

信号(一)---信号的概念

信号是进程间可以用来相互通信的一种低级方法，是Linux系统中非常重要的内容。每个信号都有一个信号名，信号名的本质就是一个整型值，所有的信号名对应的信号值都定义在头文件<signal.h>中或者可以通过 kill 命令加上 -l 选项来查看。Linux系统中支持31种不同的信号，编号从 1 到 31 （信号编号为0有特殊意义，Linux中称为空信号），每种信号都有其不同的意义，如下图所示

名 字	说 明
SIGABRT	异常终止 (abort)
SIGALRM	超时 (alarm)
SIGBUS	硬件故障
SIGCANCEL	线程库内部使用
SIGCHLD	子进程状态改变
SIGCONT	使暂停进程继续
SIGEMT	硬件故障
SIGFPE	算术异常
SIGFREEZE	检查点冻结
SIGHUP	连接断开
SIGILL	非法硬件指令
SIGINFO	键盘状态请求
SIGINT	终端中断符
SIGIO	异步I/O
SIGIOT	硬件故障
SIGKILL	终止
SIGLWP	线程库内部使用
SIGPIPE	写至无读进程的管道
SIGPOLL	可轮询事件 (poll)
SIGPROF	梗概时间超时 (setitimer)
SIGPWR	电源失效/重启动

注意：在使用信号的时候尽量使用信号名，而不要使用信号编号，这是因为有些信号在不同的系统中的编号可能是不同的

=====

当某一个进程收到一个信号时，进程对于信号的处理有三种动作：

忽略该信号

执行默认动作，下图为默认动作的具体类型（注意，默认动作的 Ign 和忽略信号虽然处理好像是一样的，但是在不同的层次上），通过 man 7 signal 这个命令可以查看各个信号的默认动作

```
Term   Default action is to terminate the process.
Ign     Default action is to ignore the signal.
Core   Default action is to terminate the process and dump core (see core(5)).
Stop    Default action is to stop the process.
Cont    Default action is to continue the process if it is currently stopped.
```

捕捉该信号

注意：

有两个信号比较特殊：SIGKILL 和 SIGSTOP 信号。对于这两个信号，进程收到后既不能忽略这两种信号，也不能捕捉这两种信号，只能执行其默认动作（终止进程）。之所以对于这两种信号采用特殊的方法，是为了保证 root 用户始终能够终止一个进程，如果一个进程能够忽略所有的信号，那么将无法通过信号的方法来杀死该进程。

几个常用的信号：

SIGINT：通过按下 `ctrl+c` 来产生

SIGTSTP：通过按下 `ctrl+z` 来产生

SIGQUIT：通过按下 `ctrl+\` 来产生

SIGTERM：通过命令 `kill` 默认产生

SIGABRT：调用`abort`函数来产生

SIGALRM：调用`alarm`函数来产生

SIGCHLD：子进程终止或者暂停时，内核发送给父进程