

NAME

mmap2 – map files or devices into memory

SYNOPSIS

```
#include <sys/mman.h>
```

```
void *mmap2(void *addr, size_t length, int prot,  
            int flags, int fd, off_t pgoffset);
```

DESCRIPTION

This is probably not the system call that you are interested in; instead, see [mmap\(2\)](#), which describes the glibc wrapper function that invokes this system call.

The **mmap2()** system call provides the same interface as [mmap\(2\)](#), except that the final argument specifies the offset into the file in 4096-byte units (instead of bytes, as is done by [mmap\(2\)](#)). This enables applications that use a 32-bit *off_t* to map large files (up to 2^{44} bytes).

RETURN VALUE

On success, **mmap2()** returns a pointer to the mapped area. On error, -1 is returned and *errno* is set appropriately.

ERRORS**EFAULT**

Problem with getting the data from user space.

EINVAL

(Various platforms where the page size is not 4096 bytes.) *offset * 4096* is not a multiple of the system page size.

mmap2() can also return any of the errors described in [mmap\(2\)](#).

VERSIONS

mmap2() is available since Linux 2.3.31.

CONFORMING TO

This system call is Linux-specific.

NOTES

On architectures where this system call is present, the glibc **mmap()** wrapper function invokes this system call rather than the [mmap\(2\)](#) system call.

This system call does not exist on x86-64.

On ia64, the unit for *offset* is actually the system page size, rather than 4096 bytes.

SEE ALSO

[getpagesize\(2\)](#), [mmap\(2\)](#), [mremap\(2\)](#), [msync\(2\)](#), [shm_open\(3\)](#)

COLOPHON

This page is part of release 4.16 of the Linux *man-pages* project. A description of the project, information about reporting bugs, and the latest version of this page, can be found at <https://www.kernel.org/doc/man-pages/>.