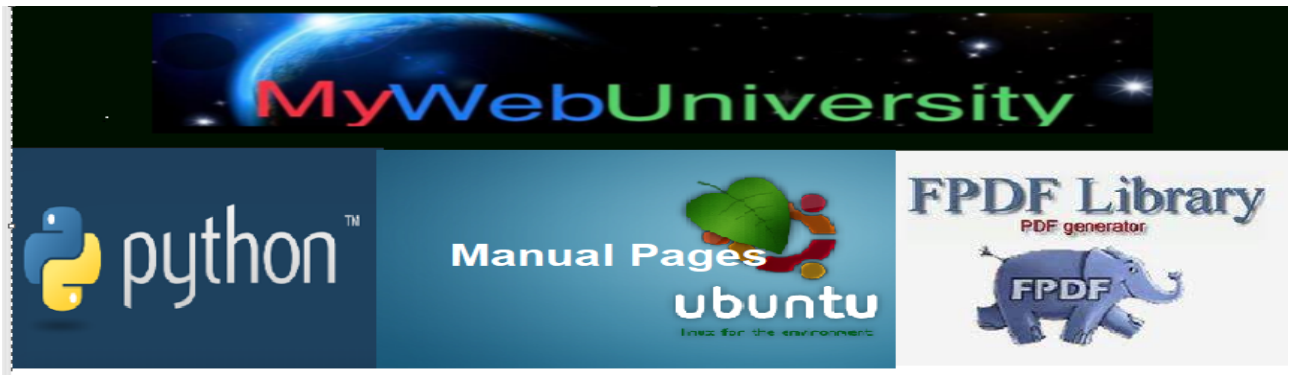

Full credit is given to the above companies including the OS that this PDF file was generated!

Linux Ubuntu 26.04 Manual Pages on command 'euidaccess.3'

$ man euidaccess.3

euidaccess(3) Library Functions Manual euidaccess(3)

NAME

euidaccess, eaccess - check effective user's permissions for a file

LIBRARY

Standard C library (libc, -lc)

SYNOPSIS

```
#define _GNU_SOURCE         /* See feature_test_macros(7) */
#include <unistd.h>
int euidaccess(const char *path, int mode);
int eaccess(const char *path, int mode);
```

DESCRIPTION

Like access(2), euidaccess() checks permissions and existence of the file identified by its argument path. However, whereas access(2) performs checks using the real user and group identifiers of the process, euidaccess() uses the effective identifiers.

mode is a mask consisting of one or more of R_OK, W_OK, X_OK, and F_OK, with the same meanings as for access(2).

eaccess() is a synonym for euidaccess(), provided for compatibility with some other systems.

RETURN VALUE

On success (all requested permissions granted), zero is returned. On error (at least one bit in mode asked for a permission that is denied, or some other error occurred), -1 is returned, and errno is set to indicate the error.

ERRORS

As for access(2).

ATTRIBUTES

For an explanation of the terms used in this section, see attributes(7).

Interface	Attribute	Value
euidaccess(), eaccess()	Thread safety	MT-Safe

VERSIONS

Some other systems have an eaccess() function.

STANDARDS

None.

HISTORY

eaccess()

glibc 2.4.

NOTES

Warning: Using this function to check a process's permissions on a file before performing some operation based on that information leads to race conditions: the file permissions may change between the two steps. Generally, it is safer just to attempt the desired operation and handle any permission error that occurs.

This function always dereferences symbolic links. If you need to check the permissions on a symbolic link, use faccessat(2) with the flags AT_EACCESS and AT_SYMLINK_NOFOLLOW.

SEE ALSO

access(2), chmod(2), chown(2), faccessat(2), open(2), setgid(2), setuid(2), stat(2), credentials(7), path_resolution(7)