

旗標



旗标系列图书

RED HAT

Linux 7

指令参考手册

施威铭研究室 编著
蒋心晓 牛红 牛荣健 张堃 改编

ECHO

SLEEP

MKDIR

CAT

人民邮电出版社

www.pptph.com.cn

旗标出版股份有限公司

旗标系列图书

Linux 7 指令参考手册

施威铭研究室 编著

蒋心晓 牛 红
牛荣健 张 堃 改编

人 民 邮 电 出 版 社

图书在版编目 (CIP) 数据

Linux 7 指令参考手册 / 施威铭研究室编著; 蒋心晓等改编.
—北京: 人民邮电出版社, 2001.10

ISBN 7-115-09523-X

I. L... II. ①施... ②蒋... III. Linux 操作系统—指令系统—手册 IV. TP316.81

中国版本图书馆 CIP 数据核字 (2001) 第 064611 号

版 权 声 明

本书为台湾旗标出版股份有限公司独家授权的中文简化字版本。本书的专有出版权属人民邮电出版社所有。在没有得到本书原版出版者和本书出版者的书面许可之前, 任何单位和个人不得擅自摘抄、复制本书的部分或全部内容, 以任何形式 (包括资料和出版物) 进行传播。

本书贴有旗标 (FLAG) 激光防伪标签, 无标签者不得销售。

版权所有, 侵权必究。

旗标系列图书

Linux 7 指令参考手册

-
- ◆ 编 著 施威铭研究室
改 编 蒋心晓 牛 红 牛荣健 张 堃
责任编辑 俞 彬
 - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子函件 315@ pptph.com.cn
网址 <http://www.pptph.com.cn>
读者热线 010-67129212 010-67129211 (传真)
北京汉魂图文设计有限公司制作
北京顺义振华印刷厂印刷
新华书店总店北京发行所经销
 - ◆ 开本: 800 × 1000 1/16
印张: 50.25
字数: 1 024 千字 2001 年 10 月第 1 版
印数: 1 - 4 000 册 2001 年 10 月北京第 1 次印刷

著作权合同登记 图字: 01 - 2000 - 3664 号

ISBN 7-115-09523-X/TP·2376

定价: 72.00 元

本书如有印装质量问题, 请与本社联系 电话: (010) 67129223

前 言

Linux 的热潮已如火如荼地展开，各种 Linux 的发行版也如雨后春笋般一个个地冒出头来。X Windows 的图形界面不断地改进，使得操作 Linux 系统也愈来愈容易。但是不管再怎么改变，Linux 指令仍然是操控系统的根本，有无可取代的地位。

想要灵活地掌握 Linux 系统，如果只会在 X Windows 图形界面下用鼠标东按按、西按按，那是绝对不够的。Linux 系统的众多指令，就如同工具箱中的锤子、螺丝钉，每个指令都有其独特的功用。这些数量众多且功能齐备的指令，就是 Linux 的精华所在。只要学会这些指令，将来不管是使用哪一种 Linux 或 UNIX 系统，皆可如鱼得水且怡然自得了。

施威铭研究室
2001 年 6 月

adduser



常用指数:

a**功能说明:** 建立用户帐号。

语 法: adduser [-mMnr][[-c<备注>]][-d<登录目录>][[-e<有效期限>]][-f<缓冲天数>][[-g<群组>]][-G<群组>][[-s<shell>]][-u<uid>][使用者帐号] 或
adduser -D [-b] [-e<群组>][[-f<缓冲天数>]][-g<群组>][[-G<群组>]][[-s<shell>]]

相关指令: groupadd, groupdel, groupmod, id, passwd, userdel, usermod.

补充说明: adduser 可用来建立用户帐号。帐号建好之后, 再用 passwd 设置帐号的密码。而可用 userdel 删除帐号。此指令在 Red Hat Linux 中, 其实是 useradd 的符号连接。使用 adduser 指令所建立的帐号, 实际上保存在/etc/passwd 文本文件中。

参 数:

-c<备注>	加上备注文字。备注文字会保存在 passwd 的备注栏中。
-d<登录目录>	指定用户登录时的开始目录。
-D	更改默认值。默认值的选项说明如下: -b <用户目录>: 在指定的<用户目录>下, 建立所有用户的<登录目录>。默认值为/home。 -e <有效期限>: 指定默认帐号的有效期限。 -f <缓冲天数>: 指定默认在密码过期多少天后即关闭该帐号。 -g <群组>: 指定用户默认所属的群组。 -G <群组>: 指定用户默认所属的附加群组。 -s <shell>: 指定使用默认的 shell。
-e<有效期限>	指定帐号的有效期限。<有效期限>的格式为月份/日期/公元年后两位。
-f<缓冲天数>	指定在密码过期后多少天即关闭该帐号。
-g<群组>	指定用户所属的群组。
-G<群组>	指定用户所属的附加群组。
-m	自动建立用户的登录目录。
-M	不要自动建立用户的登录目录。
-n	不建立以用户名称为名的群组。
-r	建立系统帐号。
-s<shell>	指定用户登录后所使用的 shell。
-u<uid>	指定用户 ID。

范 例:



建立名称为 `alkane` 的用户帐号

```
$ adduser alkane
```



建立名称为 `alkane` 的用户帐号, 并指定登录目录为 `/home/alpha`:

```
$ adduser -d /home/alpha alkane
```



建立名称为 `alkane` 的用户帐号, 并指定帐号的有效期限为 2000 年 12 月 31 日。

```
$ adduser -e 12/31/00 alkane
```



建立名称为 `alkane` 的用户帐号, 并指定所属群组为 `users`:

```
$ adduser -g users alkane
```



建立名称为 `alkane` 的用户帐号, 并指定其登录后所使用的 shell 为 C shell:

```
$ adduser -s /bin/csh alkane
```

alias



功能说明: 设置指令的别名。

语 法: `alias [别名][=指令]`

相关指令: `unalias`。

补充说明: 用户可利用 `alias`, 自定义指令的别名。若仅输入 `alias`, 则可列出目前所有的别名设置。`alias` 的作用范围仅及于该次登录的操作。若要每次登录时即自动设置好别名, 请在 `.profile` (`bash`) 或 `.cshrc` (`tcsh`) 中设置指令的别名。

参 数: 若不加任何参数, 则列出目前所有的别名设置。

范 例:

 列出目前所有的别名设置:

```
$ alias
alias cp='cp -i'
alias mv='mv -i'
alias rm='rm -i'
alias which='type -path'
...
```

 将 mkdir 指令的别名设置为 md:

```
$ alias md=mkdir
$ alias md          ← 显示 md 的别名设置
alias md='mkdir'
```

 将 'ls -al' 指令的别名设置为 dir:

```
$ alias dir='ls -al'
      ↑
    指定别名时可加上指令的参数
```

apachectl (Apache control interface)

常用指数: 

功能说明: 可用来控制 Apache HTTP 服务器的程序。

语法: apachectl [configtest] [fullstatus] [graceful] [help] [restart] [start] [status] [stop]

相关指令: httpd。

补充说明: 各 Linux 发行版内附的 Apache HTTP 服务器可能有些微的差异, 因此不一定都会有这个 script 文件。

参数:

configtest	检查设置文件中的语法是否正确。
-------------------	-----------------

续表

fullstatus	显示服务器完整的状态信息。
graceful	重新启动 Apache 服务器。与 restart 参数的差异在于使用 graceful 参数时，不会中断原有的连接。
help	显示说明信息。
restart	重新启动 Apache 服务器。
start	启动 Apache 服务器。
status	显示服务器摘要的状态信息。
stop	停止 Apache 服务器。

范 例:

启动 Apache 服务器:

```
$ apachectl start
./apachectl start: httpd started
```

重新启动 Apache 服务器:

```
$ apachectl restart
./apachectl restart: httpd restarted
```

重新启动 Apache 服务器，但不中断现有的连接:

```
$ apachectl graceful
./apachectl graceful: httpd graceful restarted
```

显示 Apache 服务器目前的状态。执行时须先取消 access.conf 设置文件中 <Location /server-status> 到 </Location> 区段的“#”号:

```
$ apachectl status
Apache Server Status for top.flag.com.tw      ← 执行主机的地址
Server Version: Apache/1.3.6 (Unix)           ← 服务器的版本编号
Server Built: May 4 1999 13:53:31              ← 服务器此版本的编译时间
-----
Current Time: Tuesday, 11-Apr-2000 11:26:09 CST ← 系统目前的时间
Restart Time: Tuesday, 11-Apr-2000 10:43:14 CST ← 服务器最后一次重新启动的时间
```


a

目前有 4 个执行中的程序 目前有 6 个程序是闲置的

Scoreboard Key:

"_" Waiting for Connection, "S" Starting up, "R" Reading Request,
"W" Sending Reply, "K" Keepalive (read), "D" DNS Lookup,
"L" Logging, "G" Gracefully finishing, "." Open slot with no current
process

\$ **http://IP 地址/server-status**

PID Key:

👉 停止执行 Apache 服务器:

```
$ apachectl stop
```

```
./apachectl stop: httpd stopped
```

 检查 Apache 服务器配置文件的语法:

```
$ apachectl configtest
```

```
Syntax OK
```

 检查结果正常

apmd (advanced power management daemon)

常用指数:



功能说明: 高级电源管理服务程序。

语法: apmd [-quvVW] [-p <百分比增量>] [-w <百分比值>]

相关指令: 无。

补充说明: apmd 负责 BIOS 高级电源管理 (APM) 相关的记录、警告与管理工

参数:

-p <百分比增量> 或 -percentage <百分比增量>	当电力变化的幅度超出设置的百分比增量时, 即记录事件。百分比增量的默认值为 5, 若设置值超过 100, 则关闭此功能。
-q	取消 -w 参数的功能。
-u 或 -utc	将 BIOS 时钟设置为 UTC(GMT, 格林威治标准时间), 以便从挂起模式 (suspend mode)恢复时, 将-u 参数传送到 clock 或 hwclock (设置硬件时钟) 程序。
-v 或 -verbose	记录所有的 APM 事件。
-V 或 -version	显示版本信息。
-w <百分比值> 或 -warn <百分比值>	当电池不在充电状态, 且电池电量低于设置的百分比值时, 则在 syslog(2) 的 ALERT 级别记录警告信息。百分比值的默认值为 10, 若设置值为 0, 则关闭此功能。
-W 或 -wall	发出警告信息给所有登录者。

范 例:

 当电池的电力低于 20% 时，即警告所有的用户：

```
$ apmd -w 20 -w
```

 取消电力低于某个百分比，即发出警告的功能：

```
$ apmd -q
```

 当电力变化幅度超出 10% 时，即记录事件：

```
$ apmd -p 10
```

ar

常用指数:  

功能说明: 建立、修改库文件(archive)，或从库文件中抽取文件。

语 法: ar [-dmpqrtx] [cfosSuvV] [a <成员文件>] [b<成员文件>] [i<成员文件>] [库文件] [成员文件]

相关指令: 无。

补充说明: ar 可让您集合许多文件（称为成员文件）成为单一的库文件。在库文件中，所有成员文件皆保有原来的属性与权限。ar 也可以为库文件建立符号表 (Symbol table)，功能如同文件的索引，以便直接调用库文件中的成员文件。

第 1 个参数 ([-dmpqrtx]) 为指令参数，指定 ar 所要执行的工作。第 2 个参数 ([abcfiosSuvV]) 为选项参数，用来指定工作的选项。

参 数:

指令参数	
-d	删除库文件中的成员文件。
-m	变更成员文件在库文件中的次序。若未指定第 2 个参数，则-m 参数会将指定的成员文件移到库文件的最后面。

续表

-p	显示库文件中成员文件的内容。若未指定[成员文件]参数, 则显示库文件中全部成员文件的内容。
-q	将文件附加在库文件末端, 无须检查库文件中是否有重复的成员文件。
-r	将文件插入库文件中。若文件重复, 则取代原有的成员文件。
-t	显示库文件中所包含的文件。
-x	从库文件中抽取成员文件(库文件中仍留有该成员文件)。
选项参数	
a <成员文件>	将文件插入库文件中指定的成员文件之后。
b <成员文件>	将文件插入库文件中指定的成员文件之前。
c	建立库文件。
f	为避免过长的文件名不兼容于其它系统的 ar 指令, 可利用此参数, 截掉要放入库文件中过长的成员文件名称。
i <成员文件>	将文件插入库文件中指定的成员文件之前(与 b 的作用相同)。
o	保留库文件中文件的日期。若不设置此参数, 文件从库文件中抽取出来时, 文件日期将更改为当前的日期。
s	若库文件中包含了对象模块, 可利用此参数建立库文件的符号表。建好符号表后, 其它程序便可调用库文件中的对象。
S	不产生符号表。若库文件很庞大时, 可利用此参数提高程序的执行速度。
u	只将日期较新的文件插入库文件中。
v	程序执行时显示详细的信息。
V	显示版本信息。

范 例:

将 file1、file2 与 file3 文件备份为库文件 arfile, 执行时显示详细信息; 然后列出其所包含的文件:

```
$ ar -rv arfile file1 file2 file3          ← 建立 arfile 库文件
a - file1
a - file2
a - file3
```

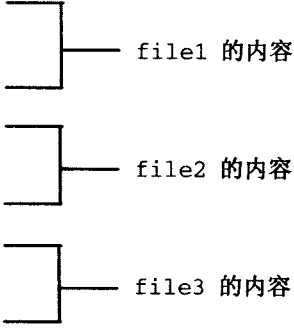
```
$ ar -t arfile
file1
file2
file3
```

← 显示 arfile 中所包含的文件

a

🐼 承上例，列出库文件 arfile 内所有成员文件的内容：

```
$ ar -p arfile
file1 content
line1 in file1
line2 in file1
file2 content
line1 in file2
line2 in file2
file3 content
line1 in file3
line2 in file3
```



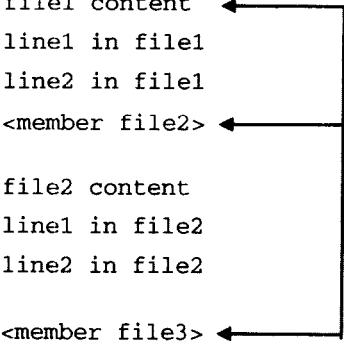
🐼 以较详细的方式列出库文件 arfile 的内容：

```
$ ar -pv arfile

<member file1>

file1 content
line1 in file1
line2 in file1
<member file2>
file2 content
line1 in file2
line2 in file2
<member file3>

file3 content
line1 in file3
line2 in file3
```



会在文件内容之前，列出库文件中成员文件的文件名

- 列出库文件 `arfile` 中 `file1` 的内容:

```
$ ar -p arfile file1
```

```
file1 content
line1 in file1
line2 in file1
```

- 承上例, 将 `newfile` 加入库文件中 `file2` 之前:

```
$ ar -rbv file2 arfile newfile
```

```
a - newfile
```

```
$ ar -t arfile
```

```
file1
newfile      ← newfile 位于 file2 之前
file2
file3
```

- 承上例, 将库文件中的 `newfile` 移到库文件的最后面:

```
$ ar -mv arfile newfile
```

```
m - newfile
```

```
$ ar -t arfile
```

```
file1
file2
file3
newfile      ← newfile 已经在库文件的最后面
```

- 承上例, 将库文件中的 `newfile` 移至 `file2` 之前:

```
$ ar -mbv file2 arfile newfile
```

```
m - newfile
```

```
$ ar -t arfile
```

```
file1
newfile      ← newfile 已经移至 file2 之前
file2
file3
```

- 承上例, 将库文件中的 `newfile` 移至 `file2` 之后:

```
$ ar -mav file2 arfile newfile
m - newfile
$ ar -t arfile
file1
file2
newfile      ← newfile 已经移至 file2 之后
file3
```

☛ 承上例，取出库文件中的 newfile，但仍保留此文件在库文件中：

```
$ ar -xv arfile newfile
x - newfile
$ ar -t arfile
file1
file2
newfile      ← 取出的 newfile 仍保留在库文件中
file3
$ ls -l newfile
-rwxr-xr-x  1 root    root          44 Apr  8 14:39 newfile
           ↑
        文件的时间为取出的时间
```

☛ 承上例，取出库文件中的 newfile，但保留取出文件的更改时间：

```
$ ar -xov arfile newfile
x - newfile
$ ls -l newfile
-rwxr-xr-x  1 root    root          44 Apr  8 14:32 newfile
           ↑
        文件时间为原本的在库文件中的时间
```

☛ 承上例，删除库文件中的 newfile：

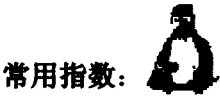
```
$ ar -dv arfile newfile
d - newfile
$ ar -t arfile
file1
file2
file3      ] — newfile 已经从库文件中删除
```



承上例，将 file2 加入库文件的末端，无须检查是否有重复的成员文件：

```
$ ar -qv arfile file2
a - file2
$ ar -t arfile
file1
file2
file3
file2
```

archive



常用指数：

功能说明： 新闻组文件备份程序。

语 法： archive [-fmr] [-a <备份目录>] [-i <索引文件>] [源文件]

相关指令： slrn, tin。

补充说明： archive 会读取新闻组的文件（默认目录为 /var/spool/news/articles），然后加以备份（默认存到 /var/spool/news/archive 目录）。archive 在备份时，会保存[源文件]中文件的目录层次。

参 数：

-a <备份目录>	指定备份的目录。
-f	将层次式目录名称的层次关系取消，以句号(.)来取代斜线 (/)。
-i <索引文件>	制作文件索引。
-m	将文件复制到目的地，建立文件链接后，以链接取代原始的文件。
-r	Archive 默认的标准错误输出为 /var/errlog。-r 参数可取消此功能。
[源文件]	在源文件中指定要备份的文件。

范 例:

备份新闻组中的文件，并以链接取代原始的文件：

```
$ cd /var/spool/news/articles
$ ls -R
.:
flag

flag:
access linux

flag/access:
1

flag/linux:
1 2 3

$ cat newsfile
/var/spool/news/articles/flag/linux/1
/var/spool/news/articles/flag/linux/2
/var/spool/news/articles/flag/linux/3

$ archive -m newsfile
$ cd /var/spool/news/archive
$ ls -R
.:
var

var:
spool

var/spool:
news

var/spool/news:
articles

var/spool/news/articles:
flag
```

← 切换到存储新闻组文件的目录

← 要备份的组里面有三篇文章

← 在 newsfile 中指定要备份的文件

← 备份文件

← 切换到 archive 默认的备份目录

archive 会保留原先的目录层次

a