过编辑 **/etc/inetd.conf** 或 **/etc/services** 文件。建议不要在命令行输入 **telnetd**。**telnetd** 守护程序缺省设置为当其未在 **/etc/inetd.conf** 文件中被注释时启动。缺省情况下, **-a** 标记也打开了。

**inetd** 守护程序从 **/etc/inetd.conf** 文件和 **/etc/services** 文件获取其信息。

在对 **/etc/inetd.conf** 或 **/etc/services** 文件作改动之后, 运行 **refresh -s inetd** 或 **kill -1 InetdPID** 命令来通知 **inetd** 守护程序其配置文件的变动。

当启动 **telnet** 会话后, **telnetd** 守护程序发送 TELNET 选项到客户 (远程) 主机指示其有能力执行选项。

### 终端协商

**telnetd** 守护程序向客户机主机的要求终端类型。收到以后, **telnetd** 守护程序检测所指示的类型是否为本地系统所支持。如果不支持, 守护程序重新要求一个终端类型。

该终端类型协商继续直到远程客户机发送一个可接受的终端类型或直到客户机在一行内发送两次相同的类型, 表明它已经没有其他可用的类型。需要时, **telnetd** 守护程序参考 **/etc/telnet.conf** 文件将客户机的终端类型字符串翻译成 **terminfo** 文件条目。

注: 因为 **telnetd** 守护程序允许发送和接收 8 位 ASCII, 所以 NLS 受支持。

如果远程客户机发送 TELNET **SAK** 命令, **telnetd** 守护程序通过 PTY 传递本地 **SAK** 字符来调用可信的 shell。

**telnetd** 守护程序支持下列 TELNET 选项:

- 二进制
- 回显/不回显
- 支持 SAK
- 禁止向前
- 计时标记

- 协商窗口大小 (NAWS)
- 认证

**telnetd** 守护程序还识别远程客户机的下列选项:

- 二进制
- 禁止向前
- 回显/不回显
- 终端类型

**telnetd** 守护程序通过系统管理界面工具 (SMIT) 或改变 **/etc/inetd.conf** 文件来控制。建议不要在命令行输入 **telnetd**。

## 认证协商

如果系统配置了 Kerberos 5 认证, **telnetd** 将接受认证选项协商。如果双方同意 Kerberos 5 认证, 客户机将忽略 DCE 主体而且 **telnetd** 将用 **kvalid\_user** 例程来决定 DCE 主体是否可访问帐户。如果它通过, 就不再请求密码。

## 用系统资源控制器来操纵 telnetd 守护程序

**telnetd** 守护程序是 **inetd** 守护程序的子服务器, 而后者是系统资源控制器 (SRC) 的子系统。**telnetd** 守护程序是 **tcpip** SRC 子系统组的成员。缺省情况下在 **/etc/inetd.conf** 文件中启用该守护程序, 并可用下列的 SRC 命令对进其行操作:

|                 |                     |
|-----------------|---------------------|
| <b>startsrc</b> | 启动子系统、子系统组或子服务器。    |
| <b>stopsrc</b>  | 停止子系统、子系统组或子服务器。    |
| <b>lssrc</b>    | 获取子系统、子系统组或子服务器的状态。 |

## 标志

|           |                                          |
|-----------|------------------------------------------|
| <b>-a</b> | 使 PTY 和套接字在内核之中直接链接, 数据处理得以保留在内核之中以提高性能。 |
| <b>-n</b> | 禁用传输层保持活动消息。缺省启用消息。                      |
| <b>-s</b> | 打开套接字级别的调试。                              |

## 安全性

**telnetd** 守护程序是支持 PAM 并带有服务名称 **telnet** 的应用程序。在 **/etc/security/login.cfg** 的 **usw** 节中, 通过修改 **auth\_type** 属性值来将要使用 PAM 进行认证的系统范围配置设置为作为 root 用户的 **PAM\_AUTH**。

启用 PAM 时使用的认证机制取决于 **/etc/pam.conf** 中的 **telnet** 服务的配置。**telnetd** 守护程序需要 **auth**、**account**、**password** 和 **session** 模块类型的 **/etc/pam.conf** 条目。以下列出了在 **/etc/pam.conf** 中 **telnet** 服务的建议配置:

```
#
# AIX telnet 配置
#
telnet auth      required      /usr/lib/security/pam_aix
telnet account   required      /usr/lib/security/pam_aix
telnet password  required      /usr/lib/security/pam_aix
telnet session   required      /usr/lib/security/pam_aix
```



## 示例

注: **telnetd** 守护程序的参数可由 **SMIT** 或编辑 **/etc/inetd.conf** 文件来指定。

1. 要启动 **telnetd** 守护程序, 输入如下:

```
startsrc -t telnet
```

该命令启动 **telnetd** 子服务器。

2. 要正常停止 **telnetd** 守护程序, 输入如下:

```
stopsrc -t telnet
```

该命令允许启动所有暂挂的连接并完成现有的连接, 但会阻止启动新的连接。

3. 要强制停止 **telnetd** 守护程序和所有 **telnetd** 连接, 输入如下:

```
stopsrc -t -f telnet
```

该命令会立刻终止所有暂挂的连接和现有的连接。

4. 要显示 **telnetd** 守护程序的简短状态报告, 输入如下:

```
lssrc -t telnet
```

该命令返回守护程序名、进程标识以及状态 (活动的或不活动的)。

## 文件

**terminfo** 描述终端能力。

## 相关信息

**ftp** 命令、**kill** 命令、**lssrc** 命令、**rcp** 命令、**refresh** 命令、**rlogin** 命令、**rsh** 命令、**startsrc** 命令、**stopsrc** 命令和 **telnet** 命令。

**kill** 命令、**lssrc** 命令、**refresh** 命令、**startsrc** 命令、**stopsrc** 命令和 **telnet** 命令。

文件格式 **/etc/inetd.conf**, 文件格式 **/etc/telnet.conf**。

**pty** 特殊文件。

**kvalid\_user** 子例程。

《网络与通信管理》中的『传输控制协议』。

《网络与通信管理》中的『TCP/IP 守护程序』。

《网络与通信管理》中的『认证和安全 rcmds』。

---

## termdef 命令

### 用途

查询终端特征。

## 语法

**termdef** [ **-c** | **-l** | **-t** ]

## 描述

**termdef**命令标识当前显示类型，活动行的设置，或当前的列设置。当转换字体时，这会简化行与列的重新设置；当转换显示时也会简化 **TERM** 环境变量的重新设置。 **terminfo** 数据库定义了每一类显示的行数与列数的缺省值，但行数与列数可根据当前使用的是哪一种字体而变化。而且，**TERM** 环境变量不能自动反映当前活动的显示

**termdef** 命令的标志是互斥的。如果在命令中使用多个标志，**termdef** 命令会识别并仅仅返回第一个标志的当前值。任意别的标志都被忽略。例如，**termdef -lc** 命令仅返回当前显示的活动的行的设置。

## 标志

- c** 返回当前列的值。
- l** 返回当前行的值。
- t** 返回当前显示名称（缺省操作）。

## 示例

为了确定当前 **TERM** 环境变量的值，输入：

```
termdef -c
```

## 文件

`/usr/bin/termdef` 包含 **termdef** 命令。

---

## test 命令

### 用途

评估条件表达式。

### 语法

**test** *Expression*

或

[ *Expression* ]

### 描述

**test** 命令评估 *Expression* 参数，如果表达式值为 **True**，返回一个零（**true**）退出值。否则，**test** 命令返回一个非零（**false**）退出值。如果没有参数，**test** 命令也返回一个非零退出值。

注：

1. 在命令的第二种格式中，[ ]（方括号）必须被空白包围。
2. 必须在 C shell 中对文件名进行显式测试。文件名替换（全局）导致 shell 脚本退出。

函数与运算符被 **test** 命令当作独立的参数。 *Expression* 参数指的是一个需要被检查为 true 或 false 的条件的语句。下列函数可用来构造此参数:

|                                            |                                                                                                                                                        |
|--------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-b</b> <i>FileName</i>                  | 如果所指定的 <i>FileName</i> 存在且是一个块特殊文件, 则返回一个 true 退出值。                                                                                                    |
| <b>-c</b> <i>FileName</i>                  | 如果所指定的 <i>FileName</i> 存在且是一个字符特殊文件, 则返回一个 true 退出值。                                                                                                   |
| <b>-d</b> <i>FileName</i>                  | 如果所指定的 <i>FileName</i> 存在且是一个目录, 则返回一个 true 退出值。                                                                                                       |
| <b>-e</b> <i>FileName</i>                  | 如果所指定的 <i>FileName</i> 存在, 则返回一个 true 退出值。                                                                                                             |
| <b>-f</b> <i>FileName</i>                  | 如果所指定的 <i>FileName</i> 存在且是一个常规文件, 则返回一个 true 退出值。                                                                                                     |
| <b>-g</b> <i>FileName</i>                  | 如果所指定的 <i>FileName</i> 存在且它的设置组标识位已设置, 则返回一个 true 退出值。                                                                                                 |
| <b>-h</b> <i>FileName</i>                  | 如果所指定的 <i>FileName</i> 存在且是一个符号链接, 则返回一个 true 退出值。                                                                                                     |
| <b>-k</b> <i>FileName</i>                  | 如果所指定的 <i>FileName</i> 存在且其粘滞位已设置, 则返回一个 true 退出值。                                                                                                     |
| <b>-L</b> <i>FileName</i>                  | 如果所指定的 <i>FileName</i> 存在且是一个符号链接, 则返回一个 true 退出值。                                                                                                     |
| <b>-n</b> <i>String1</i>                   | 如果 <i>String1</i> 变量的长度为非零, 则返回一个 true 退出值。                                                                                                            |
| <b>-p</b> <i>FileName</i>                  | 如果所指定的 <i>FileName</i> 存在且是一个命名管道 (FIFO), 则返回一个 true 退出值。                                                                                              |
| <b>-r</b> <i>FileName</i>                  | 如果所指定的 <i>FileName</i> 存在且可被当前进程读取, 则返回一个 true 退出值。                                                                                                    |
| <b>-s</b> <i>FileName</i>                  | 如果所指定的 <i>FileName</i> 存在且长度大于零, 则返回一个 true 退出值。                                                                                                       |
| <b>-t</b> <i>FileDescriptor</i>            | 如果文件描述符号为 <i>FileDescriptor</i> 的文件是打开的且与一个终端相连, 则返回一个 true 退出值。                                                                                       |
| <b>-u</b> <i>FileName</i>                  | 如果所指定的 <i>FileName</i> 存在且它的设置用户标识位已被设置, 则返回一个 true 退出值。                                                                                               |
| <b>-w</b> <i>FileName</i>                  | 如果所指定的 <i>FileName</i> 存在且写标志打开, 则返回一个 true 退出值。但是, <i>FileName</i> 在一个只读的文件系统中不可写, 即使 <b>test</b> 指示为 true。                                           |
| <b>-x</b> <i>FileName</i>                  | 如果所指定的 <i>FileName</i> 存在且执行标志打开, 则返回一个 true 退出值。如果所指定的文件存在且是一个目录, true 退出值表示当前进程被允许在目录中搜索。                                                            |
| <b>-z</b> <i>String1</i>                   | 如果 <i>String1</i> 变量的长度为 0 (零), 则返回一个 true 退出值。                                                                                                        |
| <i>String1</i> = <i>String2</i>            | 如果 <i>String1</i> 与 <i>String2</i> 的变量相同, 则返回一个 true 退出值。                                                                                              |
| <i>String1</i> != <i>String2</i>           | 如果 <i>String1</i> 与 <i>String2</i> 的变量不相同, 则返回一个 true 退出值。                                                                                             |
| <i>String1</i>                             | 如果 <i>String1</i> 变量不是一个空字符串, 则返回一个 true 退出值。                                                                                                          |
| <i>Integer1</i> <b>-eq</b> <i>Integer2</i> | 如果 <i>Integer1</i> 与 <i>Integer2</i> 的变量在数学上相等, 则返回一个 true 退出值。任一比较 <b>-ne</b> , <b>-gt</b> , <b>-ge</b> , <b>-lt</b> , <b>-le</b> 可被用来代替 <b>-eq</b> 。 |
| <i>file1</i> <b>-nt</b> <i>file2</i>       | 如果 <i>file1</i> 比 <i>file2</i> 更新, 则为 true。                                                                                                            |
| <i>file1</i> <b>-ot</b> <i>file2</i>       | 如果 <i>file1</i> 比 <i>file2</i> 旧, 则为 true。                                                                                                             |
| <i>file1</i> <b>-ef</b> <i>file2</i>       | 如果 <i>file1</i> 是 <i>file2</i> 的另外一个名称, 则为 true。                                                                                                       |

这些函数可与下列运算符连在一起:

|                     |                                                    |
|---------------------|----------------------------------------------------|
| <b>!</b>            | 一元“非”运算符                                           |
| <b>-a</b>           | 二进制“与”运算符                                          |
| <b>-o</b>           | 二进制“或”运算符 (就是说, <b>-a</b> 运算符的优先权比 <b>-o</b> 运算符高) |
| <b>(Expression)</b> | 分组括号                                               |

## 退出状态

该命令返回以下退出值:

|              |                                  |
|--------------|----------------------------------|
| <b>0</b>     | <i>Expression</i> 参数为 true。      |
| <b>1</b>     | <i>Expression</i> 参数为 false 或丢失。 |
| <b>&gt;1</b> | 发生错误。                            |

## 示例

1. 为测试一个文件是否存在且非空, 输入:

```
if test ! -s "$1"
那么
    echo $1 不存在或是空的。
fi
```

如果被第一个位置的参数指定给 shell 过程的文件, \$1, 不存在, **test** 命令显示一条出错消息。如果 \$1 存在且大小大于 0, **test** 命令什么也不显示。

**注:** 在 **-s** 函数与文件名之间必须有一空格。

\$1 两边的引号确保即使值 \$1 为一空字符串, test 也可正常工作。如果引号被省略且 \$1 是空字符串, **test** 命令显示出错消息 test: 期望参数。

2. 为做一复杂比较, 输入:

```
if [ $# -lt 2 -o ! -e "$1" ]
那么
    存在
fi
```

如果给出的 shell 过程少于两个位置参数或被 \$1 指定的文件不存在, 则 shell 过程退出。特殊 shell 变量 \$# 表示了命令行输入的用以运行 shell 过程的位置参数的个数。

《操作系统与设备管理》中的『Shell』总体上描述了 shell、定义了对理解 shell 有帮助的术语并描述了很多有用的 shell 功能。

## 文件

`/usr/bin/test` 包含 **test** 命令。

## 相关信息

**bsh** 命令, **cs** 命令, **find** 命令, **ksh** 命令, **sh** 命令。

《操作系统与设备管理》中的『Shell』。

---

## tftp 或 utftp 命令

### 用途

使用次要文件传输协议 (TFTP) 在主机间传输文件。

### 语法

```
{tftp | utftp} { -g | -o | -p | -r | -w } LocalName HostPort RemoteName [ netascii | image ] [blksize #] [timeout #] [tsize]
```

## 交互格式语法

## 命令行格式语法

### 描述

`/usr/bin/tftp` 与 `utftp` 命令使用次要文件传输协议 (TFTP) 在主机间传输文件。由于 TFTP 是最小的文件传输协议, `tftp` 与 `utftp` 命令不提供 `ftp` 命令的所有功能。例如, `tftp` 与 `utftp` 命令没有提供列出远程文件或更改远程主机目录的能力, 仅授予远程 TFTP 服务器有限的文件访问权限。`utftp` 命令是 `tftp` 命令的一种格式, 用于管道中。

远程主机必须有一个由它的 `inetd` 守护程序启动的 `tftpd` 守护程序和一个定义为限制 `tftpd` 守护程序访问的帐户。使用由 `tftpd` 命令定义的过程来设置 TFTP 环境和无人帐户。

注: 当主机正在以安全方式运行时, `tftp` 与 `utftp` 命令不可用。

如果有不适当 (将来的) 块号的块到达, `tftp` 命令忽略对任一已发送块的重复应答, 发送一个错误包并且退出。如果已经接收重复数据块并发送了一个错误包并且退出, 同样会忽略那些重复数据块。

### RFC2349 选项协商

从 AIX 5.3 开始, `tftp` 客户机能够同服务器协商以下 TFTP 选项: 块大小 (`blksize`)、传送大小 (`tsize`) 和超时 (`timeout`)。较大的传送块大小能够提高传送性能, `tsize` 在传送之前报告文件大小以检查可用空间, `timeout` 则协商重新发送的超时时间。TFTP 服务器必须支持 RFC2349, 这样才能进行选项协商。

### 访问控制

搜索 `/etc/tftpaccess.ctl` 文件以查找那些以 `allow:` 或 `deny:` 开头的行。其余行被忽略。如果文件不存在, 也允许访问。可以访问所允许的目录和文件, 不能访问被拒绝的目录。例如, 可以允许访问 `/usr` 目录而 `/usr/ucb` 目录则会被拒绝。这意味着在 `/usr` 目录中的任一目录或文件 (除了 `/usr/ucb` 目录) 都可被访问。`/etc/tftpaccess.ctl` 文件中的项必须为绝对路径名。

`/etc/tftpaccess.ctl` 文件应为 `root` 用户只写而所有 `groups` 和 `others` 可读 (即为有许可权 644 的 `root` 用户所拥有) 可读。用户 `nobody` 必须能够读取 `/etc/tftpaccess.ctl` 文件。否则, `tftpd` 守护程序就不能识别文件是否存在并允许访问整个系统。有关更多信息, 请参阅样本 `tftpaccess.ctl` 文件, 它驻留在 `/usr/samples/tcpip` 目录。

搜索算法假设在 `tftp` 命令中所用的本地路径名是绝对路径名。该算法搜索 `/etc/tftpaccess.ctl` 文件以查找 `allow:/`。它使用每一个部分路径名来重复搜索所允许的路径名, 部分路径名通过从文件路径名添加下一个组件构造而成。匹配的最长路径名也是被允许的一种。然后对于拒绝路径名进行同样操作, 从匹配最长允许路径名开始。

例如, 如果文件路径名为 `/a/b/c` 而 `/etc/tftpaccess.ctl` 文件包含 `allow:/a/b` 和 `deny:/a`, 将会产生一个允许匹配 (`/a/b`) 并且不产生以 `/a/b` 开始的拒绝匹配, 并允许访问。

如果 `/etc/tftpaccess.ctl` 文件包含 `allow:/a` 和 `deny:/a/b`, 将会产生一个允许匹配 (`/a`) 和一个以 `/a` (`/a/b`) 开始的拒绝匹配, 并拒绝访问。如果 `/etc/tftpaccess.ctl` 文件既包含 `allow:/a/b` 也包含 `deny:/a/b`, 将会拒绝访问, 因为允许的路径名被首先搜索。

注: 有关 Xstations、无盘客户机和受限项的更多信息与示例配置可在 `/usr/samples/tcpip/tftpaccess.ctl` 文件中找到。

`tftp` 与 `utftp` 命令有两种格式: 交互格式和命令行格式。

## 交互格式

在交互格式中，**tftp** 与 **utftp** 命令单独发出或与指定在此次会话中用于文件传输的缺省主机的 *Host* 参数一起执行。如果进行选择，也可以用 *Port* 参数来指定 **tftp** 或 **utftp** 连接要使用的端口，例如在 `/etc/services` 文件中为 **mail** 指定的端口。当您输入这些命令中的任一个的交互格式时，会显示 `tftp>` 提示。

当给一个远程主机传输数据时，被传输的数据放在由 *RemoteName* 参数指定的目录中。远程主机名必须是一个完整指定的文件名，远程文件必须存在并已设置其他用户的写许可权。**tftp** 命令尝试将数据写入指定文件。但是，如果远程 TFTP 服务器没有适当权限写远程文件，或该文件已不存在，传输就失败。这可使用 **tftpd** 守护程序进行重设。

## 命令行格式

**tftp** 与 **utftp** 命令的命令行格式是等价的，除了 **utftp** 命令不覆盖本地文件以外。**tftp** 命令会覆盖文件，但在执行前会提示用户。因为不是交互式的，**utftp** 命令的命令行格式在管道中比 **tftp** 命令更有用。在命令行格式中，任一命令的所有参数都要在命令行进行指定，不会显示提示。

## 子命令

**tftp** 与 **utftp** 子命令可在其交互格式或命令行格式中输入。

### 交互格式中使用的子命令

一旦显示 `tftp>` 提示，会发出以下子命令：

|                                            |                                                                                                                                                                                      |
|--------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>?</b> [ <i>Subcommand</i> ]             | 显示帮助信息。如果指定了一个 <i>Subcommand</i> 参数，则仅显示关于该子命令的信息。                                                                                                                                   |
| <b>ascii</b>                               | <b>mode ascii</b> 子命令的同义词。                                                                                                                                                           |
| <b>binary</b>                              | <b>mode binary</b> 子命令的同义词。该子命令用在交互方式中。 <b>image</b> 子命令完成与 <b>mode binary</b> 子命令同样的功能，但用于命令行。                                                                                      |
| <b>blksize</b> 字节数                         | 启用与服务器的 <b>blksize</b> 选项协商。如果协商成功，就能够显著提高传送速率。传送块大小至少必须为 8 个八位字节，最高可以达到 65464 个八位字节。缺省值为 512 个八位字节。                                                                                 |
| <b>connect</b> <i>Host</i> [ <i>Port</i> ] | 为文件传输设置远程主机，同时也有选择地设置端口。由于 TFTP 协议不会维护传输间的连接，因此 <b>connect</b> 子命令不会创建到指定主机的连接，但会为传输操作而将它存储起来。因为远程主机可被指定为 <b>get</b> 或 <b>put</b> 子命令（可对以前指定的任何主机进行重设）的一部分， <b>connect</b> 子命令就不需要了。 |

**get** *RemoteFile* [*LocalFile*]

**get** *RemoteFile RemoteFile RemoteFile* [*RemoteFile* . . . ]

从远程主机获取一个或一组文件到本地主机。每一个 *RemoteFile* 参数可用以下两种方法之一进行指定：

- 如果缺省主机已被指定，则指定为一个存在于远程主机上的文件 (*File*)。
- 指定为一个主机文件 (*Host:File*)，其中 *Host* 是远程主机，*File* 是要复制到本地系统的文件名。如果使用这种参数格式，最后一个指定的主机变成在此 **tftp** 会话中用于稍后传输的缺省主机。

**mode** *Type*

将传输方式的类型 (*Type*) 设置为 **ascii** 或 **binary**。缺省传输方式为 **ascii**。

**put** *LocalFile* [*RemoteFile*]

**put** *LocalFile LocalFile LocalFile* [*LocalFile . . .*] *RemoteDirectory*

将一个或一组文件从本地主机放到远程主机。 *RemoteDirectory* 与 *RemoteFile* 参数可用以下两种方法之一进行指定:

- 如果缺省主机已被指定, 则指定为一个存在于远程主机上的文件或目录。
- 使用 *Host:RemoteFile* 参数, 其中 *Host* 为远程主机, *RemoteFile* 是远程系统上的文件名或目录名。如果使用这种参数格式, 最后一个指定的主机变成在此 **tftp** 会话中用于稍后传输的缺省主机。

在任一情况下, 远程文件或目录名必须是完整指定的路径名, 即使本地和远程目录同名。如果指定一个远程目录, 远程主机就被假定为 UNIX 机器。 **put** 子命令的缺省值为 write-replace, 但可在 **tftpd** 守护程序中添加一个选项以允许 write-create。

**quit**  
**status**

退出 **tftp** 会话。文件结束符按键顺序也退出程序。

显示 **tftp** 程序的当前状态, 例如包括: 当前传输方式 (**ascii** 或 **binary**)、连接状态与超时值。

**timeout** *Value*

将总的传输超时设置为由 *Value* 参数指定的秒数。 *Value* 参数必须为 1 秒或者更大 (缺省值为 5 秒)。

**trace**  
**tsize**

打开或关闭数据包跟踪。

启用与服务器的 **tsize** 选项协商。该选项允许在传送启动之前就知道文件大小。如果超过分配, 就返回错误, 文件传输将不会进行。

**verbose**

打开或关闭在文件传输期间显示额外信息的详细方式。

## 命令行格式中使用的子命令

在此格式中, 如果 *Action* 标志为:

**-w** 或 **-p**

写 (或放) 由 *LocalName* 参数指定的本地数据到由 *Host* 参数指定的远程主机上的由 *RemoteName* 参数指定的文件。如果 *LocalName* 参数是一个文件名, **tftp** 命令传输指定的本地文件。如果 *LocalName* 参数指定为一个 - (破折号), **tftp** 命令从本地标准输入传输数据到远程主机。当 *LocalName* 参数为标准输入时, **tftp** 命令允许在超时前所有输入有 25 秒的时间。

**-r** 或 **-g** 或 **-o**

从由 *Host* 参数指定的远程主机上的由 *RemoteName* 参数指定的文件读 (或获取) 远程数据, 并将它写入由 *LocalName* 参数指定的文件。如果 *LocalName* 参数是一个文件名, **tftp** 命令将数据写到指定的本地文件。对于 **-r** 与 **-g** 操作, **tftp** 命令会在覆盖现有本地文件之前提示确认。对于 **-o** 操作, **tftp** 命令不提示就覆盖现有本地文件。如果 *LocalName* 参数被指定为一个 - (破折号), **tftp** 命令将数据写到本地标准输出。

**注:** 由于 **tftp -g** 与 **tftp -r** 命令在覆盖现有本地文件之前会进行提示, 故在管道中使用 **tftp** 命令是不切实际的。 **utftp** 命令执行与 **tftp** 命令一样的 **-r** 和 **-g** 操作, 但在覆盖本地文件之前就会停止。因此, **utftp** 命令更适合在管道中使用。

对于以下文件传输方式来说, *RemoteName* 参数是对其他用户设置了写许可权的文件的名称。注意, 如果 *RemoteName* 参数包含 shell 特殊字符, 则该参数必须用双引号 (" ") 括起来。

传输方式为以下之一:

**netascii** 将数据作为 8 位传输字节的 7 位 ASCII 字符来传输。这是缺省值。

**image** 将数据作为 8 位传输字节的 8 位二进制数据字节来传输，而不用转换。当在两个主机间进行传输时，**image** 传输比 **netascii** 传输更有效。当从工作站传输 ASCII 文件到不同类型的主机时，推荐使用 **netascii** 传输。

## 示例

以下示例区分了 **tftp** 命令的交互格式与命令行格式之间的差别：

### 使用 **tftp** 命令的交互格式

要输入 **tftp** 命令、检查当前状态、连接到远程主机以及将一个文件从远程主机传输到本地主机，请输入：

```
tftp
```

显示 **tftp>** 提示。紧跟该 提示输入 **status** 子命令：

```
status
```

在屏幕上显示类似于以下的消息：

```
Not connected.
Mode: netascii  Verbose: off  Tracing: Off
Max-timeout: 25 secondstftp> _
```

在 **tftp>** 提示后，请输入 **connect** 子命令以及想要连接到的远程系统名称：

```
tftp> connect host1
```

将显示 **tftp>** 提示，作为您已连接到 **host1** 上的指示。紧跟 **tftp>** 提示，输入 **get** 子命令，以将文件 **update** 从远程主机传输到本地主机。

```
get /home/alice/update update
```

远程主机上的 **/home/alice** 目录必须为其他用户设置读许可权。从 **host1** 上将 **/home/alice/update** 文件传输到本地系统上的 **update** 文件。此例中，用户连接到 **host1**，从 **host1** 上将 **update** 文件传输到本地主机。

### 使用 **tftp** 命令的命令行格式

1. 要从远程主机复制一个文本文件并将它写到本地文件，请输入：

```
tftp -g newsched host1 /home/john/schedule
$ _
```

此例中，从远程主机 **host1** 复制 **/home/john/schedule** 文件并写到本地文件 **newsched**。

2. 要从远程主机复制文件并将输出重定向到本地主机的标准输出，请输入：

```
tftp -g - host3 /etc/hosts
```

如果成功复制，在屏幕上显示类似于以下的信息：

```
192.100.13.3 nameserver
192.100.13.3 host2
192.100.13.5 host1
192.100.13.7 host3
192.100.13.3 timeserver
Received 128 bytes in 0.4 seconds
$ _
```



此例中，从远程主机 `host3` 复制 `/etc/hosts` 文件并将输出重定向到本地主机的标准输出。

3. 要从远程主机复制文件、将文件输送给 `grep` 命令并将它写到本地文件，请输入：

```
utftp -g - host1 /home/john/schedule | grep Jones > jones.todo
$ _
```

此例中，从远程主机 `host1` 复制 `/home/john/schedule` 文件。然后文件被输送给 `grep` 命令并写入到本地文件 `jones.todo`。

4. 要从别的系统复制文件，请输入：

```
tftp -p /home/jeanne/test host2 /tmp/test
```

如果成功复制，在屏幕上显示类似于以下的信息：

```
Sent 94146 bytes in 6.7 seconds
```

此例中，`/home/jeanne/test` 文件被发送到远程主机 `host2` 上的 `/tmp` 目录。

5. 要将一个二进制文件复制到另一个系统，请输入：

```
tftp -p core host3 /tmp/core image
```

如果成功复制，在屏幕上显示类似于以下的信息：

```
Sent 309295 bytes in 15 seconds
```

此例中，从当前目录将二进制文件 `core` 发送到远程主机 `host3` 上的 `/tmp` 目录。

## 文件

`/etc/tftpdaccess.ctl` 允许或拒绝对文件和目录进行访问。

## 相关信息

`ftp` 命令、`grep` 命令、`rcp` 命令。

`tftpd` 守护程序、`inetd` 守护程序、`tftpd` 守护程序、`syslogd` 守护程序。

`hosts` 文件格式、`services` 文件格式。

《网络与通信管理》中的『使用 `tftp` 和 `utftp` 命令的文件传输』。

《网络与通信管理》中的『通信与网络』。

---

## tftpd 守护程序

### 用途

为次要文件传输协议提供服务器功能。

### 语法

```
/usr/sbin/tftpd [ -c ] [ -n ] [ -p ] [ -v ] [ -t ] [ -s ] [ -x ] [ -z ] [ -d Directory ] [ -r Option ]
```

## 描述

注: **tftpd** 守护程序通常由 **inetd** 守护程序启动。也可使用 **SRC** 命令从命令行控制。

**/usr/sbin/tftpd** 守护程序运行次要文件传输协议 (TFTP) 服务器。使用 TFTP 发送的文件可在由 **tftp** 或 **utftp** 命令行上给出的完整路径名指定的目录中找到。

注: 当审计系统在使用中时, **tftp** 命令、**utftp** 命令和 **tftpd** 服务器不可用。关于更多信息, 请参阅『TCP/IP 安全性』、『审计概述』和 **audit** 命令。

使用系统管理界面程序 (SMIT) 或系统资源控制器 (SRC) 通过编辑 **/etc/inetd.conf** 或 **/etc/services** 文件可以对 **tftpd** 守护程序进行更改。当没有在 **/etc/inetd.conf** 文件中注释掉时, 缺省情况下启动 **tftpd** 守护程序。

**inetd** 守护程序从 **/etc/inetd.conf** 文件和 **/etc/services** 文件中获取其信息。

在更改 **/etc/inetd.conf** 或 **/etc/services** 文件之后, 运行 **refresh -s inetd** 或 **kill -1 InetdPID** 命令以通知 **inetd** 守护程序其配置文件的更改情况。

**tftpd** 服务器应有一个有最小特权的用户标识。**nobody** 标识允许最小权限, 是缺省用户标识。

**tftpd** 守护程序应使用系统管理界面程序 (SMIT) 或通过更改 **/etc/inetd.conf** 文件来进行控制。不推荐在命令行中输入 **tftpd**。

在 AIX 4.3.3 中, **tftpd** 服务器是一个多线程应用程序。AIX 5.3 的另一个主要更改是服务器处理选项协商 (RFC2349) 的能力。这种新的能力允许客户机协商将要传送的文件大小。还允许协商超时和较大的块大小。只有对于读请求 (RRQ) 才协商块大小 (**blksize**)。因此, 能够显著提高使用 TFTP 的无盘节点的引导时间性能。

读请求和写请求的“传送大小”选项 (**tsize**) 协商允许在传送之前知道文件大小, 这样如果超过分配, 就会在传送启动之前发出错误消息。超时选项 (**timeout**) 允许客户机与服务器协商重新发送的超时时间 (在 1 至 255 秒之间)。**TFTP** 服务器还必须支持 RFC2349, 这样才能进行选项协商。

## tftpdaccess.ctl 文件

搜索 **/etc/tftpdaccess.ctl** 文件以查找那些以 **allow:** 或 **deny:** 开头的行。其余行被忽略。如果文件不存在, 也允许访问。除去被拒绝的目录和文件, 可以访问所允许的目录和文件。例如, 可以允许访问 **/usr** 目录而 **/usr/ucb** 目录则会被拒绝。这意味着在 **/usr** 目录中的任一目录或文件 (除了 **/usr/ucb** 目录) 都可被访问。**/etc/tftpdaccess.ctl** 文件中的项必须为绝对路径名。

**/etc/tftpdaccess.ctl** 文件应为 root 用户只写而所有 groups 和 others 可读 (即为有许可权 644 的 root 用户所拥有) 可读。用户 **nobody** 必须能够读取 **/etc/tftpdaccess.ctl** 文件。否则, **tftpd** 守护程序就不能识别文件是否存在并允许访问整个系统。有关更多信息, 请参阅样本 **tftpdaccess.ctl** 文件, 它驻留在 **/usr/samples/tcpip** 目录。

搜索算法假设在 **tftp** 命令中所用的本地路径名是绝对路径名。该算法搜索 **/etc/tftpdaccess.ctl** 文件以查找 **allow:/**。它使用每一个部分路径名来重复搜索所允许的路径名, 部分路径名通过从文件路径名添加下一个组件构造而成。匹配的最长路径名也是被允许的一种。然后对于拒绝路径名进行同样操作, 从匹配最长允许路径名开始。

例如, 如果文件路径名为 **/a/b/c** 而 **/etc/tftpdaccess.ctl** 文件包含 **allow:/a/b** 和 **deny:/a**, 将会产生一个允许匹配 (**/a/b**) 并且不产生以 **/a/b** 开始的拒绝匹配, 并允许访问。

如果 `/etc/tftpaccess.ctl` 文件包含 `allow:/a` 和 `deny:/a/b`, 将会产生一个允许匹配 (`/a`) 和一个以 `/a` (`/a/b`) 开始的拒绝匹配, 并拒绝访问。如果 `/etc/tftpaccess.ctl` 文件既包含 `allow:/a/b` 也包含 `deny:/a/b`, 将会拒绝访问, 因为允许的路径名被首先搜索。

## 使用“系统资源控制器”操作 `tftpd` 守护程序

`tftpd` 守护程序是 `inetd` 守护程序的子服务器, 这是系统资源控制器 (SRC) 的子系统。`tftpd` 守护程序是 `tcpip` SRC 子系统组的成员。当没有在 `/etc/inetd.conf` 文件中注释掉时, 此守护程序被启用并可使用以下 SRC 命令操作:

|                 |                    |
|-----------------|--------------------|
| <b>startsrc</b> | 启动子系统、子系统组或子服务器。   |
| <b>stopsrc</b>  | 停止子系统、子系统组或子服务器。   |
| <b>lssrc</b>    | 获取子系统、子系统组或服务器的状态。 |

## 标志

|                            |                                                                                                                                                                                                                                      |
|----------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-c</b>                  | 指定每个进程的并行线程的最大数目, 初始线程除外。                                                                                                                                                                                                            |
| <b>-d</b> <i>Directory</i> | 指定缺省目的地目录。指定的 <i>Directory</i> 将用作主目录, 仅用来存储文件。仅当未指定全路径名时, 此缺省目录才会被使用。用于检索文件的缺省目录仍为 <code>/tftpboot</code> 。                                                                                                                         |
| <b>-i</b>                  | 记录有错误消息的调用机器的 IP 地址。                                                                                                                                                                                                                 |
| <b>-n</b>                  | 允许远程用户在本机上创建文件。如果此标志未指定, 远程用户只被允许读取那些对其他用户有读许可权的文件。                                                                                                                                                                                  |
| <b>-p</b>                  | 指定输入请求的端口号。                                                                                                                                                                                                                          |
| <b>-r</b> <i>Option</i>    | 指定要禁用的 <code>tftp</code> 选项协商。可以使用多个 <b>-r</b> 标志。例如, <code>/etc/inetd.conf</code> 文件中的以下行禁用了 <code>tsize</code> 和 <code>blksize</code> 的选项协商:<br><pre>tftp dgram udp6 SRC nobody /usr/sbin/tftpd tftpd -n -r tsize -r blksize</pre> |
| <b>-s</b>                  | 打开套接字级别的调试。                                                                                                                                                                                                                          |
| <b>-t</b>                  | 为数据报指定超时值。                                                                                                                                                                                                                           |
| <b>-v</b>                  | 当所有文件被 <code>tftpd</code> 守护程序成功传输时, 记录指示性消息。此记录跟踪是谁使用 <code>tftpd</code> 守护程序在系统之间来回远程传输文件。                                                                                                                                         |
| <b>-x</b>                  | 指定等待数据报的最大超时。                                                                                                                                                                                                                        |
| <b>-z</b>                  | 指定用于传输的最大允许段大小。                                                                                                                                                                                                                      |

## 示例

注: `tftpd` 守护程序的参数可以通过使用 SMIT 或编辑 `/etc/inetd.conf` 文件来指定。

1. 要启动 `tftpd` 守护程序, 请输入以下命令:

```
startsrc -t tftp
```

此命令启动 `tftpd` 子服务程序。

2. 要正常停止 `tftpd` 守护程序, 请输入以下命令:

```
stopsrc -t tftp
```

该命令允许启动所有暂挂的连接并完成现有的连接, 但会阻止启动新的连接。

3. 要强制停止 `tftpd` 守护程序和所有 `tftpd` 连接, 请输入以下命令:

```
stopsrc -t -f tftp
```

该命令会立刻终止所有暂挂的连接和现有的连接。

4. 要显示 **tftpd** 守护程序的简短状态报告，请输入以下命令：

```
lssrc -t tftp
```

该命令返回守护程序名、进程标识以及状态（活动的或不活动的）。

## 相关信息

**kill** 命令、**lssrc** 命令、**refresh** 命令、**startsrc** 命令、**stopsrc** 命令、**tftp** 命令。

**inetd** 守护程序。

**/etc/inetd.conf** 文件格式。

《网络与通信管理》中的『次要接文件传输协议（TFTP）』。

《网络与通信管理》中的『TCP/IP 守护程序』。

---

## tic 命令

### 用途

将术语信息描述文件从源格式转换成编译格式。

### 语法

```
tic [ -v [Number] ] [-c] FileName
```

### 描述

**tic** 命令将术语信息文件从源格式转换成编译格式。**tic** 命令将结果放在 **/usr/share/lib/terminfo** 目录下。如果设置了 **TERMINFO** 环境变量，结果放在该环境中而不是放在 **/usr/share/lib/terminfo** 目录下。

**tic** 命令编译 *FileName* 中的所有术语信息描述。当 **tic** 命令查找到一个 `use= entry-name` 字段，它首先搜索当前文件，如果没有找到项 `-name`，就从 **/usr/share/lib/terminfo** 目录下的二进制文件获取该项。如果设置了 **TERMINFO**，在搜索 **/usr/share/lib/terminfo** 之前先搜索术语信息目录。

全部编译项不能超过 4096 字节，名称字段不能超过 128 字节。

### 标志

**-v[Number]** 写有关 **tic** 命令进展的跟踪信息。*Number* 是一个 1 到 10（包括 1 和 10）之间的整数，提高了详细程度的级别。如果 *Number* 被省略，缺省级别为 1。信息输出的量随 *Number* 的增大而增加。

**-c** 仅检查 *FileName* 有无错误。`use=entry-name` 中的错误不会被检测到。

### 文件

**/usr/share/lib/termi**

**nfo/?/\*** 包含编译的终端能力数据库。

## 相关信息

**terminfo** 文件格式。

《AIX 5L V5.3 通用编程概念: 编写并调试程序》中的『Curses 编程概述』。

---

## time 命令

### 用途

显示一条命令的执行时间。

### 语法

**time** [ **-p** ] *Command* [ *Argument ...* ]

### 描述

**time** 命令以秒为单位将一条命令执行期间所用的时间、系统时间和 **time** 命令的执行时间显示在标准错误中。

注: 休眠时间不能算入系统时间, 也不能算入用户时间。

**time** 命令还内置于 C shell (**cs**) 和 Korn shell (**ks**), 但具有不同的格式。要在 **cs** 与 **ks** shell 中运行 **time** 命令, 请输入:

```
/usr/bin/time
```

### 标志

**-p** 将定时输出写到标准错误。秒数被表达为基数字符后至少跟一位的浮点数。

此标志的标准格式如下:

```
"real %f\nuser %f\nsys %f\n", <real seconds>, <user seconds>, <system seconds>
```

### 退出状态

如果使用 *command* 参数, **time** 命令的退出状态就是所指定命令的退出状态。否则, **time** 命令会用以下值之一退出:

- 1-125** 表示 **time** 命令中发生错误。
- 126** 表示由 *Command* 参数指定的命令已找到, 但不能被调用。
- 127** 表示由 *Command* 参数指定的命令找不到。

### 示例

1. 要测量运行一个程序所需的时间, 请输入:

```
/usr/bin/time -p a.out
```

此命令运行程序 **a.out**, 并以由 **-p** 标志指定的格式将实际时间、用户时间和系统时间写入标准错误; 例如:

```
实际时间    10.5  
用户时间    0.3  
系统时间    3.6
```

2. 要在文件中保存 **time** 命令信息的记录，请输入：

```
/usr/bin/time a.out 2> a.time
```

## 文件

**/usr/bin/time** 指定 **time** 命令的路径。

## 相关信息

**timex** 命令。

《操作系统与设备管理》中的『设置记帐子系统』。

《操作系统与设备管理》中的『记帐命令』。

《操作系统与设备管理》中的『系统记帐』。

《性能管理》中的『使用 **time** 命令来测量 CPU 的使用』。

---

## timed 守护程序

### 用途

调用时间服务器守护程序。

### 语法

```
/usr/sbin/timed [ -c ] [ -M ] [ -t ] [ [ -n Network ] ... | [ -i Network ] ... ]
```

注：使用 **rc.tcpip** 文件来启动带有每个初始程序装入的守护程序。可在命令行指定 **times** 守护程序。也可以从命令行使用 **SRC** 命令来控制 **timed** 守护程序。

### 描述

**timed** 守护程序使一台机器的时钟与局域网中也运行 **timed** 守护程序的其他机器的时钟同步。**timed** 守护程序使一些机器的时钟变慢，而使另外一些机器的时钟加快来创建平均网络时间。

当 **timed** 守护程序不带 **-M** 标志启动时，机器找到最近的主时间服务器并请求网络时间。然后机器使用 **date** 命令将机器的时钟设置为该网络时间。机器定期接受主时间服务器发送的同步消息，并调用 **adjtime** 子例序来执行所需的机器时钟的校正。

当 **timed** 守护程序带 **-M** 标志启动时，机器轮询其局域网中的每台机器来确定哪些网络有主时间服务器。机器成为没有主时间服务器的网络上的主时间服务器。机器成为已有主时间服务器的网络上的一台副时间服务器。当 **timed** 守护程序带 **-M** 标志启动时，**timed** 守护程序会创建 **/var/adm/timed.masterlog** 文件。**/var/adm/timed.masterlog** 文件包含一个在本地机器时钟和网络其他机器时钟之间的  $\delta$  记录，而本地机器是该网络的主时间服务器。**/var/adm/timed.masterlog** 文件大约每 4 分钟更新一次，但从不清除。可能需要清除此文件以保存磁盘空间。如果该机器仅为其网络上的副时间服务器，**/var/adm/timed.masterlog** 文件保持为空。要清除 **/var/adm/timed.masterlog** 文件，请输入：

```
cat /dev/null > /var/adm/timed.masterlog
```

如果网络上的主时间服务器停止工作，就要从该网络上的副时间服务器中选取一个新的主时间服务器。**timedc** 命令可让您选择哪一个副时间服务器成为主时间服务器。

**timed** 守护程序可用系统资源控制器（SRC）、系统管理界面程序（SMIT）或命令行控制。缺省情况下，不启动 **timed** 守护程序。使用 **rc.tcpip** 文件来启动带有每个初始程序装入的 **timed** 守护程序。

## 使用系统资源控制器来操作 **timed** 守护程序

**timed** 守护程序是一个由 SRC 控制的子系统。**timed** 守护程序是 SRC **tcpip** 系统组的成员。使用以下 SRC 命令来操作 **timed** 守护程序：

|                 |                                                                          |
|-----------------|--------------------------------------------------------------------------|
| <b>startsrc</b> | 启动子系统、子系统组或子服务器。                                                         |
| <b>stopsrc</b>  | 停止子系统、子系统组或子服务器。                                                         |
| <b>lssrc</b>    | 获取子系统、子系统组或子服务器的短状态。对于 <b>timed</b> 守护程序，不支持通常在 <b>lssrc</b> 中查找到的长状态选项。 |

## 标志

|                   |                                                                                                                                                                                                                              |
|-------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-c</b>         | 指定当计算平均网络时间时主 <b>timed</b> 守护程序应忽略从其他从 <b>timed</b> 守护程序获得的时间值。此标志将网络时间更改为与主 <b>timed</b> 守护程序上的系统时钟一样的时间。                                                                                                                   |
| <b>-i Network</b> | 指定一个要从时钟同步中排除的网络。 <i>Network</i> 变量可以是网络地址或网络名。如果为 <i>Network</i> 变量指定一个网络名，此网络名必须在 <b>/etc/networks</b> 文件中进行定义。使用每个 <b>-i</b> 标志来指定网络地址或网络名。不得将此标志与 <b>-n</b> 标志一起使用。                                                      |
| <b>-M</b>         | 指定该机器为其局域网上的主时间服务器或副时间服务器。如果网络上的主时间服务器当前不可用，该机器就成为该网络的主时间服务器。如果网络已存在一个主时间服务器，则该机器成为该网络上的副时间服务器。但是，如果当前主时间服务器不起作用，该机器能成为主时间服务器。当 <b>timed</b> 守护程序带 <b>-M</b> 标志启动时， <b>timed</b> 守护程序会创建 <b>/var/adm/timed.masterlog</b> 文件。 |
| <b>-n Network</b> | 指定要包含在时钟同步中的网络。 <i>Network</i> 变量可以是网络地址或网络名。如果为 <i>Network</i> 变量指定一个网络名，此网络名必须在 <b>/etc/networks</b> 文件中进行定义。使用每个 <b>-n</b> 标志来指定网络地址或网络名。不得将此标志与 <b>-i</b> 标志一起使用。                                                        |
| <b>-t</b>         | 允许 <b>timed</b> 守护程序跟踪它接收到的消息并将它们存储在 <b>/var/adm/timed.log</b> 文件中。还可使用 <b>timedc</b> 命令来激活跟踪功能。                                                                                                                             |

## 示例

1. 要使用 SRC 控制来启动 **timed** 守护程序，请输入：

```
startsrc -s timed
```

此命令启动守护程序。可在 **rc.tcpip** 文件中或命令行上使用此命令。**-s** 标志指定接下来子系统将要被启动。

2. 要使用 SRC 控制来正常停止 **timed** 守护程序，请输入：

```
stopsrc -s timed
```

该命令停止守护程序。**-s** 标志指定接下来子系统将会被停止。

3. 为从 **timed** 守护程序获取短状态报告，请输入：

```
lssrc -s timed
```

该命令返回守护程序的名称、进程标识和守护程序的状态（活动或不活动）。

4. 要使用 SRC 控制来启动 **timed** 守护程序作为主时间服务器或副时间服务器并要从时钟同步中排除网络 **net1** 和 **net2**，请输入：

```
startsrc -s timed -a "-M -i net1 -i net2"
```

此命令启动守护程序。机器成为其网络的主时间服务器或副时间服务器。网络 `net1` 和 `net2` 从时钟同步中被排除。`-s` 标志指定接下来子系统将要被启动。`-a` 标志指定 `timed` 守护程序应带后面的一些标志启动。标志必须包含在引号中。

5. 要启动 `timed` 守护程序、激活跟踪功能并将 `net1` 与 `net2` 包含在时钟同步中，请输入：

```
timed -t -n net1 -n net2
```

此命令启动守护程序。跟踪功能被激活，并且 `net1` 和 `net2` 都被包含进时钟同步中。

## 文件

`/var/adm/timed.log`

包含跟踪 `timed` 守护程序的消息。当 `timed` 守护程序带 `-t` 标志启动时或当使用 `timedc` 命令启用跟踪功能时，创建此文件。

`/etc/rc.tcpip`

包含在系统启动时要执行的 `SRC` 命令。

`/var/adm/timed.masterlog`

包含一个在主时间服务器时钟与网络上其他机器的时钟之间的  $\delta$  记录。当 `timed` 守护程序带 `-M` 标志启动时创建此文件。但是，此文件仅包含该机器在其中为主时间服务器的那些网络的信息。

## 相关信息

`date` 命令、`timedc` 命令。

`adjtime` 子例程、`gettimeofday` 子例程。

`networks` 文件格式。

《网络与通信管理》中的『TCP/IP 守护程序』。

---

## timedc 命令

### 用途

返回有关 `timed` 守护程序的信息。

### 语法

```
timedc [ Subcommand [ Parameter ... ] ]
```

### 描述

`timedc` 命令控制 `timed` 守护程序的操作。`timedc` 命令执行以下操作：

- 测定网络上不同机器的时钟之间的差异。
- 查找主时间服务器的位置。
- 启用或禁用 `timed` 守护程序接收到的消息的跟踪功能。
- 调试。

如果没有任何变量，`timedc` 假定采用交互方式并提示从标准输入中输入子命令。如果提供了变量，`timedc` 命令将第一个变量解释为一条子命令，而其余变量解释为这条子命令的参数。可以重定向标准输入以使 `timedc` 命令从文件中读取子命令。



## 变量

**timedc** 命令识别以下子命令:

**? [ Parameter ... ]**

显示在参数列表中指定的每个变量的简短描述。**?** 子命令仅以交互方式工作。如果没有给出变量, **?** 子命令就会显示一个可被 **timedc** 命令识别的子命令列表。

**clockdiff Host ...**

计算主机的时钟与作为变量给定的机器的时钟之间的时差。

**election Host ...**

请求指定主机上的 **timed** 守护程序复位其选择定时器, 并确保 **timed** 主控服务器可用。最多可指定 4 台主机。如果主 **timed** 服务器不再可用, 则指定主机上的 **timed** 守护程序将请求成为新的 **timed** 主控服务器。

**help [ Parameter ... ]**

指定的主机必须使用 **-M** 标志以次级方式运行 **timed** 守护程序。显示每条在参数列表中指定的子命令的简短描述。如果没有给出变量, **help** 子命令显示一个可被 **timedc** 命令识别的子命令列表。

**msite**

查找主站点位置。

**quit**

退出 **timedc** 命令。

**trace { on | off }**

启用或禁用跟踪进入 **timed** 守护程序的消息的功能。消息被保存在 **/var/adm/timed.log** 文件中。

可以使用别的命令来测试和调试 **timed** 守护程序。使用 **help** 命令来查找这些命令。

**timedc** 命令会出现以下错误消息:

模糊命令

缩写匹配多个命令。

无效命令

未发现匹配。

特权命令

命令仅可由 root 用户执行。

## 示例

1. 要显示本地主机 **sahara** 与远程主机 **sandy** 之间的时差, 请输入:

```
timedc clockdiff sandy
```

输出为:

在 **sandy.austin.century.com** 上的时间比 **sahara.austin.century.com** 上的时间提前 37904247 毫秒。

2. 要显示 **timed** 守护程序的客户机位置, 请输入:

```
timedc msite
```

输出为:

客户机 **timed** 守护程序运行在 **bupu.austin.century.com** 上

## 相关信息

**date** 命令。

**timed** 守护程序。

**adjtime** 子例程。

《网络与通信管理》中的『TCP/IP 守护程序』。

---

## timex 命令

### 用途

以秒为单位，报告一条命令的所用时间、用户时间和系统执行时间。

### 语法

**timex** [ **-o** ] [ **-p** ] [ **-s** ] *Command*

### 描述

**timex** 命令以秒为单位报告一条命令的所用时间、用户时间和系统执行时间。使用指定的标志，**timex** 命令列出或汇总一条命令及其所有子命令的进程记帐数据。*Command* 是系统中任一可执行文件的名称。也可报告在执行间隔内的全部系统活动。输出被写到标准错误。系统使用 **/var/adm/pacct** 文件来选择与该命令有关的进程记录，并包含有相同用户标识、工作站标识和执行时间窗口的后台进程。

### 标志

- o** 报告由一条命令及其所有子命令进行读或写的块数和传输的所有字符。
- p** 列出一条命令及其所有子命令的进程记帐记录。报告读或写的块数和传输的字符数。**-p** 标志采用在 **acctcom** 命令中定义的 **f**、**h**、**k**、**m**、**r** 和 **t** 参数来修改其他数据项。
  - f** 在输出中显示 **fork/ exec** 标志和系统退出状态栏。
  - h** 不表示平均内存大小，而是显示进程 (**hogfactor**) 所消耗的全部可用 CPU 时间的百分比。
  - k** 不表示平均内存大小，而是显示全部 **kcore** 分钟 (以运行时间每分钟使用多少千字节段来评估内存)。
  - m** 显示平均主存储器大小。这是缺省值。**-h** 标志或 **-k** 标志可关闭 **-m** 标志。
  - r** 显示 CPU 分数。
  - t** 显示个别系统和用户 CPU 时间。
- s** 报告命令执行期间的所有系统活动。也报告所有在 **sar** 命令中列出的数据项。

注：必须打开记帐以使用 **-o** 或 **-p** 标志。

### 示例

1. 要报告由 **ls** 命令读取的块总数和传输的所有字符，请输入：

```
timex -o ls
```

2. 要列出 **ps** 命令的进程记帐记录，请输入：

```
timex -p ps -fe
```

3. 要报告 **ls** 命令执行时的所有系统活动，请输入：

```
timex -s ls
```

### 文件

**/var/adm/pacct** 用来选择与命令有关的记录。

## 相关信息

**acctcom** 命令、**sar** 命令、**time** 命令。

《操作系统与设备管理》中的『设置记帐子系统』描述建立记帐系统必须采取的步骤。

请参阅《操作系统与设备管理》中的『记帐命令』，以获取可自动运行或可从键盘输入的记帐命令列表。

有关“记帐系统”、每日和每月报告的准备以及记帐文件的更多信息，请参阅《操作系统与设备管理》中的『系统记帐』。

《性能管理》中的『监视和调节命令和子例程』。

---

## tip 命令

### 用途

连接到远程系统。

### 语法

```
tip [ -v ] [ - BaudRate ] { SystemName | PhoneNumber }
```

### 描述

**tip** 命令连接到远程系统，并允许您像直接登录一样在远程系统上工作。

需要 *SystemName* 参数或 *PhoneNumber* 参数。*SystemName* 参数指定了要连接的远程系统名称。远程系统必须在 **/etc/remote** 文件或者在由 **REMOTE** 环境变量指定的文件中进行定义。*PhoneNumber* 参数指定了通过调制解调器连接的拨号号码。

当 **tip** 命令带 *SystemName* 参数被调用时，它搜索 **remote** 文件以查找以该系统名开头的项。当命令带 *PhoneNumber* 参数被调用时，它搜索 **remote** 文件以查找具有格式 **tipBaudRate** 的项，其中 *BaudRate* 是连接的波特率。如果 **-BaudRate** 标志未使用，**tip** 命令查找 **tip1200** 项，因为 1200 是缺省波特率。

**tip** 命令的操作可以用标志、转义符号和变量控制。**tip** 命令读取 **/etc/remote** 文件来查找如何连接远程系统并发现与该系统通信时使用的转义发送顺序。此外，此命令还可以检查 **/etc/phones** 文件以找出连接远程系统的电话号码。

**tip** 用户可以按照 **/usr/lib/remote-file** 文件的格式来创建个人远程文件，然后用 **REMOTE** 环境变量指定要使用的文件。用户也可以按照 **/usr/lib/phones-file** 文件的格式来创建个人电话文件，然后用 **PHONES** 环境变量指定要使用的文件。但是缺省情况下，**tip** 命令不会读取 **/usr/lib/remote-file** 文件或 **/usr/lib/phones-file** 文件。**tip** 命令使用的缺省文件是 **/etc/remote** 文件和 **/etc/phones** 文件。

**tip** 用户可创建一个 **\$HOME/.tiprc** 文件来指定 **tip** 变量的初始设置。此外，在远程文件、电话文件及 **.tiprc** 文件中所做的设置可通过使用转义符号在 **tip** 命令运行时进行重设。例如：转义符号也可用来启动和停止文件传输或中断与远程系统的连接。

**tip** 命令使用 **/etc/locks** 目录下的锁定文件来锁定设备不会多次访问和防止多个用户登录到同一系统上。

当 **tip** 命令提示响应时，编辑您使用标准键输入的行。输入 ~.（代字号，句点）对提示作出反应，或者按“中断”键来中止 **tip** 对话并返回到远程系统。

可以使用 **tip** 命令在远程系统之间来回传输文件。也可用 **tip** 命令转义符号来启动和停止文件传输。几个 **tip** 命令变量合作控制文件传输。

文件传输通常使用 **tandem** 方式来控制数据流。如果远程系统不支持 **tandem** 方式，将 *echocheck* 变量设置为 **on**，以使得 **tip** 命令在传输每个字符后与远程系统同步。当使用 **~<** 和 **~>** 转义符号传输文件时，在写文件时使用 **eofread** 和 *eofwrite* 变量来指定文件结束，并在读文件时识别文件结束。

如果 *verbose* 变量设置为 **on**，**tip** 命令执行以下操作：

- 在文件传输期间写被传输的行数的运行计数值。
- 在拨号时写指示其操作的消息。

可使用脚本编制来记录与 **tip** 命令的对话。使用 *script* 变量来启动脚本编制。

注：

1. 只有拥有 **root** 用户权限的用户才可以更改 *dialtimeout* 变量。
2. 虽然任何用户都能在命令行指定一个主机，但仅有 **root** 用户能够在 **tip** 命令启动之后更改 *host* 变量设置。但是，这并不会更改 **tip** 命令当前所连接的系统。

## 标志

**-v** 当从 **.tiprc** 文件读取变量时显示变量设置。  
**-BaudRate** 覆盖缺省波特率（1200 波特）。

## 转义符号

使用转义符号可以指示 **tip** 命令终止、自远程系统注销和传输文件。行开头的转义字符表示一个转义符号。缺省转义字符为 **~**（代字号）。字符可使用 *escape* 变量来更改。所有其他输入的字符被直接发送到远程系统。**tip** 命令可识别以下转义符号：

|                                |                                                                                     |
|--------------------------------|-------------------------------------------------------------------------------------|
| <b>~^D~</b>                    | 终止连接并退出。用户可能仍要登录到远程系统；如果是这样，可发出另一个 <b>tip</b> 命令来重新连接到该远程系统。                        |
| <b>~c</b> [ <i>Directory</i> ] | 在本地系统上，转到由 <i>Directory</i> 变量指定的目录。如果未包含 <i>Directory</i> 变量， <b>tip</b> 命令更改到主目录。 |
| <b>~!</b>                      | 转到本地系统上的 <i>shell</i> 。当从 <i>shell</i> 退出时，返回到 <b>tip</b> 命令。                       |
| <b>~&gt;</b>                   | 将文件从本地系统复制到远程系统。 <b>tip</b> 命令提示输入本地文件的名称。                                          |
| <b>~&lt;</b>                   | 将文件从远程系统复制到本地系统。 <b>tip</b> 命令提示输入远程文件的名称。                                          |

**tip** 文件下载将只下载文件，直到遇到列在 **eofread** 命令变量中的 EOF 字符之一。如果没有遇到这些字符之一，那么文件复制将不能成功。

当下载一个带有 **~<** 符号的文件时，会提示用户输入一个本地文件名。用户可输入任一有效的可写文件名。当提示输入远程命令时，用户要在被读文件末尾附加一个 EOF 字符。

此符号可像以下示例中显示的那样使用：

```
List command for remote system? echo "\04" | cat /etc/passwd
```

此示例假设字符 **0x4** 出现在 **tip** *eofread* 变量。确保该字符存在于变量中的最佳方法是在用户的 **.tiprc** 文件（该文件应驻留在用户的主目录中）中指定它。

要完成该任务，可发出以下命令：

```
echo"eofread=\04" >> ~/.tiprc
```

- ~p** *Source* [*Dest*]            使用 **cat** 命令将 *Source* 文件复制到 *Dest* 文件中，以此来将 *Source* 文件发送（放置）到远程 UNIX 主机系统中。如果 *Dest* 文件名没有指定，**cat** 命令使用 *Source* 文件的名称。如果 *Dest* 文件在远程主机上存在，就会被 *Source* 文件替换。此符号是特定于 UNIX 版本的 ~> 符号。
- ~t** *Source* [*Dest*]            使用 **cat** 命令将 *Source* 文件复制到本地系统的 *Dest* 文件中，以此来将 *Source* 文件从远程 UNIX 主机系统传输（带）到本地系统中。如果 *Dest* 文件名没有被指定，**cat** 命令使用 *Source* 文件的名称。如果 *Dest* 文件在本地系统上存在，就会被 *Source* 文件替换。此符号是特定于 UNIX 版本的 ~< 符号。
- ~l**                                将一个远程命令的输出输送到本地进程。发送到本地系统的命令字符串由 shell 处理。

仅当来自远程管道的数据由列在 **eofread tip** 命令变量中的 EOF 字符之一终止时，远程管道才能成功。如果没有遇到这些字符之一，那么输出管道将不能成功。

当带 ~l 符号管道传送远程输出时，将提示用户输入一个本地命令名。用户可输入任一有效的命令名。当提示输入远程命令时，用户要在被读文件末尾附加一个 EOF 字符。

此符号可像以下示例中显示的那样使用：

```
Local command? cat
List command for remote system? echo
"asdfasdfasdf\04"
```

此示例假设字符 0x4 出现在 **tip eofread** 变量中。确保该字符存在于变量中的最佳方法是在用户的 **.tiprc** 文件（该文件应驻留在用户的主目录中）中指定它。

要完成该任务，可发出以下命令：

```
echo"eofread=\04" >> ~/.tiprc
```

- ~\$**                                将本地进程的输出输送到远程系统。发送到远程系统的命令字符串由 shell 处理。
- ~#**                                发送一个 **BREAK** 符号到远程系统。
- ~s** { *Variable=Value* | [!]*BoolVariable* | **all** | *Variable?* }    设置或查询 **tip** 命令变量。  
  
要更改一个非布尔变量的值，请输入此变量名或缩写，后面跟一个 =（等号）再跟新值。例如，输入 ~s rc=^U 来更改所用字符以打开或关闭大写转换（*raisechar* 变量）。  
  
要更改一个布尔变量的值，请输入此变量名或缩写。要将该变量重新设置为其缺省值，请在名称前输入！（感叹号）。例如，输入 ~s !ec 来将 *echocheck* 变量重新设置为其缺省值。  
  
要显示用户可读的所有变量，将 **all** 指定为 ~s 符号的一个参数。也可通过将 ?（问号）附加到变量名来请求显示一个特殊变量。例如，输入命令 ~s eol? 来显示当前行结束字符串（*eol* 变量）。  
  
**~^Z**                              停止 **tip** 命令。~^Z 符号仅可与作业控制一起使用。  
**~^Y**                              停止 **tip** 命令的本地部分。显示远程系统输出的远程部分继续运行。~^Y 符号仅可与作业控制一起使用。  
**~?**                                显示转义符号列表。

## 变量

**tip** 命令使用控制其操作的变量。这些变量可以是数字、字符串、字符或布尔值。其中一些变量可由能运行 **tip** 命令的任何用户进行更改。但是，以下变量只能由拥有 **root** 用户权限的用户进行更改：*baudrate* 变量和 *dialtimeout* 变量。

**\$HOME/.tiprc** 文件中的变量可以在运行时进行初始化。另外，当已通过使用 **~s** 转义符号运行 **tip** 命令时，可以显示和设置变量。

变量可以是数字、字符串、字符或布尔值。要设置一个非布尔变量，请输入此变量名或缩写，后跟 **=**（等号）和值。例如，输入 **~s host=zeus** 或 **~s ho=zeus** 来将 **host** 名称更改为 **zeus**。在 **.tiprc** 文件中，请输入 **host=zeus** 或 **ho=zeus**。

要更改一个布尔变量值，请输入变量名或缩写作为 **~s** 符号的一个参数或在 **.tiprc** 文件的一行上输入变量名或缩写。要将该变量重新设置为其缺省值，请在名称前输入 **!**（感叹号）。例如，在运行 **tip** 命令时，请输入 **~s !echocheck**，以将 *echocheck* 变量重新设置为其缺省值。

以下为公共变量、其类型、缩写和缺省值。

| 变量（缩写）                             | 类型  | 缺省值                                                                                                   |
|------------------------------------|-----|-------------------------------------------------------------------------------------------------------|
| <i>beautify</i> ( <i>be</i> )      | 布尔  | 当编写会话时，指示 <b>tip</b> 命令废弃那些不可显示的字符。不得废弃 <i>exceptions</i> 变量指定的字符。缺省设置为 <b>on</b> 。                   |
| <i>baudrate</i> ( <i>ba</i> )      | 数字  | 反映了连接的波特率。更改此变量值将不会更改所连接的 <b>tty</b> 设备的当前波特率设置。                                                      |
| <i>dialtimeout</i> ( <i>dial</i> ) | 数字  | 指定拨号时 <b>tip</b> 命令等待连接的时间（以秒计）。缺省值为 <b>60</b> 秒。 <b>dialtimeout</b> 设置仅可由拥有 <b>root</b> 用户权限的用户进行更改。 |
| <i>echocheck</i> ( <i>ec</i> )     | 布尔  | 指示 <b>tip</b> 命令在传输下一字符前等待最后一个被发送字符的回应，以便在文件传输过程中与远程系统同步。缺省设置为 <b>off</b> 。                           |
| <i>eofread</i> ( <i>eofr</i> )     | 字符串 | 指定在远程到本地 ( <b>~&lt;</b> 或 <b>~t</b> ) 文件传输期间表示传输结束符的字符集。                                              |
| <i>eofwrite</i> ( <i>eofw</i> )    | 字符串 | 在本地到远程 ( <b>~&gt;</b> 或 <b>~p</b> ) 文件传输期间，指定要发送的表示传输结束的字符串。                                          |
| <i>eol</i> (无)                     | 字符串 | 指定表示行结束的字符串。 <b>tip</b> 命令只有跟在行结束字符串之后时才能识别转义符号。                                                      |
| <i>escape</i> ( <i>es</i> )        | 字符  | 指定转义符号的字符前缀。缺省值为 <b>~</b> （代字号）。                                                                      |
| <i>etimeout</i> ( <i>et</i> )      | 数字  | 当 <i>echocheck</i> 变量被设置为 <b>on</b> 时指定等待响应的的时间。如果在指定时间内没有接收到回应，文件传输中止。缺省时间为 <b>28</b> 秒。             |
| <i>exceptions</i> ( <i>ex</i> )    | 字符串 | 指定即使当 <b>beautify</b> 开关设置为 <b>on</b> 时也不能废弃的字符集。缺省字符串为 <b>\t\n\f\b</b> 。                             |
| <i>force</i> ( <i>fo</i> )         | 字符  | 指定在二进制文件传输期间用来强制文字数据传输的字符。缺省字符为 <b>^P</b> 。文字数据传输一直被关闭，直到用户输入 <i>force</i> 变量指定的字符为止。                 |
| <i>framesize</i> ( <i>fr</i> )     | 数字  | 指定从远程系统接收文件时文件系统写之间用于缓冲的字节数量。                                                                         |
| <i>host</i> ( <i>ho</i> )          | 字符串 | 指定调用 <b>tip</b> 命令时您连接到的远程系统的名称。此变量不可更改。                                                              |
| <i>halfduplex</i> ( <i>hdx</i> )   | 布尔  | 切换半双工方式。缺省设置为 <b>off</b> 。                                                                            |
| <i>localecho</i> ( <i>le</i> )     | 布尔  | 切换本地回应方式。缺省设置为 <b>off</b> 。                                                                           |

| 变量 (缩写)                         | 类型  | 缺省值                                                                                                                                                                                                                                                                     |
|---------------------------------|-----|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>log</i> (无)                  | 字符串 | 定义用来记录使用 <b>tip</b> 命令的拨出的文件。缺省文件为 <b>/var/spool/uucp/.Admin/aculog</b> 。此记录文件仅可由拥有 root 用户权限的用户进行修改。                                                                                                                                                                   |
| <i>parity</i> ( <i>par</i> )    | 字符串 | 定义文件传输的奇偶性。缺省值为以下字符串：<br>no parity, 8 data bits                                                                                                                                                                                                                         |
| <i>phones</i> (无)               | 字符串 | 指定用户的电话文件的名称。此文件可有任何有效的文件名，但必须按照 <b>/usr/lib/phones-file</b> 文件的格式进行设置。缺省文件为 <b>/etc/phones</b> 。如果一个文件由 <b>PHONES</b> 环境变量指定，它被用来替换（而不是加到） <b>/etc/phones</b> 文件。                                                                                                      |
| <i>prompt</i> ( <i>pr</i> )     | 字符  | 指定远程主机上表示行结束的字符。此字符用于数据传输期间的同步。 <b>tip</b> 命令在文件传输期间，根据接收到 <b>prompt</b> 字符的次数，对被传输的行进行计数。缺省字符为 \n。                                                                                                                                                                     |
| <i>raise</i> ( <i>ra</i> )      | 布尔  | 当设置为 on 时，指示 <b>tip</b> 命令在传输字符到远程系统前将所有小写字母转换为大写字母。缺省设置为 off。                                                                                                                                                                                                          |
| <i>raisechar</i> ( <i>rc</i> )  | 字符  | 指定用于切换大写转换的字符。缺省字符为 ^A。                                                                                                                                                                                                                                                 |
| <i>rawftp</i> ( <i>raw</i> )    | 布尔  | 如果 <i>rawftp</i> 变量设置为 on，数据在文件传输期间通过连接进行传输，而不进行其他处理。即发送文件时，换行不会映射到已执行的换行/回车。                                                                                                                                                                                           |
| <i>record</i> ( <i>rec</i> )    | 字符串 | 指定 <b>tip</b> 命令记录会话脚本的文件的名称。缺省文件为 <b>tip.record</b> 。 <b>tip</b> 命令将文件放在本地系统的用户当前目录中。                                                                                                                                                                                  |
| <i>remote</i> (无)               | 字符串 | 指定用户远程系统定义文件的名称。此文件可有任何有效的文件名，但必须按照 <b>/usr/lib/remote-file</b> 文件的格式进行设置。缺省文件为 <b>/etc/remote</b> 。如果一个文件由 <b>REMOTE</b> 环境变量指定，则它被用来替换（而不是加到） <b>/etc/remote</b> 文件。                                                                                                  |
| <i>script</i> ( <i>sc</i> )     | 布尔  | 当 <b>script</b> 开关设置为 on 时， <b>tip</b> 命令就会将远程系统传输的一切都记录在本地系统的一个文件中。文件名由 <i>record</i> 变量指定。如果 <b>beautify</b> 开关设置为 on，只有可显示的 ASCII 字符（在 040 与 0177 之间的那些字符）会被记录在脚本文件中。 <i>exceptions</i> 变量指定了那些即使 <b>beautify</b> 开关设置为 on 也将被记录的不可显示字符。 <b>script</b> 开关的缺省设置为 off。 |
| <i>tabexpand</i> ( <i>tab</i> ) | 布尔  | 使 <b>tip</b> 命令在文件传输期间将制表符扩展到 8 个空格。缺省设置为 off。                                                                                                                                                                                                                          |
| <i>verbose</i> ( <i>verb</i> )  | 布尔  | 当 <b>verbose</b> 开关设置为 on 时， <b>tip</b> 命令就会显示拨号时的消息、显示文件传输期间传输的当前行号以及显示与连接有关的其他状态信息。缺省设置为 on。                                                                                                                                                                          |
| <i>SHELL</i> (无)                | 字符串 | 指定 shell 的类型，用于 <b>~!</b> 符号。缺省值为 <b>/usr/bin/sh</b> 或从环境获取。                                                                                                                                                                                                            |
| <i>HOME</i> (无)                 | 字符串 | 指定用于 <b>~c</b> 符号的主目录。缺省值从环境获取。                                                                                                                                                                                                                                         |

## 示例

1. 当进行直接连接时要指定波特率，请输入：

```
tip -300 hera
```

这表示当联系远程系统 hera 时，**tip** 命令使用的波特率为 300。

2. 要使用调制解调器连接远程系统，请输入：

```
tip 9,343-2132
```

**tip** 命令在拨一个 9 到达外部线路之后，通过拨电话号码 343-2132 将本地系统连到远程系统。

3. 要直接连接到远程系统并显示变量，请输入：

```
tip -v hera
```

**-v** 标志使得 **tip** 命令从 **\$HOME/.tiprc** 文件读取变量时显示变量值。如果 **.tiprc** 文件包含以下设置：

```
sc
be
rec=/home/jimk/callout
```

则 **-v** 标志的输出如下所示：

```
set script
set beautify
set record=/home/jimk/callout
```

## 文件

|                             |                                                                             |
|-----------------------------|-----------------------------------------------------------------------------|
| <b>/usr/bin/tip</b>         | 包含 <b>tip</b> 命令。                                                           |
| <b>/etc/locks /*</b>        | 包含了防止对设备的多重使用和对系统的多重调用的锁定文件。                                                |
| <b>/etc/remote</b>          | 包含 <b>tip</b> 命令的系统描述。如果设置了 <i>remote</i> 变量或 <b>REMOTE</b> 环境变量，就使用该文件。    |
| <b>/usr/lib/remote-file</b> | 包含样本 <b>remote</b> 文件。如果设置了 <i>remote</i> 变量或 <b>RECORD</b> 环境变量，就使用该文件。    |
| <b>/etc/phones</b>          | 包含 <b>tip</b> 命令的电话号码数据库。如果设置了 <i>phones</i> 变量或 <b>PHONES</b> 环境变量，就使用该文件。 |
| <b>/usr/lib/phones-file</b> | 包含 <b>tip</b> 命令的电话号码数据库。如果设置了 <i>phones</i> 变量或 <b>PHONES</b> 环境变量，就使用该文件。 |
| <b>\$HOME/.tiprc</b>        | 定义 <b>tip</b> 命令变量的初始设置。                                                    |
| <b>tip.record</b>           | 包含 <b>tip</b> 命令脚本。缺省情况下，文件存储在当前目录。用户可使用 <i>record</i> 变量来更改文件名和目录。         |

## 相关信息

**cu** 命令、**uucp** 命令。

《网络与通信管理》中的『使用 **tip** 命令与连接的 UNIX 系统通信』。

---

## tokstat 命令

### 用途

显示令牌环设备驱动程序与设备统计信息。

### 语法

```
tokstat [ -d -r -t ] Device_Name
```



## 描述

**tokstat** 命令显示了由指定的令牌环设备驱动程序收集到的统计信息。除了显示设备驱动程序统计信息之外，用户还可有选择地指定显示特定于设备的统计信息。如果没有指定任何标志，仅显示设备驱动程序统计信息。

当 **netstat** 命令带 **-v** 标志运行时，也可调用此命令。**netstat** 命令并不发出任何 **tokstat** 命令标志。

如果指定一个无效的 *Device\_Name*，**tokstat** 命令就会生成一个错误消息，表示无法连接到该设备。

## 标志

- d** 显示所有设备驱动程序统计信息，包括特定于设备统计信息。
- r** 将所有统计信息重新设置回其初始值。此标志仅可由特权用户发出。
- t** 在一些设备驱动程序中切换调试跟踪。

## 参数

*Device\_Name* 例如，令牌环设备名为 **tok0**。

## 统计信息字段

注：一些适配器可能不支持特定的统计信息。那些不支持的统计信息字段的值总是为 0。

显示在 **tokstat** 命令的输出中的统计信息字段及其描述为：

### 标题字段

|       |                                                                                                          |
|-------|----------------------------------------------------------------------------------------------------------|
| 设备类型  | 显示适配器类型的描述。                                                                                              |
| 硬件 地址 | 显示设备当前所用的令牌环网络地址。                                                                                        |
| 所用时间  | 显示自从统计信息上次被重新设置以来所占用的实际时间周期。当检查到有硬件错误时，设备驱动程序可在错误恢复期间复位部分统计信息。当发生这种情况时，在输出中间将会显示另外一个占用时间，以便反映统计信息间的时间差别。 |

### 传输统计信息字段

|                   |                                            |
|-------------------|--------------------------------------------|
| 数据包               | 设备成功传输的数据包数目。                              |
| 字节数               | 设备成功传输的字节数。                                |
| 中断                | 驱动程序从适配器接收到的传输中断的数目。                       |
| 传输错误              | 此设备上遇到的输出错误的数目。这是一个由于硬件 / 网络出错引起的失败传输的计数器。 |
| 被删除的数据包           | 设备驱动程序接受到的用于传输、但（因为某些原因）没有送达设备的数据包数目。      |
| S/W 传输队列的最大数据包数   | 曾经列入软件传输队列的输出数据包的最大数目。                     |
| S/W 传输队列溢出        | 溢出软件传输队列的输出数据包的数目。                         |
| 当前 S/W+H/W 传输队列长度 | 暂挂在软件传输队列或硬件传输队列中的输出信息包的数目。                |
| 广播数据包             | 传输的无错误的广播数据包数目。                            |
| 多点广播数据包           | 传输的无错误的多点广播数据包数目。                          |
| 超时错误              | 由于适配器报告的超时错误引起的失败传输的数目。                    |
| 当前 SW 传输队列长度      | 当前在软件传输队列的输出数据包数目。                         |

当前 HW 传输队列长度

当前在硬件传输队列的输出数据包数目。

## 接收统计信息字段

|         |                                                |
|---------|------------------------------------------------|
| 数据包     | 设备成功接收的数据包数目。                                  |
| 字节数     | 设备成功接收的字节数。                                    |
| 中断      | 驱动程序从适配器接收到的接收中断的数目。                           |
| 接收错误    | 此设备上遇到的输入错误的数目。这是一个由于硬件 / 网络出错引起的失败接收的计数器。     |
| 被删除的数据包 | 设备驱动程序从此设备接收到的、但（因为某些原因）没有送达网络 demuxer 的数据包数目。 |
| 损坏的数据包  | 设备驱动程序接收到（保存）的损坏的数据包数目。                        |
| 广播数据包   | 接收的无错误的广播数据包数目。                                |
| 多点广播数据包 | 接收的无错误的多点广播数据包数目。                              |
| 接收拥塞错误  | 由于一个没有资源错误而被硬件删除的输入数据包数目。                      |

## 常规统计信息字段

|           |                                                                                                                                                                  |
|-----------|------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 无 mbuf 错误 | mbuf 对设备驱动程序不可用的次数。这通常发生在驱动程序必须获取 mbuf 缓冲区来处理入站数据包的接收操作期间。如果所请求大小的 mbuf 池是空的，数据包就会被废弃。netstat -m 命令可以用来对此进行确认。                                                   |
| 插接瓣线路错误   | 适配器检测到插接瓣数据路径中的断路或短路的次数（例如，电缆没有插好）。                                                                                                                              |
| 异常终止错误    | 适配器传输出现问题的次数。                                                                                                                                                    |
| AC 错误     | 适配器接收到多个 AMP（活动监控程序出现）或 SMP（备用监控程序出现）帧的次数，这些帧可将已识别的地址和已复制的帧所有位设置为零。这表示有邻近通知问题。每个站点可从 AMP 和 SMP 帧中得知并记住哪一个是其“最近的活动上行邻居”（NAUN）。当站点报告有问题时，也会报告哪一个是其 NAUN。这有助于定义故障区。 |
| 突发错误      | 适配器检测到信号极性在必要时没有转换的次数。                                                                                                                                           |
| 帧复制错误     | 适配器检测到有其特定地址的帧被其他适配器复制的次数。                                                                                                                                       |
| 频率错误      | 适配器检测到输入信号频率超过 IEEE 802.5 标准所允许的期望频率的倍数。检查负责环上的主时钟的活动监控程序和频率抖动补偿。                                                                                                |
| 硬错误       | 适配器传输或接收信标 MAC 帧的次数。                                                                                                                                             |
| 内部错误      | 适配器发生内部错误的次数。                                                                                                                                                    |
| 行错误       | 适配器在帧或令牌中检测到无效字符的次数。                                                                                                                                             |
| 帧丢失错误     | 适配器传输帧但未收回帧的次数。                                                                                                                                                  |
| 仅有一个站点错误  | 适配器认为自己是环上唯一的适配器的次数。                                                                                                                                             |
| 令牌错误      | 作为活动监控程序工作的适配器检测到令牌丢失的次数。这也许是因为环重新配置的缘故。如果常发生这种情况，请查看其他软错误是否表示有特殊问题。                                                                                             |
| 接收到除去请求   | 适配器接收到除去环站 MAC 帧请求的次数。                                                                                                                                           |
| 环恢复       | 环被清除并恢复回正常操作状态的次数。                                                                                                                                               |
| 信号丢失错误    | 适配器检测到缺少接收信号的次数。                                                                                                                                                 |
| 软错误       | 适配器检测到软错误（可用 MAC 层协议恢复）的次数。                                                                                                                                      |
| 传输信标错误    | 适配器传输信标帧的次数。                                                                                                                                                     |
| 驱动程序标志    | 当前打开的设备驱动程序内部状态标志。                                                                                                                                               |

## 设备特定统计信息字段

显示的该部分对于适配器的每种类型都不同。可能包含特定于适配器的信息和一些未包含在一般统计信息中的扩展统计信息。有些适配器可能没有任何特定于设备的统计信息。可列在这部分的字段有：

ARI/FCI 错误

ARI/FCI 不匹配也被称为接收器拥塞。如果适配器从环上经过的一个帧获取地址匹配（地址识别指示，ARI），但没有地方来复制该帧（帧复制指示，FCI），就会发生 ARI/FCI 不匹配。当帧经过时，适配器会在帧结尾的 FS 字节打开 ARI 位但不会打开 FCI 位。

DMA 总线错误  
DMA 奇偶错误  
接收超限

换句话说，适配器虽然看到了要接收的帧，但又不能接收，因为接收缓冲器已经耗尽了。两秒钟后适配器会发出一个“报告软错误”MAC 帧表明发生接收器拥塞错误。

接收欠载运行

适配器完成 DMA 传输并检测到有总线错误的次数。  
适配器完成 DMA 传输并检测到有奇偶错误的次数。  
当适配器试图接收帧时适配器接收 FIFO 已满的次数。

发出读日志命令的次数

在检测到帧结束符号之前，适配器发送 FIFO 为空的次数。

适配器错误计数器超限（达到 255）以及设备驱动程序发出读日志命令来读取（和重设）错误计数器的次数。

## 示例

1. 要显示 **tok0** 设备驱动程序统计信息，请输入：

```
tokstat tok0
```

这将产生以下输出：

```

TOKEN-RING STATISTICS (tok0) :
Device Type: Token-Ring High-Performance Adapter (8fc8)
Hardware Address: 10:00:5a:4f:26:c1
Elapsed Time: 0 days 0 hours 8 minutes 33 seconds
Transmit Statistics:                Receive Statistics:
-----
Packets: 191                        Packets: 8342
Bytes: 17081                         Bytes: 763227
Interrupts: 156                      Interrupts: 8159
Transmit Errors: 0                   Receive Errors: 0
Packets Dropped: 0                  Packets Dropped: 0
Max Packets on S/W Transmit Queue: 17 Bad Packets: 0
S/W Transmit Queue Overflow: 0
Current S/W+H/W Transmit Queue Length: 0

Broadcast Packets: 1                 Broadcast Packets: 8023
Multicast Packets: 0                 Multicast Packets: 0
Timeout Errors: 0                    Receive Congestion Errors: 0
Current SW Transmit Queue Length: 0
Current HW Transmit Queue Length: 0

General Statistics:
-----
No mbuf Errors: 0                    Lobe Wire Faults: 0
Abort Errors: 0                       AC Errors: 0
Burst Errors: 0                       Frame Copy Errors: 0
Frequency Errors: 0                   Hard Errors: 0
Internal Errors: 0                     Line Errors: 0
Lost Frame Errors: 0                   Only Station: 0
Token Errors: 0                        Remove Received: 0

```

```
Ring Recovered: 0                Signal Loss Errors: 0
Soft Errors: 0                   Transmit Beacon Errors: 0
Driver Flags: Up Broadcast Running
AlternateAddress ReceiveFunctionalAddr
```

2. 要显示 **tok0** 令牌环设备驱动程序统计信息和令牌环特定于设备统计信息, 请输入:

```
tokstat -d tok0
```

这将产生以下输出:

```
TOKEN-RING STATISTICS (tok0) :
Device Type: Token-Ring High-Performance Adapter (8fc8)
Hardware Address: 10:00:5a:4f:26:c1
Elapsed Time: 0 days 2 hours 48 minutes 38 seconds

Transmit Statistics:                Receive Statistics:
-----
Packets: 389                        Packets: 153216
Bytes: 42270                        Bytes: 14583150
Interrupts: 354                     Interrupts: 151025
Transmit Errors: 0                  Receive Errors: 0
Packets Dropped: 0                 Packets Dropped: 0
Max Packets on S/W Transmit Queue:17 Bad Packets: 0
S/W Transmit Queue Overflow: 0
Current S/W+H/W Transmit Queue Length: 0

Broadcast Packets: 1                Broadcast Packets: 152642
Multicast Packets: 0                Multicast Packets: 0
Timeout Errors: 0                   Receive Congestion Errors: 0
Current SW Transmit Queue Length: 0
Current HW Transmit Queue Length: 0

General Statistics:
-----
No mbuf Errors: 0                   Lobe Wire Faults: 0
Abort Errors: 0                     AC Errors: 0
Burst Errors: 0                     Frame Copy Errors: 0
Frequency Errors: 0                 Hard Errors: 0
Internal Errors: 0                  Line Errors: 0
Lost Frame Errors: 0                Only Station: 0
Token Errors: 0                     Remove Received: 0
Ring Recovered: 0                   Signal Loss Errors: 0
Soft Errors: 0                       Transmit Beacon Errors: 0
Driver Flags: Up Broadcast Running
AlternateAddress ReceiveFunctionalAddr

Token-Ring High-Performance Adapter (8fc8) Specific Statistics:
-----
DMA Bus Errors: 0                   DMA Parity Errors: 0
ARI/FCI Errors: 0
```

## 相关信息

**atmstat** 命令、**entstat** 命令、**fddistat** 命令、**netstat** 命令。

---

## topas 命令

### 用途

报告所选的本地和远程系统的统计信息。

## 语法

**topas** [ **-d** *number\_of\_monitored\_hot\_disks* ] [ **-h** ] [ **-i** *monitoring\_interval\_in\_seconds* ] [ **-n** *number\_of\_monitored\_hot\_network\_interfaces* ] [ **-p** *number\_of\_monitored\_hot\_processes* ] [ **-w** *number\_of\_monitored\_hot\_WLM\_classes* ] [ **-c** *number\_of\_monitored\_hot\_CPUs* ] [ **-l** *remote\_polling\_interval* ] [ **-U** *username\_owned\_processes* ] | [ **-C** [ **-o** *field = value,...* ] **-D** | **-L** | **-P** | **-W** ] [ **-m** ]

## 描述

**topas** 命令报告选定的本地系统活动的统计信息。该命令使用 `curses` 库以合适的格式来显示其输出，该格式适合于在一个 80x25 基于字符的显示器上或至少同样大小的图形显示器的窗口中进行查看。**topas** 命令需要在系统中安装 **bos.perf.tools** 和 **perfagent.tools** 文件集。

**topas** 命令还可以从属于同一硬件平台的远程 AIX 分区报告有限的性能指标集。『交叉分区视图』这部分中描述了这种支持。

如果 **topas** 命令调用时没有标志，则运行时正如用以下命令行来调用：

```
topas -d20 -i2 -n20 -p20 -w20 -c20
```

程序按由 *monitoring\_interval\_in\_seconds* 参数指定的时间间隔从系统抽取统计信息。缺省输出如下所示，包含两个固定部分和一个可变部分。显示器左边的最上方两行显示了 **topas** 命令运行所在的系统的名称、上次查看的日期和时间以及监视时间间隔。

第二个固定部分占用了显示器的最右端的 25 个位置。它包含统计信息的 5 个子节，如下所示：

|         |                                           |
|---------|-------------------------------------------|
| 事件 / 队列 | 显示选定的系统全局事件的每秒频率、线程运行和等待队列的平均大小：          |
| 上下文切换   | 在监视时间间隔内每秒上下文切换的数量。                       |
| 系统调用    | 在监视时间间隔内每秒执行的系统调用的总数。                     |
| 读       | 在监视时间间隔内每秒执行的 <code>read</code> 系统调用的数量。  |
| 写       | 在监视时间间隔内每秒执行的 <code>write</code> 系统调用的数量。 |
| 派生      | 在监视时间间隔内每秒执行的 <code>fork</code> 系统调用的数量。  |
| 执行      | 在监视时间间隔内每秒执行的 <code>exec</code> 系统调用的数量。  |
| 运行队列    | 准备运行但需要等待处理器可用的平均线程数目。                    |
| 等待队列    | 正在等待页面调度完成的平均线程数目。                        |

## 文件/TTY

显示所选文件与 **tty** 统计信息的每秒频率。

**读字符** 在监视时间间隔内 **read** 系统调用每秒读的字节数。

**写字符** 在监视时间间隔内 **write** 系统调用每秒写的字节数。

### 原始输入

在监视时间间隔内每秒从 TTY 中读取的原始字节数。

### Tty 输出

在监视时间间隔内每秒写入 TTY 中的字节数。

**Igets** 在监视时间间隔内每秒调用索引节点查找例程的数量。

**Namei** 在监视时间间隔内每秒调用路径名查找例程的数量。

**目录块** 在监视时间间隔内被目录搜索例程每秒扫描的目录块数目。

## 页面调度

显示页面调度统计信息的每秒频率。

**故障** 在监视时间间隔内每秒缺页故障的总数。这包括不能激活页面调度的缺页故障。

**占用** 在监视时间间隔内每秒钟有物理内存 4K 帧被虚拟内存管理器占用。

### 调页空间输入

在监视时间间隔内每秒钟从调页空间读取 4K 页面的数量。

### 调页空间输出

在监视时间间隔内每秒钟将 4K 页面写到调页空间的数量。

### 页面调进

在监视时间间隔内每秒钟读取 4K 页面的数量。这包括与从文件系统读取有关的页面调度活动。从此值中减去 **PgspIn** 就可得到在监视时间间隔内每秒钟从文件系统读取的 4K 页面的数量。

### 页面调出

在监视时间间隔内每秒钟写 4K 页面的数量。这包括与写入文件系统有关的页面调度活动。从此值中减去 **PgspOut** 就可得到在监视时间间隔内每秒钟写入文件系统的 4K 页面的数量。

## 内存

**Sios** 在监视时间间隔内虚拟内存管理器每秒钟发出的 I/O 请求的数目。

显示实内存大小与内存使用的分布情况。

### 实内存, MB

以 MB 为单位的实内存大小。

### 计算页面百分比

当前分配给计算页面帧的实内存的百分比。计算页面帧通常是那些被调页空间支持的帧。

### 非计算页面百分比

当前分配给非计算页面帧的实内存的百分比。非计算页面帧通常是那些被文件空间（可以是数据文件、可执行文件或共享库文件）支持的帧。

### 客户机百分比

当前被分配用来对远程安装的文件进行高速缓存的实内存的百分比。

## 调页空间

显示调页空间的大小和使用率。

### 大小, MB

系统上所有调页空间的总和, 以 MB 为单位。

### 已使用百分比

当前正在使用的调页空间所占的总百分比。

### 空闲百分比

当前未使用的调页空间所占的总百分比。

## NFS

显示每秒调用的 NFS 状态

- 服务器 V2 calls/sec
- 客户机 V2 calls/sec
- 服务器 V3 calls/sec
- 客户机 V3 calls/sec

**topas** 显示的变量部分可有一、二、三、四或五个子节。如果有多个子节显示, 则总是按照以下顺序显示:

- CPU
- 网络接口
- 物理磁盘
- 工作负载管理类
- 进程

当 **topas** 命令启动时, 会显示热实体受监视的所有子节。一个例外就是工作负载管理 (WLM) 类子节, 仅当 WLM 活动时才显示该子节。

## CPU 使用率

此子节显示一个条形图表来表示累积的 CPU 使用率。如果有多个 CPU, 按 **c** 键两次就可显示 CPU 列表。仅按 **c** 键一次将会关闭此子节。以下字段通过两种格式显示:

### 用户百分比

这显示以用户方式执行的程序所使用的 CPU 的百分比。(缺省按用户百分比排序)

### 内核百分比

这显示以内核方式执行的程序所使用的 CPU 的百分比。

### 等待百分比

这显示用于等待 IO 的时间的百分比。

### 空闲百分比

这表示 CPU 空闲时间的百分比。

**Physc** 消耗的物理处理器数目。仅当使用共享处理器运行分区时才显示。

**%Entc** 消耗的授权容量百分比。仅当使用共享处理器运行分区时才显示。

当此子节首先显示热 CPU 列表时, 就按用户百分比字段来对列表排序。但是, 可以按其他字段来对列表排序, 只要将光标移到期望栏的顶部就可以了。

## 网络接口

此子节显示了热网络接口的列表。所显示接口的最大数目是正在受监视的热接口数目，如 **-n** 标志指定的那样。如果其他子节也正在被显示，则将显示接口的一个较小数目。按 **n** 键可关闭此子节。再次按 **n** 键显示一行所有网络接口活动的报告摘要。两个报告都显示以下字段：

**接口** 网络接口的名称。

**千位/秒**

在监视时间间隔内每秒钟以 MB 为单位的总吞吐量。该字段是每秒接收到的千字节和发送的千字节的总和。

**输入包** 在监视时间间隔内每秒钟接收到的数据包的数目。

**输出包** 在监视时间间隔内每秒钟发送的数据包的数目。

**输入千字**

在监视时间间隔内每秒钟接收到的千字节的数目。

**输出千字节**

在监视时间间隔内每秒钟发送的千字节的数目。

当此子节首先显示热网络接口列表时，按 **KBPS** 字段对列表排序。但是，可以按其他字段来对列表排序，只要将光标移到期望栏的顶部就可以了。排序仅对最多 16 个网络适配器有效。

## 物理磁盘

此子节显示热物理磁盘列表。所显示物理磁盘的最大数目是正受监视的热物理磁盘数目，正如 **-d** 标志指定的那样。如果其他子节也正在被显示，则将显示物理磁盘的一个较小数目。按 **d** 键可关闭此子节。再次按 **d** 键显示一行所有物理磁盘活动的报告摘要。两个报告都显示以下字段：

**磁盘** 物理磁盘的名称。

**忙碌百分比**

表示物理磁盘活动时间的百分比（驱动器带宽使用率）。

**千位/秒**

在监视时间间隔内每秒钟读和写的千字节的数目。此字段是 **KB-Read** 和 **KB-Writ** 的总和。

**TPS** 每秒钟向物理磁盘发出的传输数目。传输是对物理磁盘的 I/O 请求。多个逻辑请求可组合成对磁盘的单个 I/O 请求。传输大小不确定。

**读取千字节**

每秒钟从物理磁盘读取的千字节的数目。

**写入千字节**

每秒钟写到物理磁盘的千字节的数目。

当此子节首先显示热物理磁盘列表时，按 **KBPS** 字段对列表排序。但是，可以按其他字段来对列表排序，只要将光标移到期望栏的顶部就可以了。排序仅对最多 128 个物理磁盘有效。

## WLM 类

此子节显示热工作负载管理（WLM）类的列表。所显示 WLM 类的最大数目是正在受监视的热 WLM 类数目，如 **-w** 标志指定的那样。如果其他子节也正在被显示，则将显示 WLM 类的一个较小数目。按 **w** 键可关闭此子节。对于每一个类会显示以下字段：

**CPU 使用率百分比**

在监视时间间隔内 WLM 类的平均 CPU 使用率。

**内存使用率百分比**

在监视时间间隔内 WLM 类的平均内存使用率。

**块 I/O 百分比**

在监视时间间隔内 WLM 类的块 I/O 的平均百分比。

当此子节首先显示热 WLM 类列表时，就按 CPU 百分比字段来对列表排序。但是，可以按其他字段来对列表排序，只要将光标移到期望栏的顶部就可以了。



## 进程

此子节显示热进程列表。所显示进程的最大数目是正在受监视的热过程数目，如 **-p** 标志指定的那样。如果其他子节也正在被显示，则将显示进程的一个较小数目。按 **p** 键可关闭此子节。按监视时间间隔内进程的 CPU 使用率来对进程排序。对于每一进程会显示以下字段：

**名称** 在进程中执行的可执行程序名称。名称已被除去任何路径名和参数信息并被截断到 9 个字符的长度。

### 进程标识

进程的进程标识。

### CPU 使用率百分比

在监视时间间隔内进程的平均 CPU 使用率。第一次显示进程时，此值表示整个进程生命期上的平均 CPU 使用率。

### 已使用调页空间

分配给此进程的调页空间的大小。这可被认为是进程覆盖区的一种表达，但并不包括用来保持可执行程序和他依赖的任何共享库的内存。

### 进程所有者（如果 WLM 部分被关闭）

拥有此进程的用户的用户名。

### 工作负载管理（WLM）类（如果 WLM 部分被打开）

进程所属的 WLM 类。

## 交叉分区视图

对于该面板可确定为属于同一硬件平台的所有 AIX 分区，它显示与 **lparstat** 命令相似的度量值。专用分区和共享分区及相应的度量值显示在单独的部分中。顶端部分表示从分区集收集的聚集数据，以显示整个分区、内存和处理器活动。

要远程启用此面板以从其他分区进行收集，需要将用于支持此功能的 **perfagent.tools** 和 **bos.perf.tools** 更新到最新版本。对于 AIX 的较早版本，**topas** 命令还从安装了 Performance Aide 产品 (**perfagent.server**) 的分区中收集远程数据。**topas -C** 可能不能定位到驻留在其他子网上的分区。为了避免发生这种情况，请创建 **\$HOME/Rsi.hosts** 文件，包含每个分区（包括域）的全限定主机名，每行一个主机。

以下度量值显示在最初的“交叉分区”面板中。可以通过使用在『其他交叉分区面板子命令』部分中确定的键切换来显示带有完整描述标签的其他度量值。

```
Partition totals
  Shr      Number of shared partitions
  Ded      Number of dedicated partitions
Memory (in GB)
  Mon      Monitored partitions total memory
  Avl      Memory available to partition set
  InUse    Memory in use on monitored partitions
Processor
  Shr      Number of shared processors
  Ded      Number of dedicated processors
  PSz     Active physical CPUs in the shared processor pool being used by this LPAR
  APP     Available physical processors in the shared pool
  Shr_PhysB Shared Physical Busy
  Ded_PhysB Dedicated Physical Busy
Individual partition data
  Host     Hostname
  OS       Operating system level
  M        Mode
          For shared partitions
          C - SMT enabled & capped
```

```

c - SMT disabled & capped
U - SMT enabled & uncapped
u - SMT disabled & uncapped
For dedicated partitions
S - SMT enabled
' ' (blank) - SMT enabled

```

```

Mem      Total memory in GB.
InU      Memory in use in GB.
Lp       Number of logical processors
Us       Percentage of CPU used by programs executing in user mode.
Sy       Percentage of CPU used by programs executing in kernel mode.
Wa       Percentage of time spent waiting for IO.
Id       Percentage of time the CPU(s) is idle.
Ded_PhysB Dedicated Physical Busy
Ent       Entitlement granted (shared-only)
%Entc    Percent Entitlement consumed (shared-only)
Vcsw     Virtual context switches average per second (shared-only)
PhI      Phantom interrupts average per second (shared-only)

```

## 实现细节

启动 **topas** 或任何其他 **SPMI** 使用者后添加的磁盘和网络适配器将不会反映在 **topas** 中。必须停止 **topas** 和所有使用 **SPMI** 的客户机，然后在对磁盘和网络适配器作了更改后重新启动。

## 缺省输出样本

下面为 **topas** 命令产生的显示示例:

```

Topas Monitor for host:  niller          EVENTS/QUEUES  FILE/TTY
Mon Mar 13 15:56:32 2000  Interval:  2    Cswitch      113  Readch      1853576
                               Syscall      2510 Writech      49883
CPU   User%   Kern%   Wait%   Idle%   Reads        466  Rawin        0
cpu0   7.0     4.0     0.0    89.0   Writes        12  Ttyout       706
cpu1   1.0     8.0     0.0    91.0   Forks         0  Igets        0
cpu2   0.0     0.0     0.0   100.0  Execs         0  Namei        0
                               Runqueue     0.0  Dirblk       0
                               Waitqueue    0.0

Interf  KBPS   I-Pack  O-Pack  KB-In  KB-Out
lo0     100.4  45.7    45.7    50.2   50.2
tr0     2.0    4.4     3.4     1.4    0.6

Disk   Busy%   KBPS     TPS  KB-Read  KB-Writ
hdisk0 0.0     0.0     0.0   0.0     0.0
hdisk1 0.0     0.0     0.0   0.0     0.0

WLM-Class (Active) CPU%   Mem%   Disk%
System             8     41     12
Shared             1     24     9

Name   PID     CPU%   PgSP  Class
topas  (35242) 3.0    0.3   System
X      (3622)  1.4   44.4   System
notes (25306) 1.3  123.3  System

EVENTS/QUEUES  FILE/TTY
PAGING         MEMORY
Faults         1  Real,MB      255
Steals         0  % Comp       81.0
PgspIn        0  % Noncomp    19.0
PgspOut       0  % Client     3.0
PageIn        0
PageOut       0  PAGING SPACE
Sios          0  Size,MB      0
              % Used
              % Free

NFS      calls/sec
ServerV2  0  Press:
ClientV2  0  "h" for help.
ServerV3  0  "q" to quit.
ClientV3  0

```

## 全屏幕进程输出样本

```

Topas Monitor for host:  mothra          Interval:  2    Wed Nov 8 12:27:34 2000
                               DATA  TEXT  PAGE  PGFAULTS
USER   PID  PPID  PRI  NI   RES  RES  SPACE  TIME CPU%  I/O  OTH  COMMAND
root   1806  0    37  41   16  3374  16    13:25 1.0   0    0  gil
root   1032  0    16  41   3   3374  3     0:00 0.0   0    0  lrud
root   1290  0    60  41   4   3374  4     0:02 0.0   0    0  xmgc
root   1548  0    36  41   4   3374  4     0:26 0.0   0    0  netm
root    1    0    60  20  197   9   180   0:24 0.0   0    0  init
root   2064  0    16  41   4   3374  4     0:04 0.0   0    0  wlmsched

```

```

root    2698    1 60 20   14    2   14   0:00 0.0   0   0 shlap
root    3144    1 60 20   40    1   36   5:19 0.0   0   0 syncd
root    3362    0 60 20    4 3374    4   0:00 0.0   0   0 lvmbb
root    3666    1 60 20  135   23  123   0:00 0.0   0   0 errdemon
root    3982    0 60 20    4 3374    4   0:01 0.0   0   0 rtcmd
root    4644    1 17 20    6 3374    6   0:00 0.0   0   0 dog
root    4912    1 60 20  106   13   85   0:00 0.0   0   0 srcmstr
root    5202  4912 60 20   94    8   84   0:01 0.0   0   0 syslogd
root    5426  4912 60 20  195   76  181   0:12 0.0   0   0 sendmail
root    5678  4912 60 20  161   11  147   0:01 0.0   0   0 portmap
root    5934  4912 60 20  103   11   88   0:00 0.0   0   0 inetd
root    6192  4912 60 20  217   61  188   0:21 0.0   0   0 snmpd
root    6450  4912 60 20  137   10  116   0:00 0.0   0   0 dpid2
root    6708  4912 60 20  157   29  139   0:06 0.0   0   0 hostmibd
root     0     0 16 41    3 3374    3   7:08 0.0   0   0
root    6990    1 60 20  106   10   86   0:06 0.0   0   0 cron

```

## 全屏幕工作负载管理类输出样本

```

Topas Monitor for host:   mothra      Interval:  2   Wed Nov  8 12:30:54 2000
WLM-Class (Active)      CPU%      Mem%      Disk-I/O%
System                  0          0          0
Shared                  0          0          0
Default                 0          0          0
Unmanaged               0          0          0
Unclassified           0          0          0

```

```

=====
USER      PID  PPID  PRI  NI   DATA  TEXT  PAGE      TIME CPU%  I/O  OTH  COMMAND
root      1    0 108 20   197    9   180    0:24 0.0   0   0  init
root     1032  0 16 41    3 3374    3    0:00 0.0   0   0  lrud
root     1290  0 60 41    4 3374    4    0:02 0.0   0   0  xmgc
root     1548  0 36 41    4 3374    4    0:26 0.0   0   0  netm
root     1806  0 37 41   16 3374   16   13:25 0.0   0   0  gil
root     2064  0 16 41    4 3374    4    0:04 0.0   0   0  wlmsched
root     2698  1 108 20   14    2   14    0:00 0.0   0   0  shlap
root     3144  1 108 20   40    1   36    5:19 0.0   0   0  syncd
root     3362  0 108 20    4 3374    4    0:00 0.0   0   0  lvmbb
root     3666  1 108 20  135   23  123    0:00 0.0   0   0  errdemon
root     3982  0 108 20    4 3374    4    0:01 0.0   0   0  rtcmd

```

## 全屏幕交叉分区输出样本

```

Topas CEC Monitor      Interval:  10      Wed Mar  6 14:30:10 2005
Partitions      Memory (GB)      Processors
Shr:  4      Mon:  24 InUse:  14      Mon:  8 PSz:  4 Shr_PhysB:  1.7
Ded:  4      Avl:  24      Avl:  8 APP:  4 Ded_PhysB:  4.1

Host      OS  M Mem InU Lp  Us Sy Wa Id  PhysB  Ent  %EntC  Vcsw  Phi
-----shared-----
ptools1   A53 u 1.1 0.4  4  15  3  0 82   1.30  0.50  22.0  200  5
ptools5   A53 U 12 10  1  12  3  0 85   0.20  0.25  0.3  121  3
ptools3   A53 C 5.0 2.6  1  10  1  0 89   0.15  0.25  0.3   52  2
ptools7   A53 c 2.0 0.4  1   0  1  0 99   0.05  0.10  0.3  112  2
-----dedicated-----

```

```
ptools4    A53 S 0.6 0.3 2 12 3 0 85 0.60
ptools6    A52  1.1 0.1 1 11 7 0 82 0.50
ptools8    A52  1.1 0.1 1 11 7 0 82 0.50
ptools2    A52  1.1 0.1 1 11 7 0 82 0.50
```

## 标志

- c** 指定要受监视的热 CPU 的数目。这也是在屏幕有足够可用空间时显示的 CPU 的最大数目。当此数目超过可用的 CPU 数目时，只有已安装的 CPU 将受监视和显示。如果此参数被省略，假定 2 为缺省值。如果指定值为 0（零），就不监视 CPU 信息。
- C** 显示“交叉分区”面板。**topas** 命令从在同一硬件平台上运行的 AIX 分区收集一组度量值。这些度量值与 **lparstat** 命令收集的度量值类似。此时显示专用和共享分区，一组聚集值概述了整个硬件系统分区集。如果 HMC 连接不可用，某些来自 HMC 平台的值仅可以通过 **line** 命令进行设置。
- -o availmem = [分配给所有分区的总内存量，以 GB 为单位]
  - -o unavailmem = [未从 HMC 分配的总内存量，以 GB 为单位]
  - -o availcpu = [分配给所有分区的物理处理器的总数]
  - -o unavailcpu = [未从 HMC 分配的物理处理器的总数]
  - -o poolsize = [在 HMC 处理器使用率权限限制访问的情况下所需的已定义的池大小]
  - -o partitions = [在 HMC 上定义的分区的数目]
  - -o reconfig = [两次检查 HMC 配置更改之间的秒数。允许的值为 30、60、90、120、180、240 和 300 秒。缺省值为 60 秒]
- D** 显示“磁盘度量值”显示。该显示报告磁盘服务时间、磁盘排队度量值以及磁盘吞吐量。报告以下度量值：
- AQD** 等待发送到磁盘的请求的平均数量。
- AQW** 每个请求的平均队列等待时间，以毫秒计。
- ART** 表示从主管服务器接收对于发送的读请求的响应所用的平均时间。后缀表示时间的单位。缺省的时间单位是毫秒。
- AWT** 表示从主管服务器接收对于发送的写请求的响应所用的平均时间。后缀表示时间的单位。缺省的时间单位是毫秒。
- MRT** 表示从主管服务器接收对于发送的读请求的响应所用的最长时间。后缀表示时间的单位。缺省的时间单位是毫秒。
- MWT** 表示从主管服务器接收对于发送的写请求的响应所用的最长时间。后缀表示时间的单位。缺省的时间单位是毫秒。
- d** 指定要受监视的磁盘数目。这也是在屏幕有足够的可用空间时显示的磁盘的最大数目。当此数目超过所安装的磁盘数目时，只有已安装的磁盘将受监视和显示。如果此参数被省略，假定 2 为缺省值。如果指定值为 0（零），就不监视磁盘信息。
- h** 用以下格式显示帮助信息：
- ```
用法: topas [-d number-of-monitored-hot-disks]
           [-h]
           [-i monitoring-interval_in_seconds]
           [-n number-of-monitored-hot-network-interfaces]
           [-p number-of-monitored-hot-processes]
           [-w number-of-monitored-hot-WLM classes]
           [-c number-of-monitored-hot-CPU]s]
           [-U username_owned_processes]
           [-D|-P|-W|-L]
           [-m]
```
- i** 以秒为单位设置监视时间间隔。缺省值为 2 秒。

|                                          |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-l</b> <i>remote_polling_interval</i> | 对于交叉分区显示, 设置从远程分区中收集数据的采样时间间隔。缺省值为 10 秒。允许的值为 10、15、30、60 和 120 秒。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| <b>-L</b>                                | 显示逻辑分区显示。此显示将相似数据报告给提供到 <b>mpstat</b> 和 <b>lparstat</b> 的内容。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| <b>-m</b>                                | 以单色方式显示 (无颜色)。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| <b>-n</b>                                | 指定要受监视的热网络接口的数目。这也是在屏幕有足够可用空间时显示的网络接口的最大数目。当此数目超过所安装的网络接口数目时, 只有已安装的网络接口将受监视和显示。如果此参数被省略, 假定 2 为缺省值。如果指定值为 0 (零), 就不监视网络信息。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| <b>-P</b>                                | 与 <b>ps</b> 命令相似, <b>-P</b> 标志显示全屏进程显示。显示了最忙进程的列表, 与缺省显示时的进程子节类似, 只是要用更多的栏来显示每个过程的更多规格信息。此表可按任意一栏来排序。以下是显示的度量值。<br><br><b>USER</b> 进程所有者的登录名。<br><b>PID</b> 进程的进程标识。<br><b>PPID</b> 父进程的进程标识。<br><b>PRI</b> 进程或内核线程的优先级; 数字越大表示优先级越低。<br><b>NI</b> 细调值; 用于计算调度其他策略的优先级。<br><br><b>Data RSS</b><br>进程的实内存数据 (驻留集) 大小 (4KB 页)。<br><br><b>TEXT RSS</b><br>进程的实内存文本 (驻留集) 大小 (4KB 页)。<br><br><b>PAGE SPACE</b><br>进程使用的虚拟工作集大小 (4KB 页)。注: 使用 <b>topas</b> 命令无法获得每个进程的真实调页空间分配。关于更多详细报告的信息, 请参阅 <b>svmon</b> 命令。<br><br><b>TIME</b> 进程的执行总时间。<br><br><b>CPU%</b> CPU 使用百分比。<br><br><b>PGFAULTS</b><br>显示 I/O 及其他缺页故障数。<br><br><b>COMMAND</b><br>包含命令名。将命令名截断到 9 个字符。 |
| <b>-p</b>                                | 指定要受监视的热进程数目。这也是在屏幕有足够可用空间时显示的进程的最大数目。如果此参数被省略, 假定 20 为缺省值。如果指定值为 0, 就不监视进程信息。对进程信息的检索构成了 <b>topas</b> 的主要开销。如果不需要进程信息, 总是使用此选项来指定不需要进程信息。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| <b>-U</b> <i>username</i>                | 用 <b>-P</b> 标志显示 <i>username</i> 拥有的进程。仅 <i>username</i> 拥有的进程将显示在所有进程显示中。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| <b>-W</b>                                | 显示全屏幕 WLM 类显示, 此为分割显示。显示器的顶部显示热 WLM 类的列表, 类似于缺省显示时的 WLM 类子节, 只是要用足够可用空间来显示全部类名。此表可按任意一栏来排序。<br><br>显示器的低部显示最忙进程的列表, 与全屏幕进程显示类似, 但仅显示属于一个 WLM 类 (用 <b>f</b> 键选择) 的进程。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| <b>-w</b>                                | 指定要受监视的热工作负载管理 (WLM) 类的数目。这也是在屏幕有足够可用空间时显示的 WLM 类的最大数目。当此数目超过所安装的 WLM 类数目时, 只有已安装的 WLM 类将受监视和显示。如果此参数被省略, 假定 2 为缺省值。如果指定值为 0 (零), 就不监视 WLM 类信息。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |

## 一般子命令

当 **topas** 运行时，它接受单字符的子命令。每次监视时间间隔过去时，程序会对以下子命令之一进行检查并对所请求的操作作出响应。

|          |                                                                                                                                       |
|----------|---------------------------------------------------------------------------------------------------------------------------------------|
| <b>a</b> | <b>a</b> 键可显示所有正在受监视的变量子节（CPU、网络、磁盘、WLM、过程）。按 <b>a</b> 键总使 <b>topas</b> 命令返回到初始主显示屏幕。                                                 |
| <b>c</b> | <b>c</b> 键可使 CPU 子节在累积报告、关闭和最忙 CPU 列表这三者之间进行切换。所显示的最忙 CPU 数目将取决于屏幕上的可用空间。                                                             |
| <b>C</b> | 大写的 <b>C</b> 键会激活“交叉分区”面板，或者，如果它当前是活动的，则会重新设置面板以显示全局摘要、专用部分和共享部分。请参阅下面的『其他交叉分区面板子命令』部分以获取特定于此面板的选项。                                   |
| <b>d</b> | <b>d</b> 键可使磁盘子节在最忙磁盘列表、关闭和系统全部磁盘活动的报告这三者之间进行切换。所显示的最忙磁盘数目将取决于屏幕上的可用空间。                                                               |
| <b>D</b> | 大写 <b>D</b> 键将当前显示替换为“磁盘度量值”显示。该显示提供关于磁盘访问时间以及磁盘排队的其他信息。再次按 <b>D</b> 键时，就会切换回缺省主屏幕。                                                   |
| <b>f</b> | 将光标移动到一个 WLM 类上，并按 <b>f</b> 键就可在 WLM 屏幕底部显示该类中顶部进程的列表。该键功能只有 <b>topas</b> 在全屏幕 WLM 显示（通过使用 <b>W</b> 键或 <b>-W</b> 标志）时才有效。             |
| <b>h</b> | 显示帮助屏幕。                                                                                                                               |
| <b>H</b> | 显示本地面板的帮助屏幕（如果可用）。                                                                                                                    |
| <b>L</b> | 大写 <b>L</b> 键用逻辑分区显示替换了当前显示；显示了类似于 <b>lparstat</b> 和 <b>mpstat</b> 提供的内容的 LPAR、微分区和同时多线程度量值。                                          |
| <b>n</b> | <b>n</b> 键可使网络接口子节在最忙接口列表、关闭和系统全部网络活动的报告这三者之间进行切换。所显示的最忙接口数目将取决于屏幕上的可用空间。                                                             |
| <b>p</b> | <b>p</b> 键可在打开与关闭热进程子节之间进行切换。所显示的最忙进程数目取决于屏幕上的可用空间。                                                                                   |
| <b>P</b> | 大写 <b>P</b> 键用全屏幕进程显示来替换缺省显示。有关运行在系统上的进程，此显示提供了比主显示的进程部分更详细的信息。再次按 <b>P</b> 键时，就会切换回缺省主显示。                                            |
| <b>q</b> | 退出程序。                                                                                                                                 |
| <b>r</b> | 刷新显示。                                                                                                                                 |
| <b>w</b> | <b>w</b> 键可在打开与关闭工作负载管理（WLM）类子节之间进行切换。所显示的最忙 WLM 类数目取决于屏幕上的可用空间。                                                                      |
| <b>W</b> | 大写 <b>W</b> 键用全屏幕 WLM 类显示来替换缺省显示。此显示提供了有关 WLM 类和指定给类的进程的更详细信息。再次按 <b>W</b> 键时，就会切换回缺省主显示。                                             |
| 箭头键与跳格键  | 主显示中的各个子节，诸如 CPU、网络、磁盘、WLM 类、全屏幕 WLM 与进程显示的子节可按不同标准进行排序。将光标移动到一栏上，然后会按该栏进行排序。各项总是从最高值到最低值排序。使用跳格键或箭头键来移动光标。排序只对 128 个磁盘和 16 个网络适配器有效。 |

## 其他交叉分区面板子命令

当 **topas** “交叉分区”面板处于活动状态时，它将接受以下更多单字符子命令。每次监视时间间隔过去时，程序都会对以下的一个子命令进行检查并对任何请求的操作作出响应。

|          |                                                                       |
|----------|-----------------------------------------------------------------------|
| <b>d</b> | <b>d</b> 键可使得专用分区部分在打开和关闭之间切换。                                        |
| <b>g</b> | <b>g</b> 键可使面板顶部全局部分在简短列表、详细列表和关闭之间切换。                                |
| <b>r</b> | 如果连接可用，则 <b>r</b> 键可强制执行 <b>topas</b> 搜索 HMC 配置更改。这包括发现的新分区、处理器或内存分配。 |
| <b>s</b> | <b>s</b> 键可使得共享分区部分在打开和关闭之间切换。                                        |

## 示例

1. 要每 5 秒显示最多 20 个“热”磁盘而省略网络接口、WLM 类和进程信息，请输入：

```
topas -i5 -n0 -p0 -w0
```

2. 要显示 5 个最活跃的进程和最多 20 个最活跃的 WLM 类（省略 **-w** 标志时此为缺省值），而不显示网络或磁盘信息，请输入：

```
topas -p5 -n0 -d0
```

3. 要使用缺省选项运行程序，请输入：

```
topas
```

4. 要直接进入进程显示，请输入：

```
topas -P
```

5. 要直接进入 WLM 类显示，请输入：

```
topas -W
```

6. 要直接进入逻辑分区显示，请输入：

```
topas -L
```

7. 要直接进入磁盘度量值显示，请输入：

```
topas -D
```

## 文件

**/usr/bin/topas** 包含 **topas** 命令。

## 相关信息

**iotstat** 命令、**vmstat** 命令、**wlmstat** 命令、**sar** 命令、**ps** 命令、**mpstat** 命令、**lparstat** 命令。

*Performance Toolbox Version 2 and 3 for AIX: Guide and Reference* 中的 System Performance Measurement Interface。

随 **perfgent.server** 一起提供的 **lchmon** 样本程序。

---

## topasout 命令

### 用途

将 **xmwlm** 记录文件的内容制成表格。

### 语法

```
topasout [ -c | -s ] [ -m type ] recording_file
```

### 描述

**topasout** 命令提供对 **xmwlm** 所作记录的后处理。它可生成 ASCII 或电子表格格式的报告。

### 标志

**-c**

指定 **ptxtab** 应该将输出文件的格式编排为以逗号分隔的 ASCII 格式。输出文件中的每一行都包含一个时间戳记和一个观察值。两个字段前都有描述该字段的标签。

**-s**

指定 **topasout** 应该将输出文件的格式编排为适合输入电子表格程序的格式。如果指定了此标志，则始终假定还指定了 **-r** 标志。

**-m type**

在缺省情况下，后处理程序只输出平均值。其他记录值和完整集合是通过其他选项（min、max、mean、stdev、set 和 exp）提供的。

## 示例

1. 下面是 **topasout** 的以逗号分隔的输出格式的示例:

```
#Monitor: Nice Monitor --- hostname: nchris
Time="1994/01/07 15:36:03", PagSp/%totalused=27.82
Time="1994/01/07 15:36:03", PagSp/%totalfree=72.18
Time="1994/01/07 15:36:03", Mem/Virt/pagein=8
Time="1994/01/07 15:36:03", Mem/Virt/pageout=20
Time="1994/01/07 15:36:07", PagSp/%totalused=27.82
Time="1994/01/07 15:36:07", PagSp/%totalfree=72.18
Time="1994/01/07 15:36:07", Mem/Virt/pagein=7
Time="1994/01/07 15:36:07", Mem/Virt/pageout=17
```

2. 下面是 **topasout** 的电子表格输出格式的示例:

```
#Monitor: Nice Monitor --- hostname: nchris
"Timestamp" "PagSp/%totalused" "PagSp/%totalfree" "Mem/Virt/pagein" "Mem/Virt/pageout"
"1994/01/07 15:36:03" 27.8 72.2 8 20
"1994/01/07 15:36:07" 27.8 72.2 7 17
"1994/01/07 15:36:11" 27.8 72.2 3 283
"1994/01/07 15:36:15" 27.8 72.2 28 48
"1994/01/07 15:36:19" 28.2 71.8 56 41
"1994/01/07 15:36:23" 29.5 70.5 29 38
"1994/01/07 15:36:27" 31.5 68.5 0 62
"1994/01/07 15:36:31" 32.4 67.6 70 1
"1994/01/07 15:36:35" 32.6 67.4 73 32
"1994/01/07 15:37:04" 28.0 72.0 15 0
```

## 位置

**/usr/bin/topasout**

## 文件

**/usr/bin/topasout**

包含 **topasout** 命令。**topasout** 命令包含在 `perfgent.tools` 文件集中。

## 相关信息

**topas** 命令和 **xmwlm** 命令。

---

## topsvcs 命令

### 用途

在一个集群节点上启动或重新启动拓扑服务。

### 语法

**topsvcs**



## 描述

使用 **topsvcs** 脚本来启动集群的拓扑服务操作。

**topsvcs** 脚本通常不能从命令行执行。它通常被 **topsvcsctrl** 控制脚本调用，该控制脚本接下来又被 HACMP/ES 启动进程调用。

**topsvcs** 脚本发出以下命令：

```
no -o nonlocsrcroute=1
no -o ipsrcroutesend=1
no -o ipsrcrouterrecv =1
no -o ipsrcrouteforward=1
```

这些命令启用 IP 源路由。不得更改此设置，因为拓扑服务子系统要求此设置正常工作。如果更改了此设置，拓扑服务子系统和许多依赖于它的其他子系统将不再正常运行。

## 标志

- s** 指示拓扑服务守护程序拒绝那些明显延迟的消息。
- d** 指示拓扑服务守护程序不得拒绝那些明显延迟的消息（这是缺省值）。

## 安全性

必须拥有 **root** 特权来运行此命令。

## 退出状态

- 0** 说明命令已成功完成。
- 1** 说明命令失败。

## 环境变量

### **HB\_SERVER\_SOCKET**

此环境变量应在此命令执行前进行设置。它必须被设置为拓扑服务客户机用来连接到拓扑服务守护程序的 UNIX 域套接字的位置。此环境变量必须设置为 **/var/ha/soc/hats/server\_socket,partition name**。

### **HA\_SYSPAR\_NAME**

如果 **HB\_SERVER\_SOCKET** 未设置，则 **HA\_SYSPAR\_NAME** 必须被设置为分区名。

## 限制

该命令只在 HACMP 环境中有效。

请只在 IBM 支持中心的指导下使用该命令。

## 标准输出

当指定 **-h** 标志时，将该命令的用法语句写入标准输出。所有详细消息都写到标准输出中。

## 标准错误

必要时此命令将错误消息写入标准错误。

## 示例

要指示本地节点上的拓扑服务守护程序开始废弃明显延迟的消息，请输入：

```
export HA_SYSPAR_NAME=分区1
```

```
/usr/sbin/rsct/bin/hatsoptions -s
```

## 位置

**/usr/sbin/rsct/bin/topsvcs** 包含 **topsvcs** 脚本

## 文件

**/var/ha/soc/hats/server\_socket.partition name**

## 相关信息

命令：**lssrc**、**startsrc**、**stopsrc**

脚本：**topsvcsctrl**

---

## topsvcsctrl 命令

### 用途

启动拓扑服务子系统。

### 语法

```
topsvcsctrl { -a | -s | -k | -d | -c | -u | -t | -o | -r | -h }
```

### 描述

**topsvcsctrl** 控制脚本控制拓扑服务子系统的运行。此子系统由系统资源控制器（SRC）进行控制，并属于一个称为 **topsvcs** 的子系统组。此脚本通常由 HACMP/ES 启动进程来启动。

拓扑服务子系统的实例运行在集群的每一个节点上。

从操作观点来看，拓扑服务子系统组采取如下组织：

子系统 拓扑服务

子系统组 **topsvcs**

SRC 子系统 **topsvcs**

**topsvcs** 子系统与 **hatsd** 守护程序和 **topsvcs** 脚本关联。**topsvcs** 脚本配置并启动 **hatsd** 守护程序。节点上的子系统名为 **topsvcs**。每个节点都有一个子系统并与节点所属的集群关联。

守护程序 **hatsd**

提供拓扑服务。**topsvcs** 脚本配置并启动 **hatsd** 守护程序。

**topsvcsctrl** 脚本通常不能从命令行执行。它通常被 HACMP/ES 启动命令调用。

**topsvcsctrl** 脚本提供了多种拓扑服务子系统的操作控制：

- 添加、启动、停止和删除子系统
- 清除子系统，即从所有系统分区将其删除
- 打开和关闭跟踪功能
- 刷新子系统

在执行任一这些功能之前，脚本获取当前集群名（使用 **cllscsstr** 命令）和节点号（使用 **clhandle** 命令）。如果节点号为 **0**，控制脚本就在控制工作站上运行。

除了清除与取消配置这两个功能，其余功能都可在当前系统分区作用域内执行。

**添加子系统：**如果指定了 **-a** 标志，此控制命令将使用 **mkssys** 命令将该拓扑服务子系统添加到 SRC。控制脚本操作如下：

1. 确保 **topsvcs** 子系统已停止。
2. 从 SRC 中除去 **topsvcs** 子系统（以防它仍在那儿）。
3. 将 **topsvcs** 子系统添加到 SRC 中。

**启动子系统：**当指定 **-s** 标志时，控制脚本使用 **startsrc** 命令来启动拓扑服务子系统 **topsvcs**。

**停止子系统：**当指定 **-k** 标志时，控制脚本使用 **stopsrc** 命令来停止拓扑服务子系统 **topsvcs**。

**删除子系统：**当指定 **-d** 标志时，控制脚本使用 **rmssys** 命令将拓扑服务子系统从 SRC 中除去。控制脚本操作如下：

1. 确保 **topsvcs** 子系统已停止。
2. 使用 **rmssys** 命令从 SRC 中除去 **topsvcs** 子系统。
3. 从 **/etc/services** 文件中除去端口号。

**清除子系统：**当指定 **-c** 标志时，控制脚本停止并从 SRC 的所有集群分区中除去拓扑服务子系统。控制脚本操作如下：

1. 使用 **stopsrc -g topsvcs** 命令停止集群中子系统的所有实例。
2. 从 **/etc/services** 文件中将 **topsvcs** 子系统的所有项都除去。

**打开跟踪功能：**当指定 **-t** 标志时，控制脚本使用 **traceson** 命令为 **hatsd** 守护程序打开跟踪功能。

**关闭跟踪功能：**当指定 **-o** 标志，控制脚本使用 **tracesoff** 命令为 **hatsd** 守护程序关闭跟踪功能（返回其缺省级别）。

**刷新子系统：**当指定 **-r** 标志时，控制脚本使用 **topsvcs refresh** 命令和 **refresh** 命令刷新子系统。在全局对象数据管理器（ODM）中重建有关节点和适配器配置的信息，并给守护程序发信号去读取重建信息。

**记录：**当拓扑服务守护程序（**hatsd**）运行时，它通过将项写入一个称为 **/var/ha/log/topsvcscluster\_name** 的日志文件来提供有关其操作和错误的信息。

## 标志

- |           |        |
|-----------|--------|
| <b>-a</b> | 添加子系统。 |
| <b>-s</b> | 启动子系统。 |
| <b>-k</b> | 停止子系统。 |
| <b>-d</b> | 删除子系统。 |

- c** 清除子系统。
- u** 从所有分区删除此拓扑服务子系统。
- t** 打开子系统跟踪功能。
- o** 关闭子系统跟踪功能。
- r** 刷新子系统。
- h** 将脚本的用法语句写入标准输出。

## 安全性

必须拥有有效的 **root** 用户标识才能使用此脚本。

## 退出状态

- 0** 指示脚本成功完成。
- 1** 表明发生错误。

## 环境变量

### **HB\_SERVER\_SOCKET**

此环境变量应在此命令执行前进行设置。它必须被设置为拓扑服务客户机用来连接到拓扑服务守护程序的 UNIX 域套接字的位置。此环境变量必须设置为 */var/ha/soc/hats/server\_socket,partition name*。

### **HA\_SYSPAR\_NAME**

如果 **HB\_SERVER\_SOCKET** 未设置，则 **HA\_SYSPAR\_NAME** 必须被设置为分区名。

## 限制

该命令只在 HACMP 环境中有效。

请只在 IBM 支持中心的指导下使用该命令。

## 标准输出

当指定 **-h** 标志时，将该命令的用法语句写入标准输出。所有详细消息都写到标准输出中。

## 标准错误

必要时此脚本将错误消息写入标准错误。

## 示例

1. 要将拓扑服务子系统添加到 SRC 中，请输入：

```
topsvcsctrl -a
```

2. 要启动拓扑服务子系统，请输入：

```
topsvcsctrl -s
```

3. 要停止拓扑服务子系统，请输入：

```
topsvcsctrl -k
```

4. 要从 SRC 中删除拓扑服务子系统，请输入：

```
topsvcsctrl -d
```

5. 要清除拓扑服务子系统，请输入：

```
topsvcsctrl -c
```

6. 要打开拓扑服务守护程序的跟踪功能，请输入：

```
topsvcsctrl -t
```

7. 要关闭拓扑服务守护程序的跟踪功能，请输入：

```
topsvcsctrl -o
```

## 位置

`/usr/sbin/rsct/bin/topsvcsctrl` 包含 `topsvcsctrl` 脚本

## 文件

`/var/ha/log/topsvcs.cluster_name`

包含名为 `cluster_name` 的集群上的 `hatsd` 守护程序的记录。

## 相关信息

命令：`lssrc`、`startsrc`、`stopsrc`

脚本：`topsvcs`

---

## touch 命令

### 用途

更新文件的访问和修改时间。

### 语法

```
touch [ -a ] [ -c ] [ -m ] [ -f ] [ -r RefFile ] [ Time | -t Time ] { File ... | Directory ... }
```

### 描述

`touch` 命令更新由 `Directory` 参数指定的每个目录下的由 `File` 参数指定的每个文件的访问和修改时间。如果没有指定 `Time` 变量值，`touch` 命令就使用当前时间。如果指定了一个不存在的文件，`touch` 命令就创建此文件，除非指定了 `-c` 标志。

`touch` 命令的返回码是时间没有被成功修改的文件数目（包括不存在的文件和没有创建的文件）。

注：任何超出 2038 年（包含2038年）的日期都是无效的。

### 标志

|                         |                                                                        |
|-------------------------|------------------------------------------------------------------------|
| <code>-a</code>         | 更改由 <code>File</code> 变量指定的文件的访问时间。不要更改修改时间，除非也指定了 <code>-m</code> 标志。 |
| <code>-c</code>         | 如果文件不存在，则不要进行创建。没有写任何有关此条件的诊断消息。                                       |
| <code>-f</code>         | 尝试强制 <code>touch</code> 运行，而不管文件的读和写许可权。                               |
| <code>-m</code>         | 更改 <code>File</code> 的修改时间。不要更改访问时间，除非也指定了 <code>-m</code> 标志。         |
| <code>-r RefFile</code> | 使用由 <code>RefFile</code> 变量指定的文件的相应时间，而不用当前时间。                         |

*Time* 以 *MMDDhhmm*[*YY*] 的格式指定新时间戳记的日期和时间, 其中:

- MM* 指定一年的哪一月 (从 01 到 12)。
- DD* 指定一月的哪一天 (从 01 到 31)。
- hh* 指定一天中的哪一小时 (从 00 到 23)。
- mm* 指定一小时的哪一分钟 (从 00 到 59)。
- YY* 指定年份的后两位数字。如果 *YY* 变量没有被指定, 缺省值为当前年份。

**-t** *Time* 使用指定时间而不是当前时间。 *Time* 变量以十进制形式 *[[CC]YY]MMDDhhmm[.SS]* 指定, 其中:

- CC* 指定年份的前两位数字。
- YY* 指定年份的后两位数字。
- MM* 指定一年的哪一月 (从 01 到 12)。
- DD* 指定一月的哪一天 (从 01 到 31)。
- hh* 指定一天中的哪一小时 (从 00 到 23)。
- mm* 指定一小时的哪一分钟 (从 00 到 59)。
- SS* 指定一分钟的哪一秒 (从 00 到 59)。

注:

1. **touch** 命令调用 **utime ()** 子例程来更改所涉及文件的修改和访问时间。当没有真正拥有该文件, 即使对文件有写许可权, 使用标志时也可能使 **touch** 命令失败。
2. 当使用 **touch** 命令时, 如果接收到错误消息, 不要指定完整路径名 **/usr/bin/touch**。

## 退出状态

该命令返回以下退出值:

- 0** 命令成功执行。所有请求的更改已完成。
- >0** 发生错误。

## 示例

1. 要更新文件的访问和修改时间, 请输入:

```
touch program.c
```

这会将 `program.c` 文件的上次访问和修改时间设置为当前日期和时间。如果 `program.c` 文件不存在, **touch** 命令就创建一个同名的空文件。

2. 要避免创建新文件, 请输入:

```
touch -c program.c
```

3. 要仅更新修改时间, 请输入:

```
touch -m *.o
```

这会更新当前目录下的以 `.o` 扩展名结尾的文件的上次修改时间 (不更新访问时间)。 **touch** 命令经常以此方式使用, 来改变 **make** 命令的结果。

4. 要明确设置访问和修改时间, 请输入:

```
touch -c -t 02171425 program.c
```

这会将访问和修改时间设置为当前年份的 2 月 17 日的 14:25 (2:25 p.m.)。

5. 要使用另一文件的时间戳记而不用当前时间, 请输入:

```
touch -r file1 program.c
```

这会将 `program.c` 文件的时间戳记设置成与 `file1` 文件一样。

6. 要使用指定时间而不是当前时间处理文件, 请输入:

```
touch -t 198503030303.55 program.c
```

这会将 `program.c` 文件的时间戳记设置为 1985 年 3 月 3 日上午 3:03:55。

## 文件

`/usr/bin/touch` 包含 `touch` 命令。

## 相关信息

`date` 命令、`locale` 命令。

`utime` 子例程。

《操作系统与设备管理》中的『目录』描述了文件系统中目录的结构和特征。

《操作系统与设备管理》中的『文件』描述了文件、文件类型以及如何命名文件。

《操作系统与设备管理》中的『文件和目录访问方式』介绍了文件的所有权和文件与目录的访问权限。

《AIX 5L V5.3 通用编程概念: 编写并调试程序》中的『了解文件类型』介绍了控制文件的命令。

---

## tprof 命令

### 用途

报告 CPU 使用情况。

### 语法

```
tprof [ -c ] [ -C { all | CPUList } ] [ -d ] -D [ -e ] [ [ -E [ mode ] ] [ -f frequency ] ] [ -F ] [-l] [ -j ] [ -k ] [ -l ] [-L ObjectList] [ -m ObjectsList ] [ -M SourcePathList ] [-N] [ -p ProcessList ] [ -P { all | PIDsList } ] [ -s ] [ -S SearchPathList ] [ -t ] [ -T BufferSize ] [ -u ] [ -v ] [ -V VerboseFileName ] { { -r RootString } | { [ -A { all | CPUList } ] [ -r RootString ] -x Program } }
```

注:

1. 除路径列表外, 所有的列表类型输入用逗号分隔, 路径列表用冒号分隔。
2. 当此命令以实时方式运行时, 每个 CPU 的概要分析方式自动禁用。
3. 如果每个 CPU 概要分析被打开, 则其微概要分析自动禁用。
4. 如果指定 `-x` 标志不带 `-A` 标志, 则 `tprof` 以实时方式运行。
5. 如果指定 `-x` 标志带 `-A` 标志, 则 `tprof` 以自动离线方式运行。
6. 如果 `-x` 标志省略, 则 `tprof` 以后处理方式或人工离线方式运行, 这取决于有无 `cooked` 文件和 `-F` 标志。

## 描述

**tprof** 命令报告单个程序和整个系统的 CPU 使用情况。对任何使用可能是 CPU 受限的 JAVA、C、C++ 或 FORTRAN 程序的人，和想了解程序的哪些部分大量占用 CPU 的人，该命令是一个有用的工具。

**tprof** 命令负责分配对象代码、进程、线程、子例程（用户方式、内核方式以及共享库），甚至是程序的源代码行或单个指令使用 CPU 的时间。给予例程分配使用 CPU 时间被称为概要分析，而给源程序代码行分配使用 CPU 时间被称为微概要分析。

对于子例程级概要分析，不用修改可执行程序即可运行 **tprof** 命令，就是说不必使用专用编译器标志进行重新编译。除非还除去了跟踪回退表，否则即使可执行程序已卸载也还是如此运行。但是，必需进行重新编译才能获得微概要分析文件，除非列表文件已经可用。要在一个程序上执行微概要分析，要么用 **-g** 来编译该程序且源文件应可访问 **tprof**，要么用 **-qlist** 来编译该程序，且对象列表文件和源文件或者只有对象列表文件对于 **tprof** 应该是可访问的。要充分利用 **tprof** 微概要分析能力，最好既提供 **.lst** 列表文件又提供源文件。

**tprof** 命令可运行于以下方式：

1. 实时或在线方式
2. 人工离线方式
3. 自动离线方式
4. 后处理方式

如果指定 **-x** 标志但未指定 **-A** 标志，则 **tprof** 以实时方式运行。在实时方式下，**tprof** 启动后台的 **AIX trace** 实用程序，并处理生成的跟踪数据。当被概要分析的程序结束时，**tprof** 收集符号名称信息并生成 **tprof** 报告。

注：此方式不允许对每个 CPU 进行概要分析。

如果指定 **-A** 标志带 **-x** 标志，则 **tprof** 以自动离线方式运行。在此方式下，**tprof** 启动 **AIX trace** 实用程序，并将跟踪数据记录到文件中。一旦跟踪数据收集完成，就收集符号名信息，**tprof** 打开跟踪日志文件并处理数据以生成报告。在此方式下，除了 **tprof** 报告文件，**tprof** 生成以下文件：

1. *RootString.syms*
2. *RootString.trc[-cpuid]*

注：**tprof** 使用的所有输入文件和报告文件都被命名为 *RootString.suffix*，其中 *RootString* 要么用 **-r** 指定，要么是用 **-x** 指定的程序名。

注：在实时方式和自动离线方式下，在进行概要分析的程序数据区的 **ulimit** 值被设置为 **unlimited**。

注：在自动离线方式下，指定 **-N** 标志将源行信息收集到生成的 **RootString.syms** 文件。并指定 **-I** 标志将二进制指令信息收集到生成的 **RootString.syms** 文件。

任何时候 **tprof** 命令都可重新处理这些文件来生成概要分析报告。这被称为人工离线方式。**RootString.syms** 文件包含符号名称信息，与 **gensyms** 命令的输出相似。*RootString.trc [-cpuid]* 文件是跟踪日志文件。当每个 CPU 跟踪功能打开时，**-cpuid** 被添加到名称上。在此情况下，每个文件仅包含来自一个 CPU 的跟踪数据。

注：如果 **-c** 和 **-A** 一起使用，则不生成这两个文件。而是创建了以下两个文件：

1. *RootString.csyms*
2. *RootString.ctrc[ -cpuid ]*

这些文件是 *cooked*，即它们是正常跟踪和名称文件的一个预处理版本。**tprof** 后处理 *cooked* 文件更快。



如果 **-A** 标志和 **-x** 标志都未指定, 则 **tprof** 要么以人工离线方式运行, 要么以后处理方式运行。要 **tprof** 以后处理方式运行, 以下文件必须可用:

1. *RootString.csyms*
2. *RootString.ctrc* [ **-cpuid** ]

当 **tprof** 带 **-c** 标志 (以除了后处理方式以外的任何方式) 运行时, 就生成了这些文件。要 **tprof** 以人工离线方式运行, 以下文件必须可用:

1. *RootString.syms*
2. *RootString.trc* [ **-cpuid** ]

这些文件是用户手工使用 **gensyms** 命令和 AIX 跟踪工具生成的, 或者是在自动离线方式下不使用 **-c** 标志运行 **tprof** 而生成的。

注: **tprof** 总是首先查找 *RootString.csyms* 和 *RootString.ctrc*[**-cpuid**]。仅当这些文件不可用时, 才查找 *RootString.syms* 和 *RootString.trc*[**-cpuid**]。要阻止 **tprof** 查找 *rootstring.csyms* 和 *RootString.ctrc*[**-cpuid**], 也就是说, 强制使用人工离线方式, 请使用 **-F** 标志。

注: 在自动离线方式下, 指定 **-N** 标志将源行信息收集到生成的 *RootString.syms* 文件。并指定 **-I** 标志将二进制指令信息收集到生成的 *RootString.syms* 文件。

**tprof** 生成一个名为 *RootString.prof* 的 **tprof** 报告文件, 该文件包含进程、线程、对象文件和子例程级别概要分析报告。该文件可包含以下部分和子节:

1. 摘要报告部分
  - a. 按进程名的 CPU 使用情况摘要
  - b. 按线程 (tid) 的 CPU 使用情况摘要
2. 全局 (关于系统中所有进程的执行情况) 概要文件部分
  - a. 用户方式例程的 CPU 使用情况
  - b. 内核例程的 CPU 使用情况, 包含内核方式调用的 milicode 例程
  - c. 内核扩展的 CPU 使用情况摘要
  - d. 每个内核扩展的子例程的 CPU 使用情况
  - e. 共享库的 CPU 使用情况摘要, 包含用户方式中调用的 milicode 例程
  - f. 每个共享库的子例程的 CPU 使用情况
  - g. 每个 JAVA 类的 CPU 使用情况
  - h. 每个 JAVA 类的每个 JAVA 方法的 CPU 使用情况
3. 进程和线程级概要分析部分 (每个进程或线程为一个部分)
  - a. 适用于本进程 / 线程的用户方式例程的 CPU 使用情况
  - b. 本进程 / 线程的内核例程的 CPU 使用情况, 包含内核方式调用的 milicode 例程
  - c. 本进程 / 线程的内核扩展的 CPU 使用情况摘要
  - d. 本进程 / 线程的每个内核扩展的子例程的 CPU 使用情况
  - e. 本进程 / 线程的共享库的 CPU 使用情况摘要, 包含以用户方式调用的 milicode 例程
  - f. 本进程 / 线程的每个共享库的子例程的 CPU 使用情况
  - g. 本进程 / 线程的每个 JAVA 类的 CPU 使用情况
  - h. 本进程 / 线程的每个 JAVA 类的 JAVA 方法的 CPU 使用情况

摘要报告部分总是在 *RootString.prof* 报告文件中。基于概要分析标志, 全局概要分析部分的各个子节可被打开和关闭:

- **-u** 打开子节 a
- **-k** 打开子节 b

- **-e** 打开子节 c 和 d
- **-s** 打开子节 e 和 f
- **-j** 打开子节 g 和 h

进程和线程级概要分析部分是为用 **-p**、**-P** 和 **-t** 标志选择的进程和线程创建的。存在于每个进程的每个线程的每个部分中的子节等同于存在于全局部分中的子节，使用概要分析标志 (**-u**、**-s**、**-k**、**-e**、**-j**) 来选择这些子节。

当用 **-C** 标志调用时，**tprof** 也有选择地生成每个 CPU 的概要分析报告（每个 CPU 生成一个概要分析报告）。生成的 **tprof** 报告有相同的结构，用约定的规则给它命名：*RootString.prof[-cpuid]*。

如果指定 **-m**，则 **tprof** 生成微概要分析报告。报告使用以下命名约定：*RootString.source.mprof*，其中 *source* 是源文件的基本名。如果有多个源文件有相同的基本名，则一个唯一标识它们的号码附加在报告文件名后面，例如，*RootString.Filename.c.mprof-1*。微概要分析报告含如下信息：

1. 带注释的源文件的全路径名。
2. 与概要文件样本符合的、含有源文件的所有行号的热行概要文件部分，它按 CPU 的使用情况进行排序。对于每个源行，用一行来报告所有进程所用时间的百分比，其后跟有按单个进程细分的其他行。
3. 一个该源文件中每个功能的源行概要分析部分，包括 CPU 使用情况。该部分包含源行号、CPU 的使用情况和源代码。如果该源文件的 *.lst* 文件可访问 **tprof**，则将来自 *.lst* 文件的指令行和来自源文件的源行集结在一起并适当分配 CPU 的使用。对于每个源文件，这将由指令提供中断。

如果一个源文件不存在，但有一个 *.lst* 文件存在，**tprof** 仅显示基于源文件行和来自 *.lst* 文件的指令的 CPU 使用情况。

如果这些文件都不存在，但源文件用 **-g** 编译过，则 **tprof** 可检索源文件行号，并生成一个相似的报告，但源文件代码列丢失。

**注：**如果请求每个 CPU 进行概要分析，则微概要分析自动禁用。如果 *.c* 文件包含在另一个 *.c* 文件中，则 **tprof** 不能报告正确的源文件行信息。**tprof** 不能对 JAVA 类或方法进行微概要分析。

**注：**如果指定了 **-m**，则将自动指定 **-N** 以自动离线方式将源行信息收集到符号文件中。

如果 **-Z** 标志与 **-m** 标志一起使用，那么每个子例程都会生成一个报告文件。然后将使用以下命名约定：*RootString.source.routine.mprof*，其中 *routine* 是列在源文件中的子例程之一的名称。此外，还将创建一个名为 *RootString.source.HOT\_LINES.mprof* 的文件，该文件包含以上描述的热行概要分析信息。

如果指定了 **-L** 标志，**tprof** 命令会生成带注释的列表文件。这些文件使用以下命名约定：*RootString.source.alst*，其中 *source* 是源文件的基本名称。如果有多个源文件具有相同的基本名称，那么将会在报告文件名之后附加一个唯一标识这些文件的号码。例如，*RootString.Filename.c.alst-1*。如果 **-Z** 标志与 **-L** 标志一起使用，那么每个子例程都会生成一个报告文件。然后将使用以下命名约定：*RootString.source.routine.alst*，其中 *routine* 是列在源文件中的子例程之一的名称。

如果指定了 **-N** 标志或 **-I** 标志，则当使用 JPA (**-x java -Xrunjpa**) 概要分析一个 java 程序时，如果将相应的参数添加到 **-Xrunjpa** 标志，则可以收集 JIT 源行号和指令：

- **source=1** 打开 JIT 源行收集（需要 IBM JRE 1.5.0）。
- **instructions=1** 打开 JIT 指令收集。

以下限制适用于运行 **tprof** 命令的非 root 用户：

1. **tprof** 将不能验证正在运行的内核与 */unix* 文件是否相同。这就意味着即使显示警告消息，在多数情况下，正在运行的内核与 */unix* 也是相同的，因此数据应该准确。

2. 当由非 root 用户运行 **gensyms** 命令时, 将给出与限制 #1 (以上所述) 相同的警告, 并标记 **gensyms** 文件。如果在离线方式下运行 **tprof**, 用 **gensyms** 命令创建的文件将根据未验证内核来标记 **tprof**。
3. **tprof** 将无法打开和读取未设置读许可权的文件上的符号。一些专用、共享库不具有读许可权, 而一些内核扩展不可读。

## 基于时间的概要分析与基于事件的概要分析

在缺省情况下, **tprof** 是基于时间的概要分析并且是由衰减器中断驱动的。概要分析的另一种方式是基于事件的概要分析, 在这种方式下, 中断是由基于软件的事件或通过“性能监视器”事件驱动的。对于基于事件的概要分析, 采样频率和概要分析事件都会随着命令行的不同而不同。

**-E** 标志启用基于事件的概要分析。**-E** 标志参数是 4 个基于软件的事件 (EMULATION、ALIGNMENT、ISLBMISS 和 DSLBMISS) 中的一个, 或者是“性能监视器”事件 (PM\_\*)。在缺省情况下, 概要分析事件是处理器周期。所有“性能监视器”事件前面都有 PM\_ (例如, PM\_CYC 表示处理器周期, PM\_INST\_CMPL 表示指令已完成)。**pmlist** 可列出处理器支持的所有“性能监视器”事件。

对于基于事件的概要分析, **-f** 标志可更改采样频率。对于基于软件的事件和处理器周期, 受支持的频率范围是 1 至 500 毫秒, 缺省值为 10 毫秒。对于所有其他的“性能监视器”事件, 其范围是 10,000 至发生事件的 MAXINT 次数, 缺省值为 10,000 个事件。

**.prof** 文件中已添加更多信息以反映处理器名称、概要分析事件和采样频率。

## Java™ 应用程序概要分析

对于概要文件 JAVA 应用程序, 必须指定 **-j** 标志, 而且必须以 **-Xrunjpa java** 命令行选项启动此应用程序。如果选择了此选项, 当新类和方法装入内存时, JVM 将自动调用 **jpa** 库。库将依次收集地址来命名 **tmp/JavaPID.syms** 文件中的方法和类的映射信息, 其中 **PID** 是运行 Java 虚拟机的进程的进程标识。**tprof** 命令将自动查找此目录来获取这种文件。

在以自动离线方式运行或选择 **cooking** 选项时, **tprof** 命令会将 **JavaPID.syms** 文件中包含的信息复制到 **RootString.syms** 或 **RootString.csyms** 文件。然后可以删除 **/tmp** 中的相应文件。**tprof** 命令用户应该将目录内容保持最新。当停止与特定 **JavaPID.syms** 相符的 JVM 时, 应该删除此文件。

## 概要分析的准确性

解析 CPU 活动的程度取决于获得的样本数目以及热点的程度。当一个带有几个热点的程序可用相对较少的样本进行概要分析时, 程序的较少执行的部分在概要分析报告中不可见, 除非获得更多的样本。在用户程序运行少于一分钟的情况下, 要在评估中获得高可信度的分辨率可能不够。

一个简单的方案是重复执行用户程序或脚本直到达到所需的分辨率。程序运行时间越长, 概要分析分辨率就越高。如果您怀疑概要分析的准确度, 运行 **tprof** 命令几次即可, 然后比较所产生的概要分析文件。

## tprof 对 prof 和 gprof 之间的比较

在这三个命令之间最重要区别是 **tprof** 收集数据而不会影响被概要分析的程序的执行时间, 而且除了生成微概要分析报告以外, 无需重新编译就可以处理已实现优化和分解的二进制文件。**gprof** 和 **prof** 都没有微概要分析的能力, 也不能处理已优化的二进制文件, 但它们的确需要特殊的编译标志, 并且会在执行期间导致速度下降 (可能会很明显)。**prof** 不能处理已分解的二进制文件。

**prof** 和 **gprof** 工具在许多 UNIX 系统 (包括此操作系统) 上都是受支持的标准概要分析工具。**prof** 和 **gprof** 提供了子程序概要分析功能, 以及每个子程序被调用的次数的精确计数。**gprof** 命令也提供了一个非常有用的调用图, 显示每个子程序被特定父程序调用的次数和每个子程序调用其子的次数。**tprof** 命令既不提供子程序调用计数, 也不提供调用图信息。

像 **tprof** 命令一样，**prof** 和 **gprof** 命令都通过采样用户程序的程序计数器来获取每个子程序的 CPU 消耗量估计。

**tprof** 收集整个系统的 CPU 的使用信息，而 **prof** 和 **gprof** 仅收集单个程序和仅在用户方式下所花时间的概要分析信息。**tprof** 也提供已概要分析的用户程序执行期间所有活动进程的摘要以及完全支持库和内核方式概要分析的摘要。

**tprof** 支持 JAVA 应用程序的概要分析，而 **prof** 和 **gprof** 不支持。

## 标志

- A** { **all** | *CPUList* } 打开自动离线方式。无参数则关闭每个 CPU 跟踪程序。**all** 启用所有 CPU 跟踪程序。*CPUList* 是一个用逗号分隔的要被跟踪的 CPU 标识列表。
- c** 打开 cooked 文件的生成。
- C** **all** | *CPUList* 打开每个 CPU 概要分析功能。指定 **all** 来生成所有 CPU 的概要分析报告。如果用列表 *CPUList*，则 CPU 的数目应用逗号隔开（如，0,1,2）。  
**注：** 仅当每个 CPU 跟踪程序要么在自动离线方式下打开，要么在人工离线方式下使用的时候，每个 CPU 概要分析才可能进行。而在在线方式下，则完全不可能。
- d** 打开延时跟踪方式，该方式延时数据收集直到 **trcon** 被调用为止。
- D** 打开详细概要分析功能，它通过每个子例程中的偏移指令显示 CPU 使用情况。
- e** 打开内核扩展概要分析功能。
- E** [ *mode* ] 启用基于事件的概要分析。可能的方式为：  
**PM\_event**  
将硬件事件指定到概要文件。如果没有对 **-E** 标志指定方式，则缺省事件是处理器周期（**PM\_CYC**）。
- EMULATION**  
启用模拟概要分析方式。
- ALIGNMENT**  
启用对齐概要分析方式。
- ISLBMISS**  
启用指令段后备缓冲区不命中概要分析方式。
- DSLBMISS**  
启用数据段后备缓冲区不命中概要分析方式。
- f** *frequency* 指定采样频率。对于处理器周期和 **EMULATION**、**ALIGNMENT**、**ISLBMISS** 以及 **DSLBMISS** 事件，采样频率可以从 1 至 500 毫秒，对于其他“性能监视器”事件，采样频率可以从 10000 至事件发生的 **MAXINT** 次数。
- F** 覆盖 cooked 文件（如果存在）。如果没有用 **-x** 标志使用的话，则强制使用人工离线方式。
- I** 打开二进制指令收集。  
**注：** **-I** 以自动离线方式生成符号文件或 cooked 符号文件时激活二进制指令收集。然而，在人工离线方式下，**-I** 不会影响报告文件。
- j** 打开 JAVA 类和方法概要分析功能。
- k** 启用内核概要分析功能。
- l** 启用长文件名报告。缺省情况下，如果子程序、程序和源文件名不适合概要分析报告中的可用空间，则 **tprof** 截断这些名称。该标志禁用截取。

- L ObjectList** 为由逗号分隔的列表 *Objectlist* 指定的对象启用列表注释功能。可执行程序以及共享库可以给它们的列表文件加上注释。请为库指定归档名。
- 注:**
1. 要启用程序的列表注释功能，必须打开用户方式概要分析功能 (**-u**)。
  2. 要启用共享库的列表注释功能，必须打开共享库概要分析功能 (**-s**)。
  3. 要注释由 IPA 编译生成的列表，请指定 *a.lst* 作为 *ObjectList*。
- m ObjectsList** 启用由逗号分隔的列表 *Objectlist* 指定的对象的微概要分析功能。可执行文件、共享库以及内核扩展都可进行微概要分析。给库和内核扩展指定归档名。
- 注:**
1. 要启用程序的微概要分析功能，必须打开用户方式概要分析功能 (**-u**)。
  2. 要启用共享库的微概要分析功能，必须打开共享库概要分析功能 (**-s**)。
  3. 要启用内核扩展例程微概要分析功能，必须打开内核扩展例程概要分析功能 (**-e**)。
- M PathList** 指定源路径列表。*PathList* 是一个用冒号分隔的路径列表，搜索这些路径以查找微概要分析和列表注释所需的源文件和 *.lst* 文件。
- 注:** 缺省情况下，源路径列表是对象搜索路径列表。
- N** 打开源行号信息收集。
- 注:** **-N** 以自动离线方式生成符号文件或 *cooked* 符号文件时激活源行信息收集。然而，在人工离线方式下，**-N** 不影响报告文件。
- p ProcessList** 启用在 *ProcessList* 中指定的进程名的进程级概要分析功能。*ProcessList* 是一个用逗号分隔的进程名列表
- 注:** 仅当至少有一个概要分析方式 (**-u**、**-s**、**-k**、**-e** 或 **-j**) 被打开，进程级概要分析才启用。
- P { all | PIDList }** 启用遇到的所有进程或 *PIDList* 指定进程的进程级概要分析功能。*PIDList* 是一个用逗号分隔的进程标识列表。
- 注:** 仅当至少有一个概要分析方式 (**-u**、**-s**、**-k**、**-e** 或 **-j**) 被打开，进程级概要分析才启用。
- r RootString** 指定 *RootString*。所有 **tprof** 输入和报告文件都有 *RootString.suffix* 格式的名称。
- 注:** 如果未指定 **-r**，则 *RootString* 缺省为 **-x** 标志指定的程序名。
- R** 指定 **tprof** 命令应该使用由 *PURR* 增量值加权的样本来计算百分比。这是在同时多线程或微分区环境中运行时的首选方式。
- 注:** **-R** 标志不能与 **-z** 标志或 **-Z** 标志一起使用。
- s** 启用共享库概要分析功能。
- S PathList** 指定对象搜索 *PathList*。*PathList* 是一个用冒号隔开的路径列表，搜索该列表以查找可执行文件、共享库和内核扩展。
- 注:** 缺省对象搜索 *PathList* 是环境路径列表 (*\$PATH*)。
- t** 启用线程级概要分析功能。
- 注:** 如果 **-p** 或 **-P** 没有带 **-t** 标志一起指定，则 **-t** 等价于 **-P all -t**。否则，为选定的进程启用线程级报告。仅当至少有一个概要分析方式 (**-u**、**-s**、**-k**、**-e**、**-j**) 启用时，线程级概要分析才启用。
- T BufferSize** 指定跟踪 *BufferSize*。
- 注:** 该标志表示仅在实时方式或自动离线方式下有效。
- u** 启用户方式概要分析功能。
- v** 启用详细方式。
- V File** 将详细输出存储在指定的 *File* 中。
- x Program** 指定将由 **tprof** 执行的程序。当 *Program* 完成或用 **trccoff** 或 **trcstop** 手工停止 **trace** 时，停止数据收集
- 注:** **-x** 标志必须是在 **tprof** 中指定的标志列表中的最后一个标志。
- z** 用先前版本的 **tprof** 来启用兼容方式。缺省情况下，CPU 使用情况仅用百分比来报告。当使用 **-z** 时，**tprof** 也报告滴答数。该标志还在子例程报告中添加“地址”栏和“字节”栏。
- Z** 将报告切换为使用记号而不是百分比（与 **-z** 标志相同），并将作过注释的列表（当与 **-L** 标志一起使用时）和作过注释的源文件（当与 **-m** 标志一起使用时）按每个子例程一个文件，分割为多个文件。

## 示例

### 1. 基本全局程序和线程级摘要

```
$ tprof -x sleep 10
Mon May 20 00:39:26 2002 System: AIX 5.2 Node: dreaming Machine: 000671894C00
Starting Command sleep 10
stopping trace collection.
Generating sleep.prof
```

产生的 **sleep.prof** 文件仅包含摘要报告部分。

### 2. 带有所有选项的全局概要分析

```
$ tprof -skeuj -x sleep 10
Mon May 20 00:39:26 2002
System: AIX 5.2 Node: dreaming Machine: 000671894C00
Starting Command sleep 10
stopping trace collection.
Generating sleep.prof
```

产生的 **sleep.prof** 文件包含摘要报告部分和全局概要信息部分。

### 3. 单个进程级概要分析

```
$ tprof -u -p workload -x workload
Mon May 20 00:39:26 2002
System: AIX 5.2 Node: dreaming Machine: 000671894C00
Starting Command workload stopping trace collection.
Generating workload.prof
```

产生的 **workload.prof** 文件包含摘要报告和全局用户方式概要分析部分，以及仅包含一个用户方式概要分析子节的进程 ‘workload’ 的一个进程级概要分析部分。

### 4. 多个进程级概要分析

```
$ tprof -se -p send,receive -x startall
Mon May 20 00:39:26 2002
System: AIX 5.2 Node: dreaming Machine: 000671894C00
Starting Command startall
stopping trace collection.
Generating startall.prof
```

生成的 **startall.prof** 文件包含摘要报告、全局共享库方式概要信息、全局内核扩展概要信息部分以及两个进程级概要信息部分：一个用于进程 “send”，一个用于进程 “receive”。每个进程级部分包含两个子部分：一个为共享库概要分析信息，一个为内核扩展概要分析信息。

### 5. 微概要分析和列表注释

```
$ tprof -m ./tcalc -L ./tcalc -u -x ./tcalc
Mon May 20 00:47:09 2002
System: AIX 5.2 Node: dreaming Machine: 000671894C00
Starting Command ./tcalc
stopping trace collection.
Generating tcalc.prof
Generating tcalc.tcalc.c.mprof
Generating tcalc.tcalc.c.alst
```

产生的 **tcalc.prof** 文件包含摘要报告和全局用户方式概要分析部分。生成的 **tcalc.tcalc.c.mprof** 和 **tcalc.tcalc.c.alst** 文件包含微概要分析报告和带注释的列表。

### 6. 对于处理器周期所进行的基于事件的概要分析，要每 100 毫秒就进行一次采样，请输入：

```
tprof -E -f 100 -Askex sleep 10
```

输出类似以下显示:

```
Starting Command sleep 10
stopping trace collection.
Tue Apr 26 14:44:02 2005
System: AIX 5.3 Node: bigdomino Machine: 00C0046A4C00
Generating sleep.trc
Generating sleep.prof
Generating sleep.syms
```

7. 对于已完成的指令所进行的基于事件的概要分析, 要每完成 20,000 条指令就进行一次采样, 请输入:

```
tprof -E PM_INST_CMPL -f 20000 -Askex sleep 10
```

输出类似以下显示:

```
Starting Command sleep 10
stopping trace collection.
Tue Apr 26 14:42:44 2005
System: AIX 5.3 Node: bigdomino Machine: 00C0046A4C00
Generating sleep.trc
Generating sleep.prof
Generating sleep.syms
```

8. 对于仿真中断所进行的基于事件的概要分析, 要每 10,000 个事件进行一次采样, 请输入:

```
tprof -E EMULATION -Askex sleep 10
```

输出类似以下显示:

```
Starting Command sleep 10
stopping trace collection.
Tue Apr 26 14:41:44 2005
System: AIX 5.3 Node: bigdomino Machine: 00C0046A4C00
Generating sleep.trc
Generating sleep.prof
Generating sleep.syms
```

9. 自动离线方式

```
tprof -c -A all -x sleep 10
Starting Command sleep 10
stopping trace collection.
Mon May 20 00:52:52 2002
System: AIX 5.2 Node: dreaming Machine: 000671894C00
Generating sleep.ctrc
Generating sleep.csyms
Generating sleep.prof
```

产生的 **sleep.prof** 文件只有一个摘要报告部分, 而两个 **cooked** 文件准备进行重新后处理。

10. 自动离线方式启用源行收集。

```
$ tprof -A -N -x sleep 10
Starting Command sleep 10
stopping trace collection.
Wed Feb 8 15:12:41 2006
System: AIX 5.3 Node: aixperformance Machine: 000F9F3D4C00
Generating sleep.trc
Generating sleep.prof
Generating sleep.syms
```

结果 **sleep.prof** 文件仅包含摘要报告部分, 而 **sleep.syms** 包含源行信息。

11. 自动离线方式启用源行和指令收集。

```
$ tprof -A -N -I -r RootString -x sleep 10
Starting Command sleep 10
stopping trace collection.
Wed Feb 8 15:16:37 2006
```

```
System: AIX 5.3 Node: aixperformance Machine: 000F9F3D4C00
Generating RootString.trc
Generating RootString.prof
Generating RootString.syms
```

生成 **rootstring.prof** 结果文件，而 **rootstring.syms** 包含源行信息和二进制指令。

12. 实时方式启用 java 源行和指令收集。

```
$ tprof -N -I -x java -Xrunjpa:source=1,instructions=1 HelloAIX
Thu Feb 9 13:30:38 2006
System: AIX 5.3 Node: perftdev Machine: 00CEBB4A4C00
Starting Command java -Xrunvpjpa:source=1,instructions=1 HelloAIX
Hello AIX!
stopping trace collection.
Generating java.prof
```

生成 **java.prof** 结果文件，并且它包含 JIT 源行信息和 JIT 指令。

## 消息

如果您的系统显示如下消息：

```
/dev/systrace: 设备忙或 trcon: TRCON: 无此设备
```

这表示 **trace** 工具正在使用。输入 **trcstop** 停止跟踪后，停止程序并再试一次。

## 相关信息

**gensyms** 命令、**gprof** 命令、**prof** 命令、**stripnm** 命令、**trace** 守护程序和 **tprof2xml** 命令。

---

## tput 命令

### 用途

查询 **terminfo** 数据库以获取与终端相关的信息。

### 语法

#### 用于输出终端信息

```
tput [ -T Type ] [ CapabilityName {clear, init, longname, reset} [ Parameters... ] ]
```

#### 用于使用标准输入来处理多种能力

```
tput [ -S ]
```

### 描述

**tput** 命令使用 **terminfo** 数据库以使终端相关信息适用于 shell。如果属性 *CapabilityName* 的类型为 *string*，则 **tput** 命令输出一个字符串。如果该属性的类型为 *integer*，则输出字符串是一个整数。如果属性的类型为 *Boolean*，则 **tput** 命令设置出口值（TRUE 为 0，FAISE 为 1），不再产生其他输出。

### XTERM 描述限制

AIX V4 上的 DEC.TI 文件中的 **xterm** 终端描述通过使用 **SGR** 属性提供下划线方式。**SMUL** 和 **RMUL** 属性目前在 AIX V4 上的 **XTERM** 终端描述中尚无定义。使用名为 **SGR** 的更通用的能力。

```
tput sgr x y
```



其中  $x$  是 1 或 0, 分别为打开或关闭突出方式, 而  $y$  是 1 或 0, 分别为打开或关闭下划线方式。有关 SGR 能力的更详细信息, 请参阅文章 “**terminfo** 文件格式”

```
tput sgr 0 1    关闭突出方式; 打开下划线方式。
tput sgr 0 0    关闭突出方式; 关闭下划线方式。
tput sgr 1 1    打开突出方式; 打开下划线方式。
tput sgr 1 0    打开突出方式; 关闭下划线方式。
```

## 标志

除了能力名称之外, 还支持以下字符串作为 **tput** 子例程参数。

|                 |                                                                                  |
|-----------------|----------------------------------------------------------------------------------|
| <b>clear</b>    | 显示清除屏幕顺序 (这也是一个能力名称)。                                                            |
| <b>init</b>     | 显示以依赖于实现的方式对用户终端进行初始化的顺序。                                                        |
| <b>reset</b>    | 显示将以依赖于实现的方式对用户终端进行重新设置的顺序。                                                      |
| <b>longname</b> | 显示长名称和指定的终端 (如果未指定终端, 则显示当前终端)。                                                  |
| <b>-S</b>       | 使用标准输入。这可以使 <b>tput</b> 处理多种能力。当使用 <b>-S</b> 选项时, 不得在命令行中输入能力。输入 <b>^D</b> 标记完成。 |
| <b>-TType</b>   | 指示终端类型。如果没有指定 <b>-T</b> , 则 <b>TERM</b> 环境变量可用于该终端。                              |

## 退出状态

该命令返回以下退出值:

|              |                     |
|--------------|---------------------|
| <b>0</b>     | 已经成功地写入了所请求的字符串。    |
| <b>1</b>     | 未指定。                |
| <b>2</b>     | 用法错误。               |
| <b>3</b>     | 有关指定的终端类型, 无任何可用信息。 |
| <b>4</b>     | 指定的操作数无效。           |
| <b>&gt;4</b> | 发生错误。               |

## 示例

1. 要清除当前终端的屏幕, 请输入:

```
tput clear
```

2. 要显示当前终端的列数, 请输入:

```
tput cols
```

3. 要显示 **aixterm** 终端的列数, 请输入:

```
tput -Taixterm cols
```

4. 要将 **shell** 变量 **bold** 设置为开始突出方式顺序并将 **shell** 变量 **offbold** 设置为结束突出方式顺序, 请输入:

```
bold=`tput smso`
```

```
offbold='tput rmso'
```

输入这些指令后, 可能会出现以下提示符:

```
echo "${bold}Name: ${offbold} \c"
```

5. 要设置出口值以指示当前终端是否是一个硬拷贝终端, 请输入:

```
tput hc
```

6. 要对当前终端进行初始化, 请输入:

```
tput init
```

## 文件

`/usr/share/lib/terminfo/*/*`  
`/usr/include/term.h`

包括终端描述符文件。  
包括定义文件。

## 相关信息

`stty` 命令。

`terminfo` 文件格式。

---

## tr 命令

### 用途

转换字符。

### 语法

```
tr [ -c | -cds | -cs | -C | -Cds | -Cs | -ds | -s ] [ -A ] String1 String2
```

```
tr { -cd | -cs | -Cd | -Cs | -d | -s } [ -A ] String1
```

### 描述

`tr` 命令从标准输入删除或替换字符，并将结果写入标准输出。根据由 `String1` 和 `String2` 变量指定的字符串以及指定的标志，`tr` 命令可执行三种操作。

### 转换字符

如果 `String1` 和 `String2` 两者都已指定，但 `-d` 标志没有指定，则 `tr` 命令就会从标准输入中将 `String1` 中所包含的每一个字符都替换成 `String2` 中相同位置上的字符。

### 使用 `-d` 标志删除字符

如果 `-d` 标志已经指定，则 `tr` 命令就会从标准输入中删除 `String1` 中包含的每一个字符。

### 用 `-s` 标志除去序列

如果 `-s` 标志已经指定，则 `tr` 命令就会除去包含在 `String1` 或 `String2` 中的任何字符串系列中的除第一个字符以外的所有字符。对于包含在 `String1` 中的每一个字符，`tr` 命令会从标准输出中除去除第一个出现的字符以外的所有字符。对于包含在 `String2` 中的每一个字符，`tr` 命令除去标准输出的字符序列中除第一个出现的字符以外的所有字符。

### 表达字符串的特殊序列

`String1` 和 `String2` 变量中所包含的字符串可以使用以下的约定来表示：

`C1-C2`

指定了 `C1` 所指定的字符和 `C2` 所指定的字符之间（包括 `C1` 和 `C2`）进行整理的字符串。`C1` 所指定的字符必须整理放在由 `C2` 所指定的字符之前。

**注：**在使用本方法指定子范围时，当前语言环境对结果有重要影响。如果需要用命令来产生与语言环境无关的一致结果，则应该避免使用子范围。

`[C*Number]`

`Number` 是一个整数，它指定了由 `C` 所指定的字符的重复次数。除非其首位数字是 0，否则 `Number` 一律视为是十进制整数；如果首位数字是 0，则视为八进制整数。

[*C*\*] 用 *C* 指定的字符填写字符串。该选项只用于包含在 *String2* 中的字符串末尾，它强制 *String2* 中的字符串具有与由 *String1* 变量所指定的字符串一样的字符数。\*（星号）后面指定的任何字符都被忽略。

[ :*ClassName*: ] 指定由当前语言环境中的 *ClassName* 所命名的字符类中的所有字符。类名可以是下述名称中的任何一种：

|       |        |
|-------|--------|
| alnum | lower  |
| alpha | print  |
| blank | punct  |
| cntrl | space  |
| digit | upper  |
| graph | xdigit |

除 [:lower:] 和 [:upper:] 转换字符类之外，其他字符类指定的字符都按未指定的顺序放入数组中。由于未定义字符类指定的字符的顺序，仅当目的为将多个字符映射为一个时才使用这些字符。转换字符类除外。

有关字符类的详细情况，请参阅 **ctype** 子例程。

[ =*C*= ] 指定所有的字符具有与 *C* 所指定的字符相同的等价类。

\*Octal*

指定字符，其编码由 *Octal* 所指定的八进制值表示。*Octal* 可以是 1 位、2 位或 3 位八进制整数。空字符可以用 '\0' 表示，并可以像任何其他字符那样进行处理。

\*ControlCharacter*

指定与 *ControlCharacter* 所指定的值相应的控制字符。可以表示以下值：

|    |       |
|----|-------|
| \a | 警告    |
| \b | 退格键   |
| \f | 换页    |
| \n | 换行    |
| \r | 回车    |
| \t | 制表键   |
| \v | 垂直制表键 |

\\ 规定 “\”（反斜杠）就是作反斜杠使用，而无作为转义字符的任何特殊意义。

\[ 指定 “[”（左括号）就作为左括号使用，而无作为特定字符串序列的开始字符的任何特殊意义。

\- 指定 “-”（负号）就作为负号使用，而无作为范围分隔符的任何特殊意义。

如果某个字符在 *String1* 中被指定过多次，则该字符就被转换成 *String2* 中为与 *String1* 中最后出现的字符相对应的字符。

如果由 *String1* 和 *String2* 指定的字符串长度不相同，则 **tr** 命令就会忽略较长一个字符串中的多余字符。

## 标志

**-A** 使用范围和字符类 ASCII 整理顺序、一个字节一个字节地执行所有操作，而不是使用当前语言环境整理顺序。

**-C** 指定 *String1* 值用 *String1* 所指定的字符串的补码替换。*String1* 的补码是当前语言环境的字符集中的所有字符，除了由 *String1* 指定的字符以外。如果指定了 **-A** 和 **-c** 标志都已指定，则与所有 8 位字符代码集合有关的字符将被补足。如果指定了 **-c** 和 **-s** 标志，则 **-s** 标志适用于 *String1* 的补码中的字符。

如果没有指定 **-d** 选项，则由 *String1* 指定的字符的补码将放置到升序排列的数组中（如 **LC\_COLLATE** 的当前设置所定义）。

- c** 指定 *String1* 值用 *String1* 所指定的字符串的补码替换。*String1* 的补码是当前语言环境的字符集中的所有字符，除了由 *String1* 指定的字符以外。如果指定了 **-A** 和 **-c** 标志都已指定，则与所有 8 位字符代码集合有关的字符将被补足。如果指定了 **-c** 和 **-s** 标志，则 **-s** 标志适用于 *String1* 的补码中的字符。
- 如果没有指定 **-d** 选项，则由 *String1* 指定的值的补码将放置到通过二进制值升序排列的数组中。
- d** 从标准输入删除包含在由 *String1* 指定的字符串中的每个字符。
- 注:
  1. 当 **-C** 选项和 **-d** 选项一起指定时，将删除所有除 *String1* 指定的那些字符以外的字符。忽略 *String2* 的内容，除非也指定了 **-s** 选项。
  2. 当 **-c** 选项和 **-d** 选项一起指定时，将删除所有除 *String1* 指定的那些字符以外的字符。忽略 *String2* 的内容，除非也指定了 **-s** 选项。
- s** 在重复字符序列中除去除第一个字符以外的所有字符。将 *String1* 所指定的字符序列在转换之前从标准输入中除去，并将 *String2* 所指定的字符序列从标准输出中除去。
- String1* 指定一个字符串。
- String2* 指定一个字符串。

## 退出状态

该命令返回以下退出值:

- 0** 所有输入处理成功。
- >0** 发生错误。

## 示例

1. 若要将大括号转换为小括号，请输入:

```
tr '{}' '(') < textfile > newfile
```

这便将每个 { (左大括号) 转换成 ( (左小括号)，并将每个 } (右大括号) 转换成 ) (右小括号)。所有其他的字符都保持不变。

2. 若要将大括号转换成方括号，请输入:

```
tr '{}' '\[]' < textfile > newfile
```

这便将每个 { (左大括号) 转换成 [ (左方括号)，并将每个 } (右大括号) 转换成 ] (右方括号)。左方括号必须与一个 "\" (反斜杠) 转义字符一起输入。

3. 若要将小写字符转换成大写，请输入:

```
tr 'a-z' 'A-Z' < textfile > newfile
```

4. 若要创建一个文件中的单词列表，请输入:

```
tr -cs '[:lower:][:upper:]' '[\n*]' < textfile > newfile
```

这便将每一序列的字符 (小、大写字母除外) 都转换成单个换行符。\* (星号) 可以使 **tr** 命令重复换行符足够多次以使第二个字符串与第一个字符串一样长。

5. 若要从某个文件中删除所有空字符，请输入:

```
tr -d '\0' < textfile > newfile
```

6. 若要用单独的换行替换每一序列的一个或多个换行，请输入:

```
tr -s '\n' < textfile > newfile
```

或

```
tr -s '\012' < textfile > newfile
```

7. 若要以 “?” (问号) 替换每个非显示字符 (有效控制字符除外), 请输入:

```
tr -c '[:print:][:cntrl:]' '[?*' < textfile > newfile
```

这便对不同语言环境中创建的文件进行扫描, 以查找当前语言环境下不能显示的字符。

8. 要以单个 “#” 字符替换 <space> 字符类中的每个字符序列, 请输入:

```
tr -s '[:space:]' '[#*]'
```

## 相关信息

**ed** 命令、**trbsd** 命令。

**ctype** 子例程。

《AIX 5L V5.3 本地语言支持指南和参考大全》中的『本地语言支持概述』。

---

## 跟踪守护程序

### 用途

记录选定的系统事件。

### 语法

```
trace [ -a [ -g ] ] [ -f | -l ] [ -b | -B ] [ -c ] [ -C [ CPUList | all ] ] [ -d ] [ -e string-cmd ] [ -h ]  
[ -j Event [ ,Event ] ] [ -k Event [ ,Event ] ] [ -J Event-group [ ,Event-group ] ] [ -K Event-group [ ,Event-group  
] ] [ -m Message ] [ -M ] [ -n ] [ -o Name ] [ -o- ] [ -p ] [ -r reglist ] [ -s ] [ -A process-id [ ,process-id ] ] [ -t thread-id [ ,thread-id ] ] [ -x program-specification | -X program-specification ] [ -l ] [ -P  
trace-propagation ] [ -L Size ] [ -T Size ]
```

### 描述

跟踪守护程序配置一个跟踪会话, 并启动系统事件的收集。跟踪功能所收集的数据记录在跟踪日志中。跟踪日志的报告可以用 **trcrpt** 命令生成。

当用 **-a**、**-x** 或 **-X** 标志调用时, 将异步运行跟踪守护程序 (即作为一个后台任务)。否则, 它就会交互式地运行, 并提示您输入子命令。

可以使用系统管理界面程序 (SMIT) 来运行跟踪守护程序。要使用 SMIT, 输入:

```
smit trace
```

跟踪数据收集的方式如下:

备用 (缺省值)

循环 (**-l**)

单一 (**-f**)

所有的跟踪事件都在跟踪日志文件中捕获。

跟踪事件包括在内存缓冲区中, 不在跟踪日志文件中捕获, 直到跟踪数据收集工作停止为止。

当内存跟踪缓冲区已满, 且缓冲区中的内容已捕获在跟踪日志文件中时, 跟踪事件的收集工作就结束。

## 缓冲区分配

跟踪缓冲区可以从内核堆分配，或放入独立的段中。缺省情况下，除非请求的缓冲区大小很大以致不能装入内核堆（这种情况下，则在独立的段中分配缓冲区），否则缓冲区一律都从内核堆中分配。从独立的段分配缓冲区，多少有点妨碍跟踪性能。但是，独立段中的缓冲区不会占用调页空间，只占一点锁定内存。缓冲区分配的类型可以用可选的 **-b** 或 **-B** 标志来指定。

从 AIX 5.3 开始，您可以选择仅跟踪选定的进程或线程。还可以跟踪单个程序。可以指定是否要传播跟踪，或者将跟踪扩展到新建的进程或线程。可以在这种跟踪中选择包括中断事件。这仅对跟踪通道 0 有效。

### 注:

1. 除非在启动被跟踪进程之前就启动了跟踪，否则将不会捕获进程启动事件。如果在启动被跟踪的进程之前就启动了跟踪，则还将捕获来自被跟踪进程之外的其他进程的一些事件。
2. 当跟踪使用来自内核堆的内存时（使用 **-B** 选项（仅限 32 位内核）时就是如此），该内存将保持作为内核内存的一部分，直到系统下一次重新引导。因此，在使用大缓冲区时，应该十分当心。此外，必须注意，在带有 5200-05 维护级别的 AIX 5.2 之前的版本中，在对所有的 CPU 使用多 CPU 跟踪（使用 **-C all** 选项）时，将为所有可能的 CPU 分配内存，而不仅仅为启动跟踪时正在运行的那些 CPU 分配内存。这在带有 5200-05 维护级别的 AIX 5.2 和 AIX 5.3 中进行了修正。

## 标志

**-a**

异步运行跟踪守护程序（即作为一个后台任务）。一旦以此种方式启动跟踪，则可以使用 **trcon**、**trcoff** 和 **trcstop** 命令来分别启动跟踪、停止跟踪或退出跟踪会话。这些命令可以作为链接到跟踪的链路来实现。仅跟踪列出的进程和 / 或其子代。*process-id* 是十进制数字。多个进程标识可以用逗号分隔，或者加上引号并用空格分隔。**-A** 标志仅对跟踪通道 0 有效；**-A** 和 **-g** 标志不兼容。

**-A** *process-id*[\_*process-id*]

启动跟踪时指定进程的现有的所有线程受到跟踪。在缺省情况下，如果在启动跟踪后，受到跟踪的进程将创建其他线程或进程，除非指定 **-P** 标志，否则将不跟踪它们。

**-b**

从内核堆分配缓冲区。如果不能从内核堆获取所请求的缓冲区空间，则命令失败。

注: **-b** 标志只对 32 位内核有效。

**-B**

在独立段中分配缓冲区。

注: **-B** 标志只对 32 位内核有效。

**-c**

保存跟踪日志文件，添加 **.old** 到其文件名。

**-C** [*CPUList* | **all** ]

使用一个缓冲区集合跟踪 *CPUList* 中的每个 CPU。多个 CPU 可以用逗号隔开，或包含在双引号中然后用逗号或空格隔开。要跟踪所有 CPU，请指定 **all**。因为该标志对每个 CPU 使用了一个缓冲区集合并为每个 CPU 生成一个文件，所以它可能消耗大量内存和文件空间，因此应该小心使用。产生的文件名为 **trcfile**、**trcfile-0** 和 **trcfile-1** 等，其中 **0**、**1** 等是 CPU 编号。如果指定了 **-T** 或 **-L**，则大小适用于每一个缓冲区集合和每一个文件。对于单处理机系统，可以指定 **-C all**，但是带有 CPU 编号列表的 **-C** 将被忽略。在 AIX 5.3 之前，如果 **-C** 用来指定多个 CPU（例如 **-C all** 或 **-C "0 1"**），则使用 **trcdead** 将不从系统转储抽取相关缓冲区。

警告: 只有 root 用户才能使用 **-C** 标志。

|                                        |                                                                                                                                                                                                                                         |
|----------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-d</b>                              | 禁用自动启动跟踪数据收集。延时启动跟踪数据收集。正常情况下，在启动跟踪守护程序时就会自动启动跟踪数据收集。使用 <b>trcon</b> 命令来启动跟踪数据收集。                                                                                                                                                       |
| <b>-e string-cmd</b>                   | 启动跟踪前，通过运行带有参数 <i>string-cmd</i> 的 <b>ctctrl</b> 来配置组件跟踪。也就是说，运行 <b>ctctrl string-cmd</b> 。允许传递多个 <b>-e</b> 选项，这等同于成功运行带每个 <i>string-cmd</i> 参数的 <b>ctctrl</b> 命令。在启动跟踪系统前，（通过将系统跟踪方式设置为 On、更改跟踪级别等）该选项可以用于对某些组件配置系统跟踪方式。               |
| <b>-f</b>                              | 以单一方式运行跟踪。当内存缓冲区已满时，立即停止跟踪数据收集。然后将跟踪数据写入跟踪日志。使用 <b>trcon</b> 命令来重新启动跟踪数据收集，并捕获另一个已满的数据缓冲区。如果在缓冲区满之前发出 <b>trcoff</b> 子命令，就停止跟踪数据收集并将缓冲区当前内容写入跟踪日志。                                                                                       |
| <b>-g</b>                              | 在一般跟踪通道（通道 1 至 7）中启动跟踪会话。该标志仅当跟踪异步运行（ <b>-a</b> ）时才有效。命令的返回码就是通道编号；通道编号随后必须用于一般跟踪子例程调用。要停止一般跟踪会话，可使用命令 <b>trcstop -&lt;channel_number&gt;</b> 。                                                                                         |
| <b>-h</b>                              | 省略跟踪日志的头记录。通常，跟踪守护程序在跟踪日志开始部位以日期和时间（通过 <b>date</b> 命令）来写头记录；系统名称、版本和发行版、节点标识、机器标识（通过 <b>uname -a</b> 命令）以及用户定义的消息。在跟踪日志开始部位，头记录的信息包括在 <b>trcrpt</b> 命令的输出中。                                                                             |
| <b>-l</b>                              | 跟踪中断事件。当用 <b>-A</b> 或 <b>-t</b> 指定时， <b>-l</b> 标志除了包括指定的进程或线程的事件之外，还包括中断事件。如果指定了 <b>-l</b> ，但是既没有指定 <b>-A</b> 也没有指定 <b>-t</b> ，则仅跟踪中断级别事件。 <b>-l</b> 标志仅对跟踪通道 0 有效； <b>-l</b> 和 <b>-g</b> 标志不兼容。                                        |
| <b>-j Event[, Event]</b>               | 指定您想要收集（ <b>-j</b> ）或排除（ <b>-k</b> ）其跟踪数据的用户定义事件。 <i>Event</i> 列表项可以用逗号隔开，或包含在双引号中并用逗号或空格隔开。<br><b>注：</b> 以下事件用于确定 <b>trcrpt</b> 报告中的 pid、cpuid 和 exec 路径名：<br>106 分派<br>10C 分派空闲进程<br>134 EXEC 系统调用<br>139 FORK 系统调用<br>465 KTHREAD 创建 |
| <b>-k Event[,Event]</b>                |                                                                                                                                                                                                                                         |
| <b>-J Event-group [, Event-group ]</b> | 如果这些事件中有任一事件丢失，则 <b>trcrpt</b> 命令所报告的信息就不完整。因此：当使用 <b>-j</b> 标志时，应该包括 <i>Event</i> 列表中的所有这些事件；相反，当使用 <b>-k</b> 标志时，则不应该包括 <i>Event</i> 列表中的这些事件。如果用 SMIT 或 <b>-J</b> 标志启动跟踪，则这些事件就在 <b>tidhk</b> 组中。                                    |
| <b>-K Event-group [ ,Event-group]</b>  |                                                                                                                                                                                                                                         |
| <b>-l</b>                              | 指定要被包含（ <b>-J</b> ）或排除（ <b>-K</b> ）的事件组。事件组在“调试和性能跟踪”中描述。 <b>-J</b> 和 <b>-K</b> 标志除了有事件组而不是个别 hook 标识以外，其他方面均同 <b>-j</b> 和 <b>-k</b> 作用相似。可以指定所有四个标志 <b>-j</b> 、 <b>-J</b> 、 <b>-k</b> 和 <b>-K</b> 。                                    |
| <b>-l</b>                              | 以循环方式运行跟踪。在停止跟踪数据收集时，跟踪守护程序将跟踪数据写入跟踪日志中。仅捕获跟踪数据的最后一个缓冲区。使用 <b>trcoff</b> 命令来停止跟踪数据收集时，应该使用 <b>trcon</b> 命令来重新启动它。                                                                                                                       |
| <b>-L Size</b>                         | 用所规定的值来重设缺省跟踪日志文件 1 MB 的大小。将文件大小指定为 0，便将跟踪日志文件大小设置为缺省值。<br><b>注：</b> 在循环方式和备用方式中，跟踪日志文件大小必须至少是跟踪缓冲区大小的两倍。在单一方式中，跟踪日志文件必须至少是缓冲区的大小。有关控制跟踪缓冲区大小的信息，请参阅 <b>-T</b> 标志。                                                                      |
| <b>-m Message</b>                      | 指定将要包括在跟踪日志的消息字段头记录中的文本。                                                                                                                                                                                                                |

- M** 将运行进程的地址映射转储到跟踪。如果要通过 **tprof** 命令处理跟踪文件，则必须指定 **-M** 标志。
- n** 将这些信息添加到跟踪日志头：锁定信息、硬件信息以及对于每一个装入程序项的符号名称、地址和类型。
- o Name** 重设 **/var/adm/ras/trcfile** 缺省跟踪日志文件并将跟踪数据写入用户定义的文件。
- o -** 重设缺省跟踪日志名称，并将跟踪数据写入标准输出。当使用此标志时，**-c** 标志被忽略。如果指定 **-o-** 和 **-C**，则会产生一个错误。
- p** 每一个 hook 都包含当前处理器的 cpuid。该标志仅对 64 位内核跟踪有效。
- P propagation** **注：trcrpt** 命令会向 cpuid 报告是否指定了该选项。用字母 **p** 指定传播以用于跨进程创建的传播、字母 **t** 用于跨线程创建的传播，而字母 **n** 用于无传播。跨进程创建的传播意味着跨线程创建的传播。例如，如果指定 **-A** 来跟踪进程，启动跟踪时此进程的存在的所有线程将受到跟踪。**-Pt** 标志导致此进程随后创建的所有线程也将受到跟踪。如果指定了 **-Pp**，将跟踪此进程随后创建的所有进程和线程。如果指定 **-t all** 来跟踪所有线程，则忽略 **-P**。**-P** 标志仅对跟踪通道 0 有效；**-P** 和 **-g** 标志不兼容。
- r reglist** 可选，并仅对在 64 位内核上运行的跟踪有效。*reglist* 项目用逗号分隔，或者加上引号并用空格分隔。最多可指定 8 个寄存器。有效 *reglist* 值为：
- PURR - PURR**  
此 CPU 的寄存器
- MCR0、MCR1、MCRA - MCR**  
寄存器、0、1 和 A
- PMC1、PMC2、…… PMC8 - PMC**  
寄存器 1 到 8。
- 注：**并非所有的寄存器都对所有的处理器有效。
- s** 当填写跟踪日志时，停止跟踪。当跟踪日志填满后，通常跟踪守护程序都回绕跟踪日志，并继续收集跟踪数据。在异步操作过程中，该标志可以使跟踪守护程序停止跟踪数据收集。（在交互式操作中，必须用 **quit** 子命令来停止跟踪。）
- t thread-id[,thread-id]** 仅跟踪列出的线程。线程标识是十进制数字。多个线程标识可以用逗号分隔，或者加上引号并用空格分隔。
- 另外，线程列表可以为 **all** 或 **\***，表示将跟踪所有线程。如果跟踪所有线程相关的事件而不跟踪中断相关的事件，那么它很有用。然而，如果同时指定了 **-t all** 和 **-I**，则等同于两者都未指定；将跟踪所有事件。换言之，**trace** 和 **trace -It all** 相同。
- t** 标志仅对跟踪通道 0 有效，**-t** 和 **-g** 标志不兼容。



**-T** *Size*

用规定的值覆盖跟踪缓冲区大小缺省值 128 KB。您必须为 root 用户来请求大于 1 MB 的缓冲区空间。最大可能的大小为 268435184 字节，除非使用 **-f** 标志（此时大小为 536870368 字节）。最小可能的大小为 8192 字节，除非使用 **-f** 标志（此时大小为 16392 字节）。当使用 **-f** 标志时，8192 和 16392 之间的大小都可使用；但是实际使用的大小为 16392 字节。

**注：**在循环方式和备用方式中，跟踪缓冲区的大小必须是跟踪日志文件大小的一半，或小于跟踪日志文件大小。在单一方式中，跟踪日志文件必须至少是缓冲区的大小。有关控制跟踪日志文件大小的信息，请参阅 **-L** 标志。另外也请注意，跟踪缓冲区使用锁定内存，这意味着它们不可页面调度。因此，跟踪缓冲区越大，就有越少的物理内存适用于应用程序。

除非指定 **-b** 或 **-B** 标志，否则系统都会试图从内核堆分配缓冲区空间。如果该请求不能得到满足，则系统会试图将缓冲区作为独立段来分配。

**-f** 标志实际上使用了两个缓冲区，它们一起作为单个缓冲区来运行（除了当第一个缓冲区满时，将记录一个缓冲区回绕跟踪 hook 这种情况以外）。

**-x** *program-specification*

跟踪指定的程序。除非如果还给定程序名称以外的内容，而必须给程序规范加引号，否则 *program-specification* 将指定程序和参数作为从 shell 运行程序时的程序和参数。当程序退出并返回程序的返回码时，跟踪将自动停止。在缺省情况下，还跟踪此程序创建的任何进程和线程；如同指定了 **-Pp**。要更改此行为，使用 **-Pn** 来指定不使用跟踪传播，或者使用 **-Pt** 将跟踪只传播到程序的原始进程创建的线程。

**-X** *program-specification*

**注：****-x** 标志意味着异步跟踪，如同还指定了 **-a** 标志。除了在程序退出时跟踪不自动停止以外，**-X** 标志与 **-x** 标志的作用相同。跟踪派生进程然后终止的程序时，它很有用，您希望也跟踪这些新进程。

## 子命令

当以交互方式运行时，跟踪会识别以下一些子命令：

|                        |                                    |
|------------------------|------------------------------------|
| <b>trcon</b>           | 启动跟踪数据收集。                          |
| <b>trcoff</b>          | 停止跟踪数据收集。                          |
| <b>q</b> 或 <b>quit</b> | 停止跟踪数据收集并退出跟踪。                     |
| <b>! Command</b>       | 运行由 <i>Command</i> 参数指定的 shell 命令。 |
| <b>?</b>               | 显示 <b>trace</b> 子命令摘要。             |

## 信号

**INTERRUPT** 信号充当启动和停止跟踪数据收集的开关。中断被设置为 **SIG\_IGN** 用于被跟踪的进程。

## 示例

1. 若要以交互方式使用跟踪，请输入 **trace**（此时将显示 > 提示符），然后指定所需的子命令。例如，要在运行 *anycmd* 命令期间跟踪系统事件，请输入：

```
trace
> !anycmd
> q
```

2. 要避免在命令完成时发生延时，可以只使用一个命令行异步启动跟踪（**-a**）。输入：

```
trace -a; anycmd; trcstop
```

3. 要跟踪系统本身 10 秒左右，请输入：

```
trace -a; sleep 10; trcstop
```

4. 要输出跟踪数据到特定的跟踪日志文件（而不是 **/var/adm/ras/trcfile** 缺省跟踪日志文件），请输入：

```
trace -a -o /tmp/my_trace_log; anycmd; trcstop
```

5. 要捕获 **cp** 命令的执行，从收集进程排除特殊事件，请输入：

```
trace -a -k "20e,20f" -x "cp /bin/track /tmp/junk"
```

在上例中，**-k** 选项抑制从 **lockl** 和 **unlockl** 函数收集事件（20e 和 20f 事件）。

还要注意使用了 **-x** 标志，因此将只跟踪与 **cp** 命令进程相关的 hook，并且将不跟踪中断活动。

6. 要跟踪 hook 234 以及能使用户可以看到进程名称的 hook，请使用：

```
trace -a -j 234 -J tidhk
```

这可以跟踪事件组“tidhk”中的 hook 和 hook 234。

7. 要使跟踪对每个 CPU 使用一个缓冲区集合，请指定：

```
trace -aC all
```

产生的文件为 **/var/adm/ras/trcfile**、**/var/adm/ras/trcfile-0**、**/var/adm/ras/trcfile-1** 等，直至 **/var/adm/ras/trcfile-(n-1)**，其中 *n* 为系统中 CPU 的数目。

注：trace -aCall -o mylog 产生文件 mylog-0、mylog-1、...

8. 要跟踪启动守护程序的程序，并在原始程序完成之后继续跟踪守护程序，则使用

```
trace -X "mydaemon"
```

必须用 **trcstop** 停止跟踪。

9. 要跟踪当前正在运行的 *mydaemon*，则使用：

```
trace -A mydaemon-process-id -Pp
```

其中 *mydaemon-process-id* 是通过 **ps** 命令返回的 *mydaemon* 的进程。**-Pp** 标志在跟踪运行时指示跟踪还要跟踪 *mydaemon* 创建的任何进程和线程。

10. 要捕获 PURR、PMC1 和 PMC2，请输入

```
trace -ar "PURR PMC1 PMC2"
```

## 文件

**/usr/include/sys/trcmacros.h**  
**/var/adm/ras/trcfile**

定义 **trchhook** 和 **utrchhook** 宏。  
包括缺省跟踪日志文件。

## 相关信息

**ctctrl** 命令、**trcevgrp** 命令、**trcnm** 命令、**trcrpt** 命令和 **trcstop** 命令。

**trchhook** 子例程、**trcgen** 子例程、**trcstart** 子例程、**trcon** 子例程、**trcoff** 子例程、**trcstop** 子例程。

**trcgenk** 内核服务。

《AIX 5L V5.3 通用编程概念: 编写并调试程序》中的『跟踪设施概述』。

《性能管理》中的『用跟踪工具进行性能分析』。

AIX 5L Version 5.3 Kernel Extensions and Device Support Programming Concepts 中的 Debug and Performance Tracing。

---

## traceroute 命令

### 用途

显示 IP 信息包至某个网络主机的路由。

### 语法

```
traceroute [ -m Max_ttl ] [ -n ] [ -p Port ] [ -q Nqueries ] [ -r ] [ -s SRC_Addr ] [ -t TypeOfService ] [ -v ] [ -w WaitTime ] Host [ PacketSize ]
```

### 描述

提示: **traceroute** 命令旨在用于网络测试、评估和管理。它应主要用于手动故障隔离。由于它对网络施加的负荷, **traceroute** 命令不应用于正常的操作中或自动脚本。

**traceroute** 命令试图跟踪 IP 信息包至某个因特网主机的路由, 其具体方法是: 先启动具有小的最大存活时间值 (*Max\_ttl* 变量) 的 UDP 探测信息包, 然后侦听从网关开始一路上的 ICMP **TIME\_EXCEEDED** 响应。探测以一个一跳跃位的 *Max\_ttl* 值开始, 该值一次增加一个跳跃值, 直至返回 ICMP **PORT\_UNREACHABLE** 消息。ICMP **PORT\_UNREACHABLE** 消息可以指出主机已经被定位, 或命令已经达到允许跟踪的最大跳跃数目。

**traceroute** 命令在每一个 *Max\_ttl* 设置上发送三个探测值以记录以下一些内容:

- *Max\_ttl* 值
- 网关地址
- 每次成功探测的往返时间

通过使用 **-q** 标志, 可以增加发送的探测数目。如果探测的回答来自不同的网关, 则命令会显示各个响应系统的地址。如果在 3 秒的超时时间间隔中没有来自探测的应答, 则会针对该探测显示出 \* (星号)。

如果 *Max\_ttl* 值为一个跳跃或更小, 则 **traceroute** 命令会在往返时间后显示一个 “!” (感叹号)。一个跳跃或更小的最大存活时间值一般都指出了不同网络软件处理 ICMP 应答的方式的不兼容性。不兼容性通常是通过将上次使用的 *Max\_ttl* 值加倍并再次尝试来解析的。

在往返注释后, 其他可能的注释有:

**IH** 主机不可到达  
**IN** 网络不可达  
**IP** 协议不可达  
**IS** 源路由失败  
**IF** 需要碎片

如果有许多探测都产生某一错误, 则 **traceroute** 命令退出。

**traceroute** 命令唯一的强制性参数就是目标主机名称或 IP 数字。**traceroute** 命令将根据输出接口的最大传输单元 (MTU) 确定探测信息包的长度。UDP 探测信息包被设置为一个不可能的值, 以防止目标主机的处理。

## 标志

|                                |                                                                                                                                                                            |
|--------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-m</b> <i>Max_ttl</i>       | 设置用于输出探测信息包的最大存活时间 (最大的跳跃数)。缺省值为 30 个跳跃 (TCP 连接也使用相同的缺省值)。                                                                                                                 |
| <b>-n</b>                      | 以数字方式而不以符号加数字的方式显示跳跃地址。该标志为在路径上找到的每个网关保存名称服务器的“地址到姓名”查询。                                                                                                                   |
| <b>-p</b> <i>Port</i>          | 设置用于探测的基本 UDP 端口号。缺省值为 33434。 <b>traceroute</b> 命令取决于目标主机的开放式 UDP 端口范围, <i>base</i> 到 <i>base + nhops - 1</i> 。如果 UDP 端口不可用, 则该选项可以用于选择一个未曾使用的端口范围。                        |
| <b>-q</b> <i>Nqueries</i>      | 指定 <b>traceroute</b> 命令在每个 <i>Max_ttl</i> 设定值处发出的探测数目。缺省值为三次探测。                                                                                                            |
| <b>-r</b>                      | 忽略正常的路由表, 并直接发送探测信息包至已链接网络上的主机。如果指定的主机不在直接连接的网络上, 则返回一个错误。该选项可以用于通过 <b>routed</b> 守护程序路由表中未注册的接口向本地主机发出 <b>ping</b> 命令。                                                    |
| <b>-s</b> <i>SRC_Addr</i>      | 以数字格式将下一 IP 地址用作输出探测信息包的源地址。在具有多个 IP 地址的主机上, 可以用 <b>-s</b> 标志来强制源地址成为不同于探测软件包被发送到的那个接口的 IP 地址的其他内容。如果下一个 IP 地址不是机器接口地址之一, 则返回一个错误且不发送任何内容。                                 |
| <b>-t</b> <i>TypeOfService</i> | 将探测信息包中的 <i>TypeOfService</i> 变量设置为 0 到 255 范围内的一个十进制整数。缺省值为 0。该标志可用于调查不同的服务类型是否产生了不同的路径。有关更多信息, 请参阅《网络与通信管理》中的『TCP/IP 协议』。有用的数值为 <b>-t 16</b> (低延迟) 和 <b>-t 8</b> (高吞吐量)。 |
| <b>-v</b>                      | 接收除 <b>TIME_EXCEEDED</b> 和 <b>PORT_UNREACHABLE</b> 以外的信息包 (详细输出)。                                                                                                          |
| <b>-w</b> <i>WaitTime</i>      | 设置等待探测响应的的时间 (以秒计)。缺省值为 3 秒。                                                                                                                                               |

## 参数

|                   |                                                            |
|-------------------|------------------------------------------------------------|
| <i>Host</i>       | 通过主机名或 IP 数字指定目标主机。该参数是必要的。                                |
| <i>PacketSize</i> | 指定探测数据报长度。缺省信息包大小可以通过 <b>traceroute</b> 命令根据输出接口的 MTU 来确定。 |

## 示例

1. 一个使用和输出的样本为:

```
[yak 71]% traceroute nis.nsf.net.
traceroute to nis.nsf.net (35.1.1.48), 30 hops max, 56 byte packet
 1 helios.ee.lbl.gov (128.3.112.1) 19 ms 19 ms 0 ms
 2 lilac-dmc.Berkeley.EDU (128.32.216.1) 39 ms 39 ms 19 ms
 3 lilac-dmc.Berkeley.EDU (128.32.216.1) 39 ms 39 ms 19 ms
 4 ccngw-ner-cc.Berkeley.EDU (128.32.136.23) 39 ms 40 ms 39 ms
 5 ccn-nerif22.Berkeley.EDU (128.32.168.22) 39 ms 39 ms 39 ms
 6 128.32.197.4 (128.32.197.4) 40 ms 59 ms 59 ms
 7 131.119.2.5 (131.119.2.5) 59 ms 59 ms 59 ms
 8 129.140.70.13 (129.140.70.13) 99 ms 99 ms 80 ms
 9 129.140.71.6 (129.140.71.6) 139 ms 239 ms 319 ms
10 129.140.81.7 (129.140.81.7) 220 ms 199 ms 199 ms
11 nic.merit.edu (35.1.1.48) 239 ms 239 ms 239 ms
```

由于第二个跳跃系统 (lbl-csam.arpa) 内核中的错误 (向前转发了零存活时间的信息包), 结果第二行和第三行相同。在第 6 到第 10 行中, 不显示主机名称, 因为国家科学基金会网络 (NSFNet, 129.140) 不提供其节点的地址到名称的转换。

2. 另外一个输出样本如下:

```
[yak 72]% traceroute rip.Berkeley.EDU (128.32.131.22)
traceroute to rip.Berkeley.EDU (128.32.131.22), 30 hops max
 1 helios.ee.lbl.gov (128.3.112.1) 0 ms 0 ms 0 ms
 2 lilac-dmc.Berkeley.EDU (128.32.216.1) 39 ms 19 ms 39 ms
 3 lilac-dmc.Berkeley.EDU (128.32.216.1) 19 ms 39 ms 19 ms
 4 ccngw-ner-cc.Berkeley.EDU (128.32.136.23) 39 ms 40 ms 19 ms
 5 ccn-nerif35.Berkeley.EDU (128.32.168.35) 39 ms 39 ms 39 ms
 6 csgw/Berkeley.EDU (128.32.133.254) 39 ms 59 ms 39 ms
 7 * * *
 8 * * *
 9 * * *
10 * * *
11 * * *
12 * * *
13 rip.Berkeley.EDU (128.32.131.22) 59 ms! 39 ms! 39 ms!
```

在本示例中，恰好“丢失”了 12 个网关跳跃的一半（13 是最后的目标）。但是，这些跳跃实际并不是网关。目标主机，一个运行 Sun OS3.5 的 Sun-3 工作站，将来自到达数据报的 ttl 用作其 ICMP 回复的 ttl，因此回复会在返回路径上发生超时。由于 ICMP 不是对 ICMP 发送的，因此不会接收到任何通知。每个往返时间后的！（感叹号）指示某种类型的软件不相兼容性问题。（其原因是在 **traceroute** 命令发布一个两倍路径长的探测之后进行诊断。目标主机实际只是在 7 个跳跃以外。）

## 相关信息

**netstat** 命令、**nslookup** 命令和 **ping** 命令。

《网络与通信管理》中的『TCP/IP 名称解析』。

---

## tracesoff 命令

### 用途

关闭对子系统、子系统组或子服务器的跟踪。

### 语法

#### 子系统

```
tracesoff [ -h Host ] { -g Group | -p SubsystemPID | -s Subsystem }
```

#### 子服务器

```
tracesoff [ -h Host ] -t Type [ -p SubsystemPID ] { -o Object | -P SubserverPID }
```

### 描述

**tracesoff** 命令向系统资源控制器发送一个子系统请求信息包，该信息包将被转发到该子系统以关闭跟踪功能。如果子系统间通信方法为信号法，则跟踪失败。

注：跟踪是依赖于子系统的。

### 标志

**-g** Group 指定一组子系统来关闭跟踪。如果 Group 名称没有包含在子系统对象类中，则该命令失败。

|                               |                                                                                                                                                                                                                                         |
|-------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-h</b> <i>Host</i>         | 指定请求在其上进行跟踪操作的外部主机。本地用户必须作为 <code>root</code> 用户运行。必须配置远程系统以接受系统资源控制器的请求。也就是说， <code>srcmstr</code> 守护程序（请参阅 <code>/etc/inittab</code> ）必须用 <code>-r</code> 标志启动，并且必须配置 <code>/etc/hosts.equiv</code> 或 <code>.rhosts</code> 文件以允许远程请求。 |
| <b>-o</b> <i>Object</i>       | 指定子服务器的 <i>Object</i> 名称作为字符串传递到子系统中。                                                                                                                                                                                                   |
| <b>-p</b> <i>SubsystemPID</i> | 指定子系统的特定实例，要关闭跟踪或指定将跟踪子服务器请求传递到其中。                                                                                                                                                                                                      |
| <b>-P</b> <i>SubserverPID</i> | 指定 <i>SubserverPID</i> 作为字符串传递到该子系统。                                                                                                                                                                                                    |
| <b>-s</b> <i>Subsystem</i>    | 指定一个子系统以关闭跟踪。 <i>Subsystem</i> 名称可以是实际的子系统名称或该子系统的同义名称。如果 <i>Subsystem</i> 名称没有包含在子系统对象类中，则该命令失败。                                                                                                                                       |
| <b>-t</b> <i>Type</i>         | 指定一个子系统子服务器以关闭跟踪。如果 <i>Type</i> 没有包含在子系统对象类中，则该命令失败。                                                                                                                                                                                    |

## 示例

要关闭对一个子系统的跟踪，请输入：

```
tracesoff -s tcpip
```

这会关闭对 `tcpip` 子系统的跟踪。

## 文件

|                                      |                                |
|--------------------------------------|--------------------------------|
| <code>/usr/bin/tracesoff</code>      | 包含 <code>tracesoff</code> 命令。  |
| <code>/etc/objrepos/SRCsubsys</code> | 指定 SRC 子系统配置对象类。               |
| <code>/etc/objrepos/SRCsubsvr</code> | 指定 SRC 子服务器配置对象类。              |
| <code>/etc/services</code>           | 定义用于因特网服务的套接字和协议。              |
| <code>/dev/SRC</code>                | 指定 <code>AF_UNIX</code> 套接字文件。 |
| <code>/dev/.SRC-unix</code>          | 指定临时套接字文件位置。                   |

## 相关信息

`traceson` 命令。

《操作系统与设备管理》中的『系统资源控制器』说明了子系统、子服务器和系统资源控制器。

---

## traceson 命令

### 用途

打开对子系统、子系统组或子服务器的跟踪功能。

### 语法

#### 子系统

```
traceson [ -h Host] [ -l] { -g Group | -p SubsystemPID| -s Subsystem}
```

#### 子服务器

```
traceson [ -h Host] [ -l] -t Type [ -o Object] [ -p SubsystemPID] [ -P SubserverPID]
```

## 描述

**traceson** 命令向系统资源控制器发送一个子系统请求信息包，该信息包被转发到该子系统以打开跟踪功能。如果子系统间通信方法为信号法，则跟踪失败。

注：跟踪是依赖于子系统的。

跟踪的格式可长可短。当缺少 **-l** 标志时，假定跟踪请求是短期跟踪。

## 标志

|                               |                                                                                                                                                                                              |
|-------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-g</b> <i>Group</i>        | 指定一组子系统来打开跟踪。如果 <i>Group</i> 名称没有包含在子系统对象类中，则该命令失败。                                                                                                                                          |
| <b>-h</b> <i>Host</i>         | 指定请求在其上进行跟踪操作的外部主机。本地用户必须作为“root”用户运行。必须配置远程系统以接受系统资源控制器的请求。也就是说， <b>srcmstr</b> 守护程序（请参阅 <b>/etc/inittab</b> ）必须用 <b>-r</b> 标志启动，并且必须配置 <b>/etc/hosts.equiv</b> 或 <b>.rhosts</b> 文件以允许远程请求。 |
| <b>-l</b>                     | 指定请求一个长期跟踪。                                                                                                                                                                                  |
| <b>-o</b> <i>Object</i>       | 指定子服务器对象作为字符串传递给子系统。                                                                                                                                                                         |
| <b>-p</b> <i>SubsystemPID</i> | 指定子系统的特定实例，要打开跟踪或指定将跟踪子服务器请求传递到其中。                                                                                                                                                           |
| <b>-P</b> <i>SubserverPID</i> | 指定子服务器 <b>PID</b> 是作为字符串传递给子系统的。                                                                                                                                                             |
| <b>-s</b> <i>Subsystem</i>    | 指定要打开跟踪的子系统。 <i>Subsystem</i> 名称可以是实际的子系统名称或该子系统的同义名称。如果 <i>Subsystem</i> 名称没有包含在子系统对象类中，则该命令失败。                                                                                             |
| <b>-t</b> <i>Type</i>         | 指定一个子服务器来打开跟踪。如果 <i>Type</i> 没有包含在子系统对象类中，则该命令失败。                                                                                                                                            |

## 示例

要打开对外部主机上的 **tcpip** 子系统的跟踪功能，请输入：

```
traceson -h odin -s tcpip
```

这会打开对 **odin** 外部主机上的 **tcpip** 子系统的跟踪功能。

## 文件

|                                |                          |
|--------------------------------|--------------------------|
| <b>/usr/bin/traceson</b>       | 包含 <b>traceson</b> 命令。   |
| <b>/etc/objrepos/SRCsubsys</b> | 指定 <b>SRC</b> 子系统配置对象类。  |
| <b>/etc/objrepos/SRCsubsvr</b> | 指定 <b>SRC</b> 子服务器配置对象类。 |
| <b>/etc/services</b>           | 定义用于因特网服务的套接字和协议。        |
| <b>/dev/SRC</b>                | 指定 <b>AF_UNIX</b> 套接字文件。 |
| <b>/dev/.SRC-unix</b>          | 指定临时套接字文件位置。             |

## 相关信息

**tracesoff** 命令。

《操作系统与设备管理》中的『系统资源控制器』提供了对子系统、子服务器和系统资源控制器的说明。

---

## trbsd 命令

### 用途

翻译字符（BSD 版）。

## 语法

```
trbsd [ -c ] [ -d ] [ -s ] [ -A ] [ String1 [ String2 ] ]
```

## 描述

**trbsd** 命令从标准输入删除或替换字符，然后将其结果写入标准输出。**trbsd** 命令是 **tr** 命令的 BSD 版。根据由参数指定的字符串和指定标志，**trbsd** 命令可执行三种操作。*String1* 或 *String2* 参数的缺省值为空字符串。

### 转换字符

如果 *String1* 和 *String2* 参数都已指定，但 **-d** 标志没有指定，**trbsd** 命令会从标准输出中将 *String1* 参数所指定的每一个字符替换成 *String2* 参数中同样位置上的字符。

如果 *String1* 参数多次指定某个字符，则该字符就在 *String2* 参数中被翻译成为与 *String1* 参数中最后出现的字符相对应的字符。

### 使用 **-d** 标志删除字符

如果指定了 **-d** 标志，**trbsd** 命令就会从标准输入中删除 *String1* 参数所指定的每一个字符。

### 使用 **-s** 标志除去字符序列

如果指定了 **-s** 标志，**trbsd** 命令将从标准输入中删除 *String2* 参数所指定的重复出现两次或多次的字符的序列中除第一个字符外的所有字符。

当 **-d** 和 **-s** 标志都指定时，*String1* 和 *String2* 参数必须都指定。

注：**trbsd** 命令在开始处理之前从标准输入里删除所有空字符。

## 表达字符串的特殊序列

包含在 *String1* 和 *String2* 参数中的字符串可以使用以下约定来表达：

- C1-C2** 指定在 *C1* 字符串指定的字符和 *C2* 字符串指定的字符之间（包括 *C1* 和 *C2*）进行整理的字符串。*C1* 字符串指定的字符必须整理放在 *C2* 字符串指定的字符之前。
- \Octal** 指定其编码由指定的八进制值表示的字符。八进制值可以是一个 1 位、2 位或 3 位八进制整数。多字节字符可以通过对每一个字节写反斜线八进制序列来表示。
- \-** "\-"（反斜线符，减号）表示减号字符本身，没有作为转义字符的任何特殊含意。

如果由 *String1* 和 *String2* 参数指定的字符串长度不同，**trbsd** 命令填充较短字符串使其长度与较长字符串相同。字符填充是通过复制较短字符串的最后一个字符来完成的，复制的次数由需要决定。

## 标志

- A** 使用范围和字符类的 ASCII 整理顺序而不是当前语言环境整理顺序来一个字节一个字节地执行所有操作。
- c** 指定 *String1* 参数的值由该字符串的补码来替换。该补码为当前语言环境的字符集中的所有字符，除了由 *String1* 参数指定的字符以外。如果一起指定了 **-A** 和 **-c** 标志，则与所有 8 位字符代码集合有关的字符将被补足。
- d** 从标准输入中删除包含在 *String1* 参数里的每一个字符。
- s** 从标准输入中删除包含在 *String2* 参数中的重复出现两次或多次的字符的序列中除第一个字符外的所有字符。

## 示例

1. 要将大括号转换成小括号，请输入：

```
trbsd '{} '()' < textfile > newfile
```



这便将每个 { (左大括号) 转换成 ( (左小括号), 并将每个 } (右大括号) 转换成 ) (右小括号)。所有其他的字符都保持不变。

2. 要让加号和减号以及斜杠字符和星号字符互换, 请输入:

```
trbsd '+\-/ *' '\-+*/' < textfile > newfile
```

减号必须与反斜杠转义字符一同输入。

3. 要将小写字符转换成大写, 请输入:

```
trbsd 'a-z' 'A-Z' < textfile > newfile
```

4. 要创建文件中的单词列表, 请输入:

```
trbsd -cs 'a-zA-Z' '\012' < textfile > newfile
```

这便将每一序列的字符 (小、大写字母除外) 都转换成单个换行符。八进制值 012 是换行符的编码。

5. 要用单个换行替换每一序列的一个或多个换行, 请输入:

```
trbsd -s '\012' < textfile > newfile
```

## 文件

**/usr/bin/trbsd**

包含 **trbsd** 命令。

**/usr/ucb/tr**

包含与 **trbsd** 命令的符号链接。

## 相关信息

**ed** 命令和 **tr** 命令。

《AIX 5L V5.3 本地语言支持指南和参考大全》中的『本地语言支持概述』。

---

## trcctl 命令

### 用途

更改并显示系统跟踪参数。

### 语法

```
trcctl [ -d Directory -l -L LogfileSize -M LMT_log_dir -N NonrootUserBufferMax -o Logfile -r -T BufferSize ]
```

### 描述

**trcctl** 命令将显示或更改系统跟踪缺省参数。如果指定了 **-l** 选项 (或没有指定参数), **trcctl** 将显示如下值:

```
Default Buffer Size: 131072
Default Log File Size: 1310720
Default Log File: /var/adm/ras/trcfile
Non-root User Buffer Size Maximum: 1048576
Default Components Directory File: /var/adm/ras/trc_ct
Default LMT Log Dir: /var/adm/ras/mtrcdir
```

请注意, 缺省缓冲区和日志文件大小最初取决于内核。然而, 一旦使用此命令对其进行设置, 则这两个内核的有效值相同。其他参数允许更改这些缺省值。要更改缺省值, 用户必须是系统组的成员。许多与 **trcctl** 一同使用的标志与 **trace** 守护程序使用的那些标志相符。

## 标志

|                              |                                                                                                     |
|------------------------------|-----------------------------------------------------------------------------------------------------|
| <b>-d</b> <i>Directory</i>   | 指定缺省组件跟踪日志目录路径。缺省值是 <code>/var/adm/ras/trc_ct</code> 。                                              |
| <b>-l</b>                    | 列出当前值。                                                                                              |
| <b>-L</b> <i>Value</i>       | 指定缺省日志文件大小。原始缺省值为 1310720 字节（对于 32 位内核）和 2621440 字节（对于 64 位内核）。如果用 <b>-L</b> 指定缺省值，则缺省值将同时适用于这两个内核。 |
| <b>-M</b> <i>LMT_log_dir</i> | 指定缺省轻量级内存跟踪日志目录路径。缺省值是 <code>/var/adm/ras/mtrcdir</code> 。                                          |
| <b>-N</b> <i>Value</i>       | 指定非 root 用户可以指定的最大缓冲区大小。缺省值为 1 MB，1048576 字节。                                                       |
| <b>-o</b> <i>Path</i>        | 指定缺省日志文件路径。缺省值为 <code>/var/adm/ras/trcfile</code> 。                                                 |
| <b>-r</b>                    | 恢复原始缺省值。                                                                                            |
| <b>-T</b> <i>Value</i>       | 指定缺省跟踪缓冲区大小。对于 32 或 64 位内核，原始缺省值为 128 KB 和 256 KB。如果用 <b>-T</b> 指定缺省值，则缺省值将同时适用于这两个内核。              |

## 参数

如果使用“k”、“m”或“#k”、“#m”作为 **-N**、**-L** 和 **-T** 选项的参数，**trcctl** 会将这些参数转换为它们各自的字节总数。

k = 1024

m = 1048576

如果仅使用“k”或“m”，则 **trcctl** 假定您分别想要 1 千字节或 1 兆字节。root 用户可以执行此方法：

```
trcctl -L 10m -N m -T 256k
```

## 安全

用户必须是系统组的成员。

## 相关信息

**trace** 守护程序。

**ctctrl**、**trcrpt**、**trcevgrp**、**traceon** 和 **traceoff** 命令。

---

## trcdead 命令

### 用途

从系统转储映像中抽取跟踪缓冲区。

### 语法

```
trcdead [ -1 -2 -3 ... -7 ] [-c] [-M] [ -oName ] DumpImage
```

### 描述

如果系统停机而跟踪工具仍然是活动的，则内部跟踪缓冲区的内容在系统转储中被捕获。然后可以使用 **trcdead** 命令从系统转储中抽取八个活动系统跟踪通道、所有组件跟踪缓冲区和轻量级内存跟踪缓冲区。没有提供标志时抽取系统跟踪通道 0。通过 **-channelnum** 标志标识除通道 0 外的系统跟踪通道。通过 **-c** 标志标识组件跟踪缓冲区。通过 **-M** 标志标识轻量级内存跟踪缓冲区。一次仅能抽取一种类型的跟踪缓冲区或一个特定的系统跟踪通道。

**-o** 标志可用于表示应将抽取的缓冲区写到非缺省跟踪日志或目录。系统跟踪通道被抽取到跟踪日志文件。组件跟踪缓冲区和轻量级内存跟踪缓冲区被抽取到目录。如果没有选择 **-o** 标志，则 **trcdead** 命令将写到缺省跟踪日志文件或目录。使用 **trcctl** 命令可以查看和修改缺省日志文件名称和目录名称。

在 AIX 之前，如果使用 **trace -C** 为多个 CPU 请求多 CPU 跟踪缓冲，则相关的缓冲区不会被转储。

使用 **trcrpt** 命令格式化来自一个或多个跟踪日志文件的报告。

## 标志

|                    |                                            |
|--------------------|--------------------------------------------|
| <b>-1, ..., -7</b> | 从通道 1、2、3、4、5、6 和 7 中检索跟踪缓冲区条目。缺省值是通道 0。   |
| <b>-c</b>          | 抽取所有活动的组件跟踪组件的所有缓冲区。                       |
| <b>-M</b>          | 抽取轻量级内存跟踪缓冲区。                              |
| <b>-oName</b>      | 指定数据被写到的目标文件或目录 ( <b>-c</b> 和 <b>-M</b> )。 |

## 参数

*DumpImage* 指定要操作的转储映像。

## 示例

注：要决定哪一个示例更适合您的计算机系统，请使用 **sysdumpdev** 命令来显示当前所指定的转储设备。

1. 要将系统跟踪缓冲区从位于 **/var/adm/ras/dumpfile** 的转储抽取到名为 **trace\_extract** 的文件，请输入：

```
trcdead -o trace_extract /var/adm/ras/dumpfile
```

2. 要从被写到一个设备的转储映像中抽取系统跟踪缓冲区，请输入：

```
trcdead /dev/hd7
```

3. 要从转储映像 **vmcore.0** 抽取轻量级内存跟踪信息并放入 **/tmp** 目录，请输入：

```
trcdead -o /tmp -M vmcore.0
```

## 文件

|                              |                       |
|------------------------------|-----------------------|
| <b>/usr/bin/trcdead</b>      | 包含 <b>trcdead</b> 命令。 |
| <b>/var/adm/ras/dumpfile</b> | 包含缺省系统转储文件。           |
| <b>/var/adm/ras/trcfile</b>  | 包含缺省系统跟踪日志。           |
| <b>/var/adm/ras/trc_ct</b>   | 包含缺省组件跟踪日志。           |
| <b>/var/adm/ras/mtrcdir</b>  | 包含缺省轻量级内存跟踪日志。        |

## 相关信息

**errdead** 命令、**sysdumpdev** 命令、**trcnm** 命令和 **trcrpt** 命令。

**trace** 守护程序。

《AIX 5L V5.3 通用编程概念：编写并调试程序》中的『跟踪设施概述』。

---

## trcevgp 命令

### 用途

操作跟踪事件组

### 语法

列出事件组

```
trcevgp -l [ event-group [ ... ] ]
```

除去事件组

```
trcevgp -r [ event-group [ ... ] ]
```

添加事件组

```
trcevgp -a -d "group-description" -h "hook-list" event-group
```

更新事件组

```
trcevgp -u [ -d "group-description" ] [ -h "hook-list" ] event-group ]
```

### 描述

**trcevgp** 命令用来维护跟踪事件组。您必须进入系统组来添加、删除或更改跟踪事件组。不能修改或删除其类型为 `reserved` 的事件组。

### 标志

**-a** [ **-d** *group-description* | **-h** *hook-list* ] 创建新事件组。只有能指定一个事件组名。使用 **-a** 标志时，**-d** *description* 和 **-h** *hook-list* 都必须指定。如果未指定 **-d** 或 **-h**，则会产生一个错误。

**-d** *group-description* 指定挂钩描述。所有新组都需要有一个描述。

**-h** *hook-list* 挂钩列表由跟踪挂钩标识组成。使用 **-a** 标志时 **-h** 标志是必需的。当更新一个事件组 (**-u** 标志) 时，挂钩列表 (如果已指定) 必须包含该组所有的挂钩标识。列表参数项可以用逗号隔开，或包含在双引号中然后用逗号或空格隔开。

**-l** *event-group* 将指定的组列到标准输出中。如果没有指定事件组，则列出所有项。列表格式如下：

```
group name - group-description (type) "hook list"
```

例如：

- `-l tidhk - Hooks needed to display thread name (reserved) "106,10C,134,139,465"`
- `-l gka - GENERAL KERNEL ACTIVITY (files,execs,dispatches) (reserved) "106,10C,134,139,465,107,135,15b,12e,116,117,200,20E,20F"`

**-r** *event-group* 除去指定的事件组。

**-u** [ **-d** "*group-description*" | **-h** "*hook-list*" ] *event-group* 用于更新事件组信息。必须指定 **-d** *description* 或 **-h** *hook-list*。

## 示例

1. 要获取所有事件组的列表，请输入：

```
trcevgrp -l
```

2. 要添加一个新事件组，请输入：

```
trcevgrp -a -d my group description -h "500,501,502" mygrp
```

这便将添加一个名为 `mygrp` 的组，赋予其描述 `my group description`，而且将拥有挂钩 500、501 和 502。

3. 要将另外一个挂钩添加到 `mygrp`，请输入：

```
trcevgrp -u -d my group description -h "500,501,502,503" mygrp
```

注：必需指定所有挂钩标识。

文件事件组当前保存在 `SWserveAt ODM` 数据库中。

## 相关信息

`trcdead` 命令、`trcnm` 命令、`trcrpt` 命令、`trcstop` 命令和 `trcupdate` 命令。

`trace` 守护程序。

`trcfmt` 文件格式。

《AIX 5L V5.3 通用编程概念：编写并调试程序》中的『跟踪设施概述』。

---

## trcnm 命令

### 用途

生成内核名称列表。

### 语法

```
trcnm [ -a [ FileName ] ] | [ FileName ] | -KSymbol ...
```

### 描述

`trcnm` 命令生成一个 `trcrpt` 命令使用的内核名称列表。内核名称列表由一个符号表和一个对象文件的装载程序符号表组成。`trcrpt` 命令在格式化来自跟踪日志文件的报告时使用内核名称列表文件来解释地址。有关更多信息，请参阅 `trcrpt -n` 命令。

如果没有指定 `FileName` 参数，则缺省 `FileName` 参数值为 `/unix`。

### 标志

**-a** 将所有的装载程序符号写到标准输出。缺省情况为只写系统调用的装载程序符号。  
**-KSymbol...** 通过 `knlist` 命令系统调用获取所有命令行符号的值。

## 示例

1. 要获取 `/unix` 中的符号的值，请输入：

```
trcnm -K environ errno
```

此命令序列显示以下内容:

```
environ 2FF7FFF8
errno 2FF7FFFC
```

2. 要为系统调用显示符号表, 请输入:

```
trcnm
```

生成与以下类似的列表:

```
pin_obj_start    00000000
header_offset    00000008
ram_disk_start   0000000C
ram_disk_end     00000010
dbg_avail        00000014
base_conf_start  00000018
base_conf_end    0000001C
base_conf_disk   00000020
pin_com_start    00000024
start            00000028
ipl_cb           00000028
...
```

## 文件

|                                   |                           |
|-----------------------------------|---------------------------|
| <code>/var/adm/ras/trcfile</code> | 包含缺省日志文件。                 |
| <code>/tlo-tvl2/trcnam</code>     | 包含 <code>trcnm</code> 命令。 |
| <code>/etc/trcfmt</code>          | 包含跟踪格式文件。                 |

## 相关信息

`trcdead` 命令、`trcrpt` 命令、`trcstop` 命令和 `trcupdate` 命令。

`trace` 守护程序。

`trcfmt` 文件格式。

《AIX 5L V5.3 通用编程概念: 编写并调试程序》中的『跟踪设施概述』。

---

## trcrpt 命令

### 用途

格式化来自跟踪日志的报告。

### 语法

```
trcrpt [ -c ] [ -C [ CPUList | all ] ] [ -d List ] [ -D Event-group-list ] [ -e Date ] [ -G ] [ -h ] [ -j ]
[ -k List ] [ -K Group-list ] [-m] [ -n Name ] [ -o File ] [ -p List ] [ -r ] [ -s Date ] [ -t File ] [
-T List ] [ -v ] [ -O Options ] [ -x ] [-M common | rare | all[:LMT_dir]] [ -l ComponentList | all[:CT_dir]
] [ FileOrDirectory ]
```

### 描述

`trcrpt` 命令读取 `-M`、`-l` 和 `File` 或 `Directory` 参数中指定的跟踪日志、格式化跟踪条目并向标准输出写报告。系统生成跟踪报告的缺省文件是 `/var/adm/ras/trcfile` 文件, 但是您可以使用 `-M`、`-l` 和 `File` 或 `Directory` 参

数指定备用日志文件。可以指定一个或多个文件或目录。如果指定一个文件，则它必须是有效的跟踪日志文件，即与跟踪相关的命令生成的任何文件。如果指定一个目录，则它必须包含一个组件跟踪主文件。如果指定 **-m** 标志，则所有指定的跟踪都将按照时间顺序合并。

可以通过输入 **SMIT** 快速路径使用系统管理界面工具 (**SMIT**) 运行 **trcrpt** 命令：

```
smit trcrpt
```

## 标志

|                                           |                                                                                                                                                                                                                                                                                                              |
|-------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-c</b>                                 | 检查模板文件的语法错误。                                                                                                                                                                                                                                                                                                 |
| <b>-C</b> [ <i>CPUList</i>   <b>all</b> ] | 生成报告，包含指定 CPU 上发生的事件。多个 CPU 可以用逗号隔开，或包含在双引号中然后用逗号或空格隔开。要生成所有 CPU 的报告，请指定 <b>trace -C all</b> 命令。 <b>-C</b> 标志是不必要的，除非您只想看到被跟踪 CPU 的一个子集，或者在报告中显示 CPU 号。如果没有指定 <b>-C</b> ，而且该跟踪为多 CPU 跟踪，则 <b>trcrpt</b> 生成所有 CPU 的跟踪报告，但是对每个挂钩都不显示 CPU 号，除非您指定 <b>-O cpuid=on</b> 。                                          |
| <b>-d</b> <i>List</i>                     | 将报告限制在 <i>List</i> 变量指定的挂钩标识范围中。 <i>List</i> 参数项可以用逗号隔开，或包含在双引号中并用逗号或空格隔开。                                                                                                                                                                                                                                   |
| <b>-D</b> <i>Event-group-list</i>         | 将报告限制在事件组列表里的挂钩标识，加上用 <b>-d</b> 标志指定的任意挂钩标识。列表参数项可以用逗号隔开，或包含在双引号中然后用逗号或空格隔开。在调试和性能跟踪中，对事件组有描述。                                                                                                                                                                                                               |
| <b>-e</b> <i>Date</i>                     | 用指定日期或指定日期之前的条目结束报告时间。 <i>Date</i> 变量的格式为 <i>mmddhhmmssyy</i> （月、日、时、分、秒以及年）。只有当开始和停止跟踪数据收集时，才在跟踪数据里记录日期和时间。如果您在一次跟踪对话期间多次停止和重新启动跟踪数据收集，每一次启动或停止跟踪数据收集时都会记录下日期和时间。将此标志与 <b>-s</b> 标志联合使用将跟踪限制在特定时间间隔期所收集的数据上。<br><b>注：</b> <b>-e</b> 和 <b>-s</b> 标志仅对没有使用 <b>trace -C</b> 标志收集的跟踪日志文件有效。                     |
| <b>-G</b>                                 | 列出所有事件组。组列表、每组的挂钩标识以及每组的描述都在标准输出中列出。                                                                                                                                                                                                                                                                         |
| <b>-h</b>                                 | 省略跟踪报告的头信息，只将格式化过的跟踪条目写到标准输出。                                                                                                                                                                                                                                                                                |
| <b>-j</b>                                 | 显示挂钩标识列表。 <b>trcrpt -j</b> 命令可以与包括跟踪事件标识的 <b>trace -j</b> 命令或者不包括跟踪事件标识的 <b>trace -k</b> 命令一起使用。                                                                                                                                                                                                             |
| <b>-k</b> <i>List</i>                     | 从报告中排除 <i>List</i> 变量指定的挂钩标识。 <i>List</i> 参数项可以用逗号隔开，或包含在双引号中并用逗号或空格隔开。                                                                                                                                                                                                                                      |
| <b>-K</b> <i>Event-group-list</i>         | 从报告中排除 <i>event-groups</i> 列表中的报告挂钩标识，加上 <b>-k</b> 标志所指定的任意挂钩标识。列表参数项可以用逗号隔开，或包含在双引号中然后用逗号或空格隔开。在调试和性能跟踪中对事件组有描述。                                                                                                                                                                                            |
| <b>-l</b> <i>ComponentList</i>            | 使用 <b>ctctrl -D</b> 或 <b>trcdead</b> 生成多组件跟踪的报告。多个组件可以用逗号隔开，或括在双引号中，然后用逗号或空格隔开。 <b>-l</b> 标志不是必要的，除非您只想看到被跟踪组件的一个子集。如果没有指定 <b>-l</b> 并且给定一个目录作为命令行的输入，则命令假设跟踪为一个多组件跟踪。不在缺省目录中的多组件跟踪日志文件必须在命令行中指定了它们的目录，或带有与 <b>-l</b> 标志组合使用的 <i>CT_dir</i> 参数。可以使用 <b>-l all</b> 选项选择所有可用的组件。可以使用多个 <b>-l</b> 标志指定不同目录中的组件。 |
| <b>-m</b>                                 | 根据时间戳记合并所有指定的跟踪文件。从另一分区、另一系统或从同一系统中的两个或更多不同的引导合并的文件将产生不可预测的结果。不使用 <b>-m</b> 标志，每个日志文件的报告都将附加到指定的输出文件。                                                                                                                                                                                                        |

**-M common | rare | all[:LMT\_dir]** 从通过 **mtrcsave** 或 **trcdead** 命令获取的 LMT 日志文件生成报告。

如果您只是想报告来自公共 LMT 缓冲区的事件，则使用 **common** 关键字；如果您只是想报告来自特殊 LMT 缓冲区的事件，则使用 **rare** 关键字；如果您想要报告公共事件和特殊事件，则使用 **all** 关键字。

除非指定了 *LMT\_dir* 参数，否则该标志仅搜索缺省 LMT 日志目录。使用该参数，**trcrpt** 命令将在指定的目录中而不是在缺省 LMT 日志目录中搜索 LMT 文件。要合并公共缓冲区和特殊缓冲区，必须使用 **all** 关键字和 **-m** 标志。**-M** 标志只能出现一次。

**-n Name**

指定用于解释输出地址的内核名称列表文件。通常，在将一个跟踪日志文件移动到另外一个系统时会用到该标志。

**-o File**

将报告写入文件而不是标准输出。



## -O Options

指定改变 **trcrpt** 命令内容和表示的选项。选项的参数必须以逗号分隔，或括在双引号中，然后用逗号或空格分隔。有效选项为：

### **2line=[on|off]**

在报告里为每一个跟踪事件使用两行而不是一行。缺省值为 **off**。

### **component=[on|off]**

在跟踪报告中显示完整的组件名称。缺省值为 **off**。

### **cpuid=[on|off]**

在跟踪报告里显示物理处理器号。缺省值为 **off**。

### **endtime=Seconds**

显示指定秒数之前记录的事件的跟踪报告数据。秒数可以是整数或有理数表示。如果此选项与 **starttime** 选项同时使用，可以显示特定的范围。

### **exec=[on|off]**

在跟踪报告里显示执行路径名。缺省值为 **off**。

### **filename=[on|off]**

显示检索到事件的文件名。如果文件名超过 40 个字符，则文件名左边将被截断。缺省值为 **off**。

### **hist=[on|off]**

记录每个挂钩标识遇到的实例数。此数据可用来生成直方图。缺省值为 **off**。此选项不可与任何其他选项同时运行。

### **ids=[on|off]**

在跟踪报告的第一列显示跟踪挂钩标识号。缺省值为 **on**。

### **pagesize=Number**

控制跟踪报告每一页的行数，是 0 到 500 范围内的一个整数。每一页都有列标题。当缺省值被设为 0 时，没有分页符。

### **pid=[on|off]**

显示跟踪报告里的进程标识。缺省值为 **off**。

### **reportedcpus=[on | off ]**

显示保留 CPU 的数量。该选项仅在多 CPU 跟踪中有意义，**trace -C**。例如，如果您在读一个拥有 4 个 CPU 的系统的报告，报告的 CPU 值从 4 变成 3，那您就知道不再会报告关于该 CPU 的挂钩了。

### **PURR=[ on | off ]**

指示 **trcrpt** 同任何时间戳一起显示 PURR。PURR 在任何时间戳之后显示。

如果 PURR 对跟踪的处理器无效，那么将显示所用时间而不是 PURR。如果 PURR 有效，或者 hook 的 **cpuid** 为未知但未受跟踪，则 PURR 字段包含星号 (\*)。

## -O 选项 (续)

### removedups=[on | off]

启用重复事件检测。DUPS 列中的计数显示报告中每个事件所代表的事件数。如果将该选项设置为 **off**，则将禁用重复事件检测。缺省值为 **on**。该选项仅在通过 **-m** 标志合并日志文件时有效。只有在从跟踪条目本身就能知道 CPU 标识时才能检测到重复条目，在必须推断 CPU 标识时则无法检测到重复条目。在以下情况中可以从条目获取到 CPU 标识：

- 轻量级内存跟踪
- 使用了 **trace -C** 命令选项的多 CPU 系统跟踪
- 使用 **-p** 选项启动的 64 位系统跟踪
- 64 位组件跟踪。

### starttime=*Seconds*

显示指定秒数之后记录的事件的跟踪报告数据。指定的秒数是从跟踪文件开始算起。秒数可以是整数或有理数表示。如果此选项与 **endtime** 选项同时使用，可以显示特定的范围。

### svc=[on|off]

显示跟踪报告中系统调用的值。缺省值为 **off**。

### tid=[on|off]

显示跟踪报告里的进程标识。缺省值为 **off**。

### timestamp=[0|1|2|3]

控制对时间戳记的报告，该时间戳记与跟踪报告中的事件相关联。它的值可能是：

- 0** 用时自跟踪启动之时起开始计时，记录距前一次事件的时间变化量。用时以秒计而时间变化量则以毫秒计。两个值都报告最近的时间（以纳秒计）。这是缺省值。
- 1** 短期用时。只报告从跟踪启动之时起的用时（以秒计）。用时报告最近的时间（以微秒计）。
- 2** 微秒时间变化量。这就像 0，除了间隔时间是以微妙计以外，报告最近的时间（以微秒计）。
- 3** 没有时间戳记。

### -p *List*

为 *List* 变量指定的每一个事件报告进程标识。*List* 变量可以是进程标识列表或进程名列表。以数字字符开头的列表项被认为是进程标识。列表项可以用逗号隔开，或包含在双引号中然后用逗号或空格隔开。

### -r

输出未格式化（原始）的跟踪条目，并将跟踪日志的内容逐条写入标准输出。使用 **-h** 和 **-r** 标志排除报头。要在一个多 CPU 跟踪中获取 CPU 原始报告，请使用 **-r** 和 **-C** 标志。

### -s *Date*

用指定日期或指定日期之前的条目启动报告时间。*Date* 变量的格式为 *mmddhhmmssyy*（月、日、时、分、秒以及年）。只有当开始和停止跟踪数据收集时，才在跟踪数据里记录日期和时间。如果您在一次跟踪对话期间多次停止和重新启动跟踪数据收集，每一次启动或停止跟踪数据收集时都会记录下日期和时间。将此标志与 **-e** 标志联合使用将跟踪限制在特定时间间隔期所收集的数据上。

**注：****-e** 和 **-s** 标志仅对没有使用 **trace -C** 标志收集的跟踪日志文件有效。

### -t *File*

将 *File* 变量所指定的文件用作模板文件。缺省为 */etc/trcfmt* 文件。

### -T *List*

将报告限制在 *List* 参数所指定的内核线程标识。列表项是内核线程标识，使用逗号分隔，或括在双引号中，然后用逗号或空格分隔。以内核线程标识开头的列表将报告的内容限制为列表中的所有内核线程标识。以！（感叹号）后跟内核线程标识作为开头的列表将报告的内容限制为不在列表中的所有内核线程标识。

- v 在文件被打开时显示文件名。更改为详细设置。
- x 显示执行路径名和系统调用值。

## 示例

1. 要格式化跟踪日志文件并显示结果，请输入：  
`trcrpt | qprt`
2. 要将跟踪报告发送给 `/tmp/newfile` 文件，请输入：  
`trcrpt -o /tmp/newfile`
3. 要显示跟踪报告中的进程标识和执行路径名，请输入：  
`trcrpt -0 pid=on,exec=on`
4. 要创建跟踪标识直方图数据，请输入：  
`trcrpt -0 hist=on`
5. 要产生所有事件组列表，请输入：  
`trcrpt -G`

此报告的格式显示在 `trcevgrp` 命令下。

6. 要从公共缓冲区和特殊缓冲区生成连续的 LMT 报告，请指定：  
`trcrpt -M all`
7. 在上面的示例中，如果 LMT 文件驻留在 `/tmp/mydir` 中而我们希望合并 LMT 跟踪，则请指定：  
`trcrpt -m -M all:/tmp/mydir`
8. 要将系统跟踪与 `sddisk.hdisk0` 组件跟踪合并，请指定：  
`trcrpt -m -l sddisk.hdisk0 /var/adm/ras/trcfile`
9. 要将 LMT 和系统跟踪合并同时不清除重复事件，请指定：  
`trcrpt -0 removedups=off -m -M all /var/adm/ras/trcfile`
10. 要将 `/tmp/mydir` 中的所有组件跟踪和缺省 LMT 目录中的 LMT 跟踪合并，并且显示每个跟踪事件的源文件，则请指定：  
`trcrpt -0 filename=on -m -M all /tmp/mydir`

注：这等同于：

```
trcrpt -0 filename=on -m -M all -l all:/tmp/mydir
```

注：如果跟踪来自 64 位内核，则将除去重复条目。然而，在 32 位内核中，因为不知道组件跟踪中条目的 CPU 标识，所以不会除去重复条目。

## 文件

|                                   |                             |
|-----------------------------------|-----------------------------|
| <code>/usr/bin/trcrpt</code>      | 包含 <code>trcrpt</code> 命令。  |
| <code>/var/adm/ras/trcfile</code> | 包含缺省日志文件。                   |
| <code>/var/adm/ras/mtrcdir</code> | 缺省 LMT 转储目录的位置。             |
| <code>/var/adm/ras/trc_ct</code>  | 缺省 CT <sup>®</sup> 转储目录的位置。 |
| <code>/etc/trcfmt</code>          | 包含跟踪格式文件。                   |

## 相关信息

**ctctrl** 命令、**trcctl** 命令、**trcdead** 命令、**trcevgrp** 命令、**trcnm** 命令、**trcstop** 命令和 **trcupdate** 命令。

**trace** 守护程序。

**trcfmt** 文件格式。

《AIX 5L V5.3 通用编程概念: 编写并调试程序》中的『跟踪设施概述』。

《性能管理》中的『监视和调节命令和子例程』。

AIX 5L Version 5.3 Kernel Extensions and Device Support Programming Concepts 中的 Debug and Performance Tracing。

---

## trcstop 命令

### 用途

停止跟踪功能

### 语法

**trcstop**

### 描述

**trcstop** 命令结束跟踪会话。

使用系统管理界面工具 (SMIT) 运行 **trcstop** 命令。要使用 SMIT, 输入:

```
smit trcstop
```

### 示例

要终止跟踪后台进程, 请输入:

```
trcstop
```

### 文件

**/usr/bin/trcstop**                    包含 **trcstop** 命令

## 相关信息

**trcrpt** 命令。

**trace** 守护程序。

《AIX 5L V5.3 通用编程概念: 编写并调试程序》中的『跟踪设施概述』。

---

## trcupdate 命令

### 用途

添加、替换或删除跟踪报告格式模板。

### 语法

```
trcupdate [ -o ] [ -t File ] [ -v ] [ -x IDList ] [ File ]
```

### 描述

**trcupdate** 命令添加、替换或删除 `/etc/trcfmt` 或 `/etc/trcfmt.Z` 文件中的跟踪报告格式模板。使用 `/etc/trcfmt.Z` 文件时，**trcupdate** 命令解压缩、更新和再压缩文件。**trcupdate** 命令创建“撤销”文件并在指定目录中命名为 `File.undo.trc`。

**trcupdate** 命令为文件名添加 `.trc` 后缀并从该文件中读取更新命令。如果指定 `-o`（覆盖）标志，撤销文件输入到 **trcupdate** 命令。指定 `-o` 标志，**trcupdate** 命令撤销文件前面所做改动。

每个模板第一个字段包含一个运算符：

- + 加号表示要添加或替换模板。此运算符后的字段包含要替换的模板。
- 减号表示要删除模板。此运算符后的字段包含要删除模板的挂钩标识。操作按出现顺序执行。

对 **trcupdate** 命令的输入应包含以下内容作为第一行：

```
* /etc/trcfmt
```

以下是一个样本跟踪文件：

```
* /etc/trcfmt
+ 15A 1.0 new_fmt
- 1B3
- A14
```

添加或替换时，**trcupdate** 命令将每一输入模板的版本号和相同挂钩标识模板的版本号相比较。如果输入模板的版本号晚于现存模板版本，**trcupdate** 命令以输入模板替换旧模板。如果模板不存在，则在文件中添加输入模板。

更新文件中检测到语法错误时，**trcupdate** 命令不修改 `/etc/trcfmt` 文件。

### 标志

- `-o` 不验证任一模板版本号而以输入模板覆盖旧模板。
- `-t File` 指定一个文件而不是 `/etc/trcfmt` 或 `/etc/trcfmt.Z` 文件用作模板文件。
- `-v` 每个文件打开时显示文件名。
- `-x IDList` 从模板文件中抽取 `IDList` 中的指定模板并写到标准输出。`IDList` 参数列出挂钩标识。

### 安全性

访问控制：无，但必须具有对正在更改的模板文件的写权限。缺省值为 `/etc/trcfmt`。

### 示例

1. 要添加模板，请输入：

```
trcupdate
* /etc/trcfmt
+ 15A 1.0 new_fmt
```

2. 要删除模板, 请输入:

```
trcupdate
* /etc/trcfmt
- 15A 1.0 new_fmt
```

3. 要替换模板, 请输入:

```
trcupdate
* /etc/trcfmt
+ 15A 1.0 new_fmt
```

## 文件

**/usr/bin/trcupdate**

包含 **trcupdate** 命令。

**/etc/trcfmt**

包含跟踪格式文件。

**/usr/include/sys/trcmacros.h**

定义 **trchhook** 和 **utrchhook** 宏。

## 相关信息

**trcdead** 命令和 **trcrpt** 命令。

**trace** 守护程序。

**trcfmt** 文件格式。

《AIX 5L V5.3 通用编程概念: 编写并调试程序》中的『跟踪设施概述』。

---

## troff 命令

### 用途

为在排版设备上显示而对文本进行格式化。

### 语法

```
troff [ -a ] [ -i ] [ -q ] [ -z ] [ -F Directory ] [ -n Number ] [ -o List ] [ -r ANumber ] [ -s Number ] [ -T Name ] [ -mm | -me | -mptx | -ms | -man | -mv ] [ -M Media ] [ File ... | - ]
```

### 描述

**troff** 命令读取一个或多个文件并将文本格式化, 以便在照排机或者其他兼容的设备上显示。然后, 需要使用后处理器来对 **troff** 命令到目标设备的输出进行后处理。请参阅下文的示例。

如果没有指定任何文件, 或者最后一个参数不是 **-** (减号) 标志, 则缺省值就是读取标准输入。

对于 3812、3816 和 Hewlett-Packard LaserJet 系列 II 打印机来说, 缺省的字体是打印机本机字体。这些打印机也可以使用其他字体, 这些字体可以通过使用 **troff .fp** 伪指令来装入。这些字体储存在主机的 **/usr/lib/font/devPrinter/bitmaps** 目录下, 可以在需要的时候下载到打印机。

### 字型

以四种样式提供三种不同字型。下表展示了字型、样式和 **troff** 命令用于存取字体的名称之间的关系。

注: 本套字体集基于 Donald E Knuth 开发的计算机现代字母格式。(请参阅 Knuth,Donald: *Computer Modern Typefaces*. Addison-Wesley, 1986。)

| 字型               | 常规 | 斜体 | 粗体 | 斜体 |
|------------------|----|----|----|----|
| Roman            | cr | cR | Cr | CR |
| Sans Serif       | cs | cS | Cs | CS |
| Typewriter       | ct | cT | Ct | CT |
| troff special sp |    |    |    |    |

15 种标准 troff 大小提供了所有这些字体: 6, 7, 8, 9, 10, 11, 12, 14, 16, 18, 20, 22, 24, 28 和 36 磅。

例如 `.fp 1 Cr` 将罗马粗体字体装入位置 1。

注: 在对 **troff** 命令的输入中, **.tl** 请求不能在生成第一个中断的请求之前使用。

## 标志

|                    |                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|--------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-a</b>          | 将结果的可显示 ASCII 近似值发送到标准输出。                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| <b>-FDirectory</b> | 从 Directory/ <b>devName</b> 目录而不是缺省的 <b>/usr/lib/font/devName</b> 目录 (此目录下的 <i>Name</i> 由 <b>-T</b> 标志指定) 存取字体信息。                                                                                                                                                                                                                                                                                                                                            |
| <b>-i</b>          | 在没有其他文件后读取标准输入。                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| <b>-M Media</b>    | 指定纸张大小, 进而确定纸上的可成像面积。Media 变量的有效值为:<br><b>A4</b> 指定纸张大小为 8.3 X 11.7 英寸 (210 X 297 厘米)。<br><b>A5</b> 指定纸张大小为 5.83 X 8.27 英寸 (148 X 210 厘米)。<br><b>B5</b> 指定纸张大小为 6.9 X 9.8 英寸 (176 X 250 厘米)。<br><b>EXEC</b> 指定纸张大小为 7.25 X 10.5 英寸 (184.2 X 266.7 厘米)。<br><b>LEGAL</b><br>指定纸张大小为 8.5 X 14 英寸 (215.9 X 355.6 厘米)。<br><b>LETTER</b><br>指定纸张大小为 8.5 X 11 英寸 (215.9 X 279.4 厘米)。此为缺省值。<br>注: Media 变量不区分大小写。                                                    |
| <b>-nNumber</b>    | 用 Number 变量指定的值为第一显示页编号。                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| <b>-oList</b>      | 只显示 List 变量指定的页, 这样的页中包含一个以逗号隔开的页号和范围的列表: <ul style="list-style-type: none"><li>• 开始页-结束页范围表示显示从开始页到结束页之间的页数。例如: 9-15 表示显示第 9 页到第 15 页。</li><li>• 初始-结束页表示从开头显示到结束页。</li><li>• 最终开始页-表示从开始页显示到结尾。</li><li>• 显示页的编号和显示范围联合使用, 来显示指定页。例如: -3,6-8,10,12- 显示从开头到第 3 页, 从第 6 页到第 8 页, 第 10 页以及第 12 页到结尾。<br/>注: 在管道中使用该标志 (例如: 含有一个或多个 <b>pic</b>, <b>eqn</b> 或 <b>tbl</b> 命令) 时, 如果文档的最后一页没有在 List 变量中指定, 用户可能会接收到中断管道消息。不过, 该中断管道消息并不指示有任何问题, 可以忽略不计。</li></ul> |
| <b>-q</b>          | 调用 <b>.rd</b> 请求的同步输入和输出方式。                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| <b>-rANumber</b>   | 将由 A 变量指定的寄存器设置为指定的编号。A 变量值必须具有单字符的 ASCII 名称。                                                                                                                                                                                                                                                                                                                                                                                                                |
| <b>-sNumber</b>    | 生成输出, 使排版机停止每一个指定页编号。                                                                                                                                                                                                                                                                                                                                                                                                                                        |

|                       |                                                                                                                                                                      |
|-----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-T</b> <i>Name</i> | 为指定的打印设备准备输出。照排机或其他兼容打印设备在操作系统国际扩展字符上使用以下 <i>Name</i> 变量。缺省值为 <b>ibm3816</b> 。<br><b>注：</b> 如果设备不支持指定的磅值，您就会收到bad point size的消息。 <b>troff</b> 命令会使用最接近的有效磅值来继续进行格式化。 |
| <b>canonls</b>        | Canon Lasershot LBP-B406S/D/E,A404/E,A304E。                                                                                                                          |
| <b>ibm3812</b>        | 3812 页式打印机 II。                                                                                                                                                       |
| <b>ibm3816</b>        | 3816 页式打印机。                                                                                                                                                          |
| <b>hplj</b>           | Hewlett-Packard LaserJet II。                                                                                                                                         |
| <b>ibm5585H-T</b>     | 5585-H01 繁体中文语言支持。                                                                                                                                                   |
| <b>ibm5587G</b>       | 5587-G01, 5584-H02, 5585-H01, 5587-H01 和 5589-H01 日语汉字打印机，支持多字节语言。                                                                                                   |
| <b>psc</b>            | PostScript 打印机。                                                                                                                                                      |
| <b>X100</b>           | AIXwindows 显示。                                                                                                                                                       |
|                       | <b>注：</b> 也可以将 <b>TYPESETTER</b> 环境变量设置为前述值之一，而不使用 <b>troff</b> 命令中的 <b>-T Name</b> 标志。                                                                              |
| <b>-man</b>           | 选择 <b>man</b> 宏处理软件包。                                                                                                                                                |
| <b>-me</b>            | 选择 <b>me</b> 宏处理软件包。                                                                                                                                                 |
| <b>-mm</b>            | 选择 <b>mm</b> 宏处理软件包。                                                                                                                                                 |
| <b>-mptx</b>          | 选择 <b>mptx</b> 宏处理软件包。                                                                                                                                               |
| <b>-ms</b>            | 选择 <b>ms</b> 宏处理软件包。                                                                                                                                                 |
| <b>-mv</b>            | 选择 <b>mv</b> 宏处理软件包。                                                                                                                                                 |

请参阅《宏软件包之格式化工具》一书，以了解更多有关宏的信息。

- z** 只打印由 **.tm**（工作站消息）请求生成的消息。
- 强制从标准输入中读取输入。

## 环境变量

**TYPESETTER** 包含特定打印设备的信息。

## 示例

以下是一个 **troff** 命令的示例：

```
troff -Tibm3812 File | ibm3812 | qprt
```

## 宏软件包之格式化工具

以下的宏软件包是《文本格式化系统中的格式化工具》的一部分，在接下来的页中有更详细信息的描述：

- man** 使用户可以依靠联机手册页来创建自己的手册页。
- me** 提供格式化页的宏。



|             |                                             |
|-------------|---------------------------------------------|
| <b>mm</b>   | 使用 <b>nroff</b> 和 <b>troff</b> 格式化程序来格式化文档。 |
| <b>mptx</b> | 格式化置换索引。                                    |
| <b>ms</b>   | 为不同样式的文章、论文和书籍提供格式化程序实用工具。                  |
| <b>mv</b>   | 使用 <b>troff</b> 命令对英文视图和幻灯片进行排版。            |

## 用于 **nroff** 和 **troff** 命令的 **man** 宏软件包

**man** 宏软件包的提供，使用户可以依靠用 **nroff** 命令或 **troff** 命令处理过的联机手册页来创建自己的手册页。

**man** 宏软件包与 **nroff** 命令或 **troff** 命令同时使用。

**man** 宏软件包内部包含特殊宏、字符串和数字寄存器，还有以下的格式化宏、字符串和寄存器的列表。除了由 **troff** 命令和 **d**、**m** 和 **y** 数字寄存器预定义的名称外，所有这样的内部名称都是 *SymbolAlpha* 格式的，其中 *Symbol* 是 **)**、**] 或 **}** 之一，*Alpha* 为任意一个字母数字字符。**

**man** 宏软件包只使用罗马字体。如果某项的输入文本包含使用其他字体的请求（如 **.I** 格式化宏、**.RB** 请求或 **\fl** 请求），就必须安装对应的字体。

### 格式化宏

以下宏用来修改使用 **man** 宏软件包格式化的手册页的特性。

在每个段落之前和处理字体和大小设置的宏之后（如 **.I**、**.SM** 和 **.B** 格式化宏），输入字体和大小都将复位为缺省值。

除了 **.DT** 和 **.TH** 格式化宏以外，其他格式化宏都不能够使用也不能够设置制表符停止位。

#### **.B** [*Text*]

使文本字体变为粗体。

*Text* 变量代表最多 6 个单词；使用 “ ”（双引号）以包含单词中的空格字符。如果该变量是空的，可以在下一个包含待打印文本的输入文本行应用该处理。例如：使用 **.I** 格式化宏使整个行变为斜体，或使用 **.SM** 和 **.B** 格式化宏使整个行的文字变为小粗体字体。缺省情况下，**nroff** 命令关闭连字符处理，而在 **troff** 命令中可以使用。

**.DT** **nroff** 命令中每隔 5 ens 恢复一次缺省的制表符设置；在 **troff** 命令中每隔 7.2 ens 恢复一次。

#### **.HP** [*Indent*]

使得段落开头为悬挂式缩进，缩进尺寸由 *Indent* 变量指定。

如果省略 *Indent* 变量，则使用前一次设置的 *Indent* 值。此值的缺省值（对于 **nroff** 命令为 5 ens，对于 **troff** 命令为 7.2 ens）由 **.TH**、**.P** 和 **.RS** 格式化宏来设置；值的恢复由 **.RE** 格式化宏来进行。*Indent* 变量的缺省单位是 ens。

#### **.I** [*Text*]

使文本字体变为斜体。

*Text* 变量代表最多 6 个单词；使用 “ ”（双引号）以包含单词中的空格字符。如果该变量是空的，可以在下一个包含待打印文本的输入文本行应用该处理。例如：使用 **.I** 格式化宏使整个行变为斜体，或使用 **.SM** 和 **.B** 格式化宏使整个行的文字变为小粗体字体。缺省情况下，**nroff** 命令关闭连字符处理，而在 **troff** 命令中可以使用。

#### **.IP** [*Tag*] [*Indent*]

与 **.TP** *Indent* 宏对 *Tag* 变量的操作的效果是一样的；如果 *Tag* 变量的值是 **NULL**，则段落的开头是缩进的。该宏通常用来使段落缩进而不需再使用制表符。

如果省略 *Indent* 变量，则使用前一次使用的 *Indent* 值。该值（对于 **nroff** 命令为 5 ens，对于 **troff** 命令为 7.2 ens），由 **.TH**、**.P** 和 **.RS** 格式化宏来设置为缺省值，由 **.RE** 格式化宏进行恢复。*Indent* 变量的缺省单位是 ens。

**.P** 用正常字体、磅大小和缩进开始段落。**.PP** 宏就是 **mm** 宏软件包中的 **.P** 宏。

**.PD** [*Number*]

将段落间距设置为 *Number* 参数指定的垂直间距数。缺省的 *Number* 变量值在 **troff** 命令中是 0.4v，在 **nroff** 命令中是 1v。

**.PM** [*Indicator*]

设置的专利标记如下：

| 指示符      | 标记      |
|----------|---------|
| <b>P</b> | 专用      |
| <b>N</b> | 注意事项    |
| 无指定的指示符  | 关闭专利标记。 |

**.RE** [*Number*]

结束由 *Number* 变量指定的缩进级别位置的相对缩进 (**.RS**)。如果省略了 *Number* 变量值，则返回到最近的低一级的缩进级别。

**.RI** *Character1Character2...*

将罗马体的 *Character1* 和斜体的 *Character2* 并置；改变两种字体，得到 6 种 *Character1Character2* 集。类似的可以改变罗马体、斜体和粗体，形成任两种组合在一起的宏包括：**.IR**、**.RB**、**.BR**、**.IB** 和 **.BI** 宏。

**.RS** [*Indent*]

增加相关缩进量（初始为零）。将所有输出按照 *Indent* 变量指定的缩进量从左空白缩进额外的单元数。

如果省略了 *Indent* 变量，则使用前一 *Indent* 值。该值（对于 **nroff** 命令为 5 ens，对于 **troff** 命令为 7.2 ens）由 **.TH**、**.P** 和 **.RS** 格式化宏来设置为缺省值，由 **.RE** 格式化宏来进行恢复。*Indent* 变量的缺省单位是 ens。

**.SH** [*Text*]

放置副标题文本。

*Text* 变量代表最多 6 个单词；使用 “ ”（双引号）以包含单词中的空格字符。如果该变量是空的，可以在下一个包含待打印文本的输入文本行应用该处理。例如：使用 **.I** 格式化宏使整个行变为斜体，或使用 **.SM** 和 **.B** 格式化宏使整个行的文字变为小粗体字体。缺省情况下，**nroff** 命令关闭连字符处理，而在 **troff** 命令中可以使用。

**.SM** [*Text*]

使文本字大小比缺省值小 1 磅。

*Text* 变量代表最多 6 个单词；使用 “ ”（双引号）以包含单词中的空格字符。如果该变量是空的，可以在下一个包含待打印文本的输入文本行应用该处理。例如：使用 **.I** 格式化宏使整个行变为斜体，或使用 **.SM** 和 **.B** 格式化宏使整个行的文字变为小粗体字体。缺省情况下，**nroff** 命令关闭连字符处理，而在 **troff** 命令中可以使用。

**.SS** [*Text*]

放置副标题文本。

*Text* 变量代表最多 6 个单词；使用 “ ”（双引号）以包含单词中的空格字符。如果该变量是空的，可以在下一个包含待打印文本的输入文本行应用该处理。例如：使用 **.I** 格式化宏使整个行变为斜体，或使用 **.SM** 和 **.B** 格式化宏使整个行的文字变为小粗体字体。缺省情况下，**nroff** 命令关闭连字符处理，而在 **troff** 命令中可以使用。

### **.TH** [*Title*][*Section*][*Comme*][*Name*]

设置标题和条目标题。该宏调用 **.DT** 格式化宏。

| 变量                | 标记     |
|-------------------|--------|
| <i>Title</i>      | 标题     |
| <i>Section</i>    | 节号     |
| <i>Commentary</i> | 外部注释   |
| <i>Name</i>       | 新的手册名。 |

注：如果 **.TH** 格式化宏值中包含字符空格，但是没有使用 “ ”（双引号）将其扩起来的话，输出会显示不规则的点。

### **.TP** [*Indent*]

以悬挂标记开始缩进段落。下一个包含文本的输入行就是标记。如果此标记不合适，会在另一行上被打印出来。

如果省略 *Indent* 变量，则使用前一次使用的 *Indent* 值。该值（对于 **nroff** 命令为 5 ens，对于 **troff** 命令为 7.2 ens）由 **.TH**、**.P** 和 **.RS** 格式化宏来设置为缺省值，由 **.RE** 格式化宏来进行恢复。*Indent* 变量的缺省单位是 ens。

## 字符串

**\\*R** 为 **nroff** 命令添加商标 (Reg.)，为 **troff** 命令添加注册商标符号。  
**\\*S** 改为缺省类型大小。  
**\\*(Tm** 添加商标指示符。

## 寄存器

**IN** 相对于副标题缩进左空白。缺省值为 **troff** 命令为 7.2 ens，**nroff** 命令为 5 ens。  
**LL** 行长度包括由 **IN** 寄存器指定的值。  
**PD** 当前的段落间距。

## 标志

**-rs1** 将缺省的页面大小由 8.5 × 11 英寸减为 6 × 9 英寸，缺省的文本面积由 6.5 × 10 英寸减为 4.75 × 8.375 英寸。该标记还将缺省的类型大小由 10 磅减为 9 磅，缺省的垂直行间距由 12 磅减为 10 磅。

## 示例

- 要对文件 `your.book` 进行处理，并在将其格式化后输出至本地行式打印机 `qprt`，请输入：  

```
nroff -Tlp -man your.book | qprt -dp
```
- 要对包含表格的文件 `my.book` 和 `dept.book` 进行处理，并在将其格式化后输出至本地行式打印机 `qprt`，请输入：  

```
tbl my.book dept.book | nroff -Tlp -man | col -Tlp | qprt -dp
```

注：在输出被发送到 `qprt` 之前，首先要由 `col` 命令对其进行过滤，以便对 `tbl` 命令使用过的反向换行进行处理。

3. 要处理包含图片、图形和表的文件 `group`，并且准备已编排格式的输出以在 IBM 3816 打印机上进行处理，请输入：

```
grap group | pic | tbl | troff -Tibm3816 -man \  
| ibm3816 | qprt -dp
```

注：

1. 如果用 `man` 宏软件包创建的手册页意在用于联机设备上，则应避免使用需要 `troff` 命令的组件，如 `grap` 或 `pic` 命令。
2. `grap` 命令在 `pic` 命令之前运行，因为它是 `pic` 命令的预处理程序；否则不能进行正确的格式化。
3. `col` 命令不需要作为 `tbl` 命令的过滤器；排版后的文档不需要反向换行。

## 用于 `nroff` 和 `troff` 命令的 `me` 宏软件包

`nroff` 和 `troff` 命令的宏定义的 `me` 软件包以不同格式为技术论文提供了格式化实用程序。某些情况下，可能需要 `col` 命令对 `nroff` 输出进行后处理。

宏请求在下一节的 `me` 请求中定义。许多 `nroff/troff` 请求在同该软件包合用的时候可能会产生不可预测的结果。但是，以下请求可以在第一个 `.pp` 请求后使用：

|                                   |                                                              |
|-----------------------------------|--------------------------------------------------------------|
| <code>.bp</code>                  | 开始新的页。                                                       |
| <code>.br</code>                  | 在此处中断输出行。                                                    |
| <code>.ce [Number]</code>         | 居中对齐下一个指定行号。缺省值为 1。                                          |
| <code>.ls [Number]</code>         | 设置行间距。如果 <code>Number</code> 被设为 1，文本为单倍行距；如果该值被设为 2，则为双倍行距。 |
| <code>.na</code>                  | 使右边界不对齐。                                                     |
| <code>.sp [Number]</code>         | 插入指定数量的空白行。                                                  |
| <code>.sz [+]<i>Number</i></code> | 增加指定数量的磅值。                                                   |
| <code>.ul [Number]</code>         | 为下一个指定的行号加下划线。缺省值为 1。                                        |

用于方程式和表格上的 `eqn`、`neqn`、`refer` 和 `tbl` 命令的预处理器输出可以当作输入使用。

## `me` 请求

以下列表中包括在 `me` 宏中所有有效的宏、字符串和数字寄存器。还包括选中的 `troff` 命令、寄存器和各种功能。

|                               |                                                              |
|-------------------------------|--------------------------------------------------------------|
| <code>\(space)</code>         | 定义不可插入式空格（ <code>troff</code> 命令的内置功能）。                      |
| <code>\“</code>               | 对行结束的注释（ <code>troff</code> 命令的内置功能）。                        |
| <code>\*#</code>              | 表示可选的、延迟的文本标记字符串。                                            |
| <code>\\$<i>Number</i></code> | 内插由 <code>Number</code> 变量指定的值（ <code>troff</code> 命令的内置功能）。 |
| <code>\n\$0</code>            | 定义节的深度（数字寄存器）。                                               |
| <code>.\$0</code>             | 在打印节标题后启动（用户可以自定义的宏）。                                        |
| <code>\n\$1</code>            | 定义第一节的节号（数字寄存器）。                                             |
| <code>.\$1</code>             | 在打印深度为 1 的节之前启动（用户可以自定义的宏）。                                  |
| <code>\n\$2</code>            | 定义第二节的节号（数字寄存器）。                                             |
| <code>.\$2</code>             | 在打印深度为 2 的节之前启动（用户可以自定义的宏）。                                  |
| <code>\n\$3</code>            | 定义第三节的节号（数字寄存器）。                                             |
| <code>.\$3</code>             | 在打印深度为 3 的节之前启动（用户可以自定义的宏）。                                  |
| <code>\n\$4</code>            | 定义第四节的节号（数字寄存器）。                                             |
| <code>.\$4</code>             | 在打印深度为 4 的节之前启动（用户可以自定义的宏）。                                  |
| <code>\n\$5</code>            | 定义第五节的节号（数字寄存器）。                                             |

|                               |                                                           |
|-------------------------------|-----------------------------------------------------------|
| <code>.\$5</code>             | 在打印深度为 5 的节之前启动 (用户可以自定义的宏)。                              |
| <code>\n\$6</code>            | 定义第六节的节号 (数字寄存器)。                                         |
| <code>.\$6</code>             | 在打印深度为 6 的节之前启动 (用户可以自定义的宏)。                              |
| <code>.\$C</code>             | 在章的开头被调用 (用户可以自定义的宏)。                                     |
| <code>.\$H</code>             | 表示文本的页眉 (用户可以自定义的宏)。                                      |
| <code>\n(\$R</code>           | 定义显示出来的相对垂直间距 (按缺省值定义的数字寄存器, 不建议更改)。                      |
| <code>\n(\$c</code>           | 定义当前栏的页眉 (数字寄存器)。                                         |
| <code>.\$c</code>             | 打印章标题 (按缺省值定义的宏; 不建议更改)。                                  |
| <code>\n(\$d</code>           | 表示延迟的文本号 (数字寄存器)。                                         |
| <code>\n(\$f</code>           | 表示脚注号 (数字寄存器)。                                            |
| <code>.\$f</code>             | 打印页脚 (按缺省值定义的宏; 不建议更改)。                                   |
| <code>.\$h</code>             | 打印页眉 (按缺省值定义的宏; 不建议更改)。                                   |
| <code>\n(\$i</code>           | 定义段落的基准缩进 (数字寄存器)。                                        |
| <code>\n(\$l</code>           | 定义列宽 (数字寄存器)。                                             |
| <code>\n(\$m</code>           | 表示有效的栏数 (数字寄存器)。                                          |
| <code>\*(\$n</code>           | 表示节的名称 (字符串)。                                             |
| <code>\n(\$p</code>           | 定义已编号的段落数 (数字寄存器)。                                        |
| <code>.\$p</code>             | 打印节标题 (按缺省值定义的宏; 不建议更改)。                                  |
| <code>\n(\$r</code>           | 定义文本的相对垂直间距 (按缺省值定义的数字寄存器; 不建议更改)。                        |
| <code>\n(\$s</code>           | 定义列缩进 (数字寄存器)。                                            |
| <code>.\$s</code>             | 将脚注从文本中分离 (按缺省值定义的宏; 不建议更改)。                              |
| <code>\n%</code>              | 定义当前页的页号 (按缺省值定义的数字寄存器; 不建议更改)。                           |
| <code>\&amp;</code>           | 表示 0 宽度的字符; 用来隐藏控件 ( <b>troff</b> 命令的内置功能)。               |
| <code>\(XX</code>             | 内插由 <i>XX</i> 变量指定的特殊字符 ( <b>troff</b> 命令的内置功能)。          |
| <code>.(b</code>              | 开始块 (宏)。                                                  |
| <code>.(c</code>              | 开始居中对齐的块 (宏)。                                             |
| <br>                          |                                                           |
| <code>.(d</code>              | 开始延迟的文本 (宏)。                                              |
| <code>.(f</code>              | 开始脚注 (宏)。                                                 |
| <code>.(l</code>              | 开始列表 (宏)。                                                 |
| <code>.(q</code>              | 开始引用 (宏)。                                                 |
| <code>.(xIndex</code>         | 在指定的索引中开始索引项 (宏)。                                         |
| <code>.(z</code>              | 开始浮动保持 (宏)。                                               |
| <code>.)b</code>              | 结束块 (宏)。                                                  |
| <code>.)c</code>              | 结束居中对齐的块 (宏)。                                             |
| <code>.)d</code>              | 结束延迟的文本 (宏)。                                              |
| <code>.)f</code>              | 结束脚注 (宏)。                                                 |
| <code>.)l</code>              | 结束列表 (宏)。                                                 |
| <code>.)q</code>              | 结束引用 (宏)。                                                 |
| <code>.)x</code>              | 结束索引项 (宏)。                                                |
| <code>.)z</code>              | 结束浮动保持 (宏)。                                               |
| <code>\*String</code>         | 内插由 <i>String</i> 变量指定的值 ( <b>troff</b> 命令的内置功能)。         |
| <code>\*String1String2</code> | 内插由 <i>String1String2</i> 变量指定的值 ( <b>troff</b> 命令的内置功能)。 |
| <code>\**</code>              | 表示可选的脚注标记字符串。                                             |

|                           |                                                                                                                                                                                      |
|---------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>++mH</code>         | 定义论文章节的宏。由 <i>m</i> 变量指定的值定义的论文的各部分。 <i>m</i> 变量可以是以下值：                                                                                                                              |
| <b>C</b>                  | 定义章。                                                                                                                                                                                 |
| <b>A</b>                  | 定义附录。                                                                                                                                                                                |
| <b>P</b>                  | 定义初步信息，如摘要和目录。                                                                                                                                                                       |
| <b>B</b>                  | 定义书目。                                                                                                                                                                                |
| <b>RC</b>                 | 定义各章从每张的第一页开始重新编号。                                                                                                                                                                   |
| <b>RA</b>                 | 定义附录从第一页开始重新编号。                                                                                                                                                                      |
|                           | <i>H</i> 参数定义新页眉。如果页眉中含有空格，则该页眉必须加引号。如果您希望在页眉中包含章号，则请使用字符串 <code>\\n(ch</code> 。例如，要将附录编号为 A.1、A.2...，请输入： <code>++ RA '''\\n(ch.%'</code> 。每部分（例如章和附录）的编号应该在 <code>.+c</code> 请求之前。 |
| <code>.+cTitle</code>     | 开始章（或附录，例如像 <code>++</code> 宏所设置的那样）。由 <i>Title</i> 变量指定的值就是章的标题（宏）。                                                                                                                 |
| <code>\*</code>           | 表示下加符号（字符串）。                                                                                                                                                                         |
| <code>\-</code>           | 表示减号（ <code>troff</code> 命令的内置功能）。                                                                                                                                                   |
| <code>\*-</code>          | 表示 3/4 单位长度的破折号（字符串）。                                                                                                                                                                |
| <code>\0</code>           | 定义不可插入式的数字宽度的空格（ <code>troff</code> 命令的内置功能）。                                                                                                                                        |
| <code>.1c</code>          | 还原为单栏输出（宏）。                                                                                                                                                                          |
| <code>.2c</code>          | 开始双列输出（宏）。                                                                                                                                                                           |
| <code>\*</code>           | 表示变音符号（字符串）。                                                                                                                                                                         |
| <code>\&lt;</code>        | 打开下标（字符串）。                                                                                                                                                                           |
| <code>\&gt;</code>        | 结束下标（字符串）。                                                                                                                                                                           |
| <code>.EN</code>          | 结束等式。由 <code>eqn</code> 命令或 <code>neqn</code> 命令（宏）生成的等式后面的空格。                                                                                                                       |
| <code>.EQXY</code>        | 开始等式；中断并添加空格。由 <i>Y</i> 变量指定的值就是等式的值。可选的 <i>X</i> 变量的值可能是以下数中的任一个：                                                                                                                   |
| <b>I</b>                  | 缩进等式（缺省值）。                                                                                                                                                                           |
| <b>L</b>                  | 左对齐等式。                                                                                                                                                                               |
| <b>C</b>                  | 居中对齐等式（宏）。                                                                                                                                                                           |
| <code>\L'Distance'</code> | 表示指定距离的垂直绘线型函数（ <code>troff</code> 命令的内置功能）。                                                                                                                                         |
| <code>.PE</code>          | 结束 <code>pic</code> 图片（宏）。                                                                                                                                                           |
| <code>.PF</code>          | 结束带有回扫的 <code>pic</code> 图片（宏）。                                                                                                                                                      |
| <code>.PS</code>          | 启动 <code>pic</code> 图片（宏）。                                                                                                                                                           |
| <code>.TE</code>          | 结束表格（宏）。                                                                                                                                                                             |
| <code>.TH</code>          | 结束表格头（宏）。                                                                                                                                                                            |
| <code>.TS X</code>        | 开始表格。如果 <i>X</i> 变量的值是 <b>H</b> ，则该表格具有重复的标题（宏）。                                                                                                                                     |
| <code>\[</code>           | 开始上标（字符串）。                                                                                                                                                                           |
| <code>\n(\$</code>        | 定义宏的选项编号（按缺省值定义的数字寄存器；不建议更改）。                                                                                                                                                        |
| <code>\n(i</code>         | 表示当前的缩进（按缺省值定义的数字寄存器；不建议更改）。                                                                                                                                                         |
| <code>\n(l</code>         | 表示当前的行长（按缺省值定义的数字寄存器；不建议更改）。                                                                                                                                                         |
| <code>\n(s</code>         | 表示当前的磅值（按缺省值定义的数字寄存器；不建议更改）。                                                                                                                                                         |
| <code>\*(4</code>         | 表示重音符（字符串）。                                                                                                                                                                          |
| <code>\*(</code>          | 表示抑音符（字符串）。                                                                                                                                                                          |
| <code>\(4</code>          | 表示重音符（ <code>troff</code> 命令的内置功能）。                                                                                                                                                  |
| <code>\(</code>           | 表示抑音符（ <code>troff</code> 命令的内置功能）。                                                                                                                                                  |
| <code>\]</code>           | 结束上标（字符串）。                                                                                                                                                                           |
| <code>\^</code>           | 表示 1/12 单位长度的窄空格（ <code>troff</code> 命令的内置功能）。                                                                                                                                       |
| <code>\^^</code>          | 表示插入记号（字符串）。                                                                                                                                                                         |

|                  |                           |                                                                                           |
|------------------|---------------------------|-------------------------------------------------------------------------------------------|
| <code>.ac</code> | <code>AuthorNumber</code> | 形成 ACM 格式的输出。 <code>Author</code> 变量指定作者名。 <code>Number</code> 变量指定总页数。必须在第一次初始化之前使用 (宏)。 |
| <code>.ad</code> |                           | 设置文本对齐方式 (宏)。                                                                             |
| <code>.af</code> |                           | 为寄存器分配格式 (宏)。                                                                             |
| <code>.am</code> |                           | 附加到宏 (宏)。                                                                                 |
| <code>.ar</code> |                           | 设置页号为阿拉伯数字 (宏)。                                                                           |
| <code>.as</code> |                           | 附加到字符串 (宏)。                                                                               |
| <code>.b</code>  | <code>X</code>            | 用粗体字打印 <code>X</code> 变量指定的值。如果省略了 <code>X</code> 变量, 则使用粗体字 (宏)。                         |
| <code>.ba</code> | <code>+Number</code>      | 通过指定的 <code>Number</code> 值增加基准缩进。在普通文本, 如段落上设置缩进 (宏)。                                    |
| <code>.bc</code> |                           | 开始新的一列 (宏)。                                                                               |
| <code>.bi</code> | <code>X</code>            | 仅在非填充方式下, 用粗斜体打印 <code>X</code> 参数指定的值。如果没有使用 <code>X</code> 参数, 跟随粗斜体文本 (宏)。             |
| <code>\n</code>  | <code>(bi)</code>         | 显示块缩进 (数字寄存器)。                                                                            |
| <code>.bl</code> |                           | 请求空行, 即使在页首 (宏)。                                                                          |
| <code>\n</code>  | <code>(bm)</code>         | 设置底部标题的页边距 (数字寄存器)。                                                                       |
| <code>.bp</code> |                           | 开始页 (宏)。                                                                                  |
| <code>.br</code> |                           | 设置中断; 启动换行 (宏)。                                                                           |
| <code>\n</code>  | <code>(bs)</code>         | 显示块的前或后间距 (数字寄存器)。                                                                        |
| <code>\n</code>  | <code>(bt)</code>         | 块保持阈值 (数字寄存器)。                                                                            |
| <code>.bu</code> |                           | 开始着重标记段落 (宏)。                                                                             |
| <code>.bx</code> | <code>X</code>            | 只将框中的 <code>X</code> 变量指定的值以非填充方式打印出来 (宏)。                                                |
| <code>\c</code>  |                           | 继续输入 ( <b>troff</b> 命令的内置功能)。                                                             |
| <code>.ce</code> |                           | 使行居中对齐 (宏)。                                                                               |
| <code>\n</code>  | <code>(ch)</code>         | 定义当前章号 (数字寄存器)。                                                                           |
| <code>.de</code> |                           | 定义宏 (宏)。                                                                                  |
| <code>\n</code>  | <code>(df)</code>         | 显示字体 (数字寄存器)。                                                                             |
| <code>.ds</code> |                           | 定义字符串 (宏)。                                                                                |
| <code>\n</code>  | <code>(dw)</code>         | 定义当前是星期几 (数字寄存器)。                                                                         |
| <code>\*</code>  | <code>(dw)</code>         | 定义当前是星期几 (字符串)。                                                                           |
| <code>\n</code>  | <code>(dy)</code>         | 定义当前的日期 (数字寄存器)。                                                                          |
| <code>\e</code>  |                           | 表示可打印的 <code>\</code> (反斜杠) 的版本 ( <b>troff</b> 命令的内置功能)。                                  |
| <code>.ef</code> | <code>'X'Y'Z'</code>      | 将偶数页的页脚设置为 <code>XYZ</code> 变量指定的值 (宏)。                                                   |
| <code>.eh</code> | <code>'X'Y'Z'</code>      | 将偶数页的页眉设置为 <code>XYZ</code> 变量指定的值 (宏)。                                                   |
| <code>.el</code> |                           | 指定 <code>if/else</code> 条件的 <code>else</code> 部分 (宏)。                                     |
| <code>.ep</code> |                           | 结束页 (宏)。                                                                                  |
| <code>\n</code>  | <code>(es)</code>         | 表示等式的前或后间距 (数字寄存器)。                                                                       |
| <code>\f</code>  | <code>Font</code>         | 设置直接插入字体, 将其更改为 <code>Font</code> 变量指定的值 ( <b>troff</b> 命令的内置功能)。                         |
| <code>\f</code>  | <code>(Font)</code>       | 设置直接插入字体, 将其更改为 <code>Fontf</code> 变量指定的值 ( <b>troff</b> 命令的内置功能)。                        |
| <code>.fc</code> |                           | 设置字段的字符 (宏)。                                                                              |
| <code>\n</code>  | <code>(ff)</code>         | 设置脚注字体 (数字寄存器)。                                                                           |
| <code>.fi</code> |                           | 填充输出行 (宏)。                                                                                |
| <code>\n</code>  | <code>(fi)</code>         | 表示脚注的缩进, 仅限于第一行 (数字寄存器)。                                                                  |
| <code>\n</code>  | <code>(fm)</code>         | 设置脚注的页边距 (数字寄存器)。                                                                         |
| <code>.fo</code> | <code>'X'Y'Z'</code>      | 将脚注设置为 <code>XYZ</code> 变量指定的值 (宏)。                                                       |
| <code>\n</code>  | <code>(fp)</code>         | 设置脚注的磅值 (数字寄存器)。                                                                          |
| <code>\n</code>  | <code>(fs)</code>         | 设置脚注的前间距 (数字寄存器)。                                                                         |
| <code>\n</code>  | <code>(fu)</code>         | 设置脚注距离右边界的缩进 (数字寄存器)。                                                                     |
| <code>\h</code>  | <code>'Distance'</code>   | 将本地水平移动值设置为指定的距离 ( <b>troff</b> 命令的内置功能)。                                                 |
| <code>.hc</code> |                           | 设置连字符 (宏)。                                                                                |
| <code>.he</code> | <code>'X'Y'Z'</code>      | 将页眉设置为 <code>XYZ</code> 变量指定的值 (宏)。                                                       |
| <code>.hl</code> |                           | 画水平线 (宏)。                                                                                 |
| <code>\n</code>  | <code>(hm)</code>         | 设置页眉边距 (数字寄存器)。                                                                           |
| <code>.hx</code> |                           | 取消下一页的页眉和页脚 (宏)。                                                                          |

|                           |                                                                                |
|---------------------------|--------------------------------------------------------------------------------|
| <code>.hy</code>          | 设置连字符连接方式 (宏)。                                                                 |
| <code>.i X</code>         | 用斜体打印 <i>X</i> 变量指定的值。如果省略了 <i>X</i> 变量, 则跟随斜体文本 (宏)。                          |
| <code>.ie</code>          | 指定 if/else 条件的 else 部分 (宏)。                                                    |
| <code>.if</code>          | 指定条件 (宏)。                                                                      |
| <code>\n(ii</code>        | 设置缩进段落缩进 (数字寄存器)。                                                              |
| <code>.in</code>          | 缩进 (瞬时); 扩大时使用 <code>.ba</code> 宏 (宏)。                                         |
| <code>.ip X Y</code>      | 打开缩进段落, 其悬挂标记由 <i>X</i> 变量指定。缩进是由 <i>Y</i> 变量指定的 <code>en</code> 值。缺省值为 5 (宏)。 |
| <code>.ix</code>          | 缩进, 没有中断 (宏)。                                                                  |
| <code>\V'Distance'</code> | 启动指定距离的水平绘线型函数 ( <code>troff</code> 命令的内置功能)。                                  |
| <code>.lc</code>          | 设置引导符的重复字符 (宏)。                                                                |
| <code>.lh</code>          | 内插本地信头 (宏)。                                                                    |
| <code>.ll</code>          | 设置行长 (宏)。                                                                      |
| <code>.lo</code>          | 读取 <code>.*x</code> 格式的本地宏文件。必须在初始化之前使用 (宏)。                                   |
| <code>.lp</code>          | 开始左对齐段落 (宏)。                                                                   |
| <code>\*(lq</code>        | 指定左引号 (字符串)。                                                                   |
| <code>.ls</code>          | 设置多行间距 (宏)。                                                                    |
| <code>.m1</code>          | 设置从页面顶部到页眉之间的间距 (宏)。                                                           |
| <code>.m2</code>          | 设置从页眉到文本之间的间距 (宏)。                                                             |
| <code>.m3</code>          | 设置从文本到页脚之间的间距 (宏)。                                                             |
| <code>.m4</code>          | 设置从页脚到页面底部之间的间距 (宏)。                                                           |
| <code>.mc</code>          | 插入边距字符 (宏)。                                                                    |
| <code>.mk</code>          | 标记垂直位置 (宏)。                                                                    |
| <code>\n(mo</code>        | 定义月份 (数字寄存器)。                                                                  |
| <br>                      |                                                                                |
| <code>\*(mo</code>        | 定义当前的月份 (字符串)。                                                                 |
| <code>\nX</code>          | 内插由 <i>X</i> 变量值指定的数字寄存器 (数字寄存器)。                                              |
| <code>\n(XX</code>        | 内插由 <i>XX</i> 变量指定的数字寄存器 (数字寄存器)。                                              |
| <code>.n1</code>          | 设置页边空白的行数 (宏)。                                                                 |
| <code>.n2</code>          | 设置页边空白的行数 (宏)。                                                                 |
| <code>.na</code>          | 关闭文本对齐 (宏)。                                                                    |
| <code>.neNumber</code>    | 设置垂直间距的指定行数 (宏)。                                                               |
| <code>.nf</code>          | 保留输出行不被填充 (宏)。                                                                 |
| <code>.nh</code>          | 关闭连字符 (宏)。                                                                     |
| <code>.np</code>          | 开始已编号的段落 (宏)。                                                                  |
| <code>.nr</code>          | 设置数字寄存器 (宏)。                                                                   |
| <code>.ns</code>          | 表示无间距方式 (宏)。                                                                   |
| <code>\*o</code>          | 表示上标的圆圈 (如 Norse A 使用的; 字符串)。                                                  |
| <code>.of'X'Y'Z'</code>   | 将奇数页的页脚设置为 <i>XYZ</i> 变量指定的值 (宏)。                                              |
| <code>.oh'X'Y'Z'</code>   | 将奇数页的页眉设置为 <i>XYZ</i> 变量指定的值 (宏)。                                              |
| <code>.pa</code>          | 开始页 (宏)。                                                                       |
| <code>.pd</code>          | 打印已延迟的文本 (宏)。                                                                  |
| <code>\n(pf</code>        | 表示段落字体 (数字寄存器)。                                                                |
| <code>\n(pi</code>        | 表示段落缩进 (数字寄存器)。                                                                |
| <code>.pl</code>          | 设置页面长度 (宏)。                                                                    |
| <code>.pn</code>          | 设置下一页页号 (宏)。                                                                   |
| <code>.po</code>          | 设置页面偏移量 (宏)。                                                                   |
| <code>\n(po</code>        | 模拟页面偏移量 (数字寄存器)。                                                               |
| <code>.pp</code>          | 开始段落, 首行缩进 (宏)。                                                                |
| <code>\n(pp</code>        | 设置段落的磅值 (数字寄存器)。                                                               |
| <code>\n(ps</code>        | 设置段落的前间距 (数字寄存器)。                                                              |
| <code>.q</code>           | 表示被加引号 (宏)。                                                                    |
| <code>\*(qa</code>        | 用于全部 (字符串)。                                                                    |
| <code>\*qe</code>         | 存在 (字符串)。                                                                      |
| <code>\n(qi</code>        | 设置引号里面文字的缩进; 同时缩短行 (数字寄存器)。                                                    |
| <code>\n(qp</code>        | 设置引号里面文字的磅值 (数字寄存器)。                                                           |



|                            |                                                                        |
|----------------------------|------------------------------------------------------------------------|
| <code>\n(qs</code>         | 设置引号里面文字的前或后间距 (数字寄存器)。                                                |
| <code>.r</code>            | 将下文设置为罗马体 (宏)。                                                         |
| <code>.rb</code>           | 设置为 <b>real</b> 粗体字 (宏)。                                               |
| <code>.re</code>           | 将制表符重新设置为缺省值 (宏)。                                                      |
| <code>.rm</code>           | 删除宏或者字符串 (宏)。                                                          |
| <code>.rn</code>           | 将宏或字符串重命名 (宏)。                                                         |
| <code>.ro</code>           | 将页号设置为罗马体 (宏)。                                                         |
| <code>\*(rq</code>         | 表示右引号 (字符串)。                                                           |
| <code>.rr</code>           | 删除寄存器 (宏)。                                                             |
| <code>.rs</code>           | 恢复寄存器 (宏)。                                                             |
|                            |                                                                        |
| <code>.rt</code>           | 返回到垂直位置 (宏)。                                                           |
| <code>\sSize</code>        | 将直接插入大小更改为指定大小 ( <b>troff</b> 命令的内置功能)。                                |
| <code>.sc</code>           | 读取特殊字符和区别符的文件。必须在初始化前使用 (宏)。                                           |
| <code>\n(sf</code>         | 设置节标题的字体 (数字寄存器)。                                                      |
| <code>.shLevelTitle</code> | 表示接下来的是节标题; 字体自动设为粗体。 <i>Level</i> 变量指定节的级别。 <i>Title</i> 变量指定节标题 (宏)。 |
| <code>\n(si</code>         | 设置每节缩进深度的相对基数 (数字寄存器)。                                                 |
| <code>.sk</code>           | 保留下一页为空白页。前面的页只存储一页 (宏)。                                               |
| <code>.smX</code>          | 设置为比 <i>X</i> 变量指定的值小一号的磅值 (宏)。                                        |
| <code>.so</code>           | 表示源输入文件 (宏)。                                                           |
| <code>\n(so</code>         | 设置附加的节标题的偏移量 (数字寄存器)。                                                  |
| <code>.sp</code>           | 表示垂直间距 (宏)。                                                            |
| <code>\n(sp</code>         | 表示节标题的磅值 (数字寄存器)。                                                      |
| <code>\n(ss</code>         | 表示节的前间距 (数字寄存器)。                                                       |
| <code>.sx</code>           | 改变节的深度 (宏)。                                                            |
| <code>.sz +Number</code>   | 增加磅值至指定值 (宏)。                                                          |
| <code>.ta</code>           | 设置制表符停止位 (宏)。                                                          |
| <code>.tc</code>           | 设置制表符重复字符 (宏)。                                                         |
| <code>\*(td</code>         | 设置当前日期 (字符串)。                                                          |
| <code>n(tf</code>          | 表示标题字体 (数字寄存器)。                                                        |
| <code>.th</code>           | 以论题格式生成论文。必须在初始化前使用 (宏)。                                               |
| <code>.ti</code>           | 表示临时缩进, 只应用于下一行 (宏)。                                                   |
| <code>.tl</code>           | 表示 3 个部分的标题 (宏)。                                                       |
| <code>\n(tm</code>         | 设置主标题的页边距 (数字寄存器)。                                                     |
| <code>.tp</code>           | 开始标题页 (宏)。                                                             |
| <code>\n(tp</code>         | 设置标题的磅值 (数字寄存器)。                                                       |
| <code>.tr</code>           | 转换 (宏)。                                                                |
| <code>.u X</code>          | 将 <i>X</i> 变量指定的值加下划线, 即使是在 <b>troff</b> 命令中。只采用非填充方式 (宏)。             |
| <code>.uh</code>           | 设置节标题继续; 字体自动采用粗体。与 <b>.sh</b> 宏相似, 但是未编号 (宏)。                         |
| <code>.ul</code>           | 将下一行加下划线 (宏)。                                                          |
| <code>\v'Distance'</code>  | 本地垂直移动指定的距离 ( <b>troff</b> 命令的内置功能)。                                   |
| <code>\*v</code>           | 为捷克语的 e 倒置 v (字符串)。                                                    |
| <code>\w'String'</code>    | 返回指定字符串的宽度 ( <b>troff</b> 命令的内置功能)。                                    |
| <code>.xl</code>           | 设置本地行长度 (宏)。                                                           |
| <code>.xpIndex</code>      | 打印指定的索引 (宏)。                                                           |
| <code>\n(xs</code>         | 设置索引项的前间距 (数字寄存器)。                                                     |
| <code>\n(xu</code>         | 设置索引的右边界缩进 (数字寄存器)。                                                    |
| <code>\n(yr</code>         | 表示年份, 只使用后两位数字 (数字寄存器)。                                                |
| <code>\n(zs</code>         | 设置浮动保持的前或后间距 (数字寄存器)。                                                  |
| <code>{</code>             | 开始条件组 ( <b>troff</b> 命令的内置功能)。                                         |
| <code> </code>             | 1/6 单位长度, 小间距 ( <b>troff</b> 命令的内置功能)。                                 |
| <code>}</code>             | 结束条件组 ( <b>troff</b> 命令的内置功能)。                                         |
| <code>\*~</code>           | 表示代字号 (字符串)。                                                           |

有关更多信息，请参阅 *-ME Reference Manual*，作者 E. P. Allman。

## 用于 mm, mmt, nroff 和 troff 命令的 mm 宏软件包

**mm** 宏软件包提供能够对多种文档格式的文本，如备忘录、信件和报告等进行格式化的宏。用户输入和编辑文档的方式本质上与该文档将稍后在终端上还是在照相排版机上被格式化无关。

可能需要 **col** 命令对 **nroff** 输出进行后处理。特定的要求请参阅 **col** 命令。

**mm** 宏和附加信息被概述为以下标题：

- 正式备忘录的开始宏
- 商业信函宏
- 结束宏（后缀信息）
- 段落
- 节标题
- 列表
- 显示、表格、等式和脚注
- 页眉和页脚
- 其他宏
- **mm** 寄存器
- **mm** 字符串
- 字符串名称
- 保留名称。

### 正式备忘录的开始宏

**.ND** *Date*

**.TL** [*ChgNumber*] [*FileNumber*]

**.AF** [*CompanyName*]

**.AU** *Name* [*Initials*] [*Loc*] [*Dept*] [*Ext*] [*Room*] [*Option...*]

**.AT** *AuthorTitle* [...]

**.TM** [*Number*]

**.AS** [ **0** | **1** | **2** ] [*Indent*]

设置新的日期。

设置标题信息。下一行文本作为文档的标题。

指定作者的公司名。

设置作者信息。

指定标题，跟在签名人的姓名后面（达 9 个选项）。

设置技术备忘录的编号。

仅启动技术备忘录和论文发行版的专用摘要：

**0**       封面和首页上的摘要

**1**       只在封面上的摘要

**2**       只在文件封面的备忘录上的摘要。

**.AE**

**.NS**

**.NE**

**.OK** [*Keyword* ...]

结束摘要。

启动符号，允许为紧跟 **.AS 2/.AE** 宏对的文件封面页打开备忘录（请参阅”结束宏“）。

结束符号，允许为紧跟 **.AS 2/.AE** 宏对的文件封面页打开备忘录（请参阅”结束宏“）。

指定其他关键字（最多 9 个选项）。

**.MT** [*type*] [*title*]

设置文档类型:

- "" 无类型。
- 0** 无类型 (内部信件)。
- 1** 文件的备忘录。
- 2** 程序员注释。
- 3** 工程师注释
- 4** 发布的论文。
- 5** 外部信件。

"String"  
打印指定字符串。

*Title*

使用用户提供的文本作为页码前缀

## 商业信函宏

**.WA**

开始作者地址。

**.WE**

结束作者地址。

**.LO CN** [*Notation*]

指定机密符号。

**.LO RN** [*Notation*]

指定参考符号。

**.IA**

开始内部 (收件人) 地址。

**.IE**

结束内部 (收件人) 地址。

**.LO AT** [*Notation*]

指定注意线。

**.LO SA** [*Notation*]

指定尊称。

**.LO SJ** [*Notation*]

指定主题行。

**.LT** [ { none **BL SB FB SP** } ]

指定商业信函类型:

**无** 块的

**BL** 块的

**SB** 半块的

**FB** 全块的

**SP** 简化的。

## 结束宏 (后缀信息)

**.FC** [*Closing*]

打印正式的结束语。

**.SG** [*Initials*] [1]

打印签名行。

**.NS** [{"0 1 2 3 4 5 6 7 8 9 10 11 12 13 String}]

启动符号:

” “

抄送

**0** 抄送

**1** 抄送 (含附件)

**2** 抄送 (无附件)

**3** 附件

**4** 多个附件

**5** 信中附件

**6** 多个信中附件

**7** 使用独立的封面

**8** 致信给

**9** 备忘录

**10** 抄送 (含多个附件)

**11** 抄送 (无附件)

**12** 专用摘要

**13** 完整的备忘录

字符串

复制 (字符串) 到

**.NE**

**.AV** Name [1]

**.CS** [Pgs] [Other] [Tot] [Figs] [TbIs] [Ref]

**.TX**

**.TY**

**.TC** [Slev] [Spacing] [Tlev] [Tab] [H1] [H2] [H3] [H4] [H5] 打印目录。

结束符号。

打印许可签名。

打印封面页。

为目录标题调用用户出口。

为目录页眉调用用户出口。

## 段落

**.P** [ {0 1 2} ]

启动段落:

**0** 左对齐 (缺省值)

**1** 缩进

**2** 除在 **.H**、**.LE** 和 **.DE** 后面以外采用缩进处理。

## 节标题

**.H** {1 2 3 4 5 6 7} [HeadingText] [FootnoteMark]

**.HU** HeadingText

指定编号的页眉。

指定未编号的页眉。

**.HM** {1 0001 A a l i}...

指定页眉标记的样式:

- 1** 阿拉伯数字
- 0001** 以 0 开头的阿拉伯数字
- A** 大写字母
- a** 小写字母
- l** 大写罗马体
- i** 小写罗马体。

**.HX** [Dlev] [Rlev] [HeadingText]

在页眉前调用用户自定义的退出宏。

**.HY** [Dlev] [Rlev] [HeadingText]

在页眉中调用用户自定义的退出宏。

**.HZ** [Dlev] [Rlev] [HeadingText]

在页眉后调用用户自定义的退出宏。

## 列表

如果最后的选项 [1] 出现在列表启动宏中, 则在各项之间没有空格。

**.AL** [ {1 A a l i} ] [TextIndent] [1]

自动启动增量列表 (1)。

**.BL** [TextIndent ] [1]

启动项目符号列表。

**.DL** [TextIndent] [1]

启动破折号列表。

**.ML** Mark [TextIndent] [1]

启动列表, 该列表中的每个列表项都被加上了指定的标记。如果 *TextIndent* 的值是 **NULL** 或者被省略, 则该值就被设为 [*Mark* - width + 1]。如果指定了第三个自变量, 则列表中的各项之间就没有空白行分隔。

**.RL** [TextIndent] [1]

启动引用列表。

**.VL** TextIndent [MarkIndet] [1]

启动变量标记列表。

**.LI** [Mark] [1]

启动列表项; 1 表示 *Mark* 变量的值被设为当前标记的前缀。结束列表项; 1 表示在列表后面输出一个空白行。缺省值为没有空白行。

**.LE** [1]

**.LB** TextIndent MarkIndet PadType [Mark] [{0 1}] [{0 1}]

开始列表:

*Type* 变量的值是:

**1=.** **2=)** **3=(** **4=[** **5=<>** **6={}**.

第六选项:

**0** 在每个列表项前没有空白行。

第七选项:

**0** 在列表前没有空白行。

清除列表状态, 直至 *Level* 变量值。

**.LC** [Level]

## 显示、表格、等式和脚注

**.DS** [{0 1 2 3}] [{0 1}] [Number]

**.DS** [{L I C CB}] [{N F}] [Number]

启动静态显示:

**0** 或 **L**

无缩进

**1 或 I**

左缩进

**2 或 C**

将每一行居中对齐

**3 或 CB**

作为一个块居中对齐

**0 或 N**

不填充

**1 或 F**

填充。

*Number*

右缩进，空格的个数由 *Number* 参数来指定。

**.DF** [{0 1 2 3}] [{0 1}] [*Number*]

**.DF** [{L I C CB}] [{N F}] [*Number*]

启动浮动显示:

**0 或 L**

无缩进

**1 或 I**

左缩进

**2 或 C**

将每一行居中对齐

**3 或 CB**

作为一个块居中对齐

**0 或 N**

不填充

**1 或 F**

填充。

*Number*

右缩进，空格的个数由 *Number* 参数来指定。

**.DE** 结束显示。

**.FG** [*Title*] [*Override*] [0 1 2]

*Override* 变量的值替换或提高缺省的编号。指定图形题注:

**0** *Override* 值被用作前缀。

**1** *Override* 值成为后缀。

**2** 替换 *Override* 值成为替换值。

**.TS** [H]

启动表格:

**H** 多页表格。

**.TH [N]**

在指定选项 **H** 到 **.TS** 的时候必须使用:

**N** 除了在新页的顶部外, 禁止表格标题。

**.TE** 结束表格。

**.TB [Title] [Override] [0 1 2]**

*Override* 变量的值替换或提高缺省的编号。指定表格的题注:

**0** *Override* 值被用作前缀。

**1** *Override* 值成为后缀。

**2** 替换 *Override* 值成为替换值。

**.EX [Title] [Override] [0 1 2]**

*Override* 变量的值替换或提高缺省的编号。指定展示题注:

**0** *Override* 值被用作前缀。

**1** *Override* 值成为后缀。

**2** 替换 *Override* 值成为替换值。

**.EQ [Label]**

使用指定的标号启动等式显示。

**.EN** 结束等式显示。

**.EC [Title] [Override] [0 1 2]**

*Override* 变量的值替换或提高缺省的编号。指定等式的题注:

**0** *Override* 值被用作前缀。

**1** *Override* 值成为后缀。

**2** 替换 *Override* 值成为替换值。

**.FS [Label]**

使用指定的标号作为指示符开始脚注。缺省是编号了的脚注。

**.FE** 结束脚注。

**.FD [{0 1 2 3 4 ... 11}] [1]**

设置脚注格式:

第一选项:

设置脚注文本的格式化样式。 **mmt** 命令的缺省值为 0。 **mm** 命令的缺省值为 10。有关值请参阅以下表格。

第二选项:

复位第一级标题的脚注计数器。

| .FD Arg. | 格式                                                            |
|----------|---------------------------------------------------------------|
| 0        | 连字号 <b>.nh</b><br>已调整 <b>.ad</b><br>缩进的文本<br>是<br>对齐的标注<br>左侧 |
| 1        | 连字号 <b>.hy</b><br>已调整 <b>.ad</b><br>缩进的文本<br>是<br>对齐的标注<br>左侧 |
| 2        | 连字号 <b>.nh</b><br>已调整 <b>.na</b><br>缩进的文本<br>是<br>对齐的标注<br>左侧 |
| 3        | 连字号 <b>.hy</b><br>已调整 <b>.na</b><br>缩进的文本<br>是<br>对齐的标注<br>左侧 |
| 4        | 连字号 <b>.nh</b><br>已调整 <b>.ad</b><br>缩进的文本<br>否<br>对齐的标注<br>左侧 |
| 5        | 连字号 <b>.hy</b><br>已调整 <b>.ad</b><br>缩进的文本<br>否<br>对齐的标注<br>左侧 |



| .FD Arg. | 格式                                                            |
|----------|---------------------------------------------------------------|
| 6        | 连字号 <b>.nh</b><br>已调整 <b>.na</b><br>缩进的文本<br>否<br>对齐的标注<br>左侧 |
| 7        | 连字号 <b>.hy</b><br>已调整 <b>.na</b><br>缩进的文本<br>否<br>对齐的标注<br>左侧 |
| 8        | 连字号 <b>.nh</b><br>已调整 <b>.ad</b><br>缩进的文本<br>是<br>对齐的标注<br>右侧 |
| 9        | 连字号 <b>.hy</b><br>已调整 <b>.ad</b><br>缩进的文本<br>是<br>对齐的标注<br>右侧 |
| 10       | 连字号 <b>.nh</b><br>已调整 <b>.na</b><br>缩进的文本<br>是<br>对齐的标注<br>右侧 |
| 11       | 连字号 <b>.hy</b><br>已调整 <b>.na</b><br>缩进的文本<br>是<br>对齐的标注<br>右侧 |

## 页眉和页脚

|                     |              |
|---------------------|--------------|
| <b>.PH</b> ”左’居中’右“ | 指定页眉。        |
| <b>.OH</b> ”左’居中’右“ | 指定奇数页的页眉。    |
| <b>.EH</b> ”左’居中’右“ | 指定偶数页的页眉。    |
| <b>.PF</b> ”左’居中’右“ | 指定页脚。        |
| <b>.OF</b> ”左’居中’右“ | 指定奇数页的页脚。    |
| <b>.EF</b> ”左’居中’右“ | 指定偶数页的页脚。    |
| <b>.BS</b>          | 启动下对齐块。      |
| <b>.BE</b>          | 结束下对齐块。      |
| <b>.PX</b>          | 为页眉标题调用用户出口。 |
| <b>.TP</b>          | 调用页面顶部的宏。    |

## 其他宏

|                                                                     |                                                                                                                                                                                                                                                                                                                          |
|---------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>.B</b> [ <i>Option</i> ] [ <i>Prev-Font-option</i> ]             | 以粗体打印（最多六个选项）。                                                                                                                                                                                                                                                                                                           |
| <b>.I</b> [ <i>Option</i> ] [ <i>Prev-Font-option</i> ]             | 以斜体打印（最多六个选项）；使用 <b>nroff</b> 命令来加下划线。                                                                                                                                                                                                                                                                                   |
| <b>.R</b>                                                           | 返回到罗马字体。                                                                                                                                                                                                                                                                                                                 |
| <b>.PM</b> [ <i>Option</i> ]                                        | 设置专利标记。如果不向 <b>.PM</b> 宏赋予选项的话，请您关闭专利标记。 <b>/usr/lib/macros/string.mm</b> 文件包含一些专利标记。该文件可以编辑以满足用户需要。                                                                                                                                                                                                                     |
| <b>.RD</b> [ <i>Prompt</i> ] [ <i>Diversion</i> ] [ <i>String</i> ] | 停止代码宏。 <i>Prompt</i> 变量是用户自定义的、不含空格的字符串。 <i>Diversion</i> 变量允许保存输入文本。 <i>String</i> 变量包含在提示符后输入的第一行文本。                                                                                                                                                                                                                   |
| <b>.RP</b> [{0 1}] [{0 1 2 3}]                                      | 生成引用页：<br><br>第一选项：<br><b>0</b> 复位引用计数器（缺省值）。<br><b>1</b> 不复位引用计数器。<br><br>第二选项：<br><b>0</b> 之后导致 <b>.SK</b> 宏（缺省值）。<br><b>1</b> 之后不导致 <b>.SK</b> 宏。<br><b>2</b> 之前不导致 <b>.SK</b> 宏。<br><b>3</b> 之前或之后都不导致 <b>.SK</b> 宏。                                                                                                 |
| <b>.RS/RP</b>                                                       | 自动对引用进行编号。                                                                                                                                                                                                                                                                                                               |
| <b>.WC</b> [{N WF -WF FF -FF WD -WD FB -FB}]                        | 控制脚注的宽度，在使用两栏的时候显示其宽度：<br><b>N</b> 正常方式（ <b>-WF</b> 、 <b>-FF</b> 、 <b>-WD</b> ）。<br><b>WF</b> 始终保持宽脚注。<br><b>-WF</b> 脚注与页面样式相同。<br><b>FF</b> 第一个脚注决定本页上其他脚注的宽度。<br><b>-FF</b> 脚注遵照 <b>WF</b> 或 <b>-WF</b> 选项的设置。<br><b>WD</b> 始终保持宽显示。<br><b>-WD</b> 显示页面样式相同。<br><b>FB</b> 浮动显示产生页面间隔符（缺省值）。<br><b>-FB</b> 浮动显示不产生页面中断。 |
| <b>.SP</b> [ <i>Lines</i> ]                                         | 向下跳行。                                                                                                                                                                                                                                                                                                                    |
| <b>.SK</b> [ <i>Number</i> ]                                        | 跳过指定的页数。（缺省值为 1。）                                                                                                                                                                                                                                                                                                        |

|                                                                 |                                                                                                                      |
|-----------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------|
| <b>.OP</b>                                                      | 在奇数页中断。                                                                                                              |
| <b>.2C</b>                                                      | 以两栏打印输出。                                                                                                             |
| <b>.1C</b>                                                      | 以一栏打印输出（按照恢复的正常行宽）。                                                                                                  |
| <b>.SA</b> [ <i>Option</i> ]                                    | 设置右边界对齐                                                                                                              |
|                                                                 | 选项:                                                                                                                  |
| <b>0</b>                                                        | 将缺省值设置为 <b>Off</b> ( <b>nroff</b> 命令的缺省值)。                                                                           |
| <b>1</b>                                                        | 将缺省值设置为 <b>on</b> ( <b>troff</b> 命令的缺省值)。                                                                            |
| <b>.SM</b> <i>String1</i> [ <i>String2</i> ] [ <i>String3</i> ] | 如果没有指定任何选项，则宏还原成当前的缺省值。<br>如果省略了 <i>String3</i> 变量值，则将 <i>String1</i> 变量的值减小 1 磅；否则，<br>将 <i>String2</i> 变量的值减小 1 磅。 |
| <b>.HC</b> <i>Character</i>                                     | 将连字符处理的字符设置为 <i>Character</i> 变量的值。                                                                                  |
| <b>.S</b> [ <i>PointSize</i> ] [ <i>VerticalSpacing</i> ]       | 设置磅值和垂直间距（只在 <b>troff</b> 命令中）。                                                                                      |
|                                                                 | 缺省值:                                                                                                                 |
|                                                                 | 磅值 = <b>10p</b>                                                                                                      |
|                                                                 | 垂直间距 = <b>12p</b>                                                                                                    |
|                                                                 | 选项 1 和 2:                                                                                                            |
|                                                                 | <i>Number</i>                                                                                                        |
|                                                                 | 新值。                                                                                                                  |
|                                                                 | <i>+/-Number</i>                                                                                                     |
|                                                                 | 增加至当前值。                                                                                                              |
| <b>D</b>                                                        | 缺省值。                                                                                                                 |
| <b>C</b>                                                        | 当前值。                                                                                                                 |
| <b>P</b>                                                        | 前一值。                                                                                                                 |
| <b>.VM</b> [ <i>Top</i> ] [ <i>Bottom</i> ]                     | 设置变量的垂直页边距。                                                                                                          |
| <b>.nP</b>                                                      | 启动段落的双行缩进。                                                                                                           |

以下的宏是用来修改字体的，都有 1 到 6 个选项:

|            |                                 |
|------------|---------------------------------|
| <b>.IB</b> | 斜体 ( <b>nroff</b> 的下划线) 和粗体互换。  |
| <b>.BI</b> | 粗体和斜体互换。                        |
| <b>.RI</b> | 罗马体和斜体互换。                       |
| <b>.IR</b> | 斜体 ( <b>nroff</b> 的下划线) 和罗马体互换。 |
| <b>.RB</b> | 罗马体和粗体互换。                       |
| <b>.BR</b> | 粗体和罗马体互换。                       |

## mm 寄存器

如果一个注册名称后面紧跟着一个 \* (星号)，则有两种方式来对注册器进行设置：在命令行中进行（请参阅 **mm** 命令中的示例）或在格式化程序读取 **mm** 宏定义之前。以下列表中括号内的数字为缺省值。

|            |                     |
|------------|---------------------|
| <b>A *</b> | 处理预打印的表单。           |
| <b>Au</b>  | 禁止作者信息出现在首页上 (1)。   |
| <b>C *</b> | 副本类型 (如原稿和草稿) (0)。  |
| <b>Cl</b>  | 内容级别 (2)。           |
| <b>Cp</b>  | 图形、表格、等式和显示的布置 (1)。 |

|                |                                                          |
|----------------|----------------------------------------------------------|
| <b>D *</b>     | 调试标志 (0)。如果设置为 1, 则 <b>mm</b> 命令持续执行, 即使遇到通常是致命性的错误也不停止。 |
| <b>De</b>      | 浮动显示后退纸 (0)。                                             |
| <b>Df</b>      | 如果设置为 1, 则要为浮动显示而格式化寄存器 (5)。                             |
| <b>Ds</b>      | 静态显示的前或后间距 (1)。                                          |
| <b>E *</b>     | 控制主题 / 日期 / 发件人字段的字体 (0): 0 = 粗体; 1 = 罗马体。               |
|                | <b>0</b> 粗体 (0)                                          |
|                | <b>1</b> 罗马体。                                            |
| <b>Ec</b>      | 等式计数器。                                                   |
| <b>Ej</b>      | 页眉的退纸标记 (0)。                                             |
| <b>Eq</b>      | 等式标号布局 (0)。                                              |
| <b>Ex</b>      | 显示计数器。                                                   |
| <b>Fg</b>      | 图形计数器。                                                   |
| <b>Fs</b>      | 垂直脚注间距 (1)。                                              |
| <b>H1...H7</b> | 页眉计数器。                                                   |
| <b>Hb</b>      | 页眉中断级别 (在 <b>.H</b> 和 <b>.HU</b> 之后) (2)。                |
| <b>Hc</b>      | <b>.H</b> 和 <b>.HU</b> 的页眉居中对齐的级别 (0)。                   |
| <b>Hi</b>      | 页眉的临时缩进 (在 <b>.H</b> 和 <b>.HU</b> 之后) (1)。               |
| <b>Hs</b>      | 页眉空格的级别 (在 <b>.H</b> 和 <b>.HU</b> 之后) (2)。               |
| <b>Ht</b>      | 页眉类型:                                                    |
|                | <b>0</b> 连续数字 (0)                                        |
|                | <b>1</b> 单个数字 (0)。                                       |
| <b>Hu</b>      | 没有编号的页眉的级别 (2)。                                          |
| <b>Hy</b>      | 连字符处理控制;                                                 |
|                | <b>0</b> 没有连字符 (0)                                       |
|                | <b>1</b> 启用连字符。                                          |
| <b>L *</b>     | 页面长度 (66v)。                                              |
| <b>Le</b>      | 目录后的等式列表 (0):                                            |
|                | <b>0</b> 不打印                                             |
|                | <b>1</b> 打印。                                             |
| <b>Lf</b>      | 目录后的图列表 (0):                                             |
|                | <b>0</b> 不打印                                             |
|                | <b>1</b> 打印。                                             |
| <b>Li</b>      | 列表缩进 (5, <b>troff</b> 命令); (6, <b>nroff</b> 命令)。         |
| <b>Ls</b>      | 列表级别下降到项间有空白的级别 (6)。                                     |
| <b>Lt</b>      | 目录后的表格列表 (0):                                            |
|                | <b>0</b> 不打印                                             |
|                | <b>1</b> 打印                                              |
| <b>Lx</b>      | 目录后的显示列表 (1):                                            |
|                | <b>0</b> 不打印                                             |
|                | <b>1</b> 打印。                                             |
| <b>N *</b>     | 编号样式 (0)。                                                |
| <b>Np</b>      | 已编号的段落:                                                  |
|                | <b>0</b> 未编号                                             |
|                | <b>1</b> 已编号 (0)。                                        |
| <b>O *</b>     | 页面偏移。                                                    |

|            |                                                       |
|------------|-------------------------------------------------------|
| <b>Oc</b>  | 目录的页面编号样式:                                            |
|            | <b>0</b> 小写罗马体                                        |
|            | <b>1</b> 阿拉伯字体 (0)。                                   |
| <b>Of</b>  | 图表标题样式 (0)。                                           |
| <b>P</b>   | 页面编号; 由 <b>mm</b> 命令管理 (0)。寄存器接受 0 或负整数的值。            |
| <b>Pi</b>  | 段落缩进 (5)。                                             |
| <b>Ps</b>  | 段落间距 (1)。                                             |
| <b>Pt</b>  | 段落类型 (0)。                                             |
| <b>Pv</b>  | 专用页眉                                                  |
|            | <b>0</b> 不打印专用                                        |
|            | <b>1</b> 只在第一页                                        |
| <b>2</b>   | 在所有的页上 (0)。                                           |
| <b>Rf</b>  | 引用计数器; 由 <b>.RS</b> 宏使用。                              |
| <b>S *</b> | <b>troff</b> 命令的缺省磅值 (10)。                            |
| <b>Si</b>  | 显示缩进 (5)。                                             |
| <b>T *</b> | <b>nroff</b> 命令输出设备的类型 (0)。                           |
| <b>Tb</b>  | 表格计数器。                                                |
| <b>U *</b> | <b>.H</b> 和 <b>.HU</b> 的下划线样式 ( <b>nroff</b> 命令) (0)。 |
| <b>W *</b> | 页面宽度 (行和标题的长度)。                                       |

## mm 字符串

使用以下的转义序列打印特殊字符串：

|               |                             |
|---------------|-----------------------------|
| <b>\*x</b>    | 用于带单字符名称的字符串 ( <b>x</b> )   |
| <b>\*(xx)</b> | 用于带双字符名称的字符串 ( <b>xx</b> )。 |

## 字符串名称

|           |                                                                                                                              |
|-----------|------------------------------------------------------------------------------------------------------------------------------|
| <b>BU</b> | 项目符号。                                                                                                                        |
| <b>Ci</b> | 目录中的标题缩进级别                                                                                                                   |
| <b>DT</b> | 当前日期。由语言环境设置指定的用于的 <b>LC_TIME</b> 类别的特定于语言环境的日期格式用作缺省设置。该设置同 <b>strftime</b> 子例程的 <b>%x</b> 格式说明符相对应。使用 <b>.ND</b> 宏来更改当前日期。 |
| <b>EM</b> | 破折号。                                                                                                                         |
| <b>F</b>  | 脚注编号。                                                                                                                        |
| <b>HF</b> | 标题级字体字符串:                                                                                                                    |
|           | <b>1</b> 罗马体                                                                                                                 |
|           | <b>2</b> 斜体                                                                                                                  |
|           | <b>3</b> 粗体 (2 2 2 2 2 2 2)。                                                                                                 |
| <b>HP</b> | 不同级别标题的磅值。                                                                                                                   |
| <b>Le</b> | 等式列表的标题。                                                                                                                     |
| <b>Lf</b> | 图形列表的标题。                                                                                                                     |
| <b>Lt</b> | 表格列表的标题。                                                                                                                     |
| <b>Lx</b> | 显示列表的标题。                                                                                                                     |
| <b>RE</b> | <b>mm</b> 宏的 SCCS SID。                                                                                                       |
| <b>Rf</b> | 引用编号器。                                                                                                                       |
| <b>Rp</b> | 引用页面的标题。                                                                                                                     |
| <b>Tm</b> | 商标。                                                                                                                          |
| <b>`</b>  | 抑音符                                                                                                                          |

, 重音符。  
 ^ 长音符。  
 ~ 代字符号。  
 : 小写变元音符号。  
 ; 大写变元音符号。  
 , 变音符。

## 保留名称

如果要自定义字符串、宏和寄存器，只能使用单个小写字母或者一个小写字母后跟一个非小写字母字符来命名。名称 **c2** 和 **nP** 是本规则的例外情况；他们是保留名称。

## 用于 **nroff** 和 **troff** 命令的 **mptx** 宏软件包

**mptx** 宏软件包为 **.xx** 宏提供了定义，该宏是用来为由 **ptx** 命令生成的改序索引进行格式化的。**mptx** 宏软件包不提供任何其他的格式化能力，如页眉和页脚等。**mptx** 宏软件包必须同 **mm** 宏软件包共同使用才能提供这些能力。在此情况下，在 **-mm** 调用后调用 **-mptx** 选项，方法如下：

```
nroff -mm -mptx File... | Printer
```

## 用于 **nroff** 和 **troff** 命令的 **ms** 宏软件包

**nroff** 和 **troff** 命令宏定义的 **ms** 宏语句包为各种样式的文章、论文和书籍提供了格式化设施。在某些情况下，后处理输出可能会需要用到 **col** 命令。

宏请求在 **ms** 请求段中定义。同该软件包在一起，许多 **nroff** 和 **troff** 命令请求可能会产生无法预料的结果。但是，以下列表中的头四个请求可以在初始化之后使用，最后两个请求可以在初始化之前使用。

|                              |                                                         |
|------------------------------|---------------------------------------------------------|
| <b>.bp</b>                   | 开始新的页。                                                  |
| <b>.br</b>                   | 中断输出行。                                                  |
| <b>.ce</b> [ <i>Number</i> ] | 居中对齐下一个指定行号。                                            |
| <b>.ls</b> [ <i>Number</i> ] | 设置行间距。将 <i>Number</i> 变量的值设为 1 以表示单空格文本；设置为 2 来表示双空格文本。 |
| <b>.na</b>                   | 关闭右页边距对齐。                                               |
| <b>.sp</b> [ <i>Number</i> ] | 插入指定数量的空白行。                                             |

允许使用 **\f** 和 **\s** 宏来更改字体和磅值。例如，**\fIword\fR** 用斜体字显示 **word**。用于等式、表格和引用的 **tbl**、**eqn** 和 **refer** 命令预处理程序的输出可以被接受作为输入。

通过内置数字寄存器可以在 **ms** 宏内对格式化距离进行控制。例如，以下数字寄存器将行长度设置为 6.5 英寸：

```
.nr LL 6.5i
```

有关 **ms** 寄存器的更多信息，请参阅 **ms** 寄存器。

## **ms** 请求

以下是外部 **ms** 宏请求：

|                         |                              |
|-------------------------|------------------------------|
| <b>.AB</b> [ <b>X</b> ] | 开始摘要。如果 <b>X</b> 为 no，不标记摘要。 |
|                         | 初始值： -                       |
|                         | 中断： <b>yes</b>               |

|                |                                                                                       |
|----------------|---------------------------------------------------------------------------------------|
| <b>.AE</b>     | 结束摘要。<br>中断: <b>yes</b> 初始值: -                                                        |
| <b>.AIName</b> | 中断: <b>yes</b><br>作者的机构。<br>初始值: -                                                    |
| <b>.AM</b>     | 中断: <b>yes</b><br>设置重音标记定义。<br>初始值: -                                                 |
| <b>.AUName</b> | 中断: <b>no</b><br>设置作者名。<br>初始值: -                                                     |
| <b>.B [X]</b>  | 中断: <b>yes</b><br>将 <b>X</b> 设置为黑体。如果没有 <b>X</b> , 切换到黑体。<br>初始值: -                   |
| <b>.B1</b>     | 中断: <b>no</b><br>打开框中所附的文本。<br>初始值: -                                                 |
| <b>.B2</b>     | 中断: <b>yes</b><br>结束框中文本并打印。<br>初始值: -                                                |
| <b>.BT</b>     | 中断: <b>yes</b><br>在页脚打印底部标题。<br>初始值: <b>date</b>                                      |
| <b>.BX X</b>   | 中断: <b>no</b><br>打印框中的字 <b>X</b> 。<br>初始值: -                                          |
| <b>.CM</b>     | 中断: <b>no</b><br>页间剪切标记。<br>初始值: <b>if t</b>                                          |
| <b>.CT</b>     | 中断: <b>no</b><br>指示章标题; 页编号移动到 CF (TM)。<br>初始值: -<br>中断: <b>yes</b><br>复位: <b>yes</b> |

|                    |                                                                                                       |
|--------------------|-------------------------------------------------------------------------------------------------------|
| <b>.DA [X]</b>     | <p>页面底部的强制日期 <b>X</b>。如果没有 <b>X</b>，日期为当天。</p> <p>初始值: <b>if n</b></p> <p>中断: <b>no</b></p>           |
| <b>.DE</b>         | <p>结束任意类型的显示（未填充文本）。</p> <p>初始值: -</p> <p>中断: <b>yes</b></p>                                          |
| <b>.DS X Y</b>     | <p>使用保留语句开始显示。 <b>X</b>=I, L, C, B; <b>Y</b>=indent.</p> <p>初始值: <b>l</b></p> <p>中断: <b>yes</b></p>   |
| <b>.ID Y</b>       | <p>显示无保留缩进。 <b>Y</b>=indent.</p> <p>初始值: <b>8n、.5i</b></p> <p>中断: <b>yes</b></p>                      |
| <b>.LD</b>         | <p>设置无保留语句的左显示。</p> <p>初始值: -</p> <p>中断: <b>yes</b></p>                                               |
| <b>.CD</b>         | <p>无保留居中显示。</p> <p>初始值: -</p> <p>中断: <b>yes</b></p>                                                   |
| <b>.BD</b>         | <p>块显示; 整块居中。</p> <p>初始值: -</p> <p>中断: <b>yes</b></p>                                                 |
| <b>.EF X</b>       | <p>设置偶数页页脚 <b>X</b> (3 作为 <b>troff</b> 命令、<b>.tl</b> 请求的一部分)。</p> <p>初始值: -</p> <p>中断: <b>yes</b></p> |
| <b>.EH X</b>       | <p>设置偶数页页眉 <b>X</b> (3 作为 <b>troff</b> 命令、<b>.tl</b> 请求的一部分)。</p> <p>初始值: -</p> <p>中断: <b>no</b></p>  |
| <b>.EN</b>         | <p>结束 <b>eqn</b> 命令生成的已显示的等式。</p> <p>初始值: -</p> <p>中断: <b>yes</b></p>                                 |
| <b>.EQ [X] [Y]</b> | <p>中断等式。 <b>X</b>=L, I, C; <b>Y</b> 是等式编号。</p> <p>初始值: -</p> <p>中断: <b>yes</b></p>                    |



|                |                                                                                       |
|----------------|---------------------------------------------------------------------------------------|
| <b>.FE</b>     | 结束将放置在页面底部的脚注。<br>初始值: -                                                              |
| <b>.FP</b>     | 中断: <b>no</b><br>数字脚注段落; 可以重新定义。<br>初始值: -                                            |
| <b>FS [X]</b>  | 中断: <b>no</b><br>开始脚注; <b>X</b> 是可选脚注标签。<br>初始值: -                                    |
| <b>.HD</b>     | 中断: <b>no</b><br>在页标题的上边距下设置可选页标题。<br>初始值: <b>undef</b>                               |
| <b>.I [X]</b>  | 中断: <b>no</b><br>以斜体显示 <b>X</b> 。如果没有 <b>X</b> , 以相当于斜体字体的 <b>.ft 2</b> 显示。<br>初始值: - |
| <b>.IP X Y</b> | 中断: <b>no</b><br>缩进段落, 由悬挂标记 <b>X. Y</b> 指定缩进空格数。<br>初始值: -                           |
| <b>.IX X Y</b> | 中断: <b>yes</b><br>复位: <b>yes</b><br>索引文字, 如 <b>X</b> 和 <b>Y</b> , 达 5 个级别。<br>初始值: -  |
| <b>.KE</b>     | 中断: <b>yes</b><br>结束任意类型的保留。<br>初始值: -                                                |
| <b>.KF</b>     | 中断: <b>no</b><br>开始浮动保持; 文本填充剩余部分。<br>初始值: -                                          |
| <b>.KS</b>     | 中断: <b>no</b><br>开始保留; 在单个页上保留多个单元。<br>初始值: -                                         |
| <b>.LG</b>     | 中断: <b>yes</b><br>设置更大的字号; 将磅值提高 2。只对 <b>troff</b> 命令有效。<br>初始值: -                    |
|                | 中断: <b>no</b>                                                                         |

|                |                                                                                                   |
|----------------|---------------------------------------------------------------------------------------------------|
| <b>.LP</b>     | 开始左边块段落。<br>初始值: -<br>中断: <b>yes</b><br>复位: <b>yes</b>                                            |
| <b>.MC X</b>   | 设置多列。 <b>X</b> 是列宽。<br>初始值: -<br>中断: <b>yes</b>                                                   |
| <b>.ND [X]</b> | 表示页脚中没有日期; <b>X</b> 是封面上的日期。<br>初始值: <b>if t</b><br>中断: <b>no</b>                                 |
| <b>.NH X Y</b> | 设置编号页眉: <b>X=level</b> ; <b>X=0</b> , 复位; <b>X=S</b> , 设置为 <b>Y</b> 。<br>初始值: -<br>中断: <b>yes</b> |
| <b>.NL</b>     | 将磅值恢复为缺省值。只对 <b>troff</b> 命令有效。<br>初始值: <b>10p</b>                                                |
| <b>.OF X</b>   | 设置奇数页页脚 <b>X</b> (3 作为 <b>me</b> 宏、 <b>.tl</b> 请求的一部分)。<br>初始值: -<br>中断: <b>no</b>                |
| <b>.OH X</b>   | 设置偶数页页眉 <b>X</b> (3 作为 <b>me</b> 宏、 <b>.tl</b> 请求的一部分)。<br>初始值: -<br>中断: <b>no</b>                |
| <b>.P1</b>     | 打印第一页页眉。<br>初始值: <b>if TM</b>                                                                     |
| <b>.PP</b>     | 段落的首行缩进。<br>初始值: -<br>中断: <b>yes</b>                                                              |
| <b>.PT</b>     | 打印页头上的页标题。<br>初始值: <b>%</b><br>中断: <b>no</b>                                                      |

|                |                                                                                                                |
|----------------|----------------------------------------------------------------------------------------------------------------|
| <b>.PX X</b>   | 打印索引（目录）； <b>X</b> =不删除标题。<br><br>初始值: -<br><br>中断: <b>yes</b>                                                 |
| <b>.QP</b>     | 给段落加引号（缩进并缩短）。<br><br>初始值: -<br><br>中断: <b>yes</b>                                                             |
| <b>.R [X]</b>  | 复位: <b>yes</b><br>返回到罗马字体。以罗马体打印。如果 <b>X</b> 丢失，则等于字体 <b>.ft1</b> 。<br><br>初始值: <b>on</b><br><br>中断: <b>no</b> |
| <b>.RE</b>     | 退回（结束相对缩进的级别）。使用 <b>.RS</b> 请求。<br><br>初始值: <b>5n</b><br><br>中断: <b>yes</b>                                    |
| <b>.RP [X]</b> | 复位: <b>yes</b><br>以发行版论文格式打印标题页； <b>X=no</b> ，停止第一页上的标题。<br><br>初始值: -<br><br>中断: <b>no</b>                    |
| <b>.RS</b>     | 在一个缩进级别中右移（启动相对缩进级别）。使用 <b>.IP</b> 请求。<br><br>初始值: <b>5n</b><br><br>中断: <b>yes</b>                             |
| <b>.SG</b>     | 复位: <b>yes</b><br>设置签名行。                                                                                       |
| <b>.SH</b>     | 设置未编号的节标题（黑体）。<br><br>初始值: -<br><br>中断: <b>yes</b>                                                             |
| <b>.SM</b>     | 复位: <b>yes</b><br>设置更小的字号；将磅值减少 2。只对 <b>troff</b> 命令有效。<br><br>初始值: -<br><br>中断: <b>no</b>                     |
| <b>.TA</b>     | 将标签设置为 8n、16n ... ( <b>nroff</b> )；5n、10n ... ( <b>troff</b> )。<br><br>初始值: <b>8n、5n</b><br><br>中断: <b>no</b>  |
| <b>.TC X</b>   | 在末尾打印目录； <b>X</b> =不删除标题。<br><br>初始值: -<br><br>中断: <b>yes</b>                                                  |

|                |                                                                                                          |
|----------------|----------------------------------------------------------------------------------------------------------|
| <b>.TE</b>     | 结束 <b>tbl</b> 命令处理的表格。<br><br>初始值: -                                                                     |
| <b>.TH</b>     | 中断: <b>yes</b><br>结束表格的多页页眉。必须同 <b>.TS H</b> 请求一起使用。<br><br>初始值: -                                       |
| <b>.TL</b>     | 中断: <b>yes</b><br>设置标题行 (黑体, 大两号)。<br><br>初始值: -                                                         |
| <b>.TM</b>     | 中断: <b>yes</b><br>设置 UC Berkeley 论文方式。<br><br>初始值: <b>off</b>                                            |
| <b>.TS X</b>   | 中断: <b>no</b><br>开始表格。如果 <b>X</b> 为 <b>H</b> , 表格将打印所有页上的页眉。<br><br>初始值: -<br><br>中断: <b>yes</b>         |
| <b>.UL X</b>   | 复位: <b>yes</b><br>下划线 <b>X</b> , 即使对于 <b>troff</b> 命令也生效。<br><br>初始值: -                                  |
| <b>.UX X</b>   | 中断: <b>no</b><br>设置 UNIX; 第一次商标消息; 已添加 <b>X</b> 。<br><br>初始值: -                                          |
| <b>.XA X Y</b> | 中断: <b>no</b><br>设置另一个索引项; <b>X</b> =页; <b>X</b> =no, 无。<br><br>初始值: -                                   |
| <b>.XE</b>     | 中断: <b>yes</b><br>结束索引条目或者 <b>.IX</b> 请求条目系列。<br><br>初始值: -                                              |
| <b>.XP</b>     | 中断: <b>yes</b><br>段落的第一行突出; 其他缩进。<br><br>初始值: -<br><br>中断: <b>yes</b>                                    |
| <b>.XS X Y</b> | 复位: <b>yes</b><br>开始索引条目; <b>X</b> =页 <b>X</b> =no, 无; <b>Y</b> =缩进。<br><br>初始值: -<br><br>中断: <b>yes</b> |

|            |                                                                                           |
|------------|-------------------------------------------------------------------------------------------|
| <b>.1C</b> | 开始一列格式，在一个新页面上。<br>初始值: <b>on</b><br>中断: <b>yes</b><br>复位: <b>yes</b>                     |
| <b>.2C</b> | 开始两列格式。<br>初始值: -<br>中断: <b>yes</b><br>复位: <b>yes</b>                                     |
| <b>.-</b>  | 设置 <b>refer</b> 命令引用的开始。<br>初始值: -<br>中断: <b>no</b>                                       |
| <b>. 0</b> | 设置不可分类的引用类型的结尾。<br>初始值: -<br>中断: <b>no</b>                                                |
| <b>. N</b> | 对于杂志类文章, <b>N=1</b> (一)。对于书, <b>N=2</b> 。对于书籍类文章, <b>N=3</b> 。<br>初始值: -<br>中断: <b>no</b> |

## ms 寄存器

以下是一个数字寄存器及其缺省值的列表:

|           |                                                                   |
|-----------|-------------------------------------------------------------------|
| <b>PS</b> | 设置磅值。对段落生效。缺省值为 10。                                               |
| <b>VS</b> | 设置垂直间距。对段落生效。缺省值为 12。                                             |
| <b>LL</b> | 设置行长度。对段落生效。缺省值为 6i。                                              |
| <b>LT</b> | 设置标题长度。对下一页生效。缺省值为 <b>LL</b> 寄存器的值。                               |
| <b>FL</b> | 设置脚注长度。对下一个 <b>.FS</b> 请求生效。缺省值为 5.5i。                            |
| <b>PD</b> | 设置段落距离。对段落生效。缺省值为 1v (在 <b>nroff</b> 中), .3v (在 <b>troff</b> 中)。  |
| <b>DD</b> | 设置显示距离。对显示生效。缺省值为 1v (在 <b>nroff</b> 中), .5v (在 <b>troff</b> 中)。  |
| <b>PI</b> | 设置段缩进。对段落生效。缺省值为 5n。                                              |
| <b>QI</b> | 设置引用缩进。对下一个 <b>.QP</b> 请求生效。缺省值为 5n。                              |
| <b>FI</b> | 设置脚注缩进。对下一个 <b>.FS</b> 请求生效。缺省值为 2n。                              |
| <b>PO</b> | 设置页偏移量。对下一页生效。缺省值为 0 (零) (在 <b>nroff</b> )、1i (在 <b>troff</b> 中)。 |
| <b>HM</b> | 设置页眉边距。对下一页生效。缺省值为 1i。                                            |
| <b>FM</b> | 设置页脚边距。对下一页生效。缺省值为 1i。                                            |
| <b>FF</b> | 设置脚注格式。对下一个 <b>.FS</b> 请求生效。缺省值为 0 (零) (1、2、3 可用)。                |

当重新设置数字寄存器的值的时候, 请确保已经指定了合适的单位。设置行长度为 7i, 而不是仅仅是 7, 后者将会导致每行只输出一个字符。将 **FF** 寄存器设置为 1 (一) 可以禁止脚注上标。将其设置为 2 也可以禁止首行缩进。将 **FF** 寄存器设置为 3 可以生成同 **.IP** 请求相似脚注段落。

以下是一个 **ms** 宏中的可用字符串寄存器列表。这些字符串寄存器可以用于文本中的任何地方。

|            |                                               |
|------------|-----------------------------------------------|
| <b>\*Q</b> | 打开引号 (”在 <b>nroff</b> 中; `` 在 <b>troff</b> 中) |
|------------|-----------------------------------------------|

|                    |                                                |
|--------------------|------------------------------------------------|
| <code>\*U</code>   | 关闭引号（“ 在 <b>nroff</b> 中; ’ ’ 在 <b>troff</b> 中） |
| <code>\*-</code>   | 破折号（— 在 <b>nroff</b> 中; - 在 <b>troff</b> 中）    |
| <code>\*(MO</code> | 月份                                             |
| <code>\*(DY</code> | 日(当前日期)                                        |
| <code>\**</code>   | 自动编号的脚注                                        |
| <code>\*'</code>   | 重音符（字母前）                                       |
| <code>\*^</code>   | 抑音符（字母前）                                       |
| <code>\*^</code>   | 抑扬音符（字母前）                                      |
| <code>\*,</code>   | 变音符（字母前）                                       |
| <code>\*:</code>   | 变元音（字母前）                                       |
| <code>\*~</code>   | 代字号（字母前）。                                      |

当和 **.AM** 请求一起使用可用的扩展重音符定义时，这些字符串应该在要加重音的字母之后，而不是在这些字母之前。

注:

1. 一定要注意：浮动保持和常规保持可以转换为相同的空格，所以他们不能混合。
2. 日期格式仅限于美国英语格式。

## 用于 **mvt** 和 **troff** 命令的 **mv** 宏软件包

该软件包简化了视图的排版，以及不同大小的投影幻灯片。虽然若干个宏可以完成制造透明度所需的大多数格式化任务，但是由 **troff**、**tbl**、**pic** 和 **grap** 命令组成的整个设施可以用来完成更为困难的任务。

输出可以在多数终端中进行预览，特别是 Tektronix 4014。对于该设备，请指定 **-rX1** 标志（它在 **mvt** 命令带 **-D4014** 标志被调用时自动指定）。要预览其他终端上的输出，请指定 **-a** 标志。

**mv** 宏在以下的标题下进行汇总:

- Foil-Start 宏
- 级别宏
- 文本控制宏
- 缺省设置宏。

### Foil-start 宏

在以下九个宏中，名称的第一个字符（**V** 或 **S**）分别指视图和幻灯片。第二个字符表示片是否是方的（**S**）、小宽度（**w**）、小高度（**h**）、大宽度（**W**）或者大高度（**H**）。幻灯片比相应的视图要窄。幻灯片的长短比例要大于视图。结果是，幻灯片可以作为视图使用，但是视图不能作为幻灯片使用。另一方面，视图可以容纳稍多的文本。

**.VS** [*FoilNumber*] [*FoilID*] [*Date*] 开始一个方形视图。幻灯片大小为常宽各 7 英寸。foil-start 宏将所有变量（如缩进和磅值等）都重新设置为初始缺省值，不包括从上一个 foil-start 宏继承的 *FoilID* 和 *Date* 变量值。**.VS** 宏还会调用 **.A** 宏。

**.Vw**、**.Vh**、**.VW**、**.VH**、**.Sw**、**.Sh**、**.SW**、**.SH** 同 **.VS** 宏相同，除了这些宏启动的视图（**V**）或者幻灯片（**S**）是小宽度（**w**）、小高度（**h**）、大宽度（**W**）、或者大高度（**H**）的。

建议使用以下宏:

- **.VS** 用于方视图或者幻灯片。
  - **.Sw**（以及如果需要时的 **.Sh**）用于 35mm 幻灯片。
- 与 **.VS** 宏相同，除了幻灯片为 7 英寸宽，5英寸高之外。  
与 **.VS** 宏相同，除了幻灯片为 5 英寸宽，7英寸高之外。

**.Vw** [*FoilNumber*] [*FoilID*] [*Date*]

**.Vh** [*FoilNumber*] [*FoilID*] [*Date*]

|                                                                    |                                           |
|--------------------------------------------------------------------|-------------------------------------------|
| <b>.VW</b> [ <i>FoilNumber</i> ] [ <i>FoilID</i> ] [ <i>Date</i> ] | 与 <b>.VS</b> 宏相同, 除了幻灯片为 7 英寸宽, 5.4英寸高之外。 |
| <b>.VH</b> [ <i>FoilNumber</i> ] [ <i>FoilID</i> ] [ <i>Date</i> ] | 与 <b>.VS</b> 宏相同, 除了幻灯片为 7 英寸宽, 9英寸高之外。   |
| <b>.Sw</b> [ <i>FoilNumber</i> ] [ <i>FoilID</i> ] [ <i>Date</i> ] | 与 <b>.VS</b> 宏相同, 除了幻灯片为 7 英寸宽, 5英寸高之外。   |
| <b>.Sh</b> [ <i>FoilNumber</i> ] [ <i>FoilID</i> ] [ <i>Date</i> ] | 与 <b>.VS</b> 宏相同, 除了幻灯片为 5 英寸宽, 7英寸高之外。   |
| <b>.SW</b> [ <i>FoilNumber</i> ] [ <i>FoilID</i> ] [ <i>Date</i> ] | 与 <b>.VS</b> 宏相同, 除了幻灯片为 7 英寸宽, 5.4英寸高之外。 |
| <b>.SH</b> [ <i>FoilNumber</i> ] [ <i>FoilID</i> ] [ <i>Date</i> ] | 与 <b>.VS</b> 宏相同, 除了幻灯片为 7 英寸宽, 9英寸高之外。   |

注: **.VW** 和 **.SW** 幻灯片意味着是 9 英寸宽, 7 英寸高。但是, 因为排版机的页面通常只有 8 英寸宽, **.VW** 和 **.SW** 片只能被打印为 7 英寸宽, 5.4 英寸高, 在作为视图使用之前必须按照 9/7 的比例扩大。

## 级别宏

|                                           |                                                                                                                                                         |
|-------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>.A</b> [ <i>X</i> ]                    | 将后面的文本置于第一个缩进级别 (左边距)。X 变量的存在可以删除前面的文本的半行空格。                                                                                                            |
| <b>.B</b> [ <i>Mark</i> ] [ <i>Size</i> ] | 将后面的文本置于第二个缩进级别。文本之前有一个指定的标记 (缺省为一个重大项目符号)。Size 变量表示该标记的磅值的增加或者减小, 增大或者减小都是相对于 <i>prevailing</i> 磅值 (缺省为 0) 的。将 Size 变量的值设为 100, 会使标记的磅值等于 Mark 变量的缺省值。 |
| <b>.C</b> [ <i>Mark</i> ] [ <i>Size</i> ] | 与 <b>.B</b> 宏相同, 不过是用于第三个缩进级别。Mark 变量的缺省值为一个破折号。                                                                                                        |
| <b>.D</b> [ <i>Mark</i> ] [ <i>Size</i> ] | 与 <b>.B</b> 宏相同, 不过是用于第四个缩进级别。Mark 变量的缺省值为一个项目符号。                                                                                                       |

## 文本控制宏

|                                                                 |                                                                                                                                                                                                                                                                                                                                                         |
|-----------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>.I</b> [+/-] [ <i>Indentation</i> ] [ <i>A</i> ][ <i>X</i> ] | 改变当前的文本缩进 (不影响标题)。指定的缩进值都是以英寸为单位的, 除非指定尺寸。缺省值为 0。如果 <i>Indentation</i> 变量标记了正负号, 缩进会相应增加或者减少。A 变量的存在可以调用 <b>.A</b> 宏并将 X 变量 (如果有) 传递给它。                                                                                                                                                                                                                 |
| <b>.S</b> [ <i>Size</i> ] [ <i>Length</i> ]                     | 设置磅值和行长度。在 Size 变量中指定的值为磅值 (缺省为前一值)。如果 Size 变量的值为 100, 当前的 foil-start 宏的磅值将还原为 <i>initial</i> 缺省值。如果 Size 变量标有正负号, 表示变量值的增加或者减少 ( <b>.VS</b> 、 <b>.VH</b> 和 <b>.SH</b> 宏的缺省值为 18, 其他 foil-start 宏的缺省值为 14)。Length 变量指定行的长度 (除非指定尺寸, 否则都是以英寸为单位; <b>.Vh</b> 宏的缺省值为 4.2 英寸, <b>.Sh</b> 宏的缺省值为 3.8 英寸, <b>.SH</b> 宏的缺省值为 5 英寸, 其他 foil-start 宏的缺省值为 6 英寸)。 |
| <b>.T</b> <i>String</i>                                         | 将 <i>String</i> 变量值打印为居中的、放大的标题。                                                                                                                                                                                                                                                                                                                        |
| <b>.U</b> <i>String1</i> [ <i>String2</i> ]                     | 给 <i>String1</i> 变量值添加下划线, 并将其同 <i>String2</i> 变量值 (如果有的话) 相连接。不推荐使用该操作。                                                                                                                                                                                                                                                                                |

## 缺省设置宏

|                                                                |                                                                                                                                                                                           |
|----------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>.DF</b> [ <i>Number Name</i> ] ...                          | 设置字体位置。不能在幻灯片输入文本内显示; 必须跟在幻灯片的输入文本后, 但是还必须在下一个 foil-start 宏前。指定的数字是 Name 变量指定的字体的位置。 <b>.DF</b> 宏可以接受多达四对 <i>Number Name</i> 变量, 如 1 H。第一个 Name 变量指定主要字体。例如: <b>.DF 1 H 2 I 3 B 4 S</b> 。 |
| <b>.DV</b> [ <i>A</i> ] [ <i>B</i> ] [ <i>C</i> ] [ <i>D</i> ] | 更改缩进级别之间的垂直空格。由 A、B、C 或 D 变量指定的值分别是 <b>.A</b> 、 <b>.B</b> 、 <b>.C</b> 或 <b>.D</b> 宏的空格。所有的非空参数都必须指定尺寸。空参数不影响相应的空格。缺省设置为: <b>.DV .5v .5v .5v 0v</b> 。                                      |

**.S**、**.DF**、**.DV** 和 **.U** 宏不会造成中断。**.I** 宏只有在被超过一个的变量调用时才会造成中断。其他所有的宏都会造成中断。

**mv** 宏软件包还能识别对应于以下小写 **troff** 命令请求的大写同义词:

- **.AD**
- **.BR**

- **.CE**
- **.FI**
- **.HY**
- **.NA**
- **.NF**
- **.NH**
- **.NX**
- **.SO**
- **.SP**
- **.TA**
- **.TI**

**Tm** 字符串产生商标符号。

## 环境变量

**LANG** 确定用于是否查询的 *y* 的语言环境的等价值。允许的肯定响应在语言环境变量 **YESSTR** 中进行定义。如果没有设置 **LANG**，或者将其设置为空字符串，将使用缺省的 C 语言环境中的 **YESSTR**。

## 用于 **nroff** 和 **troff** 命令的 **nroff** 和 **troff** 请求

以下的 **nroff** 和 **troff** 请求包括在指定的工作文件或者标准输入中。当文件或者标准输入被使用 **nroff** 或 **troff** 命令处理时，**nroff** 和 **troff** 请求控制格式化输出的特性。**nroff** 和 **troff** 请求按功能分组，在以下各部分中：

- 数值参数输入
- 字体和字符大小控制
- 页面控制
- 文本填充、调节和居中
- 垂直间距
- 行长度和缩进
- 宏、字符串、转换和位置陷阱
- 数字寄存器
- 制表符、引导符和字段
- 输入和输出约定以及字符转换
- 连字符处理
- 三部分标题
- 输出行编号
- 输入的有条件接受
- 环境转换
- 从标准输入插入
- 输入和输出文件转换
- 杂项

对于写成 *+Number* 的数字变量，可以如下表示：



- *Number* 变量本身是一个绝对值。
- *+Number* 变量增加当前的设置值。
- *-Number* 变量相对于当前值减小变量。

注: 所有的数字参数都只能使用 ASCII 阿拉伯数字来表示。

该命令结尾注释在特定的适用的请求中被引用。

## 数字参数输入

**nroff** 和 **troff** 请求都接受带有附加量度指示符的数字输入, 如下表所示, 其中 *S* 是当前输入磅值, *V* 当前垂直行间距 (按基本单位计算)。 *C* 是正常字符宽度 (按基本单位计算)。

| 刻度          |                        | 基本 nroff 单位数              |
|-------------|------------------------|---------------------------|
| 指示符         | 含义                     |                           |
| <b>i</b>    | 英寸 (对于编辑与格式化程序, 与机器有关) | 240                       |
| <b>c</b>    | 厘米                     | 240x50/127                |
| <b>P</b>    | 12 点距 = 1/6 英寸         | 240/6                     |
| <b>m</b>    | Em = <i>S</i> 点        | <i>C</i>                  |
| <b>n</b>    | En = Em/2              | <i>C</i> (与 <i>Em</i> 相同) |
| <b>p</b>    | 点 = 1/72 英寸            | 240/72                    |
| <b>u</b>    | 基本单位                   | 1                         |
| <b>v</b>    | 垂直行间距                  | <i>V</i>                  |
| <b>k</b>    | 宽度单宽度假名                | <i>C</i>                  |
| <b>K</b>    | 宽度双宽度日语汉字              | 两个 <i>C</i>               |
| <i>none</i> | 缺省                     |                           |

注:

1. 如果选择了非日语汉字的输出设备, 就要使用 *en* 宽度。
2. 如果选择了非日语汉字的输出设备, 就要使用 *em* 宽度。

在 **nroff** 请求中, *em* 和 *en* 都等于 *C*, 具体值由输出设备决定; 最常用的值为 1/10 和 1/12 英寸。 **nroff** 请求中的实际字符宽度不需要都是相同的, 使用 **->** 等预定义字符串构建的字符的宽度经常会超出宽度范围。

日语支持: 在 **nroff** 命令的输出中, 所有双宽度的日本字符, 如所有日语汉字和一些片假名字符等都有一个等于  $2C$  的固定宽度值。所有的单宽度日本字符, 如一些片假名字符, 有等于 *C* 的固定宽度。

水平方向控制字符、垂直方向控制字符以及请求 **.nr**、**.if** 和 **.ie** 的缩放如下:

| 方向 | 缺省<br>测量               | 请求或功能                                                                                                                                         |
|----|------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------|
| 水平 | Em ( <i>m</i> )        | <b>.ll</b> , <b>.in</b> , <b>.ta</b> , <b>.lt</b> ,<br><b>.po</b> , <b>.mc</b> , <b>\h</b> , <b>\l</b>                                        |
| 垂直 | 垂直行<br>空格 ( <i>v</i> ) | <b>.pl</b> , <b>.wh</b> , <b>.ch</b> , <b>.dt</b> ,<br><b>.sp</b> , <b>.sv</b> , <b>.ne</b> , <b>.rt</b> , <b>\v</b><br><b>\x</b> , <b>\L</b> |

寄存器方向 基本单位 (u) .nr、.if、.ie  
或者有条件  
杂项 点 (p) .ps、.vs、\H、\s

所有其他请求都会忽略缩放指示符。当对已经包含有合适的缩放数字的数字寄存器进行解释以提供数字输入的时候，可能需要附加单位缩放指示符 **u** 来防止多余的不合适的缺省缩放。*Number* 可以指定为十进制分数的形式，但是最终存储的参数会被四舍五入为基本单位的整数。

## 字体和字符大小控制

**.bd Font Number** 在使用 **nroff** 时，通过将指定字体的字符重叠打印指定次数来人为加粗字体，或者在使用 **troff** 时通过按 *Number* -1 基本单位分两次打印每个字符来人为地使指定字体的字符加粗。如果没有指定 *Number* 变量，粗体方式将被关闭。*Font* 值必须是一个 ASCII 字体名称或者字体位置。在 **nroff** 命令中，**.bd** 请求的缺省设置为 3 3，表示该字体的字符位于位置 3（通常为粗体），将被叠印 3 次（也就是在一个位置打印 4 次）。

字体名称本身可以替换字体位置；例如 **.bd l 3**。*Number* 变量在功能上同 **nroff** 命令的 **-u** 标志相同。（当字符被物理打印时，粗体方式必须有效。）该请求可以影响 **.b** 一般数字寄存器的内容。

粗体方式仍必须有效，或者在物理输出时重新启动。您不能关闭 **nroff** 命令中的粗体方式，如果它由打印设备进行本地控制的话，例如 DASI 300。

初始值: Off

如果没有指定值: -

**.bd S Font Number** 无论何时指定的字体是当前字体，确保字符为特定的字体粗体。当字符被物理打印的时候，该方式必须有效。*Font* 值必须是一个 ASCII 字体名称或者字体位置。该方式仍必须有效，或者在物理输出时重新启动。

初始值: Off

如果没有指定值: -

**.cs Font Number** 将恒定的字符空间（宽）方式设置为 *Font* 变量值（如果已安装）。每个字符的宽度按照 *Number* 变量中指定的值除以 36 em 设置。如果没有指定 *M* 变量，em 的宽度为字符的磅值。如果 *M* 变量已经给出，宽度由 *M* 变量减磅值指定。所有受影响的字符都集中在该空格中，包括那些实际宽度大于该空格的字符。当指定字体为当前字体时，出现的特殊字体字符也一样处理。*Font* 值必须是一个 ASCII 字体名称或者字体位置。如果 *Number* 变量不存在，该方式关闭。当字符被物理打印的时候，该方式必须有效。该请求会被 **nroff** 命令忽略。相关的值是当前环境的一部分。该方式仍必须有效，或者在物理输出时重新启动。

初始值: Off

如果没有指定值: -

**.fpFont Number[ File ]** 指定字体位置。这是一个语句，表示指定的字体被安装在 *Number* 变量指定的位置上。*Font* 变量必须是一个单字符或者双字符的 ASCII 字体名称。

**警告:** 如果没有指定 *Font* 字体变量，则是一个不可恢复错误。

**.fp** 请求接受第三个可选变量，*File* 变量，它是包含指定字体的文件的实际路径名。*File* 变量值可以是任何合法的文件名，并且可以包含扩展字符。

日语支持: *File* 值可以是任何合法的文件名。值由排版机或者打印机决定。

初始值: -

如果没有指定值: 忽略

**.ft** *Font* 将字体样式更改为指定的字体，或者如果 *Font* 值是数值型的，将字体样式更改为该位置上安装的字体。或者，嵌入 **\fFont** 命令。字体名 **P** 被保留，用来指示以前的字体。*Font* 变量值必须是一个 ASCII 字体名或者字体位置。

如果使用由两个字符构成的字体名，可以使用 **.ft** 的另外一个形式，**\f**。相关的值是当前环境的一部分。值由排版机或打印机决定。

初始值：罗马体

**.ps** [+/-][*Number*] 如果没有指定值：上一个  
将磅值设置为 +/-*Number* 变量指定的值。虽然可以请求任何正的大小值，但是当出现无效的字号时，系统将使用与之最接近的有效字号。字号 0 指上一个字号。或者，**\sNumber** 或 **\s+/-Number**；如果 *Number* 值是两位数，则使用 **\s(Number** 或者 **\s+/(Number**。要兼容老版本的 **troff** 命令，两位数值 *n* = 10、11、12、14、16、18、20、22、24、28、和 36 的格式是有效的。

该请求被 **nroff** 命令忽略。相关的值是当前环境的一部分。

初始值：10 磅

**.ss** *Number* 如果没有指定值：上一个  
将空白字符的大小设置为指定数字除以 36 em。该大小是调节后的文本中的最小字空格。该请求被 **nroff** 命令忽略。相关的值是当前环境的一部分。

初始值：12/36 em

如果没有指定值：忽略

## 页面控制

**.bp** [+/-][*Number*] 指定中断页。弹出当前页，开始新一页。如果指定了 +/-*Number* 变量，它的值将成为新页面的编号。也可参考 **.ns** 请求。

该请求类似于 **.br** 请求，通常会引起换行。带控制字符 “”（而不是 “.”）调用此请求会禁止该中断功能。

初始值： *Number*=1

**.mk** *Register* 如果没有指定值： -  
在内部寄存器（与当前的转换级别相关联）或者在指定的寄存器（如果给出了的话）标记当前垂直位置（或者当前转换中的位置）。*Register* 变量是一个数字寄存器的 ASCII 名称。方式或者相关的值与当前的转换级别相关联。有关更多信息，请参阅 **.rt** 请求。

初始值：无

**.ne** *Number D* 如果没有指定值：内部  
表示对指定的垂直空间的需求。如果所需的页面空间（*Number*）比下一个陷阱（*D*）的距离大，出现一个 *D* 大小的前向垂直空间，由它激发陷阱。如果页面上没有保留下的陷阱，由 *D* 变量指定的大小就是到页底的距离。如果到下一个陷阱的距离（*D*）小于一个垂直行间距（*v*），在陷阱激发之前，仍然可以输出另一行。在转换中，由 *D* 指定的大小为到转换陷阱的距离（如果有的话），或者非常大。

*D* 的值通常也包含在 **.t Number** 寄存器中。方式或者相关的值与当前的转换级别相关联。

初始值： *Number*=1*V*

如果没有指定值： -

|                                   |                                                                                                                                                                                                                                                                                                                                                              |
|-----------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>.pl</b> [+/-][ <i>Number</i> ] | <p>将页面长度设置为 <i>+/-Number</i> 变量的值。<b>nroff</b> 命令中的内部限制约为 136 英寸，但是会随着 <b>troff</b> 命令中的设备类型而变化。<b>troff</b> 命令的最大良好工作值为 75 英寸。当前的页面长度在 <b>.p</b> 寄存器中可用。</p> <p>初始值: 11 英寸</p>                                                                                                                                                                              |
| <b>.pn</b> [+/-][ <i>Number</i> ] | <p>如果没有指定值: 11 英寸</p> <p>指定下一页（当出现时）的页编号由 <i>+/-Number</i> 变量指定。<b>.pn</b> 请求必须出现在文本被第一次打印之前或者中断对第一页的页号造成影响之前。当前的页号在 <b>%</b> 寄存器中。</p> <p>初始值: <i>Number=1</i></p>                                                                                                                                                                                          |
| <b>.po</b> [+/-][ <i>Number</i> ] | <p>如果没有指定值: 忽略</p> <p>指定页偏移量。当前的左页边距设置为 <i>+/-Number</i> 变量值。初始 <b>troff</b> 命令值提供了 1 英寸的左页边距。有关更多信息，请参阅“行长度和缩进”。当前的页偏移量在 <b>.o</b> 寄存器中可用。</p> <p>初始值: <b>0</b> 用于 <b>nroff</b> 命令; <b>1</b> 用于 <b>troff</b> 命令。</p>                                                                                                                                        |
| <b>.rt</b> [+/-][ <i>Number</i> ] | <p>如果没有指定值: 上一个</p> <p>仅向上返回到当前转换中的已标记的垂直位置。如果已给出 <i>+/-Number</i> 变量值（相对于当前位置），此位置就是由 <i>+/-Number</i> 变量指定的从页面或转换顶部算起的值。如果 <i>Number</i> 变量没有被指定，此位置将被上一个 <b>.mk</b> 请求标记。方式或者相关的值与当前的转换级别相关联。</p> <p><b>.sp</b> 请求而不是 <b>.rt</b> 请求可以在所有情况下使用，通过留空到存储在一个明确的寄存器中的一个绝对位置，例如，当使用顺序 <b>.mk Register . . . .sp\nRu</b> 时。</p> <p>初始值: 无</p> <p>如果没有指定值: 内部</p> |

## 文本填充、调整和居中

| <b>.ad</b> <i>Indicator</i> | <p>开始行调整。如果填充方式没有打开，调整将被延迟，直到填充方式被重新开启。如果 <i>Indicator</i> 变量存在，调整类型将像以下列表所显示的那样更改：</p> <table> <thead> <tr> <th>指示符</th> <th>调整类型</th> </tr> </thead> <tbody> <tr> <td><i>l</i></td> <td>只调整左边距。</td> </tr> <tr> <td><i>r</i></td> <td>只调整右边距。</td> </tr> <tr> <td><i>c</i></td> <td>居中。</td> </tr> <tr> <td><i>b</i> 或 <i>n</i></td> <td>调整双面边距。</td> </tr> <tr> <td>空白</td> <td>不改变。</td> </tr> </tbody> </table> <p>调整指示符可以是一个从 <b>.j</b> 寄存器中获取的数字。</p> | 指示符 | 调整类型 | <i>l</i> | 只调整左边距。 | <i>r</i> | 只调整右边距。 | <i>c</i> | 居中。 | <i>b</i> 或 <i>n</i> | 调整双面边距。 | 空白 | 不改变。 |
|-----------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|------|----------|---------|----------|---------|----------|-----|---------------------|---------|----|------|
| 指示符                         | 调整类型                                                                                                                                                                                                                                                                                                                                                                                                                                                         |     |      |          |         |          |         |          |     |                     |         |    |      |
| <i>l</i>                    | 只调整左边距。                                                                                                                                                                                                                                                                                                                                                                                                                                                      |     |      |          |         |          |         |          |     |                     |         |    |      |
| <i>r</i>                    | 只调整右边距。                                                                                                                                                                                                                                                                                                                                                                                                                                                      |     |      |          |         |          |         |          |     |                     |         |    |      |
| <i>c</i>                    | 居中。                                                                                                                                                                                                                                                                                                                                                                                                                                                          |     |      |          |         |          |         |          |     |                     |         |    |      |
| <i>b</i> 或 <i>n</i>         | 调整双面边距。                                                                                                                                                                                                                                                                                                                                                                                                                                                      |     |      |          |         |          |         |          |     |                     |         |    |      |
| 空白                          | 不改变。                                                                                                                                                                                                                                                                                                                                                                                                                                                         |     |      |          |         |          |         |          |     |                     |         |    |      |

## 日语支持:

### 指示符

*k*

### 调整类型

开启 `kinsoku shori` 处理（使用 `.ad n`、`.ad b` 或 `.ad l` 关闭）。

通常，日语文本行会填充到页边空白上，而不必考虑行首和行尾的字符。当 `kinsoku shori` 处理开启以后，开括号字符就不能位于行尾，关括号字符或者标点符号也不能位于行首。如果一行以开括号结尾，该行将被左缩短，括号将开始下一行。如果一行以关括号或者标点符号开头，前面的一行将被扩展，关括号或者标点符号结束前一行。在不支持日语汉字符的输出设备上请求日语 `kinsoku shori` 处理是无效的。

相关的值是当前环境的一部分。

初始值：调整，两者

如果没有指定值：调整

**.br**

指定中断。停止对正在被收集的行进行填充，行没有经过调整就输出。以空格字符开头的行和空文本行（空白行）也会引起中断。

初始值：-

**.ce** [*Number* ]

如果没有指定值：-

在当前的行长度减去缩进的范围内，居中对齐下一个指定的输入文本行号。如果 *Number* 变量等于 0，所有的余数都会被清空。中断将会出现在每个 *Number* 变量输入行后面。如果输入行太长，将会对其进行左调整。相关的值是当前环境的一部分。该请求通常会造成同 **.br** 请求相类似的行中断。带控制字符 ” ’ “ 调用该请求（而不是 ”:“）禁止该中断功能。

初始值：Off

**.fi**

如果没有指定值：编号=1

填充后继输出行。**.u** 寄存器的值在填充方式中为 1（一），在非填充方式中为 0（零）。相关的值是当前环境的一部分。该请求类似于 **.br** 请求，通常会引起换行。带控制字符 ” ’ “（而不是 ”:“）调用该请求禁止该中断功能。

初始值：填充

**.na**

如果没有指定值：-

指定非调整方式。关闭调整；右边距被消除。**.ad** 请求的调整类型未改变。如果填充方式开，还可以进行输出行填充。相关的值是当前环境的一部分。

初始值：无

**.nf**

如果没有指定值：-

指定非填充方式。随后的输出行既不能填充也不能调整。输入文本行被直接复制到输出行，而不考虑当前的行长度。相关的值是当前环境的一部分。该请求通常会造成同 **.br** 请求相类似的行中断。带控制字符 ” ’ “（而不是 ”:“）调用该请求禁止该中断功能。

初始值：填充

如果没有指定值：-

## 垂直间距

空白文本行

引起中断并输出一空白行，同 **.sp 1** 请求完全一样。

|                          |                                                                                                                                                                                                                                                       |
|--------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>.ls</b> <i>Number</i> | <p>将行间距设置为由 <i>+/-Number</i> 变量指定的值。<i>Number -1 Vs</i> (空白行) 变量的值附加到每个输出文本行。如果文本或者前一个附属空白行到达陷阱位置, 则省略附加空白行。相关的值是当前环境的一部分。</p> <p>初始值: <b>1</b></p>                                                                                                   |
| <b>.ns</b>               | <p>如果没有指定值: 上一个</p> <p>开启无间距方式。一旦开启, 无间距方式将禁止没有下一个页编号的 <b>.sp</b> 和 <b>.bp</b> 请求。无间距方式在输出一行时被关闭, 或者用 <b>.rs</b> 请求关闭。该请求通常会造成一个中断。</p> <p>初始值: 空格</p>                                                                                                |
| <b>.os</b>               | <p>如果没有指定值: -</p> <p>输出保存垂直间距。无间距方式无效。用于输出由前一个 <b>.sv</b> 所请求的一个垂直间距块。</p> <p>初始值: -</p>                                                                                                                                                              |
| <b>.rs</b>               | <p>如果没有指定值: -</p> <p>恢复间距。无间距方式关闭。该请求通常会造成一个中断。</p> <p>初始值: 无</p>                                                                                                                                                                                     |
| <b>.sp</b> <i>Number</i> | <p>如果没有指定值: -</p> <p>在每个方向上设置垂直间距。如果 <i>Number</i> 变量值为负, 移动将是向后 (向上) 的, 并且被限制在到页面顶部的距离内。向前 (向下) 的移动被截断为到最近的陷阱的距离。如果无间距方式打开, 不会出现任何间距。请参阅 <b>.ns</b> 和 <b>.rs</b> 请求。该请求类似于 <b>.br</b> 请求, 通常会引起换行。带控制字符 ”” (而不是 ”.”) 调用该请求禁止该中断功能。</p> <p>初始值: -</p> |
| <b>.sv</b> <i>Number</i> | <p>如果没有指定值: <b>1V</b></p> <p>保存指定大小的一个连续垂直块。如果到下一个陷阱距离大于 <i>Number</i> 变量的值, 指定的垂直间距将被输出。无间距方式无效。如果此距离小于指定的垂直间距, 不会立即输出垂直间距, 但是会被记忆用于接下来的输出 (请参阅 <b>.os</b> 请求)。随后的 <b>.sv</b> 请求会覆盖所有被记忆的 <i>Number</i> 变量值。</p> <p>初始值: -</p>                       |
| <b>.vs</b> <i>Number</i> | <p>如果没有指定值: <i>Number=1V</i></p> <p>将垂直基线间距的大小 <i>V</i> 设置为 <i>Number</i> 变量的值。瞬时的额外垂直间距可以由 <b>\x N</b> 来指定。相关的值是当前环境的一部分。</p> <p>初始值: <i>Number</i> 变量对于 <b>nroff</b> 命令等于 1/16 英寸, 对于 <b>troff</b> 命令等于 12 磅。</p> <p>如果没有指定值: 上一个</p>               |

## 行长度和缩进

|                                         |                                                                                                                                                                               |
|-----------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>.in</b> [ <i>+/-</i> ] <i>Number</i> | <p>将缩进设置为 <i>+/-Number</i> 变量的值。缩进在每一个输出行之前。相关的值是当前环境的一部分。该请求类似于 <b>.br</b> 请求, 通常会引起换行。带控制字符 ”” (而不是 ”.”) 调用此请求会禁止该中断功能。</p> <p>初始值: <i>Number=0</i></p> <p>如果没有指定值: 上一个</p> |
|-----------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

|                                |                                                                                                                                                                                                                                                                                                                                    |
|--------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>.ll</b> [+/-] <i>Number</i> | <p>将行长度设置为 <i>+/-Number</i> 变量的值。在 <b>troff</b> 命令中，最大行长度加页面偏移量是由设备决定的。相关的值是当前环境的一部分。</p> <p>初始值: 6.5 英寸</p>                                                                                                                                                                                                                       |
| <b>.ti</b> [+/-] <i>Number</i> | <p>如果没有指定值: 上一个<br/>指定临时缩进。下一个输出文本行按照 <i>+/-Number</i> 变量指定的相对于当前缩进的值缩进。<i>Number</i> 变量的值如果为负的话，可能会导致当前缩进向后退，从而使总缩进值为 0（零）（等于当前的页偏移量），但是不能小于当前的页偏移量。临时缩进仅应用于跟在请求后面的一个输出行；存储在 <b>.i</b> 寄存器中的当前缩进值不会发生变化。</p> <p>相关的值是当前环境的一部分。该请求通常会造成同 <b>.br</b> 请求相类似的行中断。带控制字符 “”（而不是 “.”）调用此请求会禁止该中断功能。</p> <p>初始值: -</p> <p>如果没有指定值: 忽略</p> |

## 宏、字符串、转换和位置陷阱

|                                            |                                                                                                                                                                                                                                                                                                                                                      |
|--------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>.am</b> <i>Macro1</i> [ <i>Macro2</i> ] | <p>附加 <i>Macro 1</i>;; 附加 <b>.de</b> 请求的版本。<i>Macro1</i> 和 <i>Macro2</i> 变量都必须是一个或者两个 ASCII 字符。<i>Macro2</i> 是用来结束转换的终止序列。</p> <p>初始值: -</p>                                                                                                                                                                                                         |
| <b>.as</b> <i>StringName</i> <i>String</i> | <p>如果没有指定值: <b>.Macro2=..</b><br/>将指定的字符串附加给由 <i>StringName</i> 变量指定的值; 附加 <b>.ds</b> 请求的版本。<i>StringName</i> 变量的值必须是一个或者两个 ASCII 字符。</p> <p>初始值: -</p>                                                                                                                                                                                              |
| <b>.ch</b> <i>Macro</i> [ <i>Number</i> ]  | <p>如果没有指定值: 忽略<br/>将指定的宏的陷阱位置更改为 <i>Number</i> 变量指定的值。如果不存在 <i>Number</i> 变量，陷阱（如果有的话）将被除去。<i>Macro</i> 变量必须是一个或者两个 ASCII 字符。</p> <p>初始值: -</p>                                                                                                                                                                                                      |
| <b>.da</b> [ <i>Macro</i> ]                | <p>如果没有指定值: -<br/>转换，附加指定的宏和附加 <b>.di</b> 请求的版本。<i>Macro</i> 变量必须是一个或者两个 ASCII 字符。方式或者相关的值与当前的转换级别相关联。</p> <p>初始值: -</p>                                                                                                                                                                                                                             |
| <b>.de</b> <i>Macro1</i> [ <i>Macro2</i> ] | <p>如果没有指定值: 结束当前的转换<br/>定义或者重新定义由 <i>Macro1</i> 变量指定的值。宏的内容在下一个输入行上开始。输入行在复制方式中被复制，直到定义被一个以 <b>.Macro2</b> 开始的行停止。如果 <i>Macro2</i> 变量不存在，定义由 “.” 开始的行停止。一个宏可以包含 <b>.de</b> 请求，只要停止宏不同或者包含的定义结束符被隐藏。“.” 可以被隐藏为 “\.”，它以 “\..” 的形式复制，并被重读为 “.”。<i>Macro1</i> 和 <i>Macro2</i> 变量都必须是一个或者两个 ASCII 字符。</p> <p>初始值: -</p> <p>如果没有指定值: <b>.Macro2=..</b></p> |

**.di** [*Macro*]

转换输出到指定的宏。在转换过程中会发生普通文本处理，除非没有执行页面偏移。当 **.di** 或者 **.da** 请求没有变量时，转换结束。当使用嵌套转换被时，不会显示该类型的额外请求。*Macro* 变量必须是一个或者两个 ASCII 字符。方式或者相关的值与当前的转换级别相关联。

初始值: -

如果没有指定值: 结束

**.ds** *StringName* *String*

将由 *StringName* 变量指定的字符串定义为包含由 *String* 字符串指定的值。*String* 中的所有初始双引号都会被取消，以允许初始空白。*StringName* 变量必须是一个或者两个 ASCII 字符。

**.ds** *StringName* ^A

提供一个替换 **.ds** 语法，该语法可以将消息编目用于独立于语言的字符串定义。

<*SetNumber*>

在特定语言环境编目内的消息 *SetNumber* 和 *MessageNumber* 的基础上，消息编目在复制方式中被读取，相应的消息被置于 *StringName* 变量中。为了向后兼容性，可以省略指定消息集和消息号的初始序列。ASCII 码控制符 A (^A) 为消息标识、缺省消息和可选自变量列表定界。

<*MessageNumber*>

[^A"<*DefaultMessage*> "]

[^A<*Argument*>

ASCII 码控制符 B (^B) 为单独的可选自变量列表定界。

^B<*Argument*>

<*Argument*>...]

在以下示例中，

```
.ds {c ^A2 41^A"ERROR: (%1$s) input line \  
%2$s" ^A\n(.F^B\n(.c
```

2 是消息集的编号。

41 是消息号。

引号内的文本 ("...") 是默认的消息。

\n(.F 是当前输入文件的名称。

\n(.c 是从输入文件中读取的行号。

如果您假设 **troff** 命令以如下条件运行:

- 在消息集 2 中的 41 号消息同缺省消息相匹配。
- 当前的输入文件为 `paper.doc`
- **.ds** 伪指令在输入文件的行 124 上。

然后字符串 {c 被定义为:

错误: (paper.doc)输入 行 123

其他示例为:

```
.ds {c ^A2 41  
/* Without optional default message */
```

```
.ds {c ^A2 41^A"ERROR: (%1$s) input file \  
%2$s" /* Without optional arguments */
```



如果集编号和消息号都被设置为零，则以当前本地格式返回当前的日期。可以在缺省消息字段中定义用户定义日期格式字符串。用户定义格式字符串必须符合 *AIX 5L Version 5.3 Technical Reference: Base Operating System and Extensions* 中的 **strftime** 函数所规定的转换规格。

在以下示例中：

```
.ds DT^A0 0
```

如果当前的日期是美国英语语言环境中的 1991 年 7 月 10 日，DT 将被定义为 7/10/91。

```
.ds DT^A0 0^A"今天是 %B %d, %Y"
```

如果当前日期是美国英语环境中的 1991 年 7 月 10 日，DT 将被定义为今天是 1991 年 7 月 10 日。

第二个语法方法使用不广泛。它在随系统一同提供的 **nroff** 和 **troff** 宏文件中使用，主要便于内部生成的消息的国际化。

初始值：-

如果没有指定值：忽略

#### **.dt** *Number Macro*

当前的转换中，在 *Number* 变量指定的位置上安装转换陷阱，以开始指定的宏。另一个 **.dt** 请求重新定义转换陷阱。如果没有给出任何变量，转换陷阱将被除去。*Macro* 变量必须是一个或者两个 ASCII 字符。方式或者相关的值与当前的转换级别相关联。

初始值：-

如果没有指定值：关闭

#### **.em** *Macro*

当所有的输出都结束时，调用特定的宏。作用同指定的宏的内容处于最后被处理的文件的末尾相同。指定的宏必须是一个或者两个 ASCII 字符。

初始值：无

如果没有指定值：无

#### **.it** *Number macro*

设置一个输入行计数陷阱来在由 *Number* 变量指定的文本输入行编号被读取（不计控制或请求行）之后调用指定的宏。文本可以是内联文本，也可以是显式调用（通过内联调用）或者隐含调用（通过陷阱）的宏提供的文本。*Macro* 变量必须是一个或者两个 ASCII 字符。相关的值是当前环境的一部分。

初始值：-

如果没有指定值：关闭

#### **.rm** *Name*

除去指定的请求、宏或字符串。从名称列表中除去 *Name* 变量值，所有与之有关的存储空间都被释放。随后的引用无效。*Name* 变量必须是一个或者两个 ASCII 字符。

初始值：-

如果没有指定值：忽略

#### **.rn** *Name1 Name2*

将由 *Name1* 变量指定的请求、宏或者字符串值重命名为 *Name2* 变量指定的值。*Name1* 和 *Name2* 变量的值都必须是一个或者两个 ASCII 字符。

初始值：忽略

如果没有指定值：-

**.wh** *Number Macro*

安装一个陷阱来在页面上的由 *Number* 变量指定的位置上调用指定的宏。负的 *Number* 变量值可被解释为与页面底部有关。任何以前安装在页面上由 *Number* 变量指定的位置上的宏都会被 *Macro* 变量的值所替换。*Number* 变量值为 0 表示页面顶部。如果不存在 *Macro* 变量，在由 *Number* 变量指定的页面位置上发现的第一个陷阱将被除去（如果有的话）。*Macro* 变量必须是一个或者两个 ASCII 字符。

初始值: -

如果没有指定值: -

## 数字寄存器

**.af** *Register Indicator*

为指定的寄存器指定分配 *Indicator* 变量所指定的格式。*Register* 变量必须是一个或者两个 ASCII 字符。可用的格式 *Indicator* 变量的值如下:

指示符 编号序列

**1** 0,1,2,3,4,5, . . .

**001** 000,001,002,003,004,005, . . .

**i** 0,i,ii,iii,iv,v, . . .

**I** 0,I,II,III,IV,V, . . .

**a** 0,a,b,c, . . . ,z,aa,ab, . . . ,zz,aaa, . . .

**A** 0,A,B,C, . . . ,Z,AA,AB, . . . ,ZZ,AAA, . . .

一个有 *N* 位的阿拉伯格式指示符（例如 000000001）表示一个 *N* 位数的字段宽度。只读寄存器和宽度功能总是使用阿拉伯数字格式表示的。

日语支持: 以下值指定了用于格式化日语汉字的数字输出的字符宽度:

**k** 数字按照日语汉字字符串进行格式化。如果在指定非日语汉字代码集时要求使用此格式, 将打印警告信息并使用 **1** 格式。

初始值: 阿拉伯字体

如果没有指定值: -

**.nr** *Register +/-Number1 Number2*

参照上一个值（如果有的话），为指定的寄存器分配由 *+/-Number* 变量指定的值。自动增长的增量设定为 *Number2* 变量值。*Register* 变量必须是一个或者两个 ASCII 字符。

初始值: -

如果没有指定值: -

**.rr** *Register*

除去指定的寄存器。如果许多寄存器都是动态创建的, 则可能需要除去一些不需要的寄存器来为新的寄存器恢复内存空间。*Register* 变量必须是一个或者两个 ASCII 字符。

初始值: -

如果没有指定值: -

## 制表符、引导符和字段

- .fc Delimiter Indicator** 将字段定界符设置为指定的定界符；填充指示符设置为空格字符或者指定的指示符。如果缺少变量，字段机制将被关闭。*Delimiter* 变量的值和 *Indicator* 变量的值必须是 ASCII 字符。
- 初始值: Off
- .lc Character** 如果没有指定值: 关闭  
将引导符重复字符设置为指定字符，或者取消指定动作。*Character* 变量的值必须是一个 ASCII 字符。相关的值是当前环境的一部分。
- 初始值: .
- .taStop [Type]...** 如果没有指定值: 无  
设置制表符停止位。缺省的制表符停止位对于 **nroff** 命令每 8 个字符设置一个，对于 **troff** 命令为每半英寸设置一个。多个 *StopType* 对可以通过使用空格隔开的方法来指定；前面有 +（加号）的值会被当作相对于前一个停止值的增量来处理。
- 指定的类型决定如何在制表符停止位上调整文本。*Type* 的值如下：
- | 类型       | 调整   |
|----------|------|
| <b>R</b> | 右对齐  |
| <b>C</b> | 中间对齐 |
| 空白       | 左对齐  |
- 相关的值是当前环境的一部分。
- 初始值: 对于 **nroff** 命令为 8 ens, 对于 **troff** 命令为 0.5 英寸。
- .tcCharacter** 如果没有指定值: 无  
将制表符重复字符设置为指定字符，或者除去指定动作。*Character* 变量的值必须是一个 ASCII 字符。相关的值是当前环境的一部分。
- 初始值: 无
- 如果没有指定值: 无

## 输入/输出约定和字符翻译

- .ccCharacter** 将基本的控制字符设置为指定的字符，或者重新设置为“.”。*Character* 变量的值必须是一个 ASCII 字符。相关的值是当前环境的一部分。
- 初始值: .
- 如果没有指定值: .
- .cu [Number]** **.ul** 请求的一个变量会导致每个字符都会被加上下划线，也会导致在受影响的输出行中没有行中断发生。即 **.cu** 请求后的每个输出空格都与不可填充的空格相似。**.cu** 请求同在 **troff** 命令中的 **.ul** 请求完全相同。相关的值是当前环境的一部分。
- 初始值: Off
- 如果没有指定值: 编号=1

- .c2** *Character* 将无中断控制字符设置为指定的字符或者复位为“'”。*Character* 变量的值必须是一个 ASCII 字符。相关的值是当前环境的一部分。
- 初始值: '
- .ec** *Character* 如果没有指定值: ' 将转义字符设置为 \ (反斜线符号) 或者由 *Character* 变量指定的值 (如果给出了的话)。*Character* 变量的值必须是一个 ASCII 字符。
- 初始值: \
- .eo** 如果没有指定值: \ 关闭转义机制。
- 初始值: 开启
- .lg** [*Number*] 如果没有指定值: - 如果 *Number* 变量的值不存在或者非零的话, 打开连字方式; 如果 *Number* 变量的值为 0 的话, 关闭连字方式。如果 *Number* 变量的值为 2, 只有双字符连字会被自动调用。请求、宏、字符串、寄存器或文件名以及复制方式中禁止使用连字方式。在 **nroff** 命令中, 该请求不起作用。
- 初始值: 开启, 对于**troff** 命令
- .tr** *Character1 Character2 Character3 Character4* 如果没有指定值: 开启 将由 *Character1* 变量指定的字符的值翻译为 *Character2* 的变量值, 将由 *Character3* 变量指定的字符值翻译为 *Character4* 的变量值。如果给出了奇数个字符, 最后一个将被映射到空格字符中。要保持一致, 从输入到输出的时间内必须保持特殊的转化在起作用。所有指定的字符都必须是 ASCII 字符。要重新设置 **.tr**请求, 在请求后接上一个复制给出的变量。
- 例如, 以下 **.tr** 请求:
- ```
.tr aAbBc<C,>
```
- 可以通过输入以下命令来重新设置:
- ```
.tr aabbcc
```
- 在逻辑输出之前, 它必须在起作用。
- 初始值: 无
- .ul** [*Number*] 如果没有指定值: - 对于 *Number* 变量指定的输入文本行数字在 **nroff** 命令中加下划线 (或在 **troff** 命令中初始化)。实际上是切换成下划线字体, 要保存当前的字体以备以后恢复用。**.ul** 请求范围内的其他字体更改生效, 但是恢复只能撤销最后一次的更改。**.tl** 请求生成的输出会受到字体更改的影响, 但是不会减小 *Number* 变量的值。有关更多信息, 请参阅“三部分标题”一节。如果指定值大于 1, 调用陷阱的宏非常有可能在该范围内提供文本行; 环境切换可以避免发生这样的情况。
- 相关的值是当前环境的一部分。
- 初始值: Off
- 如果没有指定值: 编号=1

**.uf** *Font* 将字体集加下划线至 *Font* 变量指定的值。在 **nroff** 命令中, *Font* 变量不可以在位置 1 上 (Times Roman 为初始值)。 *Font* 变量的值必须是一个 ASCII 字体名。

初始值: 斜体

如果没有指定值: 斜体

## 连字符连接

**.hc** *Character* 会将连字指示符设置为 *Character* 变量指定的值或缺省值。该指示符不在输出中显示。 *Character* 变量的值必须是一个 ASCII 字符。相关的值是当前环境的一部分。

初始值: \%

如果没有指定值: \%

**.hw** *Wordl...* 指定由减号充当的连字符在单词中的磅值。以单词 s 结尾的版本是默示的; 也就是说, *dig-it* 暗示 *dig-its*。去除后缀之前、之后检查该列表。可用的间距是 1024 个字符, 即 50 到 100 个单词。

初始值:

如果没有指定值: 忽略

**.hy** *Number* 如果指定的数字等于或大于 1, 则自动打开连字符连接; 如果指定的数字等于 0 (等于 **.nh** 请求), 则关闭连字符连接。如果指定的数字是 2, 最后几行 (导致陷阱的那些行) 不被连字符连接。如果指定的数字是 4 或 8, 则单词的开头两个字符或结尾两个字符都不能被分开写。这些值是可以相加的; 比如: 14 调用 3 种限制 (2, 4 和 8 相加等于 14)。

相关的值是当前环境的一部分。

初始值: 没有连字符连接

如果没有指定值: 使用连字符连接

**.nh** 关闭自动的连字符连接。相关的值是当前环境的一部分。

初始值: 没有连字符连接

如果没有指定值: -

## 三部分标题

**.lt** [+/-][*Number*] 设置由 +/-*Number* 变量指定的标题长度值。行长度和标题长度是彼此独立的。在标题上不使用缩进, 经管可以使用页面偏移。相关的值是当前环境的一部分。

初始值: 6.5 英寸

如果没有指定值: 上一个

**.pc** *Character* 将页号字符设置为指定的字符或者将其除去。页号寄存器仍为 %。 *Character* 变量的值必须是一个 ASCII 字符。

初始值: %

如果没有指定值: 关闭

**.tl**'Left'Center'Right'

*Left*, *Center* 和 *Right* 变量代表的字符串分别为在当前的标题长度内的左对齐、居中对齐和右对齐。所有的字符串都可以为空，字符串的重叠也是允许的。如果在任何字段中有页号字符（初始为 %），则用当前页号替换它，该页号的格式可以是指定给 % 寄存器的格式。未在字符串中显示的所有的 ASCII 字符都可以作为字符串的定界符。

初始值: -

如果没有指定值: -

## 输出行编号

**.nm** [+/-]  
[Number] [M] [S] [I]

打开行号方式。如果指定了 *M* 变量，则只打印行号为 *M* 变量的值的整数倍的行。如果 *M* 变量不存在（缺省值为 *M=1*），则打印所有行。当使用行号方式时，一个三位的阿拉伯数字加上一个空格被预先挂起到输出文本行。此时文本行偏移四位空格，其他情况下行长保持不变。如果给定 *S* 变量，该变量将指定行号和文本之间的空格位数（缺省值为 *S=1*）。如果给定 *I* 变量，该变量将指定行号前缩进的空格位数（缺省值为 *I=0*）。

相关的值是当前环境的一部分。

初始值: -

如果没有指定值: 关闭

**.nn**Number

暂挂行编号。指定编号行未被编号。相关的值是当前环境的一部分。

初始值: -

如果没有指定值: 编号=1

## 有条件的输入接收

*Condition* 变量指定以下单字符名称中的一个:

|                                                                |                                                                                                                                                |
|----------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>o</b>                                                       | 如果当前页是奇数页。                                                                                                                                     |
| <b>e</b>                                                       | 如果当前页是偶数页。                                                                                                                                     |
| <b>t</b>                                                       | 如果格式化程序是 <b>troff</b> 命令。                                                                                                                      |
| <b>n</b>                                                       | 如果格式化程序是 <b>nroff</b> 命令。                                                                                                                      |
| <b>.if</b> <i>Condition</i> <i>Anything</i>                    | 如果 <i>Condition</i> 变量指定的值是真，则接受 <i>Anything</i> 变量指定的值为输入；多行的情况，使用 $\backslash\{Anything\}$ 。                                                 |
| <b>.if</b> <i>!Condition</i> <i>Anything</i>                   | 如果 <i>Condition</i> 变量指定的值是假，则接受 <i>Anything</i> 变量指定的值为输入。                                                                                    |
| <b>.if</b> <i>Number</i> <i>Anything</i>                       | 如果表达式表明 <i>Number</i> 变量的值大于 0，则接受 <i>Anything</i> 变量指定的值为输入。                                                                                  |
| <b>.if</b> <i>!Number</i> <i>Anything</i>                      | 如果表达式表明 <i>Number</i> 变量的值小于或等于 0，则接受 <i>Anything</i> 变量指定的值为输入。                                                                               |
| <b>.if</b> ' <i>String1</i> ' <i>String2</i> ' <i>Anything</i> | 如果 <i>String1</i> 变量的值等于 <i>String2</i> 变量的值，则接受 <i>Anything</i> 变量指定的值为输入。在 <i>String1</i> 和 <i>String2</i> 变量中没有出现的所有的非空 ASCII 字符都可以被用作定界符。  |
| <b>.if</b> <i>!'String1</i> ' <i>String2</i> ' <i>Anything</i> | 如果 <i>String1</i> 变量的值不等于 <i>String2</i> 变量的值，则接受 <i>Anything</i> 变量指定的值为输入。在 <i>String1</i> 和 <i>String2</i> 变量中没有出现的所有的非空 ASCII 字符都可以被用作定界符。 |
| <b>.el</b> <i>Anything</i>                                     | 指定 <b>if/else</b> 条件的 <b>else</b> 部分。                                                                                                          |
| <b>.ie</b> <i>Condition</i> <i>Anything</i>                    | 指定 <b>if/else</b> 条件的 <b>if</b> 部分，指定的依据是 <i>Condition</i> 变量的值。能够与前面提到的所有的 <b>.if</b> 请求的格式一起使用。                                              |

## 环境切换

- .ev Environment** 切换到指定的环境。由 *Environment* 变量指定的值必须是 0、1 或 2。切换是以下推的形式进行的，所以恢复到前一个环境只能依靠 **.ev** 请求来执行，而不是使用某个特定的引用。
- 初始值: *Environment=0*
- 如果没有指定值: 上一个

## 标准输入的插入内容

- .ex** 从 **nroff** 命令或 **troff** 命令中退出。文本处理停止，就象所有的输入已经结束时的情形。
- 初始值: -
- 如果没有指定值: -
- .rd Prompt** 读取标准输入中的插入内容，直至在一行内查找到两个换行符。如果标准输入是用户的键盘，则指定的提示符（或 ASCII BEL 字符）被写到用户终端。**.rd** 请求的行为与宏相似，附加的变量被放在 *Prompt* 变量的后面。
- 初始值: -
- 如果没有指定值: *Prompt=the ASCII BEL 字符*

## 输入、输出文件切换

- .cfFile** 连续复制指定文件的内容到 **troff** 命令在此处的输出文件。如果文件中的移动不恢复到当前的水平和垂直位置的话，会出现问题。
- 初始值: -
- 如果没有指定值: -
- .lf Number File** 校正 **troff** 命令对当前的行号（由 *Number* 变量指定）和在出现错误消息时使用的当前文件（由 *File* 变量指定）的解释。
- 初始值: -
- 如果没有指定值: -
- .nx File** 使用指定的文件作为输入文件。当前的文件被认定已结束，立刻将输入转换到指定的文件。
- 初始值: -
- 如果没有指定值: 结束文件
- .pi Program** 将输出传输至指定的程序。该请求必须在所有打印开始前提出。没有变量被传输到指定的程序。
- 初始值: -
- 如果没有指定值: -

**.so** *File* 转换源文件。指定的文件被转换成顶端输入（文件读取）级别。该文件结束后，仍然从源文件获取输入。**.so** 请求可以嵌入其他程序。

当出现 **.so** 请求时，立刻对指定的文件进行处理。此时，中止对初始文件（如，仍然处于激活状态的宏）的处理。

如果有必要的话，一个文件可以在被 **.so** 请求调用之前被预处理。**eqn**、**tbl**、**pic** 和 **grap** 命令不通过 **.so** 请求来处理对象代码。

初始值: -

如果没有指定值: -

## 其他项

**.ab** *Text* 将由 *Text* 变量指定的值打印到诊断输出（通常是终端），然后结束，不再进行进一步的处理。如果文本丢失，则打印 User Abort 消息，并刷新输出缓冲区。该请求在交互式调试时使用以进行强制输出。

**.ab** <sup>^A</sup><*SetNumber*> 提供备用语法，允许对独立语言的异常终止消息使用消息编目。打印由诊断输出（通常是终端）的参数指定的合适消息，然后结束，不再进行进一步的处理。如果没有参数，则消息编目如下：

[<sup>^A</sup><*Default*> " ]  
[<sup>^A</sup><*Argument*> troff: 用户 异常终止, 行编号 文件 文件名

<sup>^B</sup><*Argument*> 为输出。输出缓冲区被刷新。该请求在交互式调试时使用以进行强制输出。  
<sup>^B</sup><*Argument*>...]

根据特定语言环境的编目中的消息 *SetNumber* 和 *MessageNumber* 变量，用户以复制方式读取消息编目，与其相关的消息被写入用户终端。为了向后兼容性，可以省略指定消息集和消息号的初始序列。ASCII 代码控制符 A (<sup>^A</sup>) 为消息识别符、缺省的消息和可选的自变量列表定界。ASCII 代码控制符 B (<sup>^B</sup>) 为单独的可选自变量列表定界。

在以下示例中:

```
.ab ^A2 42^A"Processing has been terminated \  
at line %1$s."^A\n(c.
```

2 是消息集的编号。

42 是消息号。

引号中的文本 "... " 是缺省的消息。

\n(c. 是从输入文件中读取的行号。

如果假设 **troff** 命令在以下条件下运行:

- 处于消息集 2 和编号 42 的消息与缺省的消息相匹配。
- **.ab** 伪指令处于输入文件的 124 行。

然后以下内容将在用户终端上显示:

处理 已经 被 终止 在 行 123.

初始值: -

如果没有指定值: 用户取消



## **.Dt** *Parameter*

定义 **nroff** 或 **troff** 请求的日期返回格式。根据缺省设置，如果没有可选 *Parameter*，则使用由 **LC\_TIME** 分类设置的当前语言环境指定特定语言环境的日期格式。这与 **strftime** 的 "%x" 格式说明符相符合。*Parameter* 是格式字符串，同与 *AIX 5L Version 5.3 Technical Reference: Base Operating System and Extensions* 中的 **strftime** 函数共用的格式字符串一样。参考该函数，以了解格式说明符的完整列表。

例如，

```
.Dt "%A, %B %d, %Y (%T)"
```

为使用英语的地区提供了以下输出：

```
星期四, 1 月 31 日, 1991 (10:40:00)
```

%A 格式被具体地区的星期名称替代。%B 格式被具体地区的月份名称替代。%d 格式被使用两位格式的日期替代。%Y 格式被十进制数的世纪年份替代。%T 格式被十进制数的时（24小时制）、分、秒时间代替。该格式提供闰秒和双闰秒。

## **.fl**

刷新输出缓冲区。该请求类似于 **.br** 请求，通常会引起换行。带控制符 ” ’ “（而不是”.”）调用该请求禁止该中断功能。

初始值： -

如果没有指定值： -

## **.ig** *Macro*

忽略输入行。**.ig** 请求和 **.de** 请求的操作完全相同，除了前者的输入被废弃。获取更多相关信息，请参阅 ”宏、字符串、转换和位置陷阱“。输入以复制方式被读取，所有的自动递增寄存器都会受到影响。*Macro* 变量必须是一个或者两个 ASCII 字符。

初始值： -

如果没有指定值： **.Macro=.**

## **.mc** [*Character*] [*N*]

使用指定字符作为边界字符来显示到每个非空文本行（**.tl** 请求生成的文本行除外）的右边界的指定的距离（*N*）。如果输出行过长（非填充方式下经常出现的情况），则将字符附加到行。如果没有给定 *N* 变量，则使用前一个 *N* 变量。第一个 *N* 变量在 **nroff** 命令中是 0.2 英尺，在 **troff** 命令中是 1 em。

相关的值是当前环境的一部分。

初始值：在 **nroff** 中是 2；在 **troff** 中是 1 em。

如果没有指定值：关闭

## **.pm** [*Character*]

打印宏。在用户终端中打印所有已定义的宏和字符串的名称和大小。如果将一个 ASCII 字母数字字符作为变量，则只打印总大小。在 128 字符块中给出来大小。

初始值： -

如果没有指定值：全部

## **.sy** *Command* [标志]

指定的命令被运行但是它的输出在此没有被捕获。指定命令的标准输入被关闭。必须将输出在输出文件中明确的保存，以备以后进行处理。通常，**.sy** 伪指令后面会跟随 **.so** 伪指令，用来包含前一个命令的结果。

例如：

```
.sy date > /tmp/today
今天 是
.so /tmp/today
```

初始值： -

如果没有指定值： -

**.tm** *String* 指定的字符串被写入用户终端。

**.tm** *^A*<SetNumber> 根据特定语言环境目录中的消息集的编号和消息号，消息目录被以复制方式进行读取，相关消息被写入用户终端。为了向后兼容性，可以省略指定消息集和消息号的初始序列。

<MessageNumber> ASCII 码 Ctrl-A ^A 将消息识别符、缺省的消息和可选的自变量列表定界。ASCII 码 Ctrl-B [^A"<DefaultMessage> "]" ^B 为单独的可选自变量列表定界。

[^A<Argument> ^B<Argument> ^B<Argument> ...]

在以下示例中：

```
.tm ^A2 23^A"The typesetter is %1$s.On line
%2$s."^A\*(.T^B\n(c.
```

2 是消息集的编号。

23 是消息号。

引号中的文本 "... " 是缺省的消息。

\\*(.T 是 troff 中 -T 值的第一个自变量。

\n(c. 是从输入文件中读的行号。

如果假设 **troff** 命令在以下条件下运行：

- 处于消息集 2 和编号 23 的消息与缺省的消息相匹配。
- 命令行含有 **troff**，它正在使用 -T 选项和 PSC 设备。
- **.tm** 伪指令处于输入文件的 539 行。

然后以下内容将在用户终端上显示：

```
The typesetter is psc. On line 538.
```

在 `/usr/lib/nls/msg/$LANG/macros.cat` 中可以找到特定语言环境的消息编目。

初始值： -

如果没有指定值： 新行

## 注释

以下的注释在 **nroff** 和 **troff** 请求中应用。他们以应用处的请求编号作参考。

1. **.L** 字符串寄存器包含所有目录的当前程序语言环境值。
2. **.m** 字符串寄存器包含着 **LC\_MESSAGES** 目录中的语言环境值。
3. **.t** 字符串寄存器包含着 **LC\_TIME** 目录中的语言环境值。
4. **.L**、**.t** 和 **.m** 字符串寄存器提供对一些环境值的访问，同时还可以使用一项更为通用的技术来对其他的环境变量进行访问。例如：如果 **TED** 环境变量被导出，则接下来的 **troff** 命令：

```
.sy echo .ds z $TED >x
.so x
.sy rm x
```

设置 *z* 字符串寄存器中包含 **\$TED** 值。

## 环境变量

**LC\_ALL** 指定供所有语言环境目录使用的语言环境。它将覆盖所有其他的语言环境变量的设置。

**LC\_MESSAGES** 指定 **LC\_MESSAGES** 目录的语言环境值。在 **LC\_ALL** 环境变量未设置的情况下使用。

**LC\_TIME** 指定 **LC\_TIME** 目录的语言环境值。在 **LC\_ALL** 环境变量未设置的情况下使用。

## LANG

指定供所有语言环境目录使用的语言环境。在上面提到的所有的环境变量未设置的情况下使用它。这是在指定语言环境时最常用的环境变量。

## 文件

|                                                 |                                                                    |
|-------------------------------------------------|--------------------------------------------------------------------|
| <code>/usr/share/lib/tmac/tmac.*</code>         | 包含标准宏文件的指针。                                                        |
| <code>/usr/share/lib/macros/*</code>            | 指示标准宏文件。                                                           |
| <code>/usr/share/lib/tmac/tmac.an</code>        | 包含 <b>man</b> 宏软件包的指针。                                             |
| <code>/usr/share/lib/macros/an</code>           | 包含 <b>man</b> 宏软件包。                                                |
| <code>/usr/share/lib/tmac/tmac.e</code> 文件      | 包含 <b>me</b> 宏的定义文件。                                               |
| <code>/usr/share/lib/me</code> 目录               | 包含宏定义文件。                                                           |
| <code>/usr/share/lib/tmac/tmac.m</code>         | 包含 <b>mm</b> 宏软件包的指针。                                              |
| <code>/usr/share/lib/macros/mmn</code>          | 包含 <b>mm</b> 宏软件包。                                                 |
| <code>/usr/share/lib/macros/mmt</code>          | 包含 <b>mm</b> 宏软件包。                                                 |
| <code>/usr/share/lib/tmac/tmac.ptx</code>       | 指向宏软件包。                                                            |
| <code>/usr/share/lib/macros/ptx</code>          | 包含宏软件包。                                                            |
| <code>/usr/share/lib/tmac/tmac.x</code>         | 包含宏定义文件。                                                           |
| <code>/usr/share/lib/ms</code>                  | 包含 <b>ms</b> 宏的定义。                                                 |
| <code>/usr/share/lib/tmac/tmac.v</code>         | 包含宏的定义。                                                            |
| <code>/usr/share/lib/macros/vmca</code>         | 包含宏的定义。                                                            |
| <code>/usr/lib/nls/msg/\$LANG/macros.cat</code> | 包含 <b>mm</b> 、 <b>me</b> 、 <b>ms</b> 和 <b>mv</b> 宏软件包的特定语言环境的消息编目。 |
| <code>/usr/lib/font/dev*/*</code>               | 包含字体宽度表。                                                           |
| <code>/var/tmp/trtmp*</code>                    | 指示临时文件。                                                            |

## 相关信息

**col** 命令、**eqn** 命令、**grap** 命令、**hplj** 命令、**ibm3812** 命令、**ibm3816** 命令、**mm** 命令、**mmt** 命令、**mvt** 命令、**neqn** 命令、**nroff** 命令、**pic** 命令、**ptx** 命令、**refer** 命令、**tbl** 命令、**tc** 命令、**xpreview** 命令。

**nroff** 和 **troff** 输入 文件格式、**troff** 文件格式、**troff** 字体文件格式。

**setlocale** 函数 **strftime** 函数。

《AIX 5L V5.3 本地语言支持指南和参考大全》中的『消息设施』。

《AIX 5L V5.3 本地语言支持指南和参考大全》中的『本地语言支持概述』。

---

## trpt 命令

### 用途

执行 TCP 套接字协议跟踪。

### 语法

```
trpt [ -a ] [ -f ] [ -j ] [ -pAddress ]... [ -s ] [ -t ]
```

### 描述

**trpt** 命令查询传输控制协议 (TCP) 跟踪记录的缓冲区。当套接字被标记以用于调试 **setsockopt** 子例程时创建此缓冲区。然后 **trpt** 命令打印这些跟踪记录的描述。

注：可使用 **traceson** 命令为守护程序打开套接字级调试。

不指定任何选项时，**trpt** 命令打印系统中找到的所有跟踪记录，然后根据 TCP/IP 连接协议控制块（PCB）对它们进行分组。

在可以使用 **trpt** 命令前，必须：

1. 隔离问题并标记调试连接使用的一个或多个套接字。
2. 通过使用 **netstat -aA** 命令来查找与这些套接字相连的协议控制块地址。
3. 然后可以运行 **trpt** 命令，使用 **-p** 标志提供相连的协议控制块地址。您可以使用单个 **trpt** 命令指定多个 **-pAddress** 标志。

一旦定位，**-f** 标志可用于追踪跟踪日志。**-j** 标志可用于检查疑问套接字跟踪记录的存在。

如果系统映像没有合适的符号来查找跟踪缓冲区，**trpt** 命令失败。

## 输出字段

**trpt** 命令输出的信息根据使用的标志的不同而有所变化。以下各种输出类型中包含了字段的定义：

|          |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|----------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 协议控制块标识符 | 标识要跟踪的协议块，如下例所示：<br>4c500c:                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| 时间戳记     | 指定尝试连接的时间，如下例所示：<br>500                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| 连接状态     | 指定与协议控制块的连接状态：<br><b>CLOSED</b><br>连接关闭。<br><b>LISTEN</b><br>侦听连接。<br><b>SYN_SENT</b><br>活动的，已发送 SYN。表示发送连接请求后等待匹配连接请求。<br><b>SYN_RCVD</b><br>已发送和接收 SYN。表示收到接收和发送连接请求后等待确认连接请求通知。<br><b>ESTABLISHED</b><br>连接建立。<br><b>CLOSE_WAIT</b><br>已接收 FIN，等待接收 CLOSE。<br><b>LAST_ACK</b><br>已接收 FIN 和 CLOSE，等待 FIN ACK。<br><b>FIN_WAIT_1</b><br>已关闭，发送 FIN。<br><b>CLOSING</b><br>关闭，交换的 FIN，等候 FIN。<br><b>FIN_WAIT_2</b><br>已关闭，已通知 FIN，等候 FIN。<br><b>TIME_WAIT</b><br>关闭后以 2MSL（两倍于最大段长度）静态等待。 |

## Action

指定分包跟踪连接的当前状态。命令输出随操作不同而变化。

**Input** 接收输入包。输出语法是:

```
input (SourceAddress, Port, DestinationAddress,  
Port) <Sequence Number of the First Data Octet> @  
AcknowledgementNumber
```

如下例所示:

```
input (src=129.353173176,23, dst=129.35.17.140, 1795) fb9f5461@fb9e4c68
```

**Output** 传输包。输出语法是:

```
output (SourceAddress, Port, DestinationAddress,  
Port) <Sequence Number Of The First Data Octet>..  
<Sequence Number of the Last Data Octet>@  
AcknowledgementNumber)
```

如下例所示:

```
output (src=129.35.17.140,1795, dst=129.35.17.176, 23) fb9e4c68@fb9f5462
```

## Window Size

指定发送或接收包的窗口大小, 如下例所示:

```
(win=1000)
```

**User** 指定用户请求。如下是用户请求示例:

SLOWTIMO<KEEP>

用户请求类型及其定义如下:

**PRU\_ATTACH** 连接协议使之启动。

**PRU\_DETACH** 从启动中拆离协议。

**PRU\_BIND** 为地址绑定套接字。

**PRU\_LISTEN** 侦听连接。

**PRU\_CONNECT**  
建立同级设备连接。

**PRU\_ACCEPT** 接受与同级设备连接。

**PRU\_DISCONNECT**  
与同级设备断开连接。

**PRU\_SHUTDOWN**  
不发送任何数据。

**PRU\_RCVD** 已接收数据, 有更多空间。

**PRU\_SEND** 发送此数据。

**PRU\_ABORT** 异常终止 (快速 DISCONNECT, DETACH)。

**PRU\_CONTROL**  
控制协议操作。

**PRU\_SENSE** 返回到 m 状态。

**PRU\_RCVOOB** 检索区域外数据。

**PRU\_SENDOOB**  
发送区域外数据。

**PRU\_SOCKADDR**  
取得套接字地址。

**PRU\_PEERADDR**  
取得同级设备地址。

**PRU\_CONNECT2**  
连接两个套接字。

**PRU\_FASTTIMO**  
200 毫秒超时。

**PRU\_SLOTIMO** 500 毫秒超时。

**PRU\_PROTORCV**  
从下接收。

**PRU\_PROTOSEND**  
往下发送。

**Drop** 指定数据处于前段中, 删除数据。

**Window and Sequence** 窗口和序列变量类型如下:

### Variables

*rcv\_nxt* 进入段上要使用的下一序列号。

*rcv\_wnd*  
接收窗口大小。

*snd\_una* 最早未通知的序列号。

*snd\_nxt* 下一要发送的序列号。

*snd\_max*  
发送的最高序列号。

*snd\_sl1* 窗口更新段序列号。

*snd\_wl1* 窗口更新段 ack 号。

*snd\_wnd*  
发送窗口。

## 标志

- a** 除正常输出之处, 为每个记录包打印源地址和目的地址值。
- f** 跟随出现时追踪, 短暂等待每次日志结束时的附加记录。
- j** 仅列出跟踪记录存在的协议控制块地址。
- pAddress** 仅显示与协议控制块有关的跟踪记录, 该协议控制块由 *Address* 变量指定为十六进制。必须重复 **-p** 标志和每个指定的 *Address* 变量。
- s** 除正常输出外, 打印包序列信息的详细信息描述。
- t** 除正常输出外, 打印每个跟踪点的所有计时器值。

## 示例

1. 为每个记录包打印跟踪信息及源地址和目的地址, 输入:

```
$ trpt -a
```

可能显示如下输出:

```
124b0c:
900 ESTABLISHED:input (src=192.9.201.3,4257, dst=192.9.201.2,102
5)2326e6e5@ad938c02(win=200)<ACK,FIN,PUSH> -> CLOSE_WAIT
900 CLOSE_WAIT:output (src=192.9.201.2,1025, dst=192.9.201.3,425
7)ad938c02@2326e6e6(win=4000)<ACK> -> CLOSE_WAIT
900 LAST_ACK:output (src=192.9.201.2,1025, dst=192.9.201.3,4257)
ad938c02@2326e6e6(win=4000)<ACK,FIN> -> LAST_ACK
900 CLOSE_WAIT:user DISCONNECT -> LAST_ACK
900 LAST_ACK:user DETACH -> LAST_ACK 12500c:
800 ESTABLISHED:output (src=192.9.201.2,1024, dst=192.9.201.3,51
2)ad8eaa13@2326e6e5(win=4000)<ACK> -> ESTABLISHED
800 ESTABLISHED:input (src=192.9.201.3,512, \
dst=192.9.201.2,1024)
[2326e6e5..2326e727]@ad8eaa13(win=1ef)<ACK,PUSH> -> ESTABLISHED
800 ESTABLISHED:user RCVD -> ESTABLISHED
900 ESTABLISHED:output (src=192.9.201.2,1024, dst=192.9.201.3,51
2)ad8eaa13@2326e727(win=4000)<ACK> -> ESTABLISHED
900 ESTABLISHED:input (src=192.9.201.3,512, \
dst=192.9.201.2,1024)
[2326e727..2326e82f]@ad8eaa13(win=1ef)<ACK,PUSH> -> ESTABLISHED
900 ESTABLISHED:user RCVD -> ESTABLISHED
900 ESTABLISHED:output (src=192.9.201.2,1024, dst=192.9.201.3,51
2)ad8eaa13@2326e82f(win=4000)<ACK> -> ESTABLISHED
```

```

900 ESTABLISHED:input (src=192.9.201.3,512, \
dst=192.9.201.2,1024)
2326e82f@ad8eaa13(win=1ef)<ACK,FIN,PUSH> -> CLOSE_WAIT
900 CLOSE_WAIT:output (src=192.9.201.2,1024, \
dst=192.9.201.3,512)
ad8eaa13@2326e830(win=4000)<ACK> -> CLOSE_WAIT
900 LAST_ACK:output (src=192.9.201.2,1024, dst=192.9.201.3,512)a
d8eaa13@2326e830(win=4000)<ACK,FIN> -> LAST_ACK
900 CLOSE_WAIT:user DISCONNECT -> LAST_ACK
900 LAST_ACK:user DETACH -> LAST_ACK
$ _

```

- 列出有跟踪记录的协议控制块，输入：

```
trpt -j
```

可能显示如下输出：

```
124b0c, 12500c
```

- 打印与单协议控制块关联的跟踪记录，输入：

```
trpt -p 12500c
```

可能显示如下输出：

```

800 ESTABLISHED:output ad8eaa13@2326e6e5(win=4000)<ACK> ->
ESTABLISHED
800 ESTABLISHED:input [2326e6e5..2326e727]@ad8eaa13(win=1ef)
<ACK,PUSH> -> ESTABLISHED
800 ESTABLISHED:user RCVD -> ESTABLISHED
900 ESTABLISHED:output ad8eaa13@2326e727(win=4000)<ACK> -> ESTABLISHED
900 ESTABLISHED:input [2326e727..2326e82f]@ad8eaa13(win=1ef) <ACK,PUSH> -> ESTABLISHED
900 ESTABLISHED:user RCVD -> ESTABLISHED
900 ESTABLISHED:output ad8eaa13@2326e82f(win=4000)<ACK> -> ESTABLISHED
900 ESTABLISHED:input 2326e82f@ad8eaa13(win=1ef)<ACK,FIN,PUSH> -> CLOSE_WAIT
900 CLOSE_WAIT:output ad8eaa13@2326e830(win=4000)<ACK> -> CLOSE_WAIT
900 LAST_ACK:output ad8eaa13@2326e830(win=4000)<ACK,FIN> -> LAST_ACK
900 CLOSE_WAIT:user DISCONNECT -> LAST_ACK
900 LAST_ACK:user DETACH -> LAST_ACK
$ _

```

## 相关信息

**netstat** 命令、**tracesoff** 命令和 **traceson** 命令。

**setsockopt** 子例程。

《网络与通信管理》中的『传输控制协议/因特网协议』、『TCP/IP 协议』和『TCP/IP 路由』。

---

## true 或 false 命令

### 用途

返回零（true）或非零（false）的退出值。

### 语法

**true**

**false**



## 描述

**true** 命令返回零退出值。**false** 命令返回非零退出值。这些命令常用作 shell 脚本的一部分。

## 示例

要构造每秒显示一次日期和时间的循环，请在 shell 脚本中使用以下代码：

```
while true
do
    date
    sleep 60
done
```

## 相关信息

《操作系统与设备管理》中的『创建和运行 shell 脚本』提供了有关创建和执行 shell 过程的信息。

《操作系统与设备管理》中的『命令概述』。

---

## truss 命令

### 用途

跟踪进程的系统调用、动态装入的用户级函数调用、接收的信号和造成的机器故障。

### 语法

```
truss [ -f ] [ -c ] [ -a ] [ -l ] [ -d ] [ -D ] [ -e ] [ -i ] [ { -t | -x } [!] Syscall [...] ] [ -s [!] Signal [...] ] [ { -m } [!] Fault [...] ] [ { -r | -w } [!] FileDescriptor [...] ] [ { -u } [!] LibraryName [...] :: [!] FunctionName [ ... ] ] [ -o Outfile ] { Command } -p pid [ . . . ]
```

### 描述

**truss** 命令执行指定命令或附加在列出进程标识上，并产生对系统调用、接收的信号和进程造成的机器故障的跟踪。每行跟踪输出报告 *Fault* 或 *Signal* 名称或 *Syscall* 名称和参数及返回值。系统库定义的子例程对于内核的严格系统调用并不是必要的。**truss** 命令不报告这些子例程，而是报告子例程的基本系统调用。可能的话，系统调用参数使用有关系统头文件定义符号显示。对于路径名指针参数，**truss** 显示指向的字符串。未定义的系统调用缺省显示系统名称、所有八个可能的变量及十六进制格式的返回值。

当 **-o** 标志与 **truss** 一起使用，或标准错误改指向非终端文件时，**truss** 忽略挂断、中断和信号进程。它促进跟踪交互式程序从终端获得 **interrupt** 和 **quit** 信号。

若跟踪输出保持指向终端或跟踪现存过程（使用 **-p** 标志），**truss** 对 **hangup**、**interrupt** 和 **quit** 信号做出反应，释放所有跟踪进程并退出。它使用户可终止过多的跟踪输出并释放前现存进程。释放进程继续到功能恢复正常。

对与含有一列变量的选项，名称 **all** 可用作指定列表中所有可能成员的简写。以 **!** 开头的列表，选项含义是相反的（示例：排除而非跟踪）。同一选项可能指定为多种表现形式。同一列表的相同名称，后面的选项（右边选项）覆盖前面的选项（左边选项）。

除了页面故障以外的所有机器故障均导致公布信号到发生故障进程。除了进程阻塞信号，每一机器故障报告后立即给出接收的信号报告。

为避免与其他控制进程冲突，**truss** 不跟踪由其检测到并由带有 **/proc** 接口的另一进程控制的进程。

多进程跟踪输出不是按严格的时间顺序产生。示例：管道读取可以在相应的写之前报告。然而每一进程输出是严格按时间排序。跟踪输出含有制表符，且每八个位置设置标准制表符停止位。

系统跟踪子进程时可运行于每个用户进程槽之外。这是因为跟踪多个进程时，**truss** 作为所跟踪进程的控制进程运行，将用于所给进程的进程槽数目加倍。运行多进程跟踪之前应考虑通常系统强制的每个用户 25 进程的限制。

操作系统为进程跟踪强加某种安全性限制。必须有跟踪命令的存取特权。**set-uid** 和 **set-gid** 进程仅可由特权用户跟踪。除由特权用户运行外，**truss** 命令无法控制执行 **set-id** 或不可读对象代码的进程。这些未跟踪的进程从执行点保持正常并与 **truss** 无关。

**truss** 输出提到的轻量级进程 (LWP) 是真正的内核线程。选项 **-l** 在每行跟踪输出显示 LWP 标识 (即线程标识)。

AIX 库中的用户库函数有静态和动态两种装入函数调用。选项 **-u** 跟踪仅用于动态装入函数调用。

**-u** 选项提供动态装入函数调用的用户级函数调用跟踪。此选项产生函数调用的进入/退出跟踪。

## 标志

|                    |                                                                                                                                                                 |
|--------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-a</b>          | 显示在每次执行系统调用时传递的参数字符串。                                                                                                                                           |
| <b>-c</b>          | 计数跟踪系统调用、故障和信号而不是逐行显示跟踪结果。跟踪命令终止或 <b>truss</b> 中断时生成摘要报告。若还使用 <b>-f</b> 标志，计数包含所有跟踪的系统调用、故障和子进程信号。                                                              |
| <b>-d</b>          | 每行输出包含时间戳记。时间从跟踪开始以每秒显示。跟踪输出的第一行显示测量单个时间戳记的基本时间。缺省不显示时间戳记。                                                                                                      |
| <b>-D</b>          | 每行输出显示增量时间。增量时间表示从由该线程引起的最后报告事件起计时引起事件的 LWP 的逝去时间。缺省不显示增量时间。                                                                                                    |
| <b>-e</b>          | 显示在每次执行系统调用时传递的环境字符串。                                                                                                                                           |
| <b>-f</b>          | 跟在 <b>fork</b> 系统调用产生的所有子进程之后，并包含跟踪输出中的信号、故障和系统调用。通常，仅跟踪第一级命令或进程。如果指定 <b>-f</b> 标志，进程标识与每行跟踪输出一一起显示哪个进程执行系统调用或接收信号。                                             |
| <b>-i</b>          | 阻止显示可中断的休眠系统调用。某些终端设备或管道的系统调用，如 <b>open</b> 和 <b>kread</b> ，可无限期休眠且是可中断的。通常，如果这种休眠系统调用保持超过一秒休眠， <b>truss</b> 将报告。系统调用完成后一秒内报告系统调用。 <b>-i</b> 标志使这种系统调用完成后仅报告一次。 |
| <b>-l</b>          | 显示有关 LWP 进程的标识 (线程标识) 及 <b>truss</b> 输出。输出中缺省不显示 LWP 标识。                                                                                                        |
| <b>-m</b> [!]Fault | 跟踪进程中的机器故障。必须用逗号分隔要跟踪的机器故障。可按名称或代码指定故障 (参见 <b>sys/procfs.h</b> 头文件)。如果列表以符号 “!” 开始，指定的故障不再被跟踪且不显示跟踪输出。缺省值是 <b>-mall -m!fltpage</b> 。                            |
| <b>-o</b> Outfile  | 指定用于跟踪输出的文件。缺省时输出指向标准错误。                                                                                                                                        |
| <b>-p</b>          | 将参数作为一系列现存进程的进程标识而不是要执行的命令解释到 <b>truss</b> 。倘若进程用户标识或组标识与用户的用户标识或组标识匹配或者用户是特权用户， <b>truss</b> 控制并开始跟踪每个进程。                                                      |

**-r** [!] *FileDescriptor*

显示每一个读取指定文件描述符的 I/O 缓冲区全内容。输出采用每行 32 字节的格式并以 ASCII 字符（跟在空格后）形式显示字节，或以控制字符的双字符 C 语言转义序列形式显示字节，如水平制表符 (\t) 或换行符 (\n)。如果不能解释成 ASCII 格式，字节以双字符十六进制形式显示。即使没有 **-r** 标志，对于每次被跟踪的读操作，仍会显示 I/O 缓冲区的前 12 个字节。缺省值是 **-r!all**。

**-s** [!] *Signal*

允许列出要跟踪或除去的 *Signals*。跟踪列表中指定的信号（由逗号分隔）。信号被忽略但不是被进程阻塞时，跟踪输出仍报告每个指定信号的接收。直到进程释放时才可接收阻塞信号。可按名称或代码指定信号（参见 `sys/signal.h`）。如果该列表以“!”符号开头，则列出的信号将从显示跟踪输出中排除。缺省值是 **-s all**。

**-t** [!] *Syscall*

包含或排除来自跟踪进程的系统调用。要跟踪的系统调用必须在列表中指定并以逗号分隔。如果列表以符号“!”开始，指定的系统调用将被从跟踪输出中排除。缺省值是 **-tall**。

**-u** [!] [*LibraryName* [...]:[!]*FunctionName* [...] ]

跟踪来自用户库动态装入的用户级函数调用。*LibraryName* 是逗号分隔的库名列表。*FunctionName* 是逗号分隔的函数名列表。两种情况下的名称都包含名称匹配元字符 \*、? 和 [], 这些字符与由 shell 解释的意义相同，但适用于库/函数名称空间而非文件。

**-w** [!] *FileDescriptor*

任一列表中以 ! 开头的列指定不跟踪库或函数的排除名称列表。除去库将除去该库中所用函数。忽略库除去列表后的任何函数列表。可指定多个 **-u** 选项并将它们从左到右排列。缺省不跟踪库/函数调用。

**-x** [!] *Syscall*

显示列出文件描述符每个写过程的 I/O 缓冲区内容（参见 **-r**）。缺省值是 **-w!all**。  
以原始格式显示跟踪系统调用指定参数的数据，通常按十六进制而不是符号化方式。缺省值是 **-x!all**。

## 示例

1. 在终端上跟踪 **find** 命令，输入：

```
truss find . -print >find.out
```

2. 跟踪 **lseek**、**close**、**statx** 和开放系统调用，输入：

```
truss -t lseek,close,statx,open find . -print > find.out
```

3. 显示 **find** 命令的线程标识和常规输出，输入：

```
truss -l find . -print >find.out
```

4. 显示 **find** 命令的时间戳记和常规输出，输入：

```
truss -d find . -print >find.out
```

5. 显示 **find** 命令的增量时间和常规输出，输入：

```
truss -D find . -print >find.out
```

6. 运行 **ls** 命令时，在 **libc.a** 库中跟踪 **malloc()** 函数调用及除去 **strlen()** 函数调用，输入：

```
truss -u libc.a::malloc,!strlen ls
```

7. 运行 **ls** 命令时，跟踪 **libc.a** 库中名称以“m”开头的的所有函数调用，输入：

```
truss -u libc.a::m*,!strlen ls
```

8. 运行可执行 **foo** 时，跟踪来自 **libcurses.a** 库的所有函数调用及除去来自 **libc.a** 的调用，输入：

```
truss -u libcurses.a,!libc.a:* foo
```

9. 运行可执行 `foo` 时，跟踪来自 `libcurses.a` 的 `refresh()` 函数调用和来自 `libc.a` 的 `malloc()` 函数调用，输入：

```
truss -u libc.a::malloc -u libcurses.a::refresh foo
```

## 文件

`/proc` 文件系统

## 相关信息

`fork` 子例程。

`/proc` 文件。

---

## tset 命令

### 用途

初始化终端。

### 语法

```
tset [ -e C ] [ -k C ] [ -i C ] [ - ] [ -s ] [ -l ] [ -Q ] [ -m [ Identifier ] [ TestBaudRate ] :Type ] ... [ Type ]
```

### 描述

**tset** 命令可设置终端特征。它执行终端依赖性的处理，比如：设置擦除和杀死字符、设置或复位延迟以及发送任何需要的序列以正常初始化终端。

**tset** 命令首先确定所涉及的终端类型（由 *Type* 参数指定）。然后执行必要的初始化和方式设置。附加在每个端口上的终端类型在对象数据管理器（ODM）数据库内指定。`terminfo` 数据库包含可能的终端类型名称。如果端口与指定终端没有永久相连（即非硬连线），**tset** 命令分配给端口合适的一般标识，如 `dialup`。

未指定标志时，**tset** 命令从 `TERM` 环境变量中读取终端类型并重新初始化终端。

当 **tset** 命令用于启动脚本（`.profile` 文件用于 `sh` 用户或 `.login` 文件用于 `cs`h 用户）时，脚本应包括通常使用在未硬连线的端口上终端的类型的信息。这些端口在 ODM 数据库中指定为 `dialup`、`plugboard` 或 `ARPANET`。指定通常哪种终端类型可用在这些端口上，使用 **-m** 标志（其后跟随适当的端口类型标识），一种可选的波特率规范和终端类型。如果指定一个以上映射，则优先使用第一个适用的映射。缺失的端口类型标识匹配所有标识。任何 `terminfo` 数据库给出的备用一般名称都可用作标识。

可在 **tset** 命令中指定波特率，如同 `stty` 命令一样。波特率与诊断输出速度（可能为控制终端）是可以比较的。波特率测试可以是下列字符的任一组合：

- .（句点）
- @（at 符号）
- <（小于符号）
- !（感叹号）

@ (at 符号) 代表前置词 at, 而 ! (感叹号) 反转测试意义。为避免元字符造成的问题, 将 **-m** 标志参数放在 '' (单引号) 里。 **cs**h 命令用户必须将 \ (反斜杠) 放在任一 ! (感叹号) 之前。

下列示例设置终端类型为 adm3a, 如果使用中的端口以大于 300 波特的速度拨号。设置终端类型到 dw2, 如果端口是等于或小于 300 波特速度的拨号端口:

```
tset -m 'dialup>300:adm3a' -m dialup:dw2 -m 'plugboard:?adm3a'
```

如果 *Type* 参数以 ? (问号) 开头, 提示验证类型。使用指定类型, 按 Enter 键。使用不同类型, 输入需要的类型。所给示例中, 提示验证 adm3 插板端口类型。

如果无映射应用且命令行给出最终类型选项 (没有以 **-m** 标志开头), 使用该类型。另外, 缺省终端类型是 ODM 数据库中标识的类型。硬连线端口都应在 ODM 数据库中标识。

已知终端类型时, **tset** 命令参与终端驱动方式设置。通常包括设置:

- 终端初始化序列
- 单字符擦除和可选的行杀死 (整行擦除) 字符
- 特殊字符延迟

终端初始化序列传输过程中会关闭制表符和换行扩展。

在可退格但不能叠印的终端上 (如 CRT), 且当擦除字符是缺省擦除字符 (标准系统上的 #) 时, 擦除字符替换为退格符 (Ctrl-H)。

## 标志

|                                              |                                                                                                              |
|----------------------------------------------|--------------------------------------------------------------------------------------------------------------|
| <b>-e</b> <i>C</i>                           | 将擦除字符设置为由 <i>C</i> 参数指定的字符。缺省值为退格字符。                                                                         |
| <b>-l</b>                                    | 禁止传输终端初始化字符串。                                                                                                |
| <b>-i</b> <i>C</i>                           | 将中断字符设置为由 <i>C</i> 参数指定的字符。 <i>C</i> 缺省为 ^C (插入符号 C)。该 ^ (插入符号) 字符也可用于此选项。                                   |
| <b>-k</b> <i>C</i>                           | 将行杀死字符设置为由 <i>C</i> 参数指定的字符。 <i>C</i> 参数缺省为 ^X (插入符号 X)。该 ^ (插入符号) 字符也可用于此选项。                                |
| <b>-m</b> <i>IdentifierTestbaudRate:Type</i> | 指定哪种终端类型 (在 <i>Type</i> 参数中) 通常可用在 <i>Identifier</i> 参数中标识的端口上。缺失标识匹配所有标识。可选择在 <i>TestBaudRate</i> 参数里指定波特率。 |
| <b>-Q</b>                                    | 禁止打印 Erase set to 和 Kill set to 消息。                                                                          |
| <b>-s</b>                                    | 打印序列 <b>cs</b> h 命令, 该命令基于决定终端的名称初始化 <b>TERM</b> 环境变量。                                                       |
| <b>-</b>                                     | 决定终端的名称按标准输出形式输出。这是 <b>TERM</b> 环境变量。                                                                        |

## 示例

以下示例皆假定 Bourne shell 和使用 - 标志。如果使用 **cs**h 命令, 则请使用上述变量。**.profile** 或 **.login** 文件中 **tset** 命令的典型用法包括 **-e** 和 **-k** 标志, 通常也包括 **-n** 或 **-Q** 标志。为简化示例, 此处不包含这些标志。

**注:** 不管示例中使用多少行都要确保每行输入全部 **tset** 命令。

1. 现在您是 2621 终端。除非您一直是 2621 终端, 否则不要在您的 **.profile** 文件中使用下列示例。

```
export TERM; TERM='\tset \- 2621\'
```

2. 您在家中拨号使用的是 h19 终端, 但办公室终端是硬连接且在 ODM 数据库中指定。

```
export TERM; TERM='\tset \- \-m dialup:h19''
```

3. 您有连接所有设备的转换开关并使其不能为您进入的端口加密。您可在办公室以 9600 波特使用 vt100 而以 1200 波特在 2621 上从家中拨号到转换端口。有时您在工作中使用不同的终端。高速下，您想验证您的终端类型，但在 1200 波特下，您始终在 2621 上。注意引号怎样保护由 shell 解释的大于符号和问号。

```
export TERM; TERM=\tset \- \-m 'switch>1200:?vt100' \-m
'switch<=1200:2621'
```

如果没有任何条件，使用 ODM 数据库中指定的终端类型。

4. 如果您总是以同波特率拨号到不同终端，可使用下列记录。您的最公共终端是 adm3a。总是提示验证终端类型，其缺省值为 adm3a。

```
export TERM; TERM=\tset \- \?adm3a\'
```

5. 如果没有正确安装 ODM 数据库而您想对整个波特率加密，输入：

```
export TERM; TERM=\tset \- \-m 'switch>1200:?vt100' \-m
'switch<=1200:2621'
```

6. 您在 Concept100 上以等于或小于 1200 波特拨号，有时通过转换端口，有时通过常规拨号。使用不同的终端以超过 1200 的速度通过转换端口，通常是办公室里的终端即 vt100。然而，有时您从大学通过 ARPANET 登录；这时您在仿真 dm2500 的 ALTO 上。经常登录不同的硬连接端口，如控制台，这些端口都正确输入到 ODM 数据库中。要设置擦除字符为 Ctrl-H、杀死字符为 Ctrl-U，输入：

```
export TERM
TERM=\tset \-e \-k(hat)U \-Q \- "-m 'switch<1200:concept100'
"-m 'switch:?vt100' \-m dialup:concept100 "1-m arpanet: dm2500"
```

这也可阻止 tset 命令打印下列行：

```
Erase set to Backspace, Kill set to Ctrl-U
```

7. 要设置擦除字符为一控制字符，输入：

```
tset -e ^Y
```

## 文件

`/usr/share/lib/terminfo`

包含终端能力数据库。

## 相关信息

`csch` 命令、`reset` 命令、`sh` 命令和 `stty` 命令。

`environ` 文件和 `terminfo` 文件。

《操作系统与设备管理》中『TTY 终端设备』的“不同显示器和终端的 TERM 值”部分。

---

## tsh 命令

### 用途

调用可信的 shell。

### 语法

依顺序按下：Ctrl+X、Ctrl+R 键。

`tsh` 命令

## 描述

**tsh** 命令是命令解释器，它提供优于 **Kone shell**（标准登录 shell）的安全性。通常用户在登录后按下 **Ctrl+X**、**Ctrl+R**、安全注意键（SAK）顺序来调用 **tsh shell**。**tsh shell** 还可通过在 **/etc/passwd** 文件中将其定义为登录 shell 来调用。

要使用 SAK 顺序来调用可信的 shell，用户使用的终端必须启用 SAK，而且必须允许该用户使用可信路径。关于在终端上启用 SAK 的信息，请参阅《操作系统与设备管理》中的『可信计算库』，关于允许用户访问可信路径的信息，请参阅 **/etc/security/user** 文件和 **chuser** 命令。

要从 **tsh shell** 退出，使用以下任一命令：**logout** 命令、**shell** 命令和 **su** 命令。**logout** 命令结束登录会话，而其他命令执行用户的初始化程序并继续登录会话。

可信的 shell 与 Korn shell 不同之处在于：

- 不支持函数和别名定义。仅在 **/etc/tsh\_profile** 文件中支持别名定义。
- 不能重新定义 **IFS** 和 **PATH** 环境变量。
- 只有可信的程序可从 **tsh shell** 运行。
- 不支持历史记录机制。
- 唯一使用的概要文件是 **/etc/tsh\_profile** 文件。
- 可信的 shell 具有以下内置命令：

|               |                                  |
|---------------|----------------------------------|
| <b>logout</b> | 退出登录会话并终止所有进程。                   |
| <b>shell</b>  | 重新初始化用户的登录会话。与登录系统的效果是一样的。       |
| <b>su</b>     | 将有效标识复位为系统上的用户标识并执行另一个可信的 shell。 |

## 安全性

访问控制：此命令应是标准用户程序并有 **trusted computingbase** 属性。

访问的文件：

| 方式 | 文件                      |
|----|-------------------------|
| r  | <b>/etc/tsh_profile</b> |

## 示例

要调用可信的 shell，按 **Ctrl+X**、**Ctrl+R** 按键顺序、安全注意键（SAK）。

## 文件

|                                |                     |
|--------------------------------|---------------------|
| <b>/usr/bin/tsh</b>            | 包含 <b>tsh</b> 命令。   |
| <b>/etc/tsh_profile</b>        | 包含可信的 shell 的初始化命令。 |
| <b>/etc/passwd</b>             | 包含基本用户属性。           |
| <b>/etc/security/user</b>      | 包含用户的扩展属性。          |
| <b>/etc/security/login.cfg</b> | 包含配置信息。             |

## 相关信息

**chuser** 命令、**init** 命令、**ksh** 命令、**logout** 命令、**shell** 命令、**su** 命令和 **tsm** 命令。

有关在安装过程中使用的“单源双对象”（SSDO）命令的更多信息，请参阅《AIX 5L V5.3 本地语言支持指南和参考大全》中的『本地语言支持概述』。

关于用户的标识和认证、自主访问控制、可信计算库和审计的更多信息，请参阅《安全性》中的 Securing the network。

关于可信路径和在终端上启用 SAK 的更多信息，请参阅《安全性》中的 Trusted Computing Base。

---

## tsm 命令

### 用途

提供终端状态管理。

### 语法

**tsm** *Port*

### 描述

**tsm** 命令调用终端状态管理器，该管理器控制在可信路径中使用的端口。功能为：

- 建立线路通信方式和规程 - 由 **getty** 命令执行的功能。
- 验证用户的帐户和标识，并设置初始进程凭证和环境 - 由 **login** 命令执行的功能。
- 如果对端口启用了安全注意键（SAK）且使用了系统登录程序，执行可信路径管理。

**注：** **tsm** 命令不在命令行中输入。

可信路径管理发生在两个阶段：

**login** 如果用户未成功登录，则此阶段有效。如果检测到安全注意键（SAK）信号，系统重新启动 **getty-login** 类型处理。如果端口和用户支持可信的状态，下一次登录将用户置入可信的状态。

**shell** 此阶段在用户认证成功之后发生。此命令根据用户的 **tpath** 属性发挥作用。以下值是有效的：

**on** 提供标准可信路径管理。检测到安全注意键（SAK）信号后，下一次试图访问端口时将终止除 **tsm** 进程和它的兄弟进程（包含可信的 **shell**）之外的所有访问该端口的进程。端口复位为其初始状态并标记为可信的，同时可信的 **shell** 命令（**tsh** 命令）将会执行。

**notsh** 当检测到安全注意键（SAK）信号时，用户会话终止。

**always** 不允许用户离开可信路径。用户的 **shell** 始终为可信的 **shell**，**tsh**。

**nosak** 对于终端禁用安全注意键（SAK），且运行用户的初始程序。

### 安全性

访问控制：此命令应将执行（x）权限授权给任何用户。该命令应对 **root** 用户设置用户标识并具有可信计算库属性。

访问的文件：

| 方式       | 文件                             |
|----------|--------------------------------|
| <b>r</b> | <b>/etc/objrepos/CuAt</b>      |
| <b>r</b> | <b>/usr/lib/objrepos/PdAt</b>  |
| <b>r</b> | <b>/etc/security/login.cfg</b> |



访问的文件:

|    |                    |
|----|--------------------|
| 方式 | 文件                 |
| r  | /etc/security/user |

## 示例

要对 `tty0` 提供终端状态管理, 添加以下行到 `/etc/inittab` 文件:

```
tty0:2:respawn:/usr/sbin/tsm /dev/tty0
```

此命令初始化端口 `/dev/tty0` 并设置该端口的特征。

## 文件

`/usr/sbin/tsm`

包含 `tsm` 命令。

`/etc/security/login.cfg`

包含配置信息。

`/etc/security/user`

包含扩展的用户属性。

## 相关信息

`getty` 命令、`init` 命令、`login` 命令、`logout` 命令、`setgroups` 命令、`shell` 命令、`su` 命令和 `tsh` 命令。

关于用户的标识和认证、自主访问控制、可信计算库和审计的更多信息, 请参阅《安全性》中的 `Securing the network`。

---

## tsort 命令

### 用途

为有序对的无序列列表排序 (一种拓扑排序)。

### 语法

```
tsort [ - ] [ File ]
```

### 描述

`tsort` 命令从 `File` 或标准输入中读取有序对的无序列列表, 建立完全有序列表并将其写成标准输出。

输入 `File` 应包含成对的由空格分隔的非空字符串。不同项的对表示相对顺序。相同项的对表示存在但没有相对顺序。您可以使用 `tsort` 命令为 `lorder` 命令的输出排序。

如果 `File` 包含奇数个字段, 则显示相应的错误消息。

### 标志

`-` (双连字符) 将 `-` 标志后的全部变量解释为文件名。如果文件以 `-` 命名, 使用 `tsort - -`。

## 退出状态

该命令返回以下退出值:

0 成功完成。  
>0 发生错误。

## 文件

`/usr/ccs/bin/tsort` 包含 `tsort` 命令。  
`/usr/ccs/bin/tsort` 包含到 `tsort` 命令的符号链接。

## 相关信息

`ar` 命令、`ld` 命令、`lorder` 命令和 `xargs` 命令。

《操作系统与设备管理》中的『命令概述』。

---

## ttt 命令

### 用途

启动 tic-tac-toe 游戏。

### 语法

`ttt [ -e ] [ i ]`

### 描述

`ttt` 命令启动 tic-tac-toe 游戏。这是学习版本但它学起来较慢。它在完全掌握游戏之前会输掉近 80 局游戏。当您启动游戏时会得到提示 Accumulated knowledge? (Yes or No)。输入 y 为计算机提供以前游戏所获得的知识。

您始终是 X，而您的对手始终是 O。您可以移动第一步或由您的对手先行。要让对手先行，在游戏开始受到提示 Your move? 时按 Enter 键。首先在一行中达成 3 个相同标记的一方赢得游戏。例如:

```
new game
123
456
789
Your move?
1
X03
456
789
Your move?
9
X00
456
78X
Your move?
5
You win
```

示例中，您的第一步是将 X 放在 1 所在的位置。计算机将 O 放在 2 所在的位置。游戏继续进行直到您在对角线（1，5，9）上得到三个 X 为止。游戏重复进行直到您退出。要退出游戏，按 Interrupt（Ctrl-C）或 End Of File（Ctrl-D）按键顺序。

## 标志

- e** 加快学习的速度。
- i** 在游戏开始前显示指示信息。

## 文件

- \$HOME/ttt.a** 指定学习文件的位置。
- /usr/games** 指定系统游戏的位置。

## 相关信息

**arithmetic** 命令、**back** 命令、**bj** 命令、**craps** 命令、**fish** 命令、**fortune** 命令、**hangman** 命令、**moo** 命令、**number** 命令、**quiz** 命令、**turnoff** 命令、**turnon** 命令和 **wump** 命令。

---

## tty 命令

### 用途

将终端的全路径名写到标准输出。

### 语法

**/usr/bin/tty** [ **-s** ]

### 描述

**tty** 命令将终端的名称写到标准输出。

如果标准输入不是终端且没有指定 **-s** 标志，将得到消息 Standard input is not a tty.

以下环境变量影响 **tty** 命令执行：

- LANG** **LC\_ALL** 变量和以 **LC\_** 开头的相应环境变量都未指定语言环境时，确定用于语言环境类别的语言环境。
- LC\_ALL** 确定要使用的语言环境。此变量覆盖由以 **LC\_** 开头的任何其他环境变量或 **LANG** 变量指定的语言环境类别的任意值。
- LC\_CTYPE** 确定将文本数据的序列字节序列解释成字符的语言环境。例如，这种变量可以指定多字节字符而不是单字节字符。
- LC\_MESSAGES** 确定消息的语言。

## 标志

- s** 禁止报告路径名。

## 退出状态

该命令返回以下退出值：

- 0 标准输入是终端。
- 1 标准输入不是终端。
- >1 发生错误。

## 示例

1. 要显示全路径名：

```
tty
```

2. 要测试标准输入是否是终端：

```
if tty -s
then
echo 'Enter the text to print:' >/dev/tty
qprt -
fi
```

如果标准输入是终端，则显示消息“Enter the text to print:”作为提示并打印用户输入的文本。如果标准输入不是终端，则无显示；仅打印从标准输入读取的文本。

即使重定向 shell 过程的标准输出，回显 `... >/dev/tty` 仍在屏幕上显示提示。这样形式的提示永远不会写入输出文件。特殊文件 `/dev/tty` 始终是指您的终端，尽管它还有其他名称比如 `/dev/console` 或 `/dev/tty2`。

## 文件

`/usr/bin/tty` 包含 `tty` 命令。  
`/dev/tty` 指定 `tty` 伪设备。

## 相关信息

《AIX 5L V5.3 本地语言支持指南和参考大全》中的『本地语言支持概述』讨论 `LC_` 变量。

---

## tunchange 命令

### 用途

更新文件中的一个或多个可调节。

### 语法

```
tunchange -f Filename ( -t Stanza ( {-o Parameter[=Value]} | -D ) | -m Filename2 )
```

### 描述

`tunchange` 命令无条件地更新可调文件。它还能够将另一个文件与当前文件合并起来。

注：没有消息显示（即使当更改了 `bosboot` 类型的参数时也是如此）。

## 标志

|                                  |                                                                                                                                                               |
|----------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-f</b> <i>Filename</i>        | 更新的可调文件的名称。如果该名称不包括 ‘/’（正斜杠），则认为它与 <b>/etc/tunables</b> 相关。                                                                                                   |
| <b>-t</b> <i>Stanza</i>          | 要更新的节名称。 <i>Stanza</i> 为 <b>schedo</b> 、 <b>vmo</b> 、 <b>ioo</b> 、 <b>no</b> 、 <b>nfso</b> 或 <b>raso</b> 。 <i>Stanza</i> 与可以更新由 <b>-o</b> 标志指定一个或多个参数的命令名相对应。 |
| <b>-o</b> <i>Parameter=Value</i> | 将要设置为 <i>Value</i> 的参数。它必须在由 <b>-t</b> 标志指定的 <i>Stanza</i> 中有效，并且与由 <b>-f</b> 标志指定的其他参数一致。                                                                    |
| <b>-D</b>                        | 将 <i>Stanza</i> 的所有参数设置为其缺省值。                                                                                                                                 |
| <b>-m</b> <i>Filename2</i>       | 将 <i>Filename2</i> 文件与当前的 <i>Filename</i> 文件合并。                                                                                                               |

## 退出状态

|    |                                                                                                                                                                                              |
|----|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 0  | 更改当前有效。                                                                                                                                                                                      |
| >0 | 以下条件之一导致了错误： <ul style="list-style-type: none"><li>指定的 <i>Filename</i>、<i>Filename2</i> 或 <i>Stanza</i> 无效。</li><li><i>Parameter=Value</i> 对于 <i>Parameter</i> 无效。</li><li>没有提供消息。</li></ul> |

## 示例

- 要在 **/etc/tunables/nextboot** 文件中更新 **pacefork** 文件，请输入：

```
tunchange -f nextboot -t schedo -o pacefork=10
```
- 要在 **/home/mine/mytunable** 文件中更新 **pacefork**，请输入：

```
tunchange -f /home/mine/mytunable -t schedo -o pacefork=10
```
- 要在 **/etc/tunables/nextboot** 文件中将所有 **schedo** 节复位为其缺省值，请输入：

```
tunchange -f nextboot -t schedo -D
```
- 要将 **/home/mine/mytunable** 文件与 **/etc/tunables/nextboot** 文件合并，请输入：

```
tunchange -f nextboot -m /home/mine/mytunable
```

## 文件

|                            |                         |
|----------------------------|-------------------------|
| <b>/usr/sbin/tunchange</b> | 包含 <b>tunchange</b> 命令。 |
| <b>/etc/tunables/</b>      | 包含缺省可调文件。               |

## 相关信息

**raso**、**schedo**、**vmo**、**ioo**、**no**、**nfso**、**tuncheck**、**tundefault**、**tunrestore** 和 **tunsave** 命令。

*AIX 5L Version 5.3 Files Reference* 中的 Tunables File Format。

*AIX 5L Version 5.3 Performance Tools Guide and Reference* 中的 Kernel Tuning。

---

## tuncheck 命令

### 用途

验证可调文件。

### 语法

**tuncheck** [ **-r** | **-p** ] **-f** *Filename*

### 描述

**tuncheck** 命令验证可调文件。检查指定文件中列出的所有可调参数的范围和相关性。如果检测到问题，则发出警告。

有两种类型的验证:

#### 对于当前上下文

检查是否能立即应用 *Filename*。将 *Filename* 中未列出的可调参数解释为当前值。如果以较小值而非其当前值列出 **Incremental** 类型的可调参数，则检查将失败；如果以不同值而非其当前值列出 **Bosboot** 或 **Reboot** 类型的可调参数，则检查也将失败。

#### 对于下一个引导上下文

检查在重新引导过程中是否能应用 *Filename*，也就是，它是否是有效的下一个引导文件。允许减小 **Incremental** 类型的可调参数。如果以不同值而非其当前值列出 **Bosboot** 或 **Reboot** 类型的可调参数，将发出警告但检查不会失败。

另外，如果 *Filename* 包含未知节或在已知节中包含未知的可调参数，则发出警告。然而，这并不会使检查失败。

检查成功后，更新了已检查文件的信息节中的 **AIX\_level**、**Kernel\_type** 和 **Last\_validation** 字段。

### 标志

|                           |                                                                                     |
|---------------------------|-------------------------------------------------------------------------------------|
| <b>-f</b> <i>Filename</i> | 指定要检查的可调文件的名称。如果它不包含 ‘/’（正斜杠）字符，则该名称与 <b>/etc/tunables</b> 相关。                      |
| <b>-p</b>                 | 在当前和引导上下文中检查 <i>Filename</i> 。这就等于运行 <b>tuncheck</b> 两次，一次不带任何标志，一次带有 <b>-r</b> 标志。 |
| <b>-r</b>                 | 在引导上下文中检查 <i>Filename</i> 。                                                         |

如果未指定 **-p** 或 **-r**，则根据当前上下文检查 *Filename*。

### 可调参数类型

|             |                                      |
|-------------|--------------------------------------|
| Dynamic     | 可以随时更改                               |
| Static      | 永不能更改                                |
| Reboot      | 只能在重新引导序列过程中更改                       |
| Bosboot     | 只有在运行 <b>bosboot</b> 命令并重新启动机器时才能更改。 |
| Mount       | 只有在以后加载文件系统或目录时所作的更改才生效。             |
| Incremental | 除了在引导时进行操作，否则只能递增。                   |
| Connect     | 更改仅对以后套接字连接有效。                       |

## 退出状态

- 0     *Filename* 有效。  
>0    *Filename* 无效，提供消息。

## 示例

1. 要检查 **mytunable** 是否能够立即应用，请输入：  
    `tuncheck -f ./mytunable`
2. 要检查在重新引导过程中是否能应用 **/etc/tunables/nextboot**，输入：  
    `tuncheck -r -f nextboot`
3. 要检查是否能立即和在重新引导后应用 **/etc/tunables/nextboot**，输入：  
    `tuncheck -p -f nextboot`

## 文件

|                                 |                        |
|---------------------------------|------------------------|
| <code>/usr/sbin/tuncheck</code> | 包含 <b>tuncheck</b> 命令。 |
| <code>/etc/tunables</code>      | 包含所有可调文件。              |

## 相关信息

**raso**、**schedo**、**vmo**、**ioo**、**no**、**nfso**、**tunchange**、**tundefault**、**tunrestore** 和 **tunsave** 命令。

*AIX 5L Version 5.3 Files Reference* 中的 Tunables File Format。

*AIX 5L Version 5.3 Performance Tools Guide and Reference* 中的 Kernel Tuning。

---

## tundefault 命令

### 用途

将所有可调参数复位为其缺省值。

### 语法

**tundefault** [ **-r** | **-p** ]

### 描述

**tundefault** 命令启动所有带有 **-D** 标志的调整命令 (**ioo**、**vmo**、**schedo**、**no**、**nfso** 和 **raso**)。它将所有 AIX 可调参数复位为其缺省值，但 **Bosboot** 和 **Reboot** 类型的参数除外；除非指定了 **-r**，否则将 **Incremental** 类型的参数值设置为大于其缺省值。对于不能进行更改的任何参数，将会显示错误消息。

### 标志

- p**     进行永久的更改：将所有可调参数复位为其缺省值，并更新 **/etc/tunables/nextboot** 文件。  
**-r**     将复位为其缺省值操作延迟到下一次重新引导时。该命令将清除 **/etc/tunables/nextboot** 文件中各个节，且如果必要，将提出 **bosboot**，并警告需要进行重新引导。

## 可调参数类型

|             |                                            |
|-------------|--------------------------------------------|
| Dynamic     | 可以随时更改                                     |
| Static      | 永不能更改                                      |
| Reboot      | 只能在重新引导序列过程中更改                             |
| Bosboot     | 只有在运行 <code>bosboot</code> 命令并重新启动机器时才能更改。 |
| Mount       | 只有在以后加载文件系统或目录所作的更改才生效。                    |
| Incremental | 除了在引导时进行操作，否则只能递增。                         |
| Connect     | 更改仅对以后套接字连接有效。                             |

## 示例

1. 要将所有可调参数永久复位为其缺省值，输入：

```
tundefault -p
```

启动所有调整命令并带有 **-Dp** 标志。该操作将所有可调参数复位为其缺省值。同时还将更新 `/etc/tunables/nextboot` 文件。该命令将所有可调参数完全、永久地复位为其缺省值。

2. 要将对所有可调参数的设置延迟到下一次重新引导时，输入：

```
tundefault -r
```

调用所有带有 **-Dr** 的调整命令。该操作将清除 `/etc/tunables/nextboot` 文件中的所有节，且如果必要，将提出 `bosboot` 并显示消息警告必须进行重新引导才能使更改生效。

## 文件

`/usr/sbin/tundefault`  
`/etc/tunables/`

包含 `tundefault` 命令。  
包含所有可调文件。

## 相关信息

`raso`、`schedo`、`vmo`、`ioo`、`no` 和 `nfso` 命令。

第 502 页的『`tunchange` 命令』、第 504 页的『`tuncheck` 命令』、『`tunrestore` 命令』和第 508 页的『`tunsave` 命令』。

*AIX 5L Version 5.3 Files Reference* 中的 Tunables File Format。

*AIX 5L Version 5.3 Performance Tools Guide and Reference* 中的 Kernel Tuning。

---

## tunrestore 命令

### 用途

从文件恢复可调参数值。

### 语法

```
tunrestore [ -r ] -f Filename
```

```
tunrestore -R
```



注: **tunrestore -R** 只能从 **inittab** 中调用。

## 描述

**tunrestore** 命令将恢复文件中存储的所有可调参数的值。

**tunrestore -f Filename** 立即应用于 *Filename*。所有 *Filename* 中列出的可调参数将被设置为该文件中定义的值。*Filename* 中未列出的可调参数将保持未变。明示设置为 **DEFAULT** 的可调参数将被设置为其缺省值。

**tunrestore -r -f Filename** 在下次启动时应用于 *Filename*。该操作是通过检查指定文件中不一致之处（相当于运行 **tuncheck**）并将其复制到 **/etc/tunables/nextboot** 来完成的。如果需要 **bosboot**，将会提示用户运行该命令。

**tunrestore -R** 只能在重新启动时运行。将会修改所有未根据 **nextboot** 文件定义值设置的可调项。未在 **nextboot** 文件中列出的可调项将被强制更改为缺省值。所有的操作、警告和错误将记录在 **/etc/tunables/lastboot.log** 中。

此外，将会自动生成一个称为 **/etc/tunables/lastboot** 的新可调文件。该文件包含所有以数字值列出的可调项。代表缺省值的值带有注释 **DEFAULT VALUE**（缺省值）。其信息节包括 **/etc/tunables/lastboot.log** 文件的校验和，以确认可识别 **lastboot/lastboot.log** 文件对。

## 标志

|                           |                                                                      |
|---------------------------|----------------------------------------------------------------------|
| <b>-f</b> <i>Filename</i> | 指明要应用的可调文件名。如果它不包含 <b>'/'</b> （正斜杠）字符，则该名称与 <b>/etc/tunables</b> 相关。 |
| <b>-r</b>                 | 使指定的文件变为新的 <b>nextboot</b> 文件。                                       |
| <b>-R</b>                 | 在启动进程中恢复 <b>/etc/tunables/nextboot</b> 。                             |

## 可调参数类型

|             |                                      |
|-------------|--------------------------------------|
| Dynamic     | 可以随时更改                               |
| Static      | 永不能更改                                |
| Reboot      | 只能在重新引导序列过程中更改                       |
| Bosboot     | 只有在运行 <b>bosboot</b> 命令并重新启动机器时才能更改。 |
| Mount       | 只有在以后加载文件系统或目录时所作的更改才生效。             |
| Incremental | 除了在引导时进行操作，否则只能递增。                   |
| Connect     | 更改仅对以后套接字连接有效。                       |

## 示例

- 要恢复所有存储在 **/etc/tunables/mytunable** 中的可调值，输入：  

```
tunrestore -f mytunable
```
- 要验证 **/etc/tunables/mytunable** 并使其成为新的 **nextboot** 文件，输入：  

```
tunrestore -r -f mytunable
```

## 文件

|                               |                          |
|-------------------------------|--------------------------|
| <b>/usr/sbin/tunrestore</b>   | 包含 <b>tunrestore</b> 命令。 |
| <b>/etc/tunables</b>          | 包含可调文件。                  |
| <b>/etc/tunables/nextboot</b> | 包含在下次启动时要应用的值。           |

`/etc/tunables/lastboot`  
`/etc/tunables/lastboot.log`

包含上一次启动后所有可调值。  
包含在上一次启动时由 `tunrestore` 发出的消息、警告和错误。

## 相关信息

`raso`、`schedo`、`vmo`、`ioo`、`no` 和 `nfso` 命令。

第 502 页的『`tunchange` 命令』，第 504 页的『`tuncheck` 命令』、第 505 页的『`tundefault` 命令』和『`tunsave` 命令』。

*AIX 5L Version 5.3 Files Reference* 中的 Tunables File Format。

*AIX 5L Version 5.3 Performance Tools Guide and Reference* 中的 Kernel Tuning。

---

## tunsave 命令

### 用途

将当前可调参数值保存到文件。

### 语法

```
tunsave [ -a | -A ] -f | -F Filename [ -d Description ]
```

### 描述

`tunsave` 命令将可调参数的当前状态保存在一个文件中。

如果还不存在 *Filename*，则创建一个新文件。如果该文件已存在，则除非指定了 **-F** 标志，（在这种情况下将覆盖现有的文件）否则会打印出错误消息。

### 标志

|                              |                                                                                                                                    |
|------------------------------|------------------------------------------------------------------------------------------------------------------------------------|
| <b>-a</b>                    | 保存全部可调参数，包括当前设置为其缺省值的参数。这些参数保存时带有特殊值 <code>DEFAULT</code> 。                                                                        |
| <b>-A</b>                    | 保存全部可调参数，包括当前设置为其缺省值的参数。这些参数以数字形式保存并将注释（ <code># DEFAULT VALUE</code> ）附加到行中作为标志。                                                  |
| <b>-d</b> <i>Description</i> | 指定 <i>Description</i> 字段使用的文本。在 <i>Description</i> 字段中必须转义或引用特殊字符。                                                                 |
| <b>-f</b> <i>Filename</i>    | 指定保存可调参数的可调文件的名称。如果 <i>Filename</i> 已存在，将会打印错误消息。如果不包含 <code>'/'</code> （正斜杠）字符，则 <i>Filename</i> 与 <code>/etc/tunables</code> 相关。 |
| <b>-F</b> <i>Filename</i>    | 指定保存可调参数的可调文件的名称。如果 <i>Filename</i> 已存在，现有文件将会被覆盖。如果不包含 <code>'/'</code> （正斜杠）字符，则文件名与 <code>/etc/tunables</code> 相关。              |

### 示例

1. 要将全部与其缺省值不同的可调参数保存在 `/etc/tunables/mytunable` 中，输入：

```
tunsave -f mytunable
```

2. 要保存全部可调参数，包括目前设置为其缺省值的参数，但要以特殊值 `DEFAULT` 替换缺省值，输入：

```
tunsave -a -f /home/admin/mytunable
```

3. 要保存全部可调参数，包括目前使用所有数字值设置为其缺省值的参数，但要以注释 `DEFAULT VALUE` 来标志缺省值，输入：

```
tunsave -A -f mytunable
```

## 文件

`/usr/bin/tunsave`  
`/etc/tunables`

包含 `tunsave` 命令。  
包含全部已保存的文件。

## 相关信息

`raso`、`schedo`、`vmo`、`ioo`、`no` 和 `nfso` 命令。

第 502 页的『`tunchange` 命令』、第 504 页的『`tuncheck` 命令』、第 505 页的『`tundefault` 命令』和 第 506 页的『`tunrestore` 命令』。

*AIX 5L Version 5.3 Files Reference* 中的 Tunables File Format。

*AIX 5L Version 5.3 Performance Tools Guide and Reference* 中的 Kernel Tuning。

---

## turnacct 命令

### 用途

给 `accton` 命令提供接口以打开或关闭进程记帐。

### 语法

```
/usr/sbin/acct/turnacct on | off | switch
```

### 描述

`turnacct` 命令给 `accton` 命令提供接口以打开或关闭进程记帐。因为没有缺省值，所以必须指定是否要打开或关闭进程记帐。

`switch` 标志关闭记帐，并将当前活动数据文件（`/var/adm/pacct`）移动到 `/var/adm/pacctincr` 文件中的下一个自由名称，其中，`incr` 是从 1 开始的数字，随着每个附加的 `pacct` 文件而递增。移动 `pacct` 文件后，`turnacct` 命令再次打开记帐。

`turnacct switch` 命令通常由 `ckpacct` 命令调用，在 `cron` 守护程序下运行，使活动的 `pacct` 数据文件保持在可管理的大小内。

### 安全性

访问控制：此命令只授予 `adm` 组的成员以执行（x）权限。

## 文件

`/usr/sbin/acct`  
`/var/adm/pacct`  
`/var/adm/pacct*`

包含记帐命令的路径。  
包含进程记帐的当前文件。  
如果 `pacct` 文件变得过大时使用。

## 相关信息

**accton** 命令、**ckpacct** 命令。

**cron** 守护程序。

《操作系统与设备管理》中的『设置记帐子系统』说明了建立记帐系统所必须采取的步骤。

有关“记帐系统”、每日和每月报告的准备以及记帐文件的更多信息，请参阅《操作系统与设备管理》中的『系统记帐』。

---

## turnoff 命令

### 用途

设置关闭 **/usr/games** 目录中文件的权限代码。

### 语法

**turnoff**

### 描述

**turnoff** 命令设置 **/usr/games** 目录中文件的权限代码。运行该命令需要有 **root** 用户权限。

**turnoff** 命令在 **/usr/games** 中寻找权限设置为 111 的文件，并将其权限设置为 000。如果在 **/usr/games** 目录中安装任何新游戏，请将权限设置为 111。

### 文件

**/usr/games**            包含系统游戏的位置。

## 相关信息

**arithmetic** 命令、**back** 命令、**bj** 命令、**craps** 命令、**fish** 命令、**fortune** 命令、**hangman** 命令、**moo** 命令、**number** 命令、**quiz** 命令、**ttt** 命令、**turnon** 命令和 **wump** 命令。

---

## turnon 命令

### 用途

设置打开游戏目录中文件的权限代码。

### 语法

**turnon**

### 描述

**turnon** 命令设置 **/usr/games** 目录中文件的权限代码。运行该命令需要有 **root** 用户权限。

**turnon** 命令寻找权限设置为 000 的文件，并将其设置成 111（所有用户的执行权限制）。如果在 **/usr/games** 目录中安装任何新游戏，请将权限设置为 111。

# 文件

`/usr/games` 包含系统游戏的位置。

## 相关信息

**arithmetic** 命令、**back** 命令、**bj** 命令、**craps** 命令、**fish** 命令、**fortune** 命令、**hangman** 命令、**moo** 命令、**number** 命令、**quiz** 命令、**ttt** 命令、**turnoff** 命令和 **wump** 命令。

---

## tvi 命令

### 用途

以全屏显示提供一个可信的编辑器。

### 语法

```
tvi [ -l ] [ -R ] [ -w Number ] [ -c [ Subcommand ] ] [ File ... ]
```

### 描述

**tvi** 命令调用 **tvi** 编辑器 (**vi** 编辑器的可信的版本) 以编辑由 *File* 参数指定的文件。文件按照指定的顺序编辑。如果未提供文件名, 命令将打开一个可创建文本的新文件, 但如果尝试保存文本到文件时, 将会提示用户添加文件名到保存命令, 例如 **:w File**。关于更多信息, 请参阅示例部分。

在命令方式中进入或离开 **tvi** 编辑器, 但要添加或更改文本, 必须进入文本输入方式。关于启动文本输入方式的子命令的信息, 请参阅文本输入方式的描述。要离开文本输入方式, 请按下 **Esc** 键。这样您就能返回命令方式, 在命令方式中可以使用一个 **:w** 命令保存文本到文件中, 例如对于退出 **tvi** 编辑器, 使用 **:q** 命令。

因为由 **tvi** 命令启动的全屏显示的编辑器是基于 **ex** 编辑器的, 用户可在 **tvi** 编辑器中使用 **ex** 子命令。子命令在显示屏的光标位置起作用。

**tvi** 编辑器制作一份在编辑缓冲区中编辑的文件副本。除非用户保存更改, 否则文件内容将不会发生更改。

**注:** **tvi** 编辑器不支持 **vi** 编辑器的几个功能。如果参考关于 **vi** 编辑器的信息, 请注意 **tvi** 编辑器永远不支持 **-r** 标志、**-t** 标志、shell 转义、用户定义宏、键映射以及设置 **vi** 选项。

### tvi 编辑器限制

**tvi** 编辑器的最大限制假设为单字节字符。这些限制如下:

- 每个全局命令列表 256 个字符
- 在 shell 转义命令中有 2048 个字符
- 在一个字符串值的选项中 128 个字符
- 在一个标记名称中 30 个字符
- 静默强制 524,230 行
- 128 个映射宏, 总计 2048 字符

## 编辑方式

**tv**i 编辑器按照以下方式操作:

|        |                                                                                                                                                                                                                                                                                                                                                                        |
|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 命令方式   | <b>tv</b> i 编辑器在命令方式中启动。可以调用任何子命令，仅在文本输入方式中更改文本的子命令除外。要了解子命令的描述，请参见 <b>tv</b> i 编辑器子命令中的主题。要识别不能从命令方式调用的子命令，请参见在输入方式中更改文本。子命令和其他方式结束后， <b>tv</b> i 编辑器返回到命令方式。按下 <b>Esc</b> 键取消不完全的子命令。                                                                                                                                                                                |
| 文本输入方式 | 当用户使用允许的命令添加或更改文本时， <b>tv</b> i 编辑器进入文本输入方式。要查看启用文本输入方式的子命令列表，请参考向文件添加文本、从命令方式更改文本的子命令、 <b>C</b> 子命令和 <b>cx</b> 子命令。进入以上任意一个子命令后，用户可以使用在文本输入方式中起作用的任一子命令编辑文本。要了解关于子命令的描述，请参见『 <b>tv</b> i 编辑器子命令』中的主题。要从文本输入方式返回到命令方式，通常按 <b>Esc</b> 键退出，或者按 <b>Ctrl+C</b> 键创建一个 <b>INTERRUPT</b> 信号。                                                                                  |
| 最后行方式  | 某些子命令读取屏幕底端显示的一行的输入。这些子命令包括带有以下前缀的命令： <b>(</b> （冒号）、 <b>/</b> （斜杠）和 <b>?</b> （问号）。当用户输入起始字符时， <b>tv</b> i 编辑器将光标定位于屏幕的底端，使用户可输入剩余的命令字符。要运行子命令，按下 <b>Enter</b> 键。要运行子命令，按下 <b>Enter</b> 键。要取消子命令，按下 <b>Ctrl+C</b> 键，创建一个 <b>INTERRUPT</b> 信号。当用户使用 <b>(</b> （冒号）进入最后行方式时，以下字符如果用在指定计数的命令前，有特定的含义：<br><br><b>%</b> 所有与光标位置无关的行<br><br><b>\$</b> 最后一行<br><br><b>.</b> 当前行 |

## 定制 **tv**i 编辑器

用户可按照“设置 **vi** 编辑器选项”中的指示，临时定制 **tv**i 编辑器。“永久设置 **vi** 选项”节内容不适用于 **tv**i 编辑器。

## **tv**i 编辑器子命令

适用于 **tv**i 编辑器的 **vi** 编辑器子命令的信息，请见以下列表：

- **vi** 常规 子命令语法。
- 用于调节屏幕的 **vi** 子命令。
- 用 **vi** 编辑器编辑文本。
- **tv**i 编辑器不支持 **vi** 编辑器中的 **Entering Shell** 命令。
- 用 **vi** 编辑器操作文件。
- 中断和结束 **vi** 编辑器的子命令。

## 标志

|                                 |                                                                                                                                                                                             |
|---------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-c</b> [ <i>Subcommand</i> ] | 编辑开始前，执行 <b>ex</b> 编辑器子命令。该子命令提供一个面向行的文本编辑器。当对于 <i>Subcommand</i> 参数输入了空操作数时，就像 <b>-c</b> ，编辑器将光标定位于文件的最后一行。                                                                                |
| <b>-l</b>                       | 在 <b>ILSP</b> 方式中进入编辑器。在该方式中，编辑器为 <b>LISP</b> 代码相应缩进， <b>(, ), {, }, [[</b> 和 <b>]]</b> 子命令也为适应 <b>LISP</b> 相应作出修改。这些子命令将光标定位于特定的 <b>LISP</b> 函数的位置。关于更多 <b>LISP</b> 子命令的信息，请参见移动到语句、段落和章节。 |
| <b>-R</b>                       | 设置 <b>readonly</b> 选项，保护文件不被重写。                                                                                                                                                             |
| <b>-w</b> <i>Number</i>         | 将窗口大小的缺省值设置为由 <i>Number</i> 参数指定的值。当您在低速行使用编辑器时，该参数十分有用。                                                                                                                                    |

+ [Subcommand] 与 **-c** 子命令相同。

## 安全性

访问控制: 此命令向所有拥有**可信计算库**的用户授予执行 (x) 权限。

审计事件:

| 事件         | 信息  |
|------------|-----|
| <b>TVI</b> | 文件名 |

## 示例

1. 要调用可信的编辑器编辑 `plans` 文件, 输入:

```
tvi plans
```

该命令使 **tvi** 编辑器进入命令方式。要添加和更改文本, 必须进入文本输入方式, 或者使用命令方式接受的命令。更多信息, 请参见文本输入方式中的描述。

2. 要保存经由 **tvi** 编辑器创建的文本, 按 `Esc` 键离开文本输入方式, 然后输入以下保存命令之一: **:w**、**:w File** 或者 **:w! File**, 例如:

```
:w plans
```

在该示例中, 如果未经指定文件名即发出 **tvi** 命令, 如 `plans` 之类的文件名是必须的。如果文件已命名, **:w** 命令则不需要 *File* 参数。如果希望覆盖现存的文件, 使用以 *File* 参数指定要覆盖文件的 **:w! File** 命令。

如果未经提供文件名即试图保存一个未命名的文件, 将会出现以下消息:

```
No current filename
```

如果出现这种情况, 重复带有文件名的 **:w** 命令。

3. 要从文本输入方式退出 **tvi** 编辑器, 按 `Esc` 键, 输入命令方式, 然后输入:

```
:q!
```

如果编辑器已处于命令方式, 给出退出 (**q!**) 命令前, 无需按 `Esc` 键。命令。

## 文件

`/usr/bin/tvi` 包含 **tvi** 命令。

## 相关信息

**ex** 命令、**vi** 命令。

关于用户的标识和认证、自主访问控制、可信计算库和审计的更多信息, 请参阅 《安全性》 中的 `Securing the network`。

---

## twconvdict 命令

### 用途

将其他用户字典转换成操作系统用户字典。该命令仅适用于 AIX 4.2 或更高版本。

### 语法

```
twconvdict [ -i Type ] [ -v CodePage ] [ -f Source ] [ -t Target ]
```

### 描述

**twconvdict** 命令将字典转换成操作系统用户字典。支持的代码页为 SOPS、PS55 和 ET。字典类型包括 Tseng\_Jye 和 Phonetic 用户字典。

### 标志

|                           |                                                                                   |
|---------------------------|-----------------------------------------------------------------------------------|
| <b>-f</b> <i>Source</i>   | 指定要转换的字体文件名称。                                                                     |
| <b>-i</b> <i>Type</i>     | 指定要转换成的字典类型。类型 可以是:<br><br><b>TJ</b> Tseng_Jye, 或<br><br><b>PH</b> Phonetic,      |
| <b>-t</b> <i>Target</i>   | 指定转换后的字体文件名称。                                                                     |
| <b>-v</b> <i>CodePage</i> | 指定要转换成的代码页类型。代码页可以是:<br><br><b>SOPS</b><br><br><b>PS55</b> , 或<br><br><b>ET</b> . |

### 退出状态

该命令返回以下退出值:

|              |       |
|--------------|-------|
| <b>0</b>     | 成功完成。 |
| <b>&gt;0</b> | 发生错误。 |

### 安全性

访问控制: 您必须具有 root 用户权限以运行该命令。

审计事件: 不适用

### 示例

要将字典 USRFONT.C12 转换成 SOPS 类型代码页的操作系统字典以及名为 aix 的 Tseng\_Jye 类型字典, 输入:

```
twconvdict -i TJ -v SOPS -f USRFONT.C12 -t aix
```

### 文件

**/usr/lpp/tls/bin/twconvdict** 包含 **twconvdict** 命令。



---

## twconvfont 命令

### 用途

将其他字体文件转换成 BDF 字体文件。

### 语法

```
twconvfont [ -v CodePage ] [ -f Source ] [ -t Target ]
```

### 描述

twconvfont 命令将一种字体文件类型转换成为 BDF 字体文件。支持的代码页为 SOPS、PS55 和 ET。

### 标志

**-f Source** 指定要转换的字体文件名称。  
**-t Target** 指定转换后的字体文件名称。  
**-v CodePage** 指定要转换成的代码页类型。代码页可以是：

**SOPS**

**PS55**, 或者

**ET**。

### 退出状态

该命令返回以下退出值：

**0** 成功完成。  
**>0** 发生错误。

### 安全性

访问控制：您必须具有 root 用户权限以运行该命令。

审计事件：不适用

### 示例

要将字体文件 USRFONT.C12 转换成名为 user.bdf 、类型为 SOPS 的代码页的 BDF 字体文件，请输入：

```
twconvfont -v SOPS -f USRFONT.C12 -t user.bdf
```

### 文件

**/usr/lpp/tls/bin/twconvfont**

包含 **twconvfont** 命令。

---

## type 命令

### 用途

写命令类型的描述

### 语法

**type***CommandName* ...

### 描述

**type** 命令的标准输出包含有关指定命令的信息，并标识该命令是否为 shell 内置命令、子例程、别名或关键字。

**type** 命令表示如何解释指定的命令（如果使用了该命令）。如果适用的话，**type** 命令显示相关的路径名。

因为 **type** 命令必须知道当前 shell 环境的内容，所以该命令将作为 Korn shell 或 POSIX shell 常规内置命令提供。如果在独立的命令执行环境中调用 **type** 命令，则该命令可能无法产生精确的结果。以下示例中正是这种情况：

```
nohup type writer
find . -type f | xargs type
```

### 退出状态

返回以下退出值：

**0** 成功完成。  
**>0** 发生错误。

### 示例

1. 要了解 **cd** 命令是否基础命令、别名或者某种其他命令类型，输入：

```
type cd
```

屏幕显示出以下信息：

```
cd is a shell builtin
```

2. 要查看 **find** 命令的位置，输入：

```
type find
```

屏幕显示出以下信息：

```
find is /usr/bin/find
```

### 文件

**/usr/bin/ksh** 包含 Korn shell **type** 内置命令。

### 相关信息

**bsh** 命令、**command** 命令、**ksh** 命令。

---

## ucfgif 方法

### 用途

从内核中卸载接口实例。

### 语法

**ucfgif** [ **-l** *InterfaceInstance* ]

### 描述

**ucfgif** 方法从内核中除去接口实例。要除去接口实例，**ucfgif** 方法进行以下操作：

1. 通过调用 **/usr/sbin/ifconfig** 接口拆离以卸载接口软件。
2. 将接口实例的状态标志设置成 **defined**。

**注：****ucfgif** 方法是一种编程工具，不应在命令行中执行。

### 标志

**-l** *InterfaceInstance* 指定要取消配置的接口实例。如果未指定接口名称，将取消配置所有已配置的接口实例。

### 示例

要将接口实例从内核中除去，按照以下格式输入该方法：

```
ucfgif -l tr0
```

在本示例中，接口实例名称为 **tr0**。

### 相关信息

**ifconfig** 命令。

**odm\_run\_method** 子例程。

《网络与通信管理》中的『TCP/IP 网络接口』。

*Kernel Extensions and Device Support Programming Concepts* 中的 Writing a Device Method。

*General Programming Concepts* 中的 Object Data Manager (ODM) Overview for Programmers。

---

## ucfginet 方法

### 用途

从内核卸载因特网实例和所有相关的接口实例。

### 语法

**ucfginet**

## 描述

**ucfginet** 方法从内核中卸装因特网实例。该子例程同时还删除“地址系列域”开关表和“网络输入接口”开关表中相应的条目。**ucfginet** 方法还将实例的状态标志设置为 **defined**。**ucfginet** 方法由 **rmdev** 高层命令来调用。

注: **ucfginet** 方法是一种编程工具, 不应在命令行中执行。

## 相关信息

**cfginet** 方法、**rmdev** 命令和 **ucfgif** 方法。

**odm\_run\_method** 子例程。

《网络与通信管理》中的『TCP/IP 网络接口』。

*Kernel Extensions and Device Support Programming Concepts* 中的 Writing a Device Method。

*General Programming Concepts* 中的 Object Data Manager (ODM) Overview for Programmers。

---

## ucfgqos 方法

### 用途

从内核取消配置并卸装“服务质量”(QoS)实例。

### 语法

**ucfgqos**

### 描述

**ucfgqos** 方法在主机上禁用 TCP/IP 协议组的“服务质量”(QoS)。该方法将 QoS 实例从 TCP/IP 实例中分离, 并将其从内核卸装。

注: **ucfgqos** 方法是一种编程工具, 并非意在命令行中调用。

### 示例

要在主机上配置 QoS, 使用以下格式:

```
ucfgqos
```

## 相关信息

**cfgqos** 方法和 **ucfginet** 方法。

《网络与通信管理》中的『TCP/IP 服务质量(QoS)』。

---

## ucfgvsd 命令

### 用途

**ucfgvsd** - 取消配置虚拟共享磁盘。

## 语法

**ucfgvsd** **{-a | vsd\_name ...}**

## 描述

**ucfgvsd** 命令取消配置指定的虚拟共享磁盘。要取消配置的指定虚拟共享磁盘必须处于停止状态。该命令不更改任何虚拟共享磁盘的定义。它将虚拟共享磁盘从停止状态移动到已定义状态。

可以使用“系统管理界面工具”（SMIT）来运行该命令。要使用 SMIT，输入：

```
smit vsd_mgmt
```

并选择 **Unconfigure a Virtual Shared Disk** 选项。

在常规情况下，您不应发出该命令。“可恢复的”虚拟共享磁盘子系统使用该命令以受控制的方式来管理共享磁盘。如果您发出该命令，结果可能是不可预测的。

## 标志

**-a** 指定处于停止状态的所有虚拟共享磁盘都要进行取消配置。

## 参数

*vsd\_name* 指定一个虚拟共享磁盘。指定的磁盘必须处于停止状态。如果所有磁盘都已取消配置，且您指定了 **VSD0**，则该命令将试图从内核卸装设备驱动程序。

## 安全

您必须具有 **root** 用户权限以运行该命令。

## 退出状态

**0** 说明命令已成功完成。

非零 表明发生错误。

## 限制

在常规情况下，您不应发出该命令。“可恢复的”虚拟共享磁盘子系统使用该命令以受控制的方式来管理共享磁盘。如果您发出该命令，结果可能是不可预测的。

## 标准输出

当前 **RVSD** 子系统运行级别。

## 示例

要取消配置处于停止状态的虚拟共享磁盘 **vsd1vg1n1**，请输入：

```
ucfgvsd vsd1vg1n1
```

## 位置

**/opt/rsct/vsd/bin/ucfgvsd**

## 相关信息

命令: **cfgvsd**、**lsvsd**、**preparevsd**、**resumevsd**、**startvsd**、**stopvsd**、**suspendvsd**

---

## uconvdef 命令

### 用途

编译或生成由 **iconv** 库使用的 UCS-2 (Unicode) 转换表。

### 语法

**uconvdef** [ **-f** *SrcFile* ] [ **-v** ] *UconvTable*

### 描述

**uconvdef** 命令读取 *SrcFile* 并在 *UconvTable* 中创建一个已编译的转换表。*SrcFile* 定义了 UCS-2 和多字节代码集 (每个字符由一个或多个字节组成) 之间的映射。*UconvTable* 的格式可以由位于 **/usr/lib/nls/loc/uconv** 目录中的 UCSTBL 转换方法装入。该方法使用表来支持两个方向上的 UCS-2 转换。

### 标志

|                          |                                                             |
|--------------------------|-------------------------------------------------------------|
| <b>-f</b> <i>SrcFile</i> | 指定转换表源文件。如果未使用该标志, 则读取标准输入。                                 |
| <b>-v</b>                | 使输出已处理的文件语句。                                                |
| <i>UconvTable</i>        | 指定由 <b>uconvdef</b> 命令创建的已编译表的路径名。这应是定义 UCS-2 的进出转换的代码集的名称。 |

### 退出状态

返回以下退出值:

|    |       |
|----|-------|
| 0  | 成功完成。 |
| >0 | 发生错误。 |

### 示例

要访问已编译的 UCS-2 转换表:

1. 使用多字节代码集的名称创建已编译的 *UconvTable*。例如, IBM-850 和 UCS-2 之间的转换表可通过以下方式编译:

```
uconvdef -f IBM-850.ucmap IBM-850
```

2. 将该表放入名为 **uconvTable** 的目录中。缺省的系统目录为 **/usr/lib/nls/loc/uconvTable**。如果使用其他目录, 需要将 **LOCPATH** 环境变量设置为包含父目录 (例如, **/usr/lib/nls/loc**)。

```
mv IBM-850 /usr/lib/nls/loc/uconvTable
```

3. 在名为 **iconv** 的目录中为每个方向的转换创建符号链接。这些链接的名称应由 “From” 代码集和 “To” 代码集并置组成, 中间以一道下划线隔开。链接应设置为指向 **/usr/lib/nls/loc/uconv/UCSTBL** 转换方法。这些链接的缺省目录为 **/usr/lib/nls/loc/iconv**。如果使用了其他目录, 需要将 **LOCPATH** 环境变量设置为包含父目录 (例如, **/usr/lib/nls/loc**)。

```
ln -s /usr/lib/nls/loc/uconv/UCSTBL \  
/usr/lib/nls/loc/iconv/IBM-850_UCS-2
```

```
ln -s /usr/lib/nls/loc/uconv/UCSTBL \  
/usr/lib/nls/loc/iconv/UCS-2_IBM-850
```

注：只有在命令分割成两行时才需要行连续字符 \ (反斜杠)。

## 相关信息

**iconv** 命令

**iconv** 子例程、**iconv\_close** 子例程、**iconv\_open** 子例程。

**unconvdef** 源文件格式。

《AIX 5L V5.3 本地语言支持指南和参考大全》中的代码集概述、转换器概述和 UCS-2 交换转换器列表。

---

## undefif 方法

### 用途

将接口对象从系统配置数据库中除去。

### 语法

**undefif** [ **-l** *InterfaceInstance* ]

### 描述

**undefif** 方法通过以下方法，从系统配置数据库中删除指定的接口实例：

1. 除去与接口实例关联的数据库对象。
2. 除去与接口实例关联的连接和属性信息。

### 标志

**-l** *InterfaceInstance*

指定要取消定义的接口实例。如果未指定接口实例，**undefif** 方法将会取消定义所有已定义的接口实例。

### 示例

要将接口实例从数据库中除去，请输入与以下类似的方法：

```
undefif -l tr0
```

在本示例中，要除去的接口实例是 `tr0`。

## 相关信息

**rmdev** 命令、**undefinet** 方法。

**odm\_run\_method** 子例程。

《网络与通信管理》中的『TCP/IP 网络接口』。

*Kernel Extensions and Device Support Programming Concepts* 中的 Writing a Device Method。

*General Programming Concepts* 中的 Object Data Manager (ODM) Overview for Programmers。

---

## udfinet 方法

### 用途

在配置数据库中取消定义因特网实例。

### 语法

**udfinet**

### 描述

**udfinet** 方法除去与因特网实例关联的数据库信息，包括与其关联的属性信息。

注: **udfinet** 方法是一种编程工具，不应在命令行中执行。

### 相关信息

**rmdev** 命令。

**odm\_run\_method** 子例程。

《网络与通信管理》中的『TCP/IP 网络接口』。

*Kernel Extensions and Device Support Programming Concepts* 中的 Writing a Device Method。

*General Programming Concepts* 中的 Object Data Manager (ODM) Overview for Programmers。

---

## udfcheck 命令

### 用途

在 UDF 文件系统上执行一个文件系统检查。

### 语法

**udfcheck -d device [ -t tempfile ]**

### 描述

**udfcheck** 命令检查并修复指定设备上的 UDF 卷。

### 标志

**-d device**

指定 **udfcheck** 在其上检查并修复 UDF 卷的设备。

**-t tempfile**

指定 **udfcheck** 命令在其中存储执行文件系统检查所需信息的文件。

### 示例

1. 要在 **/dev/cd1** 设备上检查 UDF 文件系统的内容，输入以下内容：

```
udfcheck -d /dev/cd1
```



## 文件

`/usr/sbin/udfcheck`  
`/usr/lib/libudf.a`

包含 `udfcheck` 命令  
包含 `udfcheck` 命令调用的库例程

## 相关信息

『`udfcreate` 命令』和 第 524 页的『`udflabel` 命令』。

---

## udfcreate 命令

### 用途

创建 UDF 文件系统。

### 语法

```
udfcreate -d device [ -b bitmap_location ] [ -f formatType ]
```

### 描述

`udfcreate` 命令在指定设备上创建一个 UDF 文件系统，并用通用的集标识 (*setID*) 和卷名 (*volName*) 对其标号。

### 标志

**-b**

指定 *bitmap\_location*。可为以下数种之一：**b**、**e** 或 **m**。**b** 表示位图将位于分区的起始位置。**e** 表示位图将位于分区的结束位置。**m** 表示位图将位于分区的中间位置。位图的缺省位置是分区的开始位置。

**-d** *device*

指定要在其上创建 UDF 卷的设备。

**-f**

表示在介质上所要求的 UDF 版本。格式类型 1 代表 UDF V1.5，2 代表 UDF V2.0，而 3 代表 UDF V2.01。缺省版本是 UDF 1.5。

### 示例

1. 要在设备 `/dev/cd1` 上创建一个新的 UDF 文件系统，请输入以下内容：

```
udfcreate -d /dev/cd1
```

## 文件

`/usr/sbin/udfcreate`  
`/usr/lib/libudf.a`

包含 `udfcreate` 命令  
包含由 `udfcreate` 命令调用的库例程

## 相关信息

第 524 页的『`udflabel` 命令』和 第 522 页的『`udfcheck` 命令』。

---

## udflabel 命令

### 用途

在 UDF 文件系统中取回和更改标号。

### 语法

```
udflabel -d device [ -l label ]
```

### 描述

**udflabel** 命令显示或更改 UDF 卷名。如果未提供标号，该命令将会显示设备上的当前 UDF 卷名。如果提供了标号，则该命令将设备上的当前 UDF 卷名设置为新标号。

### 标志

|                         |                 |
|-------------------------|-----------------|
| <b>-d</b> <i>device</i> | 指定包含 UDF 卷的设备。  |
| <b>-l</b> <i>label</i>  | 在当前 UDF 卷上设置标号。 |

### 示例

1. 要更改设备 **/dev/cd1 hello** 上的当前标号，输入以下内容：

```
udflabel -d /dev/cd1 -l hello
```

2. 要显示 **device /dev/cd1** 上的当前标号，输入以下内容：

```
udflabel -d /dev/cd1
```

### 文件

|                           |                              |
|---------------------------|------------------------------|
| <b>/usr/sbin/udflabel</b> | 包含 <b>udflabel</b> 命令        |
| <b>/usr/lib/libudf.a</b>  | 包含由 <b>udflabel</b> 命令调用的库例程 |

### 相关信息

第 523 页的『udfcreate 命令』和 第 522 页的『udfcheck 命令』。

---

## uil 命令

### 用途

启动 AIXwindows 系统的“用户界面语言”（UIL）编译器。

### 语法

```
uil [ -lPathName ] InputFile [ -m ] [ -o FileName ] [ -s ] [ -v FileName ] [ -w ] [ -wmd FileName ]
```

## 描述

**uil** 命令调用 UIL 编译器。UIL 是一种规范语言，描述 AIXwindows 应用程序的用户界面的初始状态。该规范描述了界面中使用的对象（菜单、对话框、标签、按钮及其他），指定了当用户交互作用致使界面更改状态时所要调用的函数。

## 标志

|                             |                                                                                                                           |
|-----------------------------|---------------------------------------------------------------------------------------------------------------------------|
| <b>-I</b> <i>PathName</i>   | 指定包含 <i>PathName</i> ，中间无空格。如果缺省路径中未找到包含文件，使编译器在指定的目录中寻找包含文件。（大写 i）                                                       |
| <b>-m</b>                   | 指定列出的机器代码。该命令指导编译器将添加到“用户界面定义”（UID）中的记录描述放入列表文件。这将帮助您隔离错误。缺省值为无机器代码。                                                      |
| <b>-o</b> <i>FileName</i>   | 指导编译器生成 UID。缺省情况下，UIL 创建一个名为 <b>a.uid</b> 的 UID。该文件指定了 UID 文件名。如果编译器发出任何归类为错误或严重错误的诊断信息，则不会生成 UID。                        |
| <b>-s</b>                   | 指导编译器在编译任何文件前设置语言环境。语言环境以依靠实现的方式进行设置。在基于 C 的 ANSI 系统上，通常调用 <b>setlocale (LC_ALL, "")</b> 函数来设置语言环境。如果未指定该选项，则编译器不会设置语言环境。 |
| <b>-v</b> <i>Filename</i>   | 指导编译器生成列表。该文件为列表指定文件名。如果未出现 <b>-v</b> 选项，则编译器不会生成任何列表。缺省值为无列表。                                                            |
| <b>-w</b>                   | 指定编译器禁止所有的警告消息和信息性消息。如果未出现该选项，则无论严重性如何，所有的消息都会生成。                                                                         |
| <b>-wmd</b> <i>FileName</i> | 指定要使用的二进制窗口小部件元语言（WML）描述文件，替代缺省的 WML 描述。                                                                                  |

## 示例

要启动 UIL 编译器，输入：

```
uil -I. -o ex.uid ex.uil
```

## 退出状态

该命令返回以下退出值：

- 0** 指示成功完成。
- >0** 指示发生错误。

## 相关信息

**X** 命令。

---

## uimx 命令

### 用途

启动“X Window 系统”的 UIM/X 用户界面管理系统。

### 语法

```
uimx [ -dir Path ] [ -file FileName ] [ -workspace Name ] [ -xrm Options ]
```

## 描述

**uimx** 命令启动“X Window 系统”的 UIM/X 用户界面管理系统。它支持 Motif 1.2 并为开发图形用户界面 (GUI) 提供完整的编程环境。UIM/X 支持 C 和 C++ 两种面向对象编程语言。

UIM/X 保存并装入使用 Xt 资源语法的文本文件以描述界面和项目。它还能装入 UIL 文件。它生成 C、C++ 和 UIL 代码。它也可以为应用程序生成 makefile、消息编目和资源文件。

UIM/X 包含有内置“C 解释器”和以下工具及编辑器:

- “Motif 选项板”窗口小部件
- 浏览复杂窗口小部件层次结构的“窗口小部件浏览器”
- 绘制界面的 WYSIWYG 布局编辑器
- 设置窗口小部件属性初始值的“属性编辑器”；初始值可以是字面值或 C 语言表达式。
- 输入回调代码的“回调编辑器”
- 事件、操作和转换编辑器
- 菜单和主窗口编辑器
- 编辑用于界面的已生成代码的“声明编辑器”
- 编辑已生成的主程序和 makefile 的“程序布局编辑器”；该编辑器可使您直接访问主事件循环。

UIM/X 支持两种操作方式：“设计”和“测试”。在“测试”方式中，内置的“C 解释器”允许您测试应用程序的工作情况。在 Design 方式中，C 解释器验证用户输入到不同 UIM/X 编辑器中的代码。

UIM/X 提供方便函数库，简化了使用 X 和 Motif 编程的任务。

## 标志

|                              |                                                                                                  |
|------------------------------|--------------------------------------------------------------------------------------------------|
| <b>dir</b> <i>Path</i>       | 将 UIM/X 的当前目录设置为路径。                                                                              |
| <b>file</b> <i>FileName</i>  | 装入一个名为 <i>FileName</i> 的现有项目、界面或选项板文件。 <i>FileName</i> 可包含绝对路径名、当前目录的相对路径名或 <b>-dir</b> 值的相对路径名。 |
| <b>workspace</b> <i>Name</i> | 将 UIM/X 装入名为 <i>name</i> 的对应的 CDE 工作空间中。                                                         |
| <b>xrm</b> <i>Options</i>    | 使您能够输入任何资源规格 ( <i>options</i> )，否则就得在资源文件中输入。                                                    |

## 安全性

访问控制: 任何用户

访问的文件: 无

## 示例

要启动 UIM/X，输入:

```
uimx
```

## 文件

**/usr/uimx2.8/bin/uimx** 包含 **uimx** 命令。

---

## ul 命令

### 用途

执行加下划线。

### 语法

```
ul [ -i ] [ -t Terminal ] [ File ... ]
```

### 描述

**ul** 命令读取由 *File* 参数（或如果未给定文件则由标准输入）指定的命名文件，并将出现的下划线转换为序列，该序列表示为使用中的终端加下划线，如同由 **TERM** 环境变量所指定的那样。

### 标志

- i** 使 **ul** 命令通过一个包含相应的 `_`（下划线字符）的独立行来表示加下划线。使用这个方法可在 CRT 终端上的 **nroff** 命令输出流中看到出现下划线。
- t Terminal** 重设在环境中指定的终端类型。读取 **terminfo** 文件以确定加下划线的相应序列。如果终端不能加下划线，但是具有突出方式，则使用突出方式来代替。如果终端可以叠印或自动加下划线，则 **ul** 命令的作用与 **cat** 命令相同，并在屏幕上显示。如果终端不能加下划线，并且无其他替代方式可用，则将忽略加下划线。

如未指定 **-t** 标志，**ul** 命令为由 **TERM** 环境变量指定的终端类型进行转换。如果 *Terminal* 变量的值不是有效的终端类型，**ul** 命令为哑终端进行转换。

### 文件

**/usr/share/lib/terminfo/\*** 包含终端能力数据库。

### 相关信息

**cat** 命令、**colcrt** 命令、**man** 命令和 **nroff** 命令。

**terminfo** 文件。

---

## ulimit 命令

### 用途

设置或报告用户资源极限。

### 语法

```
ulimit [ -H ] [ -S ] [ -a ] [ -c ] [ -d ] [ -f ] [ -m ] [ -n ] [ -s ] [ -t ] [ Limit ]
```

### 描述

**ulimit** 命令设置或报告用户进程资源极限，如 **/etc/security/limits** 文件所定义。文件包含以下缺省值极限：

```
fsize = 2097151
core = 2097151
cpu = -1
data = 262144
rss = 65536
stack = 65536
nofiles = 2000
```

当新用户添加到系统中时，这些值被作为缺省值使用。当向系统中添加用户时，以上值通过 **mkuser** 命令设置，或通过 **chuser** 命令更改。

极限分为软性或硬性。通过 **ulimit** 命令，用户可将软极限更改到硬极限的最大设置值。要更改资源硬极限，必须拥有 **root** 用户权限。

很多系统不包括以上一种或数种极限。特定资源的极限在指定 *Limit* 参数时设定。*Limit* 参数的值可以是每个资源中指定单元中的数字，或者为值 **unlimited**。要将特定的 **ulimit** 设置为 **unlimited**，可使用词 **unlimited**。

**注：**在 **/etc/security/limits** 文件中设置缺省极限就是设置了系统宽度极限，而不仅仅是创建用户时用户所需的极限。

省略 *Limit* 参数时，将会打印出当前资源极限。除非用户指定 **-H** 标志，否则打印出软极限。当用户指定一个以上资源时，极限名称和单元在值之前打印。如果未给予选项，则假定带有了 **-f** 标志。

由于 **ulimit** 命令影响当前 **shell** 环境，所以它将作为 **shell** 常规内置命令提供。如果在独立的命令执行环境中调用该命令，则不影响调用者环境的文件大小极限。以下示例中正是这种情况：

```
nohup ulimit -f 10000
env ulimit 10000
```

一旦通过进程减少了硬极限，若无 **root** 特权则无法增加，即使返回到原值也不可能。

关于用户和系统资源极限的更多信息，请参见 *AIX 5L Version 5.3 Technical Reference: Base Operating System and Extensions Volume 1* 中的 **getrlimit**、**setrlimit** 或 **vlimit** 子例程。

## 标志

- a** 列出所有当前资源极限。
- c** 以 512 字节块为单位，指定核心转储的大小。
- d** 以 K 字节为单位指定数据区域的大小。
- f** 使用 *Limit* 参数时设定文件大小极限（以块计），或者在未指定参数时报告文件大小极限。缺省值为 **-f** 标志。
- H** 指定设置某个给定资源的硬极限。如果用户拥有 **root** 用户权限，可以增大硬极限。任何用户均可减少硬极限。
- m** 以 K 字节为单位指定物理存储器的大小。
- n** 指定一个进程可以拥有的文件描述符的数量的极限。
- s** 以 K 字节为单位指定堆栈的大小。
- S** 指定为给定的资源设置软极限。软极限可增大到硬极限的值。如果 **-H** 和 **-S** 标志均未指定，极限适用于以上二者。
- t** 指定每个进程所使用的秒数。

## 退出状态

返回以下退出值：

- 0** 成功完成。
- >0** 拒绝对更高的极限的请求，或发生错误。

## 示例

要将文件大小极限设置为 51,200 字节，输入：

```
ulimit -f 100
```

## 文件

`/usr/bin/ksh` 包含 `ulimit` 内置命令。

## 相关信息

`ksh` 命令。

*AIX 5L Version 5.3 Technical Reference: Base Operating System and Extensions Volume 1* 中的 `ulimit` 子例程、`getrlimit`、`setrlimit` 或 `vlimit` 子例程。

---

## umask 命令

### 用途

显示或设置文件方式创建掩码。

### 语法

```
umask [ -S ] [ Mask ]
```

### 描述

如果未指定 `Mask` 参数，`umask` 命令会将当前 shell 环境的文件方式创建掩码显示为标准输出。如果使用三位八进制数字或符号代码指定了 `Mask` 参数，`umask` 命令将设置当前 shell 执行环境的文件创建掩码。文件创建掩码中设置的位数用于清除创建文件时应用程序或命令所请求的相应位数。

`chmod` 命令描述了如何使用符号和数字代码来设置权限。

`-S` 标志产生符号输出。如果未指定该标志，缺省的输出格式为八进制。

如果在子外壳程序或独立命令执行环境中调用 `/usr/bin/umask` 命令，则其不会影响调用者环境的文件方式创建掩码。以下示例中正是这种情况：

```
(umask 002)
nohup umask ...
find . -exec umask ... \;
```

### 标志

`-S` 产生符号输出。

### 退出状态

返回以下退出值：

**0** 更改文件方式创建掩码成功，或未提供任何 `Mask` 参数。

>0 发生错误。

## 示例

1. 要设置方式掩码以便清除后继创建的文件清除其 **S\_IWOTH** 位，输入：

```
umask a=rx,ug+w
```

或

```
umask 002
```

设置了方式掩码后，通过输入以下命令来显示方式掩码的当前值：

```
umask
```

屏幕显示以下值：

```
02
```

2. 要生成符号输出，输入：

```
umask -S
```

屏幕显示以下值：

```
u=rwx,g=rwx,o=rx
```

3. 数字或符号输出都可在 **umask** 命令的后继调用中作为 *Mask* 参数使用。假设方式掩码如示例 2 所示那样设置。要设置方式掩码以便后继创建的文件清除其 **S\_IWGRP** 和 **S\_IWOTH** 位，输入：

```
umask g-w
```

4. 要设置方式掩码以便后继创建的文件清除其所有的写入位，输入：

```
umask -- -w
```

注：-r、-w 和 -x *Mask* 参数值（或任何以连字符开头的值）前必须带有 `--`（双连字符，中间无空格），以防止它被解释为选项。

## 文件

|                             |                                  |
|-----------------------------|----------------------------------|
| <code>/usr/bin/ksh</code>   | 包含 Korn shell <b>umask</b> 内置命令。 |
| <code>/usr/bin/umask</code> | 包含 <b>umask</b> 命令。              |

## 相关信息

**bsh** 命令、**chmod** 命令、**csh** 命令和 **ksh** 命令。

---

## umcode\_latest 命令

### 用途

识别其固件或微码可以从映像文件的指定源更新的系统资源。

### 语法

```
umcode_latest [-s source] [-l] [-A] | [-a[-q][-r]] -i | -h
```



## 描述

**umcode\_latest** 命令列出或下载的系统资源的固件或微码级别比为这些系统资源指定的源上的固件或微码级别低。

注：该命令不支持系统类型为 8842/8844/7047/7013/7015/7017 和 7025-F50 的系统固件映像。对于具有临时和永久系统固件映像的系统，**umcode\_latest** 命令使用临时系统固件映像与指定源上的映像对比。系统固件映像文件名称必须以 **.img** 结尾。

## 标志

|                  |                                                           |
|------------------|-----------------------------------------------------------|
| <b>-a</b>        | 更新源上具有更新微码的所有系统资源。                                        |
| <b>-A</b>        | 当源上的任何映像与当前列出或更新的映像不同时，将列出或更新资源。缺省情况下，每当源具有更新的映像时都将列出或更新。 |
| <b>-h</b>        | 提供扩展用途的帮助。                                                |
| <b>-i</b>        | 提供交互方式以提示每个需要更新的资源。                                       |
| <b>-l</b>        | 列出需要更新的系统资源。这是缺省值。                                        |
| <b>-q</b>        | 制止询问是否处理所有更新。                                             |
| <b>-r</b>        | 制止询问是否处理需要系统 IPL 的更新。                                     |
| <b>-s source</b> | 指向微码映像的源。缺省值是 <b>/etc/microcode</b> 。                     |

## 退出状态

|    |         |
|----|---------|
| 0  | 命令成功完成。 |
| >0 | 发生错误。   |

## 示例

- 要列出其固件或微码可以从 **/etc/microcode** 中的映像更新的所有系统资源，请输入：  

```
/usr/lpp/diagnostics/bin/umcode_latest
```
- 要列出其固件或微码可以从 **/tmp/fwupdate** 目录中的映像更新的所有系统资源，请输入：  

```
/usr/lpp/diagnostics/bin/umcode_latest -s /tmp/fwupdate
```
- 要列出其固件或微码可以从 **/tmp/fwupdate** 目录中的映像更新的所有系统资源，并在这时对每个资源询问是否更新该资源，请输入：  

```
/usr/lpp/diagnostics/bin/umcode_latest -s /fwupdate -i
```
- 要自动更新其固件或微码在 ISO 9660 格式 CD-ROM（已插入到 cd1 驱动器）中有更新映像的所有系统资源，请输入：  

```
/usr/lpp/diagnostics/bin/umcode_latest -s cd1 -a -q
```

## 限制

该命令不支持系统类型为 8842/8844/7047/7013/7015/7017 和 7025-F50 的系统固件映像。对于具有临时和永久系统固件映像的系统，**umcode\_latest** 命令使用临时系统固件映像与指定源上的映像对比。系统固件映像文件名称必须以 **.img** 结尾。

## 位置

**/usr/lpp/diagnostics/bin/umcode\_latest**

## 相关信息

**diag** 命令。

---

## umount 或 unmount 命令

### 用途

卸载已经安装的文件系统、目录或文件。

### 语法

```
{ umount | umount } [ -f ] [ -a ] | [ all | allr | Device | Directory | File | FileSystem | -n Node | -t Type ]
```

### 描述

**umount** 命令的另一个名字是**unmount**。任一名字均可使用。可以使用“系统管理界面工具”（SMIT）来运行该命令。要使用 SMIT，输入：

```
smit umount
```

**umount** 命令卸载已经安装的设备、目录、文件或文件系统。处理文件系统、目录或文件结束，命令即将其卸载。系统组成员和有 root 用户权限的用户可发出任何 **umount** 命令。只有有 root 用户权限的用户或系统组成员能卸载目录或文件。

注：SMIT 不会卸载 SMIT 帮助所在的 **/usr/lpp/info/\$LANG** 目录。典型的目录是 CD-ROM。

本地卸载时可指定其所在的设备、目录、文件或文件系统。

如所卸载的文件系统是 JFS2 快照，尽管快照仍然处于活动状态，**umount** 命令也会卸载该快照。就必须用 **snapshot** 命令删除该快照。

如果卸载的是安装有快照的被快照过的文件系统，**umount** 命令会显示存在装有快照警告，并退出而没有卸载文件系统。必须首先卸载快照。

注：如果 **cdromd** CD 和 DVD 自动安装守护程序已启用，则这些设备将如 **/etc/cdromd.conf** 文件所指定的那样被自动安装。使用 **cdumount** 或 **cdeject** 目录来卸载自动安装的 CD 或 DVD。使用“**stopsrc -s cdromd**”禁用 CD/DVD 自动安装守护程序。

### 标志

|                |                                                                                                                                                                                                                                           |
|----------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-a</b>      | 卸载所有安装的文件系统。                                                                                                                                                                                                                              |
| <b>all</b>     | 卸载所有安装的文件系统。                                                                                                                                                                                                                              |
| <b>allr</b>    | 卸载所有远程安装的文件系统。<br>注：对于远程安装，指定设备、目录、文件或文件系统参数。如果指定 <b>allr</b> 标志， <b>umount</b> 命令会卸载所有远程的安装。                                                                                                                                             |
| <b>-f</b>      | 在远程环境中进行强行卸载。当服务器当机、无法解析服务器路径名，或必须将仍在使用中的文件系统卸载时，可用于释放客户机。 <b>-f</b> 标志不支持日志文件系统。<br>注：对于远程文件系统，使用该标志导致文件系统上除了 <b>close()</b> 和 <b>unmap()</b> 外的所有文件操作均失败。任何由应用程序写入但是未传送到服务器的文件数据都将丢失。强制卸装 NFS V 4 文件系统还可能导致从相同服务器安装的其他文件系统中处于打开状态的文件丢失。 |
| <b>-n Node</b> | 指定要卸载的已安装目录占用的节点。 <b>umount -n Node</b> 命令卸载所有用 <i>Node</i> 参数进行的远程安装。                                                                                                                                                                    |

**-t** *Type*      卸载包含 **type=Type** 标志并已安装的 **/etc/filesystems** 文件中的所有节。*Type* 参数是字符串值，比如指定组名的远程值。

**注：**不能在正在使用的设备上使用 **umount** 命令。如果因任何原因打开文件或用户的当前目录在该设备上，该设备即为正在使用。

## 示例

1. 要从远程节点 Node A 卸载所有安装，输入：

```
umount -n nodeA
```

2. 要卸载特定类型的文件和目录，输入

```
umount -t test
```

这样便卸载了在含 **type=test** 属性的 **/etc/filesystems** 文件中有一个节的所有文件或目录。

## 文件

**/etc/filesystems**      列出已知文件系统并定义其特征。

## 相关信息

**cdcheck** 命令、**cdeject** 命令、**cdmount** 命令、**cdromd** 命令、**cdumount** 命令、**cdutil** 命令、**mount** 命令和 **fuser** 命令。

**mount** 子例程、**umount** 子例程、**vmount** 子例程、**uvmount** 子例程和 **mntctl** 子例程。

《操作系统与设备管理》中的『系统管理界面工具（SMIT）』说明了结构、主菜单和 SMIT 完成的任务。

《操作系统与设备管理》中的『安装』说明了安装文件和目录、安装点和自动安装。

---

## umountall 命令

### 用途

卸装成组不可卸装的设备或文件系统。

### 语法

```
umountall [ -k ] [ -s ] [ -F FileSystemType ] [ -l | -r ]
```

```
umountall [ -k ] [ -s ] [ -h Host ]
```

### 描述

缺省情况下，**umountall** 命令卸装除 **root**、**/proc**、**/var** 和 **/usr** 外的所有不可卸装的文件系统或设备。如果已指定了 *FileSystemType*，**umountall** 限制了对指定文件系统类型的操作。不保证 **umountall** 将卸装正在运行的文件系统，即使已指定了 **-k** 选项。

## 标志

**-F** *FileSystemType*

指定要卸载的文件系统类型。*FileSystemType* 对应由 `mount` 命令打印出的 `vfs` 栏。将卸载所有给定类型的不可卸载的文件系统。此标志不能与 **-h** 标志组合使用。

**-h** *Host*

指定主机节点。将卸载从这台主机远程加载的全部文件系统。

**-k**

卸载前向加载点上的每个进程发送一个 **SIGKILL**。此选项在内部用 `fuser -k` 命令杀死所有正在加载点运行的进程。由于该选项使得加载点上的每个进程都被杀死，所以该加载点的卸载并不会立刻进行。不保证 `umountall` 将卸载正在运行的文件系统，即使已指定了 **-k** 选项。只有杀死了使用加载点的全部进程后，才能试图卸载加载点。

**-l**

限制对本地文件系统的操作。

**-r**

限制对远程文件系统的操作。

**-s**

这是在序列化 `unmounts` 时为 System V 兼容性提供的非操作标志。`umount` 命令的序列化通过使用 **-k** 选项终止加载点上的全部相关进程来完成。

## 退出状态

**0** 命令成功完成。

**>0** 发生错误。

## 示例

1. 要卸载所有不可卸载的文件系统，输入：

```
umountall
```

2. 要卸载所有 **jfs** 类型的不可卸载的文件系统，输入：

```
umountall -F jfs
```

3. 要卸载所有从 `host.domain` 加载的不可卸载的文件系统，输入：

```
umountall -h host.domain
```

4. 要卸载所有远程加载的文件系统，输入：

```
umountall -r
```

## 文件

`/usr/sbin/umountall`

包含 `umountall` 命令。

## 相关信息

`umount` 命令。

---

## unalias 命令

### 用途

除去别名定义。

## 语法

**unalias** **-a**

**unalias** *AliasName* ...

## 描述

**unalias** 命令除去每个别名的指定的定义，或除去全部别名定义（如果使用了 **-a** 标志）。别名定义从当前 shell 环境中除去。

由于 **unalias** 命令影响当前的 shell 执行环境，所以它将作为 Korn shell 或 POSIX shell 常规内置命令提供。

## 标志

**-a** 从当前 shell 环境中除去全部别名定义。

## 退出状态

返回以下退出值：

- 0 成功完成。
- >0 指定的一个别名未代表一个有效的别名定义，或发生错误。

## 文件

**/usr/bin/ksh** 包含 Korn shell **unalias** 内置命令。  
**/usr/bin/unalias** 包含 **unalias** 命令。

## 相关信息

**alias** 命令、**csch** 命令、**ksh** 命令。

---

## uname 命令

### 用途

显示当前操作系统名称。

### 语法

**uname** [ **-a** | **-x** | **-S** *Name* ] [ **-F** ] [ **-f** ] [ **-l** ] [ **-L** ] [ **-m** ] [ **-M** ] [ **-n** ] [ **-p** ] [ **-r** ] [ **-s** ] [ **-T** *Name* ] [ **-u** ] [ **-v** ]

### 描述

**uname** 命令将正在使用的操作系统名写到标准输出中。

机器 ID 号码包括以下数字格式的 12 个字符 *xyyyyyymmss*。*xx* 位置指示系统且始终为 00。*yyyyyy* 位置包含整个系统的唯一标识号。*mm* 位置代表型号标识。*ss* 位置为子型号并且始终为 00。型号标识描述 CPU 平板标识，而不是描述整个系统的型号。

有时可使用 `uname -m` 命令决定使用什么型号。以下列表并非全面。参考硬件供应商提供的文档以获取 E0-FF 范围内的值。并且注意并非所有的机器类型都有机器 ID。许多新机器都共享 4C 的公共机器 ID。系统型号十六进制代码 (*mm*) 为:

| 机器类型 | 机器型号 | 机器标识    |
|------|------|---------|
| 7006 | 410  | 42      |
| 7007 | N40  | F0      |
| 7008 | M20  | 43      |
| 7008 | M20A | 43      |
| 7009 | C10  | 48      |
| 7011 | 220  | 41      |
| 7011 | 230  | 47      |
| 7011 | 250  | 46      |
| 7012 | 320  | 31      |
| 7012 | 320H | 35      |
| 7012 | 340  | 37      |
| 7012 | 350  | 38 或 77 |
| 7012 | 355  | 77      |
| 7012 | 360  | 76      |
| 7012 | 365  | 76      |
| 7012 | 370  | 75      |
| 7012 | 375  | 75      |
| 7012 | 380  | 58      |
| 7012 | 390  | 57      |
| 7012 | G30  | A6      |
| 7012 | G40  | A7      |
| 7013 | 520  | 30      |
| 7013 | 520H | 34      |
| 7013 | 530  | 10      |
| 7013 | 530H | 18      |
| 7013 | 540  | 14 或 11 |
| 7013 | 550  | 1C      |
| 7013 | 550L | 77      |
| 7013 | 560  | 5C      |
| 7013 | 570  | 67      |
| 7013 | 580  | 66      |
| 7013 | 58H  | 71      |
| 7013 | 590  | 70      |
| 7013 | 590H | 72      |
| 7013 | J30  | A0      |
| 7013 | J40  | A1      |
| 7015 | 930  | 20 或 02 |
| 7015 | 950  | 2E      |
| 7015 | 970  | 63      |
| 7015 | 970B | 63      |
| 7015 | 980  | 64      |
| 7015 | 980B | 64      |
| 7015 | 990  | 80      |
| 7015 | R10  | 67      |
| 7015 | R20  | 72      |
| 7015 | R24  | 81      |
| 7015 | R30  | A3      |
| 7015 | R40  | A4      |
| 7016 | 730  | 10      |
| 7018 | 740  | 30      |
| 7018 | 770  | 67      |
| 7024 | E20  | C0      |
| 7025 | F30  | C4      |
| 7030 | 3AT  | 58      |
| 7030 | 3BT  | 57      |

|      |     |    |
|------|-----|----|
| 7043 | 140 | 4C |
| 7043 | 240 | 4C |
| 7248 | 43P | 4C |

当安装了新的操作系统软件级别时，**uname** 命令返回的机器标识符值可能改变。这一改变影响使用该值访问许可程序的应用程序。需要查看标识符，输入 **uname -m** 命令。

如果应用程序受到影响，联系合适的支持组织。

## 标志

- a** 显示 **-m**、**-n**、**-r**、**-s** 和 **-v** 标志指定的所有信息。不能与 **-x** 或 **-SName** 标志连用。如果 **-x** 标志和 **-a** 标志一起指定，**-x** 标志会覆盖它。
- F** 显示由十六进制字符构成的系统标识字符串。此标识字符串对特定系统上的所有分区都是相同的。
- f** 除分区号还用于此字符串的计算之外，与 **F** 标志类似。产生的标识字符串对特定系统上的每个分区都是唯一的。
- l** 显示 LAN 网络号码。
- L** 显示 LPAR 号码和 LPAR 名称。如果 LPAR 不存在，“-1”显示为 LPAR 号码，**NULL** 为 LPAR 名称
- m** 显示硬件运行系统的机器 ID 号。  
**注：****-m** 标志不能为 LPAR 环境中的分区生成唯一的机器标识。
- M** 显示系统型号名称。如果型号名称属性不存在，显示空字符串
- n** 显示节点名称。可能是用以标识系统且为 UUCP 通信网络所知的名称。
- p** 显示系统处理器的体系结构。
- r** 显示操作系统的发行版号。
- s** 显示系统名。标志缺省为开。
- S Name** 设置节点名。这是系统的 UUCP 通信网络名。
- T Name** 设置系统名。这是系统的 UUCP 通信网络名。
- u** 显示系统 ID 号码。如果这一属性未被定义，输出与 **uname -m** 显示的输出一致。
- v** 显示操作系统版本。
- x** 显示 **-a** 标志指定的信息和 **-l** 标志指定的 LAN 网络号。

如果输入标志无效，**uname** 命令退出，生成错误消息、错误返回状态，并无输出。

**注：**经过系统重新引导后，“uname”命令不保存新系统名和节点名称值。

## 退出状态

该命令返回以下退出值：

- 0** 请求信息成功写入。
- >0** 发生错误。

## 示例

显示完整系统名和版本栏，输入：

```
uname -a
```

## 文件

**/usr/bin/uname** 包含 **uname** 命令。

## 相关信息

`uname` 或 `unamex` 子例程。

---

## uncompress 命令

### 用途

恢复压缩文件

### 语法

```
uncompress [ -c ][ -F ][ -f ][ -n ][ -q ][ -V ][ File ... ]
```

### 描述

`uncompress` 命令恢复由 `compress` 命令压缩的原文件。每个由 *File* 参数指定的压缩文件被除去而由展开后的副本所替换。展开后的文件与压缩的版本名字相同，但没有 `.Z` 扩展名。如果用户有 `root` 用户权限，展开后的文件保留与原文件相同的所有者、组、方式和修改时间。如果用户没有 `root` 用户权限，则文件保留相同的方式和修改时间，但获取新的所有者和组。如果没有指定文件，则标准输入将展开为标准输出。

### 标志

|                                   |                                                                                                          |
|-----------------------------------|----------------------------------------------------------------------------------------------------------|
| <code>-c</code>                   | 写到标准输出。没有文件被更改。                                                                                          |
| <code>-f</code> 或 <code>-F</code> | 强制展开。 <code>-f</code> 和 <code>-F</code> 标志是可互换的。如果文件已存在则将覆盖该文件。系统不会提示用户现有的文件将被覆盖。文件大小可能实际上并不会缩小。         |
| <code>-n</code>                   | 省略压缩文件的压缩文件头。<br><b>注：</b> 如果文件使用 <code>-n</code> 标志压缩，请使用该选项。否则，将不进行压缩文件。                               |
| <code>-q</code>                   | 禁止显示由 <code>-v</code> 标志生成的压缩统计信息。如果同一命令行中有几个 <code>-v</code> 和 <code>-q</code> 标志，则最后一个指定的标志将控制统计信息的显示。 |
| <code>-V</code>                   | 将当前版本和编译选项写到标准错误。                                                                                        |

### 参数

*File ...*                    指定要恢复的压缩文件。

### 返回值

如果发生以下事件，`uncompress` 命令检测到错误并以状态 1 退出。

- 输入文件不是由 `compress` 命令生成的。
- 输入文件无法读取或输出文件无法写入。

如果没有错误发生，退出状态为 0。

### 退出状态

- |    |       |
|----|-------|
| 0  | 成功完成。 |
| >0 | 发生错误。 |



## 示例

要解压 `foo.Z` 文件，输入：

```
uncompress foo.Z
```

`foo.Z` 文件被解压，并重命名为 `foo`。

## 相关信息

**compress** 命令、**dmpuncompress** 命令、**pack** 命令、**unpack** 命令和 **zcat** 命令。

《操作系统与设备管理》中的『命令概述』。

---

## undefvsd 命令

### 用途

**undefvsd** - 取消定义虚拟共享磁盘。

### 语法

```
undefvsd vsd_name ...
```

### 描述

该命令用来从所有虚拟共享磁盘节点上给定的 *vsd\_names* 的 **/dev** 处删除虚拟共享磁盘定义和任何特殊的设备文件。必须取消配置该虚拟共享磁盘并且在所有虚拟共享磁盘节点上它必须处于已定义状态。

可以使用“系统管理界面工具”（SMIT）来运行 **undefvsd** 命令。要使用 SMIT，输入：

```
smit delete_vsd
```

并选择 **Undefine a Virtual Shared Disk** 选项。

### 标志

无

### 参数

*vsd\_name* 指定虚拟共享磁盘，您不再要其底层的逻辑卷可以通过任何虚拟共享磁盘节点来访问。

### 安全

您必须具有 `root` 用户权限以运行该命令。

### 退出状态

**0** 说明命令已成功完成。

非零 表明发生错误。

## 限制

您必须从对等域中联机的节点发出该命令。要使对等域联机，请使用 **startprdomain** 命令。要使现有对等域中一个特定的节点联机，请使用 **startprnode** 命令。有关创建和管理 RSCT 对等域的更多信息，请参阅 *RSCT: Administration Guide*。

## 标准输出

当前 RVSD 子系统运行级别。

## 示例

要删除与虚拟共享磁盘 **vsd1vg2n1** 有关的信息，请输入：

```
undefvsd vsd1vg2n1
```

## 位置

**/usr/lpp/vsd/bin/undefvsd**

## 相关信息

命令：**defvsd**

---

## unexpand 命令

### 用途

写到带有恢复的制表符的标准输出。

### 语法

```
unexpand [ -a | -t TabList ] [ File ... ]
```

### 描述

**unexpand** 命令将制表符放回到来自标准输入或已命名文件的数据中，并将结果写到标准输出。缺省情况下，只有行距空格和制表符重新转换为最大制表符字符串。

注：*File* 参数必须是文本文件。

### 标志

**-a** 在任何有制表符通过替换两个或多个字符压缩了结果文件的地方插入制表符。  
**-t** *TabList* 指定制表符停止位的位置。制表符停止位的缺省值为 8 个列位置。

*TabList* 变量必须包括单个正十进制整数或多个正十进制整数。多个整数必须以升序排列，并用逗号或空格字符隔开，整数两边加引号。单精度 *TabList* 变量将制表符停止位设置为相等的间隔列位置数。多精度 *TabList* 变量将制表符停止位设置为与 *TabList* 变量中的整数相对应的列位置。

在多精度 *TabList* 变量中指定的最后一个位置之外，不会发生对于字符的空格到制表符的转换。

注：当指定了 **-t** 标志时，将忽略 **-a** 标志且转换不限于处理行距空格字符。

## 退出状态

该命令返回以下退出值:

- 0 命令成功运行。
- >0 发生错误。

## 示例

要在 **xyz** 文件中用制表符替换空格字符, 输入:

```
unexpand xyz
```

## 文件

**/usr/bin/unexpand** 包含 **unexpand** 命令。

## 相关信息

**expand** 命令、**newform** 命令、**tab** 命令和 **untab** 命令。

《操作系统与设备管理》中的『文件』。

《操作系统与设备管理》中的『输入和输出重定向概述』。

---

## unfencevsd 命令

### 用途

**unfencevsd** - 给在节点或节点组上运行的应用程序权限以访问虚拟共享磁盘或虚拟共享磁盘组, 先前这些磁盘或磁盘组是受防护的, 那些节点上运行的应用程序无法访问它们。

### 语法

```
unfencevsd {-a | -v vsd_name_list} {-n node_list [-f] }
```

### 描述

在某些环境下, 当一个节点实际上是正常运行的, 但是被切断了与其他运行同样应用程序的节点的通信时, 系统就可能认为该节点变得不可操作并可能开始恢复过程。在这种情况下, 只有直到恢复完成并且其他运行该应用程序的节点识别出这个有问题的节点是正常运行的, 才能允许该节点为对其通常管理的虚拟共享磁盘的请求提供服务。**fencevsd** 命令阻止有问题的节点满足对其虚拟共享磁盘的请求。**unfencevsd** 命令允许受防护的节点重新获得访问虚拟共享磁盘的权限。

您可以从对等域中联机的任何节点发出该命令。

### 标志

- a** 指定所有的虚拟共享磁盘。
- f** 允许受防护的节点取消自己的防护。
- n node\_list** 指定一个或多个节点号, 用逗号隔开。

**-v** *vsd\_name\_list*

指定一个或多个虚拟共享磁盘名称，用逗号隔开。

## 参数

无

## 安全

您必须具有 `root` 用户权限以运行该命令。

## 退出状态

**0** 说明命令已成功完成。

**非零** 表明发生错误。

## 限制

您必须从对等域中联机的节点发出该命令。要使对等域联机，请使用 `startprdomain` 命令。要使现有对等域中一个特定的节点联机，请使用 `startprnode` 命令。有关创建和管理 RSCT 对等域的更多信息，请参阅 *RSCT: Administration Guide*。

## 标准输出

当前 RVSD 子系统运行级别。

## 示例

1. 要对虚拟共享磁盘 `vsd1` 和 `vsd2` 取消防护节点 5，请输入：

```
unfencevsd -v vsd1,vsd2 -n 5
```

2. 当 `unfencevsd` 命令必须从节点 7 输入时，要对虚拟共享磁盘 `vsd1` 和 `vsd2` 取消防护节点 7，请输入：

```
unfencevsd -v vsd1,vsd2 -n 7 -f
```

## 位置

`/opt/rsct/vsd/bin/unfencevsd`

## 相关信息

命令：`fencevsd`、`lsvsd`

有关“可恢复的”虚拟共享磁盘子系统的更多信息和如何可以使用 `fencevsd`、`unfencevsd` 和 `lsfensevsd` 命令以在应用程序恢复过程中保持数据完整性，请参考 *RSCT: Managing Shared Disks*。

---

## unget 命令 (SCCS)

### 用途

取消先前的 `get` 命令。

### 语法

```
unget [ -rSID ] [ -s ] [ -n ] File ...
```

## 描述

**unget** 命令允许您在创建新的变化量前恢复通过 **get -e** 创建的 g-file。所有更改均因此而废弃。如果为 *File* 的值指定 - (短划线)，则读取标准输入，且标准输入的每一行解释为 SCCS 文件的名字。文件结束符终止输入。

如果为 *File* 值指定目录，则 **unget** 命令对当前处于编辑过程中的所有 SCCS 文件（这些文件带有 **s.** 的前缀）进行所请求的操作。

一旦对某个文件运行 **unget** 命令，必须重新发出 **get -e** 命令以更改该文件。**unget** 命令自动删除 g-file。

## 标志

每个或每组标志独立应用于每个已命名的文件。

- n** 阻止自动删除 g-file。该标志允许您保留文件编辑后的版本而不带变化量。
- rSID** 指定下一次使用 **delta** 命令将创建的新的变化量。如果同一个登录名下的文件有两个或更多暂挂变化量，则必须用该标志。可以看一下 p-file 以查看在同一登录名下是否有一个以上的变化量暂挂在一个特定的 SID 上。SID 规范必须明白地指定仅废弃一个 SID，否则 **unget** 命令显示错误消息并停止运行。
- s** 禁止显示已删除的 SID。

## 退出状态

该命令返回以下退出值：

- 0** 成功完成。
- >0** 发生错误。

## 示例

要在运行 **get -e** 命令后废弃对 SCCS 文件所作的更改，输入：

```
unget s.prog.c
```

## 文件

**/usr/bin/unget** 包含到 SCCS **unget** 命令的路径。

## 相关信息

**delta** 命令、**get** 命令和 **sact** 命令。

**sccsfile** 文件格式。

《AIX 5L V5.3 通用编程概念：编写并调试程序》中的『SCCS 命令列表』。

《AIX 5L V5.3 通用编程概念：编写并调试程序》中的『源代码控制系统（SCCS）概述』。

---

## unifdef 命令

### 用途

从文件中除去 **ifdef** 行。

## 语法

**unifdef** [ **-t** ] [ **-l** ] [ **-c** ] [ **-DSymbol** ] [ **-USymbol** ] [ **-idSymbol** ] [ **-iuSymbol** ] [ *File* ]

## 描述

**unifdef** 命令用于除去文件中的 **ifdef** 行，而不会进行别的操作。为了正确运行，**unifdef** 命令认出嵌套的 **ifdefs**、注释、C 语言语法的单引号和双引号，但不包括文件或不解释宏。**unifdef** 命令认出但并不除去注释。

如果没有指定 *File*，**unifdef** 命令从标准输入得到输入，并复制输出到标准输出。

一旦指定 *Symbol*，**ifdef** 中的行最终被复制到输出或除去。与符号关联的 **ifdef**、**ifndef**、**else**、**elif** 和 **endif** 行也要除去。涉及未指定的 **ifdef** 不会更改并和相关 **ifdef**、**else**、**elif** 和 **endif** 行一起拷出。如果相同符号在一个以上的自变量里出现，仅第一个出现的符号有意义。例如，如果 **ifdef X** 嵌套在另一个 **ifdef X** 内，内部的 **ifdef** 即被认为是无法识别的符号。

当使用 **ifdef** 定界非 C 语言行，如注释或未完成的代码时，有必要指定为此目的使用哪些符号。否则，**unifdef** 命令将试图分析 **ifdef** 行的引用和注释。

**unifdef** 命令不能处理 **cpp** 构造，例如：

```
#if defined(X) || defined(Y)
```

或

```
#elif X
```

或

```
#elif defined(X) || defined(Y)
```

## 关键字

**unifdef** 命令识别以下关键字：

- **ifdef**
- **ifndef**
- **else**
- **endif**
- **elif**

## 标志

|                   |                                                                          |
|-------------------|--------------------------------------------------------------------------|
| <b>-c</b>         | 补充 <b>unifdef</b> 命令的操作。即保留被除去的行，反之亦然。                                   |
| <b>-D Symbol</b>  | 指定定义的符号。                                                                 |
| <i>File</i>       | 指定输入源。                                                                   |
| <b>-id Symbol</b> | <b>unifdef</b> 命令不会试图识别指定的 <b>ifdef</b> 内的注释、单引号或双引号，但这些行将被拷出。           |
| <b>-iu Symbol</b> | <b>unifdef</b> 命令不会试图识别指定的 <b>ifdef</b> 内的注释、单引号或双引号。这些行不会拷出。            |
| <b>-l</b>         | 用空行替换除去的行，而不是将其删除。                                                       |
| <b>-t</b>         | 允许 <b>unifdef</b> 命令用于纯文本（不是 C 语言代码）： <b>unifdef</b> 命令不会试图识别注释、单引号和双引号。 |
| <b>-U Symbol</b>  | 指定要取消定义的符号。                                                              |

## 退出状态

该命令返回以下退出值:

- 0 输出为输入的精确副本。
- 1 输出不是输入的精确副本。
- 2 命令因为 EOF 过早发生、不相称的 **else**、**elif** 或 **endif** 而失败。

## 示例

1. 示例如下:

```
unifdef -DA original.c > modified.c
```

使 **unifdef** 命令阅读 `original.c` 文件, 并除去 `#ifdef A` 行。然后除去与 `#ifdef A` 相关的在 `#elif/#else` 后的所有行, 一直到 `#endif`。输出放在 `modified.c` 文件中。

2. 示例如下:

```
unifdef -UA original.c > modified.c
```

使 **unifdef** 命令读 `original.c` 文件, 并除去 `#ifdef A`, 一直到其与其关联的 `#elif/#else` 或 `#endif`。对于 `#elif`, `#elif` 由 `#if` 替换。对于 `#else`, `#else` 同与其相关的 `#endif` 一同删除。输出放在 `modified.c` 文件中。

## 文件

`/usr/bin/unifdef` 包含 **unifdef** 命令。

## 相关信息

**cpp** 命令。

《操作系统与设备管理》中的『命令』。

---

## uniq 命令

### 用途

报告或删除文件中重复的行。

### 语法

```
uniq [ -c | -d | -u ] [ -f Fields ] [ -s Characters ] [ -Fields ] [ +Characters ] [ InFile [ OutFile ] ]
```

### 描述

**uniq** 命令删除文件中的重复行。**uniq** 命令读取由 *InFile* 参数指定的标准输入或文件。该命令首先比较相邻的行, 然后除去第二行和该行的后续副本。重复的行一定相邻。(在发出 **uniq** 命令之前, 请使用 **sort** 命令使所有重复行相邻。)最后, **uniq** 命令将最终单独的行写入标准输出或由 *OutFile* 参数指定的文件。*InFile* 和 *OutFile* 参数必须指定不同的文件。

输入文件必须是文本文件。文本文件是包含组织在一行或多行中的字符的文件。这些行的长度不能超出 2048 个字节(包含所有换行字符), 并且其中不能包含空字符。

缺省情况下，**uniq** 命令比较所有行。如果指定了 **-f Fields** 或 **-Fields** 标志，**uniq** 命令忽略由 *Fields* 变量指定的字段数目。*field* 是一个字符串，用一个或多个 <空格> 字符将它与其他字符串分隔开。如果指定了 **-s Characters** 或 **-Characters** 标志，**uniq** 命令忽略由 *Characters* 变量指定的字段数目。为 *Fields* 和 *Characters* 变量指定的值必须是正的十进制整数。

当前本地语言环境决定了 **-f** 标志使用的 <空白> 字符以及 **-s** 标志如何将字节解释成字符。

如果执行成功，**uniq** 命令退出，返回值 0。否则，命令退出返回值大于 0。

## 标志

|                      |                                                                                                                                                                                                                                                   |
|----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-c</b>            | 在输出行前面加上每行在输入文件中出现的次数。                                                                                                                                                                                                                            |
| <b>-d</b>            | 仅显示重复行。                                                                                                                                                                                                                                           |
| <b>-f Fields</b>     | 忽略由 <i>Fields</i> 变量指定的字段数目。如果 <i>Fields</i> 变量的值超过输入行中的字段数目， <b>uniq</b> 命令用空字符串进行比较。这个标志和 <b>-Fields</b> 标志是等价的。                                                                                                                                |
| <b>-u</b>            | 仅显示不重复的行。                                                                                                                                                                                                                                         |
| <b>-s Characters</b> | 忽略由 <i>Characters</i> 变量指定的字符的数目。如果 <i>Characters</i> 变量的值超过输入行中的字符的数目， <b>uniq</b> 用空字符串进行比较。如果同时指定 <b>-f</b> 和 <b>-s</b> 标志， <b>uniq</b> 命令忽略由 <b>-s Characters</b> 标志指定的字符的数目，而从由 <b>-f Fields</b> 标志指定的字段后开始。这个标志和 <b>+Characters</b> 标志是等价的。 |
| <b>-Fields</b>       | 忽略由 <i>Fields</i> 变量指定的字段数目。这个标志和 <b>-f Fields</b> 标志是等价的。                                                                                                                                                                                        |
| <b>+Characters</b>   | 忽略由 <i>Characters</i> 变量指定的字符的数目。如果同时指定 <b>-Fields</b> 和 <b>+Characters</b> 标志， <b>uniq</b> 命令忽略由 <b>+Characters</b> 标志指定的字符数目，并从由 <b>-Fields</b> 标志指定的字段后开始。这个标志和 <b>-s Characters</b> 标志是等价的。                                                   |

## 退出状态

该命令返回以下退出值：

|    |         |
|----|---------|
| 0  | 命令成功运行。 |
| >0 | 发生错误。   |

## 示例

要删除名为 *fruit* 文件中的重复行并将其保存到一个名为 *newfruit* 的文件中，输入：

```
uniq fruit newfruit
```

如果 *fruit* 文件包含下列行：

```
apples
apples
peaches
pears
bananas
cherries
cherries
```

则在您运行**uniq** 命令后 *newfruit* 文件将包含下列行：

```
apples
peaches
pears
bananas
cherries
```



# 文件

`/usr/bin/uniq` 包含 `uniq` 命令。

## 相关信息

`comm` 命令, `sort` 命令。

---

## units 命令

### 用途

将一种计量单位转换为另一种等效的计量单位。

### 语法

`units [ - ] [ File ]`

### 描述

`units` 命令将一种计量单位表示的数量转换为另一种计量单位的等效值。`units` 命令是一个交互式命令。它提示要转换自的单位和转换到的单位。该命令只进行倍增刻度变化。即, 只有需要进行乘法转换时命令才能将一个值转换为另一个值。例如, 该命令不能进行华氏和摄氏温度间的转换, 因为这种转换中必须加上或减去值 32。

可以指定一个数量作为相乘的单位组合, 可选择一个数字乘数置前。

输入下标正整数表示幂, / (斜线) 表示除法。

`units` 命令将 `lb` 识别为质量单位, 但将 `pound` 识别为英镑。复合名称共同起作用 (例如 `lightyear`)。前缀英制单位与带 `br` (例如, `brgallon`) 美制对应单位不同。

`/usr/share/lib/unittab` 文件包含 `units` 命令使用的单位的完整列表。在此文件中可以定义新的单位。`File` 参数可用于覆盖 `/usr/share/lib/unittab` 文件所列标准转换因子的值。指定文件必须与 `unittab` 文件格式保持一致。

`units` 命令能识别包括下列单位在内的最常见的单位、缩写和公制前缀:

|                    |                     |
|--------------------|---------------------|
| <code>pi</code>    | 圆周率                 |
| <code>c</code>     | 光速                  |
| <code>e</code>     | 单位电子电荷              |
| <code>g</code>     | 重力加速度               |
| <code>force</code> | 与 <code>g</code> 相同 |
| <code>mole</code>  | 阿伏加德罗常数             |
| <code>water</code> | 每单位高度的水的压力差         |
| <code>au</code>    | 天文单位                |

### 标志

- 在提示输入转换前列出 `/usr/share/lib/unittab` 文件中包含的转换因子。

## 示例

1. 显示英寸到厘米的转换因子，输入：

```
units
you have: in
you want: cm
```

**units** 命令返回以下值：

```
* 2.540000e+00
/ 3.937008e-01
```

输出说明用英寸乘以 2.540000e+00 得到厘米，用厘米乘以 3.937008e-01 得到英寸。

数字为标准指数计数制，因此 3.937008e-01 意为  $3.937008 \times 10^{-1}$ ，与 0.3937008 相同。

**注：**第二个数始终是第一个数的倒数；例如，2.54 等于  $1/0.3937008$ 。

2. 同一计量方式转换为不同单位，输入：

```
units
you have: 5 years
you want: microsec
```

**units** 命令返回以下值：

```
* 1.577846e+14
/ 6.337753e-15
```

输出显示 5 years 等于  $1.577846 \times 10^{14}$  毫秒，一毫秒等于  $6.337753 \times 10^{-15}$  年。

3. 在计量中使用分数，输入：

```
units
you have: 1|3 mi
you want: km
```

**units** 命令返回以下值：

```
* 5.364480e-01
/ 1.864114e+00
```

“|”（竖线）表示分数线，因此 1|3 意指三分之一。这表示三分之一英里与 0.536448 公里相等。

4. 在计量中包括指数，输入：

```
units
you have: 1.2-5 gal
you want: floz
```

**units** 命令返回以下值：

```
* 1.536000e-03
/ 6.510417e+02
```

表达式 1.2-5 gal 与  $1.2 \times 10^{-5}$  gal 相同。在指数前不要输入 e（即，1.2e-5 gal 不正确）。示例表明  $1.2 \times 10^{-5}$  (0.000012) 加仑等于  $1.536 \times 10^{-3}$  (0.001536) 液量盎司。

5. 指定复数单位，输入：

```
units
you have: gram centimeter/second2
you want: kg-m/sec2
```

**units** 命令返回以下值：

```
* 1.000000e-05
/ 1.000000e+05
```

单位 gram centimeter/second2 意为“克 x 厘米/秒<sup>2</sup>。”同样，kg-m/sec2 意为“千克 x 米/秒<sup>2</sup>”，常读为“千克米每平方秒。”

6. 如果在 you have: 和 you want: 后指定的单位矛盾:

```
you have: ft
you want: lb
```

**units** 命令返回以下消息和值:

```
conformability
3.048000e-01 m
4.535924e-01 kg
```

conformability 消息意为指定的单位不能转换。英尺是长度单位，磅为质量单位，所以转换无意义。所以，**units** 命令显示标准单位中的同一值。

换句话说，示例显示一英尺等于 0.3048 米，一磅等于 0.4535924 千克。**units** 命令显示米制和千克制的等价数，因为这些都是标准长度和质量计量单位。

## 文件

**/usr/bin/units**  
**/usr/share/lib/unittab**

包含 **units** 命令。  
列出 **units** 命令创建和用户定义的单位。

## 相关信息

**bc** 命令和 **dc** 命令。

---

## unlink 命令

### 用途

执行 **unlink** 子例程。

### 语法

**unlink** *File*

### 描述

**unlink** 命令在指定的文件上执行 **unlink** 子例程。

当关联的子例程失败时，**unlink** 命令不发出错误消息；必须检查退出值以确定该命令是否已正常完成。如果成功，则返回值为 0；如果指定参数过少或过多，则返回值为 1；如果系统调用失败，则返回值为 2。

**注：****unlink** 命令允许有 root 用户权限的用户处理非常规的问题，例如将整个目录移到目录树的另一部分。该命令也允许您创建无法到达或退出的目录。请遵守以下规则小心保持目录结构:

- 保证每个目录都有一个 . (点) 与其自身链接。
- 保证每个目录都有一个 .. (点点) 与其父目录链接。
- 保证每个目录与其自身或其父目录的链接只有一个。

- 保证每个目录可从其文件系统 root 访问。

尝试除去某个已导出供 NFS V4 服务器使用的文件或目录将会失败，并出现一条消息，指示资源正忙。必须取消供 NFS V4 使用的目录和文件的导出，这样才能除去它。

## 示例

要删除由 file2 指向的目录项，输入：

```
unlink file2
```

## 文件

`/usr/sbin/unlink` 包含 **unlink** 命令。

## 相关信息

**fsck** 命令、**link** 命令和 **ln** 命令。

**link** 子例程、**unlink** 子例程。

《操作系统与设备管理》中的『文件系统』说明了文件系统类型、管理、结构以及维护。

《操作系统与设备管理》中的『文件』提供了使用文件的信息。

《操作系统与设备管理》中的『目录』说明了如何使用目录和路径名称。

---

## unloadipsec 命令

### 用途

从“IP 安全性”子系统中卸装加密模块。

### 语法

```
unloadipsec -c crypto_mod_name
```

### 描述

**unloadipsec** 命令从“IP 安全性”子系统中卸装加密模块。当不再使用加密模块或加密模块被更新的版本替换时，可以使用**unloadipsec** 命令。

只有“IP 安全性”设备停止后，才能卸装加密模块。替换加密模块的步骤为：将“IP 安全性”设备更改为已定义的状态；用此命令卸装旧的加密模块；卸载旧的加密模块并安装新的模块，将“IP 安全性”设备恢复到可用的状态。

### 标志

**-c** *crypto\_mod\_name*

指定要卸装的加密模块的名称。当使用该命令而不带标志时，命令列出所有已安装的加密模块（但不一定已装入）。

---

## unmirrorvg 命令

### 用途

除去卷组或指定磁盘上存在的镜像。该命令仅适用于 AIX 4.2.1 或更新的版本。

### 语法

```
unmirrorvg [ -c Copies ] VolumeGroup [ PhysicalVolume .. ]
```

### 描述

**unmirrorvg** 命令在指定的卷组上取消所有检测到的逻辑卷的镜像。如果对卷组中的每个单独的逻辑卷执行 **rmlvcopy** 命令，也可以手工完成同样的功能。

缺省情况下，**unmirrorvg** 将从已镜像的卷组中选出要除去的镜像集。如果希望某个驱动器不再包含镜像，您必须在输入参数 *PhysicalVolume* 中包含磁盘列表。

当 *PhysicalVolume* 参数在命令中列出时，这表示仅应除去这块磁盘上存在的镜像。卷组中存在于别的驱动器上，但是没在用户提供的磁盘列表中列出的镜像将保留不变，保持镜像状态。

#### 注:

1. 如果 LVM 未识别出某个磁盘已发生故障，则 LVM 可能会除去另一个镜像。因此，如果知道某个磁盘发生了故障，而 LVM 没有显示磁盘失踪，您应在命令行中指定故障的磁盘，或者用 **replacepv** 更换该磁盘，或者用 **reducevg** 除去该磁盘。
2. 如果逻辑卷副本范围超过一个磁盘，驻留在用户未列出的磁盘上的那部分逻辑卷副本也会被除去。
3. 快照卷组上不允许执行 **unmirrorvg** 命令。

当执行 **unmirrorvg** 时，每个逻辑卷的 COPIES 缺省值变为 1。如果希望将卷组由三重镜像转换为两重镜像，请使用 **-c** 选项。

**注:** 要使用该命令，您必须具有 root 用户权限，或是 **system** 组的成员。

**警告:** 由于错误检查的复杂性和卷组中要取消镜像的逻辑卷的数目，**unmirrorvg** 命令可能要花相当长的时间才能完成。

可以使用“系统管理界面工具”（SMIT）来运行该命令。要使用 SMIT，输入：

```
smit unmirrorvg
```

### 标志

**-c *Copies*** 指定 **unmirrorvg** 命令完成执行后每个逻辑卷必须具有的最小副本数目。如果不要所有的逻辑卷都具有相同数目的副本，那么请用 **rmlvcopy** 命令来手工减少镜像。如果未使用该选项，则副本数目的缺省值为 1。

以下是对 **rootvg** 的描述:

#### **rootvg unmirroring**

当完成了 **rootvg** 取消镜像时, 必须执行三个附加的任务: **bosboot**、**bootlist** 和 **reboot**。

需要 **bosboot** 命令重新初始化剩余磁盘上的引导记录。需要执行 **bootlist** 命令, 以便系统仅引导 **rootvg** 中剩余的磁盘。因为取消镜像会使 **rootvg** 的定额恢复, 所以需要重新引导以使该操作生效。

## 示例

1. 要取消三重镜像卷组的镜像并保留两个副本, 输入:

```
unmirrorvg -c 2 workvg
```

在 **workvg** 上保留的逻辑卷中的逻辑分区现在具有 2 个副本。

2. 要以缺省值对 **rootvg** 取消镜像, 输入:

```
unmirrorvg rootvg
```

**rootvg** 现在只有 1 个副本。

3. 要更换已镜像卷组中的坏磁盘驱动器, 输入:

```
unmirrorvg workvg hdisk7
reducevg workvg hdisk7
rmdev -l hdisk7 -d
replace the disk drive, let the drive be renamed hdisk7
extendvg workvg hdisk7
mirrorvg workvg
```

**注:** 在此示例中, 缺省情况下 **mirrorvg** 将尝试在 **workvg** 中创建 2 个逻辑卷副本。它将尝试在更换好的磁盘驱动器上创建新的镜像。但是, 如果原系统已进行了三重镜像, 则可能不会在 **hdisk7** 上创建新的镜像, 因为该逻辑卷的其他副本可能都已存在。这和 **unmirrorvg** 将镜像副本数目减少为 1 的缺省值操作相似。

**注:** 当 **unmirrorvg workvg hdisk7** 运行时, **hdisk7** 将成为卷组中的剩余驱动器。该驱动器实际上没有从卷组除去。必须运行 **migratepv** 命令来将数据从将要从系统中除去的磁盘移动到磁盘 **hdisk7**。

## 文件

**/usr/sbin** **unmirrorvg** 命令驻留的目录。

## 相关信息

**migratepv** 命令、**mkivcopy** 命令、**mirrorvg** 命令、**syncvg** 命令、**reducevg** 命令和 **extendvg** 命令。

《操作系统与设备管理》中的『逻辑卷存储』。

---

## unpack 命令

### 用途

展开文件

## 语法

**unpack** *File* ...

## 描述

**unpack** 命令展开由 **pack** 命令创建的文件。对每个指定的文件，**unpack** 命令搜索名为 *File.z* 的文件。如果这个文件是压缩文件，则 **unpack** 命令用其展开的版本来替换它。**unpack** 命令通过从 *File* 除去 **.z** 后缀来给新的文件命名。如果用户有 root 用户权限，则新的文件具有和原文件一样的访问方式、访问和修改时间、所有者和组。如果用户没有 root 用户权限，则该文件保留同样的访问方式、访问时间和修改时间，但获取新的所有者和组。

**unpack** 命令仅对以 **.z** 结束的文件进行操作。结果，当指定不以 **.z** 结束的文件名时，**unpack** 添加该后缀并搜索目录以寻找带有这种后缀的文件名。

退出值是 **unpack** 命令不能解压的文件数。如果发生以下情况则文件不能解压：

- 文件名（除 **.z** 文件之外）多于 253 字节。
- 文件不能打开。
- 文件不是压缩文件。
- 有展开文件名的文件已存在。
- 不能创建被解压的文件。

**注：**如果正在解压的文件有链接，**unpack** 命令会将警告写到标准错误。与其来源的压缩文件比较，新解压的文件有不同的索引节点。但是，任何其他链接到压缩文件原索引节点的文件仍然存在，仍然处于压缩状态。

## 退出状态

该命令返回以下退出值：

- 0** 命令成功运行。
- >0** 发生错误。

## 示例

要解压压缩文件：

```
unpack chap1.z chap2
```

这样就展开了压缩文件 *chap1.z* 和 *chap2.z*，并用名为 *chap1* 和 *chap2* 的文件来替换它们。请注意给 **unpack** 命令的文件名可以带有或不带 **.z** 后缀。

## 文件

**/usr/bin/unpack** 包含 **unpack** 命令。

## 相关信息

**cat** 命令、**compress** 命令和 **pack** 命令。

《操作系统与设备管理》中的『文件』。

---

## untab 命令

### 用途

将制表符改为空格

### 语法

**untab** [ *FileName* ... ]

### 描述

**untab** 命令读取由 *FileName* 参数或标准输入指定的文件，并用空格字符替换输入中的制表符。如果用 *FileName* 参数指定文件，**untab** 命令将结果文件写回成原文件。如果输入为标准输入，**untab** 命令写成标准输出。**untab** 命令假定制表符停止位设置在每隔八列处，并从第九列开始。为 *FileName* 参数指定的文件名长度不能超过 **PATH\_MAX**-9 个字节。

### 示例

要用空格字符替换 File 文件中的制表符，输入：

```
untab File
```

### 文件

**/usr/bin/untab** 包含 **untab** 命令。

### 相关信息

**expand** 命令、**newform** 命令、**tab** 命令和 **unexpand** 命令。

《操作系统与设备管理》中的『文件』。

《操作系统与设备管理》中的『输入和输出重定向』。

---

## update 命令

### 用途

定期更新超级块。

### 语法

**update**

### 描述

**update** 命令每 30 秒就执行 **sync** 子例程。该操作确保在系统崩溃的情况下文件系统是最新的。

### 文件

**/usr/sbin/update** 包含 **update** 命令。



## 相关信息

**init** 命令、**rc** 命令和 **sync** 命令。

**cron** 守护程序。

**sync** 子例程。

---

## updatevsdnode 命令

### 用途

修改虚拟共享磁盘子系统选项。

### 语法

```
updatevsdnode      -n {ALL | node_number [,node_number ...]}
                   {[-a VSD_adapter]
                   [-b min_buddy_buffer_size]
                   [-x max_buddy_buffer_size]
                   [-s max_buddy_buffers]
                   [-M vsd_max_ip_packet_size]}
                   [-f] [-c cluster_name | NONE]
```

### 描述

使用 **updatevsdnode** 可修改虚拟共享磁盘子系统选项。

**注：** 该命令只修改子系统的选项。要有效地配置虚拟共享磁盘，您首先必须取消配置所有的虚拟共享磁盘，卸载设备驱动程序，然后重新配置这些共享磁盘。

可以使用“系统管理界面工具”（SMIT）来运行该命令。要使用 SMIT，输入：

```
smit vsd_mgmt
```

并选择 **Set/Show Virtual Shared Disk Device Driver Operational Parameters** 选项或 **Update virtual shared disk Device Driver Node Parameters** 选项。

### 标志

- n** 指定节点的数目（您要该命令去更新这些节点的信息），或 **RSCT** 对等域中的所有节点。可以发出命令 **/usr/bin/lscfg** 以查找您正在运行的节点的节点号。
- a** 指定用于虚拟共享磁盘与这个或这些节点的通信的适配器名称。您必须指定 **m10** 作为适配器名称。
- b** 指定服务器用来满足对虚拟共享磁盘的远程请求的最小伙伴缓冲区。该值必须是 2 乘方并且大于或等于 4096。IBM 建议使用值 4096（4KB）。
- x** 服务器将用来满足远程请求的最大伙伴缓冲区。该值必须是 2 乘方并且大于或等于 *min\_buddy\_buffer\_size*。推荐值为 262144（256KB）。该值在 **RSCT** 对等域中的所有节点上必须都是一样的。

- s** 这是要分配的 *max\_buddy\_buffer\_size* 缓冲区的数目。当首次装入时，虚拟共享磁盘的设备驱动程序将具有缓冲区初始的大小，然后将根据需要动态地分配和收回附加的空间。对于 32 位内核，建议的起始值为 128 个 256KB 的缓冲区。推荐值为 2000 个 256KB 的缓冲区。
- 伙伴缓冲区仅在服务器上使用。在客户机节点上，您可能要将 *max\_buddy\_buffers* 设置为 1。
- 注：** **statvsd** 命令将指示远程请求是否在排队等待伙伴缓冲区。
- M** 指定虚拟共享磁盘的最大消息大小（以字节为单位）。该值不能大于网络的最大传输单元（MTU）大小。建议的值为：
- 对于交换机，是 61440（60KB）
  - 对于巨型帧以太网，是 8192（8KB）
  - 对于 1500 字节 MTU 的以太网，是 1024（1KB）
- f** 指定该命令将通过重新配置 RSCT 对等域中所有节点（当前已在这些节点上配置虚拟共享磁盘）上的一个或多个虚拟共享磁盘来强制更新虚拟共享磁盘子系统选项。
- c cluster\_name | NONE**
- 更改节点所属的集群。NONE 会将节点从集群中除去。
- 注：** *cluster\_name* 只有对于 SSA（串行存储器体系结构）磁盘才是必需的。

## 参数

*vsd\_name* 指定虚拟共享磁盘，您不再要其底层的逻辑卷可以通过任何虚拟共享磁盘节点来访问。

## 安全

您必须具有 **root** 权限才能运行该命令。

## 退出状态

**0** 说明命令已成功完成。

**非零** 表明发生错误。

## 限制

您必须从对等域中联机的节点发出该命令。要使对等域联机，请使用 **startprdomain** 命令。要使现有对等域中一个特定的节点联机，请使用 **startprnode** 命令。关于创建和管理 RSCT 对等域的更多信息，请参阅 *RSCT Administration Guide*。

## 示例

要在节点 3 上将伙伴缓冲区大小增加到 48 个最大的伙伴缓冲区，请输入：

```
updatevsdnode -n 3 -s 48
```

**注：** 必须从内核取消配置设备驱动程序并重新装入以使该更改生效。

## 位置

**/opt/lpp/vsd/bin/updatevsdnode**

## 相关信息

命令: **lsvsd**、**vsdata1st**、**vsdnode**

---

## updatevsdtab 命令

### 用途

**updatevsdtab** - 更改虚拟共享磁盘子系统属性。

### 语法

**updatevsdtab** **{-v vsd\_names | -a}** **{[-s ]}** **[-f]**

### 描述

使用该命令来更新虚拟共享磁盘的大小。当您使用 **updatevsdtab** 命令来更改虚拟共享磁盘的大小后，直到取消配置并再次配置该虚拟共享磁盘时更改才会生效。

如果指定了 **-f** 标志，则将在所有正常运行的节点上重新配置涉及的虚拟共享磁盘，起初即在这些节点上配置过这些虚拟共享磁盘。

可以使用“系统管理界面工具”（SMIT）来运行该命令。要使用 SMIT，输入：

```
smit vsd_mgmt
```

并选择 Set/Show virtual shared disk Device Driver Operational Parameters 选项或 Update virtual shared disk Options 选项。

### 标志

- v vsd\_names** 指定一系列要更新的虚拟共享磁盘名称。
- a** 指定系统或系统分区的所有节点上要更改的选项。
- s** 在有关的逻辑卷大小更改后更新虚拟共享磁盘的大小。
- f** 通过重新配置当前系统分区中所有节点上的虚拟共享磁盘来强制进行更改，虚拟共享磁盘就在节点上配置过。

### 参数

无

### 安全

您必须具有 **root** 用户权限以运行该命令。

### 退出状态

- 0** 说明命令已成功完成。
- 非零** 表明发生错误。

## 限制

您必须从对等域中联机的节点发出该命令。要使对等域联机，请使用 **startprdomain** 命令。要使现有对等域中一个特定的节点联机，请使用 **startprnode** 命令。有关创建和管理 RSCT 对等域的更多信息，请参阅 *RSCT: Administration Guide*。

## 标准输出

当前 RVSD 子系统运行级别。

## 示例

1. 要重新设置名为 USER1n3 的虚拟共享磁盘的大小，请输入：

```
updatevsdtab -v USER1n3 -s
```

## 位置

**/usr/lpp/csd/bin/updatevsdtab**

## 相关信息

命令：**defvsd**, **updatevsdnode**

---

## updatevsdvg 命令

### 用途

更改虚拟共享磁盘全局卷组的特性。

### 语法

```
updatevsdvg { -a | -g global_volgrp { -k VSD -p primary_node -b secondary_node | -k CVSD -l server_list [-c cluster_name] } }
```

### 描述

**updatevsdvg** 命令更改虚拟共享磁盘全局卷组的特性。该命令允许您将全局卷组从并发虚拟共享磁盘卷组更改为串行存取（或非并发）虚拟共享磁盘卷组，以及相反方向的更改。可以在服务器节点数目发生更改的任何时候（比如替换或重新接线服务器使得新的服务器数目不同，或者当您需要删除一台服务器时）使用该命令。

该命令执行以下操作：

1. 暂挂作为该卷组一部分的所有虚拟共享磁盘
2. 停止作为该卷组一部分的所有虚拟共享磁盘
3. 对该卷组发出 **varyoffvg** 命令
4. 验证卷组在新的服务器上存在，如果它不存在，则尝试导入该卷组
5. 更新全局卷组的特性
6. 为卷组发出 **varyonvg** 命令到相应的服务器
7. 启动作为该卷组一部分的所有虚拟共享磁盘

注:

1. 如果您发出此命令并带有 **-a** 标志, 则可恢复的虚拟共享磁盘子系统不应该是活动的子系统。否则, 可以在可恢复的虚拟共享磁盘子系统活动的时候运行该命令, 只要没有应用程序在使用作为正在更新的卷组一部分的虚拟共享磁盘。
2. 对于实现 AIX SCSI 设备驱动程序 of SCSI-3 持久保留型的磁盘以及 SSA (串行存储器体系结构) 磁盘, 并发虚拟共享磁盘是受支持的。

## 标志

- a** 指定对于该节点所服务的所有 VSD 卷组, 应该在对象数据管理器 (ODM) 中重新建立持久保留信息。此标志是在允许多个 `cluster` 访问相同的虚拟共享磁盘的初始设置阶段使用的。它还可以用于在无意中除去了设备 ODM 条目之后进行恢复。
- 此标志将引起由该节点服务的所有卷组发生脱机改变。卷组将在此节点以及卷组的所有其他服务器上脱机改变。因此, 在发出带有此标志的 `updatevsdvg` 命令之前, 您应该停止可恢复的虚拟共享磁盘子系统。
- b secondary\_node** 指定辅助节点。
- c cluster\_name** 为服务器节点指定集群名, 这些节点将为并行存取的共享磁盘服务。该标志仅适用于 SSA (串行存储器体系结构) 磁盘, 并且必须为 SSA 指定一个 `cluster_name`。
- g global\_volgrp** 指定现有的全局卷组名。
- k VSD | CVSD** 指定卷组将是并发虚拟共享磁盘类型还是串行存取 (非并发) 虚拟共享磁盘类型。
- l server\_list** 为并发虚拟共享磁盘指定一系列服务器, 用冒号分隔。
- p primary\_node** 指定主节点。

## 参数

`vsd_name` 指定虚拟共享磁盘, 您不再要其底层的逻辑卷可以通过任何虚拟共享磁盘节点来访问。

## 安全性

您必须具有 **root** 权限才能运行该命令。

## 退出状态

- 0** 说明命令已成功完成。
- 非零** 表明发生错误。

## 限制

您必须从对等域中联机的节点发出该命令。要使对等域联机, 请使用 `startprdomain` 命令。要使现有对等域中一个特定的节点联机, 请使用 `startprnode` 命令。关于创建和管理 RSCT 对等域的更多信息, 请参阅 *RSCT Administration Guide*。

## 标准输出

当前 RVSD 子系统运行级别。

## 示例

1. 要将名为 **ess\_gvg** 的全局卷组从虚拟共享磁盘全局卷组更改为具有三台服务器的并发全局卷组，假定这些磁盘都已正确接线且磁盘子系统支持持久保留（比如 ESS 磁盘），请输入：

```
updatevsdvg -g ess_gvg -k CVSD -l 9:17:21
```

2. 要从名为 **ssa\_gvg** 的 SSA 全局卷组中除去一台服务器，其中全局卷组的原始服务器列表是 **9:10** 且属于名为 **cluster9\_10** 的 SSA 集群（即命令 **vsdata1st -c** 显示 SSA 集群信息），请输入：

```
updatevsdvg -g ssa_gvg -k CVSD -l 9 -c cluster9_10
```

3. 要将名为 **ess\_gvg** 的并发全局卷组更改回虚拟共享磁盘全局卷组，其中原始的服务器列表是 **9:17:21**，新的主节点号是 9，新的辅助节点号是 21，请输入：

```
updatevsdvg -g ess_gvg -k VSD -p 9 -b 21
```

## 位置

`/opt/rsct/vsd/bin/updatevsdvg`

## 相关信息

命令：**vsdata1st**、**vsdvg**

---

## uprintfd 守护程序

### 用途

构造并写内核消息

### 语法

**uprintfd**

### 描述

**uprintfd** 守护程序检索、转换、格式化内核消息并将其写到进程控制终端。内核消息通过 **NLuprintf** 和 **uprintf** 内核服务提交。因为 **uprintfd** 守护程序从不退出，所以只能运行一次。

## 相关信息

**NLuprintf** 内核服务和 **uprintf** 内核服务。

《AIX 5L V5.3 通用编程概念：编写并调试程序》中的『输入和输出处理程序员概述』。

---

## uptime 命令

### 用途

显示系统正常运行多少时间。

### 语法

**uptime**

## 描述

**uptime** 命令显示当前时间、系统正常运行的时间长度、联机用户数目和平均负载。负载平均值是以 1 分钟、5 分钟、15 分钟时间间隔开头的可运行的进程。**uptime** 命令的输出实质上就是 **w** 命令提供的标题行。

## 相关信息

**ruptime** 命令和 **w** 命令。

---

## useradd 命令

### 用途

创建新的用户帐户。

### 语法

```
useradd [ -c comment ] [ -d dir ] [ -e expire ] [ -g group ] [ -G group1,group2 ... ] [ -m [ -k skel_dir ] ] [ -u uid ] [ -s shell ] [ -r role1,role2 ... ] login
```

### 描述

**useradd** 命令创建新的用户帐户。*login* 参数必须是唯一的字符串（可以由管理员使用 **chdev** 命令配置该参数的长度）。您不能在用户名中使用 **ALL** 或 **default** 关键字。

**useradd** 命令不会为用户创建密码信息。它使用星号 (\*) 初始化 **password** 字段。稍后，将使用 **passwd** 或 **pwdadm** 命令设置该字段。在使用 **passwd** 或 **pwdadm** 命令将认证信息添加到 **/etc/security/passwd** 文件之前，新帐户被禁用。

**useradd** 命令总是检查目标用户注册表以确保新帐户的标识对于目标注册表是唯一的。还可以使用 **dist\_uniqid** 系统属性配置 **useradd** 命令检查系统的所有用户注册表。**dist\_uniqid** 系统属性是 **/etc/security/login.cfg** 文件中 **usw** 节的一个属性，可以使用 **chsec** 命令进行管理。

**dist\_uniqid** 系统属性具有以下值：

**never** 不检查非目标注册表的标识冲突。这是缺省设置。

**always**

检查所有其他注册表的标识冲突。如果检测到目标注册表和任何其他注册表间存在冲突，则帐户创建或修改失败。

**uniqbyname**

检查所有其他注册表的标识冲突。仅当要创建的帐户与现有帐户同名时才允许注册表间存在冲突。

注：不论 **dist\_uniqid** 系统属性如何，始终强制执行目标注册表中的标识冲突检测。

**uniqbyname** 系统属性设置在两个注册表中正常工作。如果有两个以上的注册表并且两个注册表之间已经存在标识冲突，则在使用冲突的标识值的第三个注册表中创建新帐户时未指定 **useradd** 命令的行为。根据检查注册表的顺序，创建新帐户可能成功，也可能失败。

对标识冲突的检查只是为了强制在本地注册表和远程注册表之间或两个远程注册表之间标识唯一。在远程注册表中新创建的帐户和使用同一远程注册表的其他系统中现有的本地用户之间，不保证标识唯一。如果命令运行时无法访问远程注册表，则 **useradd** 命令将绕过远程注册表。

## 标志

|                                    |                                                                                                                                                                                                                                                |
|------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-c</b> <i>comment</i>           | 提供有关 <i>login</i> 参数指定的用户的一般信息。 <i>comment</i> 参数是一个没有嵌入冒号 (:) 字符并且不能以字符 “#!” 结束的字符串。                                                                                                                                                          |
| <b>-d</b> <i>dir</i>               | 标识 <i>login</i> 参数指定的用户的主目录。 <i>dir</i> 参数是完整路径名。                                                                                                                                                                                              |
| <b>-e</b> <i>expire</i>            | 标识帐户的截止日期。 <i>expire</i> 参数是一个以 <i>MMDDhhmmyy</i> 格式表示的 10 个字符的字符串，其中 <i>MM</i> 是月， <i>DD</i> 是天， <i>hh</i> 是小时， <i>mm</i> 是分钟， <i>yy</i> 是从 1939 年到 2038 年的最后 2 位数字。所有字符都是数字。如果 <i>expire</i> 参数为 0，则该帐户未过期。缺省值是 0。请参阅 <b>date</b> 命令以获取更多信息。 |
| <b>-g</b> <i>group</i>             | 标识用户的主组。 <i>group</i> 参数必须包含有效的组名并且不能是空值。                                                                                                                                                                                                      |
| <b>-G</b> <i>group1,group2,...</i> | 标识用户所属的组。 <i>group1,group2,...</i> 参数是使用逗号分隔的组名列表。                                                                                                                                                                                             |
| <b>-k</b> <i>skel_dir</i>          | 将缺省文件从 <i>skel_dir</i> 复制到用户的主目录。仅与 <b>-m</b> 标志一起使用。                                                                                                                                                                                          |
| <b>-m</b>                          | 如果用户的主目录不存在，则建立一个。缺省情况下不建立主目录。                                                                                                                                                                                                                 |
| <b>-r</b> <i>role1,role2,...</i>   | 列出该用户的管理角色。 <i>role1,role2,...</i> 参数是一个使用逗号分隔的角色名称列表。                                                                                                                                                                                         |
| <b>-s</b> <i>shell</i>             | 定义在会话初始化时为用户运行的程序。 <i>shell</i> 参数是完整路径名。                                                                                                                                                                                                      |
| <b>-u</b> <i>uid</i>               | 指定用户标识。 <i>uid</i> 参数是一个唯一的整数字符串。避免更改该属性，以免破坏系统安全性。                                                                                                                                                                                            |

## 退出状态

|    |         |
|----|---------|
| 0  | 命令成功完成。 |
| >0 | 发生错误。   |

## 示例

1. 要使用缺省值创建 *davis* 用户帐户，请输入：

```
useradd davis
```

## 限制

要防止登录不一致，请避免全用大写字母字符组成用户名。虽然 **useradd** 命令支持多字节用户名，但是限制用户名只能使用 POSIX 可移植文件名字符集中的字符。

要确保您的用户数据库保持不受损坏，您在命名用户时必须小心。用户名不能以连字符 (-)、加号 (+)、at 符号 (@) 或波浪号 (~) 开始。不能在用户名中使用关键字 **ALL** 或 **default**。另外，不要在用户名字符串中使用以下任一字符：

|   |     |
|---|-----|
| : | 冒号  |
| " | 双引号 |
| # | 镑符号 |
| , | 逗号  |
| = | 等于号 |



|   |     |
|---|-----|
| \ | 反斜杠 |
| / | 斜杠  |
| ? | 问号  |
| ' | 单引号 |
| ` | 右引号 |

最后, *login* 参数不能包含任何空格、制表符或换行符。

## 位置

**/usr/sbin/useradd**

## 文件

**useradd** 命令具有对以下文件的读写权限。

|                                   |                 |
|-----------------------------------|-----------------|
| <b>/etc/passwd</b>                | 包含用户的基本属性。      |
| <b>/etc/security/user</b>         | 包含用户的扩展属性。      |
| <b>/etc/security/user.roles</b>   | 包含用户的管理角色属性。    |
| <b>/etc/security/limits</b>       | 定义每个用户的资源配额和限制。 |
| <b>/etc/security/environ</b>      | 包含用户的环境属性。      |
| <b>/etc/security/audit/config</b> | 包含审计配置信息。       |
| <b>/etc/security/lastlog</b>      | 包含用户最后一次登录的属性。  |
| <b>/etc/group</b>                 | 包含组的基本属性。       |
| <b>/etc/security/group</b>        | 包含组的扩展属性。       |

## 相关信息

*chfn* 命令、*chgroup* 命令、*chgrpmem* 命令、*chsh* 命令、*lsgroup* 命令、*lsuser* 命令、*mkgroup* 命令、*mkuser* 命令、*passwd* 命令、*pwdadm* 命令、*rmgroup* 命令、*rmuser* 命令、第 69 页的『*setgroups* 命令』、第 73 页的『*setsenv* 命令』、第 228 页的『*su* 命令』、『*userdel* 命令』和第 564 页的『*usermod* 命令』。

---

## userdel 命令

### 用途

除去用户帐户。

### 语法

**userdel** [ **-r** ] *login*

### 描述

**userdel** 命令除去 *login* 参数标识的用户帐户。缺省情况下, 该命令除去用户的属性但是不除去用户的主目录。用户名必须已存在。如果指定了 **-r** 标志, 则 **userdel** 命令还将除去用户的主目录。

只有具有 **UserAdmin** 权限的一个或多个 **root** 用户可以除去管理用户。管理用户是那些在 **/etc/security/user** 文件中设置了 **admin=true** 的用户。

## 标志

**-r** 除去用户的主目录。

## 退出状态

**0** 命令成功完成。  
**>0** 发生错误。

## 示例

1. 要从本地系统中除去用户 `davis` 帐户及其属性，请输入：

```
userdel davis
```

## 位置

`/usr/sbin/userdel`

## 文件

**userdel** 命令具有对以下文件的读写权限。

|                                         |                 |
|-----------------------------------------|-----------------|
| <code>/etc/passwd</code>                | 包含用户的基本属性。      |
| <code>/etc/security/user</code>         | 包含用户的扩展属性。      |
| <code>/etc/security/user.roles</code>   | 包含用户的管理角色属性。    |
| <code>/etc/security/limits</code>       | 定义每个用户的资源配额和限制。 |
| <code>/etc/security/environ</code>      | 包含用户的环境属性。      |
| <code>/etc/security/audit/config</code> | 包含审计配置信息。       |
| <code>/etc/security/lastlog</code>      | 包含用户最后一次登录的属性。  |
| <code>/etc/group</code>                 | 包含组的基本属性。       |
| <code>/etc/security/group</code>        | 包含组的扩展属性。       |

## 相关信息

`chfn` 命令、`chgroup` 命令、`chgrpmmem` 命令、`chsh` 命令、`lsgroup` 命令、`lsuser` 命令、`mkgroup` 命令、`mkuser` 命令、`passwd` 命令、`pwdadm` 命令、`rmgroup` 命令、`rmuser` 命令、第 69 页的『`setgroups` 命令』、第 73 页的『`setsenv` 命令』、第 228 页的『`su` 命令』、第 561 页的『`useradd` 命令』和『`usermod` 命令』。

---

## usermod 命令

### 用途

更改用户属性。

### 语法

```
usermod [ -u uid ] [ -G group1,group2 ... ] [ -d dir [ -m ] ] [ -s shell ] [ -c comment ] [ -l new_name ]  
[ -e expire ] [ -R role1,role2 ... ] login
```

## 描述

**警告:** 如果系统中安装了网络信息服务 (NIS) 数据库, 则不要使用 **usermod** 命令。

**usermod** 命令更改 *login* 参数标识的用户的属性。用户名必须已存在。要更改属性, 请指定标志和新的值。以下文件包含该命令设置的本地用户属性:

- **/etc/passwd**
- **/etc/security/environ**
- **/etc/security/limits**
- **/etc/security/user**
- **/etc/security/user.roles**
- **/etc/security/audit/config**
- **/etc/group**
- **/etc/security/group**

避免更改帐户的标识以免损害系统安全性。然而, 当使用 **usermod** 命令更改标识时, 标识冲突检查还由 **/etc/security/login.cfg** 文件中 **usw** 一节的 **dist\_uniqid** 属性控制。标识冲突控制的行为与 **mkuser** 命令中的描述相同。

## 标志

|                                    |                                                                                                                                                                                                                                                  |
|------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-c</b> <i>comment</i>           | 提供有关 <i>login</i> 参数指定的用户的一般信息。 <i>comment</i> 参数是一个没有嵌入冒号 (:) 字符并且不能以字符 “#!” 结束的字符串。                                                                                                                                                            |
| <b>-d</b> <i>dir</i>               | 将主目录更改为 <i>dir</i> 参数中指定的目录。                                                                                                                                                                                                                     |
| <b>-e</b> <i>expire</i>            | 标识帐户的截止日期。 <i>expire</i> 参数是一个以 <i>MMDDhhmmyy</i> 格式表示的 10 个字符的字符串, 其中 <i>MM</i> 是月, <i>DD</i> 是天, <i>hh</i> 是小时, <i>mm</i> 是分钟, <i>yy</i> 是从 1939 年到 2038 年的最后 2 位数字。所有字符都是数字。如果 <i>expire</i> 参数为 0, 则该帐户未过期。缺省值是 0。请参阅 <b>date</b> 命令以获取更多信息。 |
| <b>-G</b> <i>group1,group2,...</i> | 标识用户所属的组。 <i>group1,group2,...</i> 参数是使用逗号分隔的组名列表。                                                                                                                                                                                               |
| <b>-l</b> <i>new_name</i>          | 指定用户的新名称。                                                                                                                                                                                                                                        |
| <b>-m</b>                          | 将用户当前主目录中的内容移动到新主目录。只能与 <b>-d</b> 标志一起使用。                                                                                                                                                                                                        |
| <b>-R</b> <i>role1,role2,...</i>   | 列出该用户的管理角色。 <i>role1,role2,...</i> 参数是一个使用逗号分隔的角色名称列表。                                                                                                                                                                                           |
| <b>-s</b> <i>shell</i>             | 定义在会话初始化时为用户运行的程序。 <i>shell</i> 参数是完整路径名。                                                                                                                                                                                                        |
| <b>-u</b> <i>uid</i>               | 指定用户标识。 <i>uid</i> 参数是一个唯一的整数字符串。避免更改该属性, 以免破坏系统安全性。                                                                                                                                                                                             |

## 退出状态

|    |         |
|----|---------|
| 0  | 命令成功完成。 |
| >0 | 发生错误。   |

## 示例

1. 要将用户 `davis` 更改为 `system` 组的成员，请输入：

```
usermod -g system davis
```

## 限制

要确保用户信息的完整性，使用 `usermod` 命令时将应用某些限制。只有具有 **UserAdmin** 权限的一个或多个 `root` 用户可以使用 **usermod** 命令执行以下任务：

- 通过将 `admin` 属性设置为 `true` 使用户成为管理用户。
- 更改管理用户的所有属性。
- 将用户添加到管理组

管理组是 `admin` 属性集设置为 `True` 的组。安全组的成员可以更改非管理用户的属性和将用户添加到非管理组。

**usermod** 命令仅操作本地用户数据。不能使用它来更改注册表服务器（如 `NIS` 和 `DCE`）中的数据。

## 位置

`/usr/sbin/usermod`

## 文件

**usermod** 命令具有对以下文件的读写权限。

|                                         |                 |
|-----------------------------------------|-----------------|
| <code>/etc/passwd</code>                | 包含用户的基本属性。      |
| <code>/etc/security/user</code>         | 包含用户的扩展属性。      |
| <code>/etc/security/user.roles</code>   | 包含用户的管理角色属性。    |
| <code>/etc/security/limits</code>       | 定义每个用户的资源配额和限制。 |
| <code>/etc/security/environ</code>      | 包含用户的环境属性。      |
| <code>/etc/security/audit/config</code> | 包含审计配置信息。       |
| <code>/etc/security/lastlog</code>      | 包含用户最后一次登录的属性。  |
| <code>/etc/group</code>                 | 包含组的基本属性。       |
| <code>/etc/security/group</code>        | 包含组的扩展属性。       |

## 相关信息

`chfn` 命令、`chgroup` 命令、`chgrpmem` 命令、`chsh` 命令、`lsgroup` 命令、`lsuser` 命令、`mkgroup` 命令、`mkuser` 命令、`passwd` 命令、`pwdadm` 命令、`rmgroup` 命令、`rmuser` 命令、第 69 页的『`setgroups` 命令』、第 73 页的『`setsenv` 命令』、第 228 页的『`su` 命令』、第 561 页的『`useradd` 命令』和第 563 页的『`userdel` 命令』。

---

## users 命令

### 用途

显示当前在系统上的用户的简要列表。

### 语法

**users** [ *File* ]

## 描述

**users** 命令以简要的一行列表格式将当前在系统上的用户的登录名列为标准输出 (**stdout**)。如果给定了自变量, 其将用作备用文件而不是 **/etc/utmp**。

## 文件

**/etc/utmp** 包含当前用户的列表。  
**/usr/bin/users** 包含 **users** 命令。

## 相关信息

**who** 命令。

---

## usrck 命令

### 用途

验证用户定义的正确性。

### 语法

```
usrck { -l [ -b ] | -n | -p | -t | -y } { ALL | User ... }
```

### 描述

**usrck** 命令通过检查所有用户或者由 *User* 参数定义的用户定义, 验证用户数据库文件里用户定义的正确性。如果指定的用户超过一个, 名字间必须有一个空格。必须选择一个标志来指示系统是否应该尝试修复错误属性。

命令首先检查**/etc/passwd** 文件里的条目。如果指示系统应该修复错误, 用户重名情况被报告并且删除重名。重复的标识符仅被报告, 因为没有系统修复。如果条目有少于六个独立冒号分隔的字段, 那么条目仅被报告而不被修复。**usrck** 命令下一步检查其他文件里规定用户的属性。

**usrck** 命令验证 **/etc/passwd**文件里列出的每一个用户名在 **/etc/security/user**文件、**/etc/security/limits** 文件和 **/etc/security/passwd** 文件中是否有节。**usrck** 命令也验证 **/etc/group** 文件里列出每一个组名在 **/etc/security/group** 文件里是否有节。**usrck** 命令用 **-y** 标志为丢失的用户名和组名在安全文件里创建节。

注: 命令将这种消息写成标准错误。

以下列出所有用户属性, 已被检查的属性用符号表明。

|                       |                                                                                                                                                        |
|-----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>account_locked</b> | 不检查。 <b>usrck</b> 命令设置属性为 True 并禁用帐户。                                                                                                                  |
| <b>admgroups</b>      | 检查 <b>admgroups</b> 在用户数据库中是否有定义, 如果指出系统需要修复错误, 命令除去不在数据库中的所有组。                                                                                        |
| <b>auditclasses</b>   | 检查 <b>auditclasses</b> 在 <b>/etc/security/audit/config</b> 文件里是否为用户定义。如果指出系统应该修复错误, 命令删除 <b>/etc/security/audit/config</b> 文件中未定义的 <b>auditclass</b> 。 |
| <b>auth1</b>          | 检查基本的认证方法。如果方法不是 NONE 或者 SYSTEM, 就必须在 <b>/etc/security/login.cfg</b> 文件里有定义, 程序的属性必须存在且能被 root 用户执行。如果指出系统应该修复错误, 那么只要发现一个错误, 系统就禁用用户帐户。               |
| <b>auth2</b>          | 检查第二认证方法。如果方法不是 NONE 或者 SYSTEM, 就必须在 <b>/etc/security/login.cfg</b> 文件里有定义, 程序的属性必须存在且能被 root 用户执行。系统不修复。                                              |

|                     |                                                                                                                                                                                                                                    |
|---------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>core</b>         | 确保值是切合实际的。如果不是，该命令将值复位为最小值，200 块。                                                                                                                                                                                                  |
| <b>core_hard</b>    | 确保值是切合实际的。如果不是，该命令将值复位为最小值，200 块。                                                                                                                                                                                                  |
| <b>cpu</b>          | 确保值是切合实际的。如果不是，该命令将值复位为最小值，120 秒。                                                                                                                                                                                                  |
| <b>cpu_hard</b>     | 确保值是切合实际的。如果不是，该命令将值复位为最小值，120 秒。                                                                                                                                                                                                  |
| <b>data</b>         | 确保值是切合实际的。如果不是，该命令将值复位为最小值，1272 块 (636K)。                                                                                                                                                                                          |
| <b>data_hard</b>    | 确保值是切合实际的。如果不是，该命令将值复位为最小值，1272 块 (636K)。                                                                                                                                                                                          |
| <b>dictionlist</b>  | 检查字典文件列表。如果指出系统应该修复错误，命令删除用户数据库中不存在的所有字典文件。                                                                                                                                                                                        |
| <b>expires</b>      | 不检查。                                                                                                                                                                                                                               |
| <b>fsize</b>        | 确保值是切合实际的。如果不是，该命令将值复位为最小值，200 块。                                                                                                                                                                                                  |
| <b>fsize_hard</b>   | 确保值是切合实际的。如果不是，该命令将值复位为最小值，200 块。                                                                                                                                                                                                  |
| <b>gecos</b>        | 不检查。                                                                                                                                                                                                                               |
| <b>histexpire</b>   | 确保值是切合实际的。如果表明系统应该修复错误，太大的值设置为最大可能值，太小的值设置为最小可能值。                                                                                                                                                                                  |
| <b>histsize</b>     | 确保值是切合实际的。如果表明系统应该修复错误，太大的值设置为最大可能值，太小的值设置为最小可能值。                                                                                                                                                                                  |
| <b>home</b>         | 通过读方式和搜索方式检查本地目录的存在和可访问性。如果指出系统应该修复错误，那么只要发现一个错误，系统就禁用用户帐户。                                                                                                                                                                        |
| <b>id</b>           | 检查用户标识的唯一性。如果指出系统应该修复错误，命令删除/etc/passwd 文件中所有无效的条目。                                                                                                                                                                                |
| <b>login</b>        | 不检查。                                                                                                                                                                                                                               |
| <b>loginretries</b> | 检查用户尝试登录不成功的次数是否多于允许的数目。如果是，系统禁用用户帐户。                                                                                                                                                                                              |
| <b>logintimes</b>   | 确保时间说明字符串是有效的。如果指出系统应该修复错误，那么只要发现一个错误，系统就禁用用户帐户。                                                                                                                                                                                   |
| <b>maxage</b>       | 确保值是切合实际的。如果表明系统应该修复错误，太大的值设置为最大可能值，太小的值设置为最小可能值。                                                                                                                                                                                  |
| <b>maxexpired</b>   | 确保值是切合实际的。如果表明系统应该修复错误，太大的值设置为最大可能值，太小的值设置为最小可能值。                                                                                                                                                                                  |
| <b>maxrepeats</b>   | 确保值是切合实际的。如果表明系统应该修复错误，太大的值设置为最大可能值，太小的值设置为最小可能值。                                                                                                                                                                                  |
| <b>minage</b>       | 确保值是切合实际的。如果表明系统应该修复错误，太大的值设置为最大可能值，太小的值设置为最小可能值。如果 <b>minage</b> 属性值比 <b>maxage</b> 属性值大，系统也显示要修复错误。                                                                                                                              |
| <b>minalpha</b>     | 确保值是切合实际的。如果表明系统应该修复错误，太大的值设置为最大可能值，太小的值设置为最小可能值。                                                                                                                                                                                  |
| <b>mindiff</b>      | 确保值是切合实际的。如果表明系统应该修复错误，太大的值设置为最大可能值，太小的值设置为最小可能值。                                                                                                                                                                                  |
| <b>minlen</b>       | 确保值是切合实际的。如果表明系统应该修复错误，太大的值设置为最大可能值，太小的值设置为最小可能值。                                                                                                                                                                                  |
| <b>minother</b>     | 确保值是切合实际的。如果表明系统应该修复错误，太大的值设置为最大可能值，太小的值设置为最小可能值。如果 <b>minage</b> 属性值加 <b>maxage</b> 属性值比密码大小最大值更大，系统也会显示要修复错误。                                                                                                                    |
| <b>name</b>         | 检查用户名的唯一性和复合性。名字是由八个或更少的字符组成的唯一的字符串。名字不能以 + (加号)、: (冒号)、- (减号) 或 ~ (波浪号) 开始。以 + (加号) 或 - (减号) 开始的字已经假定为 NIS (网络信息服务) 域中的名字，不允许有任何进一步的处理。不能在字符串中包含冒号 (:), 也不出现所有或者缺省的关键字。如果指出系统应该修复错误，只要发现一个错误，系统就禁用用户帐户且删除 /etc/passwd 文件中任何无效的条目。 |

**usrck** 命令验证 /etc/passwd 文件里列出的每一个用户名在 /etc/security/user 文件、/etc/security/limits 文件和 /etc/security/passwd 文件中是否有节。当已有定义却丢失时，命令为每一个用户添加节。usrck 命令另外验证 /etc/group 文件里列出的每一个组名在 /etc/security/group 文件里是否有节。

|                     |                                                                                           |
|---------------------|-------------------------------------------------------------------------------------------|
| <b>nofiles</b>      | 确保值是切合实际的。如果不是，将值复位为最小值，200。                                                              |
| <b>nofiles_hard</b> | 确保值是切合实际的。如果不是，将值复位为最小值，200。                                                              |
| <b>grp</b>          | 检查用户数据库里基本组的存在。如果指出系统应该修复错误，那么只要发现一个错误，系统就禁用用户帐户。                                         |
| <b>pwdchecks</b>    | 检查外部密码限制方法的列表。如果指出系统应该修复错误，命令删除用户数据库中不存在的所有方法。                                            |
| <b>pwdwarntime</b>  | 确保值是切合实际的。如果不是，系统重新将值设为 <b>maxage</b> 和 <b>minage</b> 之间的值。                               |
| <b>rlogin</b>       | 不检查。                                                                                      |
| <b>rss</b>          | 检查以确保值是切合实际的。如果不是，该命令将值复位为最小值，128 块（64K）。                                                 |
| <b>rss_hard</b>     | 检查以确保值是切合实际的。如果不是，该命令将值复位为最小值，128 块（64K）。                                                 |
| <b>shell</b>        | 通过执行方式检查 <b>shell</b> 的存在和可访问性。如果指出系统应该修复错误，那么只要发现一个错误，系统就禁用用户帐户。                         |
| <b>stack</b>        | 检查以确保值是切合实际的。如果不是，该命令将值复位为最小值，128 块（64K）。                                                 |
| <b>stack_hard</b>   | 检查以确保值是切合实际的。如果不是，该命令将值复位为最小值，128 块（64K）。                                                 |
| <b>su</b>           | 不检查。                                                                                      |
| <b>sugroups</b>     | 检查用户数据库文件中 <b>sugroups</b> 的存在。如果指出系统应该修复错误，命令删除数据库中不有的所有组。                               |
| <b>sysenv</b>       | 不检查。                                                                                      |
| <b>tpath</b>        | 如果 <b>tpath=always</b> ，检查以确保 <b>shell</b> 的属性被标记为可信进程。如果指出系统应该修复错误，那么只要发现一个错误，系统就禁用用户帐户。 |
| <b>ttys</b>         | 在用户数据库文件中检查 <b>ttys</b> 的存在。如果指出系统应该修复错误，命令删除数据库中不有的所有的 <b>ttys</b> 。                     |
| <b>usrenv</b>       | 不检查。                                                                                      |

如果修复涉及禁用用户帐户，则请使用 **chuser** 命令将 **account\_locked** 属性的值复位为 **False**。可以用“系统管理界面工具”（SMIT）来运行 **chuser** 命令，输入：

```
smit chuser
```

Root 用户或者安全组里的成员可以通过除去 **account\_locked** 属性或设置 **account\_locked** 属性为 **False** 来再次恢复用户帐户。Root 用户的帐户不能被 **usrck** 命令禁用。

一般情况下，**sysck** 命令调用 **usrck** 命令作为对信任系统安装验证的一部分。如果 **usrck** 命令在用户数据库里找到任何一个错误，root 用户或安全组成员就执行 **grpck** 命令和 **pwdck** 命令。

**usrck** 命令检查数据管理的安全文件 **/etc/passwd.nm.idx**、**/etc/passwd.id.idx**、**/etc/security/passwd.idx** 和 **/etc/security/lastlog.idx**，查看是否已经更新或者比相应的系统安全文件更新。请注意，**/etc/security/lastlog.idx** 文件不比 **/etc/security/lastlog** 文件新是允许的。如果数据管理安全文件已经过期了，就会出现一个警告消息指出 root 用户应该运行 **mkpasswd** 命令了。

**usrck**命令检查指定的用户是否登录。如果用户因为太多不成功的登录尝试或密码过期不能登录，**usrck** 命令发布警告消息表明为什么不能登录。如果指出系统应该修复错误，只要用户因为以上的原因不能登录，系统就禁用用户的帐户。

如果指定了 **-l** 标志，则 **usrck** 命令会扫描所有用户或 **User** 参数指定的用户，以确定用户是否可以访问系统。下表中列出了用来确定用户是否可以访问的条件：

表 2. 用户可访问性条件

| 条件 | 描述       | 原因                                             |
|----|----------|------------------------------------------------|
| 1  | 用户帐户被锁定。 | 用户的 <b>account_locked</b> 属性被设置为 <b>true</b> 。 |

表 2. 用户可访问性条件 (续)

| 条件 | 描述                        | 原因                                                                                                                           |
|----|---------------------------|------------------------------------------------------------------------------------------------------------------------------|
| 2  | 用户帐户已过期。                  | 用户的 <b>expires</b> 属性所设置的值（过期时间）已过期。                                                                                         |
| 3  | 用户的连续登录尝试失败次数太多。          | 用户的 <b>unsuccessful_login_count</b> 值大于用户的 <b>loginretries</b> 值。                                                            |
| 4  | 用户没有密码。                   | 用户的 <b>password</b> 字段在 <b>/etc/password</b> 或 <b>/etc/security/password</b> 中为 “*”。                                         |
| 5  | 不允许用户在此日期 / 时间登录。         | 当前的日期 / 时间不在用户的 <b>logintimes</b> 属性所定义的允许时间内。                                                                               |
| 6  | <b>/etc/nologin</b> 文件存在。 | <b>/etc/nologin</b> 文件阻止非 root 用户登录。                                                                                         |
| 7  | 用户密码已过期，只有系统管理员才能更改它。     | 用户的密码已过期并且 ADMIN 密码标志已设置。                                                                                                    |
| 8  | 用户被拒绝登录到主机。               | 用户的 <b>hostallowedlogin</b> 和 <b>hostsdeniedlogin</b> 属性不允许访问当前主机。                                                           |
| 9  | 应用程序拒绝用户访问。               | 用户的 <b>login</b> 、 <b>rlogin</b> 和 <b>su</b> 属性被设置为 false，并且 <b>rcmds</b> 属性被设置为拒绝。如果这些属性值中至少有一个但不是所有属性值都拒绝授权，则认为用户可以访问部分系统。 |
| 10 | 用户被拒绝登录到终端。               | 用户的 <b>ttys</b> 属性不允许访问当前终端。系统被视为对于用户是部分可访问的。                                                                                |

如果还指定了 **-b** 标志，输出由两个以制表符分隔的字段（即用户名和 16 个数字的位掩码）组成。位掩码中的每个数字对应于上述“用户可访问性条件”表中的条件，最右边的数字表示条件 1。如果条件的位的位置被设置为 1，则对于该用户关于此条件的检查失败。在输出中保留额外的数字以供将来使用。

下面是带有 **-l** 标志的 **usrck** 命令的示例：

```
# usrck -l testusr1 testusr2
3001-689 The system is inaccessible to testusr1, due to the following:
      User account is locked
      User denied login to terminal.

3001-689 The system is inaccessible to testusr2, due to the following:
      User account is expired.
      User has too many consecutive failed login attempts.
      User denied login to host.
```

下面是带有 **-l** 标志和 **-b** 标志的 **usrck** 命令的示例：

```
# usrck -lb testusr1 testusr2
testusr1      0000000000000001
testusr2      0000000001000110
```



## 标志

- b** 报告无法访问系统的用户及原因，原因是以位掩码的格式显示的。如果指定了 **-b** 标志，则必须指定 **-l** 标志。  
注：位掩码不会报告条件 10（用户被拒绝访问终端），因为在确定系统对用户是否不可访问时不能将它视为完整的情况。同样，如果至少有一个但不是所有属性的值都拒绝认证，位掩码不会报告条件 9（应用程序拒绝用户访问）；只有当所有 4 个属性值都拒绝认证时才会报告此条件。
- l** 扫描所有用户或由 *User* 参数指定的用户以确定用户是否可以访问系统。
- n** 报告错误但不修复。
- p** 修复错误但不报告。
- t** 报告错误，询问是否要修复。
- y** 修复错误且报告。

## 安全性

访问控制：这个命令将执行（x）访问授权给 root 用户和安全组成员。对于 root 用户命令是 **setuid**，并且有可信计算库属性。

访问的文件：

| 方式 | 文件                         |
|----|----------------------------|
| r  | /etc/passwd                |
| r  | /etc/security/user         |
| rw | /etc/security/group        |
| rw | /etc/group                 |
| rw | /etc/security/lastlog      |
| rw | /etc/security/limits       |
| rw | /etc/security/audit/config |
| rw | /etc/security/login.cfg    |

审计事件：

| 事件         | 信息         |
|------------|------------|
| USER_Check | 用户、属性错误、状态 |

## 退出状态

该命令返回以下退出值：

- 0 用户定义文件是合适的。
- >0 发生了错误，或者一个或多个用户定义文件中有错误。返回以下错误代码：
  - EINVAL ( 22 )**  
命令行实参无效
  - ENOENT ( 2 )**  
一个或多个用户定义文件不存在
  - ENOTRUST ( 114 )**  
数据库文件中的用户定义有错，或者用户无法访问系统（由 **-l** 选项发现）

## 示例

1. 若要验证用户数据库中所有用户的存在，有任何错误（不修复）都报告，输入：

```
usrck -n ALL
```

2. 要从用户定义删除那些不在用户数据库文件中的用户，并报告所有错误。输入：

```
usrck -y ALL
```

3. 要显示无法访问系统的用户的列表，请输入：

```
usrck -l ALL
```

4. 要以位掩码格式显示无法访问系统的用户的列表，请输入：

```
usrck -l -b ALL
```

## 文件

|                                   |                        |
|-----------------------------------|------------------------|
| <b>/usr/bin/usrck</b>             | 指定 <b>usrck</b> 命令的路径。 |
| <b>/etc/passwd</b>                | 包含基本用户属性。              |
| <b>/etc/security/user</b>         | 包含用户的扩展属性。             |
| <b>/etc/group</b>                 | 包含组的基本属性。              |
| <b>/etc/security/group</b>        | 包含组的扩展属性。              |
| <b>/etc/security/lastlog</b>      | 包含用户最后一次登录的属性。         |
| <b>/etc/security/limits</b>       | 包含用户进程资源限制。            |
| <b>/etc/security/audit/config</b> | 包含审计系统配置信息。            |
| <b>/etc/security/login.cfg</b>    | 包含配置信息。                |

## 相关信息

**grpck** 命令、**pwdck** 命令、**sysck** 命令。

《安全性》中的 *Securing the network* 描述了用户的标识和认证、自主访问控制、可信计算库和审计。

---

## utmpd 守护程序

### 用途

监控和维护 **/etc/utmp** 文件。

### 语法

```
/usr/sbin/utmpd [ Interval ]
```

### 描述

**utmpd** 守护程序在规则的时间间隔内监控 **/etc/utmp** 文件以获得用户进程项的有效性。根据进程表交叉校验该项的进程标识来除去**/etc/utmp** 文件中已终止的但未清除的用户进程。

**Interval** 参数指定每次 **/etc/utmp** 文件扫描间隔时间的秒数。缺省的时间间隔是 300 秒。

## 用法

要从 **/etc/inittab** 启动 **utmpd**，增加以下的项到文件中：

```
utmpd:2:respawn:/usr/sbin/utmpd
```

**init** 在系统启动过程中启动 **utmpd** 守护程序。要在不重新引导的情况下让更改立即生效，输入：

```
telinit q
```

## 安全性

只有 **root** 用户可以读取和执行这个命令。

## 文件

|                     |                         |
|---------------------|-------------------------|
| <b>/etc/inittab</b> | 指定由 <b>init</b> 命令读取的节。 |
| <b>/etc/utmp</b>    | 包含登录系统的用户的记录。           |

---

## uucheck 命令

### 用途

检查 BNU 要求的文件和目录。

### 语法

```
uucheck [ -v ] [ -x DebugLevel ]
```

### 描述

**uucheck** 命令验证“基本联网实用程序”（BNU）工具要求的文件和目录的存在。该命令也检查 **/etc/uucp/Permissions** 文件中的某些错误。

注：**uucheck** 命令不检查正确的文件和目录的方式或者 **/etc/uucp/Permissions** 文件中的错误，比如重复的登录名或机器名。

安装 BNU 程序、配置您站点的 BNU 工具或在 BNU 工具的某部分中进行更改后（比如 **/etc/uucp/Permissions** 文件），从命令行发出 **uucheck** 命令。

注：只有具备 **root** 用户权限的人才能在命令行中使用 **uucheck** 命令。

### 标志

|                             |                                                              |
|-----------------------------|--------------------------------------------------------------|
| <b>-v</b>                   | 显示 BNU 如何解释 <b>/etc/uucp/Permissions</b> 文件的详细说明。            |
| <b>-x</b> <i>DebugLevel</i> | 显示调试信息。 <i>DebugLevel</i> 变量的有效范围是从 0 到 9，缺省值是 5。数字越大，信息越详细。 |

### 示例

1. 要知道 BNU 程序如何解释 **/etc/uucp/Permissions** 文件，输入：

```
uucheck -v
```

**-v** 标志指示 **uuccheck** 命令验证存在的 BNU 文件并显示 BNU 程序如何解释 **/etc/uucp/Permissions** 文件的详细说明。输出和以下内容相似:

```
*** uuccheck: Check Required Files and Directories
*** uuccheck: Directories Check Complete

*** uuccheck: Check /etc/uucp/Permissions file
** LOGNAME PHASE (when they call us)

When a system logs in as: (unostro)
  We DO allow them to request files.
  We WILL send files queued for them on this call.
  They can send files to
  /
  They can request files from
  /
  Myname for the conversation will be plague.austin..
  PUBDIR for the conversation will be
  /var/spool/uucppublic.

** MACHINE PHASE (when we call or execute their uux requests)

When we call system(s): (nostromo)
  We DO allow them to request files.
  They can send files to
  /
  They can request files from
  /
  Myname for the conversation will be plague.austin..
  PUBDIR for the conversation will be
  /var/spool/uucppublic.

Machine(s): (nostromo)
CAN execute the following commands:
command (ALL), fullname (ALL)

*** uuccheck: /etc/uucp/Permissions Check Complete
```

有关这些权限的解释, 请参阅 **/etc/uucp/Permissions** 文件。

2. 要用 **uuccheck** 命令调试, 输入:

```
uuccheck -x8
```

**-x8** 标志产生大量调试输出。

## 文件

|                              |              |
|------------------------------|--------------|
| <b>/etc/uucp/Permissions</b> | 描述远程系统的访问权限。 |
| <b>/etc/uucp/Systems</b>     | 描述可访问的远程系统。  |

## 相关信息

**uucp** 命令、**uustat** 命令、**uux** 命令。

**uucico** 守护程序、**uusched** 守护程序。

《网络与通信管理》中的『如何配置 BNU』。

---

## uucico 守护程序

### 用途

传输“基本联网实用程序”（BNU）的命令、数据和执行文件到远程系统。

### 语法

```
uucico [ -r RoleNumber ] [ -x DebugLevel ] -s SystemName
```

### 描述

**uucico** 守护程序传输“基本联网实用程序”（BNU）命令（**C.\***）、数据（**D.\***）和执行文件（**E.\***）到指定的远程系统。执行文件由 **uucp** 和 **uux** 命令创建。本地和远程系统都可以运行 **uucico** 守护程序，两个守护程序相互通信完成传输请求。

**uucico** 守护程序执行以下操作：

1. 针对传输请求扫描本地系统上的假脱机目录（**/var/spool/uucp/SystemName**）。
2. 检查 **/etc/uucp/Devices** 文件和 **/etc/locks** 目录里的锁定文件后，选择用于通信连接的设备。
3. 在 **/etc/uucp** 目录中使用 **Systems**、**Dialers** 和 **Dialcodes** 文件发一个调用到规定的远程系统。
4. 执行按 **Systems** 文件中规定要求的登录顺序。
5. 检查 **/etc/uucp/Permissions** 文件列出的权限。
6. 检查 **/etc/uucp** 目录里的 **Maxuuscheds** 和 **Maxuuxqts** 文件的调度极限。
7. 运行所有本地和远程系统发出的传输请求，将已传递的文件放在公共目录（**/var/spool/uucppublic/\***）里。
8. 在 **/var/spool/uucp/.Log/uucico** 目录里的文件里记录传递请求和完成。
9. 通知规定用户传递请求。

当需要时，通常 **uucico** 守护程序是由 **uucp** 和 **uux** 命令调用。并由 BNU 调度守护程序 **uusched**（由 **cron** 守护程序启动）周期性启动。

调试时 **uucico** 守护程序能从命令行启动。BNU **uutry**、**Uutry** 和 **uukick** 命令也启动 **uucico** 守护程序，并且调试打开。

#### 注：

1. 当调用 **uucico** 守护程序时，必须在 **/usr/sbin/uucp** 目录中或者以完整路径 **/usr/sbin/uucp/uucico** 调用守护程序。
2. 在 **uux** 命令要求在远程系统执行命令的情形下，**uucico** 守护程序传输文件，**uuxqt** 守护程序在远程系统上执行命令。

### 标志

**-r** *RoleNumber*

定义服务器和客户机的关系。角色数字 1 表示服务器方式，0 表示客户机方式。如果 **-r** 标志未用，**uucico** 守护程序以客户机方式（**-r 0**）启动，因为 **uucico** 守护程序通常由 BNU 命令或守护程序自动启动。当 **uucico** 守护程序手工启动时，标志应该设为 1。

**-x** *DebugLevel*

在本地终端屏幕上显示调试信息。*DebugLevel* 变量的有效范围为 0 到 9，缺省值是 5，数字越高使得信息越详细。这个标志用于诊断 **/etc/uucp/Systems** 文件中预期发送序列的问题。

**-s** *SystemName* 指定远程系统的名字。当从命令行中启动 **uucico** 守护程序时需要此标志。当 **uucico** 守护程序自动启动时，*SystemName* 变量由内部产生。  
注：系统名字必须只能包含 ASCII 字符。

## 示例

为了从命令行中调用 **uucico** 守护程序，输入：

```
/usr/sbin/uucp/uucico -r 1 -s hera &
```

启动守护程序作为后台进程，访问远程系统 *hera* 。

## 文件

|                                           |                                                                   |
|-------------------------------------------|-------------------------------------------------------------------|
| <b>/etc/locks</b> /*                      | 包含了防止对设备的多重使用和对系统的多重调用的锁定文件。                                      |
| <b>/usr/sbin/uucp</b> /*                  | 包括 <b>uucico</b> 守护程序和 BNU 的配置文件。                                 |
| <b>/etc/uucp/Devices</b>                  | 包含有关可用设备的信息。                                                      |
| <b>/etc/uucp/Dialcodes</b>                | 包含拨号代码缩写。                                                         |
| <b>/etc/uucp/Dialers</b>                  | 指定连接中的初次握手。                                                       |
| <b>/etc/uucp/Maxuuscheds</b>              | 限制已调度作业。                                                          |
| <b>/etc/uucp/Maxuuxqts</b>                | 限制远程命令的执行。                                                        |
| <b>/etc/uucp/Permissions</b>              | 描述远程系统的访问权限。                                                      |
| <b>/etc/uucp/Systems</b>                  | 描述可访问的远程系统。                                                       |
| <b>/var/spool/uucp/.Admin/errors</b>      | 列出 BNU 不能纠正的 <b>uucico</b> 守护程序中的错误。                              |
| <b>/var/spool/uucp/.Log/uucico</b> /*     | 包括 <b>uucico</b> 守护程序的日志文件。                                       |
| <b>/var/spool/uucp/.Status/SystemName</b> | 列出联系远程系统的最后时间和下次尝试前的最小时间。                                         |
| <b>/var/spool/uucp/SystemName</b> /*      | 包括 <b>uucico</b> 守护程序传送的 <b>C.*</b> 、 <b>D.*</b> 和 <b>X.*</b> 文件。 |
| <b>/var/spool/uucp/SystemName/C.*</b>     | 包括命令文件。                                                           |
| <b>/var/spool/uucp/SystemName/D.*</b>     | 包括数据文件。                                                           |
| <b>/var/spool/uucp/SystemName/X.*</b>     | 包括执行文件。                                                           |
| <b>/var/spool/uucppublic</b> /*           | 包括 <b>uucico</b> 守护程序传输后的文件。                                      |

## 相关信息

**cron** 守护程序，**uusched** 守护程序。

**uucp** 命令、**uukick** 命令、**uuq** 命令、**uustat** 命令、**uusnap** 命令、**uutry** 命令、**Uutry** 命令、**uux** 命令。

《网络与通信管理》中的『监视 BNU 远程连接』、『监视 BNU 文件传输』和『使用 **uucico** 守护程序调试 BNU 登录失败』。

《网络与通信管理》中的『BNU 守护程序』和『BNU 文件和目录结构』。

---

## uuclean 命令

### 用途

从 BNU spool 目录除去文件。

## 语法

```
/usr/sbin/uucp/uuclean [ -m ] [ -nHours ] [ -pPrefix ] [ -dSubdirectory ]
```

## 描述

**uuclean** 命令检查“基本联网实用程序”（BNU）的 spool 目录（**/var/spool/uucp**）里有特定前缀的文件，删除那些在规定小时数之前的文件。如果未包括**-n Hours**标志，**uuclean** 命令删除早于 72 小时之前的文件。

如果未包括 **-p** 标志 **uuclean** 命令删除在规定 spool 目录的子目录中所有符合时限要求的文件。如果未包括 **-d** 标志，这命令删除 spool 目录所有子目录的文件（符合时限和前缀要求）。因而，如果 **-d** 和 **-p** 标志都不包括在内，**uuclean** 命令删除 **/var/spool/uucp** 目录所有子目录中符合时限要求的所有文件。

如果 **-m** 标志未定义，**uuclean** 命令发邮件给删除的所有命令文件（**C.\***）的拥有者。如果**-m** 标志有定义，这个命令发邮件给删除的每个文件的拥有者，包括数据（**D.\***）和执行（**X.\***）文件。邮件消息中包含删除文件的名字。

**uuclean** 命令通常由 **cron** 守护程序运行。

注：仅有 root 用户权限的人或作为 **uucp** 登录的人能够发布**uuclean** 命令。

## 标志

|                               |                                                                                                                                                                                  |
|-------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-d</b> <i>subdirectory</i> | 如果和给定的 <b>-n</b> 和 <b>-p</b> 标志规范相匹配，则删除 <b>/var/spool/uucp</b> 目录下指定子目录的文件。如果 <b>-d</b> 标志未规定， <b>uuclean</b> 命令检查 <b>/var/spool/uucp</b> 目录下的所有子目录。 <b>-d</b> 标志最多能规定 10 个子目录。 |
| <b>-m</b>                     | 指示 <b>uuclean</b> 命令在删除文件时发邮件给每个文件的拥有者。                                                                                                                                          |
| <b>-nHours</b>                | 如果与给定的 <b>-d</b> 和 <b>-p</b> 标志规定相符合，删除那些时限超过 <i>Hours</i> 变量规定的小时数的文件。缺省值是 72 小时。                                                                                               |
| <b>-pPrefix</b>               | 如果与给定的 <b>-n</b> 和 <b>-d</b> 标志规定相符合，删除那些有 <i>Prefix</i> 变量规定的前缀的文件。 <b>-p</b> 标志最多能规定 10 个前缀。                                                                                   |

## 示例

1. 要删除所有的旧命令文件，输入：

```
/usr/sbin/uucp/uuclean -pC
```

这个命令删除 **/var/spool/uucp** 目录所有子目录的所有以 C 开始并且时限比 72 小时（缺省值）早的文件。这个系统发邮件给每个文件的原来拥有者，表明文件已经被删除。

2. 要删除系统 **venus** 和**nostromo**所有 spool 目录里的原来的文件，输入：

```
/usr/sbin/uucp/uuclean -n84 -dvenus -dnostromo
```

这个命令删除 **/var/spool/uucp/venus** 和**/var/spool/uucp/nostromo** 目录里所有 84 小时之前的文件。缺省情况下，系统通知 **C.\*** 文件拥有者文件已经被删除，然而不告知其他已经被删除文件的拥有者。

3. 删除所有 spool 目录里所有旧文件并通知使用者它们已经被删除，输入：

```
/usr/sbin/uucp/uuclean -m
```

这个命令删除 spool 目录所有子目录的所有文件，只要文件时限早于 72 小时（缺省值）之前。该命令发邮件给每个被删除文件的所有者。

4. 为了调度通过 **cron** 守护程序周期性的启动 **uuclean** 命令，添加一个类似于下列内容的条目到 **/var/spool/cron/crontabs/uucp** 文件中：

```
15 22 * * * /usr/sbin/uucp/uuclean -n96 -pC -pD -pX
```

该输入条目将在每天 22:15（下午 10:15）使 **cron** 守护程序启动 **uuclean** 命令。每天，**uuclean** 命令将删除 **spool** 目录下所有子目录里 96 个小时之前的所有命令（**C.\***）文件、数据（**D.\***）文件和执行文件（**X.\***）。

## 文件

|                                      |                                      |
|--------------------------------------|--------------------------------------|
| <b>/usr/sbin/uucp/uuclean</b>        | 包括 <b>uuclean</b> 命令。                |
| <b>/var/spool/uucp /*</b>            | 包括被 <b>uuclean</b> 命令除去的假脱机文件。       |
| <b>/var/spool/cron/crontabs/uucp</b> | 为 <b>cron</b> 守护程序调度 <b>uucp</b> 作业。 |

## 相关信息

**uucp** 命令、**uux** 命令、**uucleanup** 命令、**uudemon.cleanu** 命令。

**uucico** 守护程序、**cron** 守护程序。

《网络与通信管理》中的『BNU 文件和目录结构』和『BNU 维护命令』。

---

## uucleanup 命令

### 用途

从“基本联网实用程序”（BNU）的假脱机目录里删除选定的文件。

### 语法

```
uucleanup [ -CDays ] [ -WDays ] [ -mString ] [ -DDays ] [ -TDays ] [ -XDays ] [ -o Days ] [ -sSystemName ]
```

### 描述

“基本联网实用程序”（BNU）**uucleanup** 命令扫描假脱机目录（**/var/spool/uucp**），查找并删除比规定天数早的文件。**uucleanup** 命令执行以下的任务：

- 通知发送和接收不能到达系统的请求的请求者。
- 警告用户规定天数一直在等待的请求。缺省值是 1 天。
- 不能发送的返回邮件返回发送方。
- 除去假脱机目录里所有比规定天数早的其他文件。

#### 注：

1. **uucleanup** 命令通常不在命令行中输入，而是由 **uudemon.cleanu** 命令（一个 shell 过程）来执行。只有 root 用户特权的人能从命令行发出 **uucleanup** 命令。
2. 若安装了 BNU，自动清除就不能启用。编辑 **/var/spool/cron/crontabs/uucp** 文件，从 **uudemon.cleanu** 命令行开始除去注释符（#），以指示 **cron** 守护程序启动 **uudemon.cleanu** 命令。



## 标志

|                     |                                                                                                                                           |
|---------------------|-------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-CDays</b>       | 除去等于或早于 <i>Days</i> 变量规定天数的 <b>C.*</b> （命令）文件，同时通知已经删除文件的请求者。缺省时间是 7 天。                                                                   |
| <b>-DDays</b>       | 除去等于或早于 <i>Days</i> 变量规定天数的 <b>D.*</b> （数据）文件。同时努力发送任何遗留未发的邮件消息。缺省时间是 7 天。                                                                |
| <b>-mString</b>     | 在由 <b>-WDays</b> 选项生成的警告消息文本里包含一个特定的行。缺省行是请本地管理员找到问题。                                                                                     |
| <b>-ODays</b>       | 除去等于或早于 <i>Days</i> 变量规定天数的其他文件。缺省时间是 2 天。                                                                                                |
| <b>-sSystemName</b> | 仅在 <i>System</i> 变量规定的假脱机目录里执行 <b>uucleanup</b> 命令。缺省情况是清除所有的 BNU 假脱机目录。<br><b>注：</b> 系统名字必须只能包含 ASCII 字符。                                |
| <b>-TDays</b>       | 除去等于或早于 <i>Days</i> 变量规定天数的 <b>TM.*</b> （临时）文件。同时努力发送任何遗留未发的邮件消息。缺省时间是 7 天。                                                               |
| <b>-WDays</b>       | 发一封电子邮件给请求者，警告等于或早于 <i>Days</i> 变量规定的天数的 <b>C.*</b> （命令）文件仍然在假脱机目录中。消息包括作业标识和邮件消息，如果请求包含邮件的话。管理员可以使用 <b>-m</b> 选项包含让谁来检查问题的消息行。缺省时间是 1 天。 |
| <b>-XDays</b>       | 除去任何早于或等于 <i>Days</i> 变量规定天数的 <b>X.*</b> （执行）文件。缺省时间是 2 天。                                                                                |

## 示例

### 警告用户他们的命令文件还未被发送。

1. 发送 **C.\***（命令）文件已过时 2 天或以上天数的警告，输入：

```
uucleanup -W2
```

此消息警告请求者文件还没有被发送。

2. 要发送带有警告的消息，输入：

```
uucleanup -m"Check these files waiting in the BNU job queue."
```

这个命令找到 **C.\***（命令）文件中过时 1 天或以上（缺省值）的文件，警告请求者文件还没有发送，并给出消息：Check these files waiting in the BNU job queue.

### 清除命令、数据、执行和其他文件。

1. 若要清除过时 5 天或更多的命令文件，输入：

```
uucleanup -C5
```

这个命令除去所有过时 5 天或以上的 **C.\***（命令）文件，并发一个适当的消息给请求者。

2. 若要清除过时 3 天或以上的数据和执行文件，输入：

```
uucleanup -D3 -X3
```

这个命令除去过时 3 天或以上的所有 **D.\***（数据）文件和所有 **X.\***（执行）文件。

3. 要使用缺省值立即清除所有文件，输入：

```
uucleanup
```

这个命令除去 **C.\***、**D.\***、**T.\*** 和 **X.\*** 文件，以及所有其他比缺省时间早的文件。

**注：**只要 **-C** 和 **-W** 标志一起使用，确保 **-W** 标志定义的值比 **-C** 定义的值要小。否则，**-C** 标志将在可打印任何警告之前删除所有 **C.\***（命令）文件。

## 清除特定系统的文件

要删除系统内文件，输入：

```
uucleanup -shera
```

这个命令用缺省值删除 heras 系统内所有文件，但不能除去任何其他系统的任何文件。

## 文件

**/usr/sbin/uucp/\***

**/var/spool/cron/crontabs/uucp**

**/var/spool/uucp/\***

包含 **uudemon.cleanu** shell 过程和所有 BNU 的配置文件。  
调度 **cron** 守护程序，包括 **uudemon.cleanu** shell 过程的 BNU 作业。

包含由 **uucleanup** 命令除去的文件。

## 相关信息

**cron** 守护程序。

**uucp** 命令、**uudemon.cleanu** 命令、**uustat** 命令、**uux** 命令、**uuclean** 命令。

《网络与通信管理》中的『BNU 维护』。

---

## uucp 命令

### 用途

从一个系统复制文件到另一个系统。

### 语法

```
uucp [ -c | -C ] [ -d | -f ] [ -gGrade ] [ -j ] [ -m ] [ -nUser ] [ -r ] [ -sFile ] [ -xDebugLevel ]  
SourceFile ... DestinationFile ...
```

### 描述

**uucp** 命令是基本联网实用程序 (BNU) 命令，它能将一个或多个源文件从一个系统复制到另一个 UNIX 系统上的一个或多个目标文件。文件能在本地系统间、本地和远程系统间与远程系统间复制。

**uucp** 命令分两步实现文件传输：第一，在本地计算机的假脱机目录中创建一个命令文件 (**C.\***)，然后调用 **uucico** 守护程序向特定的计算机发送请求。命令文件包含信息，例如源文件和目标文件的完整路径名与发送方的登录名。命令文件的完整路径的形式如下所示：

```
/var/spool/uucp/SystemName/C.SystemNameNxxxx
```

这里，*N* 表示请求的级别，*xxxx* 表示 BNU 使用的十六进制顺序数字。

如果使用带有 **-C** 标志的 **uucp** 命令复制传输文件到假脱机目录，**uucp** 命令不仅创建命令文件，而且创建包含实际源文件的数据文件 (**D.\***)。数据文件的完整路径的形式如下所示：

```
/var/spool/uucp/SystemName/D.SystemNamexxxx###
```

一旦命令文件 (数据文件，如果需要的话) 创建好，此时 **uucp** 命令调用 **uucico** 守护程序，这个守护程序接着尝试访问远程计算机以传输文件。

在发布 **uucp** 命令前，发布 **uuname** 命令以确定远程系统的确切名字是很有用的。**uulog** 命令提供关于 **uucp** 命令和另外一个系统交互活动的信息。

## 源文件和目标文件的名称

文件名字和系统名字只能包含 ASCII 字符。任何一个名字都可以是本地系统上的路径名字或有以下格式：

*SystemName!PathName*

其中 *SystemName* 来自 BNU 知道的系统名称列表。

目标 *SystemName* 也可以是名称列表，例如以下所示：

*SystemName!SystemName! . . . ! SystemName!PathName*

这种情况下，尝试用规定的路由器发送文件到目的地。确保这个路由的中间节点能够转送信息，最终链接到下一个系统。

shell 模式匹配字符 ? (问号)、\* (星号) 和 [ . . . ] (括号和省略号) 都可以在源文件的路径名中使用；相应的系统将扩展这些符号。shell 模式匹配字符不应在目标文件的路径名中使用。

如果 *DestinationFile* 是目录而不是文件，**uucp** 命令用 *SourceFile* 名字的最后部分命名传输到远程系统的文件。

## 路径名

*SourceFile* 和 *DestinationFile* 参数的路径名仅能包含 ASCII 字符。源文件的路径可以是以下之一：

- 完整路径名
- 相对路径名

*DestinationFile* 参数的路径形式可以相似于 *SourceFile* 参数或可以是以下之一：

- 由 *~User* 前置的路径名 (例如, *~jkimble*)，这里 *User* 是远程系统的登录名。特定用户登录目录可作为传输文件的目的地。如果用户规定的登录名无效，那么文件传输到公共目录 **/var/spool/uucppublic**，这个目录是缺省目录。
- 由 *~/Destination* 前置的路径名，这里 *Destination* 附加到 **/var/spool/uucppublic** 后面。除非请求传输多个文件、目的地作为目录在远程系统上已经存在或者指定目的地为目录名，否则目的地可作为文件名。

若要指定目的地为目录名，在目的地名字后面紧跟一个 / (斜杠)。例如 *~/amy/* 作为目的地创建目录 **/var/spool/uucppublic/amy** (如果目的地目录不存在)，然后将请求的文件放在这个目录中。

## 权限

- 系统管理员应该限制其他系统上的用户对本地文件的访问权。
- 在传输文件时 **uucp** 命令保留执行权限，将读和写权限授权给所有者、组和所有其他人。( **uucp** 命令拥有文件。)
- 因为安全性限制，发送文件到其他系统的任意 *DestinationFile* 路径名或者从其他系统的任意 *SourceFile* 路径名取文件经常会失败。路径名里指定的文件不但必须给予同一组内的用户读和写权限，而且必须为任何组给予这样的权限。
- 保护文件和请求者拥有的保护目录里的文件必须由 **uucp** 命令发送。

## 标志

- c** 防止文件被复制。此标志是缺省值，不能和 **-C** 标志一起用。如果两个标志都指定，那么 **-c** 标志被覆盖。
- C** 传输文件中复制本地的文件到假脱机目录。取决于 **Poll** 的配置和系统文件以及 **uusched** 守护程序运行的频率，文件可以在要求轮询时立即传输或以后传输。
- 偶而，当传输源文件时会发生问题；例如远程计算机不工作了或登录尝试失败。这个情况下，文件一直保留在假脱机目录中直到传输成功或者被 **cleanup** 命令删除。
- 这个标志屏蔽了 **-c** 标志。
- d** 创建所有的中间目录，在复制源文件到远程系统的目标文件时需要这些中间目录。取代先创建目录然后复制文件到其中这一模式，**uucp** 命令只需要目的地路径名就可执行，**BNU** 会创建所需的目录。此标志是缺省值且不能和 **-f** 标志一起使用。
- f** 在文件传输中不创建中间目录。如果目的地目录已经存在且不想 **BNU** 覆盖这个目录可使用这个标志。这个命令屏蔽了 **-d** 标志。
- gGrade** 指定在特定连接中何时传输文件。*Grade* 变量是 (0 到 9) 或 (A 到 Z, a 到 z) 的一个数字或字母；字符的 ASCII 顺序越低，文件传输越早。数字 0 是最高（最早）的级别；z 是最低（最迟）的级别。缺省值是 **N**。
- j** 在标准输出上显示传输操作作业的标识数字。作业的标识符可以和 **uustat** 或 **uuq** 命令一起使用，以获得特定作业的状态；或者和 **uustat -k** 命令或 **uuq -d** 命令一起使用，以便在传输完成前终止程序。
- m** 当源文件成功复制到远程系统的目标文件后发邮件消息给请求者。消息发到请求者邮箱 **/var/spool/mail/User**。**mail** 命令不能在本地传输中发送消息。
- m** 标志仅当发送多个文件或接收单一文件时起作用。当转发多个文件时不起作用。
- nUser** 通知 *User* 条目定义的远程系统收件人已经发出一个文件。邮件系统不能给本地传输发消息。用户名只能包含 ASCII 字符。接收由 shell 模式匹配符号 ? (问号)、\* (星号) 和 [ . . . ] (括号和省略号) 指定的多个文件不能激活 **-n** 选项。
- r** 防止 **uucico** 文件传输守护程序的启动，即使在允许调用远程系统时发布了这个命令。（缺省下，如果在 **Poll** 和系统文件规定的时间阶段内发布这个命令，就会尝试调用远程系统。）**-r** 选项在调试时很有用。
- sFile** 报告传输状态到规定的文件中。这种情形下，*File* 变量必须指定一个完整路径名。
- xDebugLevel** 在本地系统的屏幕显示调试信息。*DebugLevel* 变量是从 0 到 9 的数字。数字越大，报告越详细。

## 示例

1. 若要从本地系统复制一个文件到远程系统，输入：

```
uucp /home/geo/f1 hera!/home/geo/f1
```

这个示例中，从本地系统中复制 **f1** 文件到远程系统 **hera**。

2. 若要从远程系统中复制一个文件并放在公共目录中，输入：

```
uucp hera!geo/f2 /var/spool/uucppublic/f2
```

这个示例中，从远程系统 **hera** 中复制 **f2** 文件到公共目录中。

3. 要从远程系统复制一个文件放在一个不是公共目录的目录中，输入：

```
uucp hera!geo/f2 /home/geo/f2
```

这个示例中，从远程系统 **hera** 中复制 **f2** 文件到 **/home/geo/f2** 目录中。**geo** 登录目录必须允许其他组成员有写权限，例如，有 **777** 方式。

## 文件

`/usr/bin/uucp`  
`/etc/uucp/Poll`

`/etc/uucp/Systems`  
`/etc/uucp/Sysfiles`  
`/var/spool/uucp`  
`/var/spool/uucppublic`

`/var/spool/uucppublic/SystemName/C.*`  
`/var/spool/uucppublic/SystemName/D.*`

包含 **uucp** 命令。

文件列出自动调用远程系统时（轮询的）的时间。

文件描述了可访问的远程系统。

指定用作 **Systems** 文件的备用文件。

包含 BNU 状态信息的假脱机目录。

公共目录中包含等待由 **uucico** 守护程序传输的文件。

包括命令文件。

包括数据文件。

## 相关信息

**ct** 命令、**cu** 命令、**mail** 命令、**uuclean** 命令、**uucleanup** 命令、**uulog** 命令、**uuname** 命令、**uupick** 命令、**uuq** 命令、**uustat** 命令、**uuto** 命令、**uux** 命令。

**uucico** 守护程序，**uusched** 守护程序。

---

## uucpadm 命令

### 用途

输入基本 BNU 配置信息。

### 语法

**uucpadm**

### 描述

**uucpadm** 命令在 `/etc/uucp` 目录里的 **Devices**、**Systems**、**Permissions**、**Poll** 和 **Dialcode** 文件中提供了基本 BNU 配置信息的修改和交互式项。可以重复使用 **uucpadm** 命令用来调整同样的文件。

当您在命令行里输入 **uucpadm** 命令时，命令显示您可以更改的文件列表。选择要修改的文件以后，命令垂直显示文件里字段名字列表。在每个字段里输入适当的条目。当按下 **Enter** 键时，光标移到列表中的下一个字段。

这个命令用文件的副本记录改动。原来的文件保持不变直到在适当菜单里按顺序按下 **Ctrl+U** 或 **Ctrl+X** 键为止。按下 **Ctrl+D** 键序列可以在任何时刻退出到 **uucpadm** 主菜单，但不保存改变。

帮助例程可以帮助提供每一个数据字段的指令。在菜单的任何字段输入 **?**（问号）来访问该字段的帮助例程。

在任何字段中输入一个 **~**（波浪号）可以进入 ASCII 编辑器，从而为该字段编辑适当的文件。**uucpadm** 命令调用由 **EDITOR** 环境变量指定的编辑器。如果 **EDITOR** 变量未定义，这个命令调用 **vi** 编辑器。

如果第一个菜单项的条目和现有的一条记录相匹配，**uucpadm** 命令为更新检索那条记录。命令也可以告知第一个条目有多少条记录。如果第一个菜单项的条目和现有的记录都不匹配，**uucpadm** 命令在屏幕的顶部显示 **ADD** 单词。

**uucpadm** 命令在您输入时检查数据。如果在文件中发现不一致部分，命令显示警告消息。

如果 **uucpadm** 命令识别出为第一个菜单项所作的条目，命令以缺省值填充字段剩余部分。例如，如果在 **Devices** 文件菜单中输入 TCP 作为 Type，这个命令在剩下的每一个字段中填入 -（连字号）。命令也检查要和其他文件的连续性以及要在系统上运行的进程。例如，在输入 TCP 作为 **Devices** 文件菜单的 Type 时，**uucpadm** 命令检查 **uucpd** 守护程序是否在运行。如果这个守护程序不在运行，命令在 **Type** 字段后显示一个注释，如下所示：

```
Type: TCP
      <Note: Make certain uucpd is enabled.>
Line1: -
```

注：**uucpadm** 命令不能编辑 **/etc/uucp/Dialers** 文件。用 ASCII 编辑器编辑这个文件。

| 方式        | 文件                           |
|-----------|------------------------------|
| <b>rw</b> | <b>/etc/uucp/Devices</b>     |
| <b>rw</b> | <b>/etc/uucp/Dialcodes</b>   |
| <b>rw</b> | <b>/etc/uucp/Permissions</b> |
| <b>rw</b> | <b>/etc/uucp/Poll</b>        |
| <b>rw</b> | <b>/etc/uucp/Systems</b>     |

## 示例

- 若要启动 **uucpadm** 命令，输入以下内容：

```
/usr/sbin/uucp/uucpadm
```

显示列出可更改文件的菜单。

- 若要在 **/etc/uucp/Devices** 文件里加入条目，在 **uucpadm** 菜单里选择 Add/Change Uucp Devices 选项。以下是一个样本 **uucpadm** 屏幕，定义了从 tty3 设备到 merlin 系统的一个直接 9600 波特连接：

```
Type: merlin
line1: tty3
line2: -
line2: -
class: 9600
dialers: direct
```

- 若要在 **/etc/uucp/Systems** 文件里加入条目，在 **uucpadm** 菜单里选择 Add/Change Uucp Systems 选项。以下是一个样本 **uucpadm** 屏幕，定义了 class 2400 上从 nostromo.aus.ibm.com 系统到 ACU 设备的连接：

```
Name: nostromo.aus.ibm.com
Time: Any
Type: ACU
Class: 2400
Phone: 997-7942
Login: nuucp
Login: nuucp
Password: gotcha
```

- 若要改变 **/etc/uucp/Permissions** 文件，在 **uucpadm** 菜单里选择 Add/Change Uucp Permissions File 选项。
  - 以下是一个样本 **uucpadm** 屏幕，定义了 **Permissions** 文件中的 LOGNAME 条目：

```
L/M: LOGNAME=uucpz
Request: yes
Sendfiles: yes
Read: /
Write: NOWRITE=/etc
```

```
Callback:
Callback:
Commands:
Validate: merlin:nostromo
```

如果远程机器是merlin 或 nostromo，登录标识必须是uucpz（验证选项）。远程主机使用这个标识请求发送文件，本地主机被请求时能发送文件。使用者用这个标识能读所有授权给其他组的文件，同时写所有的文件，除了 **/etc** 目录中授权给其他组的文件。

- b. 以下是一个样本 **uucpadm** 屏幕，定义了 **Permissions** 文件中的 MACHINE 条目。

```
L/M: MACHINE=merlin
Request: yes
Sendfiles:
Read: NOREAD=/etc
Write: NOWRITE=/etc
Callback:
Callback:
Commands: ALL
Validate:
```

机器的标识符是 merlin。作出文件传输要求。用户可以读写除了 **/etc** 目录中的文件的所有文件。允许执行所有命令。

5. 若要在 **/etc/uucp/Poll** 文件里加入条目，在 **uucpadm** 菜单里选择 Add/Change Uucp File 选项。以下是一个样本 **uucpadm** 屏幕，定义了 **Poll** 文件中的一个条目。

```
System: merlin
Hours: 0 7 13 19
```

这个条目指导 BNU 在 24 时（午夜）、7 时（上午 7 点）、13 时（下午 1 点）和 19 时（下午 7 点）轮询 merlin.aus.ibm.com 系统。

6. 若要在 **/etc/uucp/Dialcodes** 文件里加入条目，在 **uucpadm** 菜单里选择 Add/Change Uucp Dialcodes 选项。以下是样本 **uucpadm** 屏幕，该屏幕显示如何在 **Dialcodes** 文件中定义条目：

```
Abr: LA
Dialcode: 1-213-
```

这个条目中指定 LA 作为洛杉矶的区域代码。

## 文件

|                               |                        |
|-------------------------------|------------------------|
| <b>/usr/sbin/uucp/uucpadm</b> | 包含 <b>uucpadm</b> 命令。  |
| <b>/etc/uucp/Devices</b>      | 包含有关可用设备的信息。           |
| <b>/etc/uucp/Dialcodes</b>    | 包含拨号代码缩写。              |
| <b>/etc/uucp/Dialers</b>      | 指定连接中的初次握手。            |
| <b>/etc/uucp/Permissions</b>  | 描述远程系统的访问权限。           |
| <b>/etc/uucp/Poll</b>         | 定义何时 BNU 轮询远程系统以便开始任务。 |
| <b>/etc/uucp/Systems</b>      | 描述可访问的远程系统。            |

## 相关信息

**uucp** 命令、**uuname** 命令、

**uucpcheck** 命令检查 **/etc/uucp/Permissions** 文件的正确的配置。

《网络与通信管理》中的『电话连接的 BNU 配置示例』、『TCP/IP 连接的 BNU 配置示例』和『直接连接的 BNU 配置示例』。

《网络与通信管理》中的『BNU 文件和目录结构』。

《网络与通信管理》中的『配置 BNU』。

---

## uucpd 守护程序

### 用途

处理 BNU 和 TCP/IP 间的通信。

### 语法

**uucpd** 守护程序不能从命令行启动。它由 **inetd** 守护程序启动。

### uucpd

### 描述

**uucpd** 守护程序是一个内部程序，使通过“基本联网实用程序”（BNU）链接的系统用户与通过令牌环、以太网或其他网络链接的其他系统建立 TCP/IP 连接。

**uucpd** 守护程序是 **inetd** 守护程序的子服务器。在 BNU 程序能够使用 TCP/IP 系统进行通信前，**uucpd** 守护程序必须在所有联网的系统上作为后台进程运行。如果 **uucpd** 守护程序没有运行，请重新配置 **inetd** 守护程序以启动 **uucpd** 守护程序。使用 **netstat** 命令以查看 **uucpd** 守护程序是否在运行。

### 文件

|                              |                          |
|------------------------------|--------------------------|
| <b>/etc/hosts</b>            | 包含 TCP/IP 所使用的主机名表。      |
| <b>/etc/inetd.conf</b>       | 包含 <b>inetd</b> 守护程序的配置。 |
| <b>/etc/services</b> 文件      | 定义 TCP/IP 所使用的接口指定。      |
| <b>/usr/sbin /uucpd</b>      | 包含 <b>uucpd</b> 守护程序。    |
| <b>/etc/uucp/Devices</b>     | 包含有关可用设备的信息。             |
| <b>/etc/uucp/Permissions</b> | 描述远程系统的访问权限。             |
| <b>/etc/uucp/Systems</b>     | 描述可访问的远程系统。              |

### 相关信息

**inetd** 守护程序。

《网络与通信管理》中的『配置 **inetd** 守护程序』。

《网络与通信管理》中的『传输控制协议 / 因特网协议』。

《网络与通信管理》中的『BNU 守护程序』和『配置 BNU』。

---

## uudecode 命令

### 用途

用电子邮件译码用于传输的二进制文件。



## 语法

**uudecode** [ **-o** *OutputFile* ] [ *InFile* ]

## 描述

**uudecode**命令阅读编码文件，去除发件人添加的前导和后续行，用规定的方式和规定的名字重新创建原来的文件。解码文件可以使结果自动保存到文件中。该文件名与原来提供给 **uuencode** 命令的远程文件变量相同，除非输出文件名使用 **-o** 标志指定。

## 标志

**-o** *OutputFile*

指定将代替包含在输入数据中的任何路径名使用的输出文件名。通过将 **/dev/stdout** 指定为 *OutputFile*，可以将 **uudecode** 的输出指向标准输出。

## 参数

*InFile* 指定要解码文件的名称。

## 示例

要在本地系统上解码用以下命令编码的文件 */tmp/con*:

```
uuencode /usr/lib/boot/unix pigmy.goat > /tmp/con
```

输入:

```
uudecode /tmp/con
```

*pigmy.goat* 文件将和原来编码的文件 */usr/lib/boot/unix* 一样。

## 文件

**/usr/bin/uudecode** 包含 **uudecode** 命令。

## 相关信息

**mail** 命令、**rmail** 命令、**sendmail** 命令、**uucp** 命令、**uuencode** 命令、**uuseid** 命令、**uux** 命令。

---

## uudemon.admin 命令

### 用途

提供关于 BNU 文件传送状态的阶段性信息。

### 语法

**uudemon.admin**

## 描述

`/usr/sbin/uucp/uudemon.admin` 命令是一个 shell 过程，在 `/var/spool/cron/crontabs/uucp` 文件指定的时间间隔内将有关“基本联网实用程序”（BNU）活动的状态信息通过邮件发送给 `uucp` 登录标识。这个命令执行 `uustat -p` 命令和 `uustat -q` 命令：

- `-p` 标志指示 `uustat` 命令为锁定文件中所有进程标识（PID）数运行 `ps -flp` 命令（进程状态，产生已指定进程标识完整的长列表）。
- `-q` 标志列出每个系统上当前排队运行的作业。这些作业或者正在等待执行，或者正在执行中。如果系统的状态文件存在，则报告该系统的日期、时间和状态信息。

每天执行 `uudemon.admin` 命令至少一次。当安装 BNU 程序时，`uudemon.admin` 命令未启用。要自动运行这个命令，编辑 `/var/spool/cron/crontabs/uucp` 文件，从控制运行 `uudemon.admin` 命令一行的开始处除去注释符（#）。

## 示例

要自动运行 `uudemon.admin` 命令，编辑 `/var/spool/cron/crontabs/uucp` 文件，从 `uudemon.admin` 命令行的开始处除去注释符（#）。更改：

```
#48 8,12,16 * * * /usr/bin/sh -c  
"/usr/sbin/uucp/uudemon.admin > /dev/null"
```

为：

```
48 8, 12, 16 * * * /usr/bin/sh -c "/usr/sbin/uucp/uudemon.admin > /dev/null"
```

符号 48 代表分钟，符号 8,12,16 代表基于 24 小时时钟上的小时，三个星号（\* \* \*）是占位符，分别代表日期、月份和星期几。因此这行命令指示 `cron` 守护程序每天在 8 时、12 时和 16 时后的 48 分，也就是在 8:48 a.m.、12:48 p.m. 和 4:48 p.m. 运行 `uudemon.admin` 命令。

注：这些运行时间间隔是缺省值。通过改变这些值，可以更改 `cron` 守护程序执行 `uudemon.admin` 命令的时间以适合您站点的需要。

## 文件

`/usr/sbin/uucp/uudemon.admin`  
`/etc/locks /*`  
`/var/spool/cron/crontabs/uucp`

包含 `uudemon.admin` 命令和 BNU 的配置文件。  
包含了防止对设备的多重使用和对系统的多重调用的锁定文件。  
调度 BNU 作业，包括 `cron` 守护程序的 `uudemon.admin` 命令。

## 相关信息

`uustat` 命令。

《网络与通信管理》中的『BNU 维护命令』。

---

## uudemon.cleanu 命令

### 用途

清除 BNU 假脱机目录和日志文件。

## 语法

**uudemon.cleanu**

## 描述

**/usr/sbin/uucp/uudemon.cleanu** 命令是一个 shell 脚本，用于清除“基本联网实用程序”（BNU）的假脱机目录和日志文件。该命令删除假脱机目录中某个指定日期或早于该日期的文件，然后删除空的假脱机目录。

**uudemon.cleanu** 命令也通过删除 3 天之前的日志信息来更新归档日志文件。该命令从 **var/spool/uucp/Log** 目录删除单个计算机的日志文件，将其合并，然后放在 **var/spool/uucp/Old** 目录中，其中包含原来的日志信息。

在执行清除操作以后，**uudemon.cleanu** 命令会将当日所收集的状态信息的摘要通过邮件发送给 **uucp** 登录标识。

指示 **cron** 守护程序每天、每周或以更长的时间间隔来运行 **uudemon.cleanu** 命令，这取决于在本地系统上 **uucico** 和 **uuxqt** 守护程序执行的事务数量。

要自动运行此命令，请在 **/var/spool/cron/crontabs/uucp** 文件中的 **uudemon.cleanu** 命令行开始处除去注释符（#）。

注：**uudemon.cleanu** 命令通常不在命令行中输入，而是由 **cron** 守护程序执行。

## 示例

要自动运行 **uudemon.cleanu** 过程，编辑 **/var/spool/cron/crontabs/uucp** 文件并将 **uudemon.cleanu** 行取消注释。更改：

```
# 45 23 * * * /usr/bin/sh -c  
"/usr/sbin/uucp/uudemon.cleanu > /dev/null"
```

为：

```
45 23 * * * /usr/bin/sh -c "/usr/sbin/uucp/uudemon.cleanu > /dev/null"
```

符号 45 代表分钟，符号 23 代表基于 24 小时时钟上的小时，而 3 个星号（\* \* \*）是占位符，分别代表日期，月份和星期几。因此这行命令指示 **cron** 守护程序在 23 时 45 分，即 11:45 p.m. 运行 **uudemon.cleanu** shell 过程。

注：

1. 这些运行的时间间隔是缺省值。通过更换它们，可以更改 **cron** 守护程序执行 **uudemon.cleanu** 命令的时间，这样它们就可以适应您站点的需要。
2. 系统为任何一个特定的日志文件分配给 BNU 程序指定数量的存储空间，而块的数量由缺省的 **ulimit** 值决定。如果 **uudemon.cleanu** 命令由于 **ulimit** 值对于本地系统的要求设置过低而未能执行，则从 **/var/spool/cron/crontabs/uucp** 文件删除 **uudemon.cleanu** 命令行（先前所示）并将以下输入添加到 **root** 下的 **crontabs** 文件 **/var/spool/cron/crontabs/root**：

```
45 23 * * * ulimit 5000; /usr/bin/su uucp  
-c "/usr/sbin/uucp/uudemon.cleanu > /dev/null"
```

在 **root** 下的 **crontabs** 文件中输入以上内容时要将文本放在一行内。

## 文件

`/usr/sbin/uucp/uudemon.cleanu`  
`/var/spool/cron/crontabs/uucp`

`/var/spool/cron/crontabs/root`  
`/var/spool/uucp/.Log /*`

包含 `uudemon.cleanu` 命令。  
调度 BNU 作业，包括 `uudemon.cleanu` 命令，用于 `cron` 守护程序。  
调度 `cron` 守护程序的 `root` 用户作业。  
包含 BNU 程序的日志文件。

## 相关信息

`uustat` 命令、`uux` 命令。

`cron` 守护程序、`uucico` 守护程序、`uuxqt` 守护程序。

《网络与通信管理》中的『BNU 日志文件』。

---

## uudemon.hour 命令

### 用途

使用 BNU 程序来启动文件传输调用到远程系统。

### 语法

`uudemon.hour`

### 描述

`/usr/sbin/uucp/uudemon.hour` 命令是一个由“基本联网实用程序”（BNU）使用的 shell 过程。与 `Poll` 文件、`uudemon.poll` 命令和 `/var/spool/cron/crontabs/uucp` 文件结合使用，`uudemon.hour` 命令可以调用远程系统。

`uudemon.hour` 命令调用下列程序，这些程序是在系统之间在指定的时间间隔（以小时计）内传输文件时所涉及的：

- `uusched` 守护程序首先在本地系统对假脱机目录搜索尚未传输到指定的远程系统的命令文件，然后调度这些文件的传输。
- `uuxqt` 守护程序对假脱机目录搜索已传输到本地系统但尚未在此系统中处理过的执行文件。

指示 `cron` 守护程序在指定的时间间隔（以小时计）内运行 `uudemon.hour` 命令。运行 `uudemon.hour` 命令的频率取决于从本地计算机启动的文件传输活动的数量。如果在本地系统的用户启动大量的文件传输，可能需要指定 `cron` 守护程序每小时多次启动 `uudemon.hour` 命令。如果从本地系统启动的文件传输的次数较少，可以指定启动时间（例如：每 4 小时启动一次）。

要自动运行 `uudemon.hour` 命令，应当在 `/var/spool/cron/crontabs/uucp` 文件的 `uudemon.hour` 命令行的开始处除去注释符（#）。

注：在命令行中通常不输入 `uudemon.hour` 命令，但是由 `cron` 守护程序执行该命令。

## 示例

要自动运行 **uudemon.hour** 命令，编辑 **/var/spool/cron/crontabs/uucp** 文件并除去 **uudemon.hour** 命令行开始处的注释符（#）。更改：

```
#25,55 * * * * /usr/bin/sh -c "/usr/sbin/uucp/uudemon.hour > /dev/null"
```

为：

```
25,55 * * * * /usr/bin/sh -c "/usr/sbin/uucp/uudemon.hour > /dev/null"
```

符号 25,55 代表分钟，4 个星号（\* \* \* \*）是占位符，分别代表小时、日期、月份和星期几。因此，这行命令指示 **cron** 守护程序在某一小时的 25 分和 55 分，例如在上午 8:25 和 8:55 以及上午 9:25 和 9:55，以及每天每个小时的同一时刻，运行 **uudemon.hour** 命令。

注：

1. 这些运行时间间隔是缺省值。通过更换它们，可以改变 **cron** 守护程序执行 **uudemon.hour** 命令的时间以适应您的站点的需要。例如，要每 4 小时运行一次 **uudemon.hour** 命令，在 **time-interval** 字段中键入数字 4。
2. 如果改变 **uudemon.hour** 命令的运行时间，还应当改变 **uudemon.poll** 命令的运行时间，这样它可在 **uudemon.hour** 命令运行之前 5 到 10 分钟轮询远程系统。

## 文件

**/usr/sbin/uucp/uudemon.hour**

**/etc/uucp/Poll**

**/var/spool/cron/crontabs/uucp**

包含 **uudemon.hour** 命令。

指定 BNU 程序应轮询远程系统以启动任务的时间。

调度 BNU 作业，包括 **uudemon.hour** 和 **uudemon.poll** 命令，用于 **cron** 守护程序。

## 相关信息

**uudemon.poll** 命令。

**cron** 守护程序、**uusched** 守护程序、**uuxqt** 守护程序。

《网络与通信管理》中的『设置 BNU 的远程系统轮询』。

《网络与通信管理》中的『BNU 维护命令』和『BNU 守护程序』。

---

## uudemon.poll 命令

### 用途

轮询在 BNU **Poll** 文件中列出的系统。

### 语法

**uudemon.poll**

## 描述

`/usr/sbin/uucp/uudemon.poll` 命令是一个由“基本联网实用程序”（BNU）使用的 shell 过程。结合 `/etc/uucp/Poll` 文件、`uudemon.hour` 命令和 `/var/spool/cron/crontabs/uucp` 文件，`uudemon.poll` 命令将启动调用到远程系统。

`uudemon.poll` 命令执行以下操作：

- 轮询（联系）在 `Poll` 文件（`/etc/uucp/Poll`）中列出的系统。
- 为在 `Poll` 文件中列出的系统创建命令（`C.*`）文件。

运行 `uudemon.poll` 命令的时间取决于运行 `uudemon.hour` 命令的时间。一般是在每小时执行的过程之前调度轮询 shell 过程。在 `cron` 守护程序运行 `uudemon.hour` 命令之前此调度启用 `uudemon.poll` 命令以创建任何需要的命令文件。

指示 `cron` 守护程序在运行 `uudemon.hour` 命令大约 5 到 10 分钟之前运行 `uudemon.poll` 命令。要自动运行此过程，请在 `/var/spool/cron/crontabs/uucp` 文件中从 `uudemon.poll` 命令行的开头除去注释符（#）。

注：`uudemon.poll` 命令通常不是在命令行中输入，而是由 `cron` 守护程序来执行。

## 示例

要自动运行 `uudemon.poll` shell 过程，请编辑 `/var/spool/cron/crontabs/uucp` 文件并在启动 `uudemon.poll` 命令的一行的开头除去 #（注释符）。更改：

```
#20,50 * * * * /usr/bin/sh -c "/usr/sbin/uucp/uudemon.poll > /dev/null"
```

为：

```
20,50 * * * * /usr/bin/sh -c "/usr/sbin/uucp/uudemon.poll > /dev/null"
```

符号 20,50 代表分钟，4 个星号（\* \* \* \*）是占位符，分别代表小时、日期、月份和星期几。因此这一行命令指示 `cron` 守护程序在每天每个小时的 20 分和 50 分 - 例如在上午 8:20 和 8:50 以及在上午 9:20 和 9:50 - 运行 `uudemon.poll` 命令。

注：更改 `cron` 守护程序执行 `uudemon.poll` 命令的时间以符合为 `uudemon.hour` 命令所设置的时间。在 `/var/spool/cron/crontabs/uucp` 文件中指定的缺省值指示 `cron` 守护程序在运行 `uudemon.hour` 命令 5 分钟之前运行 `uudemon.poll` 命令。

## 文件

`/usr/sbin/uucp/*`

`/etc/uucp/Poll`

`/var/spool/cron/crontabs/uucp`

包含 `uudemon.poll` 和 `uudemon.hour` 命令以及 BNU 的所有配置文件。

指定 BNU 程序应轮询远程系统以启动任务的时间。

调度 BNU 作业，包括 `uudemon.poll` 命令，用于 `cron` 守护程序。

## 相关信息

`uudemon.hour` 命令。

`cron` 守护程序。

《网络与通信管理》中的『为设置 BNU 的远程系统轮询』。

《网络与通信管理》中的『BNU 维护命令』和『BNU 守护程序』。

---

## uuencode 命令

### 用途

为使用电子邮件的传输编码二进制文件。

### 语法

**uuencode** [ **-m** ] [ *SourceFile* ] *OutputFile*

### 描述

**uuencode** 命令将二进制文件转换成 ASCII 数据。在使用 BNU（或 **uucp**）邮件将文件发送到远程系统之前，这是有用的。**uudecode** 命令将由 **uuencode** 命令创建的 ASCII 数据转换回来成为其原来的二进制格式。

**uuencode** 命令采用已命名的 *SourceFile*（缺省标准输出）并在标准输出上产生编码的版本。该编码只使用可显示的 ASCII 字符，并包括文件的方式和用于在远程系统上重新创建二进制映像的 *OutputFile* 文件名。

使用 **uudecode** 命令对文件进行解码。

### 标志

**-m** 使用 MIME Base64 算法对输出进行编码。如果没有指定 **-m**，则会使用旧的 **uuencode** 算法。

### 参数

*OutputFile* 指定解码文件的名称。通过将 **/dev/stdout** 指定为 *OutputFile*，可以将 **uuencode** 命令的输出指向标准输出。

*SourceFile* 指定将要转换的二进制文件的名称。缺省值是标准输入。

### 示例

1. 要在本地系统上对文件 **unix** 进行编码，并将其用邮件发送到另一个称为 **mysys** 的系统上的用户 **jsmith**，输入：

```
uuencode unix unix | mail jsmith@mysys
```

2. 要在文件 **/tmp/con** 中对名为 **pigmy.goat** 的本地系统上的文件 **/usr/lib/boot/unix** 进行编码，输入：

```
uuencode /usr/lib/boot/unix pigmy.goat > /tmp/con
```

### 文件

**/usr/bin/uuencode** 包含 **uuencode** 命令。

### 相关信息

**mail** 命令、**rmail** 命令、**sendmail** 命令、**uucp** 命令、**uudecode** 命令、**uuseid** 命令和 **uux** 命令。

---

## uuid\_gen 命令 (NCS)

### 用途

生成对象、类型和接口的“通用唯一标识” (UUID)。

### 语法

```
/etc/ncs/uuid_gen [ -c ] [ -p ] [ -C ] [ -P ]
```

### 描述

**uuid\_gen** 程序生成“通用唯一标识” (UUID)。缺省情况下，它生成 UUID 字符串表达式。**uuid\_gen** 的选项使您能够生成 UUID 的源代码表达式，适用于初始化 **uuid\_\$t** 类型的变量。

### 标志

- C** 生成 UUID 的 C 源代码表达式。
- c** 在 C 语法中生成接口定义的模板，包含 UUID 属性。
- P** 生成 UUID 的 Pascal 源代码表达式。
- p** 在 Pascal 语法中生成接口定义的模板，包含 UUID 属性。

### 示例

1. 要生成 UUID 的字符串表达式，输入以下内容：

```
/etc/ncs/uuid_gen
```

这将产生以下输出：

```
34dc23469000.0d.00.00.7c.5f.00.00.00
```

2. 要在 C 语法中生成接口定义的模板，输入：

```
/etc/ncs/uuid_gen -c
```

这将产生以下输出：

```
%c
[
uuid(34dc239ec000.0d.00.00.7c.5f.00.00.00),
version(1)
]
interface INTERFACENAME {

}
```

3. 要生成 UUID 的 C 源代码表达式，输入以下内容：

```
/etc/ncs/uuid_gen -C
```

这将产生以下输出：

```
= { 0x34dc23af,
    0xf000,
    0x0000,
    0x0d,
    {0x00, 0x00, 0x7c, 0x5f, 0x00, 0x00, 0x00} };
```

4. 要在 Pascal 语法中生成接口定义的模板，输入以下内容：

```
/etc/ncs/uuid_gen -p
```



这将产生以下输出:

```
%pascal
[
  uuid (458487b55160.02.c0.64.02.03.00.00.00),
  version (1)]
]
interface INTERFACENAME;

end;
```

5. 要生成 UUID 的 Pascal 源代码表达式, 输入以下内容:

```
/etc/ncs/uuid_gen -P
```

这将产生以下输出:

```
:= [
  time_high := 16#458487df,
  time_low := 16#9fb2,
  reserved := 16#000,
  family := chr(16#02),
  host := [chr(16#c0), chr(16#64), chr(16#02), chr(16#03),
          chr(16#00), chr(16#00), chr(16#00)]
]
```

## 相关信息

NCS 命令的列表。

---

## uukick 命令

### 用途

使用调试模式以联系指定的远程系统。

### 语法

```
uukick [ -xDebugLevel ] SystemName
```

### 描述

**uukick** 命令使用调试模式与远程系统联系, 该系统由 *SystemName* 参数命名。调试模式提供了一种监控“基本联网实用程序”(BNU) 文件传送和连接到远程计算机的方法。

**uukick** 命令启动 **uucico** 守护程序, 该守护程序实际上联系指定的远程系统。**uucico** 守护程序产生调试输出, 使您能够在它与远程系统建立连接、进行远程登录以及传送文件时监控其进度。

调试输出在本地系统屏幕上滚动。一旦系统完成显示该信息, 按“中断”键返回到提示状态。

注:

1. 当发出 **uukick** 命令时, 必须在 **/usr/lib/uucp** 目录中, 或者发出该命令必须带有全路径名 **/usr/sbin/uucp/uukick**。
2. 该 **uukick** 命令是存储在 **/usr/lib/uucp** 目录中的 shell 脚本。

## 标志

**-xDebugLevel** 覆盖该命令在屏幕上显示的调试信息中详细信息的缺省数量。 *DebugLevel* 变量值的有效范围是从 0 到 9，缺省值是 5。数字越大则最后的相关报告就越详细。如果未使用 **-x** 标志，则该 **uucico** 守护程序以缺省级别启动，产生适量的信息。

## 示例

要更改有关 **uucico** 守护程序运行进度的信息的详细数量，使用 **-x** 标志来指定一个更高或更低的调试级别。例如，输入：

```
uukick -x9 hera
```

这指示 **uukick** 命令生成有关 **uucico** 守护程序以何种方式尝试连接到系统 *hera* 的尽可能多的信息。或者，输入：

```
uukick -x3 hera
```

这指示该命令生成比有关该连接的缺省信息量更少的信息量。

## 文件

|                                |                            |
|--------------------------------|----------------------------|
| <b>/usr/sbin/uucp/uukick</b>   | 包含 <b>uukick</b> shell 脚本。 |
| <b>/etc/uucp</b>               | 包含 BNU 的配置文件。              |
| <b>/etc/uucp/Devices</b>       | 包含有关可用设备的信息。               |
| <b>/etc/uucp/Dialcodes</b>     | 包含拨号代码缩写。                  |
| <b>/etc/uucp/Dialers</b>       | 指定连接中的初次握手。                |
| <b>/etc/uucp/Permissions</b>   | 描述远程系统的访问权限。               |
| <b>/etc/uucp/Systems</b>       | 描述可访问的远程系统。                |
| <b>/var/spool/uucp/*</b>       | 包含要传送的文件和记录传送统计信息的文件。      |
| <b>/var/spool/uucppublic/*</b> | 包含已经传送的文件。                 |

## 相关信息

**tail** 命令、**uucp** 命令、**uutry** 命令、**Uutry** 命令和 **uux** 命令。

**uucico** 守护程序、**uucpd** 守护程序。

《网络与通信管理》中的『监视 BNU 远程连接』和『监视 BNU 文件传输』。

《网络与通信管理》中的『维护 BNU』。

---

## uulog 命令

### 用途

提供系统中 BNU 文件传输活动的有关信息。

### 语法

```
uulog [ -x ] [ -Number ] [ -fSystem | -sSystem ]
```

## 描述

“基本联网实用程序”（BNU）**uulog** 命令显示了日志文件的内容。该内容包含了 **uucico** 和 **uuxqt** 守护程序的活动。每一个远程系统都有自己的日志文件，本地系统使用 **uucp**、**uuto** 以及 **uux** 命令来与这些远程系统通信。

使用 **uulog** 命令显示用户和系统要求的 **uucp**、**uuto** 以及 **uux** 命令的摘要信息。所有这些事务都记录在 **/var/spool/uucp/.Log** 目录里的日志文件中。这些文件被命名为 *DaemonName/SystemName*，其中 *DaemonName* 目录是所涉及的守护程序的名字，*SystemName* 文件是为该守护程序正在连接的远程系统而命名的。

**uucp** 和 **uuto** 命令调用 **uucico** 守护程序。**uucico** 守护程序的活动记录在 **/var/spool/uucp/.Log/uucico** 目录里的 *SystemName* 文件中。

**uux** 命令调用 **uuxqt** 守护程序。**uuxqt** 活动记录在 **/var/spool/uucp/.Log/uuxqt** 目录里的 *SystemName* 文件中。

您可以通过发出 **uulog** 命令检查这些单个的日志文件。不过，也可以让 BNU 程序自动将这些临时日志文件附加到一个主日志文件中然后进行检查。这个过程称为压缩日志文件，由 shell 脚本 **uudemmon.cleanu** 命令来处理。

## 标志

- fSystem** 对由 *System* 变量指定的系统的文件传输日志发出带有 **-f** 标志的 **tail** 命令，显示日志文件的结束部分。按“中断”键离开该文件返回提示符状态。
- sSystem** 显示涉及指定系统的复制（**uucico** 守护程序）要求的摘要信息。
  - 注：
    1. 系统名称只能包含 ASCII 字符。
    2. **-f** 和 **-s** 标记不能合用。
- x** 为给定系统显示 **uuxqt** 守护程序日志文件。
- Number** 显示该文件的最后几行。*Number* 变量决定了要显示的行数。（要显示这些行，**uulog** 命令发出一个带有标明指定行数的 **-f** 标志的 **tail** 命令。）

## 示例

1. 要显示 hera 系统的 **uucico** 日志文件，输入：

```
uulog -shera
```

该命令的输出与下列内容相似：

```
uucp hera (10/30-10:18:38,3833,0) SUCCEEDED (call to hera)
uucp hera (10/30-10:18:39,3833,0) OK (startup)
jim hera heraN661d (10/30-10:18:39,3833,0) REQUEST
(nostromo!D.hera661e6c9 --> hera!X.heraN661d (jim))
jim hera heraN661d (10/30-10:18:40,3833,0) FAILED (CAN'T
READ /var/spool/uucp/hera/D.hera661e6c9 13)
uucp hera (10/30-10:18:41,3833,0) OK (conversation
complete -8)
```

上述几行记录了本地系统（nostromo）与远程系统 hera 之间的一次对话。该对话于 10 月 30 日上午 10:18:38 开始，于当日上午 10:18:41 结束。用户 jim 试图将一个数据文件 D.hera661e6c9 传输到 hera 系统中。此次与 hera 的连接是成功的，但该文件不能被传输，因为 BNU 无法读取它。

2. 要显示 **uuxqt** 日志文件，输入：

```
uulog -x
```

3. 要显示 **zeus** 系统的文件传输日志的最后四十行，输入：

```
uulog -fzeus -40
```

## 文件

**/usr/bin/uulog** 包含 **uulog** 命令。  
**/var/spool/uucp/.Log** 包含 BNU 日志文件。

## 相关信息

**tail** 命令、**uucp** 命令、**uudemon.cleanu** 命令、**uuto** 命令和 **uux** 命令。

**uucico** 守护程序、**uuxqt** 守护程序。

《网络与通信管理》中的『BNU 日志文件』。

---

## uuname 命令

### 用途

提供有关本地系统可访问的其他系统的信息。

### 语法

```
uuname [ -c | -l ]
```

### 描述

**uuname** 命令是一个“基本联网实用程序”（BNU）命令，它显示了与本地系统联网的所有计算机的列表。这个可访问系统的列表显示在本地终端的屏幕上。

为了让本地系统通过 BNU 的方式与远程系统通信，该远程系统必须：

- 具有基于 UNIX 的操作系统。
- 连接到本地系统。（电话线可用作连接介质。）

BNU 可以用于工作站和非基于 UNIX 的操作系统之间的通信，但这种通信可能需要附加的硬件或软件。当安装了 BNU 程序后，就能识别出通过 BNU 命令可访问的远程系统，并且在 **BNU Systems** 文件（缺省情况下，为 **/etc/uucp/Systems** 文件，或在 **/etc/uucp/Sysfiles** 文件中指定的一个或多个文件）中列出了这些远程系统。

在通过 **uuto** 或 **uucp** 命令将一个文件复制到另一个系统之前，发出 **uuname** 命令以确定该远程系统的确切名称。

## 标志

- c** 只显示包含在 **cu Systems** 文件（由 **/etc/uucp/Sysfiles** 文件配置）中的系统名称。省略这个标志将显示包含在 **uucico Systems** 文件（也由 **/etc/uucp/Sysfiles** 文件配置）中的系统名称。如果没有使用 **/etc/uucp/Sysfiles** 将 **cu** 和 **uucico** 配置分别放置到单独的 **Systems** 文件中，则不管是否有 **-c** 标志，在 **/etc/uucp/Systems** 中列出的所有系统名称都将显示出来。
- l** 显示本地系统的名称。

## 示例

- 要识别与本地系统连接的远程系统，输入：

```
uuname
```

系统响应类似以下内容的列表：

```
arthur  
hera  
merlin  
zeus
```

- 要识别本地系统的名称，输入：

```
uuname -l
```

系统响应类似以下内容：

```
nostramo
```

## 文件

|                              |                              |
|------------------------------|------------------------------|
| <b>/usr/bin/uuname</b>       | 包含 <b>uuname</b> 命令。         |
| <b>/etc/uucp/Systems</b>     | 列出可访问的远程系统。                  |
| <b>/etc/uucp/Sysfiles</b>    | 指定用作 <b>Systems</b> 文件的备用文件。 |
| <b>/var/spool/uucp</b>       | 包含 BNU 管理文件。                 |
| <b>/var/spool/uucppublic</b> | 包含等待传送的 BNU 文件（公共目录）。        |

## 相关信息

**ct** 命令、**cu** 命令、**uname** 命令、**uucp** 命令、**uupick** 命令、**uustat** 命令、**uuto** 命令和 **uux** 命令。

---

## uupick 命令

### 用途

完成传输并处理 **uuto** 命令所发出的文件。

### 语法

```
uupick [ -sSystem ]
```

### 描述

**uupick** 命令是一个“基本联网实用程序”（BNU）命令，它完成传输并处理 BNU **uuto** 命令发给一个指定用户标识的文件。

一旦该复制文件到达 `receive` 目录，`rmail` 命令通知接收人文件已经到达。接收人接着发出 `uupick` 命令，搜索本地系统里的公共目录以查找以下列名称的形式发送的文件：

```
/var/spool/uucppublic/receive/User/System/File
```

对于每一个搜索到的文件或目录，`uupick` 命令在本地系统的屏幕上显示如下消息：

```
from System: [file File] [dir Directory]
?
```

跟在该消息后面的问号提示符（?）暗示您现在可以输入文件处理选项中的任意一项。

## 标志

**-s** *System*        在 `/var/spool/uucppublic/receive/User/System` 中搜索从指定系统发来的文件。系统名称只包含 ASCII 字符。

## 文件处理选项

跟在该消息后面的问号提示符（?）指示应输入下列文件处理选项之一：

| 选项                            | 操作                                                                                                    |
|-------------------------------|-------------------------------------------------------------------------------------------------------|
| <b>!</b> <i>Command</i>       | 退出到 <code>shell</code> 以运行指定的命令。执行完该命令后，用户自动返回到 <code>uupick</code> 命令。                               |
| <b>*</b>                      | 显示所有文件处理选项。                                                                                           |
| <b>a</b> [ <i>Directory</i> ] | 将当前在 <code>receive</code> 目录下的 <code>uuto</code> 文件迁移到本地系统中的一个指定的目录里。缺省情况是当前的工作目录。使用完整或相对路径名来指定目的地目录。 |
| <b>Ctrl-D</b>                 | 停止处理，从 <code>uupick</code> 命令中退出。                                                                     |
| <b>d</b>                      | 删除指定文件。                                                                                               |
| <b>m</b> [ <i>Directory</i> ] | 将文件迁移到一个指定的目录。如果 <i>Directory</i> 变量没有指定完整的路径名，则指定一个相对于当前目录的目的地址。如果没有给出目的地，则缺省认定为本地系统的当前的工作目录。        |
| <b>new-line</b>               | 按下 <code>Enter</code> 键则移动到 <code>receive</code> 目录下的下一个条目。                                           |
| <b>p</b>                      | 在 workstation 屏幕上显示文件的内容。                                                                             |
| <b>q</b>                      | 停止处理，从 <code>uupick</code> 命令中退出。                                                                     |

## 示例

1. 要接收一个用 `uuto` 命令发出的文件并将其添加到当前的工作目录中，输入：

```
uupick
```

系统响应的消息类似如下内容：

```
from system anchor: file file1
?
```

输入：

```
a
```

在本例中，用 `uuto` 命令从系统 `anchor` 发出的文件 `/usr/bin/file1` 被添加到当前的工作目录中。

2. 要接收一个用 `uuto` 命令发出的文件并将其添加到本地系统中一个指定的目录中，输入：

```
uupick
```

系统响应的消息类似如下内容：

```
from system anchor: file file2
?
```

输入:

```
a /usr/bin1
```

在本例中, 用 **uuto** 命令从系统 anchor 发出的文件 /usr/bin/file2 被添加到本地系统中的 /usr/bin1 目录中。

**注:** a /usr/bin1 指令意味着要移动所有的文件而不只是一个文件。这样, 如果 ~/anchor/... 目录中有任何其他文件, 则这些文件也要被移动。

3. 要搜索从系统 anchor 发出的文件, 输入:

```
uupick -s anchor
```

系统响应的消息类似如下内容:

```
from system anchor: file file1
```

## 文件

**/usr/bin/uupick**

包含 **uupick** 命令。

**/var/spool/uucppublic**

包含 BNU 公共目录。

## 相关信息

**ct** 命令、**cu** 命令、**uucp** 命令、**uname** 命令、**uustat** 命令、**uuto** 命令和 **uux** 命令。

---

## uupoll 命令

### 用途

强制对远程 BNU 系统轮询。

### 语法

```
uupoll [ -gGrade ] [ -n ] SystemName
```

### 描述

**uupoll** 命令使“基本联网实用程序”(BNU)对由 *SystemName* 参数指定的远程系统进行轮询。此命令通常由 **cron** 守护程序运行或由想立即执行作业的用户运行。否则, 只在由 **/etc/uucp/Poll** 和 **/var/spool/cron/crontabs/uucp** 文件中调度的时间, **uudemon.poll** 命令对远程系统进行轮询。

正常的讲, 只在由 **Poll** 文件指定的时间或此远程系统有排队的作业时, **uucico** 守护程序才访问远程系统。**uupoll** 命令对远程系统的空作业进行排队然后调用 **uucico** 守护程序。这就使 **uucico** 守护程序立刻访问远程系统并尝试对它发送所等候的任何作业。使用 **-g** 标志指定只发送高优先级作业。

使用 **-n** 标志对空作业进行排队, 而不用启动 **uucico** 守护程序。选项使用于:

- 在调用 **uucico** 守护程序调试以前, 对空作业排队。
- 通常在调用 **uucico** 守护程序以前对空作业进行排队, 然后使守护程序对指定的系统进行轮询。

*SystemName* 参数是必需的参数，用于指定要轮询的远程系统的名称。

## 标志

- gGrade** 指示 **uupoll** 命令只发送已给定优先级的作业（由 *Grade* 参数指定）或在此调用中优先级更高的作业。低优先级的作业将保留在队列中直到下一次远程系统再次被轮询。
- n** 对空作业排队，但不调用 **uucico** 守护程序。

## 示例

1. 要使用 **cron** 守护程序运行 **uupoll** 命令，在您的 **crontabs** 文件中如下放入条目：

```
0 1,7,16 * * * /usr/bin/uupoll hera
```

这在以下时间轮询系统 hera: 0100 小时 (1 a.m.)、0700 小时 (7 a.m.) 和 1600 小时 (4 p.m.) 每天。

2. 如果当地的系统已经在特定的时刻运行 **uucico** 守护程序，通常您想在现运行的 **uucico** 守护程序前排队空作业。例如，如果系统以钟点计算运行 **uucico** 守护程序，在您的 **crontabs** 文件中如下放入条目：

```
0 1,7,16 * * * /usr/bin/uupoll -n zeus
0 5,12,21 * * * /usr/bin/uupoll -n hera
5 * * * * /usr/sbin/uucp/uucico -r1
```

为远程站点在给定时刻进行空作业排队，空作业在给定时刻 5 分钟后由 **uucico** 守护程序运行处理。

3. 要使 **uucico** 守护程序传递 N 优先级或更高优先级的所有作业给系统 zeus:

```
uupoll -gN zeus
```

## 文件

|                                          |                           |
|------------------------------------------|---------------------------|
| <b>/usr/bin/uupoll</b>                   | 包含 <b>uupoll</b> 命令。      |
| <b>/etc/uucp/Poll</b>                    | 指定 BNU 程序应轮询远程系统以启动任务的时间。 |
| <b>/var/spool/cron/crontabs/uucp</b>     | 调度对远程系统的自动轮询。             |
| <b>/var/spool/uucp/<i>SystemName</i></b> | 包含要传递给远程系统的文件。            |

## 相关信息

**uucp** 命令和 **uux** 命令。

**Uutry** 命令调用 **uucico** 守护程序接通调试。

**uudemon.poll** 和 **uudemon.hour** 命令按 **cron** 守护程序的调度对远程系统执行自动轮询。

**uucico** 守护程序。

《网络与通信管理》中的『理解 BNU 守护程序』。

---

## uuq 命令

### 用途

显示 BNU 作业队列，从队列中删除指定的作业。



## 语法

**uuq** [ **-l** | **-h** ] [ **-sSystemName** ] [ **-uUser** ] [ **-dJobNumber** ] [ **-rSpoolDir** ] [ **-bBaudRate** ]

注：只有具有 root 用户权限的用户可以使用 **-d** 标志。

## 描述

**uuq** 命令用于列出或删除在“基本联网实用程序”（BNU）的作业队列中的作业条目。

当列出作业时，**uuq** 命令的格式与 **ls** 命令所使用的格式类似。在缺省的格式中，**uuq** 命令只对对在队列中等待的作业代码进行列表，每个系统后面都跟着一总结行。

在总结格式中（**uuq -h**）只有总结行被列出。总结行给出：

- 系统名称
- 系统作业数量
- 要发送的字节总数

在长格式（**uuq -l**）中，速度可能会很慢，对每个作业列出的信息是：

- 作业代码
- 要传输的文件数量
- 发送作业的用户
- 要发送的字节数
- 请求作业的类型：

S     发送一个文件  
R     接收一个文件  
X     在远程系统上执行一个命令

- 要发送的文件或要接收的文件或要执行的命令

具有 root 用户权限的用户在运行 **uuq** 程序找到作业代码后，可以使用 **-dJobNumber** 标志来删除在队列中所找到的作业。

## 标志

|                     |                                                                    |
|---------------------|--------------------------------------------------------------------|
| <b>-bBaudRate</b>   | 使用已经给定的波特率来替换缺省值（1200 波特），用于计算传输时间。                                |
| <b>-d JobNumber</b> | 从 BNU 队列中删除由 <i>JobNumber</i> 变量指定的作业。只有具有 root 用户权限的用户可以从队列中删除作业。 |
| <b>-h</b>           | 只显示每个系统的总结行。                                                       |
| <b>-l</b>           | 用长格式列出输出。                                                          |
| <b>-sSystemName</b> | 只列出名称以 <i>SystemName</i> 变量指定的字符串开头系统的作业。                          |
| <b>-r SpoolDir</b>  | 搜索由 <i>SpoolDir</i> 变量指定的在假脱机目录中的文件，而不是在缺省假脱机目录中搜索文件。              |
| <b>-uUser</b>       | 只列出用户排队的作业，这些用户的登录名需要用 <i>User</i> 变量指定的字符串打头。                     |

## 示例

1. 要为 hera 系统获取所有假脱机作业的长列表，输入：

```
uuq -l -shera
```

2. 要获取所有系统的总结列表，输入：

```
uuq -h
```

3. 若用户 `nita` 要从队列中删除一个作业，首先使用 `uuq` 命令查找要删除的作业的代码，操作如下：

```
uuq -l -unita
```

为用户 `nita` 产生假脱机作业的列表。查找想除去的作业。例如，如果作业的代码是 `13451`，那么后面的命令会删除作业：

```
uuq -d13451
```

注：您必须具有 `root` 用户权限或用 `uucp` 登录，才能从队列中删除作业。

## 文件

`/usr/bin/uuq`

`/var/spool/uucp/SystemName`

`/var/spool/uucp/SystemName/C.*`

`/var/spool/uucp/SystemName/D.*`

`/var/spool/uucp/SystemName/X.*`

包含 `uuq` 命令。

对于由 `SystemName` 指定的远程系统，包含假脱机文件。

包含文件传输的指令。

包含要传输数据文件的信息。

包含执行远程命令的指令。

## 相关信息

`uucp` 命令、`uux` 命令、`uulog` 命令和 `uusnap` 命令。

《网络与通信管理》中的『BNU 守护程序』和『BNU 维护命令』。

---

## uusched 守护程序

### 用途

调度“基本联网实用程序”（BNU）文件传输程序的工作。

### 语法

```
uusched [ -uDebugLevel ] [ -xDebugLevel ]
```

### 描述

`uusched` 守护程序调度“基本联网实用程序”（BNU）文件传输程序的工作。它调度在 `/var/spool/uucp/SystemName` 目录中排队的文件传送。调度守护程序首先随机化该工作，然后启动传送文件的 `uucico` 守护程序。

`uusched` 守护程序通常由 `uudemon.hour` 命令启动。此命令是一个 `shell` 过程，定期根据 `/var/spool/cron/crontabs/uucp` 文件中的指令由 `cron` 守护程序来运行。

为了进行调试，`uusched` 守护程序也可以从命令行启动。

注：当启动 `uusched` 守护程序时，您必须处于 `/usr/sbin/uucp` 目录中，或者您必须用全路径名 `/usr/sbin/uucp/uusched` 来启动该守护程序。

## 标志

- uDebugLevel** 此标志将作为 **-xDebugLevel** 标志传递给 **uucico** 守护程序。*DebugLevel* 变量是从 0 到 9 的数字，缺省值是 5。越大的数字将给出越详细的调试信息，该信息显示在本地系统的屏幕上。
- xDebugLevel** 从 **uusched** 守护程序输出调试消息。*DebugLevel* 变量是从 0 到 9 的数字，缺省值是 5。越大的数字将给出越详细的调试信息，该信息显示在本地系统的屏幕上。

## 示例

要从命令行启动 **uusched** 守护程序，请输入：

```
/usr/sbin/uucp/uusched &
```

这将启动 **uusched** 守护程序作为后台进程。（注意命令中包含了路径名。）

## 文件

- /etc/locks /\*** 包含了防止对设备的多重使用和对系统的多重调用的锁定文件。
- /usr/sbin/uucp/\*** 包含 **uusched** 守护程序和 BNU 配置文件。
- /etc/uucp/Devices** 包含有关可用设备的信息。
- /etc/uucp/Maxuuscheds** 限制已调度作业。
- /etc/uucp/Systems** 描述可访问的远程系统。
- /var/spool/cron/crontabs/uucp** 调度 **cron** 守护程序，包括 **uudemon.hour** shell 过程的 BNU 作业。
- /var/spool/uucp/SystemName /\*** 包含等待传送的文件。

## 相关信息

**uucp** 命令、**uudemon.hour** 命令、**uustat** 命令和 **uux** 命令。

**cron** 守护程序、**uucico** 守护程序。

《网络与通信管理》中的『理解 BNU 守护程序』。

---

## uusend 命令

### 用途

将文件发送到远程主机。

### 语法

```
uusend [ -mMode ] [ -r ] Sourcefile System [ !System ... ] ! RemoteFile
```

### 描述

**uusend** 命令将文件发送到远程系统上的给定位置。远程系统不必直接连接到本地系统上，但是 UUCP 链路的一个链必须连接这两个系统，并且 **uusend** 命令在链的每一个系统上都必须可用。

系统的链由 *System[!System ...]* 参数给定，它分别列出了文件要传送到到的每一个远程系统，用！（感叹号）隔开。**!RemoteFile** 参数给出了要存储文件的名称（当它到达链中的最后一个系统时）。

注：不要在系统名称和感叹号之间或在最后一个感叹号和远程文件名称之间添加任何空格。

*SourceFile* 参数指定了在本地上文件的名称。如果使用了 - (短划线)，则 **uusend** 命令使用标准输入。

## 标志

- m Mode** 指定远程系统上文件的方式将从给定的八进制数字取得。如果没有指定此标志，将会使用输入文件的方式。
- r** 阻止启动会在系统间传输文件的 **uucico** 守护程序。缺省值是启动 **uucico** 守护程序。

当 **uusend** 命令在链中向下一个远程系统传送文件时，标志主要由该命令在内部使用。

## 示例

要将文件跨越一个系统传输到另一个系统，输入：

```
uusend /etc/motd nostromo!gandalf!~nuucp
```

*/etc/motd* 文件发送到系统 *nostromo*，然后到达系统 *gandalf*，并放在 *nuucp* 的主目录 */var/spool/uucppublic/nuucp* 中，其中 *nuucp* 是一个 BNU 登录标识。

## 文件

**/usr/bin/uusend**

包含 **uusend** 命令。

## 相关信息

**uucp** 命令和 **uux** 命令。

**uucico** 守护程序。

---

## uusnap 命令

### 用途

显示 BNU 与远程系统联系的状态。

### 语法

**uusnap**

### 描述

**uusnap** 命令显示了一张展示“基本联网实用程序”（BNU）状态的表。此表包含每个远程系统以下信息：

|             |                                                                                                                              |
|-------------|------------------------------------------------------------------------------------------------------------------------------|
| SystemName  | 指定远程系统的名字。                                                                                                                   |
| Number Cnds | 指定已排队等候远程系统的命令文件的数量（ <b>C.*</b> 文件）。                                                                                         |
| Number Data | 指定已排队等候远程系统的数据传输的数量（ <b>D.*</b> 文件）。                                                                                         |
| Number Xqts | 指定已排队等候远程系统的远程命令执行的数量（ <b>X.*</b> 文件）。                                                                                       |
| 消息          | 在 <b>/var/spool/uucp/.Status/SystemName</b> 文件中指定站点的当前状态消息。Message 字段可能包含在 BNU 重试连接到远程系统前所剩余的时间，以及 BNU 尝试连接到系统已失败的次数（如果有的话）。 |

## 示例

要查看 BNU 状态的快照，请输入：

```
uusnap
```

此命令的输出与以下内容相似：

```
nostromo 4 Cnds 2 Data 2 Xqts SUCCESSFUL
zeus      2 Cnds 1 Data 2 Xqts NO DEVICES AVAILABLE
```

这些行表示当前有四个命令文件、两个数据文件和两个执行文件在排队等候系统 `nostromo`。最后一次连接到系统 `nostromo` 是成功的。另一方面，最后一次尝试与系统 `zeus` 联系是失败的，因为在本地系统上没有可用的设备。

## 文件

`/usr/bin/uusnap`

`/var/spool/uucp/.Status/SystemName`

`/var/spool/uucp/SystemName`

`/var/spool/uucp/SystemName/C.*`

`/var/spool/uucp/SystemName/D.*`

`/var/spool/uucp/SystemName/X.*`

包含 `uusnap` 命令。

记录 BNU 与远程系统联系的状态。

包括 `uucico` 守护程序传送的 `C.*`、`D.*` 和 `X.*` 文件。

指示 BNU 有关要传送的文件。

包含由 BNU 传送的文件。

指定通过 BNU 来远程执行的命令。

## 相关信息

`uucp` 命令、`uux` 命令、`uuq` 命令。

`uucico` 守护程序。

《网络与通信管理》中的『BNU 文件和目录结构』和『BNU 维护命令』。

---

## uustat 命令

### 用途

报告 BNU 操作的状态并提供对 BNU 操作的有限的控制。

### 语法

```
uustat [ [ -n Number ] [ -a | -k JobID | -m | -p | -q | -r JobID ] | [ -s System ] [ -u User ] ]
```

### 描述

`uustat` 是一个“基本联网实用程序”（BNU）命令，显示了几个类型的 BNU 操作状态信息。在监控 BNU 请求状态时特别有用。

另外，对在远程系统上将要运行的 BNU 作业队列，`uustat` 命令也给予了用户有限的控制。通过发出带有合适标志的命令，用户可以检查 BNU 与其他系统连接的常规状态，并可以取消由 `uucp` 和 `uuto` 命令发出的复制请求。

如果 **uustat** 命令在发出时没有带任何标志，那么它会报告由当前用户发送的所有 BNU 请求（从最后一次挂起队列被清除算起）的状态。这些状态报告以如下形式显示：

```
jobid date/time status system_name user_ID size file
```

共有两种类型的 BNU 队列：

- 当前的队列，由 **-q** 标志访问，列出了在一台或多台指定计算机上排队运行或当前正在运行的 BNU 作业。
- 挂起的队列，由 **-a** 标志访问，列出了所有在一段时间内没有执行的作业。

经过一段时间后，在挂起的队列中的条目，既可用 BNU **uucleanup** 命令手工删除，又可以由 **uudemon.cleanu** 等命令自动删除，这些命令由 **cron** 守护程序启动。

当向最近没有被访问的系统发送文件时，最好用 **uustat** 命令看一下最近一次的访问时间，因为远程系统可能关闭或者不能服务。

## 标志

下列标志是互斥的。一次只能有一个标志与 **uustat** 命令一起使用。

- a** 无论是哪个用户输入了最初的 BNU 命令，所有在挂起队列中的作业信息都将被显示。
- k JobID** 取消由 *JobID* 变量指定的 BNU 进程。使用此标志的用户必须是作出 **uucp** 请求（现在已经被取消）的人或者具有 root 用户权限来进行操作。

这个标志只能取消作业还在本地计算机上的进程。当 BNU 将作业发送到远程系统上执行后，**-k JobID** 标志就不能用于取消远程作业。
- m** 报告最近用 BNU 命令试图访问指定的系统的状态。如果 BNU 请求完成，那么状态报告成功。如果作业没有完成，那么状态报告会发出错误消息，说明登录失败。
- n Number** 允许用户指定用来搜集 BNU 状态信息的机器数量。指定的数量应当大于或等于 *Systems* 文件中的机器数量。缺省值是 200。
- p** 对于在锁定文件中的所有 PID 数字运行 **ps -flp**（进程状态：指定进程 ID 的完整长列表）。
- q** 列出准备在每一个系统上运行的当前队列中的作业。这些作业或者正在等待执行或者正在执行的过程中。如果系统的状态文件存在，则报告该系统的日期、时间和状态信息。当作业完成时，BNU 从当前队列中除去作业列表。

在状态报告中，圆括号中的数字（邻接 **C.\***（命令）文件数字或 **X.\***（执行）文件数字）代表系统最先的 **C.\*** 或 **X.\*** 文件的天数。**retry** 字段代表 BNU 尝试和失败执行命令的次数，例如失败可能是由于登录失败、文件锁定或者不可用的设备。
- r JobID** 用当前日期和时间标记在由 *JobID* 变量指定的挂起队列中的文件。使用此标志以确保在作业的修改时间没有超过指定的日期之前，清除操作不会删除文件。

您可以同 **uustat** 命令一起使用一个或两个如下的标志：
- s System** 报告对由 *System* 变量指定的工作站的工作站的 BNU 请求的状态。*System* 名称只能包含 ASCII 字符。
- u User** 对于每一个工作站，报告由 *User* 变量指定的用户发送的 BNU 请求的状态。*User* 名称只能包含 ASCII 字符。

## 示例

1. 要显示在挂起队列中所有的 BNU 作业的状态，输入：

```
uustat -a
```

系统使用消息响应如下：

```

heraC3113 11/06-17:47 S hera amy 289 D.venus471afd8
zeusN3130 11/06-09:14 R zeus geo 338 D.venus471bc0a
merlinC3120 11/05-16:02 S merlin amy 828 /home/amy/tt
merlinC3119 11/05-12:32 S merlin msg rmail amy

```

| 字段 | 描述                       |
|----|--------------------------|
| 1  | 此操作的作业 ID                |
| 2  | BNU 命令发出的日期和时间           |
| 3  | 是 S 还是 R, 取决于作业是发送还是接收文件 |
| 4  | 系统名称 (命令输入的系统)           |
| 5  | 发出命令的用户标识                |
| 6  | 远程命令字段或名称的大小             |
| 7  | 文件名称。                    |

当文件的大小给定时, 如上面示例起初三行的输出, 文件名称也被显示。文件名称可以由用户给定的名称, 如在 /home/amy/tt 条目中所示; 或者由 BNU 内部指定给数据文件的名称, 此数据文件与远程执行联系, 例如 D.venus471afd8。

- 要显示在当前队列中所有作业的状态, 输入:

```
uustat -q
```

系统使用消息响应如下:

```

merlin 3C      07/15-11:02  NO DEVICES AVAILABLE
hera   2C      07/15-10:55  SUCCESSFUL
zeus   1C (2)   07/15-10:59  CAN'T ACCESS DEVICE

```

输出显示了每一个系统有多少 **C.\*** (命令) 文件等待。示例中第三行圆括号中数字 (2) 表明 **C.\*** 文件已经在队列中等待了两天。日期和时间是关于当前与系统的交互作用的信息, 还带有交互作用状态的报告。

- 要显示锁定文件中所有进程的 ID, 输入:

```
uustat -p
```

系统使用消息响应如下:

```

LCK..tty0: 881
LCK.S.0: 879
LCK..hera: 881
F S UID PID PPID C PRI NI ADDR SZ WCHAN STIME TTY
101 S uucp 881 879 26 39 39 370 296 3fffe800 09:57:03 -
TIME COMD
0:00 UUCICO -r1 -shera
101 S uuc 879 1 11 33 39 770 156 8d874 09:57:02 -
0:00 /usr/sbin/uucp/uusched

```

- 若要取消当前队列的一个作业, 首先确定作业的 ID, 然后输入取消作业的命令。要确定作业 ID, 输入:

```
uustat -a
```

系统使用消息响应如下:

```

heraC3113 11/06-17:47 S hera amy 289 D.venus471afd8
merlinC3119 11/06-17:49 S merlin geo 338 D.venus471bc0a

```

要取消 ID 为 heraC3113 的作业, 输入:

```
uustat -k heraC3113
```

5. 报告系统 hera 请求的作业状态, 输入:

```
uustat -s hera
```

系统使用消息响应如下:

```
heraN1bd7 07/15-12:09 S hera amy 522 /usr/amy/A
heraC1bd8 07/15-12:10 S hera amy 59 D.3b2a12ce4924
heraC3119 07/15-12:11 S hera amy rmail msg
```

## 文件

|                              |                   |
|------------------------------|-------------------|
| <code>/etc/locks</code>      | 包含禁止对设备多个使用的锁定文件。 |
| <code>/usr/bin/uustat</code> | 标明命令的路径名。         |
| <code>/var/spool/uucp</code> | 包含 BNU 状态信息。      |

## 相关信息

**cron** 守护程序、**ct** 命令、**cu** 命令、**echo** 命令、**stty** 命令、**uucleanup** 命令、**uucp** 命令、**uname** 命令、**uupick** 命令、**uuto** 命令和 **uux** 命令。

---

## uuto 命令

### 用途

从一个系统复制文件到另一个系统。

### 语法

```
uuto [ -m ] [ -p ] Source ... User
```

### 描述

**uuto** 命令是基本联网实用程序 (BNU) 命令, 它可将一个或多个 *Source* 文件从一个系统复制到另一个基于 UNIX 的系统上的指定 *User*。此程序使用 **uucp** 命令进行实际的文件传输, 但是 **uuto** 命令使收件人可以使用 **uupick** 命令选项在本地系统上处理传输来的文件。

发送方发出命令 **uuto** 来复制一个或多个文件到另一个系统的指定用户 ID。**uucp** 命令就复制文件到目标系统的 BNU 公共目录 `/var/spool/uucppublic` 中。**uucp** 命令还创建了另一个子目录 (称为 **receive**, 若它还没有存在) 和它的下一级目录 (在此目录中存放发送过来的文件直到收件人使用 **uupick** 命令检索它们)。复制过来的文件全路径名称如下形式:

```
/var/spool/uucppublic/receive/UserName/System/File
```

这里 *UserName* 和 *System* 目录被创建, 它们是基于 *User* 参数创建的, 此参数由命令 **uuto** 给出。

一旦复制过来的文件在目录 **receive** 中, **rmail** 命令就会通知接收方文件已到。然后接收方发出 **uupick** 命令, 此命令搜索公共目录查找发送过来的文件, 并通知收件人每个文件的位置。接收方输入一个 **uupick** 选项来处理文件。



## 源文件和目标文件的名称

发送方必须给出要发送文件的名称，以及文件要发往的用户和系统。*Source* 参数给出源文件的路径名称。如果文件的目录是 **uuto** 命令发出的目录，那么此参数可以是文件的名称。如果文件在另一个不同的目录中，那么文件的完整路径或相对路径必须给出。

*User* 参数是源文件要复制到的特定地址的路径名。此路径名必须包括接收方的用户 ID。*User* 参数格式如下：

*System!UserName*

*System* 是远程系统的名称，它与本地系统连接，*UserName* 是指定系统的传输文件接收方的登录名称。

当从一个用户复制文件到本地系统的另一个用户时，会省略 *System* 条目，目标就是文件要送往的用户 ID。系统名称只能包含 ASCII 字符。

## 标志

- m** 当源文件成功复制后，使用 **bellmail** 命令通知发送方。
- p** 复制源文件到本地系统的假脱机目录。源文件在 **uucp** 命令调用 **uucico** 守护程序以前会在假脱机目录中驻留一段时间（在 **uusched** 程序中定义），该守护程序实际发送副本到指定的远程系统的公共目录。缺省值是直接传输源文件到指定用户。

## 示例

1. 要复制文件到远程系统的用户，输入：

```
uuto /home/bin/file1 zeus!karen
```

在此示例中，`/home/bin/file1` 文件被发送到用户 `karen`，此用户在远程系统 `zeus` 上。

2. 要复制文件到远程系统的用户，无论源文件复制是否成功都发出通知，输入：

```
uuto -m /home/bin/file2 zeus!karen
```

在此示例中，`/home/bin/file2` 文件发送给用户 `karen`，此用户在远程系统 `zeus` 上，并返回发送方消息，验证复制成功。

3. 要复制文件到本地系统的另一个用户，输入：

```
uuto /home/bin/file3 ron
```

在此示例中，`/home/bin/file3` 文件被发送到本地系统的用户 `ron`。没有邮件消息发送给本地传输的接收方。

## 文件

`/usr/bin/uuto`  
`/var/spool/uucppublic`

包含 **uuto** 命令。  
是 BNU 公共目录。

## 相关信息

**bellmail** 命令、**ct** 命令、**cu** 命令、**uucp** 命令、**uuname** 命令、**uupick** 命令、**uustat** 命令和 **uux** 命令。

**uucico** 守护程序、**uusched** 守护程序。

---

## uutry 命令

### 用途

访问一个指定的远程系统，调试打开，允许用户重设缺省重试时间。

### 语法

```
uutry [ -xDebugLevel ] [ -r ] SystemName
```

### 描述

**uutry** 命令访问远程系统，远程系统由 *SystemName* 参数给出且使用调试模式。调试模式提供了一个监视“基本联网实用程序”（BNU）与远程计算机之间的连接以及文件传输的方法。**uutry** 命令调用 **uucico** 守护程序来访问远程系统。

调试输出在本地系统屏幕上滚动。一旦系统完成显示该信息，按“中断”键返回到提示状态。

如果第一次访问远程系统尝试失败，**-r** 标志可以重设缺省的重试时间。缺省重试时间为 5 分钟。

*SystemName* 参数是必需的参数，用来标明您想访问的远程系统的名称。

#### 注:

1. 您必须在 **/usr/sbin/uucp** 目录中发出 **uutry** 命令或者您在发出命令时使用全路径 **/usr/sbin/uucp/uutry**。
2. **uutry** 命令是存储在 **/usr/lib/uucp** 目录中的 shell 脚本。
3. 如果调试输出滚动速度太快而导致不能阅读，使用 **Uutry** 命令将输出保存到一个临时文件中。

### 标志

**-r** 覆盖缺省重试时间。如果由于某些原因，**uucico** 守护程序不能完成所要求的连接，则守护程序等候预定的时间之后会重试。缺省重试时间为 5 分钟。

注：远程系统最近被轮询的时间在 *SystemName* 文件中记录，此文件在目录 **/var/spool/uucp/Status** 中。

**-xDebugLevel** 重设由 **uutry** 命令显示在屏幕上的调试信息的详细资料数量。*DebugLevel* 变量的有效范围是 0 到 9，缺省值是 5。数字越大，最后的报告就会越详细。如果 **-x** 标志没有使用，**uucico** 守护程序就以缺省级别启动，产生中等的信息量。

### 示例

1. 改变详细信息数量，这些信息是由 **uutry** 命令提供的关于 **uucico** 的操作进度，使用 **-x** 标志指定不同的调试级别。例如，输入：

```
/usr/sbin/uucp/uutry -x9 venus
```

指示 **uutry** 命令生成尽可能多的关于 **uucico** 守护程序的工作情况信息。

2. 当第一次访问远程系统失败后，重试访问的缺省时间是 5 分钟。缩短访问远程系统的缺省重试时间，输入：

```
/usr/sbin/uucp/uutry -r venus
```

使用 **-r** 标志指示 **uucico** 守护程序去访问远程系统 **venus**，重设缺省重试时间。守护程序尝试访问系统 **venus**，周期性的重试直到连接成功，然后在本地系统的屏幕上显示调试输出。

## 文件

|                                           |                                            |
|-------------------------------------------|--------------------------------------------|
| <b>/usr/sbin/uucp/uutry</b>               | 包含 <b>uutry</b> 命令。                        |
| <b>/etc/uucp/Devices</b>                  | 包含有关可用设备的信息。                               |
| <b>/etc/uucp/Dialcodes</b>                | 包含拨号代码缩写。                                  |
| <b>/etc/uucp/Dialers</b>                  | 指定连接中的初次握手。                                |
| <b>/etc/uucp/Permissions</b>              | 描述远程系统的访问权限。                               |
| <b>/etc/uucp/Systems</b>                  | 描述可访问的远程系统。                                |
| <b>/var/spool/uucp/.Status/SystemName</b> | 列出最后一次由 <i>SystemName</i> 文件命名的远程系统被访问的时间。 |
| <b>/var/spool/uucppublic/*</b>            | 包含 BNU 公共目录。                               |

## 相关信息

**tail** 命令、**uucp** 命令、**Uutry** 命令、**uukick** 命令和 **uux** 命令。

**uucico** 守护程序、**uucpd** 守护程序。

《网络与通信管理》中的『监视 BNU 远程连接』和『监视 BNU 文件传输』。

《网络与通信管理》中的『维护 BNU』和『BNU 守护程序』。

---

## Uutry 命令

### 用途

在调试开启下，联系某指定的远程系统，保存调试输出到临时文件。

### 语法

**Uutry** [ **-xDebugLevel** ] [ **-r** ] *SystemName*

### 描述

**Uutry** 命令使用调试模式联系一个由 *SystemName* 参数指定的远程系统。调试模式提供了一个监视“基本联网实用程序”（BNU）与远程计算机之间的连接以及文件传输的方法。

**Uutry** 命令启动 **uucico** 守护程序，实际上由守护程序与指定系统联系。**uucico** 守护程序产生调试输出，使您能够在守护程序建立与远程系统之间的连接、执行远程登录以及传输文件时，对它的进度进行监视。

调试输出在本地系统屏幕上滚动。一旦系统完成显示该信息，按“中断”键返回到提示状态。

除了在屏幕上显示调试输出外，**Uutry** 命令还将此信息导入一个名为 **/tmp/SystemName** 的文件，其中 *SystemName* 参数是您试图联系的远程系统的名称。同样地，当所有的输出都显示之后，按“Interrupt”键返回到提示符。

*SystemName* 参数是必需的参数，用来标明您想访问的远程系统的名称。

注：

1. 当系统正在滚动由 **Uutry** 命令生成的输出时，按下 “Interrupt” 键将返回到提示符。 **uucico** 守护程序继续将调试信息放入 `/tmp/SystemName` 文件中。
2. 在发出 **Uutry** 命令时必须是在 `/usr/sbin/uucp` 目录下，否则，必须用全路径名发出命令，即 `/usr/sbin/uucp/Uutry`。
3. **Uutry** 命令是一个存储在 `/usr/sbin/uucp` 目录中的 shell 脚本。

## 标志

- r** 覆盖缺省重试时间。如果由于某些原因，**uucico** 守护程序无法完成所要求的连接，则守护程序等待一定的时间后会再次尝试。缺省重试时间为 5 分钟。
- xDebugLevel** 覆盖该命令在屏幕上显示的调试信息的详细信息数量的缺省值。`DebugLevel` 变量值的有效范围是从 0 到 9，缺省值是 5。数字越大则最后的相关报告就越详细。如果不使用 **-x** 标志，那么 **uucico** 守护程序以缺省级别启动，此级别产生适度的信息量。
- 注：** 远程系统最后受到轮询的时间记录在 `/var/spool/uucp/Status/SystemName` 文件中。

## 示例

要改变 **Uutry** 命令提供的关于 **uucico** 操作进度的详细信息数量，可以使用 **-x** 标志指定一个不同的调试级别。例如，输入：

```
/usr/sbin/uucp/Uutry -x9 venus
```

指示 **Uutry** 命令生成尽可能多的关于 **uucico** 守护程序正在工作的方法信息。

## 文件

|                                                     |                                  |
|-----------------------------------------------------|----------------------------------|
| <code>/tmp/SystemName</code>                        | 包含 <b>Uutry</b> 命令的调试输出（临时文件）。   |
| <code>/usr/sbin/uucp/Uutry</code>                   | 包含 <b>Uutry</b> 命令和 BNU 的所有配置文件。 |
| <code>/etc/uucp/Devices</code>                      | 包含有关可用设备的信息。                     |
| <code>/etc/uucp/Dialcodes</code>                    | 包含拨号代码缩写。                        |
| <code>/etc/uucp/Dialers</code>                      | 指定连接中的初次握手。                      |
| <code>/etc/uucp/Permissions</code>                  | 描述远程系统的访问权限。                     |
| <code>/etc/uucp/Systems</code>                      | 描述可访问的远程系统。                      |
| <code>/var/spool/uucp/Status/SystemName file</code> | 列出最后一次联系远程系统的时间。                 |
| <code>/var/spool/uucppublic/*</code>                | 包含 BNU 公共目录。                     |

## 相关信息

**uucico** 守护程序、**uucpd** 守护程序。

**tail** 命令、**uucp** 命令、**uustat** 命令、**uutry** 命令、**uukick** 命令和 **uux** 命令。

《网络与通信管理》中的『如何监视 BNU 远程连接』和『监视 BNU 远程连接』。

《网络与通信管理》中的『维护 BNU』和『理解 BNU 守护程序』。

---

## uux 命令

### 用途

在另外一个基于 UNIX 的系统上运行命令。

### 语法

```
uux [ -c | -C ] [ -n | -z ] [ - ] [ -aName ] [ -b ] [ -gGrade ] [ -j ] [ -p ] [ -e ] [ -r ] [ -sFile ] [ -xDebugLevel ] CommandString
```

### 描述

**uux** 命令是一种“基本联网实用程序”（BNU），在指定的基于 UNIX 的系统上运行指定的命令。同时不影响用户在本地系统上的工作。在运行指定命令之前，**uux** 命令从指定系统中收集所有必要的文件。用户可以将命令输出指向特定系统的特定文件。基于安全性的考虑，许多安装程序允许 **uux** 命令只运行 **rmail** 命令。

其他系统上的 **uux** 命令创建在本地系统运行命令的执行文件（**X.\***）。另外，为了能传输到其他系统，本地系统上的 **uux** 命令创建了命令文件（**C.\***）和数据文件（**D.\***）。执行文件包含要在指定系统中执行的命令字符串。命令文件包含 **uucp** 命令所创建的同样的信息。数据文件要么包含执行远程命令所需的数据，要么成为远程系统上的执行文件 **X.\*** 以执行远程命令。

执行文件的完整路径名如下所示：

```
/var/spool/uucp/System/X.SystemNxxxx
```

在 spooling 目录中创建好这些文件之后，**uux** 命令调用 **uucico** 守护程序将文件从本地系统的假脱机目录转移到指定的远程系统中。一旦文件被转移，该远程系统上的 **uuxqt** 守护程序执行指定系统上的 *CommandString*，将该命令的输出放置到原始 **uux** 命令要求所指定的文件中。

*CommandString* 自变量由一个或多个自变量组成，看上去就象操作系统的命令行，不同的是该 *CommandString* 自变量的前缀可能是远程系统的名称，采用 *System!* 的形式。缺省的 *System* 是本地系统。除非用户在输入 **uux** 命令时加入了 **-n** 标志，如果远程系统没有运行该命令，该命令就会通知该用户。远程系统将通过邮件的形式来响应。

### 源文件和目标文件的名称

- 当指定某个命令输出的目的地时，**uux** 命令的输入格式如下所示：
  - **uux** [*Options*] "*CommandString*> *Destination*"
  - **uux** [*Options*] *CommandString* \ *{Destination}* \.
- 目的地名称可以是以下任何一种：
  - 完整路径名
  - 前面带有 **~User** 的完整路径名，其中 *User* 是指定系统上的登录名。**uux** 命令用用户登录的目录替换了该路径名称。
- shell 模式匹配字符？（问号）、\*（星号）和 [ ... ]（括号）可以在源文件（比如由 **diff** 命令比较的文件）的路径名中使用；相应的系统将扩展这些符号。然而，使用 \* 字符可能会偶尔产生不可预料的结果。shell 模式匹配字符不应在目标路径名中使用。
- 用两个反斜杠（\ . . . \）或一对引号（" . . . "）将路径名中的模式匹配字符围起来，这样在 **uux** 命令将该命令发送到指定系统之前本地 shell 不会对它们加以解释。
- 如果您在路径名中使用了特殊的 shell 字符 ">"（大于符号）、"<"（小于符号）、";"（分号）或 "|"（竖线），请将该单个字符或整个命令字符串用符号 \ . . . \ 或符号 " . . . " 围起来。

- 不要在路径名中使用 shell 重定向字符 << 或 >>。
- **uux** 命令试图将命令行指定的所有文件移动到指定的系统中。将所有输出文件的名称围在圆括号中，这样 **uux** 命令就不会试图去转移它们。
- 当指定一个 *System* 时，总是将它放在条目中的 *CommandString* 参数前。系统名称只能包含 ASCII 字符。
- 命令中本地系统名前的 “!”（感叹号）是可选的。如果使用两个不同的远程系统中文件在本地系统运行命令时选择加入 “!”，则用 “!” 替换 *System!* 来代表本地系统，并且将 *System!* 作为到远程系统的路径名的第一项。
- 感叹号在 BUN 语法中代表一个系统，这与在 C shell 中的含义不同。在 C shell 中运行 **uux** 命令时，请将一个 \ (反斜杠) 放在系统名的感叹号前。

注：符号 ~ (波浪号) 是指定公共假脱机目录 **/var/spool/uucppublic** 的简略方式。

## 标志

- 将 **uux** 的标准输入作为 *CommandString* 变量的标准输入。
- a*Name* 用 *Name* 变量中指定的用户标识替换发出该命令的用户标识。
- b 如果退出状态不为零，将标准输入返回给该命令。
- c 将源文件转移到指定系统的目的地址。源文件被复制到假脱机目录，**uucico** 守护程序被立即调用。此标志是缺省值。
- C 将源文件转移到假脱机目录中。在设定的一段时间（在 **uusched** 程序中指定）之后，**uucico** 守护程序试图将文件转移到指定计算机的目的地址里。

在传输源文件时经常会遇到问题；例如，远程计算机可能没有正常工作或者可能出现登录失败的情况。在那种情况下，文件仍然留在假脱机目录里，直到它被成功传输出去或被 **uucleanup** 命令删除。

- e 启用文件扩展。
- g*Grade* 指定在特定连接中何时传输文件。*Grade* 变量指定一个整数（从 0 到 9）或字母（从 A 到 Z，a 到 z）；在 ASCII 码序列中，低序列的字符比高序列的字符使文件传输更优先。数字 0 处于最高（最优先）级别，z 处于最低（最迟）级别。缺省值为 **N**。
- j 显示在指定系统里运行命令的进程的作业的标识数。用 **BNUuustat** 命令和该作业标识来检查该命令的状态或用 **uustat -k** 标志来终止该进程。
- n 防止用 **mail** 命令通知用户某个命令的执行是成功还是失败。缺省情况是在命令执行失败时通知用户。
- p 将 **uux** 命令的标准输入用作 *CommandString* 参数的标准输入。- (减号) 产生同样的效果。
- r 防止启动假脱机程序在系统间传输文件。缺省设置是启动假脱机程序。
- s*File* 报告指定系统上 *File* 变量所指定的文件的传输状态。文件名只能包含 ASCII 码字符。
- x*DebugLevel* 在本地系统的屏幕显示调试信息。*DebugLevel* 变量的值必须是一个从 0 到 9 的整数。数字越大，则给出的报告越详细。
- z 通知用户命令的执行是否成功。此标志与系统缺省值相反，缺省情况是只在失败的时候通知用户。

## 示例

1. 要在远程系统上运行 **qprt** 命令，输入：

```
uux merlin!qprt /reports/memos/lance
```

在本例中，远程文件 `/reports/memos/lance` 在远程系统 `merlin` 上被显示。由于既没有指定 **-n** 标志，也没有指定 **-z** 标志，**uux** 命令只在远程系统运行命令失败时才通知用户。远程系统通过 **mail** 命令来发布该通知。

2. 要在两个远程系统上运行命令，在各个命令行上分别输入信息：

```
uux merlin!qprt /reports/memos/lance
uux zeus!qprt /test/examples/examp1
```

在本例中，远程文件/reports/memos/lance 在远程系统 merlin 中被显示，远程文件/test/examples/examp1 在远程系统 zeus 中被显示。由于既没有指定 **-n** 标志，也没有指定 **-z** 标志，**uux** 命令只在远程系统运行命令失败时才通知用户。远程系统通过 **mail** 命令来响应。

3. 要排队一个将本地系统文件和远程系统文件进行比较的作业，使用本地系统上的 **diff** 命令，获取该作业的作业标识号，输入：

```
uux -j "/usr/bin/diff /usr/amy/f1 hera!/home/amy/f2 > ~/f1.diff"
```

在本例中，本地系统上的文件 /usr/amy/f1 同远程系统 hera 上的文件 /home/amy/f2 进行了比较，输出信息放置在本地公共目录中的 f1.diff 文件里（完整路径名为 /var/spool/uucppublic/f1.diff）。输入目的地的名称时，前面必须有 ">" 符号，并且整个命令字符串用 " "（引号）围起来；或者整个目的地的名称用括号和反斜杠围起，如 `\{ DestinationName \}` 的形式。**-j** 标志导致 **uux** 命令返回该作业的 BNU 作业标识号。

4. 要使用本地系统中的 **diff** 命令比较两个不同的远程系统上的文件，输入：

```
uux "!/usr/bin/diff hera!/usr/amy/f1 venus!/home/amy/f2 > \ !f1.diff"
```

在本例中，远程系统 hera 中的文件 /usr/amy/f1 与远程系统 venus 中的文件 /home/amy/f2 进行了比较，输出信息将放置在本地系统的当前工作目录里的文件 f1.diff 中。

输出文件的状态必须是可写的。如果您对特定目标输出文件的权限状态不是很清楚，将该结果指向公共目录。代表本地系统的感叹号是可选的符号。输入目的地的名称时，前面必须有 ">"，整个命令字符串必须用 " "（引号）围起来；或者整个目的地的名称用括号和反斜杠围起，如 `\{ DestinationName \}` 的形式。

5. 要对来自不同系统的两个文件执行 **diff** 命令，输入：

```
uux "hera!/usr/bin/diff /tmp/out1 zeus/tmp/out2 > ~/DF"
```

在本例中，diff 文件在远程系统 hera 上。第一个源文件在远程系统 hera 上，第二个文件在系统 zeus 上。（zeus 可能是本地系统或另外一个远程系统。）输出被指向到本地系统公共目录下的 DF 文件中。

6. 要在一个不同的远程系统中指定一个输出文件，输入：

```
uux hera!uucp venus!/home/amy/f1 \{merlin!/home/geo/test\}
```

在本示例中，**uucp** 命令在远程系统 hera 上运行，作为 test，存储在系统 venus 上的文件 /home/amy/f1 发送给系统 merlin 上的用户 geo。输入的目的地名称要用括号和反斜杠围起。

7. 要从远程系统上的文件获取所选择的字段，并将它们放到本地系统中的文件，输入：

```
uux "cut -f1 -d: hera!\!/etc/passwd > ~/passw.cut"
```

在本例中，**cut** 命令在本地系统中运行。系统 hera 中的密码文件中每一行的第一个字段放置在本地系统公共目录下的 passw.cut 文件中。**uux** 命令在 C shell 中运行，所以 \（反斜杠）必须放在远程系统名称中的感叹号前。

8. 要使用 **uux** 管道选项来指定文件 /tmp/example 在系统 mercury 上的远程副本 /tmp/examplecopy，请使用以下语法：

```
uux -p mercury!
cp /tmp/example /tmp/examplecopy
```

用户必须键入 Ctrl-D 以便终止命令输入。按下 Ctrl-D 键后，该命令将进行假脱机以在系统 mercury 上远程执行。

## 文件

|                                    |                   |
|------------------------------------|-------------------|
| <code>/usr/bin/uux</code>          | 包含 <b>uux</b> 命令。 |
| <code>/var/spool/uucp</code>       | 是假脱机目录。           |
| <code>/var/spool/uucppublic</code> | 是公共目录。            |

## 相关信息

**ct** 命令、**cu** 命令、**mail** 命令、**rmail** 命令、**uucleanup** 命令、**uucp** 命令、**uuname** 命令、**uupick** 命令、**uustat** 命令、**uuto** 命令和 **qprt** 命令。

**sendmail** 守护程序、**uucico** 守护程序、**uuxqt** 守护程序。

---

## uuxqt 守护程序

### 用途

执行“基本联网实用程序”（BNU）远端命令请求。

### 语法

```
uuxqt [ -e ] [ -sSystemName ] [ -xDebugLevel ]
```

### 描述

“基本联网实用程序”（BNU）**uuxqt** 守护程序在指定的远程系统上执行命令。

每一个联网的系统上的 **uuxqt** 守护程序定期在假脱机目录中搜索远程执行 (**X.\***) 文件。这些文件由 **uucico** 守护程序发送到该目录以响应 **uux** 命令。

当它查找 **X.\*** 文件时，**uuxqt** 守护程序检查每个文件以确保：

- 所有要求的数据文件 (**D.\***) 都是可用的。
- 发出要求的系统有必要的权限以访问该数据文件和运行所请求的命令。

**注：****uuxqt** 守护程序使用 `/etc/uucp/Permissions` 文件来验证文件的可访问性和命令的执行权限。

如果数据文件存在，而且发出请求的系统有相应的权限，则 **uuxqt** 守护程序执行命令。

**注：****uuxqt** 命令通常从一个 shell 过程，**uudemon.hour** 命令执行，而不是从命令行输入。您必须具有 root 用户特权以从命令行发出 **uuxqt** 命令。

### 标志

|                     |                                                                                                             |
|---------------------|-------------------------------------------------------------------------------------------------------------|
| <b>-e</b>           | 启用文件扩展。                                                                                                     |
| <b>-sSystemName</b> | 指定要联系的远程系统。只有在手动启动 <b>uuxqt</b> 命令时才使用该标志。系统名称在 <b>uuxqt</b> 命令自动启动时是内部提供的。<br><b>注：</b> 系统名称只能包含 ASCII 字符。 |
| <b>-xDebugLevel</b> | 在本地系统的屏幕显示调试信息。 <i>DebugLevel</i> 变量值是一个从 0 到 9 的整数，缺省值是 5。 <i>DebugLevel</i> 变量的值越大，调试信息就越详细。              |



## 安全性

访问控制: 您必须具有 root 用户权限以从命令行启动 **uuxqt** 守护程序。

## 示例

要启动 **uuxqt** 守护程序进行调试, 输入:

```
/usr/sbin/uucp/uuxqt -svenus -x7
```

这指示该命令联系远程系统 **venus** 并提供有关此次联系的相当详细的信息。

## 文件

**/usr/sbin/uucp/uuxqt**

**/etc/locks**

**/etc/uucp/Maxuuxqts**

**/etc/uucp/Permissions**

**/var/spool/uucp/\***

包含 **uuxqt** 守护程序

包含了防止对设备的多重使用和对系统的多重调用的锁定文件。

限制远程命令的执行。

描述远程系统的访问权限。

包含执行文件和数据文件。

## 相关信息

**uucp** 命令、**uudeemon.hour** 命令、**uustat** 命令和 **uux** 命令。

**cron** 守护程序、**uucico** 守护程序。

《网络与通信管理》中的『理解 BNU 文件和目录结构』、『理解 BNU 守护程序』和『理解 BNU 安全性』。



---

## 附录. 声明

本信息是为在美国提供的产品和服务编写的。

IBM 可能在其他国家或地区不提供本文档中讨论的产品、服务或功能特性。有关您当前所在区域的产品和服务的信息，请向您当地的 IBM 代表咨询。任何对 IBM 产品、程序或服务的引用并非意在明示或暗示只能使用 IBM 的产品、程序或服务。只要不侵犯 IBM 的知识产权，任何同等功能的产品、程序或服务，都可以代替 IBM 产品、程序或服务。但是，评估和验证任何非 IBM 产品、程序或服务，则由用户自行负责。

IBM 公司可能已拥有或正在申请与本文档所述内容有关的各项专利。提供本文档并未授予用户使用这些专利的任何许可。您可以用书面方式将许可查询寄往：

IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785  
U.S.A.

**本条款不适用英国或任何这样的条款与当地法律不一致的国家或地区：** International Business Machines Corporation “按现状”提供本出版物，不附有任何种类的（无论是明示的还是暗含的）保证，包括但不限于暗含的有关非侵权、适销和适用于某种特定用途的保证。某些国家或地区在某些交易中不允许免除明示或暗含的保证。因此本条款可能不适用于您。

本信息中可能包含技术方面不够准确的地方或印刷错误。此处的信息将定期更改；这些更改将编入本资料的新版本中。IBM 可以随时对本资料中描述的产品和 / 或程序进行改进和 / 或更改，而不另行通知。

本程序的被许可方如果要了解有关程序的信息以达到如下目的：(i) 允许在独立创建的程序和其他程序（包括本程序）之间进行信息交换，以及 (ii) 允许对已经交换的信息进行相互使用，请与下列地址联系：

IBM Corporation  
Dept. LRAS/Bldg. 003  
11400 Burnet Road  
Austin, TX 78758-3498  
U.S.A.

只要遵守适当的条件和条款，包括某些情形下的一定数量的付费，都可获得这方面的信息。

本资料中描述的许可程序及其所有可用的许可资料均由 IBM 依据 IBM 客户协议、IBM 国际软件许可协议或任何同等协议中的条款提供。

有关双字节（DBCS）信息的许可查询，请与您所在国家或地区的 IBM 知识产权部门联系，或用书面方式将查询寄往：

IBM World Trade Asia Corporation  
Licensing  
2-31 Roppongi 3-chome, Minato-ku  
Tokyo 106, Japan

IBM 可以按它认为适当的任何方式使用或分发您所提供的任何信息而无须对您承担任何责任。

涉及非 IBM 产品的信息可从这些产品的供应商、其出版说明或其他可公开获得的资料中获取。IBM 没有对这些产品进行测试，也无法确认其性能的精确性、兼容性或任何其他关于非 IBM 产品的声明。有关非 IBM 产品性能的问题应当向这些产品的供应商提出。

本信息中对非 IBM Web 站点的任何引用都只是为了方便起见才提供的，不以任何方式充当对那些 Web 站点的保证。那些 Web 站点中的资料不是 IBM 产品资料的一部分，使用那些 Web 站点带来的风险将由您自行承担。

本信息包含日常商业运作所用到的数据和报表示例。为了尽可能完整地演示它们，示例中包括了个人、公司、品牌和产品的名称。所有这些名称都是虚构的，如果和实际的商务企业所用的名称和地址雷同，则纯属配合。

版权许可：

本信息包括源语言形式的样本应用程序，这些样本说明不同操作平台上的编程方法。如果是为按照在编写样本程序的操作平台上的应用程序编程接口（API）进行应用程序的开发、使用、经销或分发为目的，您可以任何形式对这些样本程序进行复制、修改、分发，而无须向 IBM 付费。这些示例并未在所有条件下作全面测试。因此，IBM 不能担保或暗示这些程序的可靠性、可维护性或功能。用户如果是为了按照 IBM 应用程序编程接口开发、使用、经销或分发应用程序，则可以任何形式复制、修改和分发这些样本程序，而无须向 IBM 付费。

此处包含的任何性能数据都是在受控环境中测得的。因此，在其他操作环境中获得的数据可能会有明显的不同。有些测量可能是在开发级的系统上进行的，因此不保证与一般可用系统上进行的测量结果相同。此外，有些测量是通过推算而估计的。实际结果可能会有差异。文档的用户应当验证其特定环境的适用数据。

---

## 商标

下列术语是 International Business Machines Corporation 在美国和 / 或其他国家或地区的商标：

AIX  
AIX 5L  
DPI  
HACMP  
IBM  
Micro Channel  
PowerPC  
System/370

Java 和所有基于 Java 的商标和徽标是 Sun Microsystems, Inc. 在美国和 / 或其他国家或地区的注册商标。

Linux 是 Linus Torvalds 在美国和 / 或其他国家或地区的商标。

UNIX 是 The Open Group 在美国和 / 或其他国家或地区的注册商标。

Microsoft、Windows、Windows NT 和 Windows 徽标是 Microsoft Corporation 在美国和 / 或其他国家或地区的商标。

其他的公司、产品或服务名称可能是其他公司的商标或服务标志。

# 索引

## [ A ]

### 安全

审计系统的安全状态

通过 sysck 命令 313

### 安装

安全系统中的文件

用 sysck 命令验证 313

安装专用密钥环 101

## [ B ]

### 编辑器

#### 流

使用 sed 命令 51

tvi 编辑器, 调用

使用 tvi 命令 511

### 编辑状态

正在显示 4

### 变量

tip 命令 368

设置 367

### 标识、用户

与会话关联的

使用 su 命令 228

### 标准输入

创建 typescript 40

复制到一个文件 331

### 表

格式化 nroff 命令

使用 tbl 命令 309

格式化 troff 命令

使用 tbl 命令 309

### 表达式

评估条件 348

### 拨号代码文件格式

#### 建立

使用 uucpadm 命令 583

## [ C ]

### 参考文献资料数据库

#### 排序

使用 sortbib 命令 145

### 查询特征

#### 终端

使用 stty 命令 218

### 超级块

更新 554

### 程序

将输出复制到一个文件 331

### 程序循环

返回退出值 490

### 程序、管理

#### SCCS 命令

使用 sccs 命令 19

处理接收的邮件, MH 93

### 处理资源分配

除去不用的模块 90

### 磁带设备

#### 复制

tcopy 命令 318

将子命令提供给一个流

使用 tctl 命令 328

#### 一致性校验

tapechk 命令 303

### 错误

在文件中修订

使用 tcbck 命令 316

## [ D ]

### 打印机

#### 更改驱动程序设置

用 splp 命令 163

#### 显示驱动程序设置

用 splp 命令 163

### 代码集 70

### 代码集映射

设置 70

### 导入

#### 认证中心的证书

使用 smimpcacert 命令 99

### 对象文件

#### 查找可显示的字符串

使用 strings 命令 205

减少 XCOFF 的大小 206

使用 stripnm 命令显示符号信息 208

显示 XCOFF 的节大小 86

## [ E ]

### 二进制数据

#### 存储在文件中

使用 sal 命令 3

### 二进制文件

#### 查找可显示的字符串

使用 strings 命令 205

## 二进制文件 (续)

### 邮件传输的编码

使用 uuencode 命令 586, 593

### 邮件传输的译码

使用 uuencode 命令 586, 593

## [ F ]

### 分析虚拟内存快照

#### 内存管理

使用 svmon 命令 244

### 服务器

#### 专用密钥环文件, 生成

使用 smgenprivkr 命令 98

### 服务器属性列表 106

### 服务器证书导入 100

### 服务器专用密钥

#### 生成

使用 smgenkeycr 命令 97

## [ G ]

### 格式化文本

#### 在排版设备上显示

使用 troff 命令 432

### 跟踪报告

#### 删除格式模板

使用 trcupdate 命令 431

#### 替换格式模板

使用 trcupdate 命令 431

#### 添加格式模板

使用 trcupdate 命令 431

### 跟踪缓冲区

从系统转储映像中抽取 420

### 跟踪会话

#### 结束

使用 trcstop 命令 430

### 跟踪日志

#### 格式化报告, 来自

使用 trcrpt 命令 424

### 跟踪事件组

#### 更改

使用 trcevgrp 命令 422

#### 删除

使用 trcevgrp 命令 422

#### 添加

使用 trcevgrp 命令 422

### 跟踪, 关闭

#### 子服务器或子系统

使用 tracesoff 命令 415

### 更新

#### 安全系统中的文件

用 sysck 命令验证 313

### 管理程序

#### SCCS 命令

使用 sccs 命令 19

### 管理信息库变量

使用 snmpinfo 命令管理 129

### 归档

使用 tar 命令 304

## [ H ]

### 行

#### 删除重复

使用 uniq 命令 545

### 核心文件

搜集核心文件 116

核心转储大小极限 527

## [ J ]

### 计数器

#### 保存操作系统数据

使用 sadc 命令 5

#### 处理操作系统数据

使用 sadc 命令 5

#### 从操作系统采样数据

使用 sadc 命令 5

### 记帐系统

#### 打开

使用 turnacct 命令 509

#### 关闭

使用 shutacct 命令 83

使用 turnacct 命令 509

#### 启动

使用 startup 命令 181

#### 总结记录

使用 sa 命令 1

### 加下划线

#### 执行

使用 ul 命令 527

### 假脱机目录 576

### 假脱机目录, BNU

#### 清除

使用 uudemon.cleanu 命令 588

### 检查

#### 安全系统中的文件安装

使用 tcbck 命令 313

### 将标准输入终端线路转换

到串行线路网际协议 (SLIP) 链路

使用 sliplogin 命令 90

- 交换 272
  - 指定附加的设备 272
- 脚本
  - snap 114
  - snmpevent 120
  - topsvcs 386
  - topsvcsctrl 388
- 校验和
  - 显示文件的
    - 使用 sum 命令 231
- 进程
  - 初始化
    - 使用 init 命令 332
    - 使用 telinit 命令 332
- 进程挂起
  - 在时间间隔中执行暂挂 89

## [ K ]

- 可靠的可伸缩集群技术 (RSCT) 拓扑服务
  - 脚本
    - topsvcs 386
  - 控制脚本
    - topsvcsctrl 388
- 可信的编辑器
  - 提供
    - 使用 tvi 命令 511
- 可信的 shell
  - 调用 497
  - 在中解释命令
    - 使用 tsh 命令 496
  - Korn shell
    - 之间的不同 497
- 空格
  - 更改为制表符
    - 使用 tab 命令 293
  - 由制表符改为
    - 使用 untab 命令 554
- 控制脚本
  - topsvcsctrl 388
- 控制、有限的
  - BNU 操作的
    - 使用 uustat 命令 607
- 块计数
  - 显示文件的
    - 使用 sum 命令 231

## [ L ]

- 连接串行线路
  - TCP/IP
    - 使用 slattach 命令 87

- 连接到远程系统
  - BNU
    - 使用 tip 命令 365
- 联网的计算机
  - 显示列表 598
- 令牌环设备驱动程序
  - 显示统计信息 370
- 轮询
  - 远程系统
    - 使用 uudemmon.poll 命令 591
- 轮询文件格式
  - 建立
    - 使用 uucpadm 命令 583
- 逻辑卷
  - 除去镜像
    - 使用 unmirrorvg 命令 551
  - 分割与副本 161
  - 将一卷复制到一新建的卷 276
  - 同步非当前的镜像
    - 使用 syncvg 命令 277

## [ M ]

- 每日报告
  - 写入文件
    - 使用 sa2 命令 3
- 密钥环文件, 生成
  - 服务器专用的
    - 使用 smgenprivkr 命令 98
- 命令
  - 报告所用时间
    - 使用 timex 命令 364
  - 报告用户时间
    - 使用 timex 命令 364
  - 显示执行时间
    - 使用 time 命令 359
- sccs 19
- sccshelp 24
- slattach 87
- slipllogin 90
- smdemon.cleanu 96
- smexpcacert 97
- smgenkeycr 97
- smgenprivkr 98
- smimpcacert 99
- snapshot 117
- snmpevent 120
- startcondresp 170
- startprdomain 174
- startprnode 177
- stopcondresp 188
- stopprdomain 190

## 命令 (续)

stoprnode 192  
stty 218  
su 228  
sum 231  
svmon 244  
tbl 309  
tcbck 313  
timedc 362  
tip 365  
touch 391  
tr 404  
tracesoff 415  
trcevgrp 422  
troff 432  
tset 494  
tsh 496  
tsm 498  
tunchange 502  
tuncheck 504  
tvi 511  
type 516  
unfencevsd 541  
updatevsdnode 555  
updatevsdtab 557  
updatevsdvg 558  
usrck 567  
uucpadm 583  
uudemon.admin 587  
uudemon.cleau 588  
uudemon.hour 590  
uudemon.poll 591  
uuq 602  
uustat 607

命令类型和变量的描述  
使用 type 命令 516

## 目录

卸载  
使用 umount 命令 532

## [ N ]

### 内部认证中心

定义 95  
取消配置 111

### 内存管理

分析虚拟内存快照  
使用 svmon 命令 244  
更新超级块 554

### 内核名称列表

生成一个 423

### 内核消息

写到终端 560

## [ P ]

### 排版设备, 显示

格式化文本  
使用 troff 命令 432

### 拼写列表

创建  
使用 spellin 命令 150  
示例 150  
验证词不在  
使用 spellout 命令 150  
示例 151

### 屏幕

创建 typescript 40  
将显示复制到文件 331

## [ Q ]

### 启动传输调用

使用 BNU 程序  
使用 uudemon.hour 命令 590

### 清除

sendmail 队列  
使用 smdemon.cleau 命令 96

## [ R ]

### 认证中心

只读信息 94  
认证中心的证书  
导入  
使用 smimpcacert 命令 99

### 日志文件, BNU

清除  
使用 uudemon.cleau 命令 588

### 日志文件 (BNU)

清除 596

### 日志, 跟踪

格式化报告, 来自  
使用 trcrpt 命令 424

## [ S ]

### 删除条目

BNU 作业队列  
使用 uuq 命令 602



- 设备
  - 定制
    - 保存有关信息 13
- 设备配置命令
  - savebase 13
- 设备文件格式
  - 建立
    - 使用 uucpadm 命令 583
- 设置特征
  - 终端
    - 使用 stty 命令 218
- 审计
  - 安全系统中的文件安装
    - 通过 sysck 命令 313
- 时间戳记
  - 更新
    - 访问时间 391
    - 修改时间 391
- 守护程序
  - tftpd 355
  - utmpd 572
- 数据区域大小极限 527
- 术语信息描述符文件。
  - 从源格式转换成编译格式 358
- 算术
  - 转换单位 547

## [ T ]

- 条件表达式
  - 评估 348
- 退出值
  - 返回 490
- 拓扑服务子系统
  - 脚本
    - topsvcs 386
  - 控制脚本
    - topsvcsctrl 388

## [ W ]

- 为本地或网络交付传送邮件 61
- 文件
  - 比较二者
    - 使用 sdiff 命令 42
  - 除去 ifdef 行 543
  - 访问时间
    - 更新 391
  - 分割成几段
    - 使用 split 命令 151
  - 合并
    - 使用 sort 命令 139

- 文件 (续)
  - 解压 538
  - 排序
    - 使用 sort 命令 139
  - 删除重复行
    - 使用 uniq 命令 545
  - 为无序列表排序 499
  - 显示二者的并排比较
    - 使用 sdiff 命令 42
  - 显示校验和
    - 使用 sum 命令 231
  - 显示块计数
    - 使用 sum 命令 231
  - 写
    - 从指定点 298
  - 卸载
    - 使用 umount 命令 532
  - 修订错误
    - 使用 tcbck 命令 316
  - 修改时间
    - 更新 391
  - 压缩 539
  - 用 tftp 命令传输 350
  - 在系统间的复制 580
  - 展开
    - 使用 unpack 命令 552
  - SCCS
    - 比较两个版本 23
    - 取消指定的版本 542
    - 显示编辑状态 4
- 文件包含
  - 处理
    - 使用 soelim 命令 138
- 文件传输, BNU
  - 状态
    - uudemon.admin 命令 587
- 文件大小极限 527
- 文件方式创建掩码 529
- 文件夹
  - 将消息显示在
    - 使用 scan 命令 18
- 文件系统
  - 除去不需要的文件
    - 使用 skulker 命令 87
  - 卸载
    - 使用 umount 命令 532
- 物理存储器大小极限 527

## [ X ]

### 系统

#### 结束操作

使用 shutdown 命令 84

#### 显示的正常运行时间

使用 uptime 命令 560

### 系统调用

#### 执行 278

系统调用接口程序 278

### 系统管理

#### 执行

使用 smit 命令 101

使用 smitty 命令 104

### 系统活动

#### 保存

使用 sar 命令 6

#### 报告

使用 sar 命令 6

#### 收集

使用 sar 命令 6

### 系统控制台

#### 暂时重定向到一个设备

使用 swcons 命令 274

#### 暂时重定向到一个文件

使用 swcons 命令 274

### 系统文件格式

#### 建立

使用 uucpadm 命令 583

### 系统转储

#### 保存 14

### 系统状态

在终端状态行显示 289

### 系统资源控制器

#### 启动守护程序

使用 srcmstr 守护程序 168

### 消息

#### 发送

使用 send 命令 59

#### 记录系统 290

#### 列表行 18

#### 排序

使用 sortm 命令 146

#### 显示

使用 show 命令 80

消息路由 165

### 消息, SCCS

#### 显示帮助信息

使用 sccshelp 命令 24

### 写

带有恢复的制表符 540

### 写入样式

#### 分析

使用 style 命令 227

## [ Y ]

### 验证

#### 安全系统中的文件安装

通过 sysck 命令 313

#### 用户定义的正确性

usrck 命令的使用 567

页面调度 272

#### 指定附加的设备

使用 swapon 命令 272

### 因特网

跟踪因特网信息包 413

### 用户

#### 重新初始化登录会话

使用 shell 命令 78

#### 复位受保护的状态环境

使用 setenv 命令 73

#### 更改会话标识

使用 su 命令 228

### 用户定义

#### 验证正确性

usrck 命令的使用 567

### 邮寄错误报表

邮寄 61

### 邮件命令

sendbug 61

sendmail 61

smdemon.cleanu 96

### 游戏

tic-tac-toe 500

游戏目录权限 510

### 与其他用户对话

使用 talk 命令 300

### 远程命令请求

执行 618

### 远程系统

#### 连接到

使用 tip 命令 365

#### 轮询

使用 uudemmon.poll 命令 591

执行命令 618

### 允许文件的格式

#### 建立

使用 uucpadm 命令 583

验证 573

## [ Z ]

- 在排版设备上显示
  - 格式化文本
    - 使用 troff 命令 432
- 正确性, 验证
  - 用户定义
    - usrck 命令的使用 567
- 证书请求
  - 处理和生成 108
  - 生成
    - 使用 smgenkeycr 命令 97
- 制表符
  - 从空格更改
    - 使用 tab 命令 293
  - 改为空格
    - 使用 untab 命令 554
- 终端
  - 操作内核消息 560
  - 查询特征
    - 使用 stty 命令 218
    - 使用 termdef 命令 347
  - 初始化
    - 使用 tset 命令 494
  - 将路径名写到标准输出 501
  - 设置特征
    - 使用 stty 命令 218
    - 使用 tset 命令 494
  - 设置制表符停止位 294
  - 指定波特率
    - 使用 tset 命令 494
- 终端对话期
  - 制作 typescript 40
- 终端映射
  - 设置 70
- 终端状态管理器
  - 调用
    - 使用 tsm 命令 498
- 主机
  - 连接本地和远程
    - 使用 telnet 命令 335
    - 使用 tn 命令 335
    - 使用 tn3270 命令 335
- 主题列表
  - 生成
    - 使用 subj 命令 230
- 转储设备
  - 更改辅助 284
  - 更改主要 284
  - 开始内核转储至辅助设备 288
  - 开始内核转储至主要设备 288

- 转换
  - 字符
    - 使用 tr 命令 404
- 转义符号
  - 使用 tip 命令 366
- 状态、报告
  - BNU 操作的
    - 使用 uustat 命令 607
- 资源极限 527
- 子服务器
  - 打开跟踪
    - 使用 traceson 命令 416
  - 关闭跟踪
    - 使用 tracesoff 命令 415
  - 启动
    - 使用 startsrc 命令 179
  - 停止
    - 使用 stopsrc 命令 194
- 子例程调用
  - 执行 278
- 子例程调用接口程序 278
- 子系统
  - 打开跟踪
    - 使用 traceson 命令 416
  - 关闭跟踪
    - 使用 tracesoff 命令 415
  - 启动
    - 使用 startsrc 命令 179
  - 停止
    - 使用 stopsrc 命令 194
- 字符
  - 转换
    - 使用 tr 命令 404
- 字符翻译 417

## [ 数字 ]

- 128 端口异步控制器
  - 查询特征 225
  - 设置特征 225

## A

- acct/\* 命令
  - shutacct 83
  - startup 181
  - turnacct 509
- aliases
  - 除去 534

## B

### BNU

#### 调试

将输出保存到文件 613

#### 调试模式

使用 595

调试远程连接 575

#### 假脱机目录

除去文件 576

从删除文件 578

清除 576, 578

#### 检查操作状态

使用 uustat 命令 607

#### 联网的计算机

列表 598

轮询远程系统 601

使用 uudemon.poll 命令 591

#### 命令

远程运行 615

远程执行 618

#### 配置信息, 输入

使用 uucpadm 命令 583

#### 启动传输调用

使用 uudemon.hour 命令 590

#### 清除假脱机目录

使用 uudemon.cleau 命令 588

#### 清除日志文件

使用 uudemon.cleau 命令 588

#### 日志文件

清除 596

正在显示 596

#### 文件

调度传送 604

完成传输 599

系统之间的传输 575

在系统间的复制 610

转换成 ASCII 593

#### 文件传送的状态

uudemon.admin 命令 587

#### 要求的文件

要求检查 573

#### 远程系统

在调试模式下联系 613

在操作系统间复制文件 580

#### 状态

获取 606

TCP/IP 间的通信 586

tip 命令 365

变量 368

转义符号 366

uucheck 命令 573

### BNU (续)

uucico 守护程序 575

uuclean 命令 576

uucleanup 命令 578

uucp 580

uucpadm 命令 583

uucpd 守护程序 586

uudemon.admin 命令 587

uudemon.cleau 命令 588

uudemon.hour 命令 590

uudemon.poll 命令 591

uukick 命令 595

uulog 命令 596

uupoll 命令 601

uuq 命令 602

uusched 守护程序 604

uusend 命令 605

uusnap 命令 606

Uutry 命令 613

uuxqt 守护程序 618

### BNU 作业队列

#### 删除条目

使用 uuq 命令 602

## C

### CA 证书

#### 导入

使用 smimpcacert 命令 99

#### CA 证书导出

使用 smexpcacert 命令 97

CA 证书列表 104

### CPU

报告用法 393

## E

### ERRM 脚本

snmpevent 120

### ERRM 命令

snmpevent 120

## F

### FORTRAN

将程序转换为 RATFOR 216

## G

groups

- 重新设置当前的登录会话
- 使用 setgroups 命令 69

## H

hlptcpdump 318  
hlpuil 524

## I

iconv 库

- 生成的转换表 520

inetd 守护程序

- uucpd 守护程序和 586

init 命令 332  
ip 安全性加密模块 550  
i-node 表

- 更新
- 使用 sync 命令 276

## M

MH

- slocal 命令 93
- spost 命令 165

## N

NCS 命令

- uuid\_gen 594

NFS 命令

- showmount 82
- spray 166

NFS 守护程序

- sprayd 167
- statd 185

nroff 命令

- 格式化表
- 使用 tbl 命令 309

## P

phones 文件格式

- 设置 phones 变量 369

## R

remote 文件格式

- 设置 remote 变量
- 使用 tip 命令 369

RSCT 拓扑服务

- 脚本
- topsvcs 386
- 控制脚本
- topsvcsctrl 388

## S

sa 命令 1  
sa1 命令 3  
sa2 命令 3  
sadc 命令 5  
sar 命令 6  
savebase 命令 13  
savecore 命令 14  
savevg 命令 15  
scan 命令 18  
SCCS

- 帮助信息 24
- 命令
- 管理 20
- 文件
  - 比较两个版本 23
  - 管理 20
  - 取消指定的版本 542
  - 显示编辑状态 4

SCCS 命令

- 管理程序
- 使用 sccs 命令 19
- 显示帮助信息
- 使用 sccshelp 命令 24

sact 4  
sccs 19  
sccsdiff 23  
sccshelp 24  
unget 542  
sccs 命令 19  
SCCS 消息

- 显示帮助信息
- 使用 sccshelp 命令 24

sccshelp 命令 24  
schedo 命令 25  
scls 命令 39  
sctpctrl 命令 40  
sdiff 命令 42  
secldapclntd 45  
secldifconv 命令 46

sectoldif 命令 48

securetcip 命令 50

sed 命令 51

sedmgr 命令 55

sendbug 命令 61

sendmail 队列  
清除  
    使用 smdemon.cleau 命令 96

sendmail 命令 61

setclock 命令 67

setea 命令 68

setgroups 命令 69

setmaps 命令 70

setsenv 命令 73

settime 命令 75

setuname 命令 76

sh 命令 77

shell  
    缺省 77  
    用登录凭证执行  
        使用 shell 命令 78

shell 脚本  
    程序循环  
        返回退出值 490

shell 命令 78

show 命令 80

showmount 命令 82

shutacct 命令 83

shutdown 命令 84

size 命令 86

skulker 命令 87

slattach 命令 87

SLIP 网络接口  
    将标准输入终端线路转换到  
        使用 sliplogin 命令 90  
    slattach 命令 87

sliplogin 命令 90

slocal 命令 93

smcaprop 命令 94

smdefca 命令 95

smdemon.cleau 命令  
    邮件 96

smexpcacert 命令 97

smgenkeycr 命令 97

smgenprivkr 命令 98

smimpcacert 命令 99

smimpservercert 命令 100

sminstkey 命令 101

smit 命令 101

smitty 命令 104

smit.log 文件  
    重定向 101, 104

smit.script 文件  
    重定向 101, 104

smlistcerts 命令 104

smrsh 命令 106

smserverprop 命令 106

smsigncert 命令 108

smtctl 命令 109

smundefca 命令 111

snap 脚本 114

snap 命令 112

snapshot 命令 117

snapsplit 命令 119

SNMP  
    切换 snmpd 代理守护程序的版本 135

SNMP V1  
    代理应用程序  
        snmpdv1 命令 123  
    作为后台进程启动 SNMP V1 代理程序 123

SNMP V3  
    代理应用程序  
        snmpdv3 命令 127  
    作为后台进程启动 SNMP V3 代理守护程序 127

snmpd 守护程序 123

snmpdv1 守护程序 123

snmpdv3 守护程序 127

snmpevent 脚本 120

snmpevent 命令 120

snmpinfo 命令 129

snmpmibd 守护程序 132

snmptrap 命令 134

snmpv3\_ssw 命令 135

SNOBOL  
    编译与解释 136

sodebug 命令 137

soelim 命令 138

sort 命令 139

sortbib 命令 145

sortm 命令 146

spell 命令 148  
    维护散列列表 148

spellin 命令 150

spellout 命令 150

splat 152

split 命令 151

splitlvcopy 命令 161

splitvg 命令 160

splp 命令 163

splist 命令 165

spray 命令 166

sprayd 守护程序 167

srcmstr 守护程序 168

startcondresp 命令 170

startprdomain 命令 174  
 startprnode 命令 177  
 startsrc 命令 179  
 startup 命令 181  
 startx 命令 183  
 start-secldapclntd 172  
 statd 守护程序 185  
 stopcondresp 命令 188  
 stopprdomain 命令 190  
 stopprnode 命令 192  
 stopsrc 命令 194  
 stop-secldapclntd 174  
 stpinet 方法 197  
 strace 命令 198  
 strchg 命令 199  
 strclean 命令 201  
 strconf 命令 201  
**STREAMS**  
   可调参数 214  
   显示信息 204  
**STREAMS 工具**  
   错误记录器  
     清除 201  
   错误日志  
     接收消息 202  
   跟踪消息  
     显示 198  
   可移植环境  
     装入和配置 210  
   模块  
     列表 39  
   配置  
     查询 201  
     更改 199  
   驱动程序名  
     列表 39  
   strerr 守护程序 202  
**STREAMS 命令**  
   scls 39  
   strace 198  
   strchg 199  
   strclean 201  
   strconf 201  
   strload 210  
   strerr 守护程序 202  
   strinfo 命令 204  
   strings 命令 205  
   stripnm 命令 208  
   strload 命令 210  
   strreset 命令 213  
   strtune 命令 214  
   sttinet 方法 217  
   stty 命令 218  
   stty-cxma 命令 225  
   style 命令 227  
   su 命令 228  
   subj 命令 230  
   sum 命令 231  
   suma 命令 232  
   survd 守护程序 242  
   svmon 命令 244  
     段报告 252  
     工作负载管理报告 250  
     层报告 250  
     类报告 250  
     进程报告 251  
     命令报告 249  
     全局报告 247  
     详细报告 257  
     用户报告 248  
     帧报告 257  
   swap 命令 270  
   swapon 命令 272  
   swcons 命令 274  
   swts 命令 275  
   sync 命令 276  
   synclvodm 命令 276  
   syncvg 命令 277  
   sysck 命令 280  
   syscorepath 命令 282  
   sysdumpdev 命令 284  
   sysdumpstart 命令 288  
   sysline 命令 289  
   syslogd 守护程序 290  
  
**T**  
   tab 命令 293  
   talk 命令 300  
   talkd 守护程序 301  
   tar 命令 304  
   tbl 命令 309  
   tc 命令 312  
   tcback 命令 313  
     安全 316  
     操作方式  
       更新方式 314  
       检测方式 314  
   tcpdump 命令 318  
**TCP/IP**  
   安全性功能  
     启用 50  
   方法  
     udefinet 521

## TCP/IP (续)

### 服务器功能

支持 talk 命令 301

支持 TELNET 协议 345

跟踪包 485

连接串行线路

使用 slattach 命令 87

时间服务器守护程序

调用 360

因特网实例

取消定义 522

卸装 517

转换到 SLIP

使用 sliplogin 命令 90

hosts

设置时间与日期 67

inet 实例

禁用 197

启用 217

TCP 套接字

跟踪 485

TFTP 服务器功能

使用 tftpd 守护程序 355

timed 守护程序信息

使用 timedc 命令 362

tracing 因特网信息包 413

## TCP/IP 方法

stpinet 197

sttinet 217

ucfgif 517

ucfginet 517

udefinet 522

## TCP/IP 命令

securetcpip 50

setclock 67

slattach 87

sliplogin 90

tftp 350

timedc 362

traceroute 413

trpt 485

utftp 350

## TCP/IP 守护程序

talkd 301

telnetd 345

tftpd 355

timed 360

tee 命令 331

telinit 命令 332

telnet 命令 335

## TELNET 协议

### 实现

使用 telnet 命令 335

使用 tn 命令 335

使用 tn3270 命令 335

telnetd 守护程序 345

telnet 选项 345

termdef 命令 347

test 命令 348

tftp 命令 350

tftpd 守护程序 355

tic 命令 358

time 命令 359

timed 守护程序 360

获取有关信息

使用 timedc 命令 362

用 SRC 操作 361

timedc 命令 362

变量 363

timex 命令 364

tip 命令 365

变量 368

转义符号 366

phones 文件格式

设置 phones 变量 369

remote 文件格式

设置 remote 变量 369

tn 命令 335

tn3270 命令 335

tokstat 命令 370

topasout 命令 385

topsvcs 脚本 386

topsvcsctrl 脚本 388

touch 命令 391

tprof 命令 393

tput 命令 402

tr 命令 404

traceroute 命令 413

tracesoff 命令 415

traceson 命令 416

trbsd 命令 417

trcctl 419

trcdead 命令 420

trcevgrp 命令 422

trcnm 命令 423

trcrpt 命令 424

trcstop 命令 430

trcupdate 命令 431

troff 命令 432

格式化表

使用 tbl 命令 309



- troff 命令 (续)
  - 命令输出解释器
  - 使用 tc 命令 312
- trpt 命令 485
  - 输出字段 486
- tset 命令 494
- tsh 命令 496
- tsm 命令 498
- tunchange 命令 502
- tuncheck 命令 504
- tundefault 命令 505
- tunrestore 命令 506
- tunsave 命令 508
- turnacct 命令 509
- tvi 编辑器
  - 操作方式 512
  - 调用
    - 使用 tvi 命令 511
  - 定制 512
  - 限制 511
- tvi 命令 511
- twconvdict 命令 514
- twconvfont 命令 515
- type 命令 516
- type, 命令
  - 写入描述
    - 使用 type 命令 516

## U

- ucgif 方法 517
- ucfginet 方法 517
- uconvdef 命令 520
- udefif 方法 521
- udefinet 方法 522
- udfcheck 命令 522
- udfcreate 命令 523
- udflabel 命令 524
- UIL 编译器
  - 启动
    - 使用 uil 命令 524
- uil 命令 524
- uimx 命令 525
- ul 命令 527
- ulimit 命令 527
- umask 命令 529
- umcode\_latest 命令 530
- umountall 命令 533
- unalias 命令 534
- uncompress 命令 538
- unexpand 命令 540
- unfencevsd 命令 541
- uniq 命令 545
- unlink 命令 549
- unlink 子例程 549
- unloadipsec 命令 550
- unmirrorvg 命令 551
- unmount 命令 532
- unpack 命令 552
- untab 命令 554
- updatevsdnode 命令 555
- updatevsdtab 命令 557
- updatevsdvg 命令 558
- uptime 命令 560
- useradd 命令 561
- userdel 命令 563
- usermod 命令 564
- users
  - 显示简要列表 566
- usrck 命令 567
- utftp 命令 350
- utmpd 572
- uuccheck 命令 573
- uucico 命令 575
- uucico 守护程序 575
- uuclean 命令 576
- uucleanup 命令 578
- uucp 命令 580
- uucpadmin 命令 583
- uucpd 命令 586
- uucpd 守护程序 586
- uudecode 命令 586
- uudemon.admin 命令 587
- uudemon.admin shell 脚本 587
- uudemon.cleau 命令 588
- uudemon.cleau shell 脚本 588
- uudemon.hour 命令 590
- uudemon.hour shell 脚本 590
- uudemon.poll 命令 591
- uudemon.poll shell 脚本 591
- uuencode 命令 593
- UUIDs
  - 创建 594
- uuid\_gen 命令 594
- uukick 命令 595
- uulog 命令 596
- uuname 命令 598
- uupick 命令 599
- uupoll 命令 601
- uuq 命令 602
- uusched 命令 604
- uusched 守护程序 604
- uusend 命令 605
- uusnap 命令 606

uustat 命令 607  
uuto 命令 599, 610  
Uutry 命令 613  
uux 命令 615  
uuxqt 命令 618  
uuxqt 守护程序 618

## X

X 会话  
  初始化  
  使用 startx 命令 183

## [ 特别字符 ]

/etc/utmp  
  监控 572  
/etc/uucp/Permissions  
  检查 573





中国印刷

S151-0039-03

