

**NAME**

**ps** – report a snapshot of the current processes.

**SYNOPSIS**

**ps** [*option* ...]

**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.

- Unix options, which may be grouped and must be preceded by a dash.
- BSD options, which may be grouped and must not be used with a dash.
- 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.

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,sses,pgid,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 **--sid 123**.

**-123** Select by process group ID (PGID).

**-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** *tylist*

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.

**-P** Add a column showing **psr**.

**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).

**-D** *format*

Set the date format of the **lstart** field to *format*. This format is parsed by **strftime(3)** and should be a maximum of 24 characters to not mis-align columns.

**--date-format** *format*

Identical to **-D**.

**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**

**--signames**

Show signal masks using abbreviated signal names and expands the column. If the column width cannot show all signals, the column will end with a plus "+". Columns with only a hyphen have no signals.

**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 *simple*, *list*, *output*, *threads*, *misc*, or *all*. 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 width 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 I/O)
- 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 Linux 2.6)
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 I/O)
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	minflt	number of minor page faults
M	majflt	number of major page faults
n	cminflt	cumulative minor page faults
N	cmajflt	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 kibibytes
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)*. 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 <b>pcpu</b> ).
%mem	%MEM	ratio of the process's resident set size to the physical memory on the machine, expressed as a percentage. (alias <b>pmem</b> ).
ag_id	AGID	The autogroup identifier associated with a process which operates in conjunction with the CFS scheduler to improve interactive desktop performance.
ag_nice	AGNI	The autogroup nice value which affects scheduling of all processes in that group.

args	COMMAND	<p>A 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 “&lt;defunct&gt;” is partly dead, waiting to be fully destroyed by its parent. Sometimes the process arguments will be unavailable; when this happens, <b>ps</b> instead reports the executable name in brackets. (alias <b>cmd</b>, <b>command</b>). See also the <b>comm</b> format keyword, the <b>-f</b> option, and the <b>c</b> option.</p> <p>When specified last, this column will extend to the edge of the display. If <b>ps</b> can not determine the 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 <b>TERM</b> variable, and so on). The <b>COLUMNS</b> environment variable or <b>--cols</b> option may be used to exactly determine the width in this case. The <b>w</b> or <b>-w</b> option may be also be used to adjust width.</p>																		
blocked	BLOCKED	mask of the blocked signals, see <i>signal(7)</i> . According to the width of the field, a 32 or 64-bit mask in hexadecimal format is displayed, unless the <b>--signames</b> option is used. (alias <b>sig_block</b> , <b>sigmask</b> ).																		
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 <b>lstart</b> , <b>start</b> , <b>start_time</b> , and <b>stime</b> .																		
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 <b>%cpu</b> ).																		
caught	CAUGHT	mask of the caught signals, see <i>signal(7)</i> . According to the width of the field, a 32 or 64 bits mask in hexadecimal format is displayed, unless the <b>--signames</b> option is used. (alias <b>sig_catch</b> , <b>sigcatch</b> ).																		
cgname	CGNAME	display name of control groups to which the process belongs.																		
cgroup	CGROUP	display control groups to which the process belongs.																		
cgroupns	CGROUPNS	Unique inode number describing the namespace the process belongs to. See <i>namespaces(7)</i> .																		
class	CLS	<p>scheduling class of the process. (alias <b>policy</b>, <b>cls</b>). Field’s possible values are:</p> <table><tr><td>–</td><td>not reported</td></tr><tr><td>TS</td><td>SCHED_OTHER</td></tr><tr><td>FF</td><td>SCHED_FIFO</td></tr><tr><td>RR</td><td>SCHED_RR</td></tr><tr><td>B</td><td>SCHED_BATCH</td></tr><tr><td>ISO</td><td>SCHED_ISO</td></tr><tr><td>IDL</td><td>SCHED_IDLE</td></tr><tr><td>DLN</td><td>SCHED_DEADLINE</td></tr><tr><td>?</td><td>unknown value</td></tr></table>	–	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
–	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 <b>policy</b> , <b>cls</b> ). Field's possible values are: <table><tr><td>–</td><td>not reported</td></tr><tr><td>TS</td><td>SCHED_OTHER</td></tr><tr><td>FF</td><td>SCHED_FIFO</td></tr><tr><td>RR</td><td>SCHED_RR</td></tr><tr><td>B</td><td>SCHED_BATCH</td></tr><tr><td>ISO</td><td>SCHED_ISO</td></tr><tr><td>IDL</td><td>SCHED_IDLE</td></tr><tr><td>DLN</td><td>SCHED_DEADLINE</td></tr><tr><td>?</td><td>unknown value</td></tr></table>	–	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
–	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 <b>args</b> . (alias <b>args</b> , <b>command</b> ).																		
comm	COMMAND	command name (only the executable name). The output in this column may contain spaces. (alias <b>ucmd</b> , <b>ucomm</b> ). See also the <b>args</b> format keyword, the <b>–f</b> option, and the <b>c</b> option.  When specified last, this column will extend to the edge of the display. If <b>ps</b> 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 <b>TERM</b> variable, and so on). The <b>COLUMNS</b> environment variable or <b>–cols</b> option may be used to exactly determine the width in this case. The <b>w</b> or <b>–w</b> option may be also be used to adjust width.																		
command	COMMAND	See <b>args</b> . (alias <b>args</b> , <b>command</b> ).																		
cp	CP	per–mill (tenths of a percent) CPU usage. (see <b>%cpu</b> ).																		
cputime	TIME	cumulative CPU time, "[DD–]hh:mm:ss" format. (alias <b>time</b> ).																		
cputimes	TIME	cumulative CPU time in seconds (alias <b>times</b> ).																		
cuc	%CUC	The CPU utilization of a process, including dead children, in an extended "###.###" format. (see also <b>%cpu</b> , <b>c</b> , <b>cp</b> , <b>cuu</b> , <b>pcpu</b> ).																		
cuu	%CUU	The CPU utilization of a process in an extended "###.###" format. (see also <b>%cpu</b> , <b>c</b> , <b>cp</b> , <b>cuc</b> , <b>pcpu</b> ).																		
docker	DOCKER	The abbreviated id of the docker container within which a task is running. If a process is not running inside a container, a dash (‘–’) will be shown.																		
drs	DRS	data resident set size, the amount of private memory <i>reserved</i> by a process. It is also known as DATA. Such memory may not yet be mapped to <b>rss</b> but will always be included in the <b>vsz</b> amount.																		
egid	EGID	effective group ID number of the process as a decimal integer. (alias <b>gid</b> ).																		
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 <b>group</b> ).																		

<b>eip</b>	<b>EIP</b>	instruction pointer. As of kernel 4.9.xx will be zeroed out unless task is exiting or being core dumped.
<b>esp</b>	<b>ESP</b>	stack pointer. As of kernel 4.9.xx will be zeroed out unless task is exiting or being core dumped.
<b>etime</b>	<b>ELAPSED</b>	elapsed time since the process was started, in the form [[DD-]hh:]mm:ss.
<b>etimes</b>	<b>ELAPSED</b>	elapsed time since the process was started, in seconds.
<b>environ</b> <b>euid</b>	<b>ENVIRON</b> <b>EUID</b>	environment variables for the process. effective user ID (alias <b>uid</b> ).
<b>euser</b>	<b>EUSER</b>	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 <b>n</b> option can be used to force the decimal representation. (alias <b>uname</b> , <b>user</b> ).
<b>exe</b>	<b>EXE</b>	path to the executable. Useful if path cannot be printed via <b>cmd</b> , <b>comm</b> or <b>args</b> format options.
<b>f</b>	<b>F</b>	flags associated with the process, see the <b>PROCESS FLAGS</b> section. (alias <b>flag</b> , <b>flags</b> ).
<b>fds</b>	<b>FDS</b>	total open file descriptors.
<b>fgid</b>	<b>FGID</b>	filesystem access group ID. (alias <b>fsgid</b> ).
<b>fgroup</b>	<b>FGROUP</b>	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 <b>fsgroup</b> ).
<b>flag</b>	<b>F</b>	see <b>f</b> . (alias <b>f</b> , <b>flags</b> ).
<b>flags</b>	<b>F</b>	see <b>f</b> . (alias <b>f</b> , <b>flag</b> ).
<b>fname</b>	<b>COMMAND</b>	first 8 bytes of the base name of the process's executable file. The output in this column may contain spaces.
<b>fuid</b>	<b>FUID</b>	filesystem access user ID. (alias <b>fsuid</b> ).
<b>fuser</b>	<b>FUSER</b>	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.
<b>gid</b>	<b>GID</b>	see <b>egid</b> . (alias <b>egid</b> ).
<b>group</b>	<b>GROUP</b>	see <b>egroup</b> . (alias <b>egroup</b> ).
<b>htprv</b>	<b>HTPRV</b>	The amount of private memory backed by hugetlbfs page which is not counted in the <b>rss</b> or <b>pss</b> format options.
<b>htshr</b>	<b>HTSHR</b>	The amount of shared memory backed by hugetlbfs page which is not counted in the <b>rss</b> or <b>pss</b> format options.

<b>ignored</b>	<b>IGNORED</b>	mask of the ignored signals, see <i>signal(7)</i> . According to the width of the field, a 32 or 64 bits mask in hexadecimal format is displayed, unless the <b>--signames</b> option is used. (alias <b>sig_ignore</b> , <b>signore</b> ).
<b>ipcns</b>	<b>IPCNS</b>	Unique inode number describing the namespace the process belongs to. See <i>namespaces(7)</i> .
<b>label</b>	<b>LABEL</b>	security label, most commonly used for SELinux context data. This is for the <i>Mandatory Access Control</i> ("MAC") found on high-security systems.
<b>lstart</b>	<b>STARTED</b>	time the command started. This will be in the form "DDD mmm HH:MM:SS YYYY" unless changed by the <b>-D</b> option.
<b>lsession</b>	<b>SESSION</b>	displays the login session identifier of a process, if systemd support has been included.
<b>luid</b>	<b>LUID</b>	displays Login ID associated with a process.
<b>lwp</b>	<b>LWP</b>	light weight process (thread) ID of the dispatchable entity (alias <b>spid</b> , <b>tid</b> ). See <b>tid</b> for additional information.
<b>lxc</b>	<b>LXC</b>	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.
<b>machine</b>	<b>MACHINE</b>	displays the machine name for processes assigned to VM or container, if systemd support has been included.
<b>majflt</b>	<b>MAJFLT</b>	The number of major page faults that have occurred with this process.
<b>minflt</b>	<b>MINFLT</b>	The number of minor page faults that have occurred with this process.
<b>mntns</b>	<b>MNTNS</b>	Unique inode number describing the namespace the process belongs to. See <i>namespaces(7)</i> .
<b>netns</b>	<b>NETNS</b>	Unique inode number describing the namespace the process belongs to. See <i>namespaces(7)</i> .
<b>ni</b>	<b>NI</b>	nice value. This ranges from 19 (nicest) to -20 (not nice to others), see <i>nice(1)</i> . (alias <b>nice</b> ).
<b>nice</b>	<b>NI</b>	see <b>ni</b> . (alias <b>ni</b> ).
<b>nlwp</b>	<b>NLWP</b>	number of lwps (threads) in the process. (alias <b>thcount</b> ).
<b>numa</b>	<b>NUMA</b>	The node associated with the most recently used processor. A <b>-1</b> means that NUMA information is unavailable.
<b>nwchan</b>	<b>WCHAN</b>	address of the kernel function where the process is sleeping (use <b>wchan</b> if you want the kernel function name).
<b>oom</b>	<b>OOM</b>	Out of Memory Score. The value, ranging from 0 to +1000, used to select task(s) to kill when memory is exhausted.

<b>oomadj</b>	<b>OOMADJ</b>	Out of Memory Adjustment Factor. The value is added to the current out of memory score which is then used to determine which task to kill when memory is exhausted.
<b>ouid</b>	<b>OWNER</b>	displays the Unix user identifier of the owner of the session of a process, if systemd support has been included.
<b>pcap</b>	<b>PCAP</b>	Permitted Capabilites of the process, displayed as a hexadecimal bitmask. See <i>capabilities(7)</i> .
<b>pcaps</b>	<b>PCAPS</b>	Permitted Capabilites of the process, displayed as a string of capability names. See <i>capabilities(7)</i> .
<b>pcpu</b>	<b>%CPU</b>	see <b>%cpu</b> . (alias <b>%cpu</b> ).
<b>pending</b>	<b>PENDING</b>	mask of the pending signals. See <i>signal(7)</i> . Signals pending on the process are distinct from signals pending on individual threads. Use the <b>m</b> option or the <b>-m</b> option to see both. According to the width of the field, a 32 or 64 bits mask in hexadecimal format is displayed, unless the <b>--signames</b> option is used. (alias <b>sig</b> ).
<b>pgid</b>	<b>PGID</b>	process group ID or, equivalently, the process ID of the process group leader. (alias <b>pgrp</b> ).
<b>pgrp</b>	<b>PGRP</b>	see <b>pgid</b> . (alias <b>pgid</b> ).
<b>pid</b>	<b>PID</b>	a number representing the process ID (alias <b>tgid</b> ).
<b>pidns</b>	<b>PIDNS</b>	Unique inode number describing the namespace the process belongs to. See <i>namespaces(7)</i> .
<b>pmem</b>	<b>%MEM</b>	see <b>%mem</b> . (alias <b>%mem</b> ).
<b>policy</b>	<b>POL</b>	scheduling class of the process. (alias <b>class</b> , <b>cls</b> ). Possible values are: <div style="margin-left: 40px;"> — 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 </div>
<b>ppid</b>	<b>PPID</b>	parent process ID.
<b>pri</b>	<b>PRI</b>	priority of the process. Higher number means higher priority.
<b>psr</b>	<b>PSR</b>	processor that process last executed on.
<b>pss</b>	<b>PSS</b>	Proportional share size, the non-swapped physical memory, with shared memory proportionally accounted to all tasks mapping it.

<b>rbytes</b>	<b>RBYTES</b>	Number of bytes which this process really did cause to be fetched from the storage layer.
<b>rchars</b>	<b>RCHARS</b>	Number of bytes which this task has caused to be read from storage.
<b>rgid</b>	<b>RGID</b>	real group ID.
<b>rgroup</b>	<b>RGROUP</b>	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.
<b>rops</b>	<b>ROPS</b>	Number of read I/O operations—that is, system calls such as <b>read(2)</b> and <b>pread(2)</b> .
<b>rss</b>	<b>RSS</b>	resident set size, the non-swapped physical memory that a task has used (in kibibytes). (alias <b>rssize</b> , <b>rsz</b> ).
<b>rssize</b>	<b>RSS</b>	see <b>rss</b> . (alias <b>rss</b> , <b>rsz</b> ).
<b>rsz</b>	<b>RSZ</b>	see <b>rss</b> . (alias <b>rss</b> , <b>rssize</b> ).
<b>rtprio</b>	<b>RTPRIO</b>	realtime priority.
<b>ruid</b>	<b>RUID</b>	real user ID.
<b>ruser</b>	<b>RUSER</b>	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.
<b>s</b>	<b>S</b>	minimal state display (one character). See section <b>PROCESS STATE CODES</b> for the different values. See also <b>stat</b> if you want additional information displayed. (alias <b>state</b> ).
<b>sched</b>	<b>SCH</b>	scheduling policy of the process. The policies <b>SCHED_OTHER</b> ( <b>SCHED_NORMAL</b> ), <b>SCHED_FIFO</b> , <b>SCHED_RR</b> , <b>SCHED_BATCH</b> , <b>SCHED_ISO</b> , <b>SCHED_IDLE</b> and <b>SCHED_DEADLINE</b> are respectively displayed as 0, 1, 2, 3, 4, 5 and 6.
<b>seat</b>	<b>SEAT</b>	displays the identifier associated with all hardware devices assigned to a specific workplace, if systemd support has been included.
<b>sess</b>	<b>SESS</b>	session ID or, equivalently, the process ID of the session leader. (alias <b>session</b> , <b>sid</b> ).
<b>sgi_p</b>	<b>P</b>	processor that the process is currently executing on. Displays "*" if the process is not currently running or runnable.
<b>sgid</b>	<b>SGID</b>	saved group ID. (alias <b>svgid</b> ).
<b>sgroup</b>	<b>SGROUP</b>	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.
<b>sid</b>	<b>SID</b>	see <b>sess</b> . (alias <b>sess</b> , <b>session</b> ).

<b>sig</b>	<b>PENDING</b>	see <b>pending</b> . (alias <b>pending</b> , <b>sig_pend</b> ).
<b>sigcatch</b>	<b>CAUGHT</b>	see <b>caught</b> . (alias <b>caught</b> , <b>sig_catch</b> ).
<b>sigignore</b>	<b>IGNORED</b>	see <b>ignored</b> . (alias <b>ignored</b> , <b>sig_ignore</b> ).
<b>sigmask</b>	<b>BLOCKED</b>	see <b>blocked</b> . (alias <b>blocked</b> , <b>sig_block</b> ).
<b>size</b>	<b>SIZE</b>	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!
<b>slice</b>	<b>SLICE</b>	displays the slice unit which a process belongs to, if systemd support has been included.
<b>spid</b>	<b>SPID</b>	see <b>lwp</b> . (alias <b>lwp</b> , <b>tid</b> ).
<b>stackp</b>	<b>STACKP</b>	address of the bottom (start) of stack for the process.
<b>start</b>	<b>STARTED</b>	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 <b>bsdstart</b> , <b>start</b> , <b>start_time</b> , and <b>stime</b> .
<b>start_time</b>	<b>START</b>	starting time or date of the process. Only the year will be displayed if the process was not started the same year <b>ps</b> was invoked, or "MmmDD" if it was not started the same day, or "HH:MM" otherwise. See also <b>bsdstart</b> , <b>start</b> , <b>lstart</b> , and <b>stime</b> .
<b>stat</b>	<b>STAT</b>	multi-character process state. See section <b>PROCESS STATE CODES</b> for the different values meaning. See also <b>s</b> and <b>state</b> if you just want the first character displayed.
<b>state</b>	<b>S</b>	see <b>s</b> . (alias <b>s</b> ).
<b>stime</b>	<b>STIME</b>	see <b>start_time</b> . (alias <b>start_time</b> ).
<b>suid</b>	<b>SUID</b>	saved user ID. (alias <b>svuid</b> ).
<b>supgid</b>	<b>SUPGID</b>	group ids of supplementary groups, if any. See <b>getgroups(2)</b> .
<b>supgrp</b>	<b>SUPGRP</b>	group names of supplementary groups, if any. See <b>getgroups(2)</b> .
<b>suser</b>	<b>SUSER</b>	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 <b>svuser</b> ).
<b>svgid</b>	<b>SVGID</b>	see <b>sgid</b> . (alias <b>sgid</b> ).
<b>svuid</b>	<b>SVUID</b>	see <b>suid</b> . (alias <b>suid</b> ).



<b>sz</b>	<b>SZ</b>	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 <b>vsz</b> and <b>rss</b> .
<b>tgid</b>	<b>TGID</b>	a number representing the thread group to which a task belongs (alias <b>pid</b> ). It is the process ID of the thread group leader.
<b>thcount</b>	<b>THCNT</b>	see <b>nlwp</b> . (alias <b>nlwp</b> ). number of kernel threads owned by the process.
<b>tid</b>	<b>TID</b>	the unique number representing a dispatchable entity (alias <b>spid</b> , <b>tid</b> ). This value may also appear as: a process ID ( <b>pid</b> ); a process group ID ( <b>pgrp</b> ); a session ID for the session leader ( <b>sid</b> ); a thread group ID for the thread group leader ( <b>tgid</b> ); and a tty process group ID for the process group leader ( <b>tpgid</b> ).
<b>time</b>	<b>TIME</b>	cumulative CPU time, "[DD-]HH:MM:SS" format. (alias <b>cputime</b> ).
<b>timens</b>	<b>TIMENS</b>	Unique inode number describing the namespace the process belongs to. See <i>namespaces(7)</i> .
<b>times</b>	<b>TIME</b>	cumulative CPU time in seconds (alias <b>cputimes</b> ).
<b>tname</b>	<b>TTY</b>	controlling tty (terminal). (alias <b>tt</b> , <b>tty</b> ).
<b>tpgid</b>	<b>TPGID</b>	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.
<b>trs</b>	<b>TRS</b>	text resident set size, the amount of physical memory devoted to executable code.
<b>tt</b>	<b>TT</b>	controlling tty (terminal). (alias <b>tname</b> , <b>tty</b> ).
<b>tty</b>	<b>TT</b>	controlling tty (terminal). (alias <b>tname</b> , <b>tt</b> ).
<b>ucmd</b>	<b>CMD</b>	see <b>comm</b> . (alias <b>comm</b> , <b>ucomm</b> ).
<b>ucomm</b>	<b>COMMAND</b>	see <b>comm</b> . (alias <b>comm</b> , <b>ucmd</b> ).
<b>uid</b>	<b>UID</b>	see <b>euid</b> . (alias <b>euid</b> ).
<b>uname</b>	<b>USER</b>	see <b>euser</b> . (alias <b>euser</b> , <b>user</b> ).
<b>unit</b>	<b>UNIT</b>	displays unit which a process belongs to, if systemd support has been included.
<b>user</b>	<b>USER</b>	see <b>euser</b> . (alias <b>euser</b> , <b>uname</b> ).
<b>userns</b>	<b>USERNS</b>	Unique inode number describing the namespace the process belongs to. See <i>namespaces(7)</i> .
<b>uss</b>	<b>USS</b>	Unique set size, the non-swapped physical memory, which is not shared with an another task.

<b>utns</b>	<b>UTSNS</b>	Unique inode number describing the namespace the process belongs to. See <i>namespaces(7)</i> .
<b>uunit</b>	<b>UUNIT</b>	displays user unit which a process belongs to, if systemd support has been included.
<b>vsiz</b>	<b>VSZ</b>	see <b>vsz</b> . (alias <b>vsz</b> ).
<b>vsz</b>	<b>VSZ</b>	virtual memory size of the process in KiB (1024-byte units). Device mappings are currently excluded; this is subject to change. (alias <b>vsiz</b> ).
<b>wbytes</b>	<b>WBYTES</b>	Number of bytes which this process caused to be sent to the storage layer.
<b>wcbytes</b>	<b>WCBYTES</b>	Number of cancelled write bytes.
<b>wchan</b>	<b>WCHAN</b>	name of the kernel function in which the process is sleeping.
<b>wchars</b>	<b>WCHARS</b>	Number of bytes which this task has caused, or shall cause to be written to disk.
<b>wops</b>	<b>WOPS</b>	Number of write I/O operations—that is, system calls such as <b>write(2)</b> and <b>pwrite(2)</b> .

## 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.

### LIBPROC\_HIDE\_KERNEL

Set this to any value to hide kernel threads normally displayed with the **-e** option. This is equivalent to selecting **--ppid 2 -p 2 --deselect** instead. Also works in BSD mode.

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

**BUGS**

The fields **bsdstart** and **start** will only show the abbreviated month name in English. The fields **lstart** and **stime** will show the abbreviated month name in the configured locale but may exceed the column width due to the different lengths for abbreviated month and day names across languages.

**SEE ALSO**

**pgrep(1)**, **pstree(1)**, **top(1)**, **strftime(3)**, **proc(5)**, **capabilities(7)**.

**STANDARDS**

This **ps** conforms to the following standards.

- Version 2 of the Single Unix Specification
- The Open Group Technical Standard Base Specifications, Issue 6
- IEEE Std 1003.1, 2004 Edition
- X/Open System Interfaces Extension [UP XSI]
- 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.