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

## *Red Hat Enterprise Linux Release 9.2 Manual Pages on 'fattach.3p' command*

***$ man fattach.3p***

FATTACH(3P) POSIX Programmer's Manual FATTACH(3P)

PROLOG

This manual page is part of the POSIX Programmer's Manual. The Linux implementation of this interface may differ (consult the corresponding Linux manual page for details of Linux behavior), or the interface may not be implemented on Linux.

NAME

fattach ? attach a STREAMS-based file descriptor to a file in the file system name space (STREAMS)

SYNOPSIS

```
#include <stropts.h>

int fattach(int fildes, const char *path);
```

DESCRIPTION

The fattach() function shall attach a STREAMS-based file descriptor to a file, effectively associating a pathname with fildes. The applica? tion shall ensure that the fildes argument is a valid open file de? scriptor associated with a STREAMS file. The path argument points to a pathname of an existing file. The application shall have appropriate privileges or be the owner of the file named by path and have write permission. A successful call to fattach() shall cause all pathnames that name the file named by path to name the STREAMS file associated with fildes, until the STREAMS file is detached from the file. A STREAMS file can be attached to more than one file and can have several

pathnames associated with it.

The attributes of the named STREAMS file shall be initialized as fol? lows: the permissions, user ID, group ID, and times are set to those of the file named by path, the number of links is set to 1, and the size and device identifier are set to those of the STREAMS file associated with fildes. If any attributes of the named STREAMS file are subse? quently changed (for example, by chmod()), neither the attributes of the underlying file nor the attributes of the STREAMS file to which fildes refers shall be affected.

File descriptors referring to the underlying file, opened prior to an fattach() call, shall continue to refer to the underlying file.

RETURN VALUE

Upon successful completion, fattach() shall return 0. Otherwise, -1 shall be returned and errno set to indicate the error.

ERRORS

The fattach() function shall fail if:

EACCES Search permission is denied for a component of the path prefix, or the process is the owner of path but does not have write per? missions on the file named by path.

EBADF The fildes argument is not a valid open file descriptor.

EBUSY The file named by path is currently a mount point or has a STREAMS file attached to it.

ELOOP A loop exists in symbolic links encountered during resolution of the path argument.

ENAMETOOLONG

The length of a component of a pathname is longer than {NAME_MAX}.

ENOENT A component of path does not name an existing file or path is an empty string.

ENOTDIR

A component of the path prefix names an existing file that is neither a directory nor a symbolic link to a directory, or the path argument contains at least one non-<slash> character and

ends with one or more trailing <slash> characters.

EPERM The effective user ID of the process is not the owner of the file named by path and the process does not have appropriate privileges.

The fattach() function may fail if:

EINVAL The fildes argument does not refer to a STREAMS file.

ELOOP More than {SYMLOOP_MAX} symbolic links were encountered during resolution of the path argument.

ENAMETOOLONG

The length of a pathname exceeds {PATH_MAX}, or pathname resolu? tion of a symbolic link produced an intermediate result with a length that exceeds {PATH_MAX}.

EXDEV A link to a file on another file system was attempted.

The following sections are informative.

## EXAMPLES

### Attaching a File Descriptor to a File

In the following example, fd refers to an open STREAMS file. The call to fattach() associates this STREAM with the file /tmp/named-STREAM, such that any future calls to open /tmp/named-STREAM, prior to breaking the attachment via a call to fdetach(), will instead create a new file handle referring to the STREAMS file associated with fd.

```
#include <stropts.h>
...
    int fd;
    char *pathname = "/tmp/named-STREAM";
    int ret;
    ret = fattach(fd, pathname);
```

## APPLICATION USAGE

The fattach() function behaves similarly to the traditional mount() function in the way a file is temporarily replaced by the root direc? tory of the mounted file system. In the case of fattach(), the replaced file need not be a directory and the replacing file is a STREAMS file.

## RATIONALE

The file attributes of a file which has been the subject of an fat?
tach() call are specifically set because of an artifact of the original
implementation. The internal mechanism was the same as for the mount()
function. Since mount() is typically only applied to directories, the
effects when applied to a regular file are a little surprising, espe?
cially as regards the link count which rigidly remains one, even if
there were several links originally and despite the fact that all orig?
inal links refer to the STREAM as long as the fattach() remains in ef?
fect.

FUTURE DIRECTIONS

The fattach() function may be removed in a future version.

SEE ALSO

fdetach(), isastream()

The Base Definitions volume of POSIX.1?2017, <stropts.h>