NAME

shmat, **shmdt** - attach or detach shared memory

LIBRARY

Standard C Library (libc, -lc)

SYNOPSIS

#include <sys/types.h>
#include <sys/ipc.h>
#include <sys/shm.h>

*void **
shmat(*int shmid, const void *addr, int flag*);

int
shmdt(*const void *addr*);

DESCRIPTION

The **shmat**() system call attaches the shared memory segment identified by *shmid* to the calling process's address space. The address where the segment is attached is determined as follows:

- If *addr* is 0, the segment is attached at an address selected by the kernel.
- If *addr* is nonzero and *SHM_RND* is not specified in *flag*, the segment is attached the specified address.
- If *addr* is specified and *SHM_RND* is specified, *addr* is rounded down to the nearest multiple of SHMLBA.

If the *SHM_REMAP* flag is specified and the passed *addr* is not NULL, any existing mappings in the virtual addresses range are cleared before the segment is attached. If the flag is not specified, *addr* is not NULL, and the virtual address range contains some pre-existing mappings, the **shmat**() call fails.

The **shmdt**() system call detaches the shared memory segment at the address specified by *addr* from the calling process's address space.

RETURN VALUES

Upon success, **shmat**() returns the address where the segment is attached; otherwise, -1 is returned and *errno* is set to indicate the error.

The **shmdt**() function returns the value 0 if successful; otherwise the value -1 is returned and the global variable *errno* is set to indicate the error.

ERRORS

The **shmat**() system call will fail if:

[EINVAL] No shared memory segment was found corresponding to *shmid*.

[EINVAL] The *addr* argument was not an acceptable address.

[ENOMEM] The specified *addr* cannot be used for mapping, for instance due to the amount of available space being smaller than the segment size, or because pre-existing mappings are in the range and no *SHM_REMAP* flag was provided.

[EMFILE] Failed to attach the shared memory segment because the per-process *kern.ipc.shmseg* sysctl(3) limit was reached.

The **shmdt**() system call will fail if:

[EINVAL] The *addr* argument does not point to a shared memory segment.

SEE ALSO

shmctl(2), shmget(2)