lchown - Man Page

change the owner and group of a symbolic link

Prolog

Эта страница руководства является частью Руководства программиста POSIX. Реализация этого интерфейса в Linux может отличаться (обратитесь к соответствующей странице руководства Linux для получения подробной информации о поведении Linux), или интерфейс может быть не реализован в Linux.

Краткий обзор

#include <unistd.h>

int lchown(const char *path, uid_t owner, gid_t group);

Описание

Функция *lchown*() должна быть эквивалентна *chown*(), за исключением случая, когда именованный файл является символической ссылкой. В этом случае *lchown*() меняет владельца самого файла символической ссылки, в то время *как chown*() изменяет владельца файла или каталога, на который ссылается символическая ссылка.

Возвращаемое значение

После успешного завершения *lchown*() возвращает 0. В противном случае он должен вернуть -1 и установить *errno*, чтобы указать на ошибку.

lchown - Man Page

Функция *lchown*() завершится ошибкой, если:

EACCES

Разрешение на поиск отклоняется для компонента префикса пути *path*.

EINVAL

The owner or group ID is not a value supported by the implementation.

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 and the last pathname component names an existing file that is neither a directory nor a symbolic link to a directory.

EPERM

lchown - Man Page

EROFS

> The file resides on a read-only file system.

The *lchown*() function may fail if:

EIO

An I/O error occurred while reading or writing to the file
system.

EINTR

> A signal was caught during execution of the function.

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
> resolution of a symbolic link produced an intermediate result
> with a length that exceeds {PATH_MAX}.

The following sections are informative.

Examples

Changing the Current Owner of a File

The following example shows how to change the ownership of the symbolic
link named **/modules/pass1** to the user ID associated with "jones" and
the group ID associated with "cnd".

lchown - Man Page

```
getgrnam() function.

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

    struct passwd *pwd;
    struct group  *grp;
    char          *path = "/modules/pass1";
    ...
    pwd = getpwnam("jones");
    grp = getgrnam("cnd");
    lchown(path, pwd->pw_uid, grp->gr_gid);
```

Application Usage

On implementations which support symbolic links as directory entries
rather than files, *lchown*() may fail.

Rationale

None.

Future Directions

None.

See Also

chown(), symlink()

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

lchown - Man Page

Referenced By

chown(3p), symlink(3p), unistd.h(0p).

2017 IEEE/The Open Group POSIX Programmer's Manual

lchown - Man Page

Главная   Блог   О нас