

NAME

syscall – indirect system call

SYNOPSIS

```
#define _GNU_SOURCE    /* or _BSD_SOURCE or _SVID_SOURCE */
#include <unistd.h>
#include <sys/syscall.h> /* For SYS_XXX definitions */
```

```
int syscall(int number, ...);
```

DESCRIPTION

syscall() performs the system call whose assembly language interface has the specified *number* with the specified arguments. Symbolic constants for system calls can be found in the header file `<sys/syscall.h>`.

RETURN VALUE

The return value is defined by the system call being invoked. In general, a 0 return value indicates success. A -1 return value indicates an error, and an error code is stored in *errno*.

NOTES

syscall() first appeared in 4BSD.

EXAMPLE

```
#define _GNU_SOURCE
#include <unistd.h>
#include <sys/syscall.h>
#include <sys/types.h>

int
main(int argc, char *argv[])
{
    pid_t tid;

    tid = syscall(SYS_gettid);
}
```

SEE ALSO

`_syscall(2)`, `intro(2)`, `syscalls(2)`

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/>.