

第23章 调试脚本

shell编程最烦人的一项工作是调试问题。有一些方法可以借鉴，但是最好能在问题出现前防止大部分错误，为此应遵循以下规则。

将设计脚本分成几个任务或过程，然后在继续下一步前分别予以测试。

本章内容有：

- 一般错误。
- set命令介绍。

没有比在脚本中发现一个难以察觉的错误更令人头疼的了，然而，随着编程经验不断丰富，查询手段也相应增加。

经常碰到的问题是忘了使用引号或在if语句末尾未加fi。

需要牢记的一点是当shell打印出一个脚本错误后，不要只看那些疑问行。而是要观察整个相关代码段。shell不会对错误进行精确定位，而是在试图结束一个语句时进行错误统计。

23.1 一般错误

23.1.1 循环错误

for、while、until和case语句中的错误是指实际语句段不正确。也许漏写了固定结构中的一个保留字。

下面错误打印信息done，这是一个很好的线索。因为这时知道正在处理一个while语句。回溯脚本段，检查while语句，是否漏写或错写了关键字，如do或者正在使用的条件语句。

```
syntax error near unexpected token 'done'
line 31: ' done'
```

23.1.2 典型的漏写引号

第二个典型错误是漏写引号错误。经常要注意这个问题，因为此错误经常出现。这里给出解决这类错误的唯一方案是在脚本中确保所有引号成对出现。

```
unexpected EOF while looking for '"'
line 36: syntax error
```

当shell打印出错误行后，通常在vi编辑器中查看文件。使用vi的set nu选项调试错误，先进入vi，然后点击<ESC>键，后跟一冒号，再键入set nu <return>，这时给出文本行号，然后进入shell打印错误行。

23.1.3 测试错误

另一个常见错误是在使用-eq语句时忘记在测试条件一边使用数字取值。

如果得到下列错误提示，通常是由于两件事情：需要在变量和方括号间加空格；在方括号里漏写操作符。

```
[ : missing ' ]'
```

23.1.4 字符大小写

经验上讲大多数错误是由于使用变量时大小写保持一致。例如经常在开始定义时用大写，然后在变量调用时用了小写字符，这样难免变量会没有赋值。

23.1.5 for循环

使用for循环时，有时会忘了在循环的列表部分用\$符号，特别是在读取字符串时。

23.1.6 echo

最有用的调试脚本工具是echo命令。一般在可能出现问题的脚本重要部分加入echo命令，例如在变量读取或修改操作其前后加入echo命令。

使用最后状态命令判断命令是否成功，这里需要注意的是，不要使用echo命令后直接加最后状态命令，因为此命令永远为真。

23.2 set命令

set命令可辅助脚本调试。以下是set命令常用的调试选项：

```
set -n 读命令但并不执行。  
set -v 显示读取的所有行。  
set -x 显示所有命令及其参数。
```

将set选项关闭，只需用+替代-。有人总认为+应该为开，而-应为关闭，但实际刚好相反。

可以在脚本开始时将set选项打开，然后在结束时关闭它。或在认为有问题的特殊语句段前后打开及关闭它。

下面看一个例子。以下脚本将名字保存在变量列表中。用户输入名字，for循环循环变量列表查看是否有匹配模式。注意这里在脚本开始时使用了set -x，并在结尾部分关闭它。

```
$ pg error  
#!/bin/sh  
# error  
# set set -x  
set -x  
LIST="Peter Susan John Barry Lucy Norman Bill Leslie"  
echo -n "Enter your Name :"  
read NAME  
  
for LOOP in $LIST  
do  
    if [ "$LOOP" = "$NAME" ]; then  
        echo "you're on the list, you're in"  
        break  
    fi  
done  
# unset set -x  
set +x
```

运行此脚本，给出一个不在列表中的名字，输出如下：

```
$ error
error
+ error
+ LIST=Peter Susan John Barry Lucy Norman Bill Leslie
+ echo -n Enter your Name :
Enter your Name :+ read NAME
Harry
+ [ Peter = Harry ]
+ [ Susan = Harry ]
+ [ John = Harry ]
+ [ Barry = Harry ]
+ [ Lucy = Harry ]
+ [ Norman = Harry ]
+ [ Bill = Harry ]
+ [ Leslie = Harry ]
```

输出显示对变量列表进行循环时所有的比较操作。当读取文件或进行字符串或取值的比较发现问题时，使用set命令是很有必要的。

23.3 小结

跟踪错误的最好方式是亲自查阅脚本，并使用 set命令并加大量的echo语句。