

# 第 00 章作业 - Linux 知识补充 - 守护进程的编写及使用方法 01

1652817 钟钰琛 计算机科学与技术

2018 年 10 月 17 日

## 1 编写程序与 makefile

makefile 如下:

---

```
CC=gcc
TARGET=test1-1 test1-2
all: $(TARGET)
$(TARGET): %: %.c
    $(CC) -o $@ $<

clean:
    rm test1-1 test1-2
```

---

## 2 运行 test1-1

程序进入死循环, 不再显示 #

```
root@vm-linux [22时57分30秒] [~/1652817-000106/01]
-> # make
gcc -o test1-1 test1-1.c
gcc -o test1-2 test1-2.c
root@vm-linux [22时57分31秒] [~/1652817-000106/01]
-> # ll
总用量 36K
-rw-r--r-- 1 root root 108 10月 16 22:51 makefile
-rwxr-xr-x 1 root root 8.3K 10月 16 22:57 test1-1
-rw-r--r-- 1 root root 102 10月 16 22:51 test1-1.c
-rwxr-xr-x 1 root root 8.3K 10月 16 22:57 test1-2
-rw-r--r-- 1 root root 101 10月 16 22:51 test1-2.c
root@vm-linux [22时57分38秒] [~/1652817-000106/01]
-> # ./test1-1
1652817
1652817
1652817
1652817
1652817
1652817
■
```

图 1: 运行 test1-1

### 3 后台运行

1. 如何后台运行?

---

```
nohup ./test1-1 &
```

---

2. 如何查看后台 test1-1 程序的信息

---

```
vim nohup.out
```

---

test1-1.c 中需要加入 `fflush(stdout);` 否则缓冲区没有刷新, `nohup.out` 为空.test1-2.c 同理.

```
root@vm-linux [23时09分15秒] [~/1652817-000106/01]
-> # nohup ./test1-1 &
[1] 16435
nohup: 忽略输入并把输出追加到"nohup.out"
root@vm-linux [23时09分23秒] [~/1652817-000106/01]
-> # ll
总用量 40K
-rw-r--r-- 1 root root 108 10月 16 22:51 makefile
-rw----- 1 root root 112 10月 16 23:09 nohup.out
-rwxr-xr-x 1 root root 8.4K 10月 16 23:05 test1-1
-rw-r--r-- 1 root root 120 10月 16 23:05 test1-1.c
-rwxr-xr-x 1 root root 8.4K 10月 16 23:05 test1-2
-rw-r--r-- 1 root root 119 10月 16 23:05 test1-2.c
root@vm-linux [23时09分34秒] [~/1652817-000106/01]
-> #
```

图 2: 后台运行 test1-1

3. 如何查看在后台的 test1-1 程序的信息  
观察 nohup.out 文件

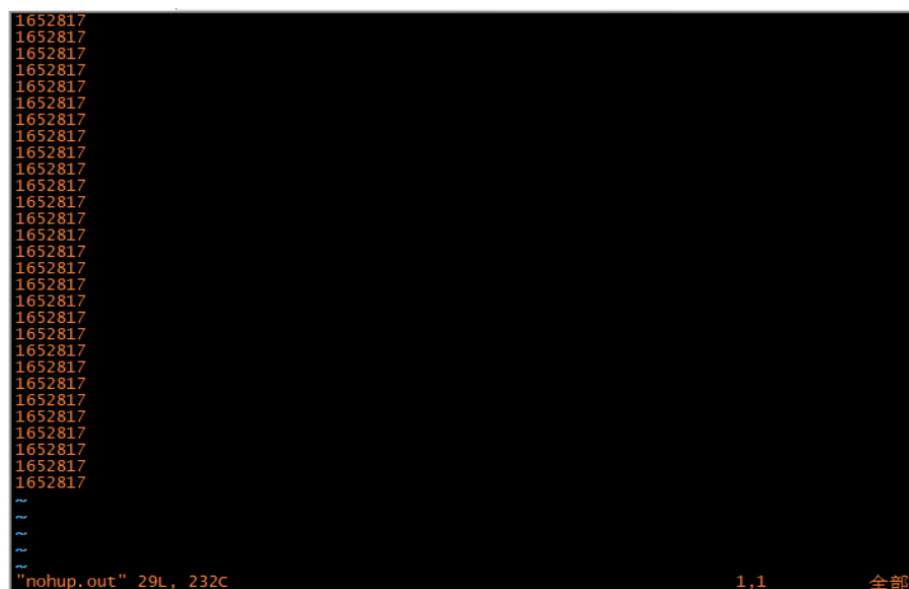


图 3: nohup.out

#### 4. 如何使后台运行的 test1-1 重新回到前台

首先用 `jobs -l` 查看所有运行程序，然后用 `fg %[number]` 把一个程序掉到前台运行

---

```
jobs -l
fg %1
```

---

```
root@vm-linux [23时21分05秒] [~/1652817-000106/01]
-> # jobs -l
[1] + 16720 running    nohup ./test1-1
root@vm-linux [23时21分10秒] [~/1652817-000106/01]
-> # fg %1
[1] + 16720 running    nohup ./test1-1
```

图 4: 重新调回前台运行

5. 再次将 test1-1 放入后台

先输入 `ctrl+z`, 使程序暂停, 然后输入

---

```
jobs -l
bg %1
```

---

对于 test1-2, 先执行, 然后也输入 `ctrl+z`, 使程序暂停, 然后输入

---

```
jobs -l
bg %2
```

---

```
root@vm-linux [23时33分16秒] [~/1652817-000106/01]
-> # jobs -l
[1] + 16930 running  nohup ./test1-1
root@vm-linux [23时33分21秒] [~/1652817-000106/01]
-> # fg %1
[1] + 16930 running  nohup ./test1-1
^Z
[1] + 16930 suspended nohup ./test1-1
root@vm-linux [23时33分33秒] [~/1652817-000106/01]
-> # jobs -l
[1] + 16930 suspended nohup ./test1-1
root@vm-linux [23时33分42秒] [~/1652817-000106/01]
-> # bg %1
[1] + 16930 continued nohup ./test1-1
root@vm-linux [23时33分50秒] [~/1652817-000106/01]
-> # nohup ./test1-2
nohup: 忽略输入并把输出追加到"nohup.out"
^Z
[2] + 16966 suspended nohup ./test1-2
root@vm-linux [23时34分04秒] [~/1652817-000106/01]
-> # jobs -l
[1] - 16930 running  nohup ./test1-1
[2] + 16966 suspended nohup ./test1-2
root@vm-linux [23时34分07秒] [~/1652817-000106/01]
-> # bg %2
[2] - 16966 continued nohup ./test1-2
root@vm-linux [23时34分27秒] [~/1652817-000106/01]
-> # jobs -l
[1] + 16930 running  nohup ./test1-1
[2] - 16966 running  nohup ./test1-2
root@vm-linux [23时34分30秒] [~/1652817-000106/01]
-> #
```

图 5: 再次放到后台

6. 如何将 test1-1/test1-2 中的某一个切换到前台  
输入 fg %1 或者 fg %2 即可
7. 查看父进程

```
root@vm-linux [23时36分51秒] [~/1652817-000106/01]
-> # jobs -l
[1] + 16930 running    nohup ./test1-1
[2] - 16966 running    nohup ./test1-2
root@vm-linux [23时36分59秒] [~/1652817-000106/01]
-> # ps -ef|grep 16930
root    16930  14634  0 23:33 pts/1    00:00:00 ./test1-1
root    16995  14634  0 23:38 pts/1    00:00:00 grep --color=auto --exclude-dir=.bzr --
exclude-dir=CVS --exclude-dir=.git --exclude-dir=.hg --exclude-dir=.svn 16930
root@vm-linux [23时38分55秒] [~/1652817-000106/01]
-> # ps -ef|grep 16966
root    16966  14634  0 23:34 pts/1    00:00:00 ./test1-2
root    17001  14634  0 23:39 pts/1    00:00:00 grep --color=auto --exclude-dir=.bzr --
exclude-dir=CVS --exclude-dir=.git --exclude-dir=.hg --exclude-dir=.svn 16966
root@vm-linux [23时39分05秒] [~/1652817-000106/01]
-> # ps -ef|grep 14634
root     14634  14632  0 22:39 pts/1    00:00:03 -zsh
root     16930  14634  0 23:33 pts/1    00:00:00 ./test1-1
root     16966  14634  0 23:34 pts/1    00:00:00 ./test1-2
root     17006  14634  0 23:39 pts/1    00:00:00 ps -ef
root     17007  14634  0 23:39 pts/1    00:00:00 grep --color=auto --exclude-dir=.bzr --
exclude-dir=CVS --exclude-dir=.git --exclude-dir=.hg --exclude-dir=.svn 14634
root@vm-linux [23时39分56秒] [~/1652817-000106/01]
-> #
```

进程14634是zsh

父进程都是14634

图 6: 父进程

## 8. 用另一个终端登陆

```
root@vm-linux [23时42分51秒] [~]
-> # ps -ef|grep test1-1
root    16930   14634   0  23:33 pts/1    00:00:00 ./test1-1
root    17083   17025   0  23:42 pts/2    00:00:00 grep --color=auto --exclude-dir=.bzr --
exclude-dir=CVS --exclude-dir=.git --exclude-dir=.hg --exclude-dir=.svn test1-1
root@vm-linux [23时42分57秒] [~]
-> # ps -ef|grep test1-2
root    16966   14634   0  23:34 pts/1    00:00:00 ./test1-2
root    17089   17025   0  23:43 pts/2    00:00:00 grep --color=auto --exclude-dir=.bzr --
exclude-dir=CVS --exclude-dir=.git --exclude-dir=.hg --exclude-dir=.svn test1-2
root@vm-linux [23时43分05秒] [~]
-> # ps -ef|grep 14634
root     14634   14632   0  22:29 pts/1    00:00:03 -zsh
root    16930   14634   0  23:33 pts/1    00:00:00 ./test1-1
root    16966   14634   0  23:34 pts/1    00:00:00 ./test1-2
root    17095   17025   0  23:44 pts/2    00:00:00 grep --color=auto --exclude-dir=.bzr --
exclude-dir=CVS --exclude-dir=.git --exclude-dir=.hg --exclude-dir=.svn 14634
root@vm-linux [23时44分55秒] [~]
-> #
```

父进程依然是14634

进程14634也依然是zsh

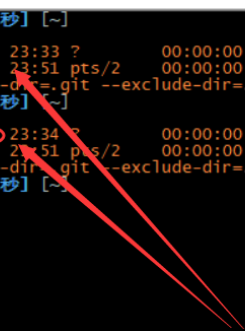
图 7: 另一个终端登陆查看 test1-1/test1-2 信息

9. 退出控制台，再查看

因为父进程死了，所以这些子进程成了孤儿进程，被 1 号进程收养。



```
root@vm-linux [23时51分01秒] [~]
-> # ps -ef|grep test1-1
root      16930      1  0 23:33 ?        00:00:00 ./test1-1
root      17179    17025  0 23:51 pts/2    00:00:00 grep --color=auto --exclude-dir=.bzip --
exclude-dir=CVS --exclude-dir=.git --exclude-dir=.hg --exclude-dir=.svn test1-1
root@vm-linux [23时51分05秒] [~]
-> # ps -ef|grep test1-2
root      16966      1  0 23:34 ?        00:00:00 ./test1-2
root      17185    17025  0 23:51 pts/2    00:00:00 grep --color=auto --exclude-dir=.bzip --
exclude-dir=CVS --exclude-dir=.git --exclude-dir=.hg --exclude-dir=.svn test1-2
root@vm-linux [23时51分12秒] [~]
-> #
```



父进程都变成了1号进程

图 8: 退出控制台, 另一个控制台查看 test1-1/test1-2 信息

10. 能否使得 test1-1/test1-2 在终端退出登陆后继续运行

其实我之前加的 nohup 就可以保证关闭终端后还能继续运行. 因为 nohup 会忽略所有挂断信号.

```
root@vm-linux [23时53分56秒] [~/1652817-000106/01]
-> # ll
总用量 40K
-rw-r--r-- 1 root root 108 10月 16 22:51 makefile
-rw----- 1 root root 3.6K 10月 16 23:53 nohup.out
-rwxr-xr-x 1 root root 8.4K 10月 16 23:05 test1-1
-rw-r--r-- 1 root root 120 10月 16 23:05 test1-1.c
-rwxr-xr-x 1 root root 8.4K 10月 16 23:05 test1-2
-rw-r--r-- 1 root root 119 10月 16 23:05 test1-2.c
root@vm-linux [23时53分59秒] [~/1652817-000106/01]
-> # ll
总用量 40K
-rw-r--r-- 1 root root 108 10月 16 22:51 makefile
-rw----- 1 root root 3.6K 10月 16 23:54 nohup.out
-rwxr-xr-x 1 root root 8.4K 10月 16 23:05 test1-1
-rw-r--r-- 1 root root 120 10月 16 23:05 test1-1.c
-rwxr-xr-x 1 root root 8.4K 10月 16 23:05 test1-2
-rw-r--r-- 1 root root 119 10月 16 23:05 test1-2.c
root@vm-linux [23时54分00秒] [~/1652817-000106/01]
-> #
```

nohup.out还在更新，说明  
test1-1 test1-2仍在运行

图 9: 仍在继续运行

11.

---

```
nohup ./xxxx &
```

---