



## shell基本命令

主讲：小美老师

创客引领未来

扫微信二维码 获取更多信息



图示：人 shell 操作系统，硬件  
通配，管道...  
一个指定 排除，任意

tab ↑

lemon

昵称：小姜老师

华清创客学院，嵌入式讲师

makerU

(大概操作 具体操作)



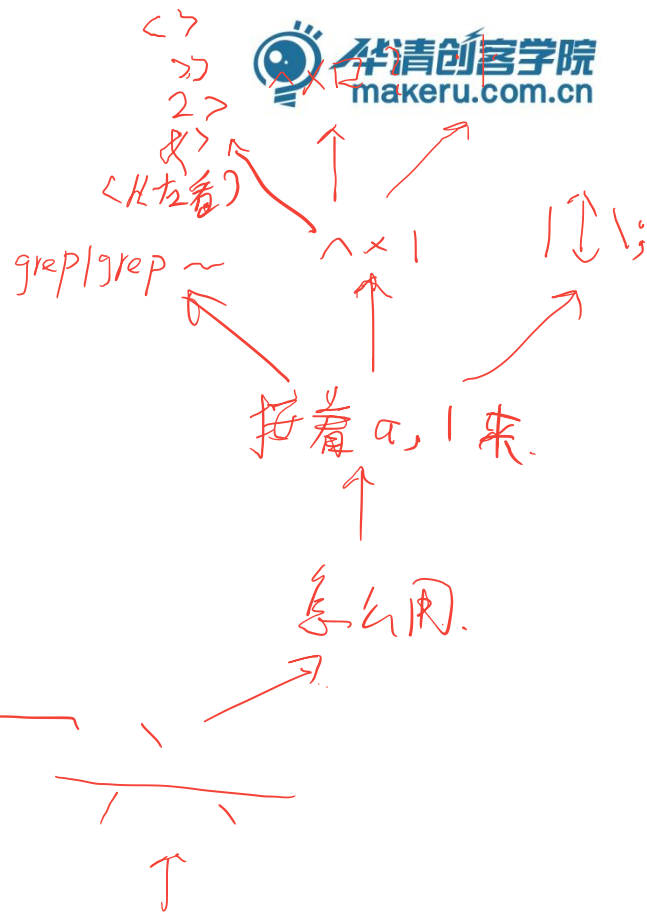
命令选项参数

进程的机目录等级 见下

- shell简介
- shell命令格式
- shell命令行操作

格式及其命令

命令



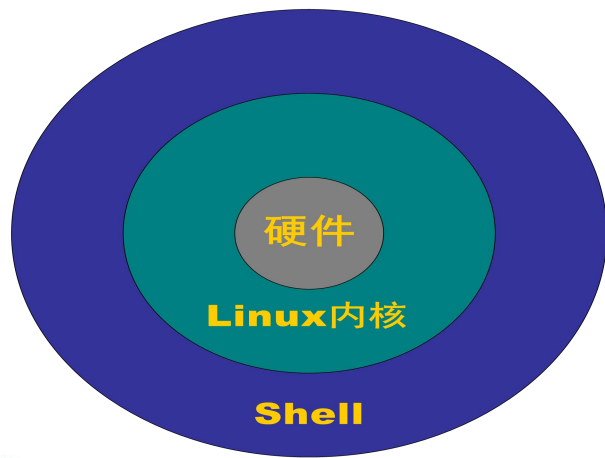
- 随着各式Linux系统的图形化程度的不断提高，用户在桌面环境下，通过点击、拖拽等操作就可以完成大部分的工作。
- 然而，许多Ubuntu Linux功能使用shell命令来实现，要比使用图形界面交互，完成的**更快、更直接**。

见下，见都想起（关机、补齐、历史等）



# shell简介

- 英文单词shell可直译为“贝壳”。“贝壳”是动物作为外在保护的一种工具。
- 可以这样认为，Linux中的shell就是Linux内核的一个外层保护工具，并负责完成用户与内核之间的交互

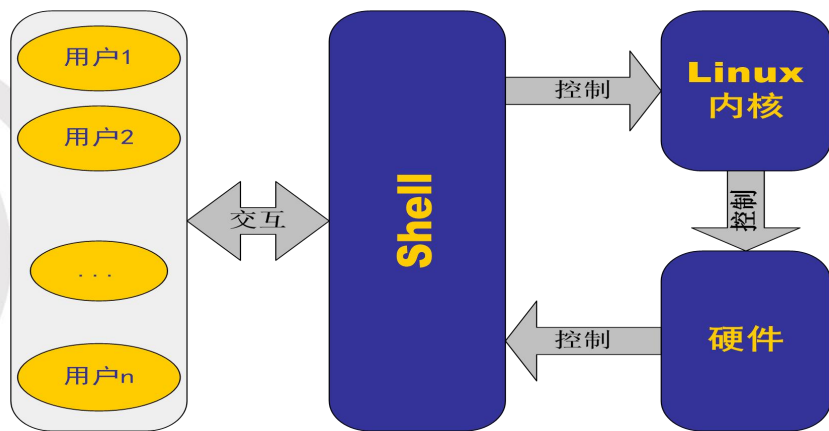


- 命令是用户向系统内核发出控制请求，与之交互的文本流。
- shell是一个命令行解释器，将用户命令解析为操作系统所能理解的指令，实现用户与操作系统的交互。
- 当需要重复执行若干命令，可以将这些命令集合起来，加入一定的控制语句，编辑成为shell脚本文件，交给shell批量执行。



## shell简介

- 用户在命令行提示符下键入命令文本，开始与Shell进行交互
- 接着，shell将用户的命令或按键转化成内核所能够理解的指令 (封装)
- 控制操作系统做出响应，直到控制相关硬件设备
- 然后，shell将输出结果通过shell提交给用户



- 最初的UNIX shell经过多年的发展，由不同的机构、针对不同的目的，开发出许多不同类型的shell程序。目前流行的shell主要有几种：
  - **Bourne Shell（简称sh）**：它是Unix的第一个shell程序，早已成为工业标准。目前几乎所有的Linux系统都支持它。不过Bourne Shell的作业控制功能薄弱，且不支持别名与历史记录等功能。
  - C Shell（简称csh）
  - Korn Shell（简称ksh）
  - ➡ **Bourne Again Shell**：能够提供环境变量以配置用户Shell环境，支持历史记录，内置算术功能，支持通配符表达式，将常用命令内置简化。



# 体验shell命令的乐趣

- 立即关机与重启

- 使用系统中的图形界面
- 使用shell命令

在命令行键入以下命令，系统立即~~关机~~：

```
linux@ubuntu:~$ sudo shutdown -h now
```

在命令行键入以下命令，系统立即~~重新启动~~：

```
linux@ubuntu:~$ sudo shutdown -r now
```

```
linux@ubuntu:~$ sudo reboot now
```



- 定时关机与重启
  - 一位老师在对学生进行上机测试，时间限定在45分钟内，否则计算机将自动关机。这位老师可能使用了下面的命令：

```
linux@ubuntu:~$ sudo shutdown -h +45 "That is all, game over."
```

- 系统正在安装软件，安装时间大概需要持续40分钟左右，软件安装之后需要重新启动电脑。用户不必一直等候在机器旁。使用下面的命令，计算机会在60分钟后自动重启。

```
linux@ubuntu:~$ sudo shutdown -r +60
```

## shell命令格式

- shell提示符标识了命令行的开始。用户在提示符后面输入一条命令并按Enter键，完成向系统提交指令。
  - 通常shell命令提示符采用以下的格式：

```
username@hostname:direction$
```

用户名 主机名 目录名

- username: 用户名，显示当前登录用户的账户名；
- hostname: 主机名，显示登录的主机名，例如若远程登录后，则显示登录的主机名；
- direction: 目录名，显示当前所处的路径，当在根目录下显示为“/”，当在用户主目录下显示为“~”；

- 通常一条命令包含三个要素：命令名称、选项、参数。命令名称是必须的，选项和参数都可能是可选项。命令格式如下所示：

```
$ Command [-Options] Argument1 Argument2 ...
```

指令          选项          参数1          参数2...

- \$: Shell提示符，如果当前用户为超级用户，提示符为“#”，其他用户的提示符均为“\$”；
- Command: 命令名称，Shell命令或程序，严格区分大小写
- Options: 命令选项，用于改变命令执行动作的类型，由“-”引导，可以同时带多个选项；
- Argument: 命令参数，指出命令作用的对象或目标，有的命令允许带多个参数。

- 一条命令的三要素之间用空格隔开;
- 若将多个命令在一行书写, 用分号 (;) 将各命令隔开;
- 如果一条命令不能在一行写完, 在行尾使用反斜杠 (\) 标明该条命令未结束。

- 不带选项和参数

```
wdl@UbuntuFisher:~/Examples$ ls
Experience ubuntu.ogg  logo-Kubuntu.png    oo-derivatives.doc    oo-trig.xls
gimp-ubuntu-splash.xcf oo-about-these-files.odt oo-payment-schedule.ods ubuntu
Sax.ogg
```

- ls命令未带任何参数，列出当前目录中的所有文件，只显示文件名称。
- 命令不带选项或参数，通常意为使用默认选项或参数。

# 命令行操作

- bash除了在命令编辑功能上比sh有很大改进外，还增加了特有功能极大地方便了用户在Shell命令行上的操作。
  - 补齐命令与文件名
  - 查询命令历史
  - 等等

(看到想起)

- 补齐命令与文件名

- 在使用Shell命令时，很多用户会经常遇到命令或文件名没有记全的情况。Bash Shell的命令和文件名补齐功能会帮助用户。在输入命令或文件名的前几个字符后，按TAB键或ESC键自动补齐剩余没有输入的字符串。
- 如果存在多个命令或文件有相同前缀，shell将列出所有相同前缀的命令或文件。shell给出的提示信息，帮助用户回忆和完成输入。之后等待用户输入足够的字符。
- 需要说明的是，连续按两下TAB键或ESC键，用于命令补齐；按下一次TAB键，用于文件名补齐。

```
linux@ubuntu:~/Examples$ net #连续按两下TAB键
net netkit-ftp network-admin netbug netcsid netcat netstat
linux@ubuntu:~/Examples$ net
```



- 查询命令历史

- 用户在Shell下的操作是有很大的连续性的，曾经输入的命令可能需要多次使用。当用户在操作中发现有问题，需要查看曾经执行过的操作。Bash将用户曾经键入的命令序列保存在一个命令历史表中。按“↑”键和“↓”键，便可查询命令历史。
- Bash Shell还提供了history命令。该命令将命令历史表按列表形式，从记录号1开始，一次性全部显示出来。

```
history [numberline]
```

- 显然history只能记录有限条的历史命令，默认保留500条命令。
- Bash Shell将历史命令容量保存在环境变量HISTSIZE中。
  - 使用“echo \$HISTSIZE”查看当前历史命令容量；
  - 通过直接赋值的方法，修改这个环境变量。

```
linux@ubuntu:~$ echo $HISTSIZE #显示历史命令容量
500
linux@ubuntu:~$ HISTSIZE=1000 #修改历史命令容量
linux@ubuntu:~$ echo $HISTSIZE
1000
```

- 本节课先对shell做了基本介绍，然后讲解了linux命令的基本格式，最后介绍了常见的命令行操作
- 思考：
  - 什么是shell?
  - 如何更改历史记录容量?
  - 历史记录如何删除?





扫微信二维码 获取更多信息