

dp 悦知文化
Delight Press

构架、
流程、
管理

实战

Linux Shell 编程与服务器管理

| 系统维护 | 备份还原 | 安全管理 |



卧龙小三

著

荣新IT培训中心

梁昌泰 张琦 黄琨

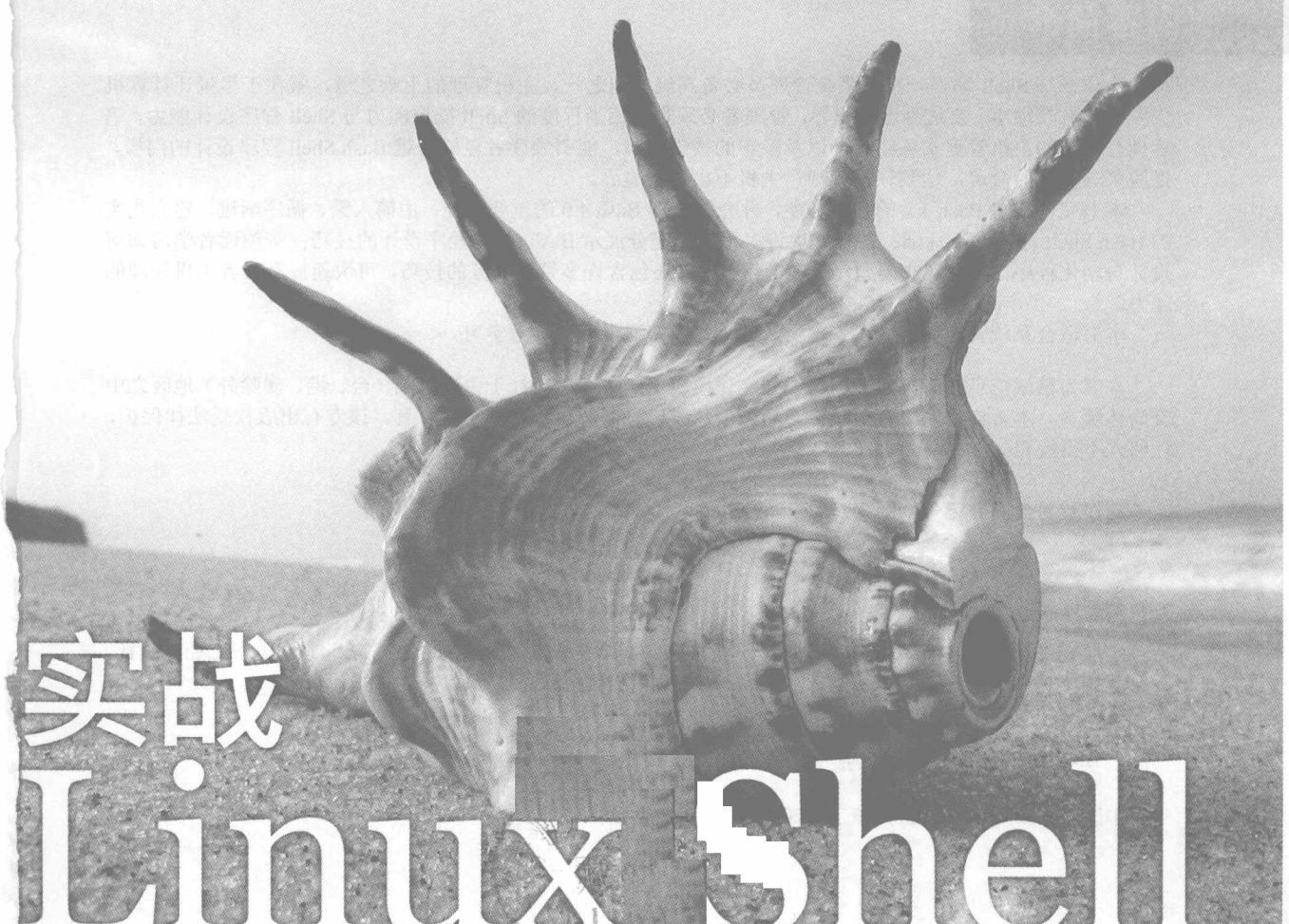
飞思科技产品研发中心

改编
监制



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>



实战

Linux Shell

编程与服务器管理

卧龙小三

著

荣新IT培训中心

梁昌泰 张琦 黄琨

飞思科技产品研发中心

改编

监制

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内容简介

操作系统 Shell 编程一直是系统管理员必备高级技能之一。主机管理的上乘之道，就在于尽量让计算机去完成所有的琐事。要达到这个境界，管理者必须具备相当程度的 Shell 操控能力与 Shell 程序设计能力，笔者拥有丰富的主机管理实务经验，以及多年的教学经验，能引领读者轻松跨越 Bash Shell 程序设计的门槛，摆脱菜鸟劳碌的命运，达到轻松管理，主机不出包的境界。

本书完整涵盖 Bash 3.x 的各项功能，并介绍最新 Bash 4.0 的重要功能；由简入繁、循序渐进，建立扎实的 Bash Shell 程序设计基础；各章提供许多范例，充分展示 Bash Shell 程序设计的技巧；带领读者学习如何设计自动化程序，轻松解决问题，增进工作效率；还包含许多管理实务的技巧，可快速提升读者主机管理的能力。

本书适合系统管理员、网络管理员、架构师及普通学习者参考学习。

本书为精诚信息股份有限公司—悦知文化授权电子工业出版社于中国大陆（台、港、澳除外）地区之中文简体版本。本著作物之专有出版权为精诚信息股份有限公司—悦知文化所有。该专有出版权受法律保护，任何人不得侵犯。

版权贸易合同登记号 图字：01-2009-7065

图书在版编目（CIP）数据

实战 Linux Shell 编程与服务器管理 / 卧龙小三著；梁昌泰，张琦，黄琨改编.—北京：电子工业出版社，2010.3

ISBN 978-7-121-10328-5

I. 实… II. ①卧…②梁…③张…④黄… III. Linux 操作系统 IV. TP316.89

中国版本图书馆 CIP 数据核字（2010）第 022687 号

责任编辑：田 蕾

印 刷：北京智力达印刷有限公司

装 订：北京中新伟业印刷有限公司

出版发行：电子工业出版社

北京市海淀区万寿路 173 信箱 邮编：100036

开 本：787×980 1/16 印张：34.25 字数：876.8 千字

印 次：2010 年 3 月第 1 次印刷

印 数：4 000 册 定价：59.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888。

质量投诉请发邮件至 zlts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线：（010）88258888。

2002 年,我曾在台南县对全县中、小学校的网管讲授 Shell 程序设计课程。当时关于 Shell 的书籍和参考资料并不多(至今仍然如此),只好自编一份入门讲义,供上课的学员参考。课程结束后,我把它放到网络上,让其他同学也能参考。没想到,这份讲义,由于写得十分清楚、易懂,受到许多人的欢迎。这几年来,陆续接到很多网络上朋友的来信,希望我能进一步把 Shell 程序设计的入门讲义写成更详细的工具书,不过,由于个人工作上的关系,一直没有实现。今年,机缘巧合,各方面的条件和时机恰好配合,所以,这本书就出现在这里啦!

许多人可能不了解,Shell 其实是操作系统的重要组件之一。当我们管理一台主机时,在系统核心和用户之间的接口就是 Shell,我们几乎必须时刻面对它。Shell 能解译使用者输入的命令并执行命令。这些命令本身,通常用来访问文件系统、建立行程、管理行程、操作输出入设备、管理内存,以及操作和网络有关的应用。因此,对 Shell 的操控能力,可以说大约就等于是主机的管理能力,所以,想要对主机管理有更深一层认识的人,不可不熟悉 Shell。

笔者常说:凡是可以交给计算机做的事,不要由人来做(因为,人会累,计算机不会累)。主机管理的上乘之道,就在于能够充分实践这句话。换言之,就是把工作予以自动化。那么,主机管理者如何把工作自动化呢?关键就在于他是否具备 Shell 程序设计的能力。我们可以说:对 Shell 的操控能力和 Shell 程序设计的能力,是每一位主机管理者应具备的基本知能。唯有如此,这样的管理人员才具备强大的生产力(有生产力的管理者,才能偷懒,有时间做自己喜欢做的事,比如:思考如何创新)。

这几年来,由于自由软件运动蓬勃发展,一般人接触 Linux/BSD 等系统的机会多了,较几年前来说,真是不可同日而语。但许多人在登录主机之后,却往往不知道要做什么,或如何运用它来解决问题,实在是很难得。本书希望能带领读者了解 Shell 的重要性,熟悉 Shell 的运用方法,建立自动化管理主机的能力,迅速、轻松地解决问题。

不过,像 Linux/BSD 这种 UNIX-like 系统,自由度是非常高的,因此,Shell 的种类也很多。那么,使用者应该挑选哪一种 Shell 呢?笔者建议:依该类 Shell 的占有率来决定。

现今,以 Linux 为核心的自由软件操作系统的发展最受瞩目,影响力也最为惊人,不但深入各种工作阶层,而且,在一般使用者的普及面上,也大有应用。运用 Linux 来担任各种网络主机的情况,可以说比比皆是,甚至,以 Linux/X Window 为计算机桌面的人,比例也越来越高。各种 Linux 套件版本,推陈出新、百家争鸣,热门流行的套件不下百种。但不管是哪一种 Linux 套件,大多数套件默认采用的 Shell 程序,都是 Bash Shell。因此,本书要介绍的是 Bash Shell 程序设计。

当然,运用之妙,存乎一心。当您熟悉 Bash Shell 之后,要再熟悉其他的 Shell,其原理、架构、方法,是可以类推的。因此,您不用担心,学习 Bash Shell 程序设计是一种孤行独市的冒险投资。相反地,学习自由软件的最大优点之一,就是其学习具有累积性。今天努力下

Chapter 1 Shell 简介

1.1 Shell 是什么	2
1.2 Shell 的历史简介	4
1.3 Bash Shell 的功能	5
1.4 第一个 Bash Shell 程序: "哈啰! Bash Shell! "	6
1.5 第一个有用的 Bash Shell	7

Chapter 2 部署 Bash Shell 的环境

2.1 在 Linux 平台部署 Bash Shell 的环境	10
2.2 升级 B2D 的 Bash 版本	10
2.3 自行编译最新版的 Bash	11
2.4 切换使用新版的 Bash Shell	13
chsh 的用法	13
2.5 在 FreeBSD 平台部署 Bash Shell 的环境	15
2.6 在 OpenBSD 平台部署 Bash Shell 的环境	16
2.7 在 Windows 平台部署 Bash Shell 的环境	18

Chapter 3 基础概念介绍

3.1 登录、注销	22
登录主机	22
注销主机、结束终端程序	23
3.2 文件、目录	23
文件	23
目录	26
3.3 通配符与转义字符	30
通配符	30
转义字符	31
3.4 字符集合	33
3.5 括号扩展	34
3.6 系统默认开启的文件	35
3.7 标准输入/输出转向	35
转向输出	36
转向附加	36
转向输入	37
转向输入和转向输出合用	37

利用转向输出做简易编辑	38
3.8 管道	38
3.9 前台工作、后台工作	40

Chapter 4 Bash Shell 程序的结构

4.1 Shell 程序的组成	42
4.2 设定执行权限	44
4.3 执行 Bash Script 的方法	44
4.4 Bash Script 排错的方法	45
4.5 Bash Script 执行的原理	47
login Shell	47
4.6 Bash Shell 的启动配置文件	50
Bash 不同的运行模式	50

Chapter 5 基本操作介绍

5.1 内置命令	56
5.2 命令行程序	71
5.3 执行多个命令的方法	88
5.4 记录命令的执行过程	90
5.5 命令行编辑的方法	91
5.6 Bash 分析命令行的方式	98

Chapter 6 变量与字符串操作

6.1 变量是什么	102
6.2 变量的命名	103
6.3 设定变量	104
6.4 取得变量值	105
6.5 取消与清空变量	109
6.6 变量和引号	110
6.7 变量的有效范围	112
6.8 环境变量	113
输出变量	113
取消环境变量值	115
6.9 Bash 的内置变量	115
6.10 调整变量的属性	125
只读变量：使用 readonly 命令	125

只读变量: 使用 <code>declare</code> 命令	126
调整变量的属性	127
6.11 别名	128
建立别名	128
替换别名	129
取消别名	130
6.12 自定义环境	130
以一般账号的角色工作时	130
以系统管理员的角色工作时	133
6.13 数组	133
数组的用法	134
取出数组所有的元素	135
取得数组元素的个数	135
取消数组或数组元素	136
6.14 Here Document	136
用 Here Document 设定变量	136
控制 Here Document 的格式	138
利用 Here Document 做多行批注	139
利用 Here Document, 打包 C (或其他程序语言) 的原始码	140

Chapter 7 高级变量

7.1 变量扩展: 测试存在性及空值	144
“变量存在”的定义	144
测试变量“存在与否”的基本用法	145
测试变量“不存在”或其值为空: “传回”一个默认值	146
测试变量“不存在”或其值为空: 给变量设一个默认值	148
测试变量是否“不存在”或其值为空: 提示错误信息	149
测试变量的“存在性”	151
7.2 变量扩展: 取字符串切片、字符串长度	152
取字符串切片	152
计算字符串长度	155
7.3 变量扩展: 对比样式	156
由字符串前面对比, 删除相符者	156
由字符串后面对比, 删除相符者	158
取代或删除部分字符串	159
把对比符合的字符串删除	160

要求样式在句首或句尾	160
7.4 变量扩展: 取变量名称列表、数组索引列表	162
取变量名称列表	162
取数组索引列表	162
7.5 命令替换	163
7.6 算术扩展	165

Chapter 8 算术运算

8.1 简介	168
8.2 算术扩展	169
8.3 使用外部程序 <code>expr</code> 做算术运算	174
8.4 使用 <code>\$[]</code> 做算术运算	177
8.5 使用内置命令 <code>declare</code> 、 <code>let</code> 做算术运算	178
<code>declare</code> 的用法	178
<code>let</code> 的用法	179

Chapter 9 流程控制

9.1 命令的结束状态	182
9.2 <code>if</code> 条件判断	183
简易的 <code>if</code> 语法: <code>if-then-else</code>	183
<code>if</code> 的完整语法	184
9.3 条件测试的写法	187
9.4 条件判断式的真假值	195
关于文件属性的判断式	195
关于字符串的条件判断式	197
关于算式的条件判断式	199
关于 <code>Bash</code> 选项的条件判断式	200
9.5 <code>case</code> 条件判断	200
<code>case</code> 的语法	200
高级样式	202
9.6 <code>for</code> 循环	204
<code>for</code> 的语法	204
<code>for</code> 无穷循环	207
9.7 <code>while</code> 循环	207
<code>while</code> 的语法	207
使用 <code>while</code> 循环, 读取文件内容	209

while 无穷循环	210
9.8 until 循环	211
until 的语法	211
until 无穷循环	212
9.9 select 命令	212
9.10 break 和 continue	214
break 命令	214
continue 命令	216
9.11 综合运用	216

Chapter 10 函数

10.1 函数的用法	220
函数的语法	220
函数的结束状态	222
10.2 函数与变量的作用范围	224
函数的作用范围	224
变量的作用范围	225
10.3 位置参数	226
传递参数	226
命令行的位置参数	229
移动位置参数	229
指定位置参数值	230
取用命令行的选项和参数	233
10.4 建立函数库	235
10.5 递归函数	237
10.6 实例应用	238

Chapter 11 转向

11.1 文件代码	244
11.2 操作文件	244
开启文件	244
关闭文件	245
复制文件代码, 建立文件连接	245
11.3 转向的用法	247
转向输入	247
转向输出	248

转向附加.....	250
标准错误伴随标准输出做转向.....	250
Here Document 转向.....	251

Chapter 12 trap——陷阱触发

12.1 信号 (signal)	254
进程编号 PID	254
信号列表.....	255
传送信号的方法.....	256
常用的信号.....	257
12.2 trap 的运用	260
trap 的语法 1.....	260
trap 的语法 2.....	261
trap 的语法 3.....	261
trap 的语法 4.....	262

Chapter 13 sed 和 awk 入门

13.1 正则表达式.....	264
何谓正则表达式.....	264
13.2 sed 的用法	266
实例应用.....	268
13.3 awk 的用法	271
实例应用.....	272

Chapter 14 进程管理和工作控制

14.1 进程管理.....	276
ps 命令.....	276
top 命令.....	281
htop 命令.....	283
传送信号.....	284
调整进程的优先级.....	284
nice 命令.....	285
renice 命令.....	286
PRI 和 NI.....	287
14.2 工作控制.....	287
以实例说明工作控制.....	287

常用的工作控制命令	290
14.3 进程替换.....	292
Chapter 15 历史指令	
15.1 历史指令的功能	296
history 命令	297
fc 命令	298
15.2 历史指令扩展	298
Chapter 16 使用 Shell Script 撰写文字和图形接口程序	
16.1 使用 dialog 撰写文本接口程序.....	306
是或否对话框 (yesno)	309
多选一对话框 (radiolist)	310
选单对话框 (menu)	312
单行输入对话框 (inputbox)	314
输入密码对话框 (passwordbox)	315
显示进度对话框 (gauge)	317
核查列表对话框 (checklist)	320
文件对话框 (textbox)	321
信息对话框 (infobox)	321
调整时间对话框 (timebox)	322
倒数计时对话框 (pause)	322
选择文本区对话框 (fselect)	322
16.2 使用 xdialog 撰写图形接口程序.....	323
显示信息 (msgbox)	323
多选一对话框 (radiolist)	324
单行输入对话框 (inputbox)	325
双行输入对话框 (2inputsbox)	326
三行输入对话框 (3inputsbox)	328
下拉列表对话框 (combobox)	329
重设大小范围的对话框 (rangebox)	329
两个滚动条的对话框	330
3 个滚动条的对话框	330
spinbox 对话框	331
编辑对话框 (editbox)	331

Chapter 17 文件操作

17.1	取得目录内容：文件列表	334
	利用 ls 取得目录内容	334
17.2	判断文件是否存在	336
17.3	建立文件	338
	建立一般文件	338
	建立目录	340
	建立区块设备文件	341
	建立字符设备文件	343
	建立 FIFO 文件	343
	建立 Socket 文件	345
	建立符号链接文件	345
17.4	取得文件的属性	346
17.5	对比文件的差异	352
	diff 的其他用法	355
17.6	搜寻文件内容	358
17.7	建立临时文件	361
17.8	复制文件	363
17.9	大、小写文件名互换	366
17.10	转换编码	371
	转换编码的工具	372
	大量转换编码	373

Chapter 18 主机系统管理

18.1	账号管理	376
18.2	DNS 自动产生器	389
18.3	DHCP 自动产生器	402
18.4	检测主机存在	407

Chapter 19 Bash 在 TCP/IP 方面的运用

19.1	Bash 网络转向	416
	wget.sh 程序说明	418
	Echo 服务器	419
19.2	自动登录 FTP	421
19.3	自动登录 SSH	424

Chapter 20 主机安全管理

20.1	收集主机信息	428
	列出主机 IP	428
	找出主机名	428
	列出核心信息	430
	列出内存使用情况	430
	列出磁盘种类	432
	列出磁盘容量	433
	列出磁盘用量	435
	列出主机系统平均负载	437
	列出系统中的隐藏文件 / 有特殊权限的文件	438
	列出占用磁盘空间较多的目录	440
20.2	主机服务信道	441
	基本概念	441
	通信状态	443
	管理主机服务信道	444
20.3	限制连接来源	446
	使用 TCP Wrapper	446
	自动建立 hosts.allow 和 hosts.deny	449
20.4	限制连接来源：使用 iptables	452
	关于 Netfilter	452
	关于 iptables	452
	iptables 的语法	453
	iptables 的用例	454
	自动建立 iptables 配置文件	455
20.5	文件完整性审核	465
	关于 checksum	465
	使用 md5sum	466
	使用 shasum	466
	使用 Shell 自制文件完整性的审核工具	467
20.6	邮件通知	470
	邮件方法	470
	监测分区磁盘空间使用量	471
	安排邮件内容的格式	475
	使用 Bash Shell 邮件和附件	478

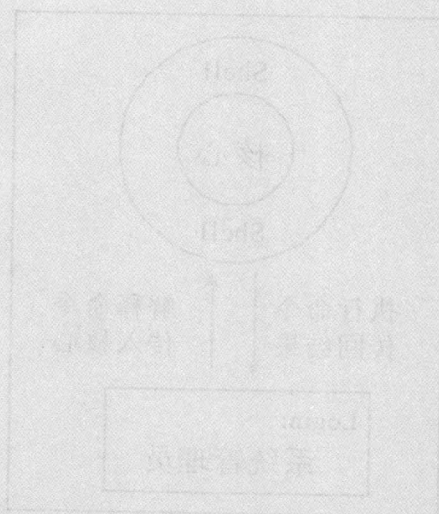
二进制文件编码 / 译码的方法	479
邮寄附件	480

Chapter 21 备份

21.1 使用 tar 做备份	488
基本用法	489
使用 tar 做完整备份	492
完整备份和差异性备份	493
使用 tar 进行差异性备份	494
还原备份文件的方法	494
21.2 使用磁带备份	495
操作磁带的方法	495
建立常用函数	501
21.3 使用 cpio 做备份	502
关于 cpio	502
基本操作法	503
使用 cpio 备份到磁带	504
使用 cpio 备份到远程的主机	504
21.4 使用 dump/restore 做备份 / 恢复	505
关于 dump 和 restore	505
/etc/fstab 和 dump 的关系	505
dump 和文件属性	505
dump 的基本操作	506
备份至磁带	509
21.5 使用 cp、dd 做备份	509
使用 cp	509
使用 dd	510
21.6 使用光盘做备份	513
21.7 使用 rsync 做备份	514
关于 rsync	514
设定 rsync server	515
设定 rsync client	517
安全性	519
rsync 设定产生器	520

附录 Bash 4 新功能

Shell 简介



当我们登录主机之后，第一个接触到的便是 Shell。在开始学习 Shell 之前，我们必须先对它有点认识才行。

1.1 Shell 是什么

Linux/BSD 系统分成 3 个重要部分：

- 核心（kernel）。
- Shell。
- 工具程序。

核心部分相当抽象，操作者不易和它直接沟通，因此，必须要有一个良好的接口（interface），以使操作时更方便，这个接口便是 Shell。换言之，Shell 就是一个连接系统核心和操作者之间的沟通接口。

那么，为何称它为 Shell 呢？Shell 的本意其实就是“壳”的意思。

没错，在核心的外面，包覆着一层外壳，用来负责接收使用者输入的指令，然后将指令解释成核心能够了解的方式，传给核心去执行，核心执行后，再将结果传回至默认的输出周边。因此，习惯上称它为 Shell（核心的外壳）。

其运行方式，如图 1-1 所示。

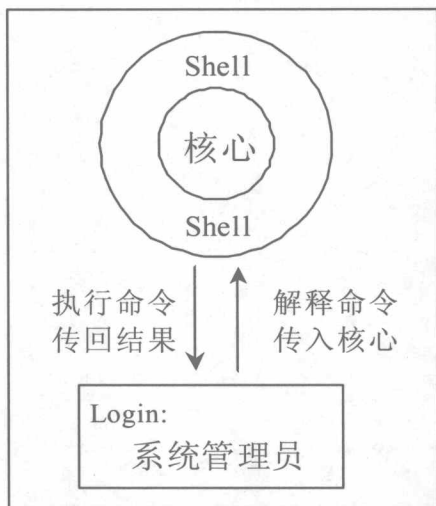


图 1-1 Shell 的运行方式

例如：键入

```
ls /
```

Shell 给出以下回应：

```
bin  cdrom  etc      home   lib      miniroot  none  proc  sbin  tmp  var
boot dev    floppy  initrd  lost+found  mnt    opt   root  sys   usr   vmlinuz
```

这个指令列出根目录的内容。

其实，不只是 Linux/BSD 有这一层 Shell，其他操作系统也有。如 DOS 的 COMMAND.COM、Windows 和 Mac OS 的 GUI (Graphical User Interface) 图形界面接口都是 Shell 的一种。

Shell 根据表现方式与读取使用者输入种类的不同，可分为两大类：

- Text base: 文字导向。
- Graphic base: 图形导向。

所谓“读取使用者输入种类的不同”是指：读取自键盘、鼠标、屏幕触控或其他串行输入等。

这样说来，Shell 好像只是命令解释器罢了？！嗯，这倒要根据不同的 OS (操作系统) 所赋予 Shell 的功能与选择性的自由度而定。

以 DOS 的 COMMAND.COM 而言，它就是一个不折不扣的命令行解释器，除了一点点 batch 文件的能力之外，其他功能并不多。Windows 平台的 GUI，则是一个图形式的命令解释器，界面十分友好。不过，这两种 OS，不能让您自由而简单地选择 Shell (虽然以前 DOS 时代，有个 4DOS 可选用，但其他选择性不多)。

Linux/BSD 的 Shell，除了作为命令解释器之外，它也是一个不错的程序语言，是系统维护管理时的重要工具。

由于 UNIX-like 家族的操作系统 (如 Linux/BSD)，对 Shell 的处理，采取独立、自由开放的政策，因此，Shell 的种类相当多，更可以让人自由地更换 (使用 chsh 指令)。

目前流行的 Shell 有：

序号	Shell 名称	执行文件名	原始开发者	有无开放原始码的版本/或 衍生版本	授权协议
1	Bourne Shell (sh)	/bin/sh	Stephen Bourne	有 (4.4BSD-Lite 以后), Charles Forsyth 版本	BSD/Public Domain
2	Bourne Again Shell (Bash)	/bin/Bash	Brian Fox	有	GPL