

NAME

`acl_equiv_mode` — check for an equivalent ACL

LIBRARY

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

SYNOPSIS

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

int
acl_equiv_mode(acl_t acl, mode_t *mode_p);
```

DESCRIPTION

The **`acl_equiv_mode()`** function checks if the ACL pointed to by the argument *acl* contains only the required ACL entries of tag types `ACL_USER_OBJ`, `ACL_GROUP_OBJ`, and `ACL_OTHER`, and contains no permissions other than `ACL_READ`, `ACL_WRITE` or `ACL_EXECUTE`. If the ACL has this form, it can be fully represented with the traditional file permission bits, and is considered equivalent with the traditional file permission bits.

If *acl* is an equivalent ACL and the pointer *mode_p* is not `NULL`, the value pointed to by *mode_p* is set to the value that defines the same owner, group and other permissions as contained in the ACL.

RETURN VALUE

On success, this function returns the value `0` if *acl* is an equivalent ACL, and the value `1` if *acl* is not an equivalent ACL. On error, the value `-1` is returned, and *errno* is set appropriately.

ERRORS

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

[EINVAL] The argument *acl* is not a valid pointer to an ACL.

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

acl_from_mode(3), *acl(5)*

AUTHOR

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