## NAME

sigreturn − return from signal handler and cleanup stack frame

## SYNOPSIS

**int sigreturn(unsigned long** *__unused***);**

## DESCRIPTION

When the Linux kernel creates the stack frame for a signal handler, a call to **sigreturn**() is inserted into the stack frame so that upon return from the signal handler, **sigreturn**() will be called.

This **sigreturn**() call undoes everything that was done—changing the process's signal mask, switching stacks (see **sigaltstack**(2))—in order to invoke the signal handler: it restores the process's signal mask, switches stacks, and restores the process's context (registers, processor flags), so that the process directly resumes execution at the point where it was interrupted by the signal.

## RETURN VALUE

**sigreturn**() never returns.

## FILES

/usr/src/linux/arch/i386/kernel/signal.c
/usr/src/linux/arch/alpha/kernel/entry.S

## CONFORMING TO

**sigreturn**() is specific to Linux and should not be used in programs intended to be portable.

## NOTES

The **sigreturn**() call is used by the kernel to implement signal handlers. It should **never** be called directly. Better yet, the specific use of the *__unused* argument varies depending on the architecture.

## SEE ALSO

**kill**(2), **sigaltstack**(2), **signal**(2), **signal**(7)

## COLOPHON

This page is part of release 3.22 of the Linux *man-pages* project. A description of the project, and information about reporting bugs, can be found at http://www.kernel.org/doc/man-pages/.