



Full credit is given to the above companies including the OS that this PDF file was generated!

Red Hat Enterprise Linux Release 9.2 Manual Pages on 'shmat.3p' command

\$ man shmat.3p

SHMAT(3P) POSIX Programmer's Manual SHMAT(3P)

PROLOG

This manual page is part of the POSIX Programmer's Manual. The Linux implementation of this interface may differ (consult the corresponding Linux manual page for details of Linux behavior), or the interface may not be implemented on Linux.

NAME

shmat ? XSI shared memory attach operation

SYNOPSIS

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

void *shmat(int shmid, const void *shmaddr, int shmflg);
```

DESCRIPTION

The shmat() function operates on XSI shared memory (see the Base Definitions volume of POSIX.1?2017, Section 3.346, Shared Memory Object).

It is unspecified whether this function interoperates with the realtime interprocess communication facilities defined in Section 2.8, 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. Also, the `shm_atime` timestamp shall be set to the current time, as described in Section 2.7.1, IPC General Description.

Otherwise, the shared memory segment shall not be attached, `shmat()` shall return `(void *)-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 Section 2.7, 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.

The following sections are informative.

EXAMPLES

None.

APPLICATION USAGE

The POSIX Realtime Extension defines alternative interfaces for interprocess communication. Application developers who need to use IPC should design their applications so that modules using the IPC routines described in Section 2.7, XSI Interprocess Communication can be easily modified to use the alternative interfaces.

RATIONALE

None.

FUTURE DIRECTIONS

None.

SEE ALSO

Section 2.7, XSI Interprocess Communication, Section 2.8, Realtime, `exec`, `exit()`, `fork()`, `shmctl()`, `shmdt()`, `shmget()`, `shm_open()`, `shm_unlink()`

The Base Definitions volume of POSIX.1-2017, Section 3.346, Shared Memory Object, `<sys_shm.h>`

COPYRIGHT

Portions of this text are reprinted and reproduced in electronic form from IEEE Std 1003.1-2017, Standard for Information Technology -- Portable Operating System Interface (POSIX), The Open Group Base Specifications Issue 7, 2018 Edition, Copyright (C) 2018 by the Institute of Electrical and Electronics Engineers, Inc and The Open Group. In the event of any discrepancy between this version and the original IEEE and The Open Group Standard, the original IEEE and The Open Group Standard is the referee document. The original Standard can be obtained online at <http://www.opengroup.org/unix/online.html>.

Any typographical or formatting errors that appear in this page are most likely to have been introduced during the conversion of the source files to man page format. To report such errors, see https://www.kernel.org/doc/man-pages/reporting_bugs.html.