

## System Calls

dup(2)

## NAME

dup - duplicate an open file descriptor

## SYNOPSIS

```
#include <unistd.h>

int dup(int fildes);
```

## DESCRIPTION

The dup() function returns a new file descriptor having the following in common with the original open file descriptor fildes:

- o same open file (or pipe)
- o same file pointer (that is, both file descriptors share one file pointer)
- o same access mode (read, write or read/write).

The new file descriptor is set to remain open across exec functions (see fcntl(2)).

The file descriptor returned is the lowest one available.

The dup(fildes) function call is equivalent to:

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

## RETURN VALUES

Upon successful completion, a non-negative integer representing the file descriptor is returned. Otherwise, -1 is returned and errno is set to indicate the error.

## ERRORS

The dup() function will fail if:

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

**EINTR** A signal was caught during the execution of the dup() function.

**EMFILE**

The process has too many open files (see getrlimit(2)).

**ENOLINK**

The fildes argument is on a remote machine and the link to that machine is no longer active.

## ATTRIBUTES

See attributes(5) for descriptions of the following attributes:

ATTRIBUTE TYPE	ATTRIBUTE VALUE
MT-Level	Async-Signal-Safe

## SEE ALSO

close(2), creat(2), exec(2), fcntl(2), getrlimit(2), open(2), pipe(2), dup2(3C), lockf(3C), attributes(5)

SunOS 5.9

Last change: 28 Dec 1996