

## 守护进程再次分裂子进程(僵尸进程的处理)

一个僵尸进程产生的过程是：父进程调用 `fork` 创建子进程后，子进程运行直至其终止，它立即从内存中移除，但进程描述符仍然保留在内存中（进程描述符占有极少的内存空间）。子进程的状态变成 `EXIT_ZOMBIE`，并且向父进程发送 `SIGCHLD` 信号，父进程此时应该调用 `wait()` 系统调用来获取子进程的退出状态以及其它的信息。在 `wait` 调用之后，僵尸进程就完全从内存中移除。因此一个僵尸存在于其终止到父进程调用 `wait` 等函数这个时间的间隙，一般很快就消失，但如果编程不合理，父进程从不调用 `wait` 等系统调用来收集僵尸进程，那么这些进程会一直存在内存中。

编译并运行程序

10 个子进程生成完 暂停

```
[root@RHEL-zby 03]# 116259 116260 1551265 sub
116259 116261 1551265 sub
116259 116263 1551265 sub
116259 116264 1551265 sub
116259 116266 1551265 sub
116259 116268 1551265 sub
116259 116269 1551265 sub
116259 116271 1551265 sub
116259 116272 1551265 sub
116259 116260 1551265 sub
116259 116274 1551265 sub
116259 116261 1551265 sub
1 116259 1551265 main
116259 116263 1551265 sub
116259 116264 1551265 sub
1 116259 1551265 main
116259 116266 1551265 sub
116259 116268 1551265 sub
1 116259 1551265 main
116259 116269 1551265 sub
1 116259 1551265 main
116259 116271 1551265 sub
116259 116272 1551265 sub
116259 116260 1551265 sub
1 116259 1551265 main
116259 116274 1551265 sub
```

查看僵尸进程 此时子进程还未退出 不存在僵尸进程

```
[root@RHEL-zby 03]# ps aux | grep -w 'Z'
root      20451  0.0  0.0 114836 1012 pts/5    S+   22:23   0:00 grep --color=auto -w Z
[root@RHEL-zby 03]#
```

```
ps -a -l
F S      UID      PID      PPID    C  PRI   NI  ADDR  SZ  WCHAN  TTY          TIME CMD
1 S        0 116374        1    0   80    0 -   1042 hrtime pts/1      00:00:00 test3-1
1 S        0 116375 116374    0   80    0 -   1042 hrtime pts/1      00:00:00 test3-1
1 S        0 116376 116374    0   80    0 -   1042 hrtime pts/1      00:00:00 test3-1
1 S        0 116378 116374    0   80    0 -   1042 hrtime pts/1      00:00:00 test3-1
1 S        0 116379 116374    0   80    0 -   1042 hrtime pts/1      00:00:00 test3-1
1 S        0 116381 116374    0   80    0 -   1042 hrtime pts/1      00:00:00 test3-1
1 S        0 116383 116374    0   80    0 -   1042 hrtime pts/1      00:00:00 test3-1
1 S        0 116384 116374    0   80    0 -   1042 hrtime pts/1      00:00:00 test3-1
1 S        0 116386 116374    0   80    0 -   1042 hrtime pts/1      00:00:00 test3-1
1 S        0 116387 116374    0   80    0 -   1042 hrtime pts/1      00:00:00 test3-1
1 S        0 116389 116374    0   80    0 -   1042 hrtime pts/1      00:00:00 test3-1
0 R        0 116394 116039    0   80    0 - 37228 -      pts/1      00:00:00 ps
[11] 完成 /test2-1
```

继续运行到子进程退出  
可以观察到程序按照预期执行  
子进程的父进程是守护进程

```
1 116259 1551265 main
116259 116269 1551265 sub
1 116259 1551265 main
116259 116271 1551265 sub
116259 116272 1551265 sub
1 116259 1551265 main
116259 116274 1551265 sub
1 116259 1551265 main
1 116259 1551265 main
1 116259 1551265 main
1 116259 1551265 main
1 116259 1551265 main
1 116259 1551265 main
1 116259 1551265 main
1 116259 1551265 main
1 116259 1551265 main
1 116259 1551265 main
```

查看僵尸进程 生成的 10 个子进程为僵尸进程

```
ps aux | grep -w 'Z'
root    116260  0.0  0.0      0   0 pts/1    Z   13:16   0:00 [test3-1] <defunct>
root    116261  0.0  0.0      0   0 pts/1    Z   13:16   0:00 [test3-1] <defunct>
root    116263  0.0  0.0      0   0 pts/1    Z   13:16   0:00 [test3-1] <defunct>
root    116264  0.0  0.0      0   0 pts/1    Z   13:16   0:00 [test3-1] <defunct>
root    116266  0.0  0.0      0   0 pts/1    Z   13:16   0:00 [test3-1] <defunct>
root    116268  0.0  0.0      0   0 pts/1    Z   13:16   0:00 [test3-1] <defunct>
root    116269  0.0  0.0      0   0 pts/1    Z   13:16   0:00 [test3-1] <defunct>
root    116271  0.0  0.0      0   0 pts/1    Z   13:16   0:00 [test3-1] <defunct>
root    116272  0.0  0.0      0   0 pts/1    Z   13:16   0:00 [test3-1] <defunct>
root    116274  0.0  0.0      0   0 pts/1    Z   13:16   0:00 [test3-1] <defunct>
root    116304  0.0  0.2 114836 1008 pts/1    S+  13:18   0:00 grep --color=auto -w Z
[root@RHEL-zby 03]# 1 116259 1551265 main
```

只需要把其父进程杀死 僵尸进程即可被 init 进程收尸

```
kill 11631 116374 1551265 main
74
[root@RHEL-zby 03]# ps -a -l
F S  UID      PID  PPID  C PRI  NI ADDR SZ WCHAN  TTY          TIME CMD
0 R    0 116415 116039  0  80   0 - 37227 -          pts/1    00:00:00 ps
[root@RHEL-zby 03]# ps aux | grep -w 'Z'
root    116418  0.0  0.2 114836 1012 pts/1    R+  13:24   0:00 grep --color=auto -w Z
[root@RHEL-zby 03]#
```

子进程在终止时会向父进程发 SIGCHLD 信号, Linux 默认是忽略该信号的, 我们可以显示安装该信号, 在信号处理函数中调用 wait 等函数来为其收尸, 这样就能避免僵尸进程长期存在于系统中, 修改后的 test3-2.c 便实现了这个功能  
在子程序全部结束后 我们暂停查看僵尸进程 发现并没有产生僵尸进程

```
1 116631 1551265 main
1 116631 1551265 main
1 116631 1551265 main
1 116631 1551265 main
1 116631 1551265 main
1 116631 1551265 main
1 116631 1551265 main
1 116631 1551265 main
1 116631 1551265 main
ps 1 116631 1551265 main
-a -l
F S UID PID PPID C PRI NI ADDR SZ WCHAN TTY TIME CMD
1 S 0 116631 1 0 80 0 - 1042 hrtime pts/1 00:00:00 test3-2
0 R 0 116664 116039 0 80 0 - 37228 - pts/1 00:00:00 ps
[1]+ 完成 ./test3-2
[root@RHEL-zby 03]#
```

```
ps aux |grep -w 'Z' 1 116631 1551265 main

root 116671 0.0 0.2 114836 1016 pts/1 S+ 13:35 0:00 grep --color=auto -w Z
[root@RHEL-zby 03]#
```

本题原来一直将输出重定向到文件做的  
可以避免输出的信息影响输入命令  
但是由于题目要求输出到屏幕只好全部重做