NAME

intro − Introduction to system calls

DESCRIPTION

Section 2 of the manual describes the Linux system calls. A system call is an entry point into the Linux kernel. Usually, system calls are not invoked directly: instead, most system calls have corresponding C library wrapper functions which perform the steps required (e.g., trapping to kernel mode) in order to invoke the system call. Thus, making a system call looks the same as invoking a normal library function.

For a list of the Linux system calls, see **syscalls**(2).

RETURN VALUE

On error, most system calls return a negative error number (i.e., the negated value of one of the constants described in **errno**(3)). The C library wrapper hides this detail from the caller: when a system call returns a negative value, the wrapper copies the absolute value into the *errno* variable, and returns −1 as the return value of the wrapper.

The value returned by a successful system call depends on the call. Many system calls return 0 on success, but some can return non-zero values from a successful call. The details are described in the individual manual pages.

In some cases, the programmer must define a feature test macro in order to obtain the declaration of a system call from the header file specified in the man page SYNOPSIS section. In such cases, the required macro is described in the man page. For further information on feature test macros, see **feature_test_macros**(7).

CONFORMING TO

Certain terms and abbreviations are used to indicate Unix variants and standards to which calls in this section conform. See **standards**(7).

NOTES

Calling Directly

In most cases, it is unnecessary to invoke a system call directly, but there are times when the Standard C library does not implement a nice wrapper function for you. In this case, the programmer must manually invoke the system call using **syscall**(2). Historically, this was also possible using one of the _syscall macros described in **_syscall**(2).

Authors and Copyright Terms

Look at the header of the manual page source for the author(s) and copyright conditions. Note that these can be different from page to page!

SEE ALSO

_syscall(2), **syscall**(2), **errno**(3), **feature_test_macros**(7), **standards**(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/.