

第 00 章作业 - Linux 知识补充 - 守护进程的编写及使用方法 04

1652817 钟钰琛 计算机科学与技术

2018 年 10 月 17 日

1. 这个程序和 test3-2 不同在于子进程也是死循环.
2. 杀死一个子进程直接 kill -9 进程号

```
root      20114  20078  0 03:47 ?        00:00:00 ./test4-1
root      20127  20078  0 03:47 ?        00:00:00 ./test4-1
root      20130  20078  0 03:47 ?        00:00:00 ./test4-1
root      20140  20078  0 03:47 ?        00:00:00 ./test4-1
root      20148  20078  0 03:47 ?        00:00:00 ./test4-1
root      20151  20078  0 03:47 ?        00:00:00 ./test4-1
root@vm-linux [03时51分27秒] [~/1652817-000106/04]
-> # ps -ef|grep test4-1|grep -v grep|cut -c 9-15
20078
20079
20086
20091
20098
20114
20127
20130
20140
20148
20151
root@vm-linux [03时51分30秒] [~/1652817-000106/04]
> # kill -9 20079
root@vm-linux [03时53分42秒] [~/1652817-000106/04]
-> # ps -ef|grep test4-1|grep -v grep|cut -c 9-15
20078
20086      这边没有20079号子进程了
20091
20098
20114
20127
20130
20140
20148
20151
root@vm-linux [03时54分14秒] [~/1652817-000106/04]
-> #
```

图 1: kill 一个子进程

3. 快速杀死子进程: 注意-P 参数是为了防止删除父进程.



```
root@vm-linux [04时02分55秒] [~/1652817-000106/04]
-> # ps -ef|grep test4-1|grep -v grep|cut -c 9-15
20995
20996
21003
21043
21058
21062
21065
21068
21072
21075
21079
root@vm-linux [04时02分56秒] [~/1652817-000106/04]
-> # ps -ef|grep test4-1|grep -v grep|awk '{print "pkill -9 -P" $2}' | sh
root@vm-linux [04时03分35秒] [~/1652817-000106/04]
-> # ps -ef|grep test4-1|grep -v grep|cut -c 9-15
20995
root@vm-linux [04时03分39秒] [~/1652817-000106/04]
-> #
```

图 2: 批量删除子进程

4. 如果杀死父进程 test4-1, 那么其所有子进程都变成孤儿进程, 被 1 号进程收养. 也就是父进程号 ppid 变为 1.

```

root@vm-linux [04时06分41秒] [~/1652817-000106/04]
-> # ps -ef|grep test4-1
root      21363      1  0 04:05 ?        00:00:00 ./test4-1
root      21364    21363  0 04:05 ?        00:00:00 ./test4-1
root      21371    21363  0 04:05 ?        00:00:00 ./test4-1
root      21374    21363  0 04:05 ?        00:00:00 ./test4-1
root      21378    21363  0 04:05 ?        00:00:00 ./test4-1
root      21381    21363  0 04:05 ?        00:00:00 ./test4-1
root      21385    21363  0 04:05 ?        00:00:00 ./test4-1
root      21388    21363  0 04:05 ?        00:00:00 ./test4-1
root      21391    21363  0 04:05 ?        00:00:00 ./test4-1
root      21395    21363  0 04:05 ?        00:00:00 ./test4-1
root      21398    21363  0 04:05 ?        00:00:00 ./test4-1
root      21461   16752  0 04:06 pts/0    00:00:00 grep --color=auto --exclude-dir=.bzip --excl
exclude-dir=CVS --exclude-dir=.git --exclude-dir=.hg --exclude-dir=.svn test4-1
root@vm-linux [04时06分46秒] [~/1652817-000106/04]
-> # kill -9 21363
root@vm-linux [04时06分53秒] [~/1652817-000106/04]
-> # ps -ef|grep test4-1
root      21364      1  0 04:05 ?        00:00:00 ./test4-1
root      21371      1  0 04:05 ?        00:00:00 ./test4-1
root      21374      1  0 04:05 ?        00:00:00 ./test4-1
root      21378      1  0 04:05 ?        00:00:00 ./test4-1
root      21381      1  0 04:05 ?        00:00:00 ./test4-1
root      21385      1  0 04:05 ?        00:00:00 ./test4-1
root      21388      1  0 04:05 ?        00:00:00 ./test4-1
root      21391      1  0 04:05 ?        00:00:00 ./test4-1
root      21395      1  0 04:05 ?        00:00:00 ./test4-1
root      21398      1  0 04:05 ?        00:00:00 ./test4-1
root      21479   16752  0 04:06 pts/0    00:00:00 grep --color=auto --exclude-dir=.bzip --excl
exclude-dir=CVS --exclude-dir=.git --exclude-dir=.hg --exclude-dir=.svn test4-1
root@vm-linux [04时06分55秒] [~/1652817-000106/04]
-> #

```

图 3: 孤儿进程

5. 如果要求杀死父进程的同时让所有子进程退出，可以通过让父进程退出时向子进程发送信号实现。

```

40
41 // change the current working directory
42 if((chdir("/") < 0)){
43     exit(EXIT_FAILURE);
44 }
45
46 // close out the standard file descriptors
47 /*
48 int x;
49 for(x = sysconf(_SC_OPEN_MAX); x >= 0; x--)
50     close(x);
51
52 openlog("zyc_daemon", LOG_PID, LOG_DAEMON);*/
53 }
54 int main(){
55     create_daemon();
56
57     int i;
58     for (i = 0; i < 10; ++i){
59         pid_t pid = fork();
60         if (pid == 0){
61             prctl(PR_SET_PDEATHSIG, SIGHUP);
62
63             while(1){
64                 printf("%d %d 1652817 sub\n", (int)getppid(), (int)get
pid());
65                 sleep(15);
66             }
67             exit(EXIT_SUCCESS);
68         }
69         sleep(3);
70     }
71
72
-- 插入 --
67,23-44 88%

```

图 4: prctl

```
root@vm-linux [04时19分50秒] [~/1652817-000106/04]
-> # ps -ef|grep test4-2
root      22105      1  0 04:18 ?        00:00:00 ./test4-2
root      22106    22105  0 04:18 ?        00:00:00 ./test4-2
root      22113    22105  0 04:18 ?        00:00:00 ./test4-2
root      22126    22105  0 04:18 ?        00:00:00 ./test4-2
root      22129    22105  0 04:18 ?        00:00:00 ./test4-2
root      22136    22105  0 04:18 ?        00:00:00 ./test4-2
root      22140    22105  0 04:18 ?        00:00:00 ./test4-2
root      22148    22105  0 04:18 ?        00:00:00 ./test4-2
root      22152    22105  0 04:18 ?        00:00:00 ./test4-2
root      22155    22105  0 04:18 ?        00:00:00 ./test4-2
root      22158    22105  0 04:18 ?        00:00:00 ./test4-2
root      22345    14260  0 04:19 pts/1    00:00:00 grep --color=auto --exclude-dir=.bzip --
exclude-dir=CVS --exclude-dir=.git --exclude-dir=.hg --exclude-dir=.svn test4-2
root@vm-linux [04时19分52秒] [~/1652817-000106/04]
-> # kill -9 22105
root@vm-linux [04时19分57秒] [~/1652817-000106/04]
-> # ps -ef|grep test4-2
root      22360    14260  0 04:19 pts/1    00:00:00 grep --color=auto --exclude-dir=.bzip --
exclude-dir=CVS --exclude-dir=.git --exclude-dir=.hg --exclude-dir=.svn test4-2
root@vm-linux [04时19分59秒] [~/1652817-000106/04]
-> #
```

图 5: 父进程退出时子进程也自动退出