Why number 9 in kill -9 command in unix? [closed]

Asked 12 years, 8 months ago Modified 6 years, 1 month ago

Viewed 160k times

110

Closed. This question does not meet Stack Overflow guidelines. It is not currently accepting answers.

This question does not appear to be about a specific programming problem, a software algorithm, or software tools primarily used by programmers. If you believe the question would be on-topic on another Stack Exchange site, you can leave a comment to explain where the question may be able to be answered.

Closed 5 years ago.

Improve this question

I understand it's off topic, I couldn't find anywhere online and I was thinking maybe programming gurus in the community might know this.
I usually use

```
kill -9 pid
```

to kill the job. I always wondered the origin of 9. I looked it up online, and it says

"9 Means KILL signal that is not catchable or ignorable. In other words it would signal process (some running application) to quit immediately" (source: http://wiki.answers.com/Q/What_does_kill_-9_do_in_unix_in_its_entirety)

But, why 9? and what about the other numbers? is there any historical significance or because of the architecture of Unix?

unix command history kill

Share

Improve this question

Follow

edited Apr 24, 2016 at 12:07

Rafał Rawicki

22.7k ● 5 ● 62 ● 79

asked Mar 30, 2012 at 23:24

Alby

5,712 ● 8 ● 44 ● 52

6 I think, that this question better belongs to superuser.com – Rafał Rawicki Mar 30, 2012 at 23:30

13 Answers

Sorted by: Highest score (default)

96

See the wikipedia article on Unix signals for the list of other signals. SIGKILL just happened to get the number 9.

You can as well use the mnemonics, as the numbers:

```
kill -SIGKILL pid
```

Share Improve this answer

Follow

answered Mar 30, 2012 at 23:27

Rafał Rawicki

22.7k ● 5 ● 62 ● 79

6 Back in the day (by which I mean 4.xBSD or so) you couldn't use the mnemonics with the shell command, which is why the number 9 is written into an awful lot of old dusty-deck shell scripts. – zwol Mar 30, 2012 at 23:33

78

There were 8 other signals they came up with first.

Share Improve this answer

Follow

answered Mar 30, 2012 at 23:26

Jonathon Reinhart

137k ● 37 ● 263 ● 339

60

I think a better answer here is simply this:

```
mike@sleepycat:~☺  kill -l
 1) SIGHUP   2) SIGINT   3) SIGQUIT  4) SIGILL
5) SIGTRAP
 6) SIGABRT  7) SIGBUS   8) SIGFPE   9) SIGKILL
10) SIGUSR1
11) SIGSEGV 12) SIGUSR2 13) SIGPIPE 14) SIGALRM
15) SIGTERM
```

```
16) SIGSTKFLT   17) SIGCHLD 18) SIGCONT 19)
SIGSTOP 20) SIGTSTP
21) SIGTTIN 22) SIGTTOU 23) SIGURG  24) SIGXCPU
25) SIGXFSZ
26) SIGVTALRM   27) SIGPROF 28) SIGWINCH    29)
SIGIO   30) SIGPWR
31) SIGSYS  34) SIGRTMIN    35) SIGRTMIN+1  36)
SIGRTMIN+2  37) SIGRTMIN+3
38) SIGRTMIN+4  39) SIGRTMIN+5  40) SIGRTMIN+6
41) SIGRTMIN+7  42) SIGRTMIN+8
43) SIGRTMIN+9  44) SIGRTMIN+10 45) SIGRTMIN+11
46) SIGRTMIN+12 47) SIGRTMIN+13
48) SIGRTMIN+14 49) SIGRTMIN+15 50) SIGRTMAX-14
51) SIGRTMAX-13 52) SIGRTMAX-12
53) SIGRTMAX-11 54) SIGRTMAX-10 55) SIGRTMAX-9
56) SIGRTMAX-8  57) SIGRTMAX-7
58) SIGRTMAX-6  59) SIGRTMAX-5  60) SIGRTMAX-4
61) SIGRTMAX-3  62) SIGRTMAX-2
63) SIGRTMAX-1  64) SIGRTMAX
```

As for the "significance" of 9... I would say there is probably none. According to The Linux Programming Interface(p 388):

> Each signal is defined as a unique (small) integer, starting sequentially from 1. These integers are defined in with symbolic names of the form SIGxxxx . Since **the actual numbers used for each signal vary across implementations**, it is these symbolic names that are always used in programs.

Share Improve this answer

Follow

answered Nov 13, 2013 at 0:59

mikewilliamson
24.8k ● 17 ● 60 ● 92

34

First you need to know what are **Signals** in Unix-like systems (It'll take just few minutes).

> **Signals**, are software interrupts sent to a (running) program to indicate that an important event has occurred.
>
> > The events can vary from user requests to illegal memory access errors. Some signals, such as the interrupt signal, indicate that a user has asked the program to do something that is not in the usual flow of control.

There are several types of Signals we can use - to get a full list of all the available/possible Signals use "**$ kill -l**" command:

```
$ kill -l
 1) SIGHUP       2) SIGINT       3) SIGQUIT      4) SIGILL
 5) SIGTRAP      6) SIGABRT      7) SIGBUS       8) SIGFPE
 9) SIGKILL     10) SIGUSR1     11) SIGSEGV     12) SIGUSR2
13) SIGPIPE     14) SIGALRM     15) SIGTERM     16) SIGSTKFLT
17) SIGCHLD     18) SIGCONT     19) SIGSTOP     20) SIGTSTP
21) SIGTTIN     22) SIGTTOU     23) SIGURG      24) SIGXCPU
25) SIGXFSZ     26) SIGVTALRM   27) SIGPROF     28) SIGWINCH
29) SIGIO       30) SIGPWR      31) SIGSYS      34) SIGRTMIN
35) SIGRTMIN+1  36) SIGRTMIN+2  37) SIGRTMIN+3  38) SIGRTMIN+4
39) SIGRTMIN+5  40) SIGRTMIN+6  41) SIGRTMIN+7  42) SIGRTMIN+8
43) SIGRTMIN+9  44) SIGRTMIN+10 45) SIGRTMIN+11 46) SIGRTMIN+12
47) SIGRTMIN+13 48) SIGRTMIN+14 49) SIGRTMIN+15 50) SIGRTMAX-14
51) SIGRTMAX-13 52) SIGRTMAX-12 53) SIGRTMAX-11 54) SIGRTMAX-10
55) SIGRTMAX-9  56) SIGRTMAX-8  57) SIGRTMAX-7  58) SIGRTMAX-6
59) SIGRTMAX-5  60) SIGRTMAX-4  61) SIGRTMAX-3  62) SIGRTMAX-2
63) SIGRTMAX-1  64) SIGRTMAX
```

In the above output it's clearly visible, that each Signal has a '**signal number**' (e.g. 1, 2, 3) and a '**signal name**' (e.g. SIGUP, SIGINT, SIGQUIT) associated with it. For a detailed look up what each and every Signal does, visit this link.

Finally, coming to the question "Why number 9 in **kill -9** command":

There are several methods of delivering signals to a program or script. One of commonly used method for sending signal is to use the **kill** command - the basic syntax is:

```
$ kill -signal pid
```

Where signal is either the **number** or **name** of the signal, followed by the process Id (pid) to which the signal will be sent.

For example - **-SIGKILL** (or **-9**), signal kills the process immediately.

```
$ kill -SIGKILL 1001
```

and

```
$ kill -9 1001
```

both command are one the same thing i.e. above we have used the 'signal name', and later we have used

'signal number'.

Verdict: One has an open choice to whether use the 'signal name' or 'signal number' with the **kill** command.

Share Improve this answer

Follow

answered Feb 27, 2015 at 14:00

Nabeel Ahmed

19.2k ● 4 ● 60 ● 63

19

It's a reference to "Revoulution 9" by the Beatles. A collection of strung together sound clips and found noises, this recording features John Lennon repeating over and over "Number 9, Number 9..." Further, this song drew further attention in 1969 when it was discovered that when played backwards, John seemed to be saying "Turn me on, dead man..."

Therefore the ninth signal was destined to be the deadliest of the kill signals.

Share Improve this answer

Follow

answered Jan 15, 2013 at 14:52

Stairbob

255 ● 2 ● 3

21 Really can't tell if you're pulling this out of your ass, or if there's a chance that it's true. – Jonathon Reinhart Apr 8, 2013 at 16:34

The end of that topic truly is epic, particularly the two comments before mine. I also stumbled upon that article about kill and which sequence to use: pthree.org/2012/08/14/appropriate-use-of-kill-9-pid – Gull_Code Feb 5, 2014 at 9:14

I so much want this to be true... :) – nafas Dec 8, 2016 at 12:45

The person saying 'number 9' is not Lennon. I suspect there's just as much truth in the rest of the story (i.e. none) but who knows :) – Stephen Kennedy May 26, 2018 at 18:04

8

There's a very long list of Unix signals, which you can view on Wikipedia. Somewhat confusingly, you can actually use `kill` to send any signal to a process. For instance, `kill -SIGSTOP 12345` forces process 12345 to pause its execution, while `kill -SIGCONT 12345` tells it to resume. A slightly less cryptic version of `kill -9` is `kill -SIGKILL`.

Share Improve this answer

Follow

answered Mar 30, 2012 at 23:32

Lawrence Velázquez

398 ● 1 ● 10

2 The underlying system call that sends signals is also called `kill`. Probably because the default behavior for most of the original set of signals (numbers 1 through 15) was to terminate the process. – zwol Mar 30, 2012 at 23:34

4

I don't think there is any significance to number 9. In addition, despite common believe, `kill` is used not only to kill processes but also send a signal to a process. If you are really curious you can read here and here.

Share Improve this answer

Follow

answered Mar 30, 2012 at 23:32

Ilia Frenkel

1,977 ● 14 ● 20

4

why kill -9 : the number 9 in the list of signals has been chosen to be SIGKILL in reference to "kill the 9 lives of a cat".

Share Improve this answer

Follow

answered May 26, 2018 at 16:17

LaFleche78

41 ● 1

3

`SIGKILL` use to kill the process. `SIGKILL` can not be ignored or handled. In Linux, Ways to give `SIGKILL`.

```
kill -9 <process_pid>
kill -SIGKILL <process_pid>
killall -SIGKILL <process_name>
killall -9 <process_name>
```

Share Improve this answer

Follow

edited Nov 5, 2018 at 12:46

KeyMaker00

6,442 ● 2 ● 54 ● 50

answered Apr 4, 2018 at 5:58

user9594626

1

Type the kill -l command on your shell

you will found that at 9th number [**9) SIGKILL**], so one can use either kill -9 or kill -SIGKILL

SIGKILL is sure kill signal, It can not be dis-positioned, ignore or handle. It always work with its default behaviour, which is to kill the process.

Share Improve this answer

Follow

answered Mar 26, 2017 at 11:05

Sandeep_black

1,431 ● 18 ● 19

1

The `-9` is the signal_number, and specifies that the kill message sent should be of the KILL (non-catchable, non-ignorable) type.

```
kill -9 pid
```

Which is same as below.

```
kill -SIGKILL pid
```

Without specifying a signal_number the default is -15, which is TERM (software termination signal). Typing `kill <pid>` is the same as `kill -15 <pid>`.

Share Improve this answer

Follow

answered May 28, 2017 at 9:54

Harini

571 • 6 • 18

0

Both are same as kill -sigkill processID, kill -9 processID. Its basically for forced termination of the process.

Share Improve this answer

Follow

answered Nov 19, 2014 at 12:45

Abhilash TC

11 • 1

0

there are some process which cannot be kill like this "kill %1" . if we have to terminate that process so special command is used to kill that process which is kill -9. eg open vim and stop if by using ctrl+z then see jobs and after apply kill process than this process will not terminated so here we use kill -9 command for terminating.

Share Improve this answer

Follow

answered May 28, 2017 at 6:51

Usman

11