

利用守护进程证明某进程收到 kill -9

1652195 肖潇灏

- 创建一个守护进程，然后 fork 出一个子进程：

```
pid=fork(); //创建子进程

if(pid<0)
{
    printf("fork error\n");
    exit(1);
}
else if(pid>0){
    exit(0); //退出父进程
}
//下面使子进程成为守护进程
setsid(); //创建新会话
chdir("/"); //改变当前工作目录为根目录
umask(0); //设置文件权限掩码

pid=fork();
```

- 然后在守护进程中，使用 waitpid 函数等待回收子进程，并带一个 status 值代表子进程被回收时的状态：

```
if(pid>0){
    int status=0,ret;
    while(1){
        sleep(1);
        ret=waitpid(pid,&status,WNOHANG);
        if(ret){
            printf("%d号进程收到: %d号信号\n",pid,status);
            break;
        }
    }
}
```

- 当子进程被 kill -9 时，父进程就能够回收这个子进程，并且得到的 status 值是子进程收到的信号值：

```
[root@vm-linux test]# gcc -o test_sig test_sig.c
[root@vm-linux test]# ./test_sig
[root@vm-linux test]# kill -9 2340
[root@vm-linux test]# 2340号进程收到: 9号信号
[root@vm-linux test]#
```

此时父进程便得知子进程收到了 kill -9 信号，并能够做出相应的处理