

SCO OpenServer Product Family Release 5

kill(C)

kill -- terminate a process

Syntax

kill -s *signame* *pid* ...

kill [-*signame*] *pid* ...

kill -l [*exit_status*]

Description

kill sends signals to processes. By default, it sends signal 15 (**SIGTERM**, software termination signal) to the processes identified by their process ID *pid*. Other signals can be specified by symbolic name or number.

A signal normally kills processes that are not configured to catch or ignore it. For details of how to handle signals in the shell, see the description of the **trap** command on the [sh\(C\)](#) and [ksh\(C\)](#) manual pages. System programmers should refer to [kill\(S\)](#), [sigaction\(S\)](#), and [signal\(S\)](#).

To find the process ID of a given process, use [ps\(C\)](#). Additionally the shell reports the process ID of each asynchronous (background) process started with ``&'' (unless more than one process is started in a pipeline, in which case the number of the last process in the pipeline is reported).

kill recognizes the following options:

-s *signame*

-*signame*

Specify signal name *signame* (see ``Signals'' below).

Note that the preferred syntax is -s *signame*; -*signame* is obsolete and may be dropped from future standards.

-l List all the signal names recognized by **kill**.

-l *exit_status*

If *exit_status* is the exit status of a previously executed command then **kill** lists the signal which terminated that process.

When determining the process or processes to send a signal to, **kill** interprets the value of *pid* as follows:

- >1 Send the signal to the process whose process ID is *pid*.
- 0 Send the signal to all processes whose process group ID is equal to the process group ID of the sender (except for processes 0 and 1).
- 1 If the effective user ID of the sender is not *root*, send the signal to all processes (except processes 0 and 1) whose real user ID is equal to the effective user ID of the sender.

If the effective user ID of the sender is *root*, send the signal to all processes (except processes 0 and 1).
- <-1 If the process ID is negative but not -1, send the signal to all processes whose process group ID is equal to the absolute (positive) value of PID.

Signals

Some of the more useful values of *signame* are shown below (*signum* is given in parentheses):

HUP (1)

Hangup signal. Sent to processes when their controlling terminal hangs up the phone line, or their process group leader exits.

INT (2)

Interrupt signal. If the process fails to trap the interrupt signal, it terminates; otherwise its behavior is program dependent.

QUIT (3)

Quit signal. Causes the process to dump core. See [core\(FP\)](#) for details.

KILL (9)

Kill signal. Always kills the receiving process since it cannot be ignored or caught.

TERM (15)

Software termination signal. This is the default signal issued by **kill**.

Note that there are many other signals, most of which are generated by error conditions within the system; they are not relevant to most users. Signals are defined in */usr/include/sys/signal.h*; strip off the **SIG** prefix before using them with **kill**. For further details, see [sigaction\(S\)](#) or [signal\(S\)](#).

Korn shell version of kill

The version of **kill** built into the Korn shell accepts job IDs as arguments as well as process IDs if job control is enabled. See [ksh\(C\)](#) for details. Refer to [jobs\(C\)](#) for details of acceptable job ID formats.

Examples

To kill processes 21344 and 104:

```
kill 21344 104
```

To send signal 2 (**SIGINT**) to process 1091:

kill -s INT 1091

To determine the signal that terminated the process **job**:

```
#!/bin/sh
job
CODE=$?
if [ ${CODE} -eq 0 ]
then
    echo "Job completed normally."
elif [ ${CODE} -gt 128 ]
then
    echo "Job terminated by SIG`kill -l ${CODE}`"
else
    echo "Job terminated with error code ${CODE}."
fi
```

Limitations

It is not always possible to kill a process. If the target process is sleeping with priority greater than or equal to 77, it ignores all signals (including **KILL**). See [ps\(C\)](#) for more information about process priorities.

See also

[core\(FP\)](#), [csh\(C\)](#), [kill\(S\)](#), [killall\(ADM\)](#), [ksh\(C\)](#), [ps\(C\)](#), [sh\(C\)](#), [sigaction\(S\)](#), [signal\(S\)](#)

Standards conformance

kill is conformant with:

ISO/IEC DIS 9945-2:1992, Information technology - Portable Operating System Interface (POSIX) - Part 2: Shell and Utilities (IEEE Std 1003.2-1992);

AT&T SVID Issue 2;

X/Open CAE Specification, Commands and Utilities, Issue 4, 1992.

1 May 1995