NAME

access − determine accessibility of file

SYNOPSIS

```
#include <sys/types.h>
#include <unistd.h>

#define R_OK    4    /* test for read permission */
#define W_OK    2    /* test for write permission */
#define X_OK    1    /* test for execute (search) permission */
#define F_OK    0    /* test for presence of file */

int access(const char *path, mode_t mode)
```

DESCRIPTION

Access checks the given file *path* for accessibility according to *mode*, which is an inclusive or of the bits **R_OK**, **W_OK** and **X_OK**. Specifying *mode* as **F_OK** (i.e., 0) tests whether the directories leading to the file can be searched and the file exists.

The real user ID and the group access list (including the real group ID) are used in verifying permission, so this call is useful to set-UID programs.

Notice that only access bits are checked. A directory may be indicated as writable by **access**, but an attempt to open it for writing will fail (although files may be created there); a file may look executable, but **execve** will fail unless it is in proper format.

RETURN VALUE

If *path* cannot be found or if any of the desired access modes would not be granted, then a −1 value is returned; otherwise a 0 value is returned.

ERRORS

Access to the file is denied if one or more of the following are true:

[ENOTDIR] A component of the path prefix is not a directory.

[ENAMETOOLONG] The path name exceeds PATH_MAX characters.

[ENOENT] The named file does not exist.

[EACCES] Search permission is denied for a component of the path prefix.

[ELOOP] Too many symbolic links were encountered in translating the pathname. (Minix-vmd)

[EROFS] Write access is requested for a file on a read-only file system.

[EACCES] Permission bits of the file mode do not permit the requested access, or search permission is denied on a component of the path prefix. The owner of a file has permission checked with respect to the ‘‘owner’’ read, write, and execute mode bits, members of the file’s group other than the owner have permission checked with respect to the ‘‘group’’ mode bits, and all others have permissions checked with respect to the ‘‘other’’ mode bits.

[EFAULT] *Path* points outside the process’s allocated address space.

[EIO] An I/O error occurred while reading from or writing to the file system.

SEE ALSO

chmod(2), **stat**(2).