

## 1 学习报告

这篇报告基于 Beginning Linux Programming 40-60 页内容。主要报告了 trap 命令的学习心得。

### 1.1 关于 trap 命令的学习体会

trap 命令可以将接受到的指令转化为一定的动作。通常，trap 的用途是当代码运行时如果收到外部的干扰，可以更加整洁。

trap 命令的格式是：

**trap** *command signal*

signal 是代码运行时会接受到的信号，command 是代码接受到了指令之后进行的动作。通常我们将 trap 放在我们想要保护的代码之前。

让 trap 命令初始化我们只需要把 command 部分写成-。而为了忽略某一个指令，我们可以将 command 设置为空字符”。

signal 部分的信号我们可以参阅signal manual 中列出来的所有信号。我们可以利用一些常用的信号来测试所学。比如 INT 信号，通常是由按下 Ctrl+C 来发出，还有 HUP，QUIT 等信号。

### 1.2 程序实验

执行以下的代码：

```
#!/bin/bash

trap 'rm -f ./temp/my_tmp_file' INT
#INT信号输入时，我们执行删除my_tmp_file的指令，使得下面的循环终止
echo creating file/tmp/my_tmy_file

date > ./temp/my_tmp_file
#将时间信息写入文件

echo "press interrupt(Ctrl+C) to interrupt ..."
while [ -f ./temp/my_tmp_file ]; do
    echo File exists
```

```
        sleep 1
done
#如果用户不按Ctrl+C, 这个循环就不会停止
```

```
exit 0
```

运行结果如下:

```
creating file/tmp/my_tmy_file
press interrupt to interrupt ...
File exists
File exists
File exists
File exists
^CEscaped!
```

可以看到, 按下 Ctrl+C 的时候跳出了循环。

对比以下的代码:

```
#!/bin/bash

trap 'rm -f ./temp/my_tmp_file' INT
trap QUIT
#QUIT信号不会反映
echo creating file/tmp/my_tmy_file
date > ./temp/my_tmp_file

echo "press interrupt to interrupt ..."
while [ -f ./temp/my_tmp_file ]; do
    echo File exists
    sleep 1
done
echo Escaped!

exit 0
```

运行后结果如下:

```
creating file/tmp/my_tmy_file
press interrupt to interrupt ...
File exists
File exists
File exists
File exists
File exists
File exists
File exists
^\. /new: line 12: 2402 Quit (core dumped) sleep 1
File exists
File exists
File exists
^\. /new: line 12: 2406 Quit (core dumped) sleep 1
File exists
File exists
File exists
^C Escaped!
```

运行结果中，有两行是测试的时候按下 Ctrl+ 按钮，在 trap 中它表示不执行任何动作。可见，只有按下 Ctrl+C 才能跳出循环。但是 Ctrl+Q 却不能跳出循环。

## 2 学到的内容

这次阅读使我学到了很多 linux 系统 shell programming 的很多知识。

shell programming 与我之前所学到的 C 语言和 Python 等内容有相同之处。像 while, case 等的功能与 C 语言十分类似。但是也有很多不同之处，最主要的区别就是用法的不同。而且，作为操作系统的指令，shell programming 的指令更突出了与文件的交互，而不去做数值的编程。

一个令我印象很深的运算符是 Dot Command，它使得同一个 shell 里拥有不同的环境，比 #include 更加方便。我认为它能很好体现 shell 编程的特点，它能够极大地方便计算机使用者，充分体现了 shell 的优势。

trap 命令是一个非常有用的命令，它有利于代码的优化，同时也让程序变得容易控制，可以为意外事件提供一定的解决方法。