

NAME

get_robust_list, set_robust_list - get/set list of robust futexes

SYNOPSIS

```
#include <linux/futex.h>
```

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

```
#include <syscall.h>
```

```
long get_robust_list(int pid, struct robust_list_head **head_ptr,  
                    size_t *len_ptr);
```

```
long set_robust_list(struct robust_list_head *head, size_t len);
```

Note: There are no glibc wrappers for these system calls; see NOTES.

DESCRIPTION

The robust futex implementation needs to maintain per-thread lists of robust futexes which are unlocked when the thread exits. These lists are managed in user space; the kernel is notified about only the location of the head of the list.

The `get_robust_list()` system call returns the head of the robust futex list of the thread whose thread ID is specified in *pid*. If *pid* is 0, the head of the list for the calling thread is returned. The list head is stored in the location pointed to by *head_ptr*. The size of the object pointed to by ***head_ptr* is stored in *len_ptr*.

The `set_robust_list()` system call requests the kernel to record the head of the list of robust futexes owned by the calling thread. The *head* argument is the list head to record. The *len* argument should be `sizeof(*head)`.

RETURN VALUE

The `set_robust_list()` and `get_robust_list()` system calls return zero when the operation is successful, an error code otherwise.

ERRORS

The `set_robust_list()` system call can fail with the following error:

EINVAL

len does not match the size of structure `struct robust_list_head` expected by kernel.

The `get_robust_list()` system call can fail with the following errors:

EPERM

The calling process does not have permission to see the robust futex list of the thread with the thread ID *pid*, and does not have the `CAP_SYS_PTRACE` capability.

ESRCH

No thread with the thread ID *pid* could be found.

EFAULT

The head of the robust futex list can't be stored at the location *head*.

VERSIONS

These system calls were added in Linux 2.6.17. No library support is provided; use [syscall\(2\)](#).

NOTES

These system calls are not needed by normal applications. No support for them is provided in glibc. In the unlikely event that you want to call them directly, use [syscall\(2\)](#).

A thread can have only one robust futex list; therefore applications that wish to use this functionality should use the robust mutexes provided by glibc.

SEE ALSO

[futex\(2\)](#)

Documentation/robust-futexes.txt and *Documentation/robust-futex-ABI.txt* in the Linux kernel source tree

COLOPHON

This page is part of release 3.74 of the Linux *man-pages* project. A description of the project, information about reporting bugs, and the latest version of this page, can be found at <http://www.kernel.org/doc/man-pages/>.