

**NAME**

**pthread\_getaffinity\_np**, **pthread\_setaffinity\_np** - manage CPU affinity

**LIBRARY**

POSIX Threads Library (libpthread, -lpthread)

**SYNOPSIS**

```
#include <pthread_np.h>
```

*int*

```
pthread_getaffinity_np(pthread_t td, size_t cpusetsize, cpuset_t *cpusetp);
```

*int*

```
pthread_setaffinity_np(pthread_t td, size_t cpusetsize, const cpuset_t *cpusetp);
```

**DESCRIPTION**

**pthread\_getaffinity\_np()** and **pthread\_setaffinity\_np()** allow the manipulation of sets of CPUs available to the specified thread.

Masks of type *cpuset\_t* are composed using the CPU\_SET macros. If the user-supplied mask is not large enough to fit all of the matching CPUs, **pthread\_getaffinity\_np()** fails with ERANGE. Calls to **pthread\_setaffinity\_np()** tolerate masks of any size with no restrictions. The kernel uses the meaningful part of the mask, where the upper bound is the maximum CPU id present in the system. If bits for non-existing CPUs are set, calls to **pthread\_setaffinity\_np()** fail with EINVAL.

The supplied mask should have a size of *cpusetsize* bytes. This size is usually provided by calling sizeof(*cpuset\_t*) which is ultimately determined by the value of CPU\_SETSIZE as defined in *<sys/cpuset.h>*.

**pthread\_getaffinity\_np()** retrieves the mask from the thread specified by *td*, and stores it in the space provided by *cpusetp*.

**pthread\_setaffinity\_np()** attempts to set the mask for the thread specified by *td* to the value in *cpusetp*.

**RETURN VALUES**

If successful, the **pthread\_getaffinity\_np()** and **pthread\_setaffinity\_np()** functions will return zero. Otherwise an error number will be returned to indicate the error.

**ERRORS**

The **pthread\_getaffinity\_np()** and **pthread\_setaffinity\_np()** functions may fail if:

- [EINVAL] The *cpusetp* argument specified when calling **pthread\_setaffinity\_np()** was not a valid value.
- [EDEADLK] The **pthread\_setaffinity\_np()** call would leave a thread without a valid CPU to run on because the set does not overlap with the thread's anonymous mask.
- [EFAULT] The *cpusetp* pointer passed was invalid.
- [ESRCH] The thread specified by the *td* argument could not be found.
- [ERANGE] The *cpusetsize* was smaller than needed to fit all of the matching CPUs.
- [EPERM] The calling thread did not have the credentials required to complete the operation.

### SEE ALSO

cpuset(1), cpuset(2), cpuset\_getid(2), cpuset\_setid(2), pthread(3), pthread\_attr\_getaffinity\_np(3), pthread\_attr\_setaffinity\_np(3), pthread\_np(3)

### STANDARDS

The **pthread\_getaffinity\_np** and **pthread\_setaffinity\_np** functions are non-standard FreeBSD extensions and may be not available on other operating systems.

### HISTORY

The **pthread\_getaffinity\_np** and **pthread\_setaffinity\_np** function first appeared in FreeBSD 7.2.

### AUTHORS

The **pthread\_getaffinity\_np** and **pthread\_setaffinity\_np** functions were written by David Xu <davidxu@FreeBSD.org>, and this manpage was written by Xin LI <delphij@FreeBSD.org>.