



Linux Ubuntu 26.04 Manual Pages on command 'utimes.2'

\$ man utimes.2

utime(2) System Calls Manual utime(2)

NAME

utime, utimes - change file last access and modification times

LIBRARY

Standard C library (libc, -lc)

SYNOPSIS

```
#include <utime.h>

int utime(const char *path,
          const struct utimbuf * _Nullable times);

#include <sys/time.h>

int utimes(const char *path,
           const struct timeval times[_Nullable 2]);
```

DESCRIPTION

Note: modern applications may prefer to use the interfaces described in utimensat(2).

The `utime()` system call changes the access and modification times of the inode specified by path to the `actime` and `modtime` fields of `times` respectively. The status change time (`ctime`) will be set to the current time, even if the other time stamps don't actually change.

If `times` is `NULL`, then the access and modification times of the file are set to the current time.

Changing timestamps is permitted when: either the process has appropriate privileges, or the effective user ID equals the user ID of the file, or `times` is `NULL` and the process has write permission for the file.

The `utimbuf` structure is:

```

struct utimbuf {
    time_t actime;    /* access time */
    time_t modtime;   /* modification time */
};

```

The `utime()` system call allows specification of timestamps with a resolution of 1 second.

The `utimes()` system call is similar, but the `times` argument refers to an array rather than a structure. The elements of this array are `timeval` structures, which allow a precision of 1 microsecond for specifying timestamps. The `timeval` structure is:

```

struct timeval {
    long tv_sec;      /* seconds */
    long tv_usec;     /* microseconds */
};

```

`times[0]` specifies the new access time, and `times[1]` specifies the new modification time.

If `times` is `NULL`, then analogously to `utime()`, the access and modification times of the file are set to the current time.

RETURN VALUE

On success, zero is returned. On error, -1 is returned, and `errno` is set to indicate the error.

ERRORS

`EACCES` Search permission is denied for one of the directories in the `path` prefix of `path` (see also `path_resolution(7)`).

`EACCES` `times` is `NULL`, the caller's effective user ID does not match the owner of the file, the caller does not have write access to the file, and the caller is not privileged (Linux: does not have either the `CAP_DAC_OVERRIDE` or the `CAP_FOWNER` capability).

`EFAULT` `path` points to an invalid address.

`ENOENT` `path` does not exist.

`EPERM` `times` is not `NULL`, the caller's effective UID does not match the owner of the file, and the caller is not privileged (Linux: does not have the `CAP_FOWNER` capability).

`EROFS` `path` resides on a read-only filesystem.

STANDARDS

`utime()`

None.

`utimes()`

POSIX.1-2024.

HISTORY

utime()

SVr4, POSIX.1-2001. Obsoleted in POSIX.1-2008. Removed in POSIX.1-2024.

utimes()

4.3BSD, POSIX.1-2001.

NOTES

Linux does not allow changing the timestamps on an immutable file, or setting the timestamps to something other than the current time on an append-only file.

SEE ALSO

chattr(1), touch(1), futimesat(2), stat(2), utimensat(2), futimens(3), futimes(3), inode(7)

Linux man-pages 6.17

2026-02-08

utime(2)