

## IEEE Standards Interpretation for IEEE Std 1003.1™-2001 IEEE Standard Standard for Information Technology -- Portable Operating System Interface (POSIX®)

Copyright © 2006 by the Institute of Electrical and Electronics Engineers, Inc. 3 Park Avenue New York, New York 10016-5997 USA All Rights Reserved.

Interpretations are issued to explain and clarify the intent of a standard and **do not** constitute an alteration to the original standard. In addition, interpretations are not intended to supply consulting information. Permission is hereby granted to download and print one copy of this document. Individuals seeking permission to reproduce and/or distribute this document in its entirety or portions of this document must contact the IEEE Standards Department for the appropriate license. Use of the information contained in this document is at your own risk.

IEEE Standards Department Copyrights and Permissions 445 Hoes Lane, Piscataway, New Jersey 08855-1331, USA

### Interpretation Request #61

**Topic:** sched\_setscheduler() **Relevant Sections:** XSH sched\_setscheduler() Page: 1615 Line: 50182,50190 Section: waitid()

In the specification of sched\_setparam(), there is the statement:

The target process, whether it is running or not running, shall be moved to the tail of the thread list for its priority.

However, there is no analogous statement in the specification of sched\_setscheduler(). Is this an oversight -- should there be such a specification? Or failing this, should there be a rationale statement about why there is a difference between sched\_setparam() and sched\_setscheduler()?

On the one implementation that I have tested (Linux), sched\_setscheduler() does indeed behave like sched\_setparam(), moving the target process to the tail of the queue, even if its scheduling priority (and policy) are not actually changed by the call.

However, I note that the man pages for FreeBSD 5.1, Tru64 5.1, and Solaris 8 all mirror the SUSv3 specification. For example, from the Solaris 8 sched\_setparam() man page:

The target process, whether it is running or not running, resumes execution after all other runnable processes of equal or greater priority have been scheduled to run.

On these systems, the sched\_setparam() man page describes the above behaviour, while the sched\_setscheduler() man page is silent on this point. This makes me wonder about the behavior on these systems, and whether I may be incorrect to suggest

amending the sched\_setscheduler() specification to say that the target process should be moved to the tail of the queue. If that is so, perhaps it would be useful to add text to sched\_setscheduler() to explicitly note that this point is unspecified (and perhaps why).

**Interpretation Response #61**

The standard does not speak to this issue, and as such no conformance distinction can be made between alternative implementations based on this. This is being referred to the sponsor.

**Rationale for Interpretation**

None.