

NAME

acl_delete_def_file, **acl_delete_def_link_np**, **acl_delete_fd_np**, **acl_delete_file_np**, **acl_delete_link_np** - delete an ACL from a file

LIBRARY

Standard C Library (libc, -lc)

SYNOPSIS

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

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

int

```
acl_delete_def_file(const char *path_p);
```

int

```
acl_delete_def_link_np(const char *path_p);
```

int

```
acl_delete_fd_np(int filedes, acl_type_t type);
```

int

```
acl_delete_file_np(const char *path_p, acl_type_t type);
```

int

```
acl_delete_link_np(const char *path_p, acl_type_t type);
```

DESCRIPTION

The **acl_delete_def_file()**, **acl_delete_def_link_np()**, **acl_delete_fd_np()**, **acl_delete_file_np()**, and **acl_delete_link_np()** each allow the deletion of an ACL from a file. The **acl_delete_def_file()** function is a POSIX.1e call that deletes the default ACL from a file (normally a directory) by name; the remainder of the calls are non-portable extensions that permit the deletion of arbitrary ACL types from a file/directory either by path name or file descriptor. The **_file()** variations follow a symlink if it occurs in the last segment of the path name; the **_link()** variations operate on the symlink itself.

IMPLEMENTATION NOTES

FreeBSD's support for POSIX.1e interfaces and features is still under development at this time.

RETURN VALUES

Upon successful completion, the value 0 is returned; 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, these functions shall return -1 and set *errno* to the corresponding value:

- | | |
|----------------|---|
| [EACCES] | Search permission is denied for a component of the path prefix, or the object exists and the process does not have appropriate access rights. |
| [EBADF] | The <i>fd</i> argument is not a valid file descriptor. |
| [EINVAL] | The ACL type passed is invalid for this file object. |
| [ENAMETOOLONG] | A component of a pathname exceeded 255 characters, or an entire path name exceeded 1023 characters. |
| [ENOENT] | The named object does not exist, or the <i>path_p</i> argument points to an empty string. |
| [ENOMEM] | Insufficient memory available to fulfill request. |
| [ENOTDIR] | A component of the path prefix is not a directory.

Argument <i>path_p</i> must be a directory, and is not. |
| [EOPNOTSUPP] | The file system does not support ACL deletion. |
| [EPERM] | The process does not have appropriate privilege to perform the operation to delete an ACL. |
| [EROFS] | The file system is read-only. |

SEE ALSO

`acl(3)`, `acl_get(3)`, `acl_set(3)`, `posix1e(3)`

STANDARDS

POSIX.1e is described in IEEE POSIX.1e draft 17. Discussion of the draft continues on the cross-platform POSIX.1e implementation mailing list. To join this list, see the FreeBSD POSIX.1e implementation page for more information.

HISTORY

POSIX.1e support was introduced in FreeBSD 4.0, and development continues.

AUTHORS

Robert N M Watson