



Red Hat Enterprise Linux Release 9.2 Manual Pages on 'fchown.3p' command

\$ man fchown.3p

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

fchown ? change owner and group of a file

SYNOPSIS

```
#include <unistd.h>

int fchown(int fildes, uid_t owner, gid_t group);
```

DESCRIPTION

The fchown() function shall be equivalent to chown() except that the file whose owner and group are changed is specified by the file descriptor fildes.

RETURN VALUE

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

ERRORS

The fchown() function shall fail if:

EBADF The fildes argument is not an open file descriptor.

EPERM The effective user ID does not match the owner of the file or the process does not have appropriate privileges and

`_POSIX_CHOWN_RESTRICTED` indicates that such privilege is required.

`EROFS` The file referred to by `fildev` resides on a read-only file system.

The `fchown()` function may fail if:

`EINVAL` The owner or group ID is not a value supported by the implementation. The `fildev` argument refers to a pipe or socket or an `fattach()`-ed `STREAM` and the implementation disallows execution of `fchown()` on a pipe.

`EIO` A physical I/O error has occurred.

`EINTR` The `fchown()` function was interrupted by a signal which was caught.

The following sections are informative.

EXAMPLES

Changing the Current Owner of a File

The following example shows how to change the owner of a file named `/home/cnd/mod1` to ```jones`" and the group to ```cnd`".

The numeric value for the user ID is obtained by extracting the user ID from the user database entry associated with ```jones`". Similarly, the numeric value for the group ID is obtained by extracting the group ID from the group database entry associated with ```cnd`". This example assumes the calling program has appropriate privileges.

```
#include <sys/types.h>
#include <unistd.h>
#include <fcntl.h>
#include <pwd.h>
#include <grp.h>

struct passwd *pwd;
struct group *grp;

int    fildev;

...

fildev = open("/home/cnd/mod1", O_RDWR);
pwd = getpwnam("jones");
```

```
grp = getgrnam("cnd");  
fchown(fildes, pwd->pw_uid, grp->gr_gid);
```

APPLICATION USAGE

None.

RATIONALE

None.

FUTURE DIRECTIONS

None.

SEE ALSO

chown()

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

COPYRIGHT

Portions of this text are reprinted and reproduced in electronic form from IEEE Std 1003.1-2017, Standard for Information Technology -- Portable Operating System Interface (POSIX), The Open Group Base Specifications Issue 7, 2018 Edition, Copyright (C) 2018 by the Institute of Electrical and Electronics Engineers, Inc and The Open Group. In the event of any discrepancy between this version and the original IEEE and The Open Group Standard, the original IEEE and The Open Group Standard is the referee document. The original Standard can be obtained online at <http://www.opengroup.org/unix/online.html> .

Any typographical or formatting errors that appear in this page are most likely to have been introduced during the conversion of the source files to man page format. To report such errors, see https://www.kernel.org/doc/man-pages/reporting_bugs.html .

IEEE/The Open Group

2017

FCHOWN(3P)