

NAME

thr_self - return thread identifier for the calling thread

LIBRARY

Standard C Library (libc, -lc)

SYNOPSIS

```
#include <sys/thr.h>
```

int

```
thr_self(long *id);
```

DESCRIPTION

The **thr_self()** system call stores the system-wide thread identifier for the current kernel-scheduled thread in the variable pointed by the argument *id*.

The thread identifier is an integer in the range from `PID_MAX + 2` (100001) to `INT_MAX`. The thread identifier is guaranteed to be unique at any given time, for each running thread in the system. After the thread exits, the identifier may be reused.

RETURN VALUES

If successful, **thr_self()** will return zero, otherwise -1 is returned, and *errno* is set to indicate the error.

ERRORS

The **thr_self()** operation may return the following errors:

[EFAULT] The memory pointed to by the *id* argument is not valid.

SEE ALSO

`_umtx_op(2)`, `thr_exit(2)`, `thr_kill(2)`, `thr_kill2(2)`, `thr_new(2)`, `thr_set_name(2)`,
`pthread_getthreadid_np(3)`, `pthread_self(3)`

STANDARDS

The **thr_self()** system call is non-standard and is used by 1:1 Threading Library (libthr, -lthr) to implement IEEE Std 1003.1-2001 ("POSIX.1") `pthread(3)` functionality.

HISTORY

The **thr_self()** system call first appeared in FreeBSD 5.2.