# NAME

BusyBox - The Swiss Army Knife of Embedded Linux

---

# SYNTAX

```
busybox <applet> [arguments...]  # or
<applet> [arguments...]          # if symlinked
```

---

# DESCRIPTION

BusyBox combines tiny versions of many common UNIX utilities into a single small executable. It provides minimalist replacements for most of the utilities you usually find in GNU coreutils, util-linux, etc. The utilities in BusyBox generally have fewer options than their full-featured GNU cousins; however, the options that are included provide the expected functionality and behave very much like their GNU counterparts.

BusyBox has been written with size-optimization and limited resources in mind. It is also extremely modular so you can easily include or exclude commands (or features) at compile time. This makes it easy to customize your embedded systems. To create a working system, just add /dev, /etc, and a Linux kernel. BusyBox provides a fairly complete POSIX environment for any small or embedded system.

BusyBox is extremely configurable. This allows you to include only the components you need, thereby reducing binary size. Run 'make config' or 'make menuconfig' to select the functionality that you wish to enable. Then run 'make' to compile BusyBox using your configuration.

After the compile has finished, you should use 'make install' to install BusyBox. This will install the 'bin/busybox' binary, in the target directory specified by CONFIG_PREFIX. CONFIG_PREFIX can be set when configuring BusyBox, or you can specify an alternative location at install time (i.e., with a command line like 'make CONFIG_PREFIX=/tmp/foo install'). If you enabled any applet installation scheme (either as symlinks or hardlinks), these will also be installed in the location pointed to by CONFIG_PREFIX.

---

# USAGE

BusyBox is a multi-call binary. A multi-call binary is an executable program that performs the same job as more than one utility program. That means there is just a single BusyBox binary, but that single binary acts like a large number of utilities. This allows BusyBox to be smaller since all the built-in utility programs (we call them applets) can share code for many common operations.

You can also invoke BusyBox by issuing a command as an argument on the command line. For example, entering

```
/bin/busybox ls
```

will also cause BusyBox to behave as 'ls'.

Of course, adding '/bin/busybox' into every command would be painful. So most people will invoke BusyBox using links to the BusyBox binary.

For example, entering

```
ln -s /bin/busybox ls
./ls
```

will cause BusyBox to behave as 'ls' (if the 'ls' command has been compiled into BusyBox). Generally speaking, you should never need to make all these links yourself, as the BusyBox build system will do this for you when you run the 'make install' command.

If you invoke BusyBox with no arguments, it will provide you with a list of the applets that have been compiled into your BusyBox binary.

---

# COMMON OPTIONS

Most BusyBox applets support the **--help** argument to provide a terse runtime description of their behavior. If the CONFIG_FEATURE_VERBOSE_USAGE option has been enabled, more detailed usage information will also be available.

---

# COMMANDS

Currently available applets include:

```
[, [[, acpid, addgroup, adduser, adjtimex, ar, arp, arping, ash,
awk, basename, beep, blkid, brctl, bunzip2, bzcat, bzip2, cal, cat,
catv, chat, chattr, chgrp, chmod, chown, chpasswd, chpst, chroot,
chrt, chvt, cksum, clear, cmp, comm, cp, cpio, crond, crontab,
cryptpw, cut, date, dc, dd, deallocvt, delgroup, deluser, depmod,
devmem, df, dhcprelay, diff, dirname, dmesg, dnsd, dnsdomainname,
dos2unix, dpkg, du, dumpkmap, dumpleases, echo, ed, egrep, eject,
env, envdir, envuidgid, expand, expr, fakeidentd, false, fbset,
fbsplash, fdflush, fdformat, fdisk, fgrep, find, findfs, flash_lock,
flash_unlock, fold, free, freeramdisk, fsck, fsck.minix, fsync,
ftpd, ftpget, ftpput, fuser, getopt, getty, grep, gunzip, gzip, hd,
hdparm, head, hexdump, hostid, hostname, httpd, hush, hwclock, id,
ifconfig, ifdown, ifenslave, ifplugd, ifup, inetd, init, inotifyd,
insmod, install, ionice, ip, ipaddr, ipcalc, ipcrm, ipcs, iplink,
iproute, iprule, iptunnel, kbd_mode, kill, killall, killall5, klogd,
last, length, less, linux32, linux64, linuxrc, ln, loadfont,
loadkmap, logger, login, logname, logread, losetup, lpd, lpq, lpr,
ls, lsattr, lsmod, lzmacat, lzop, lzopcat, makemime, man, md5sum,
mdev, mesg, microcom, mkdir, mkdosfs, mkfifo, mkfs.minix, mkfs.vfat,
mknod, mkpasswd, mkswap, mktemp, modprobe, more, mount, mountpoint,
mt, mv, nameif, nc, netstat, nice, nmeter, nohup, nslookup, od,
openvt, passwd, patch, pgrep, pidof, ping, ping6, pipe_progress,
pivot_root, pkill, popmaildir, printenv, printf, ps, pscan, pwd,
raidautorun, rdate, rdev, readlink, readprofile, realpath,
reformime, renice, reset, resize, rm, rmdir, rmmod, route, rpm,
rpm2cpio, rtcwake, run-parts, runlevel, runsv, runsvdir, rx, script,
scriptreplay, sed, sendmail, seq, setarch, setconsole, setfont,
setkeycodes, setlogcons, setsid, setuidgid, sh, sha1sum, sha256sum,
sha512sum, showkey, slattach, sleep, softlimit, sort, split,
start-stop-daemon, stat, strings, stty, su, sulogin, sum, sv,
svlogd, swapoff, swapon, switch_root, sync, sysctl, syslogd, tac,
tail, tar, taskset, tcpsvd, tee, telnet, telnetd, test, tftp, tftpd,
time, timeout, top, touch, tr, traceroute, true, tty, ttysize,
udhcpc, udhcpd, udpsvd, umount, uname, uncompress, unexpand, uniq,
unix2dos, unlzma, unlzop, unzip, uptime, usleep, uudecode, uuencode,
vconfig, vi, vlock, volname, watch, watchdog, wc, wget, which, who,
whoami, xargs, yes, zcat, zcip
```

---

# COMMAND DESCRIPTIONS

**acpid**

acpid [-d] [-c CONFDIR] [-l LOGFILE] [-e PROC_EVENT_FILE] [EVDEV_EVENT_FILE...]

Listen to ACPI events and spawn specific helpers on event arrival

Options:

```
-d      Do not daemonize and log to stderr
-c DIR  Config directory [/etc/acpi]
-e FILE /proc event file [/proc/acpi/event]
-l FILE Log file [/var/log/acpid]
```

Accept and ignore compatibility options -g -m -s -S -v

**addgroup**

addgroup [-g GID] [user_name] group_name

Add a group or add a user to a group

Options:

```
-g GID  Group id
-S      Create a system group
```

**adduser**

adduser [OPTIONS] user_name

Add a user

Options:

```
-h DIR          Home directory
-g GECOS        GECOS field
-s SHELL        Login shell
-G GRP          Add user to existing group
-S              Create a system user
-D              Do not assign a password
-H              Do not create home directory
-u UID          User id
```

**adjtimex**

adjtimex [-q] [-o offset] [-f frequency] [-p timeconstant] [-t tick]

Read and optionally set system timebase parameters. See adjtimex(2).

Options:

```
-q              Quiet
-o offset       Time offset, microseconds
-f frequency    Frequency adjust, integer kernel units (65536 is 1ppm)
                (positive values make clock run faster)
-t tick         Microseconds per tick, usually 10000
-p timeconstant
```

**ar**

ar [-o] [-v] [-p] [-t] [-x] ARCHIVE FILES

Extract or list FILES from an ar archive

Options:

```
-o      Preserve original dates
-p      Extract to stdout
-t      List
-x      Extract
-v      Verbose
```

**arp**

arp
[-vn][-H type] [-i if] -a [hostname]
[-v]  [-i if] -d hostname [pub]
[-v]  [-H type] [-i if] -s hostname hw_addr [temp]
[-v]  [-H type] [-i if] -s hostname hw_addr [netmask nm] pub
[-v]  [-H type] [-i if] -Ds hostname ifa [netmask nm] pub

Manipulate ARP cache

Options:

```
-a              Display (all) hosts
-s              Set new ARP entry
-d              Delete a specified entry
-v              Verbose
-n              Don't resolve names
-i IF           Network interface
-D              Read <hwaddr> from given device
-A, -p AF       Protocol family
-H HWTYPE       Hardware address type
```

**arping**

arping [-fqbDUA] [-c count] [-w timeout] [-I dev] [-s sender] target

Send ARP requests/replies

Options:

```
-f              Quit on first ARP reply
-q              Quiet
-b              Keep broadcasting, don't go unicast
-D              Duplicated address detection mode
-U              Unsolicited ARP mode, update your neighbors
-A              ARP answer mode, update your neighbors
-c N            Stop after sending N ARP requests
-w timeout      Time to wait for ARP reply, in seconds
-I dev          Interface to use (default eth0)
-s sender       Sender IP address
target          Target IP address
```

**awk**

awk [OPTIONS] [AWK_PROGRAM] [FILE]...

Options:

```
-v VAR=VAL      Set variable
-F SEP          Use SEP as field separator
-f FILE         Read program from file
```

**basename**

basename FILE [SUFFIX]

Strip directory path and .SUFFIX from FILE

**beep**

beep -f freq -l length -d delay -r repetitions -n

Options:

```
-f      Frequency in Hz
-l      Length in ms
-d      Delay in ms
-r      Repetitions
-n      Start new tone
```

**blkid**

blkid

Print UUIDs of all filesystems

**brctl**

brctl COMMAND [BRIDGE [INTERFACE]]

Manage ethernet bridges

Commands:

```
show                    Show a list of bridges
addbr BRIDGE            Create BRIDGE
delbr BRIDGE            Delete BRIDGE
addif BRIDGE IFACE      Add IFACE to BRIDGE
delif BRIDGE IFACE      Delete IFACE from BRIDGE
setageing BRIDGE TIME           Set ageing time
setfd BRIDGE TIME               Set bridge forward delay
sethello BRIDGE TIME            Set hello time
setmaxage BRIDGE TIME           Set max message age
setpathcost BRIDGE COST         Set path cost
setportprio BRIDGE PRIO         Set port priority
setbridgeprio BRIDGE PRIO       Set bridge priority
stp BRIDGE [1|0]                STP on/off
```

**bunzip2**

bunzip2 [OPTIONS] [FILE]

Uncompress FILE (or standard input if FILE is '-' or omitted)

Options:

```
-c      Write to standard output
-f      Force
```

**bzcat**

bzcat FILE

Uncompress to stdout

**bzip2**

bzip2 [OPTIONS] [FILE]...

Compress FILE(s) with bzip2 algorithm. When FILE is '-' or unspecified, reads standard input. Implies -c.

Options:

```
-c      Write to standard output
-d      Decompress
-f      Force
-1..-9  Compression level
```

**cal**

cal [-jy] [[month] year]

Display a calendar

Options:

```
        -j      Use julian dates
        -y      Display the entire year
```

**cat**

cat [-u] [FILE]...

Concatenate FILE(s) and print them to stdout

Options:

```
        -u      Use unbuffered i/o (ignored)
```

**catv**

catv [-etv] [FILE]...

Display nonprinting characters as ^x or M-x

Options:

```
        -e      End each line with $
        -t      Show tabs as ^I
        -v      Don't use ^x or M-x escapes
```

**chat**

chat EXPECT [SEND [EXPECT [SEND...]]]

Useful for interacting with a modem connected to stdin/stdout. A script consists of one or more "expect-send" pairs of strings, each pair is a pair of arguments. Example: chat " ATZ OK ATD123456 CONNECT " ogin: pppuser word: ppppass '~'

**chattr**

chattr [-R] [-+=AacDdijsStTu] [-v version] files...

Change file attributes on an ext2 fs

Modifiers:

```
        -       Remove attributes
        +       Add attributes
        =       Set attributes
Attributes:

        A       Don't track atime
        a       Append mode only
        c       Enable compress
        D       Write dir contents synchronously
        d       Do not backup with dump
        i       Cannot be modified (immutable)
        j       Write all data to journal first
        s       Zero disk storage when deleted
        S       Write file contents synchronously
        t       Disable tail-merging of partial blocks with other files
        u       Allow file to be undeleted
Options:

        -R      Recursively list subdirectories
        -v      Set the file's version/generation number
```

**chgrp**

chgrp [-RhLHPcvf]... GROUP FILE...

Change the group membership of each FILE to GROUP

Options:

```
-R      Recurse
-h      Affect symlinks instead of symlink targets
-L      Traverse all symlinks to directories
-H      Traverse symlinks on command line only
-P      Do not traverse symlinks (default)
-c      List changed files
-v      Verbose
-f      Hide errors
```

**chmod**

chmod [-Rcvf] MODE[,MODE]... FILE...

Each MODE is one or more of the letters ugoa, one of the symbols +-= and one or more of the letters rwxst

Options:

```
-R      Recurse
-c      List changed files
-v      List all files
-f      Hide errors
```

**chown**

chown [-RhLHPcvf]... OWNER[<.|:>[GROUP]] FILE...

Change the owner and/or group of each FILE to OWNER and/or GROUP

Options:

```
-R      Recurse
-h      Affect symlinks instead of symlink targets
-L      Traverse all symlinks to directories
-H      Traverse symlinks on command line only
-P      Do not traverse symlinks (default)
-c      List changed files
-v      List all files
-f      Hide errors
```

**chpasswd**

chpasswd [--md5|--encrypted]

Read user:password from stdin and update /etc/passwd

Options:

```
-e,--encrypted  Supplied passwords are in encrypted form
-m,--md5        Use MD5 encryption instead of DES
```

**chpst**

chpst [-vP012] [-u USER[:GRP]] [-U USER[:GRP]] [-e DIR]
[-/ DIR] [-n NICE] [-m BYTES] [-d BYTES] [-o N]
[-p N] [-f BYTES] [-c BYTES] PROG ARGS

Change the process state and run PROG

Options:

```
-u USER[:GRP]   Set uid and gid
-U USER[:GRP]   Set $UID and $GID in environment
-e DIR          Set environment variables as specified by files
                in DIR: file=1st_line_of_file
-/ DIR          Chroot to DIR
-n NICE         Add NICE to nice value
-m BYTES        Same as -d BYTES -s BYTES -l BYTES
-d BYTES        Limit data segment
-o N            Limit number of open files per process
-p N            Limit number of processes per uid
-f BYTES        Limit output file sizes
-c BYTES        Limit core file size
-v              Verbose
-P              Create new process group
-0              Close standard input
-1              Close standard output
-2              Close standard error
```

**chroot**

chroot NEWROOT [PROG [ARGS]]

Run PROG with root directory set to NEWROOT

**chrt**

chrt [OPTIONS] [PRIO] [PID | PROG [ARGS]]

Manipulate real-time attributes of a process

Options:

```
-p      Operate on pid
-r      Set scheduling policy to SCHED_RR
-f      Set scheduling policy to SCHED_FIFO
-o      Set scheduling policy to SCHED_OTHER
-m      Show min and max priorities
```

**chvt**

chvt N

Change the foreground virtual terminal to /dev/ttyN

**cksum**

cksum FILES...

Calculate the CRC32 checksums of FILES

**clear**

clear

Clear screen

**cmp**

cmp [-l] [-s] FILE1 [FILE2 [SKIP1 [SKIP2]]]

Compares FILE1 vs stdin if FILE2 is not specified

Options:

```
-l      Write the byte numbers (decimal) and values (octal)
        for all differing bytes
```

```
	-s	Quiet
```

**comm**

comm [-123] FILE1 FILE2

Compare FILE1 to FILE2, or to stdin if - is specified

Options:

```
	-1	Suppress lines unique to FILE1
	-2	Suppress lines unique to FILE2
	-3	Suppress lines common to both files
```

**cp**

cp [OPTIONS] SOURCE DEST

Copy SOURCE to DEST, or multiple SOURCE(s) to DIRECTORY

Options:

```
	-a	Same as -dpR
	-d,-P	Preserve links
	-H,-L	Dereference all symlinks (default)
	-p	Preserve file attributes if possible
	-f	Force overwrite
	-i	Prompt before overwrite
	-R,-r	Recurse
	-l,-s	Create (sym)links
```

**cpio**

cpio -[tiopdmvu] [-F FILE] [-H newc]

Extract or list files from a cpio archive, or create a cpio archive Main operation mode:

```
	-t	List
	-i	Extract
	-o	Create
	-p	Passthrough
Options:
	-d	Make leading directories
	-m	Preserve mtime
	-v	Verbose
	-u	Overwrite
	-F	Input file
	-H	Define format
```

**crond**

crond -fbS -l N -d N -L LOGFILE -c DIR

```
	-f	Foreground
	-b	Background (default)
	-S	Log to syslog (default)
	-l	Set log level. 0 is the most verbose, default 8
	-d	Set log level, log to stderr
	-L	Log to file
	-c	Working dir
```

**crontab**

crontab [-c DIR] [-u USER] [-ler]|[FILE]

```
	-c	Crontab directory
	-u	User
```

```
-l      List crontab
-e      Edit crontab
-r      Delete crontab
FILE    Replace crontab by FILE ('-': stdin)
```

**cryptpw**

cryptpw [OPTIONS] [PASSWORD] [SALT]

Crypt the PASSWORD using `crypt(3)`

Options:

```
-P,--password-fd=NUM    Read password from fd NUM
-m,--method=TYPE        Encryption method TYPE
-S,--salt=SALT
```

**cut**

cut [OPTIONS] [FILE]...

Print selected fields from each input FILE to standard output

Options:

```
-b LIST Output only bytes from LIST
-c LIST Output only characters from LIST
-d CHAR Use CHAR instead of tab as the field delimiter
-s      Output only the lines containing delimiter
-f N    Print only these fields
-n      Ignored
```

**date**

date [OPTIONS] [+FMT] [TIME]

Display time (using +FMT), or set time

Options:

```
[-s] TIME       Set time to TIME
-u              Work in UTC (don't convert to local time)
-R              Output RFC-822 compliant date string
-I[SPEC]        Output ISO-8601 compliant date string
                SPEC='date' (default) for date only,
                'hours', 'minutes', or 'seconds' for date and
                time to the indicated precision
-r FILE         Display last modification time of FILE
-d TIME         Display TIME, not 'now'
-D FMT          Use FMT for -d TIME conversion
```

Recognized TIME formats:

```
hh:mm[:ss]
[YYYY.]MM.DD-hh:mm[:ss]
YYYY-MM-DD hh:mm[:ss]
[[[[[YY]YY]MM]DD]hh]mm[.ss]
```

**dc**

dc expression...

Tiny RPN calculator. Operations: +, add, -, sub, *, mul, /, div, %, mod, **, exp, and, or, not, eor, p - print top of the stack (without altering the stack), f - print entire stack, o - pop the value and set output radix (value must be 10 or 16). Examples: 'dc 2 2 add' -> 4, 'dc 8 8 * 2 2 + /' -> 16.

**dd**

dd [if=FILE] [of=FILE] [ibs=N] [obs=N] [bs=N] [count=N] [skip=N]
[seek=N] [conv=notrunc|noerror|sync|fsync]

Copy a file with converting and formatting

Options:

```
if=FILE         Read from FILE instead of stdin
of=FILE         Write to FILE instead of stdout
bs=N            Read and write N bytes at a time
ibs=N           Read N bytes at a time
obs=N           Write N bytes at a time
count=N         Copy only N input blocks
skip=N          Skip N input blocks
seek=N          Skip N output blocks
conv=notrunc    Don't truncate output file
conv=noerror    Continue after read errors
conv=sync       Pad blocks with zeros
conv=fsync      Physically write data out before finishing
```

Numbers may be suffixed by c (x1), w (x2), b (x512), kD (x1000), k (x1024), MD (x1000000), M (x1048576), GD (x1000000000) or G (x1073741824)

**deallocvt**

deallocvt [N]

Deallocate unused virtual terminal /dev/ttyN

**delgroup**

delgroup [USER] GROUP

Delete group GROUP from the system or user USER from group GROUP

**deluser**

deluser USER

Delete USER from the system

**devmem**

devmem ADDRESS [WIDTH [VALUE]]

Read/write from physical address

```
ADDRESS Address to act upon
WIDTH   Width (8/16/...)
VALUE   Data to be written
```

**df**

df [-Pkmhai] [-B SIZE] [FILESYSTEM...]

Print filesystem usage statistics

Options:

```
-P      POSIX output format
-k      1024-byte blocks (default)
-m      1M-byte blocks
-h      Human readable (e.g. 1K 243M 2G)
-a      Show all filesystems
-i      Inodes
-B SIZE Blocksize
```

**dhcprelay**

dhcprelay CLIENT_IFACE[,CLIENT_IFACE2...] SERVER_IFACE [SERVER_IP]

Relay DHCP requests between clients and server

**diff**

diff [-abdiNqrTstw] [-L LABEL] [-S FILE] [-U LINES] FILE1 FILE2

Compare files line by line and output the differences between them. This implementation supports unified diffs only.

Options:

```
-a      Treat all files as text
-b      Ignore changes in the amount of whitespace
-d      Try hard to find a smaller set of changes
-i      Ignore case differences
-L      Use LABEL instead of the filename in the unified header
-N      Treat absent files as empty
-q      Output only whether files differ
-r      Recursively compare subdirectories
-S      Start with FILE when comparing directories
-T      Make tabs line up by prefixing a tab when necessary
-s      Report when two files are the same
-t      Expand tabs to spaces in output
-U      Output LINES lines of context
-w      Ignore all whitespace
```

**dirname**

dirname FILENAME

Strip non-directory suffix from FILENAME

**dmesg**

dmesg [-c] [-n LEVEL] [-s SIZE]

Print or control the kernel ring buffer

Options:

```
-c              Clear ring buffer after printing
-n LEVEL        Set console logging level
-s SIZE         Buffer size
```

**dnsd**

dnsd [-c config] [-t seconds] [-p port] [-i iface-ip] [-d]

Small static DNS server daemon

Options:

```
-c      Config filename
-t      TTL in seconds
-p      Listening port
-i      Listening ip (default all)
-d      Daemonize
```

**dos2unix**

dos2unix [OPTION] [FILE]

Convert FILE in-place from DOS to Unix format. When no file is given, use stdin/stdout.

Options:

```
-u      dos2unix
-d      unix2dos
```

**dpkg**

dpkg [-ilCPru] [-F option] package_name

Install, remove and manage Debian packages

Options:

```
-i              Install the package
-l              List of installed packages
-C              Configure an unpackaged package
-F depends      Ignore dependency problems
-P              Purge all files of a package
-r              Remove all but the configuration files for a package
-u              Unpack a package, but don't configure it
```

**du**

du [-aHLdclsxhmk] [FILE]...

Summarize disk space used for each FILE and/or directory. Disk space is printed in units of 1024 bytes.

Options:

```
-a      Show file sizes too
-H      Follow symlinks on command line
-L      Follow all symlinks
-d N    Limit output to directories (and files with -a) of depth < N
-c      Show grand total
-l      Count sizes many times if hard linked
-s      Display only a total for each argument
-x      Skip directories on different filesystems
-h      Sizes in human readable format (e.g., 1K 243M 2G )
-m      Sizes in megabytes
-k      Sizes in kilobytes (default)
```

**dumpkmap**

dumpkmap > keymap

Print a binary keyboard translation table to standard output

**dumpleases**

dumpleases [-r|-a] [-f LEASEFILE]

Display DHCP leases granted by udhcpd

Options:

```
-f,--file=FILE  Leases file to load
-r,--remaining  Interpret lease times as time remaining
-a,--absolute   Interpret lease times as expire time
```

**echo**

echo [-neE] [ARG...]

Print the specified ARGs to stdout

Options:

```
-n      Suppress trailing newline
-e      Interpret backslash-escaped characters (i.e., \t=tab)
-E      Disable interpretation of backslash-escaped characters
```

**ed**

ed

**eject**

eject [-t] [-T] [DEVICE]

Eject specified DEVICE (or default /dev/cdrom)

Options:

```
-s      SCSI device
-t      Close tray
-T      Open/close tray (toggle)
```

**env**

env [-iu] [-] [name=value]... [PROG [ARGS]]

Print the current environment or run PROG after setting up the specified environment

Options:

```
-, -i   Start with an empty environment
-u      Remove variable from the environment
```

**envdir**

envdir dir prog args

Set various environment variables as specified by files in the directory dir and run PROG

**envuidgid**

envuidgid account prog args

Set $UID to account's uid and $GID to account's gid and run PROG

**expand**

expand [-i] [-t NUM] [FILE|-]

Convert tabs to spaces, writing to standard output

Options:

```
-i,--initial    Do not convert tabs after non blanks
-t,--tabs=N     Tabstops every N chars
```

**expr**

expr EXPRESSION

Print the value of EXPRESSION to standard output

EXPRESSION may be:

```
ARG1 | ARG2     ARG1 if it is neither null nor 0, otherwise ARG2
ARG1 & ARG2     ARG1 if neither argument is null or 0, otherwise 0
ARG1 < ARG2     1 if ARG1 is less than ARG2, else 0. Similarly:
ARG1 <= ARG2
```

```
ARG1 = ARG2
ARG1 != ARG2
ARG1 >= ARG2
ARG1 > ARG2
ARG1 + ARG2     Sum of ARG1 and ARG2. Similarly:
ARG1 - ARG2
ARG1 * ARG2
ARG1 / ARG2
ARG1 % ARG2
STRING : REGEXP         Anchored pattern match of REGEXP in STRING
match STRING REGEXP     Same as STRING : REGEXP
substr STRING POS LENGTH Substring of STRING, POS counted from 1
index STRING CHARS      Index in STRING where any CHARS is found, or 0
length STRING           Length of STRING
quote TOKEN             Interpret TOKEN as a string, even if
                        it is a keyword like 'match' or an
                        operator like '/'
(EXPRESSION)            Value of EXPRESSION
```

Beware that many operators need to be escaped or quoted for shells. Comparisons are arithmetic if both ARGs are numbers, else lexicographical. Pattern matches return the string matched between \( and \) or null; if \( and \) are not used, they return the number of characters matched or 0.

**fakeidentd**

fakeidentd [-fiw] [-b ADDR] [STRING]

Provide fake ident (auth) service

Options:

```
-f      Run in foreground
-i      Inetd mode
-w      Inetd 'wait' mode
-b ADDR Bind to specified address
STRING  Ident answer string (default: nobody)
```

**false**

false

Return an exit code of FALSE (1)

**fbset**

fbset [OPTIONS] [MODE]

Show and modify frame buffer settings

**fbsplash**

fbsplash -s IMGFILE [-c] [-d DEV] [-i INIFILE] [-f CMD]

Options:

```
-s      Image
-c      Hide cursor
-d      Framebuffer device (default /dev/fb0)
-i      Config file (var=value):
                BAR_LEFT,BAR_TOP,BAR_WIDTH,BAR_HEIGHT
                BAR_R,BAR_G,BAR_B
-f      Control pipe (else exit after drawing image)
                commands: 'NN' (% for progress bar) or 'exit'
```

**fdflush**

fdflush DEVICE

Force floppy disk drive to detect disk change

**fdformat**

fdformat [-n] DEVICE

Format floppy disk

Options:

```
-n      Don't verify after format
```

**fdisk**

fdisk [-ul] [-C CYLINDERS] [-H HEADS] [-S SECTORS] [-b SSZ] DISK

Change partition table

Options:

```
-u              Start and End are in sectors (instead of cylinders)
-l              Show partition table for each DISK, then exit
-b 2048         (for certain MO disks) use 2048-byte sectors
-C CYLINDERS    Set number of cylinders/heads/sectors
-H HEADS

-S SECTORS
```

**find**

find [PATH...] [EXPRESSION]

Search for files. The default PATH is the current directory, default EXPRESSION is '-print'

EXPRESSION may consist of:

```
-follow         Dereference symlinks
-xdev           Don't descend directories on other filesystems
-maxdepth N     Descend at most N levels. -maxdepth 0 applies
                tests/actions to command line arguments only
-mindepth N     Do not act on first N levels
-name PATTERN   File name (w/o directory name) matches PATTERN
-iname PATTERN  Case insensitive -name
-path PATTERN   Path matches PATTERN
-regex PATTERN  Path matches regex PATTERN
-type X         File type is X (X is one of: f,d,l,b,c,...)
-perm NNN       Permissions match any of (+NNN), all of (-NNN),
                or exactly (NNN)
-mtime DAYS     Modified time is greater than (+N), less than (-N),
                or exactly (N) days
-mmin MINS      Modified time is greater than (+N), less than (-N),
                or exactly (N) minutes
-newer FILE     Modified time is more recent than FILE's
-inum N         File has inode number N
-user NAME      File is owned by user NAME (numeric user ID allowed)
-group NAME     File belongs to group NAME (numeric group ID allowed)
-depth          Process directory name after traversing it
-size N[bck]    File size is N (c:bytes,k:kbytes,b:512 bytes(def.)).
                +/-N: file size is bigger/smaller than N
-print          Print (default and assumed)
-print0         Delimit output with null characters rather than
                newlines
-exec CMD ARG ; Run CMD with all instances of {} replaced by the
                matching files
-prune          Stop traversing current subtree
-delete         Delete files, turns on -depth option
(EXPR)          Group an expression
```

**findfs**

findfs LABEL=label or UUID=uuid

Find a filesystem device based on a label or UUID

**flash_lock**

flash_lock MTD_DEVICE OFFSET SECTORS

Lock part or all of an MTD device. If SECTORS is -1, then all sectors will be locked, regardless of the value of OFFSET

**flash_unlock**

flash_unlock MTD_DEVICE

Unlock an MTD device

**fold**

fold [-bs] [-w WIDTH] [FILE]

Wrap input lines in each FILE (standard input by default), writing to standard output

Options:

```
-b      Count bytes rather than columns
-s      Break at spaces
-w      Use WIDTH columns instead of 80
```

**free**

free

Display the amount of free and used system memory

**freeramdisk**

freeramdisk DEVICE

Free all memory used by the specified ramdisk

**fsck**

fsck [-ANPRTV] [-C fd] [-t fstype] [fs-options] [filesys...]

Check and repair filesystems

Options:

```
-A      Walk /etc/fstab and check all filesystems
-N      Don't execute, just show what would be done
-P      With -A, check filesystems in parallel
-R      With -A, skip the root filesystem
-T      Don't show title on startup
-V      Verbose
-C n    Write status information to specified filedescriptor
-t type List of filesystem types to check
```

**fsck.minix**

fsck.minix [-larvsmf] /dev/name

Check MINIX filesystem

Options:

```
-l      List all filenames
-r      Perform interactive repairs
-a      Perform automatic repairs
-v      Verbose
-s      Output superblock information
-m      Show "mode not cleared" warnings
-f      Force file system check
```

**fsync**

fsync [OPTIONS] FILE...Write files' buffered blocks to disk

Options:

```
-d      Avoid syncing metadata
```

**ftpd**

ftpd [-wvS] [-t N] [-T N] [DIR]

Anonymous FTP server

ftpd should be used as an inetd service. ftpd's line for inetd.conf: 21 stream tcp nowait root ftpd ftpd /files/to/serve It also can be ran from tcpsvd:

```
tcpsvd -vE 0.0.0.0 21 ftpd /files/to/serve
```

Options:

```
-w      Allow upload
-v      Log to stderr
-S      Log to syslog
-t,-T   Idle and absolute timeouts
DIR     Change root to this directory
```

**ftpget**

ftpget [OPTIONS] HOST LOCAL_FILE REMOTE_FILE

Retrieve a remote file via FTP

Options:

```
-c,--continue   Continue previous transfer
-v,--verbose    Verbose
-u,--username   Username
-p,--password   Password
-P,--port       Port number
```

**ftpput**

ftpput [OPTIONS] HOST REMOTE_FILE LOCAL_FILE

Store a local file on a remote machine via FTP

Options:

```
-v,--verbose    Verbose
-u,--username   Username
-p,--password   Password
-P,--port       Port number
```

**fuser**

fuser [OPTIONS] FILE or PORT/PROTO

Find processes which use FILEs or PORTs

Options:

```
-m      Find processes which use same fs as FILEs
-4      Search only IPv4 space
-6      Search only IPv6 space
-s      Silent: just exit with 0 if any processes are found
-k      Kill found processes (otherwise display PIDs)
-SIGNAL Signal to send (default: TERM)
```

**getopt**

getopt [OPTIONS]

Options:

```
-a,--alternative                Allow long options starting with single -
-l,--longoptions=longopts       Long options to be recognized
-n,--name=progname              The name under which errors are reported
-o,--options=optstring          Short options to be recognized
-q,--quiet                      Disable error reporting by getopt(3)
-Q,--quiet-output               No normal output
-s,--shell=shell                Set shell quoting conventions
-T,--test                       Test for getopt(1) version
-u,--unquoted                   Don't quote the output
```

**getty**

getty [OPTIONS] BAUD_RATE TTY [TERMTYPE]

Open a tty, prompt for a login name, then invoke /bin/login

Options:

```
-h              Enable hardware (RTS/CTS) flow control
-i              Do not display /etc/issue before running login
-L              Local line, do not do carrier detect
-m              Get baud rate from modem's CONNECT status message
-w              Wait for a CR or LF before sending /etc/issue
-n              Do not prompt the user for a login name
-f ISSUE_FILE   Display ISSUE_FILE instead of /etc/issue
-l LOGIN        Invoke LOGIN instead of /bin/login
-t SEC          Terminate after SEC if no username is read
-I INITSTR      Send INITSTR before anything else
-H HOST         Log HOST into the utmp file as the hostname
```

**grep**

grep [-HhrilLnqvsoweFEABCz] PATTERN [FILE]...

Search for PATTERN in each FILE or standard input

Options:

```
-H      Prefix output lines with filename where match was found
-h      Suppress the prefixing filename on output
-r      Recurse
-i      Ignore case distinctions
-l      List names of files that match
-L      List names of files that do not match
-n      Print line number with output lines
-q      Quiet. Return 0 if PATTERN is found, 1 otherwise
-v      Select non-matching lines
-s      Suppress file open/read error messages
-c      Only print count of matching lines
-o      Show only the part of a line that matches PATTERN
-m MAX  Match up to MAX times per file
```

```
-w      Match whole words only
-F      PATTERN is a set of newline-separated strings
-E      PATTERN is an extended regular expression
-e PTRN Pattern to match
-f FILE Read pattern from file
-A      Print NUM lines of trailing context
-B      Print NUM lines of leading context
-C      Print NUM lines of output context
-z      Input is NUL terminated
```

**gunzip**

gunzip [OPTIONS] [FILE]...

Uncompress FILEs (or standard input)

Options:

```
-c      Write to standard output
-f      Force
-t      Test file integrity
```

**gzip**

gzip [OPTIONS] [FILE]...

Compress FILEs (or standard input)

Options:

```
-c      Write to standard output
-d      Decompress
-f      Force
```

**hd**

hd FILE...

hd is an alias for hexdump -C

**hdparm**

hdparm [OPTIONS] [DEVICE]

Options:

```
-a      Get/set fs readahead
-A      Set drive read-lookahead flag (0/1)
-b      Get/set bus state (0 == off, 1 == on, 2 == tristate)
-B      Set Advanced Power Management setting (1-255)
-c      Get/set IDE 32-bit IO setting
-C      Check IDE power mode status
-d      Get/set using_dma flag
-D      Enable/disable drive defect-mgmt
-f      Flush buffer cache for device on exit
-g      Display drive geometry
-h      Display terse usage information
-i      Display drive identification
-I      Detailed/current information directly from drive
-k      Get/set keep_settings_over_reset flag (0/1)
-K      Set drive keep_features_over_reset flag (0/1)
-L      Set drive doorlock (0/1) (removable harddisks only)
-m      Get/set multiple sector count
-n      Get/set ignore-write-errors flag (0/1)
-p      Set PIO mode on IDE interface chipset (0,1,2,3,4,...)
-P      Set drive prefetch count
-Q      Get/set DMA tagged-queuing depth (if supported)
-r      Get/set readonly flag (DANGEROUS to set)
```

```
-R      Register an IDE interface (DANGEROUS)
-S      Set standby (spindown) timeout
-t      Perform device read timings
-T      Perform cache read timings
-u      Get/set unmaskirq flag (0/1)
-U      Un-register an IDE interface (DANGEROUS)
-v      Defaults; same as -mcudkrag for IDE drives
-V      Display program version and exit immediately
-w      Perform device reset (DANGEROUS)
-W      Set drive write-caching flag (0/1) (DANGEROUS)
-x      Tristate device for hotswap (0/1) (DANGEROUS)
-X      Set IDE xfer mode (DANGEROUS)
-y      Put IDE drive in standby mode
-Y      Put IDE drive to sleep
-Z      Disable Seagate auto-powersaving mode
-z      Re-read partition table
```

**head**

head [OPTIONS] [FILE]...

Print first 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.

Options:

```
-n NUM  Print first NUM lines instead of first 10
-c NUM  Output the first NUM bytes
-q      Never output headers giving file names
-v      Always output headers giving file names
```

**hexdump**

hexdump [-bcCdefnosvxR] FILE...

Display file(s) or standard input in a user specified format

Options:

```
-b              One-byte octal display
-c              One-byte character display
-C              Canonical hex+ASCII, 16 bytes per line
-d              Two-byte decimal display
-e FORMAT STRING
-f FORMAT FILE
-n LENGTH       Interpret only LENGTH bytes of input
-o              Two-byte octal display
-s OFFSET       Skip OFFSET bytes
-v              Display all input data
-x              Two-byte hexadecimal display
-R              Reverse of 'hexdump -Cv'
```

**hostid**

hostid

Print out a unique 32-bit identifier for the machine

**hostname**

hostname [OPTIONS] [HOSTNAME | -F FILE]

Get or set hostname or DNS domain name

Options:

```
-s      Short
-i      Addresses for the hostname
```

```
-d      DNS domain name
-f      Fully qualified domain name
-F FILE Use FILE's content as hostname
```

**httpd**

httpd [-ifv[v]] [-c CONFFILE] [-p [IP:]PORT] [-u USER[:GRP]] [-r REALM] [-h HOME] or httpd -d/-e/-m STRING

Listen for incoming HTTP requests

Options:

```
-i              Inetd mode
-f              Do not daemonize
-v[v]           Verbose
-c FILE         Configuration file (default httpd.conf)
-p [IP:]PORT    Bind to ip:port (default *:80)
-u USER[:GRP]   Set uid/gid after binding to port
-r REALM        Authentication Realm for Basic Authentication
-h HOME         Home directory (default .)
-m STRING       MD5 crypt STRING
-e STRING       HTML encode STRING
-d STRING       URL decode STRING
```

**hwclock**

hwclock [-r|--show] [-s|--hctosys] [-w|--systohc] [-l|--localtime] [-u|--utc] [-f FILE]

Query and set hardware clock (RTC)

Options:

```
-r      Show hardware clock time
-s      Set system time from hardware clock
-w      Set hardware clock to system time
-u      Hardware clock is in UTC
-l      Hardware clock is in local time
-f FILE Use specified device (e.g. /dev/rtc2)
```

**id**

id [OPTIONS] [USER]

Print information about USER or the current user

Options:

```
-u      Print user ID
-g      Print group ID
-G      Print supplementary group IDs
-n      Print name instead of a number
-r      Print real user ID instead of effective ID
```

**ifconfig**

ifconfig [-a] interface [address]

Configure a network interface

Options:

```
[add ADDRESS[/PREFIXLEN]]
[del ADDRESS[/PREFIXLEN]]
[[-]broadcast [ADDRESS]] [[-]pointopoint [ADDRESS]]
[netmask ADDRESS] [dstaddr ADDRESS]
[outfill NN] [keepalive NN]
[hw ether|infiniband ADDRESS] [metric NN] [mtu NN]
[[-]trailers] [[-]arp] [[-]allmulti]
```

```
[multicast] [[-]promisc] [txqueuelen NN] [[-]dynamic]
[mem_start NN] [io_addr NN] [irq NN]
[up|down] ...
```

**ifdown**

ifdown [-ainmvf] ifaces...

Options:

```
-a      De/configure all interfaces automatically
-i FILE Use FILE for interface definitions
-n      Print out what would happen, but don't do it
        (note: doesn't disable mappings)
-m      Don't run any mappings
-v      Print out what would happen before doing it
-f      Force de/configuration
```

**ifenslave**

ifenslave [-cdf] master-iface <slave-iface...>

Configure network interfaces for parallel routing

Options:

```
-c, --change-active     Change active slave
-d, --detach            Remove slave interface from bonding device
-f, --force             Force, even if interface is not Ethernet
```

**ifplugd**

ifplugd [OPTIONS]

Network interface plug detection daemon

Options:

```
-n              Do not daemonize
-s              Do not log to syslog
-i IFACE        Interface
-f/-F           Treat link detection error as link down/link up
                (otherwise exit on error)
-a              Do not up interface automatically
-M              Monitor creation/destruction of interface
                (otherwise it must exist)
-r PROG         Script to run
-x ARG          Extra argument for script
-I              Don't exit on nonzero exit code from script
-p              Don't run script on daemon startup
-q              Don't run script on daemon quit
-l              Run script on startup even if no cable is detected
-t SECS         Poll time in seconds
-u SECS         Delay before running script after link up
-d SECS         Delay after link down
-m MODE         API mode (mii, priv, ethtool, wlan, auto)
-k              Kill running daemon
```

**ifup**

ifup [-ainmvf] ifaces...

Options:

```
-a      De/configure all interfaces automatically
-i FILE Use FILE for interface definitions
-n      Print out what would happen, but don't do it
        (note: doesn't disable mappings)
```

```
        -m      Don't run any mappings
        -v      Print out what would happen before doing it
        -f      Force de/configuration
```

**inetd**

inetd [-fe] [-q N] [-R N] [CONFFILE]

Listen for network connections and launch programs

Options:

```
        -f      Run in foreground
        -e      Log to stderr
        -q N    Socket listen queue (default: 128)
        -R N    Pause services after N connects/min
                (default: 0 - disabled)
```

**init**

init

Init is the parent of all processes

**inotifyd**

inotifyd PROG FILE1[:MASK] ...

Run PROG on filesystem changes. When a filesystem event matching MASK occurs on FILEn, PROG <actual_event(s)> <FILEn> [<subfile_name>] is run. Events:

```
        a       File is accessed
        c       File is modified
        e       Metadata changed
        w       Writable file is closed
        0       Unwritable file is closed
        r       File is opened
        D       File is deleted
        M       File is moved
        u       Backing fs is unmounted
        o       Event queue overflowed
        x       File can't be watched anymore
If watching a directory:
        m       Subfile is moved into dir
        y       Subfile is moved out of dir
        n       Subfile is created
        d       Subfile is deleted
```

inotifyd waits for PROG to exit. When x event happens for all FILEs, inotifyd exits

**insmod**

insmod [OPTIONS] MODULE [symbol=value]...

Load the specified kernel modules into the kernel

Options:

```
        -f      Force module to load into the wrong kernel version
        -k      Make module autoclean-able
        -v      Verbose
        -q      Quiet
        -L      Lock to prevent simultaneous loads of a module
        -m      Output load map to stdout
        -o NAME Set internal module name to NAME
        -x      Do not export externs
```

**install**

install [-cdDsp] [-o USER] [-g GRP] [-m MODE] [source] dest|directory

Copy files and set attributes

Options:

```
-c      Just copy (default)
-d      Create directories
-D      Create leading target directories
-s      Strip symbol table
-p      Preserve date
-o USER Set ownership
-g GRP  Set group ownership
-m MODE Set permissions
```

**ionice**

ionice [-c 1-3] [-n 0-7] [-p PID] [PROG]

Change I/O scheduling class and priority

Options:

```
-c      Class. 1:realtime 2:best-effort 3:idle
-n      Priority
```

**ip**

ip [OPTIONS] {address | route | link | tunnel | rule} {COMMAND}

ip [OPTIONS] OBJECT {COMMAND} where OBJECT := {address | route | link | tunnel | rule} OPTIONS := { -f[amily] { inet | inet6 | link } | -o[neline] }

**ipaddr**

ipaddr { {add|del} IFADDR dev STRING | {show|flush}
[dev STRING] [to PREFIX] }

ipaddr {add|delete} IFADDR dev STRING ipaddr {show|flush} [dev STRING] [scope SCOPE-ID] [to PREFIX] [label PATTERN] IFADDR := PREFIX | ADDR peer PREFIX [broadcast ADDR] [anycast ADDR] [label STRING] [scope SCOPE-ID] SCOPE-ID := [host | link | global | NUMBER]

**ipcalc**

ipcalc [OPTIONS] ADDRESS[[/]NETMASK] [NETMASK]

Calculate IP network settings from a IP address

Options:

```
-b,--broadcast  Display calculated broadcast address
-n,--network    Display calculated network address
-m,--netmask    Display default netmask for IP
-p,--prefix     Display the prefix for IP/NETMASK
-h,--hostname   Display first resolved host name
-s,--silent     Don't ever display error messages
```

**ipcrm**

ipcrm [-MQS key] [-mqs id]

Upper-case options MQS remove an object by shmkey value. Lower-case options remove an object by shmid value.

Options:

```
        -mM     Remove memory segment after last detach
        -qQ     Remove message queue
        -sS     Remove semaphore
```

**ipcs**

ipcs [[-smq] -i shmid] | [[-asmq] [-tcplu]]

```
        -i      Show specific resource
Resource specification:
        -m      Shared memory segments
        -q      Message queues
        -s      Semaphore arrays
        -a      All (default)
Output format:
        -t      Time
        -c      Creator
        -p      Pid
        -l      Limits
        -u      Summary
```

**iplink**

iplink { set DEVICE { up | down | arp { on | off } | show [DEVICE] }

iplink set DEVICE { up | down | arp | multicast { on | off } | dynamic { on | off } | mtu MTU } iplink show [DEVICE]

**iproute**

iproute { list | flush | { add | del | change | append |
replace | monitor } ROUTE }

iproute { list | flush } SELECTOR iproute get ADDRESS [from ADDRESS iif STRING] [oif STRING] [tos TOS]
iproute { add | del | change | append | replace | monitor } ROUTE SELECTOR := [root PREFIX] [match PREFIX]
[proto RTPROTO] ROUTE := [TYPE] PREFIX [tos TOS] [proto RTPROTO] [metric METRIC]

**iprule**

iprule {[list | add | del] RULE}

iprule [list | add | del] SELECTOR
ACTION
SELECTOR := [from PREFIX] [to PREFIX] [tos TOS] [fwmark
FWMARK]
[dev STRING] [pref NUMBER]
ACTION := [table TABLE_ID] [nat ADDRESS]
[prohibit | reject | unreachable]
[realms [SRCREALM/]DSTREALM]
TABLE_ID := [local | main | default | NUMBER]

**iptunnel**

iptunnel { add | change | del | show } [NAME]
[mode { ipip | gre | sit }]
[remote ADDR] [local ADDR] [ttl TTL]

iptunnel { add | change | del | show } [NAME]
[mode { ipip | gre | sit }] [remote ADDR] [local ADDR]
[[i|o]seq] [[i|o]key KEY] [[i|o]csum]
[ttl TTL] [tos TOS] [[no]pmtudisc] [dev PHYS_DEV]

**kbd_mode**

kbd_mode [-a|k|s|u] [-C TTY]

Report or set the keyboard mode

Options:

```
-a      Default (ASCII)
-k      Medium-raw (keyboard)
-s      Raw (scancode)
-u      Unicode (utf-8)
-C TTY  Affect TTY instead of /dev/tty
```

**kill**

kill [-l] [-SIG] PID...

Send a signal (default: TERM) to given PIDs

Options:

```
-l      List all signal names and numbers
```

**killall**

killall [-l] [-q] [-SIG] process-name...

Send a signal (default: TERM) to given processes

Options:

```
-l      List all signal names and numbers
-q      Do not complain if no processes were killed
```

**killall5**

killall5 [-l] [-SIG] [-o PID]...

Send a signal (default: TERM) to all processes outside current session

Options:

```
-l      List all signal names and numbers
-o PID  Do not signal this PID
```

**klogd**

klogd [-c N] [-n]

Kernel logger

Options:

```
-c N    Only messages with level < N are printed to console
-n      Run in foreground
```

**last**

last [-HW] [-f file]

Show listing of the last users that logged into the system

Options:

```
-W      Display with no host column truncation
-f file Read from file instead of /var/log/wtmp
```

**length**

length STRING

Print STRING's length

**less**

less [-EMNmh~I?] [FILE]...

View a file or list of files. The position within files can be changed, and files can be manipulated in various ways.

Options:

```
-E      Quit once the end of a file is reached
-M,-m   Display status line with line numbers
        and percentage through the file
-N      Prefix line number to each line
-I      Ignore case in all searches
-~      Suppress ~s displayed past the end of the file
```

**ln**

ln [OPTIONS] TARGET... LINK|DIRECTORY

Create a link LINK or DIRECTORY/TARGET to the specified TARGET(s)

Options:

```
-s      Make symlinks instead of hardlinks
-f      Remove existing destination files
-n      Don't dereference symlinks - treat like normal file
-b      Make a backup of the target (if exists) before link operation
-S suf  Use suffix instead of ~ when making backup files
```

**loadfont**

loadfont < font

Load a console font from standard input

**loadkmap**

loadkmap < keymap

Load a binary keyboard translation table from standard input

**logger**

logger [OPTIONS] [MESSAGE]

Write MESSAGE to the system log. If MESSAGE is omitted, log stdin.

Options:

```
-s      Log to stderr as well as the system log
-t TAG  Log using the specified tag (defaults to user name)
-p PRIO Priority (numeric or facility.level pair)
```

**login**

login [-p] [-h HOST] [[-f] USER]

Begin a new session on the system

Options:

```
-f      Do not authenticate (user already authenticated)
-h      Name of the remote host
-p      Preserve environment
```

**logname**

logname

Print the name of the current user

**logread**

logread [OPTIONS]

Show messages in syslogd's circular buffer

Options:

```
-f      Output data as log grows
```

**losetup**

losetup [-o OFS] LOOPDEV FILE - associate loop devices
losetup -d LOOPDEV - disassociate
losetup [-f] - show

Options:

```
-o OFS  Start OFS bytes into FILE
-f      Show first free loop device
```

**lpd**

lpd SPOOLDIR [HELPER [ARGS]]

SPOOLDIR must contain (symlinks to) device nodes or directories with names matching print queue names. In the first case, jobs are sent directly to the device. Otherwise each job is stored in queue directory and HELPER program is called. Name of file to print is passed in $DATAFILE variable. Example:

```
tcpsvd -E 0 515 softlimit -m 999999 lpd /var/spool ./print
```

**lpq**

lpq [-P queue[@host[:port]]] [-U USERNAME] [-d JOBID...] [-fs]

Options:

```
-P      lp service to connect to (else uses $PRINTER)
-d      Delete jobs
-f      Force any waiting job to be printed
-s      Short display
```

**lpr**

lpr -P queue[@host[:port]] -U USERNAME -J TITLE -Vmh [FILE]...

Options:

```
-P      lp service to connect to (else uses $PRINTER)
-m      Send mail on completion
-h      Print banner page too
-V      Verbose
```

**ls**

ls [-1AacCdeFilnpLRrSsTtuvwxXhk] [FILE]...

List directory contents

Options:

```
-1      List in a single column
-A      Don't list . and ..
-a      Don't hide entries starting with .
-C      List by columns
-c      With -l: sort by ctime
--color[={always,never,auto}]   Control coloring
-d      List directory entries instead of contents
-e      List full date and time
-F      Append indicator (one of */=@|) to entries
-i      List inode numbers
-l      Long listing format
-n      List numeric UIDs and GIDs instead of names
-p      Append indicator (one of /=@|) to entries
-L      List entries pointed to by symlinks
-R      List subdirectories recursively
-r      Sort in reverse order
-S      Sort by file size
-s      List the size of each file, in blocks
-T NUM  Assume tabstop every NUM columns
-t      With -l: sort by modification time
-u      With -l: sort by access time
-v      Sort by version
-w NUM  Assume the terminal is NUM columns wide
-x      List by lines
-X      Sort by extension
-h      List sizes in human readable format (1K 243M 2G)
```

**lsattr**

lsattr [-Radlv] [FILE]...

List file attributes on an ext2 fs

Options:

```
-R      Recursively list subdirectories
-a      Do not hide entries starting with .
-d      List directory entries instead of contents
-l      List long flag names
-v      List the file's version/generation number
```

**lsmod**

lsmod

List the currently loaded kernel modules

**lzmacat**

lzmacat FILE

Uncompress to stdout

**lzop**

lzop [-cfvd123456789CF] [FILE]...

```
-c      Write to standard output
-f      Force
```

```
-v      Verbose
-d      Decompress
-F      Don't store or verify checksum
-C      Also write checksum of compressed block
-1..9   Compression level
```

**lzopcat**

lzopcat [-vCF] [FILE]...

```
-v      Verbose
-F      Don't store or verify checksum
```

**makemime**

makemime [OPTIONS] [FILE]...

Create multipart MIME-encoded message from FILEs

Options:

```
-o FILE Output. Default: stdout
-a HDR  Add header. Examples:
        "From: user@host.org", "Date: `date -R`"
-c CT   Content type. Default: text/plain
-C CS   Charset. Default: us-ascii
```

Other options are silently ignored

**man**

man [OPTIONS] [MANPAGE]...

Format and display manual page

Options:

```
-a      Display all pages
-w      Show page locations
```

**md5sum**

md5sum [OPTIONS] [FILE]... or: md5sum [OPTIONS] -c [FILE]

Print or check MD5 checksums

Options:

```
-c      Check sums against given list
-s      Don't output anything, status code shows success
-w      Warn about improperly formatted checksum lines
```

**mdev**

mdev [-s]

```
-s      Scan /sys and populate /dev during system boot
```

It can be run by kernel as a hotplug helper. To activate it: echo /bin/mdev >/proc/sys/kernel/hotplug It uses /etc/mdev.conf with lines [-]DEVNAME UID:GID PERM [>|=PATH] [@|$|*PROG]

**mesg**

mesg [y|n]

Control write access to your terminal

yAllow write access to your terminal
nDisallow write access to your terminal

**microcom**

microcom [-d DELAY] [-t TIMEOUT] [-s SPEED] [-X] TTY

Copy bytes for stdin to TTY and from TTY to stdout

Options:

```
-d      Wait up to DELAY ms for TTY output before sending every
        next byte to it
-t      Exit if both stdin and TTY are silent for TIMEOUT ms
-s      Set serial line to SPEED
-X      Disable special meaning of NUL and Ctrl-X from stdin
```

**mkdir**

mkdir [OPTIONS] DIRECTORY...

Create DIRECTORY

Options:

```
-m      Mode
-p      No error if exists; make parent directories as needed
```

**mkdosfs**

mkdosfs [-v] [-n LABEL] FILE_OR_DEVICE [SIZE_IN_KB]

Make a FAT32 filesystem

Options:

```
-v      Verbose
-n LBL  Volume label
```

**mkfifo**

mkfifo [OPTIONS] name

Create named pipe (identical to 'mknod name p')

Options:

```
-m MODE Mode (default a=rw)
```

**mkfs.minix**

mkfs.minix [-c | -l filename] [-nXX] [-iXX] /dev/name [blocks]

Make a MINIX filesystem

Options:

```
-c              Check device for bad blocks
-n [14|30]      Maximum length of filenames
-i INODES       Number of inodes for the filesystem
-l FILENAME     Read bad blocks list from FILENAME
-v              Make version 2 filesystem
```

**mkfs.vfat**

mkfs.vfat [-v] [-n LABEL] FILE_OR_DEVICE [SIZE_IN_KB]

Make a FAT32 filesystem

Options:

```
-v      Verbose
-n LBL  Volume label
```

**mknod**

mknod [OPTIONS] NAME TYPE MAJOR MINOR

Create a special file (block, character, or pipe)

Options:

```
-m      Create the special file using the specified mode (default a=rw)
TYPEs include:

b:      Make a block device
c or u: Make a character device
p:      Make a named pipe (MAJOR and MINOR are ignored)
```

**mkpasswd**

mkpasswd [OPTIONS] [PASSWORD] [SALT]

Crypt the PASSWORD using `crypt(3)`

Options:

```
-P,--password-fd=NUM    Read password from fd NUM
-m,--method=TYPE        Encryption method TYPE
-S,--salt=SALT
```

**mkswap**

mkswap DEVICE

Prepare block device to be used as swap partition

**mktemp**

mktemp [-dt] [-p DIR] [TEMPLATE]

Create a temporary file with name based on TEMPLATE and print its name. TEMPLATE must end with XXXXXX (e.g. [/dir/]nameXXXXXX).

Options:

```
-d      Make a directory instead of a file
-t      Generate a path rooted in temporary directory
-p DIR  Use DIR as a temporary directory (implies -t)
```

For -t or -p, directory is chosen as follows: $TMPDIR if set, else -p DIR, else /tmp

**modprobe**

modprobe [-knqrsvb] MODULE [symbol=value...]

Options:

```
-k      Make module autoclean-able
-n      Dry run
-q      Quiet
-r      Remove module (stacks) or do autoclean
-s      Report via syslog instead of stderr
```

```
        -v      Verbose
        -b      Apply blacklist to module names too
```

**more**

more [FILE]...

View FILE or standard input one screenful at a time

**mount**

mount [flags] DEVICE NODE [-o OPT,OPT]

Mount a filesystem. Filesystem autodetection requires /proc.

Options:

```
        -a              Mount all filesystems in fstab
        -f              Dry run
        -i              Don't run mount helper
        -r              Read-only mount
        -w              Read-write mount (default)
        -t FSTYPE       Filesystem type
        -O OPT          Mount only filesystems with option OPT (-a only)
-o OPT:
        loop            Ignored (loop devices are autodetected)
        [a]sync         Writes are [a]synchronous
        [no]atime       Disable/enable updates to inode access times
        [no]diratime    Disable/enable atime updates to directories
        [no]relatime    Disable/enable atime updates relative to modification time
        [no]dev         (Dis)allow use of special device files
        [no]exec        (Dis)allow use of executable files
        [no]suid        (Dis)allow set-user-id-root programs
        [r]shared       Convert [recursively] to a shared subtree
        [r]slave        Convert [recursively] to a slave subtree
        [r]private      Convert [recursively] to a private subtree
        [un]bindable    Make mount point [un]able to be bind mounted
        bind            Bind a directory to an additional location
        move            Relocate an existing mount point
        remount         Remount a mounted filesystem, changing its flags
        ro/rw           Read-only/read-write mount
```

There are EVEN MORE flags that are specific to each filesystem You'll have to see the written documentation for those filesystems

**mountpoint**

mountpoint [-q] <[-dn] DIR | -x DEVICE>

Check if the directory is a mountpoint

Options:

```
        -q      Quiet
        -d      Print major/minor device number of the filesystem
        -n      Print device name of the filesystem
        -x      Print major/minor device number of the blockdevice
```

**mt**

mt [-f device] opcode value

Control magnetic tape drive operation

Available Opcodes:

bsf bsfm bsr bss datacompression drvbuffer eof eom erase fsf fsfm fsr fss load lock mkpart nop offline ras1 ras2 ras3 reset retension rewind rewoffline seek setblk setdensity setpart tell unload unlock weof wset

**mv**

mv [OPTIONS] SOURCE DEST or: mv [OPTIONS] SOURCE... DIRECTORY

Rename SOURCE to DEST, or move SOURCE(s) to DIRECTORY

Options:

```
-f      Don't prompt before overwriting
-i      Interactive, prompt before overwrite
```

**nameif**

nameif [-s] [-c FILE] [{IFNAME MACADDR}]

Rename network interface while it in the down state

Options:

```
-c FILE         Use configuration file (default: /etc/mactab)
-s              Use syslog (LOCAL0 facility)
IFNAME MACADDR  new_interface_name interface_mac_address
```

**nc**

nc [OPTIONS] HOST PORT - connect nc [OPTIONS] -l -p PORT [HOST] [PORT] - listen

Options:

```
-e PROG         Run PROG after connect (must be last)
-l              Listen mode, for inbound connects
-n              Don't do DNS resolution
-s ADDR         Local address
-p PORT         Local port
-u              UDP mode
-v              Verbose
-w SEC          Timeout for connects and final net reads
-i SEC          Delay interval for lines sent
-o FILE         Hex dump traffic
-z              Zero-I/O mode (scanning)
```

**netstat**

netstat [-laentuwxrWp]

Display networking information

Options:

```
-l      Display listening server sockets
-a      Display all sockets (default: connected)
-e      Display other/more information
-n      Don't resolve names
-t      Tcp sockets
-u      Udp sockets
-w      Raw sockets
-x      Unix sockets
-r      Display routing table
-W      Display with no column truncation
-p      Display PID/Program name for sockets
```

**nice**

nice [-n ADJUST] [PROG [ARGS]]

Run PROG with modified scheduling priority

Options:

```
-n ADJUST       Adjust priority by ADJUST
```

**nmeter**

nmeter format_string

Monitor system in real time

Format specifiers:

```
%Nc or %[cN]   Monitor CPU. N - bar size, default 10
               (displays: S:system U:user N:niced D:iowait I:irq i:softirq)
%[niface]      Monitor network interface 'iface'
%m             Monitor allocated memory
%[mf]          Monitor free memory
%[mt]          Monitor total memory
%s             Monitor allocated swap
%f             Monitor number of used file descriptors
%Ni            Monitor total/specific IRQ rate
%x             Monitor context switch rate
%p             Monitor forks
%[pn]          Monitor # of processes
%b             Monitor block io
%Nt            Show time (with N decimal points)
%Nd            Milliseconds between updates (default:1000)
%r             Print <cr> instead of <lf> at EOL
```

**nohup**

nohup PROG [ARGS]

Run PROG immune to hangups, with output to a non-tty

**nslookup**

nslookup [HOST] [SERVER]

Query the nameserver for the IP address of the given HOST optionally using a specified DNS server

**od**

od [-aBbcDdeFfHhIiLlOovXx] [-t TYPE] [FILE]

Write an unambiguous representation, octal bytes by default, of FILE to standard output. With no FILE or when FILE is -, read standard input.

**openvt**

openvt [-c N] [-sw] [PROG [ARGS]]

Start PROG on a new virtual terminal

Options:

```
-c N    Use specified VT
-s      Switch to the VT
-w      Wait for PROG to exit
```

**passwd**

passwd [OPTIONS] [USER]

Change USER's password. If no USER is specified, changes the password for the current user.

Options:

```
-a      Algorithm to use for password (choices: des, md5)
-d      Delete password for the account
-l      Lock (disable) account
-u      Unlock (re-enable) account
```

**patch**

patch [-p NUM] [-i DIFF] [-R] [-N]

```
-p NUM  Strip NUM leading components from file names
-i DIFF Read DIFF instead of stdin
-R      Reverse patch
-N      Ignore already applied patches
```

**pgrep**

pgrep [-flnovx] [-s SID|-P PPID|PATTERN]

Display process(es) selected by regex PATTERN

Options:

```
-l      Show command name too
-f      Match against entire command line
-n      Show the newest process only
-o      Show the oldest process only
-v      Negate the match
-x      Match whole name (not substring)
-s      Match session ID (0 for current)
-P      Match parent process ID
```

**pidof**

pidof [OPTIONS] [NAME...]

List PIDs of all processes with names that match NAMEs

Options:

```
-s      Show only one PID
-o PID  Omit given pid
        Use %PPID to omit pid of pidof's parent
```

**ping**

ping [OPTIONS] HOST

Send ICMP ECHO_REQUEST packets to network hosts

Options:

```
-4, -6          Force IPv4 or IPv6 hostname resolution
-c CNT          Send only CNT pings
-s SIZE         Send SIZE data bytes in packets (default:56)
-I IFACE/IP     Use interface or IP address as source
-W SEC          Seconds to wait for the first response (default:10)
                (after all -c CNT packets are sent)
-w SEC          Seconds until ping exits (default:infinite)
                (can exit earlier with -c CNT)
-q              Quiet, only displays output at start
                and when finished
```

**ping6**

ping6 [OPTIONS] HOST

Send ICMP ECHO_REQUEST packets to network hosts

Options:

```
-c CNT          Send only CNT pings
-s SIZE         Send SIZE data bytes in packets (default:56)
-I IFACE/IP     Use interface or IP address as source
-q              Quiet, only displays output at start
                and when finished
```

**pivot_root**

pivot_root NEW_ROOT PUT_OLD

Move the current root file system to PUT_OLD and make NEW_ROOT the new root file system

**pkill**

pkill [-l|-SIGNAL] [-fnovx] [-s SID|-P PPID|PATTERN]

Send a signal to process(es) selected by regex PATTERN

Options:

```
-l      List all signals
-f      Match against entire command line
-n      Signal the newest process only
-o      Signal the oldest process only
-v      Negate the match
-x      Match whole name (not substring)
-s      Match session ID (0 for current)
-P      Match parent process ID
```

**popmaildir**

popmaildir [OPTIONS] Maildir [connection-helper ...]

Fetch content of remote mailbox to local maildir

Options:

```
-b              Binary mode. Ignored
-d              Debug. Ignored
-m              Show used memory. Ignored
-V              Show version. Ignored
-c              Use tcpclient. Ignored
-a              Use APOP protocol. Implied. If server supports APOP -> use it
-s              Skip authorization
-T              Get messages with TOP instead with RETR
-k              Keep retrieved messages on the server
-t timeout      Network timeout
-F "program arg1 arg2 ..."      Filter by program. May be multiple
-M "program arg1 arg2 ..."      Deliver by program
-R size         Remove old messages on the server >= size (in bytes). Ignored
-Z N1-N2        Remove messages from N1 to N2 (dangerous). Ignored
-L size         Do not retrieve new messages >= size (in bytes). Ignored
-H lines        Type specified number of lines of a message. Ignored
```

**printenv**

printenv [VARIABLE...]

Print all or part of environment. If no environment VARIABLE specified, print them all.

**printf**

printf FORMAT [ARGUMENT...]

Format and print ARGUMENT(s) according to FORMAT, where FORMAT controls the output exactly as in C printf

**ps**

ps

Report process status

Options:

```
-o col1,col2=header     Select columns for display
-T                      Show threads
```

**pscan**

pscan [-cb] [-p MIN_PORT] [-P MAX_PORT] [-t TIMEOUT] [-T MIN_RTT] HOST

Scan a host, print all open ports

Options:

```
-c      Show closed ports too
-b      Show blocked ports too
-p      Scan from this port (default 1)
-P      Scan up to this port (default 1024)
-t      Timeout (default 5000 ms)
-T      Minimum rtt (default 5 ms, increase for congested hosts)
```

**pwd**

pwd

Print the full filename of the current working directory

**raidautorun**

raidautorun DEVICE

Tell the kernel to automatically search and start RAID arrays

**rdate**

rdate [-sp] HOST

Get and possibly set the system date and time from a remote HOST

Options:

```
-s      Set the system date and time (default)
-p      Print the date and time
```

**rdev**

rdev

Print the device node associated with the filesystem mounted at '/'

**readlink**

readlink [-fnv] FILE

Display the value of a symlink

Options:

```
-f      Canonicalize by following all symlinks
-n      Don't add newline
-v      Verbose
```

**readprofile**

readprofile [OPTIONS]

Options:

```
-m mapfile      (Default: /boot/System.map)
-p profile      (Default: /proc/profile)
-M mult         Set the profiling multiplier to mult
-i              Print only info about the sampling step
-v              Verbose
-a              Print all symbols, even if count is 0
-b              Print individual histogram-bin counts
-s              Print individual counters within functions
-r              Reset all the counters (root only)
-n              Disable byte order auto-detection
```

**realpath**

realpath pathname...

Return the absolute pathnames of given argument

**reformime**

reformime [OPTIONS] [FILE]...

Parse MIME-encoded message

Options:

```
-x prefix       Extract content of MIME sections to files
-X prog [args]  Filter content of MIME sections through prog.
                Must be the last option
```

Other options are silently ignored.

**renice**

renice {{-n INCREMENT} | PRIORITY} [[-p | -g | -u] ID...]

Change priority of running processes

Options:

```
-n      Adjust current nice value (smaller is faster)
-p      Process id(s) (default)
-g      Process group id(s)
-u      Process user name(s) and/or id(s)
```

**reset**

reset

Reset the screen

**resize**

resize

Resize the screen

**rm**

rm [OPTIONS] FILE...

Remove (unlink) the FILE(s). Use '--' to indicate that all following arguments are non-options.

Options:

```
-i      Always prompt before removing
-f      Never prompt
-r,-R   Remove directories recursively
```

**rmdir**

rmdir [OPTIONS] DIRECTORY...

Remove the DIRECTORY, if it is empty

Options:

```
-p|--parents    Include parents
--ignore-fail-on-non-empty
```

**rmmod**

rmmod [OPTIONS] [MODULE]...

Unload the specified kernel modules from the kernel

Options:

```
-w      Wait until the module is no longer used
-f      Force unloading
-a      Remove all unused modules (recursively)
```

**route**

route [{add|del|delete}]

Edit kernel routing tables

Options:

```
-n      Don't resolve names
-e      Display other/more information
-A inet{6}      Select address family
```

**rpm**

rpm -i -q[ildc]p package.rpm

Manipulate RPM packages

Options:

```
-i      Install package
-q      Query package
-p      Query uninstalled package
-i      Show information
-l      List contents
-d      List documents
-c      List config files
```

**rpm2cpio**

rpm2cpio package.rpm

Output a cpio archive of the rpm file

**rtcwake**

rtcwake [-a | -l | -u] [-d DEV] [-m MODE] [-s SEC | -t TIME]

Enter a system sleep state until specified wakeup time

```
-a,--auto       Read clock mode from adjtime
-l,--local      Clock is set to local time
-u,--utc        Clock is set to UTC time
-d,--device=DEV Specify the RTC device
-m,--mode=MODE  Set the sleep state (default: standby)
-s,--seconds=SEC Set the timeout in SEC seconds from now
-t,--time=TIME  Set the timeout to TIME seconds from epoch
```

**run-parts**

run-parts [-t] [-l] [-a ARG] [-u MASK] DIRECTORY

Run a bunch of scripts in a directory

Options:

```
-t      Print what would be run, but don't actually run anything
-a ARG  Pass ARG as argument for every program
-u MASK Set the umask to MASK before running every program
-l      Print names of all matching files even if they are not executable
```

**runlevel**

runlevel [utmp]

Find the current and previous system runlevel

If no utmp file exists or if no runlevel record can be found, print "unknown"

**runsv**

runsv dir

Start and monitor a service and optionally an appendant log service

**runsvdir**

runsvdir [-P] [-s SCRIPT] dir

Start a runsv process for each subdirectory. If it exits, restart it.

```
-P              Put each runsv in a new session
-s SCRIPT       Run SCRIPT <signo> after signal is processed
```

**rx**

rx FILE

Receive a file using the xmodem protocol

**script**

script [-afqt] [-c PROG] [OUTFILE]

Options:

```
-a      Append output
-c      Run PROG, not shell
-f      Flush output after each write
-q      Quiet
-t      Send timing to stderr
```

**scriptreplay**

scriptreplay timingfile [typescript [divisor]]

Play back typescripts, using timing information

**sed**

sed [-efinr] SED_CMD [FILE]...

Options:

```
-e CMD  Add CMD to sed commands to be executed
-f FILE Add FILE contents to sed commands to be executed
-i      Edit files in-place
-n      Suppress automatic printing of pattern space
-r      Use extended regex syntax
```

If no -e or -f is given, the first non-option argument is taken as the sed command to interpret. All remaining arguments are names of input files; if no input files are specified, then the standard input is read. Source files will not be modified unless -i option is given.

**sendmail**

sendmail [OPTIONS] [RECIPIENT_EMAIL]...

Read email from stdin and send it

Standard options:

```
-t              Read additional recipients from message body
-f sender       Sender (required)
-o options      Various options. -oi implied, others are ignored
```

Busybox specific options:

```
-w seconds      Network timeout
-H 'PROG ARGS'  Run connection helper
                Examples:
                -H 'exec openssl s_client -quiet -tls1 -starttls smtp
                        -connect smtp.gmail.com:25' <email.txt
                        [4<username_and_passwd.txt | -au<username> -ap<password>]
                -H 'exec openssl s_client -quiet -tls1
                        -connect smtp.gmail.com:465' <email.txt
                        [4<username_and_passwd.txt | -au<username> -ap<password>]
-S server[:port] Server
-au<username>   Username for AUTH LOGIN
-ap<password>   Password for AUTH LOGIN
-am<method>     Authentication method. Ignored. LOGIN is implied
```

Other options are silently ignored; -oi -t is implied Use makemime applet to create message with attachments

**seq**

seq [-w] [-s SEP] [FIRST [INC]] LAST

Print numbers from FIRST to LAST, in steps of INC. FIRST, INC default to 1

Options:

```
-w      Pad to last with leading zeros
-s SEP  String separator
```

**setarch**

setarch personality program [args...]

Personality may be:

```
linux32         Set 32bit uname emulation
linux64         Set 64bit uname emulation
```

**setconsole**

setconsole [-r|--reset] [DEVICE]

Redirect system console output to DEVICE (default: /dev/tty)

Options:

```
-r      Reset output to /dev/console
```

**setfont**

setfont FONT [-m MAPFILE] [-C TTY]

Load a console font

Options:

```
-m MAPFILE      Load console screen map
-C TTY          Affect TTY instead of /dev/tty
```

**setkeycodes**

setkeycodes SCANCODE KEYCODE...

Set entries into the kernel's scancode-to-keycode map, allowing unusual keyboards to generate usable keycodes.

SCANCODE may be either xx or e0xx (hexadecimal), and KEYCODE is given in decimal

**setlogcons**

setlogcons N

Redirect the kernel output to console N (0 for current)

**setsid**

setsid PROG [ARG...]

Run PROG in a new session. PROG will have no controlling terminal and will not be affected by keyboard signals (Ctrl-C etc). See setsid(2) for details.

**setuidgid**

setuidgid account prog args

Set uid and gid to account's uid and gid, removing all supplementary groups and run PROG

**sha1sum**

sha1sum [OPTIONS] [FILE]... or: sha1sum [OPTIONS] -c [FILE]

Print or check SHA1 checksums

Options:

```
-c      Check sums against given list
-s      Don't output anything, status code shows success
-w      Warn about improperly formatted checksum lines
```

**sha256sum**

sha256sum [OPTIONS] [FILE]... or: sha256sum [OPTIONS] -c [FILE]

Print or check SHA256 checksums

Options:

```
-c      Check sums against given list
-s      Don't output anything, status code shows success
-w      Warn about improperly formatted checksum lines
```

**sha512sum**

sha512sum [OPTIONS] [FILE]... or: sha512sum [OPTIONS] -c [FILE]

Print or check SHA512 checksums

Options:

```
-c      Check sums against given list
-s      Don't output anything, status code shows success
-w      Warn about improperly formatted checksum lines
```

**showkey**

showkey [-a | -k | -s]

Show keys pressed

Options:

```
-a      Display decimal/octal/hex values of the keys
-k      Display interpreted keycodes (default)
-s      Display raw scan-codes
```

**slattach**

slattach [-cehmLF] [-s SPEED] [-p PROTOCOL] DEVICE

Attach network interface(s) to serial line(s)

Options:

```
-p PROT Set protocol (slip, cslip, slip6, clisp6 or adaptive)
-s SPD  Set line speed
-e      Exit after initializing device
-h      Exit when the carrier is lost
-c PROG Run PROG when the line is hung up
-m      Do NOT initialize the line in raw 8 bits mode
-L      Enable 3-wire operation
-F      Disable RTS/CTS flow control
```

**sleep**

sleep [N]...

Pause for a time equal to the total of the args given, where each arg can have an optional suffix of (s)econds, (m)inutes, (h)ours, or (d)ays

**softlimit**

softlimit [-a BYTES] [-m BYTES] [-d BYTES] [-s BYTES] [-l BYTES] [-f BYTES] [-c BYTES] [-r BYTES] [-o N] [-p N] [-t N] PROG ARGS

Set soft resource limits, then run PROG

Options:

```
        -a BYTES        Limit total size of all segments
        -m BYTES        Same as -d BYTES -s BYTES -l BYTES -a BYTES
        -d BYTES        Limit data segment
        -s BYTES        Limit stack segment
        -l BYTES        Limit locked memory size
        -o N            Limit number of open files per process
        -p N            Limit number of processes per uid
Options controlling file sizes:

        -f BYTES        Limit output file sizes
        -c BYTES        Limit core file size
Efficiency opts:

        -r BYTES        Limit resident set size
        -t N            Limit CPU time, process receives
                        a SIGXCPU after N seconds
```

**sort**

sort [-nrugMcszbdfimSTokt] [-o FILE] [-k start[.offset][opts][,end[.offset][opts]] [-t CHAR] [FILE]...

Sort lines of text

Options:

```
        -b      Ignore leading blanks
        -c      Check whether input is sorted
        -d      Dictionary order (blank or alphanumeric only)
        -f      Ignore case
        -g      General numerical sort
        -i      Ignore unprintable characters
        -k      Sort key
        -M      Sort month
        -n      Sort numbers
        -o      Output to file
        -k      Sort by key
        -t CHAR Key separator
        -r      Reverse sort order
        -s      Stable (don't sort ties alphabetically)
        -u      Suppress duplicate lines
        -z      Lines are terminated by NUL, not newline
        -mST    Ignored for GNU compatibility
```

**split**

split [OPTIONS] [INPUT [PREFIX]]

Options:

```
        -b n[k|m]       Split by bytes
        -l n            Split by lines
        -a n            Use n letters as suffix
```

**start-stop-daemon**

start-stop-daemon [OPTIONS] [-S|-K] ... [-- arguments...]

Search for matching processes, and then -K: stop all matching processes. -S: start a process unless a matching process is found.

Process matching:

```
        -u,--user USERNAME|UID  Match only this user's processes
        -n,--name NAME          Match processes with NAME
                                in comm field in /proc/PID/stat
        -x,--exec EXECUTABLE    Match processes with this command
                                in /proc/PID/cmdline
        -p,--pidfile FILE       Match a process with PID from the file
        All specified conditions must match
-S only:
        -x,--exec EXECUTABLE    Program to run
        -a,--startas NAME       Zeroth argument
        -b,--background         Background
        -N,--nicelevel N        Change nice level
        -c,--chuid USER[:[GRP]] Change to user/group
        -m,--make-pidfile       Write PID to the pidfile specified by -p
-K only:
        -s,--signal SIG         Signal to send
        -t,--test               Match only, exit with 0 if a process is found
Other:

        -o,--oknodo             Exit with status 0 if nothing is done
        -v,--verbose            Verbose
        -q,--quiet              Quiet
```

**stat**

stat [OPTIONS] FILE...

Display file (default) or filesystem status

Options:

```
        -c fmt  Use the specified format
        -f      Display filesystem status
        -L      Dereference links
        -t      Display info in terse form
```

Valid format sequences for files:

```
 %a     Access rights in octal
 %A     Access rights in human readable form
 %b     Number of blocks allocated (see %B)
 %B     The size in bytes of each block reported by %b
 %d     Device number in decimal
 %D     Device number in hex
 %f     Raw mode in hex
 %F     File type
 %g     Group ID of owner
 %G     Group name of owner
 %h     Number of hard links
 %i     Inode number
 %n     File name
 %N     Quoted file name with dereference if symlink
 %o     I/O block size
 %s     Total size, in bytes
 %t     Major device type in hex
 %T     Minor device type in hex
 %u     User ID of owner
 %U     User name of owner
 %x     Time of last access
 %X     Time of last access as seconds since Epoch
 %y     Time of last modification
 %Y     Time of last modification as seconds since Epoch
 %z     Time of last change
 %Z     Time of last change as seconds since Epoch
```

Valid format sequences for file systems:

```
%a     Free blocks available to non-superuser
%b     Total data blocks in file system
%c     Total file nodes in file system
%d     Free file nodes in file system
%f     Free blocks in file system
%i     File System ID in hex
%l     Maximum length of filenames
%n     File name
%s     Block size (for faster transfer)
%S     Fundamental block size (for block counts)
%t     Type in hex
%T     Type in human readable form
```

**strings**

strings [-afo] [-n LEN] [FILE]...

Display printable strings in a binary file

Options:

```
-a      Scan whole file (default)
-f      Precede strings with filenames
-n LEN  At least LEN characters form a string (default 4)
-o      Precede strings with decimal offsets
```

**stty**

stty [-a|g] [-F DEVICE] [SETTING]...

Without arguments, prints baud rate, line discipline, and deviations from stty sane

Options:

```
-F DEVICE       Open device instead of stdin
-a              Print all current settings in human-readable form
-g              Print in stty-readable form
[SETTING]       See manpage
```

**su**

su [OPTIONS] [-] [username]

Change user id or become root

Options:

```
-p, -m  Preserve environment
-c CMD  Command to pass to 'sh -c'
-s SH   Shell to use instead of default shell
```

**sulogin**

sulogin [OPTIONS] [TTY]

Single user login

Options:

```
-t N    Timeout
```

**sum**

sum [-rs] [FILE]...

Checksum and count the blocks in a file

Options:

```
-r      Use BSD sum algorithm (1K blocks)
-s      Use System V sum algorithm (512byte blocks)
```

**sv**

sv [-v] [-w sec] command service...

Control services monitored by runsv supervisor. Commands (only first character is enough):

status: query service status up: if service isn't running, start it. If service stops, restart it once: like 'up', but if service stops, don't restart it down: send TERM and CONT signals. If ./run exits, start ./finish if it exists. After it stops, do not restart service exit: send TERM and CONT signals to service and log service. If they exit, runsv exits too pause, cont, hup, alarm, interrupt, quit, 1, 2, term, kill: send STOP, CONT, HUP, ALRM, INT, QUIT, USR1, USR2, TERM, KILL signal to service

**svlogd**

svlogd [-ttv] [-r c] [-R abc] [-l len] [-b buflen] dir...

Continuously read log data from standard input, optionally filter log messages, and write the data to one or more automatically rotated logs

**swapoff**

swapoff [-a] [DEVICE]

Stop swapping on DEVICE

Options:

```
-a      Stop swapping on all swap devices
```

**swapon**

swapon [-a] [-p pri] [DEVICE]

Start swapping on DEVICE

Options:

```
-a      Start swapping on all swap devices
-p pri  Set swap device priority
```

**switch_root**

switch_root [-c /dev/console] NEW_ROOT NEW_INIT [ARGS]

Free initramfs and switch to another root fs:

chroot to NEW_ROOT, delete all in /, move NEW_ROOT to /, execute NEW_INIT. PID must be 1. NEW_ROOT must be a mountpoint.

Options:

```
-c DEV  Reopen stdio to DEV after switch
```

**sync**

sync

Write all buffered blocks to disk

**sysctl**

sysctl [OPTIONS] [VALUE]...

Configure kernel parameters at runtime

Options:

```
-n      Don't print key names
-e      Don't warn about unknown keys
-w      Change sysctl setting
-p FILE Load sysctl settings from FILE (default /etc/sysctl.conf)
-a      Display all values
-A      Display all values in table form
```

**syslogd**

syslogd [OPTIONS]

System logging utility. Note that this version of syslogd ignores /etc/syslog.conf.

Options:

```
-n              Run in foreground
-O FILE         Log to given file (default:/var/log/messages)
-l n            Set local log level
-S              Smaller logging output
-s SIZE         Max size (KB) before rotate (default:200KB, 0=off)
-b NUM          Number of rotated logs to keep (default:1, max=99, 0=purge)
-R HOST[:PORT]  Log to IP or hostname on PORT (default PORT=514/UDP)
-L              Log locally and via network (default is network only if -R)
-D              Drop duplicates
-C[size(KiB)]   Log to shared mem buffer (read it using logread)
```

**tac**

tac [FILE]...

Concatenate FILE(s) and print them in reverse

**tail**

tail [OPTIONS] [FILE]...

Print 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.

Options:

```
-c N[kbm]       Output the last N bytes
-n N[kbm]       Print last N lines instead of last 10
-f              Output data as the file grows
-q              Never output headers giving file names
-s SEC          Wait SEC seconds between reads with -f
-v              Always output headers giving file names
```

If the first character of N (bytes or lines) is a '+', output begins with the Nth item from the start of each file, otherwise, print the last N items in the file. N bytes may be suffixed by k (x1024), b (x512), or m (1024^2).

**tar**

tar -[czjaZxtvO] [-X FILE] [-f TARFILE] [-C DIR] [FILE(s)]...

Create, extract, or list files from a tar file

Options:

```
        c       Create
        x       Extract
        t       List
Archive format selection:

        z       Filter the archive through gzip
        j       Filter the archive through bzip2
        a       Filter the archive through lzma
        Z       Filter the archive through compress
File selection:

        f       Name of TARFILE or "-" for stdin
        O       Extract to stdout
        exclude File to exclude
        X       File with names to exclude
        C       Change to directory DIR before operation
        v       Verbose
```

**taskset**

taskset [-p] [MASK] [PID | PROG [ARGS]]

Set or get CPU affinity

Options:

```
        -p      Operate on an existing PID
```

**tcpsvd**

tcpsvd [-hEv] [-c N] [-C N[:MSG]] [-b N] [-u USER] [-l NAME] IP PORT PROG

Create TCP socket, bind to IP:PORT and listen for incoming connection. Run PROG for each connection.

```
        IP              IP to listen on. '0' = all
        PORT            Port to listen on
        PROG [ARGS]     Program to run
        -l NAME         Local hostname (else looks up local hostname in DNS)
        -u USER[:GRP]   Change to user/group after bind
        -c N            Handle up to N connections simultaneously
        -b N            Allow a backlog of approximately N TCP SYNs
        -C N[:MSG]      Allow only up to N connections from the same IP
                        New connections from this IP address are closed
                        immediately. MSG is written to the peer before close
        -h              Look up peer's hostname
        -E              Do not set up environment variables
        -v              Verbose
```

**tee**

tee [OPTIONS] [FILE]...

Copy standard input to each FILE, and also to standard output

Options:

```
        -a      Append to the given FILEs, do not overwrite
        -i      Ignore interrupt signals (SIGINT)
```

**telnet**

telnet [-a] [-l USER] HOST [PORT]

Connect to telnet server

Options:

```
-a      Automatic login with $USER variable
-l USER Automatic login as USER
```

**telnetd**

telnetd [OPTIONS]

Handle incoming telnet connections

Options:

```
-l LOGIN        Exec LOGIN on connect
-f ISSUE_FILE   Display ISSUE_FILE instead of /etc/issue
-K              Close connection as soon as login exits
                (normally wait until all programs close slave pty)
-p PORT         Port to listen on
-b ADDR[:PORT]  Address to bind to
-F              Run in foreground
-i              Run as inetd service
-w SEC          Run as inetd service in wait mode, linger time SEC
-S              Log to syslog (implied by -i or without -F and -w)
```

**test**

test EXPRESSION ]

Check file types, compare values etc. Return a 0/1 exit code depending on logical value of EXPRESSION

**tftp**

tftp [OPTIONS] HOST [PORT]

Transfer a file from/to tftp server

Options:

```
-l FILE Local FILE
-r FILE Remote FILE
-g      Get file
-p      Put file
-b SIZE Transfer blocks of SIZE octets
```

**tftpd**

tftpd [-cr] [-u USER] [DIR]

Transfer a file on tftp client's request

tftpd should be used as an inetd service. tftpd's line for inetd.conf: 69 dgram udp nowait root tftpd tftpd /files/to/serve
It also can be ran from udpsvd:

```
udpsvd -vE 0.0.0.0 69 tftpd /files/to/serve
```

Options:

```
-r      Prohibit upload
-c      Allow file creation via upload
-u      Access files as USER
```

**time**

time [OPTIONS] PROG [ARGS]

Run PROG. When it finishes, its resource usage is displayed.

Options:

```
-v      Verbose
```

**timeout**

timeout [-t SECS] [-s SIG] PROG [ARGS]

Runs PROG. Sends SIG to it if it is not gone in SECS seconds. Defaults: SECS: 10, SIG: TERM.

**top**

top [-b] [-nCOUNT] [-dSECONDS] [-m]

Provide a view of process activity in real time. Read the status of all processes from /proc each SECONDS and show the status for however many processes will fit on the screen.

**touch**

touch [-c] [-d DATE] FILE [FILE]...

Update the last-modified date on the given FILE[s]

Options:

```
-c      Do not create files
-d DT   Date/time to use
```

**tr**

tr [-cds] STRING1 [STRING2]

Translate, squeeze, and/or delete characters from standard input, writing to standard output

Options:

```
-c      Take complement of STRING1
-d      Delete input characters coded STRING1
-s      Squeeze multiple output characters of STRING2 into one character
```

**traceroute**

traceroute [-FIldnrv] [-f 1st_ttl] [-m max_ttl] [-p port#] [-q nqueries]
[-s src_addr] [-t tos] [-w wait] [-g gateway] [-i iface]
[-z pausemsecs] HOST [data size]

Trace the route to HOST

Options:

```
-F      Set the don't fragment bit
-I      Use ICMP ECHO instead of UDP datagrams
-l      Display the ttl value of the returned packet
-d      Set SO_DEBUG options to socket
-n      Print hop addresses numerically rather than symbolically
-r      Bypass the normal routing tables and send directly to a host
-v      Verbose
-m max_ttl      Max time-to-live (max number of hops)
-p port#        Base UDP port number used in probes
                (default 33434)
-q nqueries     Number of probes per 'ttl' (default 3)
-s src_addr     IP address to use as the source address
-t tos          Type-of-service in probe packets (default 0)
-w wait         Time in seconds to wait for a response
                (default 3 sec)
-g              Loose source route gateway (8 max)
```

**true**

true

Return an exit code of TRUE (0)

**tty**

tty

Print file name of standard input's terminal

Options:

```
-s      Print nothing, only return exit status
```

**ttysize**

ttysize [w] [h]

Print dimension(s) of standard input's terminal, on error return 80x25

**udhcpc**

udhcpc [-Cfbnqtvo] [-c CID] [-V VCLS] [-H HOSTNAME] [-i INTERFACE] [-p pidfile] [-r IP] [-s script] [-O dhcp-option]...

```
-V,--vendorclass=CLASSID        Vendor class identifier
-i,--interface=INTERFACE        Interface to use (default eth0)
-H,-h,--hostname=HOSTNAME       Client hostname
-c,--clientid=CLIENTID  Client identifier
-C,--clientid-none      Suppress default client identifier
-p,--pidfile=file       Create pidfile
-r,--request=IP         IP address to request
-s,--script=file        Run file at DHCP events (default /usr/share/udhcpc/default.script)
-t,--retries=N          Send up to N request packets
-T,--timeout=N          Try to get a lease for N seconds (default 3)
-A,--tryagain=N         Wait N seconds (default 20) after failure
-O,--request-option=OPT Request DHCP option OPT (cumulative)
-o,--no-default-options Do not request any options (unless -O is also given)
-f,--foreground Run in foreground
-b,--background Background if lease is not immediately obtained
-S,--syslog     Log to syslog too
-n,--now        Exit with failure if lease is not immediately obtained
-q,--quit       Quit after obtaining lease
-R,--release    Release IP on quit
-a,--arping     Use arping to validate offered address
```

**udhcpd**

udhcpd [-fS] [configfile]

DHCP server

```
-f      Run in foreground
-S      Log to syslog too
```

**udpsvd**

udpsvd [-hEv] [-c N] [-u USER] [-l NAME] IP PORT PROG

Create UDP socket, bind to IP:PORT and wait for incoming packets. Run PROG for each packet, redirecting all further packets with same peer ip:port to it.

```
IP              IP to listen on. '0' = all
PORT            Port to listen on
PROG [ARGS]     Program to run
-l NAME         Local hostname (else looks up local hostname in DNS)
-u USER[:GRP]   Change to user/group after bind
-c N            Handle up to N connections simultaneously
-h              Look up peer's hostname
-E              Do not set up environment variables
-v              Verbose
```

**umount**

umount [flags] FILESYSTEM|DIRECTORY

Unmount file systems

Options:

```
-a      Unmount all file systems
-r      Try to remount devices as read-only if mount is busy
-l      Lazy umount (detach filesystem)
-f      Force umount (i.e., unreachable NFS server)
-d      Free loop device if it has been used
```

**uname**

uname [-amnrspv]

Print system information

Options:

```
-a      Print all
-m      The machine (hardware) type
-n      Hostname
-r      OS release
-s      OS name (default)
-p      Processor type
-v      OS version
```

**uncompress**

uncompress [-c] [-f] [name...]

Uncompress .Z file[s]

Options:

```
-c      Extract to stdout
-f      Overwrite an existing file
```

**unexpand**

unexpand [-f][-a][-t NUM] [FILE|-]

Convert spaces to tabs, writing to standard output

Options:

```
-a,--all        Convert all blanks
-f,--first-only Convert only leading blanks
-t,--tabs=N     Tabstops every N chars
```

**uniq**

uniq [-fscduw]... [INPUT [OUTPUT]]

Discard duplicate lines

Options:

```
-c      Prefix lines by the number of occurrences
-d      Only print duplicate lines
-u      Only print unique lines
-f N    Skip first N fields
-s N    Skip first N chars (after any skipped fields)
-w N    Compare N characters in line
```

**unix2dos**

unix2dos [OPTION] [FILE]

Convert FILE in-place from Unix to DOS format. When no file is given, use stdin/stdout.

Options:

```
-u      dos2unix
-d      unix2dos
```

**unlzma**

unlzma [OPTIONS] [FILE]

Uncompress FILE (or standard input if FILE is '-' or omitted)

Options:

```
-c      Write to standard output
-f      Force
```

**unlzop**

unlzop [-cfvCF] [FILE]...

```
-c      Write to standard output
-f      Force
-v      Verbose
-F      Don't store or verify checksum
```

**unzip**

unzip [-opts[modifiers]] file[.zip] [list] [-x xlist] [-d exdir]

Extract files from ZIP archives

Options:

```
-l      List archive contents (with -q for short form)
-n      Never overwrite existing files (default)
-o      Overwrite files without prompting
-p      Send output to stdout
-q      Quiet
-x      Exclude these files
-d      Extract files into this directory
```

**uptime**

uptime

Display the time since the last boot

**usleep**

usleep N

Pause for N microseconds

**uudecode**

uudecode [-o outfile] [infile]

Uudecode a file Finds outfile name in uuencoded source unless -o is given

**uuencode**

uuencode [-m] [infile] stored_filename

Uuencode a file to stdout

Options:

```
-m      Use base64 encoding per RFC1521
```

**vconfig**

vconfig COMMAND [OPTIONS]

Create and remove virtual ethernet devices

Options:

```
add             [interface-name] [vlan_id]
rem             [vlan-name]
set_flag        [interface-name] [flag-num] [0 | 1]
set_egress_map  [vlan-name] [skb_priority] [vlan_qos]
set_ingress_map [vlan-name] [skb_priority] [vlan_qos]
set_name_type   [name-type]
```

**vi**

vi [OPTIONS] [FILE]...

Edit FILE

Options:

```
-c      Initial command to run ($EXINIT also available)
-R      Read-only - do not write to the file
-H      Short help regarding available features
```

**vlock**

vlock [OPTIONS]

Lock a virtual terminal. A password is required to unlock.

Options:

```
-a      Lock all VTs
```

**volname**

volname [DEVICE]

Show CD volume name of the DEVICE (default /dev/cdrom)

**watch**

watch [-n seconds] [-t] PROG [ARGS]

Run PROG periodically

Options:

```
-n      Loop period in seconds (default 2)
-t      Don't print header
```

**watchdog**

watchdog [-t N[ms]] [-T N[ms]] [-F] DEV

Periodically write to watchdog device DEV

Options:

```
-T N    Reboot after N seconds if not reset (default 60)
-t N    Reset every N seconds (default 30)
-F      Run in foreground
```

Use 500ms to specify period in milliseconds

**wc**

wc [OPTIONS] [FILE]...

Print line, word, and byte counts for each FILE, and a total line if more than one FILE is specified. With no FILE, read standard input.

Options:

```
-c      Print the byte counts
-l      Print the newline counts
-L      Print the length of the longest line
-w      Print the word counts
```

**wget**

wget [-c|--continue] [-s|--spider] [-q|--quiet] [-O|--output-document file] [--header 'header: value'] [-Y|--proxy on/off] [-P DIR] [-U|--user-agent agent] url

Retrieve files via HTTP or FTP

Options:

```
-s      Spider mode - only check file existence
-c      Continue retrieval of aborted transfer
-q      Quiet
-P      Set directory prefix to DIR
-O      Save to filename ('-' for stdout)
-U      Adjust 'User-Agent' field
-Y      Use proxy ('on' or 'off')
```

**which**

which [COMMAND]...

Locate a COMMAND

**who**

who [-a]

Show who is logged on

Options:

```
-a      show all
```

**whoami**

whoami

Print the user name associated with the current effective user id

**xargs**

xargs [OPTIONS] [PROG [ARGS]]

Run PROG on every item given by standard input

Options:

```
-p      Ask user whether to run each command
-r      Do not run command if input is empty
-0      Input is separated by NUL characters
-t      Print the command on stderr before execution
-e[STR] STR stops input processing
-n N    Pass no more than N args to PROG
-s N    Pass command line of no more than N bytes
-x      Exit if size is exceeded
```

**yes**

yes [OPTIONS] [STRING]

Repeatedly output a line with STRING, or 'y'

**zcat**

zcat FILE

Uncompress to stdout

**zcip**

zcip [OPTIONS] IFACE SCRIPT

Manage a ZeroConf IPv4 link-local address

Options:

```
-f              Run in foreground
-q              Quit after obtaining address
-r 169.254.x.x  Request this address first
-v              Verbose
```

With no -q, runs continuously monitoring for ARP conflicts, exits only on I/O errors (link down etc)

---

# LIBC NSS

GNU Libc (glibc) uses the Name Service Switch (NSS) to configure the behavior of the C library for the local environment, and to configure how it reads system data, such as passwords and group information. This is implemented using an /etc/nsswitch.conf configuration file, and using one or more of the /lib/libnss_* libraries. BusyBox tries to avoid using any libc calls that make use of NSS. Some applets however, such as login and su, will use libc functions that require NSS.

If you enable CONFIG_USE_BB_PWD_GRP, BusyBox will use internal functions to directly access the /etc/passwd, /etc/group, and /etc/shadow files without using NSS. This may allow you to run your system without the need for installing any of the NSS configuration files and libraries.

When used with glibc, the BusyBox 'networking' applets will similarly require that you install at least some of the glibc NSS stuff (in particular, /etc/nsswitch.conf, /lib/libnss_dns*, /lib/libnss_files*, and /lib/libresolv*).

Shameless Plug: As an alternative, one could use a C library such as uClibc. In addition to making your system significantly smaller, uClibc does not require the use of any NSS support files or libraries.

---

# MAINTAINER

Denis Vlasenko <vda.linux@googlemail.com>

---

# AUTHORS

The following people have contributed code to BusyBox whether they know it or not. If you have written code included in BusyBox, you should probably be listed here so you can obtain your bit of eternal glory. If you should be listed here, or the description of what you have done needs more detail, or is incorrect, please send in an update.

Emanuele Aina <emanuele.aina@tiscali.it> run-parts

Erik Andersen <andersen@codepoet.org>

```
    Tons of new stuff, major rewrite of most of the
    core apps, tons of new apps as noted in header files.
    Lots of tedious effort writing these boring docs that
    nobody is going to actually read.
```

Laurence Anderson <l.d.anderson@warwick.ac.uk>

```
    rpm2cpio, unzip, get_header_cpio, read_gz interface, rpm
```

Jeff Angielski <jeff@theptrgroup.com>

```
    ftpput, ftpget
```

Edward Betts <edward@debian.org>

```
    expr, hostid, logname, whoami
```

John Beppu <beppu@codepoet.org>

```
    du, nslookup, sort
```

Brian Candler <B.Candler@pobox.com>

```
tiny-ls(ls)
```

Randolph Chung <tausq@debian.org>

```
fbset, ping, hostname
```

Dave Cinege <dcinege@psychosis.com>

```
more(v2), makedevs, dutmp, modularization, auto links file,
various fixes, Linux Router Project maintenance
```

Jordan Crouse <jordan@cosmicpenguin.net>

```
ipcalc
```

Magnus Damm <damm@opensource.se>

```
tftp client insmod powerpc support
```

Larry Doolittle <ldoolitt@recycle.lbl.gov>

```
pristine source directory compilation, lots of patches and fixes.
```

Glenn Engel <glenne@engel.org>

```
httpd
```

Gennady Feldman <gfeldman@gena01.com>

```
Sysklogd (single threaded syslogd, IPC Circular buffer support,
logread), various fixes.
```

Karl M. Hegbloom <karlheg@debian.org>

```
cp_mv.c, the test suite, various fixes to utility.c, &c.
```

Daniel Jacobowitz <dan@debian.org>

```
mktemp.c
```

Matt Kraai <kraai@alumni.cmu.edu>

```
documentation, bugfixes, test suite
```

Stephan Linz <linz@li-pro.net>

```
ipcalc, Red Hat equivalence
```

John Lombardo <john@deltanet.com>

```
tr
```

Glenn McGrath <bug1@iinet.net.au>

```
Common unarchiving code and unarchiving applets, ifupdown, ftpgetput,
nameif, sed, patch, fold, install, uudecode.
Various bugfixes, review and apply numerous patches.
```

Manuel Novoa III <mjn3@codepoet.org>

```
cat, head, mkfifo, mknod, rmdir, sleep, tee, tty, uniq, usleep, wc, yes,
mesg, vconfig, make_directory, parse_mode, dirname, mode_string,
get_last_path_component, simplify_path, and a number trivial libbb routines

also bug fixes, partial rewrites, and size optimizations in
ash, basename, cal, cmp, cp, df, du, echo, env, ln, logname, md5sum, mkdir,
mv, realpath, rm, sort, tail, touch, uname, watch, arith, human_readable,
interface, dutmp, ifconfig, route
```

Vladimir Oleynik <dzo@simtreas.ru>

```
cmdedit; xargs(current), httpd(current);
ports: ash, crond, fdisk, inetd, stty, traceroute, top;
locale, various fixes
and irreconcilable critic of everything not perfect.
```

Bruce Perens <bruce@pixar.com>

```
Original author of BusyBox in 1995, 1996. Some of his code can
still be found hiding here and there...
```

Tim Riker <Tim@Rikers.org>

```
bug fixes, member of fan club
```

Kent Robotti <robotti@metconnect.com>

```
reset, tons and tons of bug reports and patches.
```

Chip Rosenthal <chip@unicom.com>, <crosenth@covad.com>

```
wget - Contributed by permission of Covad Communications
```

Pavel Roskin <proski@gnu.org>

```
    Lots of bugs fixes and patches.
```

Gyepi Sam <gyepi@praxis-sw.com>

```
    Remote logging feature for syslogd
```

Linus Torvalds <torvalds@transmeta.com>

```
    mkswap, fsck.minix, mkfs.minix
```

Mark Whitley <markw@codepoet.org>

```
    grep, sed, cut, xargs(previous),
    style-guide, new-applet-HOWTO, bug fixes, etc.
```

Charles P. Wright <cpwright@villagenet.com>

```
    gzip, mini-netcat(nc)
```

Enrique Zanardi <ezanardi@ull.es>

```
    tarcat (since removed), loadkmap, various fixes, Debian maintenance
```

Tito Ragusa <farmatito@tiscali.it>

```
    devfsd and size optimizations in strings, openvt and deallocvt.
```