NAME

acl_check — check an ACL for validity

LIBRARY

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

SYNOPSIS

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

int
acl_check(acl_t acl, int *last);
```

DESCRIPTION

The **acl_check**() function checks the ACL referred to by the argument *acl* for validity.

The three required entries ACL_USER_OBJ, ACL_GROUP_OBJ, and ACL_OTHER must exist exactly once in the ACL. If the ACL contains any ACL_USER or ACL_GROUP entries, then an ACL_MASK entry is also required. The ACL may contain at most one ACL_MASK entry.

The user identifiers must be unique among all entries of type ACL_USER. The group identifiers must be unique among all entries of type ACL_GROUP.

If the ACL referred to by *acl* is invalid, **acl_check**() returns a positive error code that indicates which type of error was detected. The following symbolic error codes are defined:

ACL_MULTI_ERROR	The ACL contains multiple entries that have a tag type that may occur at most once.
ACL_DUPLICATE_ERROR	The ACL contains multiple ACL_USER entries with the same user ID, or multiple ACL_GROUP entries with the same group ID.
ACL_MISS_ERROR	A required entry is missing.
ACL_ENTRY_ERROR	The ACL contains an invalid entry tag type.

The **acl_error**() function can be used to translate error codes to text messages.

In addition, if the pointer *last* is not NULL, **acl_check**() assigns the number of the ACL entry at which the error was detected to the value pointed to by *last*. Entries are numbered starting with zero, in the order in which they would be returned by the **acl_get_entry**() function.

RETURN VALUE

If successful, the **acl_check**() function returns 0 if the ACL referred to by *acl* is valid, and a positive error code if the ACL is invalid. Otherwise, a value of -1 is returned and the global variable *errno* is set to indicate the error.

ERRORS

If any of the following conditions occur, the **acl_check**() function returns -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_valid(3), *acl*(5)

AUTHOR

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