

dup - Man Page

дублировать открытый файловый дескриптор

Пролог

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.

Краткое описание

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

```
int dup(int fildes);  
int dup2(int fildes, int fildes2);
```

Описание

Функция *dup()* предоставляет альтернативный интерфейс службе, предоставляемой *fcntl()* с помощью команды *F_DUPFD*. Вызов *dup(fildes)* должен быть эквивалентен:

```
fcntl(fildes, F_DUPFD, 0);
```

Функция *dup2()* должна заставить файловый дескриптор *fildes2* ссылаться на то же описание открытого файла, что и файловый дескриптор *fildes*, и совместно использовать любые блокировки и возвращать *fildes2*. Если *fildes2* уже является допустимым открытым файловым дескриптором, он должен быть закрыт первым, если *fildes* не равен *fildes2*, и в этом случае *dup2()* вернет *fildes2*, не закрывая его. Если операция *close* не закрывает *fildes2*, *dup2()* возвращает *-1* без изменения описания открытого файла, на который ссылается *fildes2*. Если *fildes* не является допустимым файловым дескриптором, *dup2()* возвращает *-1* и не закрывает *fildes2*. If *fildes2* меньше 0 или больше или равно *{OPEN_MAX}*, *dup2()* возвращает *-1* с *errno*, установленным в *[EBADF]*.

После успешного завершения, если *fildes* не равно *fildes2*, флаг *FD_CLOEXEC*, связанный с *fildes2*, должен быть очищен. Если *fildes* равно *fildes2*, флаг *FD_CLOEXEC*, связанный с *fildes2*, не должен быть изменен.

Если *fildes* ссылается на типизированный объект памяти, результат функции *dup2()* не указан.

Возвращаемое значение

После успешного завершения должно быть возвращено неотрицательное целое число, а именно файловый дескриптор; в противном случае должно быть

dup - Man Page

Ошибки

Функция `dup()` завершится ошибкой, если:

EBADF

Аргумент *fildes* не является допустимым дескриптором открытого файла.

EMFILE

Все файловые дескрипторы, доступные процессу, в настоящее время открыты.

Функция `dup2()` завершится ошибкой, если:

EBADF

Аргумент *fildes* не является допустимым дескриптором открытого файла, или аргумент *fildes2* отрицателен или больше или равен `{OPEN_MAX}` .

EINTR

Функция `dup2()` была прервана сигналом.

Функция `dup2()` может завершиться ошибкой, если:

EIO При попытке закрыть *fildes2* произошла ошибка ввода-вывода.

Следующие разделы являются информативными.

Примеры

Перенаправление стандартного вывода в файл *S*

Следующий пример закрывает стандартный вывод для текущих процессов, переназначает стандартный вывод для перехода к файлу, на который ссылается *pfd*, и закрывает исходный файловый дескриптор для очистки.

```
#include <unistd.h>
...
int pfd;
...
close(1);
dup(pfd);
закрыть (pfd);
...
```

Перенаправление сообщений об ошибках

Следующий пример перенаправляет сообщения из *stderr* в *stdout*.

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

dup - Man Page

Использование приложений

Реализации могут использовать файловые дескрипторы, которые должны быть унаследованы в дочерние процессы, чтобы дочерний процесс оставался соответствующим, например, для целей каталога сообщений или трассировки. Поэтому приложение, которое вызывает `dup2()` с произвольным целым числом для `fildes2`, рискует несоответствующим поведением, а `dup2()` может использоваться только для переносимости значений файловых дескрипторов, полученных приложением с помощью явных действий, или для трех файловых дескрипторов, соответствующих стандартным файловым потокам. Чтобы избежать состояния гонки утечки непреднамеренного файлового дескриптора в дочерний процесс, приложение должно рассмотреть возможность открытия всех файловых дескрипторов с установленным битом `FD_CLOEXEC`, если файловый дескриптор не предназначен для наследования через `exec`.

Обоснование

Функция `dup()` является избыточной. Его услуги также предоставляются функцией `fcntl()`. Он был включен в этот том POSIX.1-2017 главным образом по историческим причинам, поскольку многие существующие приложения используют его. С другой стороны, функция `dup2()` предоставляет уникальные услуги, поскольку никакой другой интерфейс не может атомарно заменить существующий файловый дескриптор.

Функция `dup2()` не помечена как устаревшая, поскольку она представляет типобезопасную версию функциональности, предоставляемой в типобезопасной версии `fcntl()`. Он используется в привязке POSIX Ada.

Функция `dup2()` не предназначена для использования в критических регионах в качестве механизма синхронизации.

В описании [EBADF]случай, когда поля находятся вне диапазона, покрывается данным случаем недопустимости полей. Описания для `fildes` и `fildes2` различны, потому что единственный вид недействительности, который имеет отношение к `fildes2`, – это то, находится ли он вне диапазона; то есть не имеет значения, ссылается ли `fildes2` на открытый файл при вызове `dup2()`.

Будущие направления

Нет.

См. Также

[Закреть\(\)](#), [fcntl\(\)](#), [Открыть\(\)](#)

Базовый том определений POSIX.1-2017, [<unistd.h>](#)

dup - Man Page

Части этого текста перепечатаны и воспроизведены в электронном виде из 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 GroupГруппа. В случае любого несоответствия между этой версией и исходным стандартом IEEE и Open Group исходный стандарт IEEE и Open Group является документом рефери. Оригинальный стандарт можно получить онлайн по адресу <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 .

Referenced By

`close(3p)`, `fstatvfs(3p)`, `open(3p)`, `posix_spawn(3p)`,
`posix_spawn_file_actions_addclose(3p)`,
`posix_spawn_file_actions_adddup2(3p)`, `posix_typed_mem_open(3p)`, `sh(1p)`,
`shm_open(3p)`, `unistd.h(0p)`, `write(3p)`.

2017 IEEE/The Open Group POSIX Programmer's Manual

dup - Man Page

[Главная](#) [Блог](#) [0 нас](#)