

第 00 章作业 - Linux 知识补充 - 守护进程的编写及使用方法 02

1652817 钟钰琛 计算机科学与技术

2018 年 10 月 17 日

1. 直接执行 ./test2，成为守护进程

```
root@vm-linux [01时46分23秒] [~/1652817-000106/02]
-> # make
gcc -c -o test2.o test2.c
gcc -o test2 test2.o
root@vm-linux [01时46分24秒] [~/1652817-000106/02]
-> # ll
总用量 24K
-rw-r--r-- 1 root root 59 10月 17 00:04 makefile
-rwxr-xr-x 1 root root 8.8K 10月 17 01:46 test2
-rw-r--r-- 1 root root 1012 10月 17 01:45 test2.c
-rw-r--r-- 1 root root 2.6K 10月 17 01:46 test2.o
root@vm-linux [01时46分26秒] [~/1652817-000106/02]
-> # ./test2
root@vm-linux [01时46分29秒] [~/1652817-000106/02]
-> # ps -xj | grep test2
1 12249 12248 12248 ? -1 S 0 0:00 ./test2
11211 12262 12261 11215 pts/1 1226 S+ 0 0:00 grep --color=auto --exclude
-dir=.bzr --exclude-dir=.CVS --exclude-dir=.git --exclude-dir=.hg --exclude-dir=.svn test
root@vm-linux [01时46分38秒] [~/1652817-000106/02]
-> #
```

PPID PID PGID SID TTY TPGID

图 1: ./test2

(输出我输出到 syslog 里面了，所以在图中没有显示)

2. 输入

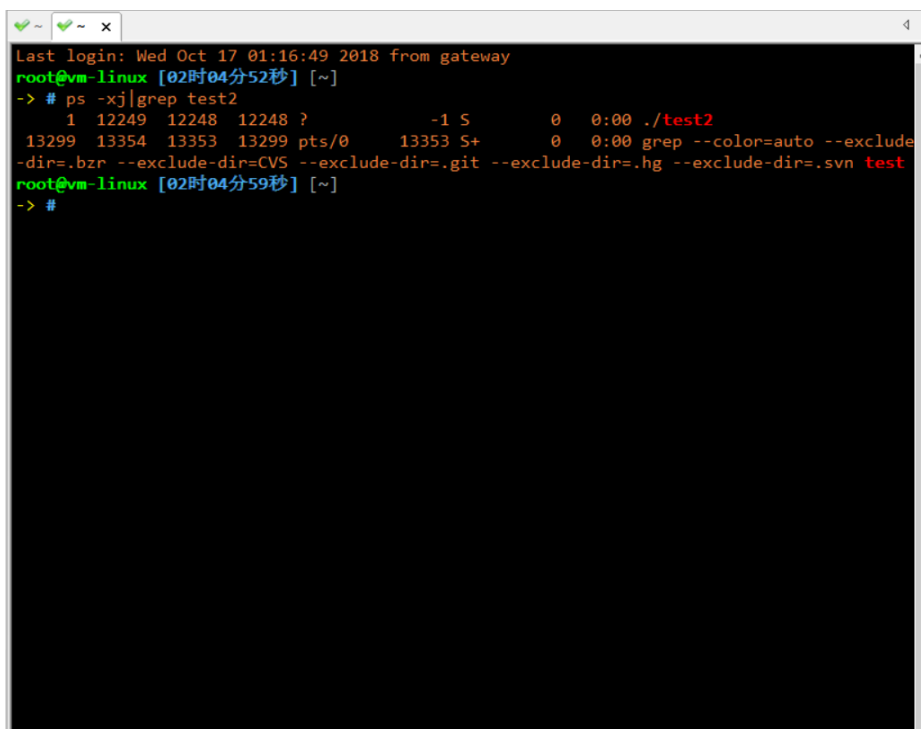
```
ps -xj | grep test2
```

查看进程信息

观察图 1，守护进程有这么几点特征

- 守护进程没有终端 `tty = ?`
- 父进程 (ppid) 是 1(init process)
- `pid != sid` 因为 `fork` 了两次

3. 在另一个控制台依然能看到 `test2` 进程信息，这时 `test2` 成为了孤儿进程，因为父进程先结束了，于是被 1 号进程收养。

A terminal window titled 'root@vm-linux [02时04分52秒] [~]' showing the command 'ps -xj | grep test2'. The output displays two lines of process information. The first line shows PID 1, PPID 12249, PID 12248, TTY '?', and command './test2'. The second line shows PID 13299, PPID 13354, PID 13353, TTY 'pts/0', and command 'grep --color=auto --exclude-dir=.bzip --exclude-dir=CVS --exclude-dir=.git --exclude-dir=.hg --exclude-dir=.svn test'. The terminal window has a title bar with window control buttons and a tab labeled 'x'.

```
Last login: Wed Oct 17 01:16:49 2018 from gateway
root@vm-linux [02时04分52秒] [~]
-> # ps -xj | grep test2
  1 12249 12248 12248 ?          -1 S      0  0:00 ./test2
13299 13354 13353 13299 pts/0    13353 S+     0  0:00 grep --color=auto --exclude
-dir=.bzip --exclude-dir=CVS --exclude-dir=.git --exclude-dir=.hg --exclude-dir=.svn test
root@vm-linux [02时04分59秒] [~]
-> #
```

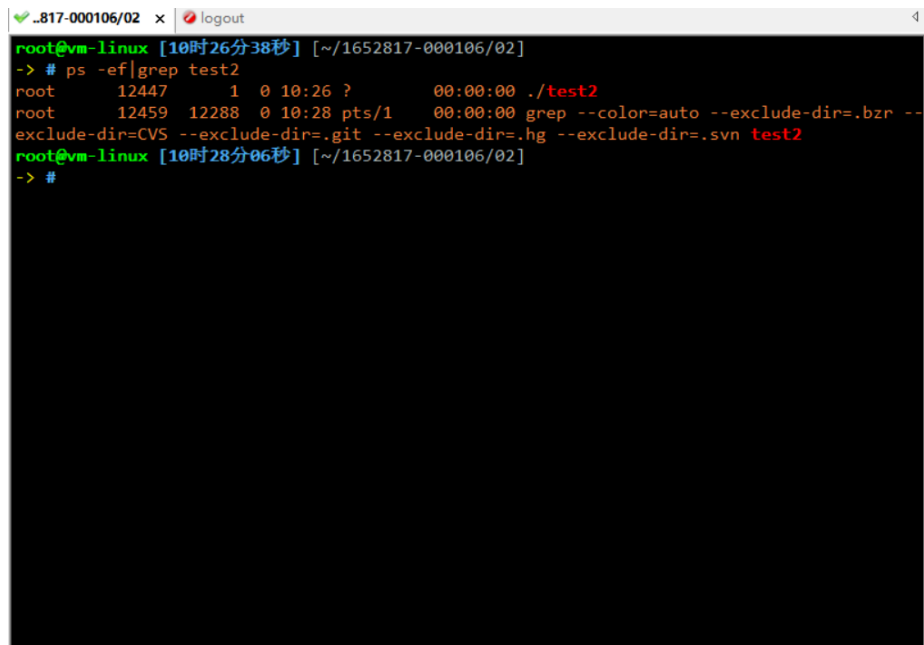
图 2: 另一控制台查看 `test2` 进程信息

4. `ctrl+D` logout 后，进程仍在运行，这一点可以根据重新登陆控制台确认

```
✓ ~ x
Last login: Wed Oct 17 02:04:52 2018 from gateway
root@vm-linux [02时19分16秒] [~]
-> # ps -xj |grep test2
    1  14186  14185  14185 ?          -1 S        0   0:00 ./test2
14260 14313  14312  14260 pts/1      14312 S+       0   0:00 grep --color=auto --exclude-dir=.b
root@vm-linux [02时19分21秒] [~]
-> #
```

图 3: 另一控制台查看 test2 进程信息

因为之前是输出到日志，为了确认打印信息是否会在另一台控制台出现，我又重新运行了./test2 输出到屏幕，logout，
登陆另一台控制台，发现并没有打印信息



```
root@vm-linux [10时26分38秒] [~/1652817-000106/02]
-> # ps -ef|grep test2
root      12447      1  0 10:26 ?          00:00:00 ./test2
root      12459  12288  0 10:28 pts/1      00:00:00 grep --color=auto --exclude-dir=.bzip --
exclude-dir=CVS --exclude-dir=.git --exclude-dir=.hg --exclude-dir=.svn test2
root@vm-linux [10时28分06秒] [~/1652817-000106/02]
-> #
```

图 4: 另一台控制台没有打印信息

5. 引申问题：应该像我下面代码一样写进日志里面。
输出日志在/var/log/messages 里面

```

1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <signal.h>
4 #include <unistd.h>
5 #include <sys/types.h>
6 #include <sys/stat.h>
7 #include <syslog.h>
8 void create_daemon(){
9     // fork off the parent process
10    pid_t pid = fork();
11    if(pid < 0){
12        exit(EXIT_FAILURE);
13    }
14
15    // exit the parent process
16    if(pid > 0){
17        exit(EXIT_SUCCESS);
18    }
19
20    // create a new sid for the child process
21    pid_t sid = setsid();
22    if (sid < 0){
23        exit(EXIT_FAILURE);
24    }
25
26    // catch, ignore and handle signals
27    signal(SIGCHLD, SIG_IGN);
28    signal(SIGHUP, SIG_IGN);
29
30    pid = fork();
31
32    if(pid < 0){
33        exit(EXIT_FAILURE);
34    }

```

1.1 顶端

```

29
30    pid = fork();
31
32    if(pid < 0){
33        exit(EXIT_FAILURE);
34    }
35
36    if(pid > 0){
37        exit(EXIT_SUCCESS);
38    }
39
40    // set new file permissions
41    umask(0);
42
43    // change the current working directory
44    if((chdir("/") < 0){
45        exit(EXIT_FAILURE);
46    }
47
48    // close out the standard file descriptors
49    int x;
50    for(x = sysconf(_SC_OPEN_MAX); x >= 0; x--){
51        close(x);
52    }
53    openlog("zyc_daemon", LOG_PID, LOG_DAEMON);
54 }
55 int main(){
56     create_daemon();
57
58     while(1){
59         syslog(LOG_NOTICE, "1652817");
60         sleep(5);
61     }
62

```

62,1 底端

图 5: test2.c

```
root@vm: /var/log x
root@vm: Linux [02时13分34秒] [/var/log]
> # grep zyc_daemon dmesg
root@vm: Linux [02时13分44秒] [/var/log]
> # grep zyc_daemon messages
Oct 17 01:32:33 vm-linux zyc_daemon[11340]: 1652817
Oct 17 01:32:38 vm-linux zyc_daemon[11340]: 1652817
Oct 17 01:32:43 vm-linux zyc_daemon[11340]: 1652817
Oct 17 01:32:48 vm-linux zyc_daemon[11340]: 1652817
Oct 17 01:32:53 vm-linux zyc_daemon[11340]: 1652817
Oct 17 01:32:58 vm-linux zyc_daemon[11340]: 1652817
Oct 17 01:33:03 vm-linux zyc_daemon[11340]: 1652817
Oct 17 01:33:08 vm-linux zyc_daemon[11340]: 1652817
Oct 17 01:33:13 vm-linux zyc_daemon[11340]: 1652817
Oct 17 01:33:18 vm-linux zyc_daemon[11340]: 1652817
Oct 17 01:33:23 vm-linux zyc_daemon[11340]: 1652817
Oct 17 01:33:28 vm-linux zyc_daemon[11340]: 1652817
Oct 17 01:33:33 vm-linux zyc_daemon[11340]: 1652817
Oct 17 01:33:38 vm-linux zyc_daemon[11340]: 1652817
Oct 17 01:33:43 vm-linux zyc_daemon[11340]: 1652817
Oct 17 01:33:48 vm-linux zyc_daemon[11340]: 1652817
Oct 17 01:33:53 vm-linux zyc_daemon[11340]: 1652817
Oct 17 01:33:58 vm-linux zyc_daemon[11340]: 1652817
Oct 17 01:34:03 vm-linux zyc_daemon[11340]: 1652817
Oct 17 01:34:08 vm-linux zyc_daemon[11340]: 1652817
Oct 17 01:34:13 vm-linux zyc_daemon[11340]: 1652817
Oct 17 01:34:18 vm-linux zyc_daemon[11340]: 1652817
Oct 17 01:34:23 vm-linux zyc_daemon[11340]: 1652817
Oct 17 01:34:28 vm-linux zyc_daemon[11340]: 1652817
Oct 17 01:34:33 vm-linux zyc_daemon[11340]: 1652817
Oct 17 01:34:38 vm-linux zyc_daemon[11340]: 1652817
Oct 17 01:34:43 vm-linux zyc_daemon[11340]: 1652817
Oct 17 01:34:48 vm-linux zyc_daemon[11340]: 1652817
Oct 17 01:34:53 vm-linux zyc_daemon[11340]: 1652817
Oct 17 01:34:58 vm-linux zyc_daemon[11340]: 1652817
Oct 17 01:35:03 vm-linux zyc_daemon[11340]: 1652817
Oct 17 01:35:08 vm-linux zyc_daemon[11340]: 1652817
```

图 6: 查看日志