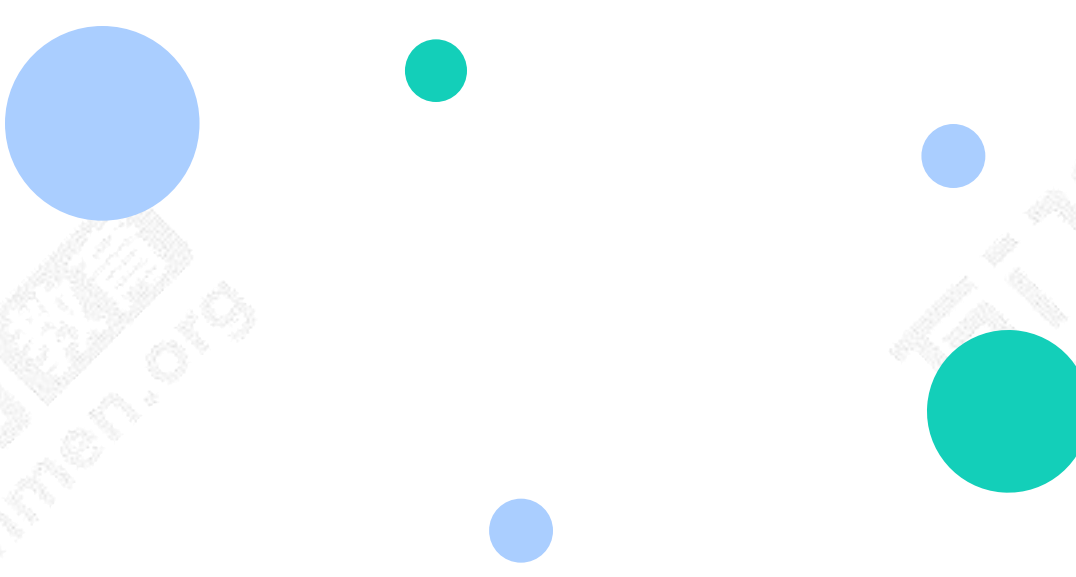


Linux自动化运维特训班

Shell编程初识

讲师：孔丹



大纲

- 为什么学习和使用Shell编程
- 什么是Shell
- 作为程序设计语言的Shell
- 如何学好Shell
- 第一个Shell程序

为什么学习和使用Shell编程

- 对于一个合格的系统管理员来说，学习和掌握Shell编程是非常重要的。通过编程，可以在很大程度上简化日常的维护工作，使得管理员从简单的重复劳动中解脱出来。

Shell程序的特点：

- 1、简单易学。
- 2、解释性语言，不需要编译即可执行

什么是shell

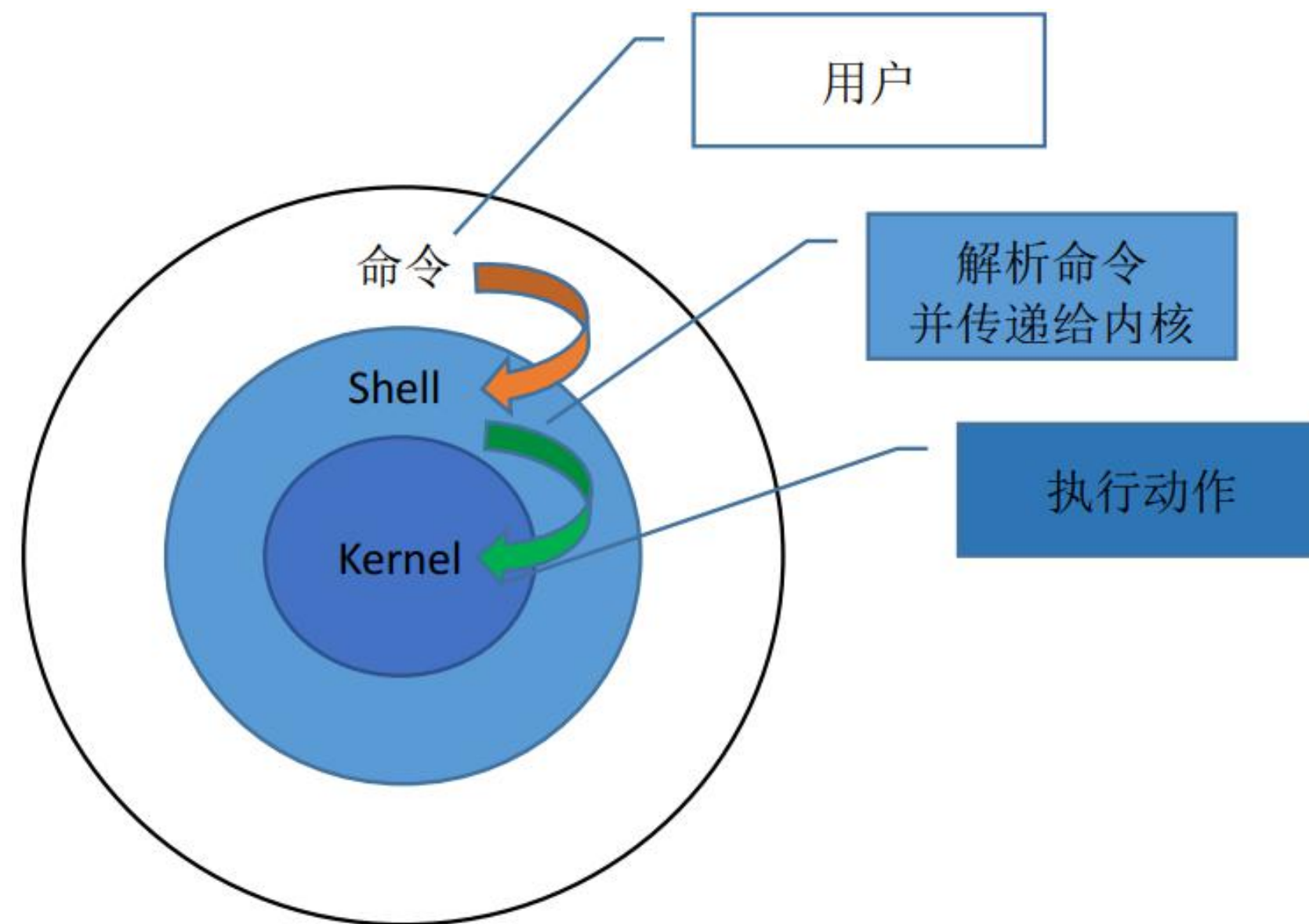
- 在学习Shell编程之前，必须弄清楚什么是Shell。为了能够使读者在学习具体的Shell编程之前对Shell有个基本的了解，本节将对Shell进行概括性的介绍，包括Shell的起源、功能和分类

shell起源

- ❑ 1964年，美国AT&T公司的贝尔实验室、麻省理工学院及美国通用电气公司共同参与开始研发一套可以安装在大型主机上的多用户、多任务的操作系统，该操作系统的名称为Multics。
- ❑ 1970年，丹尼斯·里奇和汤普逊启动了另外一个新的多用户、多任务的操作系统的项目，他们把这个项目称之为UNICS。
- ❑ 1973年，使用C语言重写编写了Unix。通过这次编写，使得Unix得以移植到其他的小型机上面。
- ❑ 1979年，第一个重要的标准UNIX Shell在Unix的第7版中推出，并以作者史蒂夫·伯恩（Stephen Bourne）的名字命名，叫做Bourne Shell，简称为sh。
- ❑ 20世纪70年代末，C Shell作为2BSD UNIX的一部分发布，简称csh。
- ❑ 之后又出现了许多其他的Shell程序，主要包括Tenex C Shell (tcsh)、Korn Shell (ksh) 以及GNU Bourne-Again shell (bash)。

什么是shell

□Shell又称命令解释器，它能识别用户输入的各种命令，并传递给操作系统。它的作用类似于Windows操作系统中的命令行，但是，Shell的功能远比命令行强大的多。在UNIX或者localhost中，Shell既是用户交互的界面，也是控制系统的脚本语言。



shell的分类

- ❑ Bourne Shell: 标识为sh, 该Shell由Steve Bourne在贝尔实验室时编写。在许多Unix系统中, 该Shell是root用户的默认的Shell。
- ❑ Bourne-Again Shell: 标识为bash, 该Shell由Brian Fox在1987年编写, 是绝大多数localhost发行版的默认的Shell。
- ❑ Korn Shell: 标识为ksh, 该Shell由贝尔实验室的David Korn在二十世纪八十年代早期编写。它完全向上兼容Bourne Shell 并包含了C Shell 的很多特性。
- ❑ C Shell: 标识为csh, 该Shell由Bill Joy在BSD系统上开发。由于其语法类似于C语言, 因此称为C Shell。

作为程序设计语言的Shell

- Shell不仅仅是充当用户与UNIX或者localhost交互界面的角色，还可以作为一种程序设计语言来使用。通过Shell编程，可以实现许多非常实用的功能，提高系统管理的自动化水平。本节将介绍作为程序设计语言的Shell的一些特性。
- shell脚本
- 如果有一系列经常需要使用的命令，把它存储在一个文件里，shell可以读取这个文件并顺序执行其中的命令，这样的文件就叫shell脚本。shell脚本按行解释。

如何学好shell

□ 学好shell编程基础知识

- 1、熟练使用vi (vim) 编辑器
- 2、熟练掌握Linux基本命令
- 3、熟练掌握文本三剑客工具 (grep、sed、awk)
- 4、熟悉常用服务器部署、优化、日志及排错

如何学好shell

□ 如何学好shell编程

- 1、掌握Shell脚本基本语法
- 2、形成自己的脚本开发风格
- 3、从简单做起，简单判断，简单循环
- 4、多模仿，多参考资料练习，多思考
- 5、学会分析问题，逐渐形成编程思维
- 6、编程变量名字要规范，采用驼峰语法表示
- 7、不要拿来主义，特别是新手

Shell脚本的基本元素

- 对于一个基本的Shell程序来说，应该拥有以下基本元素：
 - 第1行的“#!/bin/bash”。
 - 注释：说明某些代码的功能。
 - 可执行语句：实现程序的功能。

Shell脚本中的注释和风格

- 通过在代码中增加注释可以提高程序的可读性。
传统的Shell只支持单行注释，其表示方法是一个井号“#”，从该符号开始一直到行尾都属于注释的内容。
例如：
#comment1
#comment2
#comment3
...

Shell脚本中的注释和风格

□ 如何实现多行注释?

用户还可以通过其他的一些变通的方法来实现多行注释，其中，最简单的方法就是使用冒号“:”配合here

document, 语法如下:

```
:<<BLOCK
```

```
....注释内容
```

```
BLOCK
```


Shell脚本编写规范

□ (1) 开头指定脚本解释器

`#!/bin/sh`或`#!/bin/bash`

其他行`#`表示注释

名称见名知义 `backup_mysql.sh`,以`sh`结尾

□ (2) 开头加版本版权等信息

`# Date:创建日期`

`# Author:作者`

`# Mail:联系方式`

`# Function:功能`

`# Version:版本`

□ 3) 脚本中尽量不用中文注释

别吝啬添加注释，必要的注释方便自己别人理解脚本逻辑和功能；

尽量用英文注释，防止本机或切换系统环境后中文乱码的困扰；

单行注释，可以放在代码行的尾部或代码行的上部；

多行注释，用于注解复杂的功能说明，可以放在程序体中，也可以放在代码块的开始部分 针对代码修改时，以及对修改的内容

Shell脚本编写规范

- (4) 多使用内部命令

内部命令可以在性能方面为你节省很多。

- (5) 没有必要使用cat命令

eg: `cat /etc/passwd | grep guru`

使用以下方式即可

eg: `grep guru etc/passwd`

- (6) 代码缩进

- (7) 仔细阅读出错信息

有时候我们修改了某个错误并再次运行后，系统依旧会报错。然后我们再次修改，但系统再次报错。这可能会持续很长时间。但实际上，旧的错误可能已经被纠正，只是由于出现了其它一些新错误才导致系统再次报错

- (8) 脚本以.sh为扩展名

eg: `script-name.sh`

Shell脚本编写规范

使用 .vimrc 文件, 能够快速的生成开头的注释信息

```
[root@ scripts]# cat ~/.vimrc
```

```
autocmd BufNewFile *.py,*.cc,*.sh,*.java exec ":call SetTitle()"
```

```
func SetTitle()
    if expand("%:e") == 'sh'
        call setline(1,"#!/bin/bash")
        call setline(2,
"#####")
        call setline(3, "# File Name: ".expand("%"))
        call setline(4, "# Version: V1.0")
        call setline(5, "# Author: kongd")
        call setline(6, "# Email: kongd@163.com")
        call setline(7, "# Organization: http://www.cnblogs.com/kongd/")
        call setline(8, "# Created Time : ".strftime("%F %T"))
        call setline(9, "# Description:")
        call setline(10,
"#####")
        call setline(11, "")
    endif
endfunc
```

Shell脚本执行

- ❑ 方法一：切换到shell脚本所在的目录（此时，称为工作目录）执行shell脚本
- ❑ 方法二：以绝对路径的方式去执行bash shell脚本：
- ❑ 方法三：直接使用bash 或sh 来执行bash shell脚本：
- ❑ 方法四：在当前的shell环境中执行bash shell脚本：source或.

区别：

- 1、方法三：可以在脚本中不指定解释器，脚本可以没有执行权限
- 2、方法一和方法二脚本需要有执行权限， ./script_name.sh 或 /path/script_name.sh
- 3、方法四：当前shell执行，方法1-3开启子shell

学完后可以达到何种 Shell 编程高度

- 如果具备了前文提到 Linux 基础知识，认真地阅读并按照课程内容去勤加练习，相信很快便可熟练掌握 Shell 编程，搞定企业场景中的绝大多数 Shell 编程问题。

总结

- 1.shell和shell脚本。
- 2.shell脚本编程规范。
- 3.shell脚本执行方式。



谢谢观看

更多好课，请关注[万门大学APP](#)

