



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_setschedprio.3p' command

\$ man pthread_setschedprio.3p

PTHREAD_SETSCHEDPRIO(3P) POSIX Programmer's Manual PTHREAD_SETSCHEDPRIO(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_setschedprio ? dynamic thread scheduling parameters access (RE?
ALTIME THREADS)

SYNOPSIS

```
#include <pthread.h>  
  
int pthread_setschedprio(pthread_t thread, int prio);
```

DESCRIPTION

The pthread_setschedprio() function shall set the scheduling priority for the thread whose thread ID is given by thread to the value given by prio. See Scheduling Policies for a description on how this function call affects the ordering of the thread in the thread list for its new priority.

If the pthread_setschedprio() function fails, the scheduling priority of the target thread shall not be changed.

RETURN VALUE

If successful, the pthread_setschedprio() function shall return zero; otherwise, an error number shall be returned to indicate the error.

ERRORS

The `pthread_setschedprio()` function may fail if:

EINVAL The value of `prio` is invalid for the scheduling policy of the specified thread.

EPERM The caller does not have appropriate privileges to set the scheduling priority of the specified thread.

The `pthread_setschedprio()` function shall not return an error code of `[EINTR]`.

The following sections are informative.

EXAMPLES

None.

APPLICATION USAGE

None.

RATIONALE

The `pthread_setschedprio()` function provides a way for an application to temporarily raise its priority and then lower it again, without having the undesired side-effect of yielding to other threads of the same priority. This is necessary if the application is to implement its own strategies for bounding priority inversion, such as priority inheritance or priority ceilings. This capability is especially important if the implementation does not support the Thread Priority Protection or Thread Priority Inheritance options, but even if those options are supported it is needed if the application is to bound priority inheritance for other resources, such as semaphores.

The standard developers considered that while it might be preferable conceptually to solve this problem by modifying the specification of `pthread_setschedparam()`, it was too late to make such a change, as there may be implementations that would need to be changed. Therefore, this new function was introduced.

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

Scheduling Policies, `pthread_getschedparam()`

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

COPYRIGHT

Portions of this text are reprinted and reproduced in electronic form from IEEE Std 1003.1-2017, Standard for Information Technology -- Portable Operating System Interface (POSIX), The Open Group Base Specifications Issue 7, 2018 Edition, Copyright (C) 2018 by the Institute of Electrical and Electronics Engineers, Inc and The Open Group. In the event of any discrepancy between this version and the original IEEE and The Open Group Standard, the original IEEE and The Open Group Standard is the referee document. The original Standard can be obtained online at <http://www.opengroup.org/unix/online.html>.

Any typographical or formatting errors that appear in this page are most likely to have been introduced during the conversion of the source files to man page format. To report such errors, see https://www.kernel.org/doc/man-pages/reporting_bugs.html.

IEEE/The Open Group

2017

PTHREAD_SETSCHEDPRIO(3P)