NAME

acl — introduction to the POSIX.1e ACL security API

LIBRARY

Standard C Library (libc, –lc)

SYNOPSIS

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

DESCRIPTION

The system permits file systems to export Access Control Lists via the VFS, and provides a library for userland access to and manipulation of these ACLs. Not all file systems provide support for ACLs, and some may require that ACL support be explicitly enabled by the administrator. The library calls include routines to allocate, duplicate, retrieve, set, and validate ACLs associated with file objects.

This implementation of the POSIX.1e library differs from the standard in a number of non-portable ways in order to support the MacOS/Darwin ACL semantic. Where possible, these differences are implemented using the mechanisms provided in the standard for such extensions. Where routines are non-standard, they are suffixed with _np to indicate that they are not portable.

POSIX.1e describes a set of ACL manipulation routines to manage the contents of ACLs, as well as their relationships with files; almost all of these support routines are implemented.

Available functions, sorted by behavior, include:

acl_add_perm()
: This function is described in acl_add_perm(3), and may be used to add permissions to a permission set.

acl_clear_perms()
: This function is described in acl_clear_perms(3), and may be used to clear all permissions from a permission set.

acl_copy_entry()
: This function is described in acl_copy_entry(3), and may be used to copy the contents of an ACL entry.

acl_create_entry()
: This function is described in acl_create_entry(3), and may be used to create an empty entry in an ACL.

acl_delete_entry()
: This function is described in acl_delete_entry(3), and may be used to delete an entry from an ACL.

acl_delete_perm()
: This function is described in acl_delete_perm(3), and may be used to delete permissions from a permset.

acl_dup()
: This function is described in acl_dup(3), and may be used to duplicate an ACL structure.

acl_free()
: This function is described in acl_free(3), and may be used to free userland working ACL storage.

acl_from_text()
: This function is described in acl_from_text(3), and may be used to convert a text-form ACL into working ACL state, if the ACL has POSIX.1e semantics.

acl_get_entry()
This function is described in acl_get_entry(3), and may be used to retrieve a designated ACL entry from an ACL.

acl_get_fd(), **acl_get_fd_np**(), **acl_get_file**(), **acl_get_link_np**()
These functions are described in acl_get(3), and may be used to retrieve ACLs from file system objects.

acl_get_permset()
This function is described in acl_get_permset(3), and may be used to retrieve a permset from an ACL entry.

acl_get_qualifier()
This function is described in acl_get_qualifier(3), and may be used to retrieve the qualifier from an ACL entry.

acl_get_tag_type()
This function is described in acl_get_tag_type(3), and may be used to retrieve the tag type from an ACL entry.

acl_init()
This function is described in acl_init(3), and may be used to allocate a fresh (empty) ACL structure.

acl_set_fd(), **acl_set_fd_np**(), **acl_set_file**(), **acl_set_link_np**()
These functions are described in acl_set(3), and may be used to assign an ACL to a file system object.

acl_set_permset()
This function is described in acl_set_permset(3), and may be used to set the permissions of an ACL entry from a permset.

acl_set_qualifier()
This function is described in acl_set_qualifier(3), and may be used to set the qualifier of an ACL.

acl_set_tag_type()
This function is described in acl_set_tag_type(3), and may be used to set the tag type of an ACL.

acl_to_text()
This function is described in acl_to_text(3), and may be used to generate a text-form of a POSIX.1e semantics ACL.

acl_valid(), **acl_valid_fd_np**(), **acl_valid_file_np**(), **acl_valid_link_np**()
These functions are described in acl_valid(3), and may be used to validate an ACL as correct POSIX.1e-semantics, or as appropriate for a particular file system object regardless of semantics.

The syscalls between the internal interfaces and the public library routines may change over time, and as such are not documented. They are not intended to be called directly without going through the library.

SEE ALSO

ls(1), chmod(1), acl_add_perm(3), acl_clear_perms(3), acl_copy_entry(3), acl_create_entry(3), acl_delete_entry(3), acl_delete_perm(3), acl_dup(3), acl_free(3), acl_from_text(3), acl_get(3), acl_get_permset(3), acl_get_qualifier(3), acl_get_tag_type(3), acl_init(3), acl_set(3), acl_set_permset(3), acl_set_qualifier(3), acl_set_tag_type(3), acl_to_text(3), acl_valid(3), posix1e(3)

UNSUPPORTED FUNCTIONS

`acl_calc_mask`(3), **`acl_delete_def_file`**()

STANDARDS

POSIX.1e assigns security labels to all objects, extending the security functionality described in POSIX.1. These additional labels provide fine-grained discretionary access control, fine-grained capabilities, and labels necessary for mandatory access control. POSIX.2c describes a set of userland utilities for manipulating these labels.

POSIX.1e is described in IEEE POSIX.1e draft 17.

HISTORY

This manpage is closely derived from the FreeBSD manpage by Robert N M Watson

AUTHORS

Michael Smith
Robert N M Watson