Red Hat Enterprise Linux Release 9.2 Manual Pages on 'sgetmask.2' command

$ man sgetmask.2

SGETMASK(2)   Linux Programmer's Manual   SGETMASK(2)

NAME

   sgetmask, ssetmask - manipulation of signal mask (obsolete)

SYNOPSIS

   long sgetmask(void);

   long ssetmask(long newmask);

   Note: There are no glibc wrappers for these system calls; see NOTES.

DESCRIPTION

   These system calls are obsolete. Do not use them; use sigprocmask(2)

   instead.

   sgetmask() returns the signal mask of the calling process.

   ssetmask() sets the signal mask of the calling process to the value

   given in newmask. The previous signal mask is returned.

   The signal masks dealt with by these two system calls are plain bit

   masks (unlike the sigset_t used by sigprocmask(2)); use sigmask(3) to

   create and inspect these masks.

RETURN VALUE

   sgetmask() always successfully returns the signal mask. ssetmask() al?

   ways succeeds, and returns the previous signal mask.

ERRORS

   These system calls always succeed.

VERSIONS

   Since Linux 3.16, support for these system calls is optional, depending

on whether the kernel was built with the CONFIG_SGETMASK_SYSCALL op‐
tion.

CONFORMING TO

These system calls are Linux-specific.

NOTES

Glibc does not provide wrappers for these obsolete system calls; in the
unlikely event that you want to call them, use syscall(2).

These system calls are unaware of signal numbers greater than 31 (i.e.,
real-time signals).

These system calls do not exist on x86-64.

It is not possible to block SIGSTOP or SIGKILL.

SEE ALSO

sigprocmask(2), signal(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/.

Linux   2020-08-13   SGETMASK(2)