at

executes commands at a specified time.

cat - concatenate files and print on the standard output

adduser  and  addgroup  add users and groups to the system according to

      command    line    options    and    configuration    information    in

      /etc/adduser.conf.   They  are  friendlier  front ends to the low level

      tools like useradd, groupadd and usermod programs, by default  choosing

      Debian  policy conformant UID and GID values, creating a home directory

      with skeletal configuration, running a custom script,  and  other  fea‐

      tures.  adduser and addgroup can be run in one of five modes

batch

executes commands when system load levels permit; in other words, when the load average drops below 0.8, or the value specified in the invocation of atrun.

chgrp - change group ownership

Change  the  group of each FILE to GROUP.  With --reference, change the

      group of each FILE to that of RFILE.

cd- Change directory

chmod - change file mode bits

This manual page documents the GNU version of chmod.  chmod changes the

      file mode bits of each given file  according  to  mode,  which  can  be

      either a symbolic representation of changes to make, or an octal number

      representing the bit pattern for the new mode bits.

 chown - change file owner and group

This manual page documents the GNU version of chown.  chown changes the

      user and/or group ownership of each given file.  If only  an  owner  (a

      user  name or numeric user ID) is given, that user is made the owner of

      each given file, and the files' group is not changed.  If the owner  is

      followed  by  a  colon  and a group name (or numeric group ID), with no

      spaces between them, the group ownership of the  files  is  changed  as

      well.  If a colon but no group name follows the user name, that user is

      made the owner of the files and the group of the files  is  changed  to

      that  user's  login  group.   If the colon and group are given, but the

      owner is omitted, only the group of the files is changed; in this case,

      chown  performs  the same function as chgrp.  If only a colon is given,

      or if the entire operand is empty, neither the owner nor the  group  is

clear - clear the terminal screen .

cp - copy files and directories

cpio - copy files to and from archives

cut - remove sections from each line of files

dc - an arbitrary precision calculator

dd - convert and copy a file

df - report file system disk space usage

dir - list directory contents

du - estimate file space usage

echo - display a line of text

fdformat - low-level format a floppy disk

fdisk - manipulate disk partition table

file — determine file type

fmt - simple optimal text formatter

Reformat  each  paragraph  in  the FILE(s), writing to standard output.

      The option -WIDTH is an abbreviated form of –width=DIGITS.

free - Display amount of free and used memory in the system

free  displays the total amount of free and used physical and swap mem‐

      ory in the system, as well as the buffers  used  by  the  kernel.   The

      shared  memory column represents either the MemShared value (2.4 series

      kernels) or the Shmem value (2.6 series kernels and later)  taken  from

      the  /proc/meminfo  file.  The  value is zero if none of the entries is

      exported by the kernel.

fsck - check and repair a Linux filesystem

fsck is used to check and optionally repair one or more Linux  filesys‐

      tems.   filesys  can  be  a device name (e.g.  /dev/hdc1, /dev/sdb2), a

      mount point (e.g.  /, /usr, /home), or an ext2 label or UUID  specifier

      (e.g.   UUID=8868abf6-88c5-4a83-98b8-bfc24057f7bd or LABEL=root).  Nor‐

      mally, the fsck program will try to  handle  filesystems  on  different

      physical  disk  drives  in  parallel to reduce the total amount of time

      needed to check all of them.

      If no filesystems are specified on the command line, and the -A  option

      is  not  specified,  fsck  will  default  to  checking  filesystems  in

      /etc/fstab serially.  This is equivalent to the -As options.

  grep, egrep, fgrep, rgrep - print lines matching a pattern

grep  searches the named input FILEs (or standard input if no files are

      named, or if a single hyphen-minus (-) is given as file name) for lines

      containing  a  match to the given PATTERN.  By default, grep prints the

      matching lines.

      In  addition,  three  variant  programs  egrep,  fgrep  and  rgrep  are

      available.   egrep  is  the  same  as  grep -E.   fgrep  is the same as

      grep -F.  rgrep is the same as grep -r.  Direct  invocation  as  either

      egrep  or  fgrep  is  deprecated,  but  is provided to allow historical

      applications that rely on them to run unmodified.

gzip, gunzip, zcat - compress or expand files

find - search for files in a directory hierarchy

  This  manual page documents the GNU version of find.  GNU find searches

      the directory tree rooted at each given file  name  by  evaluating  the

      given  expression  from left to right, according to the rules of prece‐

      dence (see section OPERATORS), until the outcome  is  known  (the  left

      hand  side  is  false  for and operations, true for or), at which point

      find moves on to the next file name.

kill - send a signal to a process

The  default  signal  for kill is TERM.  Use -l or -L to list available

      signals.  Particularly useful signals include  HUP,  INT,  KILL,  STOP,

      CONT,  and  0.   Alternate  signals may be specified in three ways: -9,

      -SIGKILL or -KILL.  Negative PID values may be  used  to  choose  whole

      process  groups; see the PGID column in ps command output.  A PID of -1

      is special; it indicates all processes except the kill  process  itself

      and init.

 locate - find files by name

locate  reads  one or more databases prepared by updatedb(8) and writes

      file names matching at least one of the PATTERNs  to  standard  output,

      one per line.

      ls - list directory contents

List  information  about  the FILEs (the current directory by default).

      Sort entries alphabetically if none of -cftuvSUX nor --sort  is  speci‐

      fied.

make - GNU make utility to maintain groups of programs

The purpose of the make utility is  to  determine  automatically  which

      pieces of a large program need to be recompiled, and issue the commands

      to recompile them.  The manual  describes  the  GNU  implementation  of

      make,  which was written by Richard Stallman and Roland McGrath, and is

      currently maintained by Paul Smith.   Our  examples  show  C  programs,

      since  they  are most common, but you can use make with any programming

      language whose compiler can be run with a shell command.  In fact, make

      is  not limited to programs.  You can use it to describe any task where

      some files must be updated automatically from others whenever the  oth‐

      ers change.

      To  prepare to use make, you must write a file called the makefile that

      describes the relationships among files in your program, and the states

      the  commands for updating each file.  In a program, typically the exe‐

      cutable file is updated from object files, which are in  turn  made  by

      compiling source files.

mkfs - build a Linux filesystem

mkfs  is  used  to build a Linux filesystem on a device, usually a hard

      disk partition.  The device argument is either the  device  name  (e.g.

      /dev/hda1,  /dev/sdb2),  or  a  regular  file  that  shall  contain the

      filesystem.  The size argument is the number of blocks to be  used  for

      the filesystem.

      The exit code returned by mkfs is 0 on success and 1 on failure.

mke2fs - create an ext2/ext3/ext4 filesystem

      mke2fs  is used to create an ext2, ext3, or ext4 filesystem, usually in

      a disk partition.  device is the  special  file  corresponding  to  the

      device  (e.g  /dev/hdXX).   blocks-count is the number of blocks on the

      device.  If omitted, mke2fs automagically figures the file system size.

      If  called  as  mkfs.ext3  a journal is created as if the -j option was

      specified.

more — file perusal filter for crt viewing

more is a filter for paging through text one screenful at a time.  This

    version is especially primitive.  Users should realize that less(1) pro‐

    vides more(1) emulation plus extensive enhancements.

mount - mount a filesystem

All files accessible in a Unix system are arranged in one big tree, the

      file hierarchy, rooted at /.  These files can be spread out  over  sev‐

      eral  devices.  The mount command serves to attach the filesystem found

      on some device to the big file tree. Conversely, the umount(8)  command

      will detach it again.

nohup - run a command immune to hangups, with output to a non-tty

      If  standard input is a terminal, redirect it from /dev/null.  If stan‐

      dard output is a terminal, append output to  'nohup.out'  if  possible,

      '$HOME/nohup.out' otherwise.  If standard error is a terminal, redirect

      it to standard output.  To save output to FILE, use  'nohup  COMMAND  >

      FILE'.

      NOTE:  your  shell  may  have  its  own version of nohup, which usually

      supersedes the version described here.  Please refer  to  your  shell's

      documentation for details about the options it supports.

passwd - change user password

The passwd command changes passwords for user accounts. A normal user

      may only change the password for his/her own account, while the

      superuser may change the password for any account.  passwd also changes

      the account or associated password validity period.

pr - convert text files for printing

ps - report a snapshot of the current processes.

ps displays information about a selection of the active processes.  If

      you want a repetitive update of the selection and the displayed

      information, use top(1) instead.

pwd - print name of current/working directory

 rm - remove files or directories

 rmdir - remove empty directories

sed - stream editor for filtering and transforming text

SET - change a run-time parameter

The SET command changes run-time configuration parameters. Many of the

      run-time parameters listed in Chapter 18, Server Configuration, in the

      documentation can be changed on-the-fly with SET. (But some require

      superuser privileges to change, and others cannot be changed after

      server or session start.)  SET only affects the value used by the

      current session.

shutdown - bring the system down

shutdown arranges for the system to be brought down in a safe way.  All

      logged-in users are notified that the system is going down and,  within

      the last five minutes of TIME, new logins are prevented.

**su** - change user ID or become superuser

The su command is used to become another user during a login session.

Invoked without a username, su defaults to becoming the superuser. The

optional argument - may be used to provide an environment similar to

what the user would expect had the user logged in directly.

Additional arguments may be provided after the username, in which case

they are supplied to the user's login shell. In particular, an argument

of -c will cause the next argument to be treated as a command by most

command interpreters. The command will be executed by the shell

specified in /etc/passwd for the target user.

You can use the -- argument to separate su options from the arguments

supplied to the shell.

**tail** - output the last part of files

DESCRIPTION

Print the last 10 lines of each FILE to standard output. With more

than one FILE, precede each with a header giving the file name. With

no FILE, or when FILE is -, read standard input.

Mandatory arguments to long options are mandatory for short options

too.

-c, --bytes=K

output the last K bytes; alternatively, use -c +K to output

bytes starting with the Kth of each file

-f, --follow[={name|descriptor}]

output appended data as the file grows; -f, --follow, and --fol‐

low=descriptor are equivalent

-F same as --follow=name --retry

-n, --lines=K

output the last K lines, instead of the last 10; or use -n +K to

output lines starting with the Kth

**tar** — The GNU version of the tar archiving utility

Tar stores and extracts files from a tape or disk archive.

The first argument to tar should be a function; either one of the letters

Acdrtux, or one of the long function names. A function letter need not be prefixed with ``-'', and may be combined with other single-letter

options. A long function name must be prefixed with --. Some options take a parameter; with the single-letter form these must be given as separate arguments. With the long form, they may be given by appending =value to the option.

**touch** - change file timestamps

Update the access and modification times of each FILE to the current

time.

A FILE argument that does not exist is created empty, unless -c or -h

is supplied.

A FILE argument string of - is handled specially and causes touch to

change the times of the file associated with standard output.

Mandatory arguments to long options are mandatory for short options

too.

-a change only the access time

-c, --no-create

do not create any files

-d, --date=STRING

parse STRING and use it instead of current time

-f (ignored)

-h, --no-dereference

affect each symbolic link instead of any referenced file (useful

only on systems that can change the timestamps of a symlink)

-m change only the modification time

-r, --reference=FILE

use this file's times instead of current time

-t STAMP

use [[CC]YY]MMDDhhmm[.ss] instead of current time

--time=WORD

**tput**, **reset** - initialize a terminal or query terminfo database

The tput utility uses the terminfo database to make the values of ter‐

minal-dependent capabilities and information available to the shell

(see sh(1)), to initialize or reset the terminal, or return the long

name of the requested terminal type. The result depends upon the capa‐

bility's type:

string

tput writes the string to the standard output. No trailing

newline is supplied.

integer

tput writes the decimal value to the standard output, with

a trailing newline.

boolean

tput simply sets the exit code (0 for TRUE if the terminal

has the capability, 1 for FALSE if it does not), and writes

nothing to the standard output.

**uname** - print system information

Print certain system information. With no OPTION, same as -s.

**unzip** - list, test and extract compressed files in a ZIP archive

unzip will list, test, or extract files from a ZIP archive, commonly

found on MS-DOS systems. The default behavior (with no options) is to

extract into the current directory (and subdirectories below it) all

files from the specified ZIP archive. A companion program, zip(1),

creates ZIP archives; both programs are compatible with archives cre‐

ated by PKWARE's PKZIP and PKUNZIP for MS-DOS, but in many cases the

program options or default behaviors differ.

**vdir** - list directory contents

List information about the FILEs (the current directory by default).

Sort entries alphabetically if none of -cftuvSUX nor --sort is speci‐

fied.

**vim** - Vi IMproved, a programmers text editor

Vim is a text editor that is upwards compatible to Vi. It can be used to edit all kinds of plain text. It is especially useful for editing programs.

**wall** — write a message to users

Wall displays the contents of file or, by default, its standard input, on

the terminals of all currently logged in users. The command will cut over

79 character long lines to new lines. Short lines are white space padded

to have 79 characters. The command will always put carriage return and

new line at the end of each line.

Only the super-user can write on the terminals of users who have chosen

to deny messages or are using a program which automatically denies mes‐

sages.

Reading from a file is refused when the invoker is not superuser and the

program is suid or sgid.

**wc** - print newline, word, and byte counts for each file

Print newline, word, and byte counts for each FILE, and a total line if

more than one FILE is specified. With no FILE, or when FILE is -, read

standard input. A word is a non-zero-length sequence of characters

delimited by white space. The options below may be used to select

which counts are printed, always in the following order: newline, word,

character, byte, maximum line length.

**whatis** - display one-line manual page descriptions

Each manual page has a short description available within it. whatis

searches the manual page names and displays the manual page descrip‐

tions of any name matched.

name may contain wildcards (-w) or be a regular expression (-r). Using

these options, it may be necessary to quote the name or escape (\) the

special characters to stop the shell from interpreting them

**whereis** - locate the binary, source, and manual page files for a command

whereis locates source/binary and manuals sections for specified files.

The supplied names are first stripped of leading pathname components

and any (single) trailing extension of the form .ext, for example, .c.

Prefixes of s. resulting from use of source code control are also

dealt with. whereis then attempts to locate the desired program in a

list of standard Linux places.

**which** - locate a command

which returns the pathnames of the files (or links) which would be exe‐cuted in the current environment, had its arguments been given as com‐mands in a strictly POSIX-conformant shell. It does this by searching the PATH for executable files matching the names of the arguments. It does not follow symbolic links.

**who** - show who is logged on

Print information about users who are currently logged in.

**zip** - package and compress (archive) files

zip is a compression and file packaging utility for Unix, VMS, MSDOS, OS/2, Windows 9x/NT/XP, Minix, Atari, Macintosh, Amiga, and Acorn RISC OS. It is analogous to a combination of the Unix commands tar(1) and compress(1) and is compatible with PKZIP (Phil Katz's ZIP for MSDOS

systems).