**NAME**

acl, facl – get or set a file's Access Control List (ACL)

**SYNOPSIS**

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

**int acl(char *pathp, int cmd, int nentries, aclent_t *aclbufp)**

**int facl(int fildes, int cmd, int nentries, aclent_t *aclbufp)**

**DESCRIPTION**

**acl()** and **facl()** get or set the ACL of a file whose name is given by *pathp* or referenced by the open file descriptor *fildes. Nentries* specifies how many ACL entries fit into buffer *aclbufp.* **acl()** is used to manipulate ACLs on file system objects.

The following three values for *cmd* are available.

SETACL

*nentries* ACL entries, specified in buffer *aclbufp,* are stored in the file's ACL. This command can only be executed by a process that has an effective user ID equal to the owner of the file. All directories in the path name must be searchable.

GETACL

Buffer is filled with the file's ACL entries. Read access to the file is not required, but all directories in the path name must be searchable.

GETACLCNT

The number of entries in the file's ACL is returned. Read access to the file is not required, but all directories in the path name must be searchable.

**acl()** will fail if one or more of the following is true:

**[EACCESS]**
The caller does not have access to a component of the pathname.

**[EINVAL]**
*cmd* is not GETACL, SETACL, or GETACLCNT.

**[EINVAL]**
*cmd* is SETACL and *nentries* is less than three.

**[EINVAL]**
*cmd* is SETACL and the ACL specified in *aclbufp* is not valid.

**[EIO]** A disk i/o error has occurred while storing or retrieving the ACL.

**[EPERM]**
*cmd* is SETACL and the effective user ID of the caller does not match the owner of the file.

**[ENOENT]**
A component of the path does not exist.

**[ENOSPC]**
*cmd* is GETACL and *nentries* is less than the number of entries in the file's ACL.

**[ENOSPC]**

*cmd* is SETACL and there is insufficient space in the file system to store the ACL.

**[ENOTDIR]**
A component of the path sepcified by *pathp* is not a directory.

**[ENOTDIR]**
*cmd* is SETACL and an attempt is made to set a default ACL on a file type other than a directory.

**[ENOSYS]**
*cmd* is SETACL and the file specified by *pathp* resides on a file system that does not support ACLs.

**[EROFS]**
*cmd* is SETACL and the file specified by *pathp* resides on a file system that is mounted read-only.

**[EFAULT]**
*pathp* or *aclbufp* points to an illegal address.

## DIAGNOSTICS

Upon successful completion, if *cmd* is SETACL, a value of 0 is returned. If *cmd* is GETACL or GETACLCNT, the number of ACL entries is returned. Otherwise, a value of -1 is returned and *errno* is set to indicate the error.

## ERRORS

**ENOSYS** **acl( )** is not supported by this implementation.

## SEE ALSO

**getfacl**(1), **getfacl**(1), **aclcheck**(3), **aclsort**(3)