`ftruncate` - Man Page

truncate a file to a specified length

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.

Synopsis

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

int ftruncate(int *fildes*, off_t *length*);

Description

If *fildes* is not a valid file descriptor open for writing, the *ftruncate*() function shall fail.

If *fildes* refers to a regular file, the *ftruncate*() function shall cause the size of the file to be truncated to *length*. If the size of the file previously exceeded *length*, the extra data shall no longer be available to reads on the file. If the file previously was smaller than this size, *ftruncate*() shall increase the size of the file. If the file size is increased, the extended area shall appear as if it were zero-filled. The value of the seek pointer shall not be modified by a call to *ftruncate*().

Upon successful completion, if *fildes* refers to a regular file, *ftruncate*() shall mark for update the last data modification and last

ftruncate - Man Page

unsuccessful, the file is unaffected.

If the request would cause the file size to exceed the soft file size limit for the process, the request shall fail and the implementation shall generate the SIGXFSZ signal for the thread.

If *fildes* refers to a directory, *ftruncate*() shall fail.

If *fildes* refers to any other file type, except a shared memory object, the result is unspecified.

If *fildes* refers to a shared memory object, *ftruncate*() shall set the size of the shared memory object to *length*.

If the effect of *ftruncate*() is to decrease the size of a memory mapped file or a shared memory object and whole pages beyond the new end were previously mapped, then the whole pages beyond the new end shall be discarded.

References to discarded pages shall result in the generation of a SIGBUS signal.

If the effect of *ftruncate*() is to increase the size of a memory object, it is unspecified whether the contents of any mapped pages between the old end-of-file and the new are flushed to the underlying object.

Return Value

Upon successful completion, *ftruncate*() shall return 0; otherwise, -1 shall be returned and *errno* set to indicate the error.

Errors

ftruncate - Man Page

A signal was caught during execution.

EINVAL

The *length* argument was less than 0.

EFBIG or **EINVAL**

The *length* argument was greater than the maximum file size.

EFBIG

The file is a regular file and *length* is greater than the offset maximum established in the open file description associated with *fildes*.

EIO An I/O error occurred while reading from or writing to a file system.

EBADF or **EINVAL**

The *fildes* argument is not a file descriptor open for writing.

The following sections are informative.

Examples

None.

Application Usage

None.

Rationale

ftruncate - Man Page

Future Directions

None.

See Also

open(), truncate()

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 .

Referenced By

posix_fallocate(3p), posix_typed_mem_open(3p), unistd.h(0p).

ftruncate - Man Page