shmat (System Call)

NAME

shmat – XSI shared memory attach operation

SYNOPSIS
[crayon-6a9d5fdc06dba173976440/]
DESCRIPTION

The shmat() function operates on XSI shared memory (see the Base Definitions volume of IEEE Std 1003.1-2001, Section 3.340, Shared Memory Object). It is unspecified whether this function interoperates with the realtime interprocess communication facilities defined in Realtime.

The shmat() function attaches the shared memory segment associated with the shared memory identifier specified by shmid to the address space of the calling process. The segment is attached at the address specified by one of the following criteria:

If shmaddr is a null pointer, the segment is attached at the first available address as selected by the system.

If shmaddr is not a null pointer and (shmflg &SHM_RND) is non-zero, the segment is attached at the address given by (shmaddr -((uintptr_t)shmaddr %SHMLBA)). The character '%' is the C-language remainder operator.

If shmaddr is not a null pointer and (shmflg &SHM_RND) is 0, the segment is attached at the address given by shmaddr.

The segment is attached for reading if (shmflg &SHM_RDONLY) is non-zero and the calling process has read permission; otherwise, if it is 0 and the calling process has read and write permission, the segment is attached for reading and writing.

RETURN VALUE

Upon successful completion, shmat() shall increment the value of shm_nattch in the data structure associated with the shared memory ID of the attached shared memory segment and return the segment’s start address.

Otherwise, the shared memory segment shall not be attached, shmat() shall return -1, and errno shall be set to indicate the error.

ERRORS

The shmat() function shall fail if:

[EACCES]
Operation permission is denied to the calling process; see XSI Interprocess Communication.

[EINVAL]
The value of shmid is not a valid shared memory identifier, the shmaddr is not a null pointer, and the value of (shmaddr - ((uintptr_t)shmaddr %SHMLBA)) is an illegal address for attaching shared memory; or the shmaddr is not a null pointer, (shmflg &SHM_RND) is 0, and the value of shmaddr is an illegal address for attaching shared memory.

[EMFILE]
The number of shared memory segments attached to the calling process would exceed the system-imposed limit.

[ENOMEM]
The available data space is not large enough to accommodate the shared memory segment.