Full credit is given to the above companies including the OS that this PDF file was generated!

Red Hat Enterprise Linux Release 9.2 Manual Pages on 'pthread_detach.3p' command

$ man pthread_detach.3p

PTHREAD_DETACH(3P) POSIX Programmer's Manual PTHREAD_DETACH(3P)

PROLOG

This manual page is part of the POSIX Programmer's Manual. The Linux implementation of this interface may differ (consult the corresponding Linux manual page for details of Linux behavior), or the interface may not be implemented on Linux.

NAME

pthread_detach ? detach a thread

SYNOPSIS

#include <pthread.h>

int pthread_detach(pthread_t thread);

DESCRIPTION

The pthread_detach() function shall indicate to the implementation that storage for the thread thread can be reclaimed when that thread termi? nates. If thread has not terminated, pthread_detach() shall not cause it to terminate.

The behavior is undefined if the value specified by the thread argument to pthread_detach() does not refer to a joinable thread.

RETURN VALUE

If the call succeeds, pthread_detach() shall return 0; otherwise, an error number shall be returned to indicate the error.

ERRORS

The pthread_detach() function shall not return an error code of

[EINTR].

The following sections are informative.

EXAMPLES

None.

APPLICATION USAGE

None.

RATIONALE

The pthread_join() or pthread_detach() functions should eventually be called for every thread that is created so that storage associated with the thread may be reclaimed.

It has been suggested that a ``detach'' function is not necessary; the detachstate thread creation attribute is sufficient, since a thread need never be dynamically detached. However, need arises in at least two cases:

1. In a cancellation handler for a pthread_join() it is nearly essen? tial to have a pthread_detach() function in order to detach the thread on which pthread_join() was waiting. Without it, it would be necessary to have the handler do another pthread_join() to attempt to detach the thread, which would both delay the cancellation pro? cessing for an unbounded period and introduce a new call to pthread_join(), which might itself need a cancellation handler. A dynamic detach is nearly essential in this case.
2. In order to detach the ``initial thread'' (as may be desirable in processes that set up server threads).

If an implementation detects that the value specified by the thread ar? gument to pthread_detach() does not refer to a joinable thread, it is recommended that the function should fail and report an [EINVAL] error.

If an implementation detects use of a thread ID after the end of its lifetime, it is recommended that the function should fail and report an [ESRCH] error.

FUTURE DIRECTIONS

None.

SEE ALSO

pthread_join()

The Base Definitions volume of POSIX.1?2017, <pthread.h>

COPYRIGHT