NAME

ps − report a snapshot of the current processes.

SYNOPSIS

ps [*options*]

DESCRIPTION

ps displays information about a selection of the active processes. If you want a repetitive update of the selection and the displayed information, use **top** instead.

This version of **ps** accepts several kinds of options:

1 UNIX options, which may be grouped and must be preceded by a dash.
2 BSD options, which may be grouped and must not be used with a dash.
3 GNU long options, which are preceded by two dashes.

Options of different types may be freely mixed, but conflicts can appear. There are some synonymous options, which are functionally identical, due to the many standards and **ps** implementations that this **ps** is compatible with.

Note that **ps −aux** is distinct from **ps aux**. The POSIX and UNIX standards require that **ps −aux** print all processes owned by a user named *x*, as well as printing all processes that would be selected by the **−a** option. If the user named *x* does not exist, this **ps** may interpret the command as **ps aux** instead and print a warning. This behavior is intended to aid in transitioning old scripts and habits. It is fragile, subject to change, and thus should not be relied upon.

By default, **ps** selects all processes with the same effective user ID (euid=EUID) as the current user and associated with the same terminal as the invoker. It displays the process ID (pid=PID), the terminal associated with the process (tname=TTY), the cumulated CPU time in [DD−]hh:mm:ss format (time=TIME), and the executable name (ucmd=CMD). Output is unsorted by default.

The use of BSD−style options will add process state (stat=STAT) to the default display and show the command args (args=COMMAND) instead of the executable name. You can override this with the **PS_FORMAT** environment variable. The use of BSD−style options will also change the process selection to include processes on other terminals (TTYs) that are owned by you; alternately, this may be described as setting the selection to be the set of all processes filtered to exclude processes owned by other users or not on a terminal. These effects are not considered when options are described as being "identical" below, so **−M** will be considered identical to **Z** and so on.

Except as described below, process selection options are additive. The default selection is discarded, and then the selected processes are added to the set of processes to be displayed. A process will thus be shown if it meets any of the given selection criteria.

EXAMPLES

To see every process on the system using standard syntax:

ps −e
ps −ef
ps −eF
ps −ely

To see every process on the system using BSD syntax:

ps ax
ps axu

To print a process tree:

ps −ejH
ps axjf

To get info about threads:

ps −eLf
ps axms

To get security info:
ps −eo euser,ruser,suser,fuser,f,comm,label
ps axZ
ps −eM

To see every process running as root (real & effective ID) in user format:
ps −U root −u root u

To see every process with a user−defined format:
ps −eo pid,tid,class,rtprio,ni,pri,psr,pcpu,stat,wchan:14,comm
ps axo stat,euid,ruid,tty,tpgid,sess,pgrp,ppid,pid,pcpu,comm
ps −Ao pid,tt,user,fname,tmout,f,wchan

Print only the process IDs of syslogd:
ps −C syslogd −o pid=

Print only the name of PID 42:
ps −q 42 −o comm=

SIMPLE PROCESS SELECTION

a Lift the BSD−style "only yourself" restriction, which is imposed upon the set of all processes when some BSD−style (without "−") options are used or when the **ps** personality setting is BSD−like. The set of processes selected in this manner is in addition to the set of processes selected by other means. An alternate description is that this option causes **ps** to list all processes with a terminal (tty), or to list all processes when used together with the **x** option.

−A Select all processes. Identical to **−e**.

−a Select all processes except both session leaders (see *getsid*(2)) and processes not associated with a terminal.

−d Select all processes except session leaders.

−−deselect
Select all processes except those that fulfill the specified conditions (negates the selection). Identical to **−N**.

−e Select all processes. Identical to **−A**.

g Really all, even session leaders. This flag is obsolete and may be discontinued in a future release. It is normally implied by the **a** flag, and is only useful when operating in the sunos4 personality.

−N Select all processes except those that fulfill the specified conditions (negates the selection). Identical to **−−deselect**.

T Select all processes associated with this terminal. Identical to the **t** option without any argument.

r Restrict the selection to only running processes.

x Lift the BSD−style "must have a tty" restriction, which is imposed upon the set of all processes when some BSD−style (without "−") options are used or when the **ps** personality setting is BSD−like. The set of processes selected in this manner is in addition to the set of processes selected by other means. An alternate description is that this option causes **ps** to list all processes owned by you (same EUID as **ps**), or to list all processes when used together with the **a** option.

PROCESS SELECTION BY LIST

These options accept a single argument in the form of a blank−separated or comma−separated list. They can be used multiple times. For example: **ps −p "1 2" −p 3,4**

−123 Identical to **−−pid** *123*.

123 Identical to **−−pid** *123*.

−C *cmdlist*
Select by command name. This selects the processes whose executable name is given in *cmdlist*. NOTE: The command name is not the same as the command line. Previous versions of procps and

the kernel truncated this command name to 15 characters. This limitation is no longer present in both. If you depended on matching only 15 characters, you may no longer get a match.

−G *grplist*
Select by real group ID (RGID) or name. This selects the processes whose real group name or ID is in the *grplist* list. The real group ID identifies the group of the user who created the process, see *getgid*(2).

−g *grplist*
Select by session OR by effective group name. Selection by session is specified by many standards, but selection by effective group is the logical behavior that several other operating systems use. This **ps** will select by session when the list is completely numeric (as sessions are). Group ID numbers will work only when some group names are also specified. See the **−s** and **−−group** options.

−−Group *grplist*
Select by real group ID (RGID) or name. Identical to **−G**.

−−group *grplist*
Select by effective group ID (EGID) or name. This selects the processes whose effective group name or ID is in *grplist*. The effective group ID describes the group whose file access permissions are used by the process (see *getegid*(2)). The **−g** option is often an alternative to **−−group**.

p *pidlist*
Select by process ID. Identical to **−p** and **−−pid**.

−p *pidlist*
Select by PID. This selects the processes whose process ID numbers appear in *pidlist*. Identical to **p** and **−−pid**.

−−pid *pidlist*
Select by process ID. Identical to **−p** and **p**.

−−ppid *pidlist*
Select by parent process ID. This selects the processes with a parent process ID in *pidlist*. That is, it selects processes that are children of those listed in *pidlist*.

q *pidlist*
Select by process ID (quick mode). Identical to **−q** and **−−quick−pid**.

−q *pidlist*
Select by PID (quick mode). This selects the processes whose process ID numbers appear in *pidlist*. With this option **ps** reads the necessary info only for the pids listed in the *pidlist* and doesn't apply additional filtering rules. The order of pids is unsorted and preserved. No additional selection options, sorting and forest type listings are allowed in this mode. Identical to **q** and **−−quick−pid**.

−−quick−pid *pidlist*
Select by process ID (quick mode). Identical to **−q** and **q**.

−s *sesslist*
Select by session ID. This selects the processes with a session ID specified in *sesslist*.

−−sid *sesslist*
Select by session ID. Identical to **−s**.

t *ttylist* Select by tty. Nearly identical to **−t** and **−−tty**, but can also be used with an empty *ttylist* to indicate the terminal associated with **ps**. Using the **T** option is considered cleaner than using **t** with an empty *ttylist*.

−t *ttylist*
Select by tty. This selects the processes associated with the terminals given in *ttylist*. Terminals (ttys, or screens for text output) can be specified in several forms: /dev/ttyS1, ttyS1, S1. A plain

"−" may be used to select processes not attached to any terminal.

−−tty *ttylist*
Select by terminal. Identical to **−t** and **t**.

U *userlist*
Select by effective user ID (EUID) or name. This selects the processes whose effective user name or ID is in *userlist*. The effective user ID describes the user whose file access permissions are used by the process (see *geteuid*(2)). Identical to **−u** and **−−user**.

−U *userlist*
Select by real user ID (RUID) or name. It selects the processes whose real user name or ID is in the *userlist* list. The real user ID identifies the user who created the process, see *getuid*(2).

−u *userlist*
Select by effective user ID (EUID) or name. This selects the processes whose effective user name or ID is in *userlist*.

The effective user ID describes the user whose file access permissions are used by the process (see *geteuid*(2)). Identical to **U** and **−−user**.

−−User *userlist*
Select by real user ID (RUID) or name. Identical to **−U**.

−−user *userlist*
Select by effective user ID (EUID) or name. Identical to **−u** and **U**.

OUTPUT FORMAT CONTROL

These options are used to choose the information displayed by **ps**. The output may differ by personality.

−c Show different scheduler information for the **−l** option.

−−context
Display security context format (for SELinux).

−f Do full−format listing. This option can be combined with many other UNIX−style options to add additional columns. It also causes the command arguments to be printed. When used with **−L**, the NLWP (number of threads) and LWP (thread ID) columns will be added. See the **c** option, the format keyword **args**, and the format keyword **comm**.

−F Extra full format. See the **−f** option, which **−F** implies.

−−format *format*
user−defined format. Identical to **−o** and **o**.

j BSD job control format.

−j Jobs format.

l Display BSD long format.

−l Long format. The **−y** option is often useful with this.

−M Add a column of security data. Identical to **Z** (for SELinux).

O *format*
is preloaded **o** (overloaded). The BSD **O** option can act like **−O** (user−defined output format with some common fields predefined) or can be used to specify sort order. Heuristics are used to determine the behavior of this option. To ensure that the desired behavior is obtained (sorting or formatting), specify the option in some other way (e.g. with **−O** or **−−sort**). When used as a formatting option, it is identical to **−O**, with the BSD personality.

−O *format*
Like **−o**, but preloaded with some default columns. Identical to **−o pid,***format***,state,tname,time, command** or **−o pid,***format***,tname,time,cmd**, see **−o** below.

o *format*
Specify user−defined format. Identical to **−o** and **−−format**.

−o *format*
User−defined format. *format* is a single argument in the form of a blank−separated or comma−separated list, which offers a way to specify individual output columns. The recognized keywords are described in the **STANDARD FORMAT SPECIFIERS** section below. Headers may be renamed (**ps −o pid,ruser=RealUser −o comm=Command**) as desired. If all column headers are empty (**ps −o pid= −o comm=**) then the header line will not be output. Column width will increase as needed for wide headers; this may be used to widen up columns such as WCHAN (**ps −o pid,wchan=WIDE−WCHAN−COLUMN −o comm**). Explicit width control (**ps opid, wchan:42,cmd**) is offered too. The behavior of **ps −o pid=X,comm=Y** varies with personality; output may be one column named "X,comm=Y" or two columns named "X" and "Y". Use multiple **−o** options when in doubt. Use the **PS_FORMAT** environment variable to specify a default as desired; DefSysV and DefBSD are macros that may be used to choose the default UNIX or BSD columns.

s Display signal format.

u Display user−oriented format.

v Display virtual memory format.

X Register format.

−y Do not show flags; show rss in place of addr. This option can only be used with **−l**.

Z Add a column of security data. Identical to **−M** (for SELinux).

OUTPUT MODIFIERS

c Show the true command name. This is derived from the name of the executable file, rather than from the argv value. Command arguments and any modifications to them are thus not shown. This option effectively turns the **args** format keyword into the **comm** format keyword; it is useful with the **−f** format option and with the various BSD−style format options, which all normally display the command arguments. See the **−f** option, the format keyword **args**, and the format keyword **comm**.

−−cols *n*
Set screen width.

−−columns *n*
Set screen width.

−−cumulative
Include some dead child process data (as a sum with the parent).

e Show the environment after the command.

f ASCII art process hierarchy (forest).

−−forest
ASCII art process tree.

h No header. (or, one header per screen in the BSD personality). The **h** option is problematic. Standard BSD **ps** uses this option to print a header on each page of output, but older Linux **ps** uses this option to totally disable the header. This version of **ps** follows the Linux usage of not printing the header unless the BSD personality has been selected, in which case it prints a header on each page of output. Regardless of the current personality, you can use the long options **−−headers** and **−−no−headers** to enable printing headers each page or disable headers entirely, respectively.

−H Show process hierarchy (forest).

−−headers
Repeat header lines, one per page of output.

k *spec* Specify sorting order. Sorting syntax is [**+**|**−**]*key*[,[**+**|**−**]*key*[,...]]. Choose a multi−letter key from the **STANDARD FORMAT SPECIFIERS** section. The "+" is optional since default direction is increasing numerical or lexicographic order. Identical to **−−sort**.

Examples:
ps jaxkuid,−ppid,+pid
ps axk comm o comm,args
ps kstart_time −ef

−−lines *n*
Set screen height.

n Numeric output for WCHAN and USER (including all types of UID and GID).

−−no−headers
Print no header line at all. **−−no−heading** is an alias for this option.

O *order*
Sorting order (overloaded). The BSD **O** option can act like **−O** (user−defined output format with some common fields predefined) or can be used to specify sort order. Heuristics are used to determine the behavior of this option. To ensure that the desired behavior is obtained (sorting or formatting), specify the option in some other way (e.g. with **−O** or **−−sort**).

For sorting, obsolete BSD **O** option syntax is **O**[**+**|**−**]*k1*[,[**+**|**−**]*k2*[,...]]. It orders the processes listing according to the multilevel sort specified by the sequence of one−letter short keys *k1*,*k2*, ... described in the **OBSOLETE SORT KEYS** section below. The "+" is currently optional, merely re−iterating the default direction on a key, but may help to distinguish an **O** sort from an **O** format. The "−" reverses direction only on the key it precedes.

−−rows *n*
Set screen height.

S Sum up some information, such as CPU usage, from dead child processes into their parent. This is useful for examining a system where a parent process repeatedly forks off short−lived children to do work.

−−sort *spec*
Specify sorting order. Sorting syntax is [+|−]*key*[,[**+**|**−**]*key*[,...]]. Choose a multi−letter key from the **STANDARD FORMAT SPECIFIERS** section. The "+" is optional since default direction is increasing numerical or lexicographic order. Identical to **k**. For example: **ps jax −−sort=uid,−ppid,+pid**

w Wide output. Use this option twice for unlimited width.

−w Wide output. Use this option twice for unlimited width.

−−width *n*
Set screen width.

THREAD DISPLAY

H Show threads as if they were processes.

−L Show threads, possibly with LWP and NLWP columns.

m Show threads after processes.

−m Show threads after processes.

−T Show threads, possibly with SPID column.

OTHER INFORMATION

−−help *section*
Print a help message. The *section* argument can be one of *s*imple, *l*ist, *o*utput, *t*hreads, *m*isc, or *a*ll. The argument can be shortened to one of the underlined letters as in: s|l|o|t|m|a.

−−info Print debugging info.

L List all format specifiers.

V Print the procps-ng version.

−V Print the procps-ng version.

−−version
Print the procps-ng version.

NOTES

This **ps** works by reading the virtual files in /proc. This **ps** does not need to be setuid kmem or have any privileges to run. Do not give this **ps** any special permissions.

CPU usage is currently expressed as the percentage of time spent running during the entire lifetime of a process. This is not ideal, and it does not conform to the standards that **ps** otherwise conforms to. CPU usage is unlikely to add up to exactly 100%.

The SIZE and RSS fields don't count some parts of a process including the page tables, kernel stack, struct thread_info, and struct task_struct. This is usually at least 20 KiB of memory that is always resident. SIZE is the virtual size of the process (code+data+stack).

Processes marked <defunct> are dead processes (so−called "zombies") that remain because their parent has not destroyed them properly. These processes will be destroyed by *init*(8) if the parent process exits.

If the length of the username is greater than the length of the display column, the username will be truncated. See the **−o** and **−O** formatting options to customize length.

Commands options such as **ps −aux** are not recommended as it is a confusion of two different standards. According to the POSIX and UNIX standards, the above command asks to display all processes with a TTY (generally the commands users are running) plus all processes owned by a user named *x*. If that user doesn't exist, then **ps** will assume you really meant **ps aux**.

PROCESS FLAGS

The sum of these values is displayed in the "F" column, which is provided by the **flags** output specifier:

1 forked but didn't exec
4 used super−user privileges

PROCESS STATE CODES

Here are the different values that the **s**, **stat** and **state** output specifiers (header "STAT" or "S") will display to describe the state of a process:

D uninterruptible sleep (usually IO)
I Idle kernel thread
R running or runnable (on run queue)
S interruptible sleep (waiting for an event to complete)
T stopped by job control signal
t stopped by debugger during the tracing
W paging (not valid since the 2.6.xx kernel)
X dead (should never be seen)
Z defunct ("zombie") process, terminated but not reaped by its parent

For BSD formats and when the **stat** keyword is used, additional characters may be displayed:

< high−priority (not nice to other users)
N low−priority (nice to other users)
L has pages locked into memory (for real−time and custom IO)
s is a session leader
l is multi-threaded (using CLONE_THREAD, like NPTL pthreads do)
+ is in the foreground process group

OBSOLETE SORT KEYS

These keys are used by the BSD **O** option (when it is used for sorting). The GNU **−−sort** option doesn't use these keys, but the specifiers described below in the **STANDARD FORMAT SPECIFIERS** section. Note that the values used in sorting are the internal values **ps** uses and not the "cooked" values used in some of the output format fields (e.g. sorting on tty will sort into device number, not according to the terminal name displayed). Pipe **ps** output into the **sort**(1) command if you want to sort the cooked values.

KEY	LONG	DESCRIPTION
c	cmd	simple name of executable
C	pcpu	cpu utilization
f	flags	flags as in long format F field
g	pgrp	process group ID
G	tpgid	controlling tty process group ID
j	cutime	cumulative user time
J	cstime	cumulative system time
k	utime	user time
m	min_flt	number of minor page faults
M	maj_flt	number of major page faults
n	cmin_flt	cumulative minor page faults
N	cmaj_flt	cumulative major page faults
o	session	session ID
p	pid	process ID
P	ppid	parent process ID
r	rss	resident set size
R	resident	resident pages
s	size	memory size in kilobytes
S	share	amount of shared pages
t	tty	the device number of the controlling tty
T	start_time	time process was started
U	uid	user ID number
u	user	user name
v	vsize	total VM size in KiB
y	priority	kernel scheduling priority

AIX FORMAT DESCRIPTORS

This **ps** supports AIX format descriptors, which work somewhat like the formatting codes of *printf*(1) and *printf*(3). For example, the normal default output can be produced with this: **ps −eo "%p %y %x %c"**. The **NORMAL** codes are described in the next section.

CODE	NORMAL	HEADER
%C	pcpu	%CPU
%G	group	GROUP
%P	ppid	PPID
%U	user	USER
%a	args	COMMAND
%c	comm	COMMAND
%g	rgroup	RGROUP
%n	nice	NI
%p	pid	PID
%r	pgid	PGID
%t	etime	ELAPSED
%u	ruser	RUSER
%x	time	TIME
%y	tty	TTY
%z	vsz	VSZ

STANDARD FORMAT SPECIFIERS

Here are the different keywords that may be used to control the output format (e.g., with option **−o**) or to sort the selected processes with the GNU−style **−−sort** option.

For example: **ps −eo pid,user,args −−sort user**

This version of **ps** tries to recognize most of the keywords used in other implementations of **ps**.

The following user−defined format specifiers may contain spaces: **args**, **cmd**, **comm**, **command**, **fname**, **ucmd**, **ucomm**, **lstart**, **bsdstart**, **start**.

Some keywords may not be available for sorting.

CODE	HEADER	DESCRIPTION
%cpu	%CPU	cpu utilization of the process in "##.#" format. Currently, it is the CPU time used divided by the time the process has been running (cputime/realtime ratio), expressed as a percentage. It will not add up to 100% unless you are lucky. (alias **pcpu**).
%mem	%MEM	ratio of the process's resident set size to the physical memory on the machine, expressed as a percentage. (alias **pmem**).
args	COMMAND	command with all its arguments as a string. Modifications to the arguments may be shown. The output in this column may contain spaces. A process marked <defunct> is partly dead, waiting to be fully destroyed by its parent. Sometimes the process args will be unavailable; when this happens, **ps** will instead print the executable name in brackets. (alias **cmd**, **command**). See also the **comm** format keyword, the **−f** option, and the **c** option. When specified last, this column will extend to the edge of the display. If **ps** can not determine display width, as when output is redirected (piped) into a file or another command, the output width is undefined (it may be 80, unlimited, determined by the **TERM** variable, and so on). The **COLUMNS** environment variable or **−−cols** option may be used to exactly determine the width in this case. The **w** or **−w** option may be also be used to adjust width.
blocked	BLOCKED	mask of the blocked signals, see *signal*(7). According to the width of the field, a 32 or 64−bit mask in hexadecimal format is displayed. (alias **sig_block**, **sigmask**).
bsdstart	START	time the command started. If the process was started less than 24 hours ago, the output format is " HH:MM", else it is " Mmm:SS" (where Mmm is the three letters of the month). See also **lstart**, **start**, **start_time**, and **stime**.
bsdtime	TIME	accumulated cpu time, user + system. The display format is usually "MMM:SS", but can be shifted to the right if the process used more than 999 minutes of cpu time.
c	C	processor utilization. Currently, this is the integer value of the percent usage over the lifetime of the process. (see **%cpu**).
caught	CAUGHT	mask of the caught signals, see *signal*(7). According to the width of the field, a 32 or 64 bits mask in hexadecimal format is displayed. (alias **sig_catch**, **sigcatch**).

cgname CGNAME display name of control groups to which the process belongs.

cgroup CGROUP display control groups to which the process belongs.

class CLS scheduling class of the process. (alias **policy**, **cls**). Field's possible values are:

–	not reported
TS	SCHED_OTHER
FF	SCHED_FIFO
RR	SCHED_RR
B	SCHED_BATCH
ISO	SCHED_ISO
IDL	SCHED_IDLE
DLN	SCHED_DEADLINE
?	unknown value

cls CLS scheduling class of the process. (alias **policy**, **cls**). Field's possible values are:

–	not reported
TS	SCHED_OTHER
FF	SCHED_FIFO
RR	SCHED_RR
B	SCHED_BATCH
ISO	SCHED_ISO
IDL	SCHED_IDLE
DLN	SCHED_DEADLINE
?	unknown value

cmd CMD see **args**. (alias **args**, **command**).

comm COMMAND command name (only the executable name). Modifications to the command name will not be shown. A process marked <defunct> is partly dead, waiting to be fully destroyed by its parent. The output in this column may contain spaces. (alias **ucmd**, **ucomm**). See also the **args** format keyword, the **–f** option, and the **c** option.
When specified last, this column will extend to the edge of the display. If **ps** can not determine display width, as when output is redirected (piped) into a file or another command, the output width is undefined (it may be 80, unlimited, determined by the **TERM** variable, and so on). The **COLUMNS** environment variable or **––cols** option may be used to exactly determine the width in this case. The **w** or **–w** option may be also be used to adjust width.

command COMMAND See **args**. (alias **args**, **command**).

cp CP per–mill (tenths of a percent) CPU usage. (see **%cpu**).

cputime TIME cumulative CPU time, "[DD–]hh:mm:ss" format. (alias **time**).

cputimes TIME cumulative CPU time in seconds (alias **times**).

drs DRS data resident set size, the amount of physical memory devoted to other than executable code.

egid	EGID	effective group ID number of the process as a decimal integer. (alias **gid**).
egroup	EGROUP	effective group ID of the process. This will be the textual group ID, if it can be obtained and the field width permits, or a decimal representation otherwise. (alias **group**).
eip	EIP	instruction pointer.
esp	ESP	stack pointer.
etime	ELAPSED	elapsed time since the process was started, in the form [[DD−]hh:]mm:ss.
etimes	ELAPSED	elapsed time since the process was started, in seconds.
euid	EUID	effective user ID (alias **uid**).
euser	EUSER	effective user name. This will be the textual user ID, if it can be obtained and the field width permits, or a decimal representation otherwise. The **n** option can be used to force the decimal representation. (alias **uname**, **user**).
exe	EXE	path to the executable. Useful if path cannot be printed via **cmd**, **comm** or **args** format options.
f	F	flags associated with the process, see the **PROCESS FLAGS** section. (alias **flag**, **flags**).
fgid	FGID	filesystem access group ID. (alias **fsgid**).
fgroup	FGROUP	filesystem access group ID. This will be the textual group ID, if it can be obtained and the field width permits, or a decimal representation otherwise. (alias **fsgroup**).
flag	F	see **f**. (alias **f**, **flags**).
flags	F	see **f**. (alias **f**, **flag**).
fname	COMMAND	first 8 bytes of the base name of the process's executable file. The output in this column may contain spaces.
fuid	FUID	filesystem access user ID. (alias **fsuid**).
fuser	FUSER	filesystem access user ID. This will be the textual user ID, if it can be obtained and the field width permits, or a decimal representation otherwise.
gid	GID	see **egid**. (alias **egid**).
group	GROUP	see **egroup**. (alias **egroup**).
ignored	IGNORED	mask of the ignored signals, see *signal*(7). According to the width of the field, a 32 or 64 bits mask in hexadecimal format is displayed. (alias **sig_ignore**, **sigignore**).

ipcns	IPCNS	Unique inode number describing the namespace the process belongs to. See *namespaces*(7).
label	LABEL	security label, most commonly used for SELinux context data. This is for the *Mandatory Access Control* ("MAC") found on high−security systems.
lstart	STARTED	time the command started. See also **bsdstart**, **start**, **start_time**, and **stime**.
lsession	SESSION	displays the login session identifier of a process, if systemd support has been included.
luid	LUID	displays Login ID associated with a process.
lwp	LWP	light weight process (thread) ID of the dispatchable entity (alias **spid**, **tid**). See **tid** for additional information.
lxc	LXC	The name of the lxc container within which a task is running. If a process is not running inside a container, a dash ('−') will be shown.
machine	MACHINE	displays the machine name for processes assigned to VM or container, if systemd support has been included.
maj_flt	MAJFLT	The number of major page faults that have occurred with this process.
min_flt	MINFLT	The number of minor page faults that have occurred with this process.
mntns	MNTNS	Unique inode number describing the namespace the process belongs to. See *namespaces*(7).
netns	NETNS	Unique inode number describing the namespace the process belongs to. See *namespaces*(7).
ni	NI	nice value. This ranges from 19 (nicest) to −20 (not nice to others), see *nice*(1). (alias **nice**).
nice	NI	see **ni**.(**alias ni**).
nlwp	NLWP	number of lwps (threads) in the process. (alias **thcount**).
numa	NUMA	The node associated with the most recently used processor. A *−1* means that NUMA information is unavailable.
nwchan	WCHAN	address of the kernel function where the process is sleeping (use **wchan** if you want the kernel function name). Running tasks will display a dash ('−') in this column.
ouid	OWNER	displays the Unix user identifier of the owner of the session of a process, if systemd support has been included.
pcpu	%CPU	see **%cpu**. (alias **%cpu**).

pending PENDING mask of the pending signals. See *signal*(7). Signals pending on the process are distinct from signals pending on individual threads. Use the **m** option or the **−m** option to see both. According to the width of the field, a 32 or 64 bits mask in hexadecimal format is displayed. (alias **sig**).

pgid PGID process group ID or, equivalently, the process ID of the process group leader. (alias **pgrp**).

pgrp PGRP see **pgid**. (alias **pgid**).

pid PID a number representing the process ID (alias **tgid**).

pidns PIDNS Unique inode number describing the namespace the process belongs to. See *namespaces*(7).

pmem %MEM see **%mem**. (alias **%mem**).

policy POL scheduling class of the process. (alias **class**, **cls**). Possible values are:

−	not reported
TS	SCHED_OTHER
FF	SCHED_FIFO
RR	SCHED_RR
B	SCHED_BATCH
ISO	SCHED_ISO
IDL	SCHED_IDLE
DLN	SCHED_DEADLINE
?	unknown value

ppid PPID parent process ID.

pri PRI priority of the process. Higher number means lower priority.

psr PSR processor that process is currently assigned to.

rgid RGID real group ID.

rgroup RGROUP real group name. This will be the textual group ID, if it can be obtained and the field width permits, or a decimal representation otherwise.

rss RSS resident set size, the non−swapped physical memory that a task has used (in kilobytes). (alias **rssize**, **rsz**).

rssize RSS see **rss**. (alias **rss**, **rsz**).

rsz RSZ see **rss**. (alias **rss**, **rssize**).

rtprio RTPRIO realtime priority.

ruid RUID real user ID.

ruser RUSER real user ID. This will be the textual user ID, if it can be obtained and the field width permits, or a decimal representation otherwise.

s	S	minimal state display (one character). See section **PROCESS STATE CODES** for the different values. See also **stat** if you want additional information displayed. (alias **state**).
sched	SCH	scheduling policy of the process. The policies SCHED_OTHER (SCHED_NORMAL), SCHED_FIFO, SCHED_RR, SCHED_BATCH, SCHED_ISO, SCHED_IDLE and SCHED_DEADLINE are respectively displayed as 0, 1, 2, 3, 4, 5 and 6.
seat	SEAT	displays the identifier associated with all hardware devices assigned to a specific workplace, if systemd support has been included.
sess	SESS	session ID or, equivalently, the process ID of the session leader. (alias **session**, **sid**).
sgi_p	P	processor that the process is currently executing on. Displays "*" if the process is not currently running or runnable.
sgid	SGID	saved group ID. (alias **svgid**).
sgroup	SGROUP	saved group name. This will be the textual group ID, if it can be obtained and the field width permits, or a decimal representation otherwise.
sid	SID	see **sess**. (alias **sess**, **session**).
sig	PENDING	see **pending**. (alias **pending**, **sig_pend**).
sigcatch	CAUGHT	see **caught**. (alias **caught**, **sig_catch**).
sigignore	IGNORED	see **ignored**. (alias **ignored**, **sig_ignore**).
sigmask	BLOCKED	see **blocked**. (alias **blocked**, **sig_block**).
size	SIZE	approximate amount of swap space that would be required if the process were to dirty all writable pages and then be swapped out. This number is very rough!
slice	SLICE	displays the slice unit which a process belongs to, if systemd support has been included.
spid	SPID	see **lwp**. (alias **lwp**, **tid**).
stackp	STACKP	address of the bottom (start) of stack for the process.
start	STARTED	time the command started. If the process was started less than 24 hours ago, the output format is "HH:MM:SS", else it is " Mmm dd" (where Mmm is a three−letter month name). See also **lstart**, **bsdstart**, **start_time**, and **stime**.
start_time	START	starting time or date of the process. Only the year will be displayed if the process was not started the same year **ps** was invoked, or "MmmDD" if it was not started the same day, or "HH:MM" otherwise. See also **bsdstart**, **start**, **lstart**, and **stime**.

stat	STAT	multi−character process state. See section **PROCESS STATE CODES** for the different values meaning. See also **s** and **state** if you just want the first character displayed.
state	S	see **s**. (alias **s**).
stime	STIME	see **start_time**. (alias **start_time**).
suid	SUID	saved user ID. (alias **svuid**).
supgid	SUPGID	group ids of supplementary groups, if any. See **getgroups**(2).
supgrp	SUPGRP	group names of supplementary groups, if any. See **getgroups**(2).
suser	SUSER	saved user name. This will be the textual user ID, if it can be obtained and the field width permits, or a decimal representation otherwise. (alias **svuser**).
svgid	SVGID	see **sgid**. (alias **sgid**).
svuid	SVUID	see **suid**. (alias **suid**).
sz	SZ	size in physical pages of the core image of the process. This includes text, data, and stack space. Device mappings are currently excluded; this is subject to change. See **vsz** and **rss**.
tgid	TGID	a number representing the thread group to which a task belongs (alias **pid**). It is the process ID of the thread group leader.
thcount	THCNT	see **nlwp**. (alias **nlwp**). number of kernel threads owned by the process.
tid	TID	the unique number representing a dispatchable entity (alias **lwp**, **spid**). This value may also appear as: a process ID (pid); a process group ID (pgrp); a session ID for the session leader (sid); a thread group ID for the thread group leader (tgid); and a tty process group ID for the process group leader (tpgid).
time	TIME	cumulative CPU time, "[DD−]HH:MM:SS" format. (alias **cputime**).
times	TIME	cumulative CPU time in seconds (alias **cputimes**).
tname	TTY	controlling tty (terminal). (alias **tt**, **tty**).
tpgid	TPGID	ID of the foreground process group on the tty (terminal) that the process is connected to, or −1 if the process is not connected to a tty.
trs	TRS	text resident set size, the amount of physical memory devoted to executable code.
tt	TT	controlling tty (terminal). (alias **tname**, **tty**).
tty	TT	controlling tty (terminal). (alias **tname**, **tt**).
ucmd	CMD	see **comm**. (alias **comm**, **ucomm**).

ucomm	COMMAND	see **comm**. (alias **comm**, **ucmd**).
uid	UID	see **euid**. (alias **euid**).
uname	USER	see **euser**. (alias **euser**, **user**).
unit	UNIT	displays unit which a process belongs to, if systemd support has been included.
user	USER	see **euser**. (alias **euser**, **uname**).
userns	USERNS	Unique inode number describing the namespace the process belongs to. See *namespaces*(7).
utsns	UTSNS	Unique inode number describing the namespace the process belongs to. See *namespaces*(7).
uunit	UUNIT	displays user unit which a process belongs to, if systemd support has been included.
vsize	VSZ	see **vsz**. (alias **vsz**).
vsz	VSZ	virtual memory size of the process in KiB (1024−byte units). Device mappings are currently excluded; this is subject to change. (alias **vsize**).
wchan	WCHAN	name of the kernel function in which the process is sleeping, a "−" if the process is running, or a "*" if the process is multi−threaded and **ps** is not displaying threads.

ENVIRONMENT VARIABLES

The following environment variables could affect **ps**:

COLUMNS
Override default display width.

LINES
Override default display height.

PS_PERSONALITY
Set to one of posix, old, linux, bsd, sun, digital... (see section **PERSONALITY** below).

CMD_ENV
Set to one of posix, old, linux, bsd, sun, digital... (see section **PERSONALITY** below).

I_WANT_A_BROKEN_PS
Force obsolete command line interpretation.

LC_TIME
Date format.

PS_COLORS
Not currently supported.

PS_FORMAT
Default output format override. You may set this to a format string of the type used for the **−o** option. The **DefSysV** and **DefBSD** values are particularly useful.

POSIXLY_CORRECT
Don't find excuses to ignore bad "features".

POSIX2
When set to "on", acts as **POSIXLY_CORRECT**.

UNIX95
Don't find excuses to ignore bad "features".

_XPG
Cancel **CMD_ENV**=*irix* non−standard behavior.

In general, it is a bad idea to set these variables. The one exception is **CMD_ENV** or **PS_PERSONALITY**, which could be set to Linux for normal systems. Without that setting, **ps** follows the useless and bad parts of the Unix98 standard.

PERSONALITY

390	like the OS/390 OpenEdition **ps**
aix	like AIX **ps**
bsd	like FreeBSD **ps** (totally non−standard)
compaq	like Digital Unix **ps**
debian	like the old Debian **ps**
digital	like Tru64 (was Digital Unix, was OSF/1) **ps**
gnu	like the old Debian **ps**
hp	like HP−UX **ps**
hpux	like HP−UX **ps**
irix	like Irix **ps**
linux	***** **recommended** *****
old	like the original Linux **ps** (totally non−standard)
os390	like OS/390 Open Edition **ps**
posix	standard
s390	like OS/390 Open Edition **ps**
sco	like SCO **ps**
sgi	like Irix **ps**
solaris2	like Solaris 2+ (SunOS 5) **ps**
sunos4	like SunOS 4 (Solaris 1) **ps** (totally non−standard)
svr4	standard
sysv	standard
tru64	like Tru64 (was Digital Unix, was OSF/1) **ps**
unix	standard
unix95	standard
unix98	standard

SEE ALSO

pgrep(1), **pstree**(1), **top**(1), **proc**(5).

STANDARDS

This **ps** conforms to:

1 Version 2 of the Single Unix Specification
2 The Open Group Technical Standard Base Specifications, Issue 6
3 IEEE Std 1003.1, 2004 Edition
4 X/Open System Interfaces Extension [UP XSI]
5 ISO/IEC 9945:2003

AUTHOR

ps was originally written by Branko Lankester ⟨lankeste@fwi.uva.nl⟩. Michael K. Johnson ⟨johnsonm@redhat.com⟩ re−wrote it significantly to use the proc filesystem, changing a few things in the process. Michael Shields ⟨mjshield@nyx.cs.du.edu⟩ added the pid−list feature. Charles Blake ⟨cblake@bbn.com⟩ added multi−level sorting, the dirent−style library, the device name−to−number mmaped database, the

approximate binary search directly on System.map, and many code and documentation cleanups. David Mossberger−Tang wrote the generic BFD support for psupdate. Albert Cahalan ⟨albert@users.sf.net⟩ rewrote ps for full Unix98 and BSD support, along with some ugly hacks for obsolete and foreign syntax.

Please send bug reports to ⟨procps@freelists.org⟩. No subscription is required or suggested.

NAME

man − an interface to the system reference manuals

SYNOPSIS

man [*man options*] [[*section*] *page* ...] ...
man −k [*apropos options*] *regexp* ...
man −K [*man options*] [*section*] *term* ...
man −f [*whatis options*] *page* ...
man −l [*man options*] *file* ...
man −w|−W [*man options*] *page* ...

DESCRIPTION

man is the system's manual pager. Each *page* argument given to **man** is normally the name of a program, utility or function. The *manual page* associated with each of these arguments is then found and displayed. A *section*, if provided, will direct **man** to look only in that *section* of the manual. The default action is to search in all of the available *sections* following a pre-defined order (see **DEFAULTS**), and to show only the first *page* found, even if *page* exists in several *sections*.

The table below shows the *section* numbers of the manual followed by the types of pages they contain.

1 Executable programs or shell commands
2 System calls (functions provided by the kernel)
3 Library calls (functions within program libraries)
4 Special files (usually found in */dev*)
5 File formats and conventions, e.g. */etc/passwd*
6 Games
7 Miscellaneous (including macro packages and conventions), e.g. **man**(7), **groff**(7)
8 System administration commands (usually only for root)
9 Kernel routines [Non standard]

A manual *page* consists of several sections.

Conventional section names include **NAME**, **SYNOPSIS**, **CONFIGURATION**, **DESCRIPTION**, **OPTIONS**, **EXIT STATUS**, **RETURN VALUE**, **ERRORS**, **ENVIRONMENT**, **FILES**, **VERSIONS**, **CONFORMING TO**, **NOTES**, **BUGS**, **EXAMPLE**, **AUTHORS**, and **SEE ALSO**.

The following conventions apply to the **SYNOPSIS** section and can be used as a guide in other sections.

bold text	type exactly as shown.
italic text	replace with appropriate argument.
[**−abc**]	any or all arguments within [ ] are optional.
−a|−b	options delimited by | cannot be used together.
argument ...	*argument* is repeatable.
[*expression*] ...	entire *expression* within [ ] is repeatable.

Exact rendering may vary depending on the output device. For instance, man will usually not be able to render italics when running in a terminal, and will typically use underlined or coloured text instead.

The command or function illustration is a pattern that should match all possible invocations. In some cases it is advisable to illustrate several exclusive invocations as is shown in the **SYNOPSIS** section of this manual page.

EXAMPLES

man *ls*
Display the manual page for the *item* (program) *ls*.

man *man.7*
Display the manual page for macro package *man* from section *7*. (This is an alternative spelling of "**man** *7 man*".)

man '*man*(*7*)'
Display the manual page for macro package *man* from section *7*. (This is another alternative spelling of "**man** *7 man*". It may be more convenient when copying and pasting cross-references to manual pages. Note that the parentheses must normally be quoted to protect them from the shell.)

man −a *intro*
Display, in succession, all of the available *intro* manual pages contained within the manual. It is possible to quit between successive displays or skip any of them.

man −t *bash* | *lpr −Pps*
Format the manual page for *bash* into the default **troff** or **groff** format and pipe it to the printer named *ps*. The default output for **groff** is usually PostScript. **man −−help** should advise as to which processor is bound to the **−t** option.

man −l −T*dvi ./foo.1x.gz* **>** *./foo.1x.dvi*
This command will decompress and format the nroff source manual page *./foo.1x.gz* into a **device independent (dvi)** file. The redirection is necessary as the **−T** flag causes output to be directed to **stdout** with no pager. The output could be viewed with a program such as **xdvi** or further processed into PostScript using a program such as **dvips**.

man −k *printf*
Search the short descriptions and manual page names for the keyword *printf* as regular expression. Print out any matches. Equivalent to **apropos** *printf* **.**

man −f *smail*
Lookup the manual pages referenced by *smail* and print out the short descriptions of any found. Equivalent to **whatis** *smail***.**

OVERVIEW

Many options are available to **man** in order to give as much flexibility as possible to the user. Changes can be made to the search path, section order, output processor, and other behaviours and operations detailed below.

If set, various environment variables are interrogated to determine the operation of **man**. It is possible to set the "catch-all" variable $**MANOPT** to any string in command line format, with the exception that any spaces used as part of an option's argument must be escaped (preceded by a backslash). **man** will parse $**MANOPT** prior to parsing its own command line. Those options requiring an argument will be overridden by the same options found on the command line. To reset all of the options set in $**MANOPT**, **−D** can be specified as the initial command line option. This will allow man to "forget" about the options specified in $**MANOPT**, although they must still have been valid.

Manual pages are normally stored in **nroff**(1) format under a directory such as */usr/share/man*. In some installations, there may also be preformatted *cat pages* to improve performance. See **manpath**(5) for details of where these files are stored.

This package supports manual pages in multiple languages, controlled by your *locale*. If your system did not set this up for you automatically, then you may need to set $**LC_MESSAGES**, $**LANG**, or another system-dependent environment variable to indicate your preferred locale, usually specified in the **POSIX** format:

<*language*>[**_**<*territory*> [**.**<*character-set*> [**,**<*version*>]]]

If the desired page is available in your *locale*, it will be displayed in lieu of the standard (usually American English) page.

If you find that the translations supplied with this package are not available in your native language and you would like to supply them, please contact the maintainer who will be coordinating such activity.

Individual manual pages are normally written and maintained by the maintainers of the program, function, or other topic that they document, and are not included with this package. If you find that a manual page is missing or inadequate, please report that to the maintainers of the package in question.

For information regarding other features and extensions available with this manual pager, please read the documents supplied with the package.

DEFAULTS

The order of sections to search may be overridden by the environment variable $**MANSECT** or by the **SECTION** directive in */etc/manpath.config*. By default it is as follows:

1 n l 8 3 2 3posix 3pm 3perl 3am 5 4 9 6 7

The formatted manual page is displayed using a *pager*. This can be specified in a number of ways, or else will fall back to a default (see option **−P** for details).

The filters are deciphered by a number of means. Firstly, the command line option **−p** or the environment variable $**MANROFFSEQ** is interrogated. If **−p** was not used and the environment variable was not set, the initial line of the nroff file is parsed for a preprocessor string. To contain a valid preprocessor string, the first line must resemble

'\" <string>

where **string** can be any combination of letters described by option **−p** below.

If none of the above methods provide any filter information, a default set is used.

A formatting pipeline is formed from the filters and the primary formatter (**nroff** or [**tg**]**roff** with **−t**) and executed. Alternatively, if an executable program *mandb_nfmt* (or *mandb_tfmt* with **−t**) exists in the man tree root, it is executed instead. It gets passed the manual source file, the preprocessor string, and optionally the device specified with **−T** or **−E** as arguments.

OPTIONS

Non-argument options that are duplicated either on the command line, in $**MANOPT**, or both, are not harmful. For options that require an argument, each duplication will override the previous argument value.

General options

−C *file*, **−−config−file=***file*
: Use this user configuration file rather than the default of *~/.manpath*.

−d, **−−debug**
: Print debugging information.

−D, **−−default**
: This option is normally issued as the very first option and resets **man's** behaviour to its default. Its use is to reset those options that may have been set in $**MANOPT**. Any options that follow **−D** will have their usual effect.

−−warnings[=*warnings*]
: Enable warnings from *groff*. This may be used to perform sanity checks on the source text of manual pages. *warnings* is a comma-separated list of warning names; if it is not supplied, the default is "mac". See the "Warnings" node in **info groff** for a list of available warning names.

Main modes of operation

−f, **−−whatis**

Equivalent to **whatis**. Display a short description from the manual page, if available. See **whatis**(1) for details.

−k, **−−apropos**

Equivalent to **apropos**. Search the short manual page descriptions for keywords and display any matches. See **apropos**(1) for details.

−K, **−−global−apropos**

Search for text in all manual pages. This is a brute-force search, and is likely to take some time; if you can, you should specify a section to reduce the number of pages that need to be searched. Search terms may be simple strings (the default), or regular expressions if the **−−regex** option is used.

Note that this searches the *sources* of the manual pages, not the rendered text, and so may include false positives due to things like comments in source files. Searching the rendered text would be much slower.

−l, **−−local−file**

Activate "local" mode. Format and display local manual files instead of searching through the system's manual collection. Each manual page argument will be interpreted as an nroff source file in the correct format. No cat file is produced. If '−' is listed as one of the arguments, input will be taken from stdin. When this option is not used, and man fails to find the page required, before displaying the error message, it attempts to act as if this option was supplied, using the name as a filename and looking for an exact match.

−w, **−−where**, **−−path**, **−−location**

Don't actually display the manual page, but do print the location of the source nroff file that would be formatted. If the **−a** option is also used, then print the locations of all source files that match the search criteria.

−W, **−−where−cat**, **−−location−cat**

Don't actually display the manual page, but do print the location of the preformatted cat file that would be displayed. If the **−a** option is also used, then print the locations of all preformatted cat files that match the search criteria.

If **−w** and **−W** are both used, then print both source file and cat file separated by a space. If all of **−w**, **−W**, and **−a** are used, then do this for each possible match.

−c, **−−catman**

This option is not for general use and should only be used by the **catman** program.

−R *encoding*, **−−recode=***encoding*

Instead of formatting the manual page in the usual way, output its source converted to the specified *encoding*. If you already know the encoding of the source file, you can also use **manconv**(1) directly. However, this option allows you to convert several manual pages to a single encoding without having to explicitly state the encoding of each, provided that they were already installed in a structure similar to a manual page hierarchy.

Consider using **man-recode**(1) instead for converting multiple manual pages, since it has an interface designed for bulk conversion and so can be much faster.

Finding manual pages

−L *locale*, **−−locale=***locale*

man will normally determine your current locale by a call to the C function **setlocale**(3) which interrogates various environment variables, possibly including $**LC_MESSAGES** and $**LANG**. To temporarily override the determined value, use this option to supply a *locale* string directly to **man**. Note that it will not take effect until the search for pages actually begins. Output such as the help message will always be displayed in the initially determined locale.

−m *system* [,...] , **−−systems=***system* [,...]
If this system has access to other operating system's manual pages, they can be accessed using this option. To search for a manual page from NewOS's manual page collection, use the option **−m NewOS**.

The *system* specified can be a combination of comma delimited operating system names. To include a search of the native operating system's manual pages, include the system name **man** in the argument string. This option will override the $**SYSTEM** environment variable.

−M *path*, **−−manpath=***path*
Specify an alternate manpath to use. By default, **man** uses **manpath** derived code to determine the path to search. This option overrides the $**MANPATH** environment variable and causes option **−m** to be ignored.

A path specified as a manpath must be the root of a manual page hierarchy structured into sections as described in the man-db manual (under "The manual page system"). To view manual pages outside such hierarchies, see the **−l** option.

−S *list*, **−s** *list*, **−−sections=***list*
The given *list* is a colon- or comma-separated list of sections, used to determine which manual sections to search and in what order. This option overrides the $**MANSECT** environment variable. (The **−s** spelling is for compatibility with System V.)

−e *sub-extension*, **−−extension=***sub-extension*
Some systems incorporate large packages of manual pages, such as those that accompany the **Tcl** package, into the main manual page hierarchy. To get around the problem of having two manual pages with the same name such as **exit**(3), the **Tcl** pages were usually all assigned to section **l**. As this is unfortunate, it is now possible to put the pages in the correct section, and to assign a specific "extension" to them, in this case, **exit**(3tcl). Under normal operation, **man** will display **exit**(3) in preference to **exit**(3tcl). To negotiate this situation and to avoid having to know which section the page you require resides in, it is now possible to give **man** a *sub-extension* string indicating which package the page must belong to. Using the above example, supplying the option **−e tcl** to **man** will restrict the search to pages having an extension of ***tcl**.

−i, **−−ignore−case**
Ignore case when searching for manual pages. This is the default.

−I, **−−match−case**
Search for manual pages case-sensitively.

−−regex
Show all pages with any part of either their names or their descriptions matching each *page* argument as a regular expression, as with **apropos**(1). Since there is usually no reasonable way to pick a "best" page when searching for a regular expression, this option implies **−a**.

−−wildcard
Show all pages with any part of either their names or their descriptions matching each *page* argument using shell-style wildcards, as with **apropos**(1) **−−wildcard**. The *page* argument must match the entire name or description, or match on word boundaries in the description. Since there is usually no reasonable way to pick a "best" page when searching for a wildcard, this option implies **−a**.

−−names−only
If the **−−regex** or **−−wildcard** option is used, match only page names, not page descriptions, as with **whatis**(1). Otherwise, no effect.

−a, **−−all**
By default, **man** will exit after displaying the most suitable manual page it finds. Using this option forces **man** to display all the manual pages with names that match the search criteria.

−u, **−−update**
This option causes **man** to update its database caches of installed manual pages. This is only needed in rare situations, and it is normally better to run **mandb**(8) instead.

−−no−subpages
By default, **man** will try to interpret pairs of manual page names given on the command line as equivalent to a single manual page name containing a hyphen or an underscore. This supports the common pattern of programs that implement a number of subcommands, allowing them to provide manual pages for each that can be accessed using similar syntax as would be used to invoke the subcommands themselves. For example:

```
$ man −aw git diff
/usr/share/man/man1/git−diff.1.gz
```

To disable this behaviour, use the **−−no−subpages** option.

```
$ man −aw −−no−subpages git diff
/usr/share/man/man1/git.1.gz
/usr/share/man/man3/Git.3pm.gz
/usr/share/man/man1/diff.1.gz
```

Controlling formatted output

−P *pager*, **−−pager=***pager*
Specify which output pager to use. By default, **man** uses **pager**, falling back to **cat** if **pager** is not found or is not executable. This option overrides the $**MANPAGER** environment variable, which in turn overrides the $**PAGER** environment variable. It is not used in conjunction with **−f** or **−k**.

The value may be a simple command name or a command with arguments, and may use shell quoting (backslashes, single quotes, or double quotes). It may not use pipes to connect multiple commands; if you need that, use a wrapper script, which may take the file to display either as an argument or on standard input.

−r *prompt*, **−−prompt=***prompt*
If a recent version of **less** is used as the pager, **man** will attempt to set its prompt and some sensible options. The default prompt looks like

Manual page *name*(*sec*) **line** *x*

where *name* denotes the manual page name, *sec* denotes the section it was found under and *x* the current line number. This is achieved by using the $**LESS** environment variable.

Supplying **−r** with a string will override this default. The string may contain the text **$MAN_PN** which will be expanded to the name of the current manual page and its section name surrounded by "(" and ")". The string used to produce the default could be expressed as

```
\ Manual\ page\ \$MAN_PN\ ?ltline\ %lt?L/%L.:
byte\ %bB?s/%s..?\ (END):?pB\ %pB\\%..
(press h for help or q to quit)
```

It is broken into three lines here for the sake of readability only. For its meaning see the **less**(1) manual page. The prompt string is first evaluated by the shell. All double quotes, back-quotes and backslashes in the prompt must be escaped by a preceding backslash. The prompt string may end in an escaped $ which may be followed by further options for less. By default **man** sets the **−ix8** options.

The $**MANLESS** environment variable described below may be used to set a default prompt

string if none is supplied on the command line.

−7, **−−ascii**
When viewing a pure *ascii*(7) manual page on a 7 bit terminal or terminal emulator, some characters may not display correctly when using the *latin1*(7) device description with **GNU nroff**. This option allows pure *ascii* manual pages to be displayed in *ascii* with the *latin1* device. It will not translate any *latin1* text. The following table shows the translations performed: some parts of it may only be displayed properly when using **GNU nroff**'s *latin1*(7) device.

Description	Octal	latin1	ascii
continuation hyphen	255	-	-
bullet (middle dot)	267	•	o
acute accent	264	´	'
multiplication sign	327	×	x

If the *latin1* column displays correctly, your terminal may be set up for *latin1* characters and this option is not necessary. If the *latin1* and *ascii* columns are identical, you are reading this page using this option or **man** did not format this page using the *latin1* device description. If the *latin1* column is missing or corrupt, you may need to view manual pages with this option.

This option is ignored when using options **−t**, **−H**, **−T**, or **−Z** and may be useless for **nroff** other than **GNU's**.

−E *encoding*, **−−encoding=***encoding*
Generate output for a character encoding other than the default. For backward compatibility, *encoding* may be an **nroff** device such as **ascii**, **latin1**, or **utf8** as well as a true character encoding such as **UTF−8**.

−−no−hyphenation, **−−nh**
Normally, **nroff** will automatically hyphenate text at line breaks even in words that do not contain hyphens, if it is necessary to do so to lay out words on a line without excessive spacing. This option disables automatic hyphenation, so words will only be hyphenated if they already contain hyphens.

If you are writing a manual page and simply want to prevent **nroff** from hyphenating a word at an inappropriate point, do not use this option, but consult the **nroff** documentation instead; for instance, you can put "\%" inside a word to indicate that it may be hyphenated at that point, or put "\%" at the start of a word to prevent it from being hyphenated.

−−no−justification, **−−nj**
Normally, **nroff** will automatically justify text to both margins. This option disables full justification, leaving justified only to the left margin, sometimes called "ragged-right" text.

If you are writing a manual page and simply want to prevent **nroff** from justifying certain paragraphs, do not use this option, but consult the **nroff** documentation instead; for instance, you can use the ".na", ".nf", ".fi", and ".ad" requests to temporarily disable adjusting and filling.

−p *string*, **−−preprocessor=***string*
Specify the sequence of preprocessors to run before **nroff** or **troff/groff**. Not all installations will have a full set of preprocessors. Some of the preprocessors and the letters used to designate them are: **eqn** (**e**), **grap** (**g**), **pic** (**p**), **tbl** (**t**), **vgrind** (**v**), **refer** (**r**). This option overrides the $**MANROFFSEQ** environment variable. **zsoelim** is always run as the very first preprocessor.

−t, **−−troff**
Use *groff −mandoc* to format the manual page to stdout. This option is not required in conjunction with **−H**, **−T**, or **−Z**.

−T[*device*], **−−troff−device**[=*device*]
This option is used to change **groff** (or possibly **troff's**) output to be suitable for a device other than the default. It implies **−t**. Examples (provided with Groff-1.17) include **dvi**, **latin1**, **ps**, **utf8**, **X75** and **X100**.

−H[*browser*], **−−html**[=*browser*]
This option will cause **groff** to produce HTML output, and will display that output in a web browser. The choice of browser is determined by the optional *browser* argument if one is provided, by the $**BROWSER** environment variable, or by a compile-time default if that is unset (usually **lynx**). This option implies **−t**, and will only work with **GNU troff**.

−X[*dpi*], **−−gxditview**[=*dpi*]
This option displays the output of **groff** in a graphical window using the **gxditview** program. The *dpi* (dots per inch) may be 75, 75-12, 100, or 100-12, defaulting to 75; the -12 variants use a 12-point base font. This option implies **−T** with the X75, X75-12, X100, or X100-12 device respectively.

−Z, **−−ditroff**
groff will run **troff** and then use an appropriate post-processor to produce output suitable for the chosen device. If *groff −mandoc* is **groff**, this option is passed to **groff** and will suppress the use of a post-processor. It implies **−t**.

Getting help

−?, **−−help**
Print a help message and exit.

−−usage
Print a short usage message and exit.

−V, **−−version**
Display version information.

EXIT STATUS

0 Successful program execution.

1 Usage, syntax or configuration file error.

2 Operational error.

3 A child process returned a non-zero exit status.

16 At least one of the pages/files/keywords didn't exist or wasn't matched.

ENVIRONMENT

MANPATH
If $**MANPATH** is set, its value is used as the path to search for manual pages.

MANROFFOPT
Every time **man** invokes the formatter (**nroff**, **troff**, or **groff**), it adds the contents of $**MANROFFOPT** to the formatter's command line.

MANROFFSEQ
If $**MANROFFSEQ** is set, its value is used to determine the set of preprocessors to pass each manual page through. The default preprocessor list is system dependent.

MANSECT
If $**MANSECT** is set, its value is a colon-delimited list of sections and it is used to determine which manual sections to search and in what order. The default is "1 n l 8 3 2 3posix 3pm 3perl 3am 5 4 9 6 7", unless overridden by the **SECTION** directive in */etc/manpath.config*.

MANPAGER, **PAGER**
If $**MANPAGER** or $**PAGER** is set ($**MANPAGER** is used in preference), its value is used as the name of the program used to display the manual page. By default, **pager** is used, falling back to **cat** if **pager** is not found or is not executable.

The value may be a simple command name or a command with arguments, and may use shell quoting (backslashes, single quotes, or double quotes). It may not use pipes to connect multiple commands; if you need that, use a wrapper script, which may take the file to display either as an argument or on standard input.

MANLESS
If $**MANLESS** is set, its value will be used as the default prompt string for the **less** pager, as if it had been passed using the **−r** option (so any occurrences of the text **$MAN_PN** will be expanded in the same way). For example, if you want to set the prompt string unconditionally to "my prompt string", set $**MANLESS** to '**−Psmy prompt string**'. Using the **−r** option overrides this environment variable.

BROWSER
If $**BROWSER** is set, its value is a colon-delimited list of commands, each of which in turn is used to try to start a web browser for **man −−html**. In each command, *%s* is replaced by a filename containing the HTML output from **groff**, %% is replaced by a single percent sign (%), and *%c* is replaced by a colon (:).

SYSTEM
If $**SYSTEM** is set, it will have the same effect as if it had been specified as the argument to the **−m** option.

MANOPT
If $**MANOPT** is set, it will be parsed prior to **man's** command line and is expected to be in a similar format. As all of the other **man** specific environment variables can be expressed as command line options, and are thus candidates for being included in $**MANOPT** it is expected that they will become obsolete. N.B. All spaces that should be interpreted as part of an option's argument must be escaped.

MANWIDTH
If $**MANWIDTH** is set, its value is used as the line length for which manual pages should be formatted. If it is not set, manual pages will be formatted with a line length appropriate to the current terminal (using the value of $**COLUMNS**, and **ioctl**(2) if available, or falling back to 80 characters if neither is available). Cat pages will only be saved when the default formatting can be used, that is when the terminal line length is between 66 and 80 characters.

MAN_KEEP_FORMATTING
Normally, when output is not being directed to a terminal (such as to a file or a pipe), formatting characters are discarded to make it easier to read the result without special tools. However, if $**MAN_KEEP_FORMATTING** is set to any non-empty value, these formatting characters are retained. This may be useful for wrappers around **man** that can interpret formatting characters.

MAN_KEEP_STDERR
Normally, when output is being directed to a terminal (usually to a pager), any error output from the command used to produce formatted versions of manual pages is discarded to avoid interfering with the pager's display. Programs such as **groff** often produce relatively minor error messages about typographical problems such as poor alignment, which are unsightly and generally confusing when displayed along with the manual page. However, some users want to see them anyway, so, if $**MAN_KEEP_STDERR** is set to any non-empty value, error output will be displayed as usual.

LANG, **LC_MESSAGES**
Depending on system and implementation, either or both of $**LANG** and $**LC_MESSAGES** will be interrogated for the current message locale. **man** will display its messages in that locale (if available). See **setlocale**(3) for precise details.

FILES

/etc/manpath.config
man-db configuration file.

/usr/share/man
A global manual page hierarchy.

SEE ALSO

apropos(1), **groff**(1), **less**(1), **manpath**(1), **nroff**(1), **troff**(1), **whatis**(1), **zsoelim**(1), **manpath**(5), **man**(7), **catman**(8), **mandb**(8)

Documentation for some packages may be available in other formats, such as **info**(1) or HTML.

HISTORY

1990, 1991 – Originally written by John W. Eaton (jwe@che.utexas.edu).

Dec 23 1992: Rik Faith (faith@cs.unc.edu) applied bug fixes supplied by Willem Kasdorp (wkasdo@nikhefk.nikef.nl).

30th April 1994 – 23rd February 2000: Wilf. (G.Wilford@ee.surrey.ac.uk) has been developing and maintaining this package with the help of a few dedicated people.

30th October 1996 – 30th March 2001: Fabrizio Polacco <fpolacco@debian.org> maintained and enhanced this package for the Debian project, with the help of all the community.

31st March 2001 – present day: Colin Watson <cjwatson@debian.org> is now developing and maintaining man-db.

BUGS

https://savannah.nongnu.org/bugs/?group=man-db