Search online pages

NICE(2) Linux Programmer's Manual NICE(2)

NAME top

nice - change process priority

SYNOPSIS top

```
#include <unistd.h>

int nice(int inc);
```

Feature Test Macro Requirements for glibc (see feature_test_macros(7)):

```
nice(): _XOPEN_SOURCE
    || /* Since glibc 2.19: */ _DEFAULT_SOURCE
    || /* Glibc versions <= 2.19: */ _BSD_SOURCE || _SVID_SOURCE
```

DESCRIPTION top

nice() adds *inc* to the nice value for the calling thread. (A higher nice value means a low priority.)

The range of the nice value is +19 (low priority) to -20 (high priority). Attempts to set a nice value outside the range are clamped to the range.

Traditionally, only a privileged process could lower the nice value (i.e., set a higher priority). However, since Linux 2.6.12, an unprivileged process can decrease the nice value of a target process that has a suitable **RLIMIT_NICE** soft limit; see getrlimit(2) for details.

RETURN VALUE top

On success, the new nice value is returned (but see NOTES below). On error, -1 is returned, and *errno* is set appropriately.

A successful call can legitimately return -1. To detect an error, set *errno* to 0 before the call, and check whether it is nonzero after **nice**() returns -1.

ERRORS top

EPERM The calling process attempted to increase its priority by supplying a negative *inc* but has insufficient privileges. Under Linux, the **CAP_SYS_NICE** capability is required. (But see the discussion of the **RLIMIT_NICE** resource limit in setrlimit(2).)

CONFORMING TO top

POSIX.1-2001, POSIX.1-2008, SVr4, 4.3BSD. However, the raw system call and (g)libc (earlier than glibc 2.2.4) return value is nonstandard, see below.

NOTES top

For further details on the nice value, see sched(7).

Note: the addition of the "autogroup" feature in Linux 2.6.38 means that the nice value no longer has its traditional effect in many circumstances. For details, see sched(7).

C library/kernel differences

POSIX.1 specifies that **nice**() should return the new nice value. However, the raw Linux system call returns 0 on success. Likewise, the **nice**() wrapper function provided in glibc 2.2.3 and earlier returns 0 on success.

Since glibc 2.2.4, the **nice**() wrapper function provided by glibc provides conformance to POSIX.1 by calling getpriority(2) to obtain the new nice value, which is then returned to the caller.

SEE ALSO top

nice(1), renice(1), fork(2), getpriority(2), getrlimit(2), setpriority(2), capabilities(7), sched(7)

COLOPHON top

This page is part of release 4.10 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 https://www.kernel.org/doc/man-pages/.

Linux **2016-12-12** **NICE(2)**

Copyright and license for this manual page

HTML rendering created 2017-03-13 by Michael Kerrisk, author of *The Linux Programming Interface*, maintainer of the Linux *man-pages* project.

For details of in-depth **Linux/UNIX system programming training courses** that I teach, look here.

Hosting by jambit GmbH.