NAME

_secure_path - determine if a file appears to be secure

LIBRARY

System Utilities Library (libutil, -lutil)

SYNOPSIS

#include <sys/types.h>
#include <libutil.h>

int
_secure_path(*const char *path*, *uid_t uid*, *gid_t gid*);

DESCRIPTION

This function does some basic security checking on a given path. It is intended to be used by processes running with root privileges in order to decide whether or not to trust the contents of a given file. It uses a method often used to detect system compromise.

A file is considered ‘secure’ if it meets the following conditions:

1. The file exists, and is a regular file (not a symlink, device special or named pipe, etc.),

2. Is not world writable.

3. Is owned by the given uid or uid 0, if uid is not -1,

4. Is not group writable or it has group ownership by the given gid, if gid is not -1.

RETURN VALUES

This function returns zero if the file exists and may be considered secure, -2 if the file does not exist, and -1 otherwise to indicate a security failure. The syslog(3) function is used to log any failure of this function, including the reason, at LOG_ERR priority.

SEE ALSO

lstat(2), syslog(3)

HISTORY

Code from which this function was derived was contributed to the FreeBSD project by Berkeley Software Design, Inc. The function **_secure_path**() first appeared in FreeBSD 2.2.5.

BUGS

The checks carried out are rudimentary and no attempt is made to eliminate race conditions between use of this function and access to the file referenced.