

认识linux之LinuxShells简介Linux认证考试 PDF转换可能丢失  
图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/555/2021\\_2022\\_\\_E8\\_AE\\_A4\\_E8\\_AF\\_86linu\\_c103\\_555909.htm](https://www.100test.com/kao_ti2020/555/2021_2022__E8_AE_A4_E8_AF_86linu_c103_555909.htm) 什么是 Shell? Shell是一种具备特殊功能的程序，它是介于使用者和 UNIX/Linux 操作系统之核心程序(kernel)间的一个接口。为什么我们说 shell 是一种介于系统核心程序与使用者间的中介者呢？读过操作系统概论的读者们都知道操作系统是一个系统资源的管理者与分配者，当您有需求时，您得向系统提出；从操作系统的角度来看，它也必须防止使用者因为错误的操作而造成系统的伤害？众所周知，对计算机下命令得透过命令(command)或是程序(program)；程序有编译器(compiler)将程序转为二进制代码，可是命令呢？其实shell也是一支程序，它由输入设备读取命令，再将其转为计算机可以了解的机械码，然后执行它。各种操作系统都有它自己的 shell，以 DOS 为例，它的 shell 就是 command.com。如同 DOS 下有 NDOS，4DOS，DRDOS 等不同的命令解译程序可以取代标准的 command.com，UNIX 下除了 Bourne shell(/bin/sh) 外还有 C shell(/bin/csh)、Korn shell(/bin/ksh)、Bourne again shell(/bin/bash)、Tenex C shell(tcsh) ... 等其它的 shell。UNIX/Linux 将 shell 独立于核心程序之外，使得它就如同一般的应用程序，可以在不影响操作系统本身的情况下进行修改、更新版本或是添加新的功能。Shell 的激活在系统起动的時候，核心程序会被加载内存，负责管理系统的工作，直到系统关闭为止。它建立并控制着处理程序，管理内存、档案系统、通讯等等。而其它的程序，包括 shell 程序，都存放在磁盘中。核心程序将它们加载内

存，执行它们，并且在它们中止后清理系统。Shell 是一个公用程序，它在您签入时起动。藉由解译使用者输入的命令(由命令列或命令档)，Shell 提供使用者和核心程序产生交谈的功能。当您签入(login)时，一个交谈式的shell 会跟着起动，并提示您输入命令。在您键入一个命令后，接着就是 shell 的工作了，它会进行：1. 语法分析命令列 2. 处理万用字符(wildcards)、转向(redirection)、管线(pipes)与工作控制(job control) 3. 搜寻并执行命令 当您刚开始学UNIX/Linux系统时，您大部份的时间会花在于提示符号(prompt)下执行命令。如果您经常会输入一组相同形式的命令，您可能会想要自动执行那些工作。如此，您可以将一些命令放入一个档案(称为命令档，script)，然后执行该档。一个shell 命令档很像是 DOS 下的批次(如 Autoexec.bat)：它把一连串的 UNIX 命令存入一个档案，然后执行该档。较成熟的命令档还支持若干现代程序语言的控制结构，譬如说能做条件判断、循环、档案测试、传送参数等。要写着写命令档，不仅要学习程序设计的结构和技巧，而且对 UNIX/Linux 公用程序及如何运作需有深入的了解。有些公用程序的功能非常强大(例如 grep、sed 和awk)，它们常被用于命令档来操控命令输出和档案。在您对那些工具和程序设计结构变得熟悉之后，您就可以开始写命令档。当由命令档执行命令时，此刻，您就已经把 shell 当做程序语言使用了。细说 Shell 的生平 第一个有重要意义的，标准的 UNIX shell 是V7(ATamp.T 的David Korn 在 80 年代中期发明了 Korn shell，在 1986 年发行且在 1988 年成为正式的部分 SVR4 UNIX。Korn shell 实际上是 Bourne shell 的超集，且不只可在 UNIX 系统上执行，同时也可在 OS/2、VMS、和

DOS上执行。它提供了和 Bourne shell 向上兼容的能力，且增加了许多在 C shell 上受欢迎的特色，更增加了速度和效率。Korn shell 已历经许多修正版，要找寻您使用的是那一个版本可在 ksh 提示符号下按 Ctrl-v 键。三种主要的 Shell 与其分身在大部份的UNIX系统，三种著名且广被支持的shell 是 Bourne shell(AT&T发展的，以简洁、快速著名。 Bourne shell 提示符号的默认值是 \$。 C shell 是柏克莱大学(Berkeley)所开发的，且加入了一些新特性，如命令列历程(history)、别名(alias)、内建算术、档名完成(filename completion)、和工作控制(job control)。对于常在交谈模式下执行 shell 的使用者而言，他们较喜爱使用 C shell；但对于系统管理者而言，则较偏好以 Bourne shell 来做命令档，因为 Bourne shell 命令比 C shell 命令档来的简单及快速。 C shell 提示符号的默认值是 %。 Korn shell 是 Bourne shell 的超集(superset)，由 AT&T 的 David Korn 所开发。它增加了一些特色，比 C shell 更为先进。 Korn shell 的特色包括了可编辑的历程、别名、函式、正规表达式万用字符(regular expression wildcard)、内建算术、工作控制(job control)、共作处理(coprocessing)、和特殊的除错功能。 Bourne shell 几乎和 Korn shell 完全向上兼容(upward compatible)，所以在 Bourne shell 下开发的程序仍能在 Korn shell 上执行。 Korn shell 提示符号的默认值也是 \$。在 Linux 系统使用的 Korn shell 叫做 pdksh，它是指 Public Domain Korn Shell。除了执行效率稍差外，Korn shell 在许多方面都比 Bourne shell 为佳；但是，若将 Korn shell 与 C shell 相比就很困难，因为二者在许多方面都各有所长，就效率和容易使用上看，Korn shell 是优于 C shell，相信许多使用者对于 C Shell 的

执行效率都有负面的印象。在shell的语法方面，Korn shell是比较接近一般程序语言，而且它具有子程序的功能及提供较多的资料型态。至于 Bourne shell，它所拥有的资料型态是三种 shell 中最少的，仅提供字符串变量和布尔型态。在整体考量下 Korn shell 是三者中表现最佳者，其次为 C shell，最后才是 Bourne shell，但是在实际使用中仍有其它应列入考虑的因素，如速度是最重要的选择时，很可能应该采用 Bourne shell，因它是最基本的 shell，执行的速度最快。tcsh 是近几年崛起的一个免费软件(Linux 下的C shell 其实就是使用 tcsh)执行，它虽然不是UNIX的标准配备，但是从许多地方您都可以下载到它。如果您是 C shell 的拥护者，笔者建议不妨试试 tcsh，因为您至少可以将它当作是 C shell 来使用。如果您愿意花点时间学习，您还可以享受许多它新增的优越功能，例如：

1. tcsh 提供了一个命令列(command line)编辑程序。
2. 提供了命令列补全功能。
3. 提供了拼字更正功能。它能够自动检测并且更正在命令列拼错的命令或是单字。
4. 危险命令侦测并提醒的功能，避免您一个不小心执行了rm\* 这种杀伤力极大的命令。
5. 提供常用命令的快捷方式(shortcut)。

bash 对 Bourne shell 是向下兼容(backward compatible)，并融入许多 C shell 与 Korn shell 的功能。这些功能其实 C shell(当然也包括了tcsh)都有，只是过去 Bourne shell 都未支持。以下笔者将介绍 bash 六点重要的改进(详细的使用说明笔者会在以后的章节介绍)：

1. 工作控制(job control)。bash 支持了关于工作的讯号与指令，本章稍后会提及。
2. 别名功能(alias)。alias 命令是用来为一个命令建立另一个名称，它的运作就像一个宏，展开成为它所代表的命令。别名并不会替代掉命令的名称，它

只是赋予那个命令另一个名字。

### 3. 命令历程(command history)。

BASH shell 加入了C shell 所提供的命令历程功能，它以 history 工具程序记录了最近您执行过的命令。命令是由 1 开始编号，默认值为500。history 工具程序是一种短期记忆，记录您最近所执行的命令。要看看这些命令，您可以在命令列键入 history，如此将会显示最近执行过之命令的清单，并在前方加上编号。这些命令在技术上每个都称为一个事件。事件描述的是一个已经采取的行动(已经被执行的命令)。事件是依照执行的顺序而编号，越近的事件其编号码越大，这些事件都是以它的编号或命令的开头字符来辨认的。history 工具程序让您参照一个先前发生过的事件，将它放在命令列上并允许您执行它。最简单的方法是用上下键一次放一个历程事件在您的命令列上；您并不需要先使用 history 显示清单。按一次向上键会将最后一个历程事件放在您的命令列上，再按一次会放入下一个历程事件。按向下键则会将前一个事件放在命令列上。

### 4. 命令列编辑程序。

BASH shell 命令列编辑能力是内建的，让您轻松地在执行之前修改您输入的命令。若是您在输入命令时拼错了字，您不需重新输入整个命令，只需在执行命令之前使用编辑功能纠正错误即可。这尤其适合于使用冗长的路径名称当作参数的命令时。命令列编辑作业是 Emacs 编辑命令的一部份。您可以用 Ctrl-F 或向右键往前移一个字符，Ctrl-b 或向左键往回移一个字符。Ctrl-d 或DEL 键会删除光标目前所在处的字符。要增加文字的话，您只需要将光标移到您要插入文字的地方并键入新字符即可。无论何时，您都可以按 ENTER 键执行命令。

### 5. 允许使用者自订按键。

### 6. 提供更丰富的变量型态、命令与控制结构至

shell 中。bash 与 tcsh 一样可以从许多网站上免费下载，它们的性质也十分类似，都是整合其前一代的产品然后增添新的功能，这些新增的功能主要都着重在强化 shell 的程序设计能力以及让使用者能够自行定义自己偏好的作业环境。除了上述的五种 shell 之外，zsh 也是一个广为 UNIX 程序设计人员与进阶使用者所采用的 shell，zsh 基本上也是 Bourne shell 功能的扩充。Shell 的使用不论是哪一种 Shell，它最主要的功用都是解译使用者在命令列提示符号下输入的指令。Shell 语法分析命令列，把它分解成以空白区分开的符号(token)，在此空白包括了跳位键(tab)、空白和换行(New Line)。如果这些字包含了 metacharacter，shell 将会评估(evaluate)它们的正确用法。另外，shell 还管理档案输入输出及幕后处理(background processing)。在处理命令列之后，shell 会寻找命令并开始执行它们。Shell 的另一个重要功用是提供个人化的使用者环境，这通常在 shell 的初始化档案中完成(.profile、.login、.cshrc、.tcshrc 等等)。这些档案包括了设定终端机键盘和定义窗口的特征；设定变量，定义搜寻路径、权限限、提示符号和终端机类形；以及设定特殊应用程序所需要的变量，例如窗口、文字处理程序、及程序语言的链接库。Korn shell 和 C shell 加强了个别化的能力:增加历程、别名、和内建变量集以避免使用者误杀档案、不慎签出、并在当工作完成时通知使用者。Shell 也能当解译性的程序语言(interpreted programming language)。Shell 程序，通常叫做命令文件，它由列在档案内的命令所构成。此程序在编辑器中编辑(虽然也可以直接在命令列下写作程序, online scripting)，由 UNIX 命令和基本的程序结构，例如变量的指定、测试条件、和循环所构成。您不

需要编译 shell 命令。Shell 本身会解译命令档中的每一行，就如同由键盘输入一样。shell 负责解译命令，而使用者则必须了解这些命令能做什么。这本书的索引列出了一些有用的命令和它们的使用方法。Shell 的功用 为了确保任何提示符号下输入的命令都能够适当地执行。shell 担任的工作包括有：1. 读取输入和语法分析命令列 2. 对特殊字符求值 3. 设立管线、转向、和幕后处理 4. 处理讯号 5. 设立程序来执行

100Test 下载频道开通，各类考试题目直接下载。详细请访问  
[www.100test.com](http://www.100test.com)