NAME

acl_extended_fd — test for information in the ACL by file descriptor

LIBRARY

Linux Access Control Lists library (libacl, −lacl).

SYNOPSIS

```
#include <sys/types.h>
#include <acl/libacl.h>

int
acl_extended_fd(int fd);
```

DESCRIPTION

The **acl_extended_fd**() function returns `1` if the file identified by the argument *fd* is associated with an extended access ACL. The function returns `0` if the file does not have an extended access ACL.

An extended ACL is an ACL that contains entries other than the three required entries of tag types ACL_USER_OBJ, ACL_GROUP_OBJ and ACL_OTHER. If the result of the**acl_extended_fd**() function for a file object is `0`, then the ACL defines no discretionary access rights other than those already defined by the traditional file permission bits.

Access to the file object may be further restricted by other mechanisms, such as Mandatory Access Control schemes. The *access*(2) system call can be used to check whether a given type of access to a file object would be granted.

RETURN VALUE

If successful, the **acl_extended_fd**() function returns `1` if the file object identified by *fd* has an extended access ACL, and `0` if the file object identified by *fd* does not have an extended access ACL. Otherwise, the value `-1` is returned and the global variable *errno* is set to indicate the error.

ERRORS

If any of the following conditions occur, the **acl_extended_fd**() function returns `-1` and sets *errno* to the corresponding value:

[EBADF]	The *fd* argument is not a valid file descriptor.
[ENOTSUP]	The file system on which the file identified by *fd* is located does not support ACLs, or ACLs are disabled.

STANDARDS

This is a non-portable, Linux specific extension to the ACL manipulation functions defined in IEEE Std 1003.1e draft 17 (“POSIX.1e”, abandoned).

SEE ALSO

access(2), *acl_get_fd*(3), *acl*(5)

AUTHOR

Written by Andreas Gruenbacher <andreas.gruenbacher@gmail.com>.