



Linux Ubuntu 26.04 Manual Pages on command 'taskset.1'

\$ man taskset.1

TASKSET(1) User Commands TASKSET(1)

NAME

taskset - set or retrieve a process's CPU affinity

SYNOPSIS

taskset [options] mask command [argument...]

taskset [options] -p [mask] pid

DESCRIPTION

The taskset command is used to set or retrieve the CPU affinity of a running process given its pid, or to launch a new command with a given CPU affinity. CPU affinity is a scheduler property that "bonds" a process to a given set of CPUs on the system. The Linux scheduler will honor the given CPU affinity and the process will not run on any other CPUs. Note that the Linux scheduler also supports natural CPU affinity: the scheduler attempts to keep processes on the same CPU as long as practical for performance reasons. Therefore, forcing a specific CPU affinity is useful only in certain applications. The affinity of some processes like kernel per-CPU threads cannot be set.

The CPU affinity is represented as a bitmask, with the lowest order bit corresponding to the first logical CPU and the highest order bit corresponding to the last logical CPU. Not all CPUs may exist on a given system but a mask may specify more CPUs than are present. A retrieved mask will reflect only the bits that correspond to CPUs physically on the system.

If an invalid mask is given (i.e., one that corresponds to no valid CPUs on the current system) an error is returned. The masks may be specified in hexadecimal (with or without a leading "0x"), or as a CPU list with the --cpu-list option. For example,

0x00000001

is processor #0,

0x00000003

is processors #0 and #1,

FFFFFFFF

is processors #0 through #31,

0x32

is processors #1, #4, and #5,

--cpu-list 0-2,6

is processors #0, #1, #2, and #6.

--cpu-list 0-10:2

is processors #0, #2, #4, #6, #8 and #10. The suffix ":N" specifies stride in the range,

for example 0-10:3 is interpreted as 0,3,6,9 list.

When taskset returns, it is guaranteed that the given program has been scheduled to a legal CPU.

OPTIONS

-a, --all-tasks

Set or retrieve the CPU affinity of all the tasks (threads) for a given PID.

-c, --cpu-list

Interpret mask as numerical list of processors instead of a bitmask. Numbers are separated by commas and may include ranges. For example: 0,5,8-11.

-p, --pid

Operate on an existing PID and do not launch a new task.

-h, --help

Display help text and exit.

-V, --version

Display version and exit.

USAGE

The default behavior is to run a new command with a given affinity mask:

taskset mask command [arguments]

You can also retrieve the CPU affinity of an existing task:

taskset -p pid

Or set it:

taskset -p mask pid

When a cpu-list is specified for an existing process, the -p and -c options must be grouped together:

```
taskset -pc cpu-list pid
```

The --cpu-list form is applicable only for launching new commands:

```
taskset --cpu-list cpu-list command
```

PERMISSIONS

A user can change the CPU affinity of a process belonging to the same user. A user must possess CAP_SYS_NICE to change the CPU affinity of a process belonging to another user. A user can retrieve the affinity mask of any process.

EXIT STATUS

taskset returns 0 in its affinity-getting mode as long as the provided PID exists.

taskset returns 0 in its affinity-setting mode as long as the underlying sched_setaffinity(2) system call does. The success of the command does not guarantee that the specified thread has actually migrated to the indicated CPU(s), but only that the thread will not migrate to a CPU outside the new affinity mask. For example, the affinity of the kernel thread kswapd can be set, but the thread may not immediately migrate and is not guaranteed to ever do so:

```
$ ps ax -o comm,psr,pid | grep kswapd
```

```
kswapd0      4    82
```

```
$ sudo taskset -p 1 82
```

```
pid 82?s current affinity mask: 1
```

```
pid 82?s new affinity mask: 1
```

```
$ echo $?
```

```
0
```

```
$ ps ax -o comm,psr,pid | grep kswapd
```

```
kswapd0      4    82
```

```
$ taskset -p 82
```

```
pid 82?s current affinity mask: 1
```

In contrast, when the user specifies an illegal affinity, taskset will print an error and return 1:

```
$ ps ax -o comm,psr,pid | grep ksoftirqd/0
```

```
ksoftirqd/0   0    14
```

```
$ sudo taskset -p 1 14
```

pid 14's current affinity mask: 1

taskset: failed to set pid 14's affinity: Invalid argument

\$ echo \$?

1

AUTHORS

Written by Robert M. Love.

COPYRIGHT

Copyright ? 2004 Robert M. Love. This is free software; see the source for copying conditions. There is NO warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

SEE ALSO

chrt(1), nice(1), renice(1), sched_getaffinity(2), sched_setaffinity(2)

See sched(7) for a description of the Linux scheduling scheme.

REPORTING BUGS

For bug reports, use the issue tracker <<https://github.com/util-linux/util-linux/issues>>.

AVAILABILITY

The taskset command is part of the util-linux package which can be downloaded from Linux Kernel Archive <<https://www.kernel.org/pub/linux/utils/util-linux/>>.

util-linux 2.41.3

2025-12-15

TASKSET(1)