

**NAME**

setxattr, lsetxattr, fsetxattr – set an extended attribute value

**LIBRARY**

Standard C library (*libc*, *-lc*)

**SYNOPSIS**

```
#include <sys/xattr.h>

int setxattr(const char *path, const char *name,
             const void value[.size], size_t size, int flags);
int lsetxattr(const char *path, const char *name,
             const void value[.size], size_t size, int flags);
int fsetxattr(int fd, const char *name,
             const void value[.size], size_t size, int flags);
```

**DESCRIPTION**

Extended attributes are *name:value* pairs associated with inodes (files, directories, symbolic links, etc.). They are extensions to the normal attributes which are associated with all inodes in the system (i.e., the **stat(2)** data). A complete overview of extended attributes concepts can be found in **xattr(7)**.

**setxattr()** sets the *value* of the extended attribute identified by *name* and associated with the given *path* in the filesystem. The *size* argument specifies the size (in bytes) of *value*; a zero-length value is permitted.

**lsetxattr()** is identical to **setxattr()**, except in the case of a symbolic link, where the extended attribute is set on the link itself, not the file that it refers to.

**fsetxattr()** is identical to **setxattr()**, only the extended attribute is set on the open file referred to by *fd* (as returned by **open(2)**) in place of *path*.

An extended attribute name is a null-terminated string. The *name* includes a namespace prefix; there may be several, disjoint namespaces associated with an individual inode. The *value* of an extended attribute is a chunk of arbitrary textual or binary data of specified length.

By default (i.e., *flags* is zero), the extended attribute will be created if it does not exist, or the value will be replaced if the attribute already exists. To modify these semantics, one of the following values can be specified in *flags*:

**XATTR\_CREATE**

Perform a pure create, which fails if the named attribute exists already.

**XATTR\_REPLACE**

Perform a pure replace operation, which fails if the named attribute does not already exist.

**RETURN VALUE**

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

**ERRORS****EDQUOT**

Disk quota limits meant that there is insufficient space remaining to store the extended attribute.

**EEXIST**

**XATTR\_CREATE** was specified, and the attribute exists already.

**ENODATA**

**XATTR\_REPLACE** was specified, and the attribute does not exist.

**ENOSPC**

There is insufficient space remaining to store the extended attribute.

**ENOTSUP**

The namespace prefix of *name* is not valid.

**ENOTSUP**

Extended attributes are not supported by the filesystem, or are disabled,

**EPERM**

The file is marked immutable or append-only. (See **ioctl\_iflags(2)**.)

In addition, the errors documented in **stat(2)** can also occur.

**ERANGE**

The size of *name* or *value* exceeds a filesystem-specific limit.

**VERSIONS**

These system calls have been available since Linux 2.4; glibc support is provided since glibc 2.3.

**STANDARDS**

These system calls are Linux-specific.

**SEE ALSO**

**getfattr(1)**, **setfattr(1)**, **getxattr(2)**, **listxattr(2)**, **open(2)**, **removexattr(2)**, **stat(2)**, **symlink(7)**, **xattr(7)**