NAME

statfs, fstatfs − get file system statistics

SYNOPSIS

#include <sys/vfs.h> /* or <sys/statfs.h> */

int statfs(const char ***path*, struct statfs ***buf***);**
int fstatfs(int *fd***, struct statfs ***buf***);**

DESCRIPTION

The function **statfs**() returns information about a mounted file system. *path* is the pathname of any file within the mounted file system. *buf* is a pointer to a *statfs* structure defined approximately as follows:

```
struct statfs {
   long    f_type;     /* type of file system (see below) */
   long    f_bsize;    /* optimal transfer block size */
   long    f_blocks;   /* total data blocks in file system */
   long    f_bfree;    /* free blocks in fs */
   long    f_bavail;   /* free blocks avail to non-superuser */
   long    f_files;    /* total file nodes in file system */
   long    f_ffree;    /* free file nodes in fs */
   fsid_t  f_fsid;     /* file system id */
   long    f_namelen;  /* maximum length of filenames */
};
```

File system types:

ADFS_SUPER_MAGIC	0xadf5
AFFS_SUPER_MAGIC	0xADFF
BEFS_SUPER_MAGIC	0x42465331
BFS_MAGIC	0x1BADFACE
CIFS_MAGIC_NUMBER	0xFF534D42
CODA_SUPER_MAGIC	0x73757245
COH_SUPER_MAGIC	0x012FF7B7
CRAMFS_MAGIC	0x28cd3d45
DEVFS_SUPER_MAGIC	0x1373
EFS_SUPER_MAGIC	0x00414A53
EXT_SUPER_MAGIC	0x137D
EXT2_OLD_SUPER_MAGIC	0xEF51
EXT2_SUPER_MAGIC	0xEF53
EXT3_SUPER_MAGIC	0xEF53
HFS_SUPER_MAGIC	0x4244
HPFS_SUPER_MAGIC	0xF995E849
HUGETLBFS_MAGIC	0x958458f6
ISOFS_SUPER_MAGIC	0x9660
JFFS2_SUPER_MAGIC	0x72b6
JFS_SUPER_MAGIC	0x3153464a
MINIX_SUPER_MAGIC	0x137F /* orig. minix */
MINIX_SUPER_MAGIC2	0x138F /* 30 char minix */
MINIX2_SUPER_MAGIC	0x2468 /* minix V2 */
MINIX2_SUPER_MAGIC2	0x2478 /* minix V2, 30 char names */
MSDOS_SUPER_MAGIC	0x4d44
NCP_SUPER_MAGIC	0x564c
NFS_SUPER_MAGIC	0x6969
NTFS_SB_MAGIC	0x5346544e
OPENPROM_SUPER_MAGIC	0x9fa1

PROC_SUPER_MAGIC	0x9fa0
QNX4_SUPER_MAGIC	0x002f
REISERFS_SUPER_MAGIC	0x52654973
ROMFS_MAGIC	0x7275
SMB_SUPER_MAGIC	0x517B
SYSV2_SUPER_MAGIC	0x012FF7B6
SYSV4_SUPER_MAGIC	0x012FF7B5
TMPFS_MAGIC	0x01021994
UDF_SUPER_MAGIC	0x15013346
UFS_MAGIC	0x00011954
USBDEVICE_SUPER_MAGIC	0x9fa2
VXFS_SUPER_MAGIC	0xa501FCF5
XENIX_SUPER_MAGIC	0x012FF7B4
XFS_SUPER_MAGIC	0x58465342
_XIAFS_SUPER_MAGIC	0x012FD16D

Nobody knows what *f_fsid* is supposed to contain (but see below).

Fields that are undefined for a particular file system are set to 0. **fstatfs**() returns the same information about an open file referenced by descriptor *fd*.

RETURN VALUE

On success, zero is returned. On error, −1 is returned, and *errno* is set appropriately.

ERRORS

EACCES
(**statfs**()) Search permission is denied for a component of the path prefix of *path*. (See also **path_resolution**(7).)

EBADF
(**fstatfs**()) *fd* is not a valid open file descriptor.

EFAULT
buf or *path* points to an invalid address.

EINTR
This call was interrupted by a signal.

EIO An I/O error occurred while reading from the file system.

ELOOP
(**statfs**()) Too many symbolic links were encountered in translating *path*.

ENAMETOOLONG
(**statfs**()) *path* is too long.

ENOENT
(**statfs**()) The file referred to by *path* does not exist.

ENOMEM
Insufficient kernel memory was available.

ENOSYS
The file system does not support this call.

ENOTDIR
(**statfs**()) A component of the path prefix of *path* is not a directory.

EOVERFLOW
Some values were too large to be represented in the returned struct.

CONFORMING TO

Linux-specific. The Linux **statfs**() was inspired by the 4.4BSD one (but they do not use the same structure).

NOTES

The kernel has system calls **statfs**(), **fstatfs**(), **statfs64**(), and **fstatfs64**() to support this library call.

Some systems only have *<sys/vfs.h>*, other systems also have *<sys/statfs.h>*, where the former includes the latter. So it seems including the former is the best choice.

LSB has deprecated the library calls **statfs**() and **fstatfs**() and tells us to use **statvfs**(2) and **fstatvfs**(2) instead.

The f_fsid field

Solaris, Irix and POSIX have a system call **statvfs**(2) that returns a *struct statvfs* (defined in *<sys/statvfs.h>*) containing an *unsigned long f_fsid*. Linux, SunOS, HP-UX, 4.4BSD have a system call **statfs**() that returns a *struct statfs* (defined in *<sys/vfs.h>*) containing a *fsid_t f_fsid*, where *fsid_t* is defined as *struct { int val[2]; }*. The same holds for FreeBSD, except that it uses the include file *<sys/mount.h>*.

The general idea is that *f_fsid* contains some random stuff such that the pair (*f_fsid*,*ino*) uniquely determines a file. Some OSes use (a variation on) the device number, or the device number combined with the file-system type. Several OSes restrict giving out the *f_fsid* field to the superuser only (and zero it for unprivileged users), because this field is used in the filehandle of the file system when NFS-exported, and giving it out is a security concern.

Under some OSes the *fsid* can be used as second argument to the **sysfs**() system call.

SEE ALSO

stat(2), **statvfs**(2), **path_resolution**(7)

COLOPHON

This page is part of release 3.22 of the Linux *man-pages* project. A description of the project, and information about reporting bugs, can be found at http://www.kernel.org/doc/man-pages/.