Rocky Enterprise Linux 9.2 Manual Pages on command 'kill.1'

$ man kill.1

KILL(1)   User Commands   KILL(1)

NAME

    kill - terminate a process

SYNOPSIS

    kill [-signal|-s signal|-p] [-q value] [-a] [--timeout milliseconds

    signal] [--] pid|name...

    kill -l [number] | -L

DESCRIPTION

    The command kill sends the specified signal to the specified processes

    or process groups.

    If no signal is specified, the TERM signal is sent. The default action

    for this signal is to terminate the process. This signal should be used

    in preference to the KILL signal (number 9), since a process may

    install a handler for the TERM signal in order to perform clean-up

    steps before terminating in an orderly fashion. If a process does not

    terminate after a TERM signal has been sent, then the KILL signal may

    be used; be aware that the latter signal cannot be caught, and so does

    not give the target process the opportunity to perform any clean-up

before terminating.

Most modern shells have a builtin kill command, with a usage rather similar to that of the command described here. The --all, --pid, and --queue options, and the possibility to specify processes by command name, are local extensions.

If signal is 0, then no actual signal is sent, but error checking is still performed.

ARGUMENTS

The list of processes to be signaled can be a mixture of names and PIDs.

pid

Each pid can be expressed in one of the following ways:

n

where n is larger than 0. The process with PID n is signaled.

0

All processes in the current process group are signaled.

-1

All processes with a PID larger than 1 are signaled.

-n

where n is larger than 1. All processes in process group n are signaled. When an argument of the form '-n' is given, and it is meant to denote a process group, either a signal must be specified first, or the argument must be preceded by a '--' option, otherwise it will be taken as the signal to send.

name

All processes invoked using this name will be signaled.

OPTIONS

-s, --signal signal

The signal to send. It may be given as a name or a number.

-l, --list [number]

Print a list of signal names, or convert the given signal number to a name. The signals can be found in /usr/include/linux/signal.h.

-L, --table

Similar to -l, but it will print signal names and their

corresponding numbers.

-a, --all

Do not restrict the command-name-to-PID conversion to processes

with the same UID as the present process.

-p, --pid

Only print the process ID (PID) of the named processes, do not send

any signals.

--verbose

Print PID(s) that will be signaled with kill along with the signal.

-q, --queue value

Send the signal using sigqueue(3) rather than kill(2). The value

argument is an integer that is sent along with the signal. If the

receiving process has installed a handler for this signal using the

SA_SIGINFO flag to sigaction(2), then it can obtain this data via

the si_sigval field of the siginfo_t structure.

--timeout milliseconds signal

Send a signal defined in the usual way to a process, followed by an

additional signal after a specified delay. The --timeout option

causes kill to wait for a period defined in milliseconds before

sending a follow-up signal to the process. This feature is

implemented using the Linux kernel PID file descriptor feature in

order to guarantee that the follow-up signal is sent to the same

process or not sent if the process no longer exists.

Note that the operating system may re-use PIDs and implementing an

equivalent feature in a shell using kill and sleep would be subject

to races whereby the follow-up signal might be sent to a different

process that used a recycled PID.

The --timeout option can be specified multiple times: the signals

are sent sequentially with the specified timeouts. The --timeout

option can be combined with the --queue option.

As an example, the following command sends the signals QUIT, TERM

and KILL in sequence and waits for 1000 milliseconds between

sending the signals:

```
kill --verbose --timeout 1000 TERM --timeout 1000 KILL \
        --signal QUIT 12345
```

EXIT STATUS

kill has the following exit status values:

0

success

1

failure

64

partial success (when more than one process specified)

NOTES

Although it is possible to specify the TID (thread ID, see gettid(2))
of one of the threads in a multithreaded process as the argument of
kill, the signal is nevertheless directed to the process (i.e., the
entire thread group). In other words, it is not possible to send a
signal to an explicitly selected thread in a multithreaded process. The
signal will be delivered to an arbitrarily selected thread in the
target process that is not blocking the signal. For more details, see
signal(7) and the description of CLONE_THREAD in clone(2).
Various shells provide a builtin kill command that is preferred in
relation to the kill(1) executable described by this manual. The
easiest way to ensure one is executing the command described in this
page is to use the full path when calling the command, for example:
/bin/kill --version

AUTHORS

Salvatore Valente <svalente@mit.edu>, Karel Zak <kzak@redhat.com>
The original version was taken from BSD 4.4.

SEE ALSO

bash(1), tcsh(1), sigaction(2), kill(2), sigqueue(3), signal(7)

REPORTING BUGS

For bug reports, use the issue tracker at
https://github.com/karelzak/util-linux/issues.

AVAILABILITY

The kill command is part of the util-linux package which can be

downloaded from Linux Kernel Archive

<https://www.kernel.org/pub/linux/utils/util-linux/>.

util-linux 2.37.4   2022-02-14   KILL(1)