

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 #104

Topic: nohup standard input **Relevant Sections:** XCU nohup

Users of the popular security suite OpenSSH have been reporting a problem with “nohup” that suggests a correction to the POSIX specification for “nohup”.

Here’s the problem. The user logs in to a host via ssh, and types the command “nohup make &” at the shell prompt, and then logs out. Unfortunately the ssh session does not terminate: it hangs, useless, until the “make” finishes.

The problem is that “make” still has standard input open. Standard input won’t be used, but ssh doesn’t know that, and it keeps the connection open so long as any process has access to the input file description.

A simple improvement is to have “nohup” redirect standard input from /dev/null if standard input is a terminal. GNU nohup now does this. Another possible improvement would be to redirect standard input from an invalid file (e.g., “/”) so that all reads fail.

Unfortunately, POSIX currently prohibits these improvements. This is leading to practical problems in practice. There is no real utility to having nohup hold on to standard input if it is a terminal, so the POSIX requirement is counterproductive.

The standard should be changed to allow these improvements. At the very least, the problem should be documented in the standard, so that users are warned about it.

Append the following after XCU page 674 line 26132:

If the standard input is a terminal, the file that the standard input of the utility refers to is implementation defined.

Change XCU page 677 line 26215 from: `nohup sh -c 'complex-command-line'`

to: `nohup sh -c 'complex-command-line' </dev/null`

Append the following after XCU page 677 line 26126:

Historical versions of `nohup` did not affect standard input, but that causes problems in the common scenario where the user logs into a system, types the command “`nohup make &`” at the prompt, and then logs out. If standard input is not affected by `nohup`, the login session may not terminate for quite some time, since standard input remains open until “`make`” exits. To avoid this problem the current standard allows implementations to redirect standard input if it is a terminal. Since the behavior is implementation defined, portable programs that may run into the problem should redirect standard input themselves. For example, instead of “`nohup make &`” an application can invoke “`nohup make </dev/null &`”.

Interpretation Response

The standards states the requirements for `nohup` and conforming implementations must conform to this. However, concerns have been raised about this which are being referred to the sponsor.

Rationale for Interpretation

None.