

- 1) SIGHUP: 当用户退出 shell 时, 由该 shell 启动的所有进程将收到这个信号, 默认动作为终止进程。
- 2) SIGINT: 当用户按下了 <Ctrl+C> 组合键时, 用户终端向正在运行中的由该终端启动的程序发出此信号。默认动作为终止进程。
- 3) SIGQUIT: 当用户按下 <ctrl+\> 组合键时产生该信号, 用户终端向正在运行中的由该终端启动的程序发出此信号。默认动作为终止进程。
- 4) SIGILL: CPU 检测到某进程执行了非法指令。默认动作为终止进程并产生 core 文件
- 5) SIGTRAP: 该信号由断点指令或其他 trap 指令产生。默认动作为终止进程 并产生 core 文件。
- 6) SIGABRT: 调用 abort 函数时产生该信号。默认动作为终止进程并产生 core 文件。
- 7) SIGBUS: 非法访问内存地址, 包括内存对齐出错, 默认动作为终止进程并产生 core 文件。
- 8) SIGFPE: 在发生致命的运算错误时发出。不仅包括浮点运算错误, 还包括溢出及除数为 0 等所有的算法错误。默认动作为终止进程并产生 core 文件。
- 9) SIGKILL: 无条件终止进程。本信号不能被忽略, 处理和阻塞。默认动作为终止进程。它向系统管理员提供了可以杀死任何进程的方法。
- 10) SIGUSE1: 用户定义 的信号。即程序员可以在程序中定义并使用该信号。默认动作为终止进程。
- 11) SIGSEGV: 指示进程进行了无效内存访问。默认动作为终止进程

并产生 core 文件。

12) SIGUSR2: 这是另外一个用户自定义信号，程序员可以在程序中定义 并使用该信号。默认动作为终止进程。

13) SIGPIPE: Broken pipe 向一个没有读端的管道写数据。默认动作为终止进程。

14) SIGALRM: 定时器超时，超时的时间 由系统调用 alarm 设置。默认动作为终止进程。

15) SIGTERM: 程序结束信号，与 SIGKILL 不同的是，该信号可以被阻塞和终止。通常用来要示程序正常退出。执行 shell 命令 Kill 时，缺省产生这个信号。默认动作为终止进程。

16) SIGCHLD: 子进程结束时，父进程会收到这个信号。默认动作为忽略这个信号。

17) SIGCONT: 停止进程的执行。信号不能被忽略，处理和阻塞。默认动作为终止进程。

18) SIGTTIN: 后台进程读终端控制台。默认动作为暂停进程。

19) SIGTSTP: 停止进程的运行。按下<ctrl+z>组合键时发出这个信号。默认动作为暂停进程。

21) SIGTTOU: 该信号类似于 SIGTTIN，在后台进程要向终端输出数据时发生。默认动作为暂停进程。

22) SIGURG: 套接字上有紧急数据时，向当前正在运行的进程发出些信号，报告有紧急数据到达。如网络带外数据到达，默认动作为忽略该信号。

23) SIGXFSZ: 进程执行时间超过了分配给该进程的 CPU 时间，系统产生该信号并发送给该进程。默认动作为终止进程。

24) SIGXFSZ: 超过文件的最大长度设置。默认动作为终止进程。

25) SIGVTALRM: 虚拟时钟超时时产生该信号。类似于 SIGALRM，但是该信号只计算该进程占用 CPU 的使用时间。默认动作为终止进程。

26) SIGPROF: 类似于 SIGVTALRM，它不仅包括该进程占用 CPU 时间还包括执行系统调用时间。默认动作为终止进程。

27) SIGWINCH: 窗口变化大小时发出。默认动作为忽略该信号。

28) SIGIO: 此信号向进程指示发出了一个异步 IO 事件。默认动作为忽略。

29) SIGPWR: 关机。默认动作为终止进程。

30) SIGSYS: 无效的系统调用。默认动作为终止进程并产生 core 文件。

31) SIGRTMIN~(64) SIGRTMAX: LINUX 的实时信号，它们没有固定的含义（可以由用户自定义）。所有的实时信号的默认动作都为终止进程。