



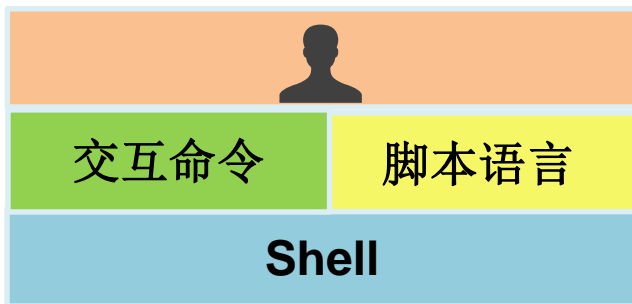
Shell的编程过程

计算机学院 盛剑会



Shell的编程过程

- 在Linux中，Shell不但能与用户进行**逐个命令交互执行**，也可以通过**脚本语言**进行**编程**。
- 通过对Shell进行编程，可使大量任务自动化，对于**系统管理**、**系统维护**方面特别有用。



Shell的编程过程

复习与Shell相关的命令

- 目前无论使用哪种Linux操作系统，默认的Shell总是/bin/bash；
- 查看系统中SHELL变量的值：

echo \$SHELL

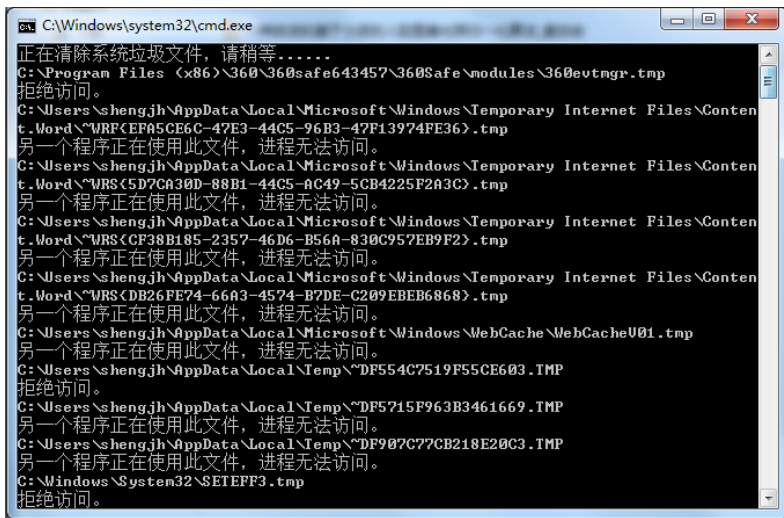
- 查看系统内置了哪些Shell可以使用，有以下两种方法：

① **cat /etc/shells**

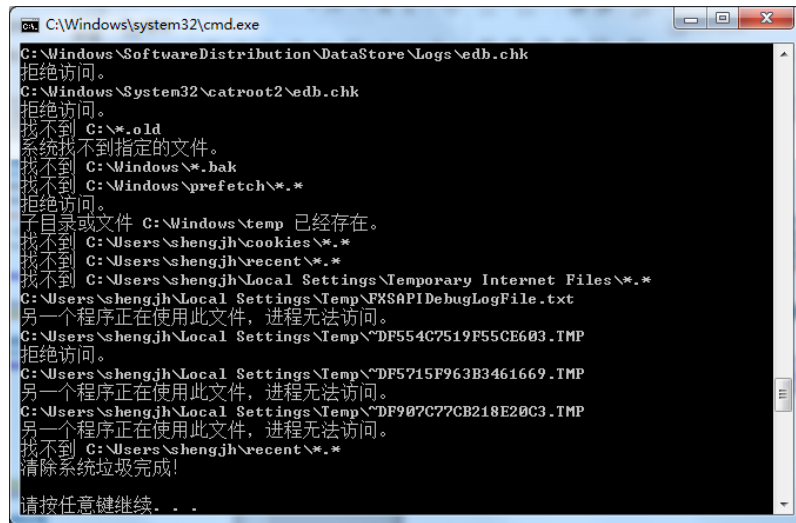
② **chsh -list**

Shell的编程过程

- 与MS-DOS系统中的**批处理文件**类似，Linux中的脚本（Shell程序）是一个**文本文件**，包含各类Linux的Shell命令。通过脚本文件将这些命令汇集在一起，**可连续执行Shell命令**。
- 举例1: [清除系统垃圾.bat](#)



```
C:\Windows\system32\cmd.exe
正在清除系统垃圾文件，请稍等.....
C:\Program Files (x86)\360\360safe\643457\360Safe\modules\360evtngr.tmp
拒绝访问。
C:\Users\shengjh\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.IE5\VRP(EFA5CE6C-47E3-44C5-96B3-47F13974FE36).tmp
另一个程序正在使用此文件，进程无法访问。
C:\Users\shengjh\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.IE5\WRS(5D7CA30D-88B1-44C5-AC49-5CB4225F2A3C).tmp
另一个程序正在使用此文件，进程无法访问。
C:\Users\shengjh\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.IE5\WRS(CF38B185-2357-46D6-B56A-830C957EB9F2).tmp
另一个程序正在使用此文件，进程无法访问。
C:\Users\shengjh\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.IE5\WRS(DB26FE74-66A3-4574-B7DE-C209EBEB6868).tmp
另一个程序正在使用此文件，进程无法访问。
C:\Users\shengjh\AppData\Local\Microsoft\Windows\WebCache\WebCacheV01.tmp
另一个程序正在使用此文件，进程无法访问。
C:\Users\shengjh\AppData\Local\Temp\DF554C7519F55CE603.TMP
拒绝访问。
C:\Users\shengjh\AppData\Local\Temp\DF5715F963B3461669.TMP
另一个程序正在使用此文件，进程无法访问。
C:\Users\shengjh\AppData\Local\Temp\DF907C77CB218E20C3.TMP
另一个程序正在使用此文件，进程无法访问。
C:\Windows\System32\SETEFF3.tmp
拒绝访问。
```



```
C:\Windows\system32\cmd.exe
C:\Windows\SoftwareDistribution\DataStore\Logs\edb.chk
拒绝访问。
C:\Windows\System32\catroot2\edb.chk
拒绝访问。
找不到 C:\*.old
系统找不到指定的文件。
找不到 C:\Windows\*.bak
找不到 C:\Windows\prefetch\*.
拒绝访问。
子目录或文件 C:\Windows\temp 已经存在。
找不到 C:\Users\shengjh\cookies\*.
找不到 C:\Users\shengjh\recent\*.
找不到 C:\Users\shengjh\Local Settings\Temporary Internet Files\*.
C:\Users\shengjh\Local Settings\Temp\FXSAPIDebugLogFile.txt
另一个程序正在使用此文件，进程无法访问。
C:\Users\shengjh\Local Settings\Temp\DF554C7519F55CE603.TMP
拒绝访问。
C:\Users\shengjh\Local Settings\Temp\DF5715F963B3461669.TMP
另一个程序正在使用此文件，进程无法访问。
C:\Users\shengjh\Local Settings\Temp\DF907C77CB218E20C3.TMP
另一个程序正在使用此文件，进程无法访问。
找不到 C:\Users\shengjh\recent\*.
清除系统垃圾完成!
请按任意键继续...
```

Shell的编程过程

- 清除系统垃圾.bat 文件中的内容

```
@echo off
echo 正在清除系统垃圾文件，请稍等.....
del /f /s /q %systemdrive%\*.tmp
del /f /s /q %systemdrive%\*._mp
del /f /s /q %systemdrive%\*.log
del /f /s /q %systemdrive%\*.gid
del /f /s /q %systemdrive%\*.chk
del /f /s /q %systemdrive%\*.old
del /f /s /q %systemdrive%\recycled\*. *
del /f /s /q %windir%\*.bak
del /f /s /q %windir%\prefetch\*. *
rd /s /q %windir%\temp & md %windir%\temp
del /f /q %userprofile%\cookies\*. *
del /f /q %userprofile%\recent\*. *
del /f /s /q "%userprofile%\Local Settings\Temporary Internet Files\*. *"
del /f /s /q "%userprofile%\Local Settings\Temp\*. *"
del /f /s /q "%userprofile%\recent\*. *"
echo 清除系统垃圾完成！
echo. & pause
```

Shell的编程过程

编写Shell程序的步骤

1、创建Shell程序

2、设置执行权限
(可选)

3、执行Shell程序



Shell编程的步骤

举例2：以 “编写一个最简单的 Hello World! 程序” 为例，介绍Shell编程的步骤。

Shell的编程过程

步骤1 创建Shell程序

- 创建一个用于保存Shell脚本文件的目录: `mkdir /usr/program`
 - 在program目录下新建一个文本程序first: `cd /usr/program; touch first`
 - 打开first文件, 输入Shell的脚本: `vim first`
 - 1: **`#!/bin/bash`** #告诉系统, 该文件后面的代码将用/bin/bash来执行; 这是一般Shell程序第1行中必须包含的内容。
 - 2: **`var= "Hello World!"`** #为变量var定义了一个字符串值
 - 3: **`echo $var`** #将变量var的值显示输出到终端
-

Shell的编程过程

步骤2 设置执行权限

- 创建好first文件后，使用`ls -l`命令可以看到文件的所有者（这里是root）对该文件**只有读和写的权限，没有执行权限**。
 - 执行`chmod u+x /usr/program/first`为脚本文件设置所有者的执行权限。当然，也可以执行`chmod a+x /usr/program/first`设置所有用户都可以运行该脚本程序。
 - **注意：**与DOS或Windows中的批处理程序不同，Linux中的脚本文件不是按扩展名来识别，而是通过**为文件添加执行权限**来允许其执行，用户可以为脚本文件设置任意扩展名。
-

Shell的编程过程

步骤3 执行Shell程序

方法一：使用Shell程序执行

- `#bash /usr/program/first`
- 使用这种方法时，不需要设置脚本程序的可执行权限，也即可以跳过步骤2。

方法二：直接执行

- `#!/usr/program/first`
 - 使用这种方法时，需确保该文件已有执行权限。
 - 如果执行当前目录下的脚本文件，通常也应该使用一个点表示当前目录，如执行当前目录下的脚本文件first，可使用命令：`# ./first`
-