NAME

create_module − create a loadable module entry

SYNOPSIS

#include <linux/module.h>

[[deprecated]] caddr_t create_module(const char ***_name_, size_t** _size_**);**

DESCRIPTION

Note: This system call is present only before Linux 2.6.

create_module() attempts to create a loadable module entry and reserve the kernel memory that will be needed to hold the module. This system call requires privilege.

RETURN VALUE

On success, returns the kernel address at which the module will reside. On error, −1 is returned and _errno_ is set to indicate the error.

ERRORS

EEXIST
A module by that name already exists.

EFAULT
name is outside the program's accessible address space.

EINVAL
The requested size is too small even for the module header information.

ENOMEM
The kernel could not allocate a contiguous block of memory large enough for the module.

ENOSYS
create_module() is not supported in this version of the kernel (e.g., Linux 2.6 or later).

EPERM
The caller was not privileged (did not have the **CAP_SYS_MODULE** capability).

VERSIONS

This system call is present only up until Linux 2.4; it was removed in Linux 2.6.

STANDARDS

create_module() is Linux-specific.

NOTES

This obsolete system call is not supported by glibc. No declaration is provided in glibc headers, but, through a quirk of history, glibc versions before glibc 2.23 did export an ABI for this system call. Therefore, in order to employ this system call, it was sufficient to manually declare the interface in your code; alternatively, you could invoke the system call using **syscall**(2).

SEE ALSO

delete_module(2), **init_module**(2), **query_module**(2)