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

$ man sem_init.3

SEM_INIT(3)   Linux Programmer's Manual   SEM_INIT(3)

NAME

sem_init - initialize an unnamed semaphore

SYNOPSIS

#include <semaphore.h>

int sem_init(sem_t *sem, int pshared, unsigned int value);

Link with -pthread.

DESCRIPTION

sem_init() initializes the unnamed semaphore at the address pointed to by sem. The value argument specifies the initial value for the sema? phore.

The pshared argument indicates whether this semaphore is to be shared between the threads of a process, or between processes.

If pshared has the value 0, then the semaphore is shared between the threads of a process, and should be located at some address that is visible to all threads (e.g., a global variable, or a variable allo? cated dynamically on the heap).

If pshared is nonzero, then the semaphore is shared between processes, and should be located in a region of shared memory (see shm_open(3), mmap(2), and shmget(2)). (Since a child created by fork(2) inherits its parent's memory mappings, it can also access the semaphore.) Any process that can access the shared memory region can operate on the semaphore using sem_post(3), sem_wait(3), and so on.

Initializing a semaphore that has already been initialized results in undefined behavior.

RETURN VALUE

sem_init() returns 0 on success; on error, -1 is returned, and errno is set to indicate the error.

ERRORS

EINVAL value exceeds SEM_VALUE_MAX.

ENOSYS pshared is nonzero, but the system does not support process-shared semaphores (see sem_overview(7)).

ATTRIBUTES

For an explanation of the terms used in this section, see at? tributes(7).

```
????????????????????????????????????????
?Interface  ? Attribute     ? Value   ?
????????????????????????????????????????
?sem_init() ? Thread safety ? MT-Safe ?
????????????????????????????????????????
```

CONFORMING TO

POSIX.1-2001.

NOTES

Bizarrely, POSIX.1-2001 does not specify the value that should be re? turned by a successful call to sem_init(). POSIX.1-2008 rectifies this, specifying the zero return on success.

EXAMPLES

See shm_open(3) and sem_wait(3).

SEE ALSO

sem_destroy(3), sem_post(3), sem_wait(3), sem_overview(7)

COLOPHON

This page is part of release 5.10 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/.