

**NAME**

**revoke** - revoke file access

**LIBRARY**

Standard C Library (libc, -lc)

**SYNOPSIS**

```
#include <unistd.h>
```

*int*

```
revoke(const char *path);
```

**DESCRIPTION**

The **revoke()** system call invalidates all current open file descriptors in the system for the file named by *path*. Subsequent operations on any such descriptors fail, with the exceptions that a **read()** from a character device file which has been revoked returns a count of zero (end of file), and a **close()** system call will succeed. If the file is a special file for a device which is open, the device close function is called as if all open references to the file had been closed using a special close method which does not block.

Access to a file may be revoked only by its owner or the super user. The **revoke()** system call is currently supported only for block and character special device files. It is normally used to prepare a terminal device for a new login session, preventing any access by a previous user of the terminal.

**RETURN VALUES**

The **revoke()** function returns the value 0 if successful; otherwise the value -1 is returned and the global variable *errno* is set to indicate the error.

**ERRORS**

Access to the named file is revoked unless one of the following:

[ENOTDIR]           A component of the path prefix is not a directory.

[ENAMETOOLONG]       A component of a pathname exceeded 255 characters, or an entire path name exceeded 1024 characters.

[ENOENT]            The named file or a component of the path name does not exist.

[EACCES]            Search permission is denied for a component of the path prefix.

- [ELOOP] Too many symbolic links were encountered in translating the pathname.
- [EFAULT] The *path* argument points outside the process's allocated address space.
- [EINVAL] The implementation does not support the **revoke()** operation on the named file.
- [EPERM] The caller is neither the owner of the file nor the super user.

**SEE ALSO**

revoke(1), close(2)

**HISTORY**

The **revoke()** system call first appeared in 4.3BSD-Reno.

**BUGS**

The non-blocking close method is only correctly implemented for terminal devices.