

IEEE Standards Interpretation for IEEE Std 1003.1™-1990 IEEE Standard for Information Technology--Portable Operating System Interfaces (POSIX®)

Copyright ©1996 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 #69

Topic: file access control **Relevant Sections:** 2.3.2, 5.6.3.2

POSIX.1 says, in 5.6.3.2 (page 105, lines 766-768): If the process has appropriate privileges, an implementation may indicate success for X_OK even if none of the execute file permission bits are set. The corresponding assertion in IEEE Std 2003.1-1992 says: 24(C) If the implementation provides a method for associating with a process the appropriate privilege to override the file access control mechanism: When the process has the appropriate privileges to override the file access control mechanism, then a call to access(path, amode) will succeed when amode is set to X_OK and any of the execute bits are set or when the file is a directory. Is the assertion correct in making a distinction between directories and other file types? My reading of POSIX.1 and POSIX.1b is that no such distinction is made in the description of access() and no such distinction is required by descriptions of file access control or of directories elsewhere in the standards. I feel that the words "or when the file is a directory" should not be in the assertion.

Interpretation Response

The standard clearly states the distinction between directory search permission and execute permission for other file types (see lines 432-438 of 2.3.2), and conforming implementations must conform to this.

Rationale for Interpretation

The sentence preceeding the one from POSIX.1 mentioned in the interpretation requests refers to section 2.3.2. That section says (page 21, lines 432-438): (1) If a process has the appropriate privilege:

(a) If read, write, or directory search permission is requested, access is granted.

(b) If execute permission is requested, access is granted if execute permission is granted to at least one user by the file permission bits or by an alternative access control mechanism; otherwise, access is denied.

The assertion rephrases this behavior in terms of the symbolic constant `X_OK`. The same symbolic constant is used both for execute permission and directory search permission (see Table 2.8 on page 28), which may obscure the correspondence between the assertion and the standard. For a process with appropriate privileges, a call to `access(path, X_OK)` must succeed if `path` refers to a directory, even if none of the execute/search permission bits are set for the directory. The call must also succeed if `path` refers to a file with any execute permission bits set. Implementations may vary in the case where `path` refers to a file and not a directory and there are no execute bits set.