NAME

tempnam, **tmpfile**, **tmpnam** - temporary file routines

LIBRARY

Standard C Library (libc, -lc)

SYNOPSIS

#include <stdio.h>

*FILE **
tmpfile(*void*);

*char **
tmpnam(*char *str*);

*char **
tempnam(*const char *tmpdir*, *const char *prefix*);

DESCRIPTION

The **tmpfile**() function returns a pointer to a stream associated with a file descriptor returned by the routine mkstemp(3). The created file is unlinked before **tmpfile**() returns, causing the file to be automatically deleted when the last reference to it is closed. The file is opened with the access value 'w+'. The file is created in the directory determined by the environment variable TMPDIR if set. The default location if TMPDIR is not set is */tmp*.

The **tmpnam**() function returns a pointer to a file name, in the P_tmpdir directory, which did not reference an existing file at some indeterminate point in the past. P_tmpdir is defined in the include file <*stdio.h*>. If the argument *str* is non-NULL, the file name is copied to the buffer it references. Otherwise, the file name is copied to a static buffer. In either case, **tmpnam**() returns a pointer to the file name.

The buffer referenced by *str* is expected to be at least L_tmpnam bytes in length. L_tmpnam is defined in the include file <*stdio.h*>.

The **tempnam**() function is similar to **tmpnam**(), but provides the ability to specify the directory which will contain the temporary file and the file name prefix.

The environment variable TMPDIR (if set), the argument *tmpdir* (if non-NULL), the directory P_tmpdir, and the directory */tmp* are tried, in the listed order, as directories in which to store the temporary file.

The argument *prefix*, if non-NULL, is used to specify a file name prefix, which will be the first part of the created file name. The **tempnam**() function allocates memory in which to store the file name; the returned pointer may be used as a subsequent argument to free(3).

RETURN VALUES

The **tmpfile**() function returns a pointer to an open file stream on success, and a NULL pointer on error.

The **tmpnam**() and **tempfile**() functions return a pointer to a file name on success, and a NULL pointer on error.

ENVIRONMENT

TMPDIR

[**tempnam**() only] If set, the directory in which the temporary file is stored. TMPDIR is ignored for processes for which issetugid(2) is true.

COMPATIBILITY

These interfaces are provided from System V and ANSI compatibility only.

Most historic implementations of these functions provide only a limited number of possible temporary file names (usually 26) before file names will start being recycled. System V implementations of these functions (and of mktemp(3)) use the access(2) system call to determine whether or not the temporary file may be created. This has obvious ramifications for setuid or setgid programs, complicating the portable use of these interfaces in such programs.

The **tmpfile**() interface should not be used in software expected to be used on other systems if there is any possibility that the user does not wish the temporary file to be publicly readable and writable.

ERRORS

The **tmpfile**() function may fail and set the global variable *errno* for any of the errors specified for the library functions fdopen(3) or mkstemp(3).

The **tmpnam**() function may fail and set *errno* for any of the errors specified for the library function mktemp(3).

The **tempnam**() function may fail and set *errno* for any of the errors specified for the library functions malloc(3) or mktemp(3).

SEE ALSO

mkstemp(3), mktemp(3)

STANDARDS

The **tmpfile**() and **tmpnam**() functions conform to ISO/IEC 9899:1990 (‘‘ISO C90’’).

SECURITY CONSIDERATIONS

The **tmpnam**() and **tempnam**() functions are susceptible to a race condition occurring between the selection of the file name and the creation of the file, which allows malicious users to potentially overwrite arbitrary files in the system, depending on the level of privilege of the running program. Additionally, there is no means by which file permissions may be specified. It is strongly suggested that mkstemp(3) be used in place of these functions.