

做个好杀手Linux的Kill命令与信号（3）PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/144/2021_2022__E5_81_9A_E4_B8_AA_E5_A5_BD_E6_c103_144771.htm 按CTRL c键发出一个SIGINT（信号2），这个信号是键盘要求取得控制权的中断信号。kill和killall这两个命令按照默认的设置都发出一个SIGTERM信号(编号15)。程序中可以设置对SIGTERM信号(15)是捕捉或者忽略，或者以不同的方式解释。因此，如果你的程序对于KILL命令的反应与你预期不同，很可能是被KILL的目标程序的问题。终止一个父进程通常也终止了它的子进程。不过，情况并不总是如此。你知道子进程是什么吗?使用ps命令加上-f选项就可以看到，如下所示: \$ ps axf
22371 ? R 2:35 _ konsole [kdeinit] 22372 pts/3 Ss 0:00 | _ /bin/bash
24322 pts/3 S 0:00 | | _ yes carla is teh awesum 22381 pts/4 Rs 0:00 | _
/bin/bash 24323 pts/4 R 0:00 | | _ ps axf现在，回到SIGHUP的话题。SIGHUP的发音是“sig-hup”，是signal hangup的缩写，含义是“中止信号”。你如何发送一个SIGHUP信号呢?这里有几种方式: # kill -HUP [pid] # killall -HUP [process-name] # kill -1 [pid] # killall -1 [process-name]因此，你可以使用PID或者名称，信号名称或者号码。那么为什么要这样做而不使用/etc/init.d/foo命令重新启动呢?使用它们自己的init(初始化)文件来控制服务是优先选择的方式，因为这些文件通常包含健全和错误检查以及额外的功能。使用“kill”命令和信号的主要原因是尽可能明确地终止挂起和失控的进程，而不必重新启动或者登出。终止进程正如你在关于信号的man page中所看到的，有十几种控制进程的方法。下面是一些常用的方

法: kill -STOP [pid] 发送SIGSTOP (17,19,23)停止一个进程，而并不消灭这个进程。 kill -CONT [pid] 发送SIGCONT (19,18,25)重新开始一个停止的进程。 kill -KILL [pid] 发送SIGKILL (9)强迫进程立即停止，并且不实施清理操作。 kill -9 -1 终止你拥有的全部进程。 SIGKILL和SIGSTOP信号不能被捕捉、封锁或者忽略，但是，其它的信号可以。所以这是你的终极武器。 Bash shell的Kill命令, Bash外壳包含一个内置的kill命令，当执行下面命令: \$ type -all kill kill is a shell built-in kill is /bin/kill命令的结果表明有两个kill命令，一个是BASH的内置命令，另一个是/bin/kill可执行程序。一般来说这两个命令不太可能遇到冲突的情况，不过，如果你确实遇到了kill命令行为异常时，你可以明确的指定/bin/kill命令。你一定要进一步查阅下面的资源中列出的参考资源来了解Linux中kill的妙用，因为这是你进入维护Linux系统领域的门票。这些知识能够让你像做外科手术一样对系统进行维护，而不用在遇到问题时每一次都重新启动系统，就像我们知道的某些蹩脚的操作系统那样。 100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com