

**NAME**

`acl_init` — initialize ACL working storage

**LIBRARY**

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

**SYNOPSIS**

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

acl_t
acl_init(int count);
```

**DESCRIPTION**

The **`acl_init()`** function allocates and initializes the working storage for an ACL of at least *count* ACL entries. The ACL created initially contains no ACL entries. A pointer to the working storage is returned.

This function may cause memory to be allocated. The caller should free any releasable memory, when the new ACL is no longer required, by calling `acl_free(3)` with the *(void\*)acl\_t* returned by **`acl_init()`** as an argument.

**RETURN VALUE**

On success, this function returns a pointer to the working storage. On error, a value of *(acl\_t) NULL* is returned, and *errno* is set appropriately.

**ERRORS**

If any of the following conditions occur, the **`acl_init()`** function returns a value of *(acl\_t) NULL* and sets *errno* to the corresponding value:

|          |                                                                                                                                       |
|----------|---------------------------------------------------------------------------------------------------------------------------------------|
| [EINVAL] | The value of <i>count</i> is less than zero.                                                                                          |
| [ENOMEM] | The <i>acl_t</i> to be returned requires more memory than is allowed by the hardware or system-imposed memory management constraints. |

**STANDARDS**

IEEE Std 1003.1e draft 17 (“POSIX.1e”, abandoned)

**SEE ALSO**

`acl_get_file(3)`, `acl_free(3)`, `acl(5)`

**AUTHOR**

Derived from the FreeBSD manual pages written by Robert N M Watson <rwatson@FreeBSD.org>, and adapted for Linux by Andreas Gruenbacher <andreas.gruenbacher@gmail.com>.