

Linux常见指令

1. 认识命令行和界面操作，理解两种操作的历史原因
2. 练习Linux入门的13个基本操作命令
3. 初步了解Linux 目录结构含义
4. 会登录，会操作，会退出

指令 vs 界面 - 故事时间

指令，可能有的小伙伴是第一次听，但是它确实历史悠久，而且先于界面

交互的需求

计算机被创造出来，是为了为人提供服务的。但是为了让计算机更好的为人服务，前提是人必须的和计算机交互起来。

- 类比一下另一种机器，那就是汽车，汽车是为了给人提供交通服务的，但是造出来之后，需要人通过某种方式去控制汽车，当然在汽车里，人和汽车的交互方式是通过方向盘，油门去交互的

满足需求的第一阶段-指令



知乎 @Kissen

打字机的产生是在19-20世纪，反正是远远早于计算机发明的。所以我们要和新发明的计算机进行交互，人们先想到的是借鉴一下打字机的输入方式，来进行常规交互，慢慢演化下，就有了现在的键盘。既然早期交互硬件只有键盘，那么初期的交互方式，也变很顺理成章的是指令方式喽（又没有鼠标可以点）。

- 思考一下，为什么键盘现在的按键布局，是现在这个样子的呢？

1946年2月14，世界第一台计算机ENIAC问世，后因交互的需求，借鉴了打字机键盘，就有了初期的指令交互方式

满足需求的第二阶段-界面

虽然我们还没有正式操作过，但是直觉告诉我们，指令这种方式，可能比较恶心。

一个问题想问问大家，PC时代，要点击界面，需要有鼠标点击(移动时代长大的土著，就不要和老夫杠了~~)，那么，是先有鼠标，还是先有真正意义上的图形化界面呢？答案是，先有鼠标，先看看世界上第一个鼠标的样子吧。



对的，你没有看错，第一个鼠标长这个样子。。。

谁发明的呢？



这个大爷，出生于1925年1月30日，美国人道格拉斯·恩格尔巴特。这个样子的鼠标在1968年面世。

有了硬件设备的支持，同时指令用起来给一般用户的体验特别不好，再这样的需求促进下，乔布斯在1979年参观完施乐公司Xerox Alto（阿尔托）系统之后，迅速模仿并商业化出来了图形化界面交互式的方式，并诞生了世界上第一台商业图形化交互方式的电脑：lisa（当然，后面因为成本问题，效果并不理想，后来乔布斯又继续改良，就有了划时代产品Macintosh（麦金塔），也就是第一代mac笔记本，乔布斯就此封神）

与此同时，比尔盖茨在借鉴苹果lisa下，带着他的windows 1.0，后续在商业上迅速跟进。

乔布斯对苹果的定位是一家计算机科技公司，属于软硬件都做的那种，有自己的软件操作系统，也有自己的硬件设计方案，而且乔布斯是一个神一样的产品经理，所以对软硬件的整合能力，对用户的理解，都造就了一款符合事宜，又设计精良的电脑问世。mac 1问世之后，迅速走红。

可是，走向世界的是苹果，覆盖世界的却是微软。

比尔盖茨看到这样的苹果公司，走的早，产品还好。如何赢？差异化

比尔盖茨对微软的定义是一家软件公司，也就是只负责软件，可是计算机毕竟是一个软硬件结合的产物，于是微软便找了自己的硬件合作方intel(开始是IBM，大家理念有差异，散了)开始搞事。合作的最大好处是成本好控制，生产效率高，毕竟是产业分工的产物，慢慢的windows便席卷全球，就有了摆在我们面前的windows操作系统~

- 苹果的操作系统，一般都在苹果的计算机上跑
- 微软的windows操作系统，什么品牌的电脑都能跑，像联想，dell，华为，鸿基，东芝等。细心的同学应该能发现，所有的这些品牌的计算机上面搭载的CPU都是Intel的X86...

我们把时间先罗列一下

- 打字机产生1808年7月24日(衍生键盘)
- 1946年2月14，世界第一台计算机ENIAC问世

- 因为有交互的需求，计算机交互便借鉴打字机的键盘，慢慢演化，就有了键盘，所以当时人们对交互的想象只能是命令行
- 鼠标在1968年面世
- 1973年施乐公司Xerox Alto（阿尔托）系统，世界上第一个非商业图形化操作系统
- 1979年乔布斯和施乐进行利益交换，带团队参观施乐计算机研究中心，借鉴并商业化带UI的操作系统，后来衍生出来mac图形化界面（期间也有windos的故事）
- 随后，微软推出了自己的桌面级操作系统windows 1.0
- 两家商业公司，两个老总，开始撕
- 就有了现在的PC界面交互方式

最后，有一个问题，就是为什么电子计算机，操作系统，各种电子设备会在1950~2000年这50年大量爆发。

这个问题有点远，有兴趣的老铁，可以在了解一下，当时的时代背景，你可能会有一些不一样的收获

指令 和 界面交互

了解过历史之后，我们能理解，为何会有指令交互和界面交互这样的现象，但是还有两个问题的在讨论一下

- 这两种方式，在操作系统层面有啥区别？
- 既然界面方式是发展的产物，为什么还要在学命令行？

区别

区别这个问题，同学们现在比较难理解，目前也不打算谈。但是在功能定位上，是一样的，都是让用户进行和系统进行交互。也就是说，界面的点击操作，一般也有对应的命令与之对应？是的

为什么要学命令行

- windows/苹果图形界面，是商业化的产物，也就是使用必须简单小白，才能有人用，so what？
- 严格意义上讲，我们必须学一下Linux命令行。因为企业后端有大量的服务器，服务器上跑的全是Linux，Linux上部署的全是服务（比如：qq的服务，王者荣耀，LOL的服务等），无论是成本角度，还是资源效率角度，后台的服务器，都不会大量给你也安装一个界面让你来控制（界面也是要占系统资源的哦）但是服务器上面的服务需要维护，需要管理，也就衍生大量的对基本操作的需求。也就是企业的用人需求
- Linux命令行是职业化的表现，想职业化/成大神，必须掌握Linux
- Linux命令行，更接近系统原生，用的多了，面对很多系统现象，会更好理解

严格意义上讲，光光学习Linux操作是远远不够的。

后续先把平台熟悉起来，后面还有系统，网络，这个后面有机会再说

Linux下基本指令-掌握

ls 指令

语法：ls [选项] [目录或文件]

功能：对于目录，该命令列出该目录下的所有子目录与文件。对于文件，将列出文件名以及其他信息。

常用选项：

- -a 列出目录下的所有文件，包括以 . 开头的隐含文件。
- -d 将目录象文件一样显示，而不是显示其下的文件。如：ls -d 指定目录
- -i 输出文件的 i 节点的索引信息。如 ls -ai 指定文件
- -k 以 k 字节的形式表示文件的大小。ls -alk 指定文件

- -l 列出文件的详细信息。
- -t 以时间排序。
- -R 列出所有子目录下的文件。(递归)

举例：

```
ls -l
```

pwd命令

语法: pwd

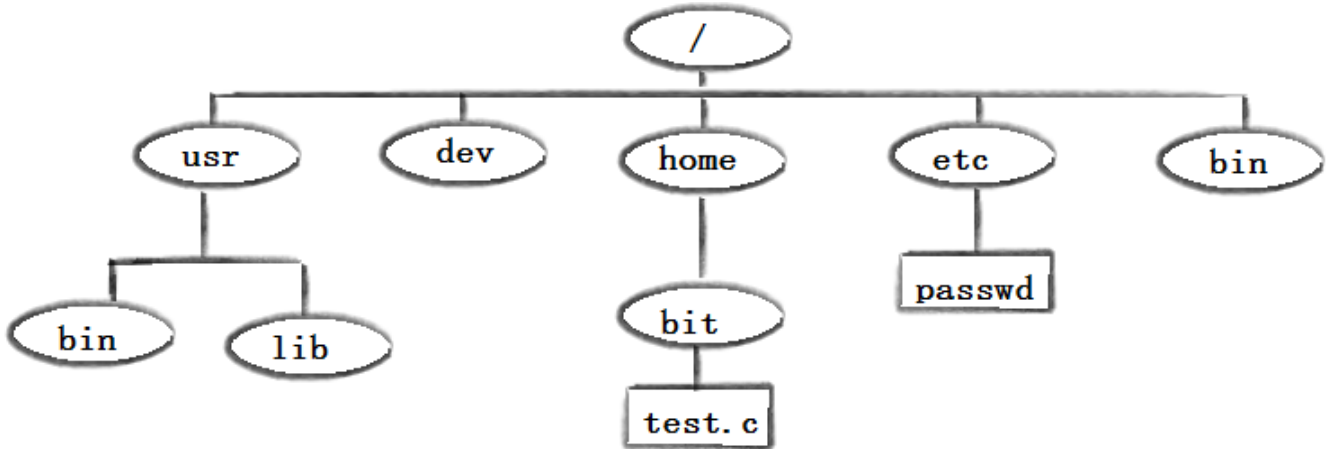
功能：显示用户当前所在的目录

常用选项：

无

cd 指令

Linux系统中，磁盘上的文件和目录被组成一棵目录树，每个节点都是目录或文件。



语法:cd 目录名

功能：改变工作目录。将当前工作目录改变到指定的目录下。

举例：

```

cd .. : 返回上级目录
cd /home/litao/linux/ : 绝对路径
cd ../day02/ : 相对路径
cd ~: 进入用户家目
cd -: 返回最近访问目录
    
```

touch指令

语法:touch [选项]... 文件...

功能：touch命令参数可更改文档或目录的日期时间，包括存取时间和更改时间，或者新建一个不存在的文件。

常用选项：

- -a 或--time=atime或--time=access或--time=use只更改存取时间。
- -d 使用指定的日期时间，而非现在的时间。
- -f 此参数将忽略不予处理，仅负责解决BSD版本touch指令的兼容性问题。

- -m 或--time=mtime或--time=modify 只更改变动时间。
- -r 把指定文档或目录的日期时间，统统设成和参考文档或目录的日期时间相同。
- -t 使用指定的日期时间，而非现在的时间。

mkdir指令（重要）：

语法：mkdir [选项] dirname...

功能：在当前目录下创建一个名为“dirname”的目录

常用选项：

- -p, --parents 可以是一个路径名称。此时若路径中的某些目录尚不存在,加上此选项后,系统将自动建立好那些尚不存在的目录,即一次可以建立多个目录;

举例：

```
mkdir -p test/test1 : 递归建立多个目录
```

[参考文档](#)

rmdir指令 && rm 指令（重要）：

rmdir是一个与mkdir相对应的命令。mkdir是建立目录，而rmdir是删除命令。

语法：rmdir [-p] [dirName]

适用对象：具有当前目录操作权限的所有使用者

功能：删除空目录

常用选项：

- -p 当子目录被删除后如果父目录也变成空目录的话，就连带父目录一起删除。

rm命令可以同时删除文件或目录

语法：rm [-f-i-r-v] [dirName/dir]

适用对象：所有使用者

功能：删除文件或目录

常用选项：

- -f 即使文件属性为只读(即写保护)，亦直接删除
- -i 删除前逐一询问确认
- -r 删除目录及其下所有文件

man指令（重要）：

Linux的命令有很多参数，我们不可能全记住，我们可以通过查看联机手册获取帮助。访问Linux手册页的命令是man

语法：man [选项] 命令

常用选项

- -k 根据关键字搜索联机帮助
- num 只在第num章节找
- -a 将所有章节的都显示出来，比如 man printf 它缺省从第一章开始搜索，知道就停止，用a选项，当按下q退出，他会继续往后面搜索，直到所有章节都搜索完毕。

解释一下,面手册分为8章

1 是普通的命令

2 是系统调用,如open,write之类的(通过这个，至少可以很方便的查到调用这个函数，需要加什么头文件)

- 3 是库函数,如printf,fread4是特殊文件,也就是/dev下的各种设备文件
- 5 是指文件的格式,比如passwd, 就会说明这个文件中各个字段的含义
- 6 是给游戏留的,由各个游戏自己定义
- 7 是附件还有一些变量,比如向environ这种全局变量在这里就有说明
- 8 是系统管理用的命令,这些命令只能由root使用,如ifconfig

man手册可以汉化，但是因为一些基本文本操作我们没有学，所以暂时不汉化，后面再说

cp指令（重要）：

语法：cp [选项] 源文件或目录 目标文件或目录

功能：复制文件或目录

说明：cp指令用于复制文件或目录，如同时指定两个以上的文件或目录，且最后的目的地是一个已经存在的目录，则它会把前面指定的所有文件或目录复制到此目录中。若同时指定多个文件或目录，而最后的目的地并非一个已存在的目录，则会出现错误信息

常用选项：

- -f 或 --force 强行复制文件或目录，不论目的文件或目录是否已经存在
- -i 或 --interactive 覆盖文件之前先询问用户
- -r 递归处理，将指定目录下的文件与子目录一并处理。若源文件或目录的形态，不属于目录或符号链接，则一律视为普通文件处理
- -R 或 --recursive 递归处理，将指定目录下的文件及子目录一并处理

mv指令（重要）：

mv命令是move的缩写，可以用来移动文件或者将文件改名（move (rename) files），是Linux系统下常用的命令，经常用来备份文件或者目录。

语法：mv [选项] 源文件或目录 目标文件或目录

功能：

1. 视mv命令中第二个参数类型的不同（是目标文件还是目标目录），mv命令将文件重命名或将其移至一个新的目录中。
2. 当第二个参数类型是文件时，mv命令完成文件重命名，此时，源文件只能有一个（也可以是源目录名），它将所给的源文件或目录重命名为给定的目标文件名。
3. 当第二个参数是已存在的目录名称时，源文件或目录参数可以有多个，mv命令将各参数指定的源文件均移至目标目录中。

常用选项：

- -f：force 强制的意思，如果目标文件已经存在，不会询问而直接覆盖
- -i：若目标文件 (destination) 已经存在时，就会询问是否覆盖！

举例-了解一下-转移：

```
将rm改造成mv
vim ~/.bashrc #修改这个文件
mkdir -p ~/.trash
alias rm=trash
alias ur=undelfile
undelfile() {
    mv -i ~/.trash/\$@ ./
}
trash() {
    mv \$@ ~/.trash/
}
```

cat

语法：cat [选项] [文件]

功能：查看目标文件的内容

常用选项：

- -b 对非空输出行编号
- -n 对输出的所有行编号
- -s 不输出多行空行

date

date 指定格式显示时间：date +%Y:%m:%d

date 用法：date [OPTION]... [+FORMAT]

1.在显示方面，使用者可以设定欲显示的格式，格式设定为一个加号后接数个标记，其中常用的标记列表如下

- %H : 小时(00..23)
- %M : 分钟(00..59)
- %S : 秒(00..61)
- %X : 相当于 %H:%M:%S
- %d : 日 (01..31)
- %m : 月份 (01..12)
- %Y : 完整年份 (0000..9999)
- %F : 相当于 %Y-%m-%d

2.在设定时间方面

- date -s //设置当前时间，只有root权限才能设置，其他只能查看。
- date -s 20080523 //设置成20080523，这样会把具体时间设置成空00:00:00
- date -s 01:01:01 //设置具体时间，不会对日期做更改
- date -s "01:01:01 2008-05-23" //这样可以设置全部时间
- date -s "01:01:01 20080523" //这样可以设置全部时间
- date -s "2008-05-23 01:01:01" //这样可以设置全部时间
- date -s "20080523 01:01:01" //这样可以设置全部时间

3.时间戳

时间->时间戳

- date +%s

时间戳->时间

- date +%Y:%m:%d -d @1599642565

Unix时间戳（英文为Unix epoch, Unix time, POSIX time 或 Unix timestamp）是从1970年1月1日（UTC/GMT的午夜）开始所经过的秒数，不考虑闰秒。

top

Linux下常用的性能分析工具，能够实时显示系统中各个进程的资源占用状况，类似于Windows的任务管理器。

我们目前要了解的选项

- -u<用户名> 指定用户名
- -p<进程号> 指定进程
- -n<次数> 循环显示的次数

shutdown-关机

语法：shutdown [选项] 常见选项：

- -h：将系统的服务停掉后，立即关机
- -r：在将系统的服务停掉之后就重新启动
- -t sec：-t 后面加秒数，亦即『过几秒后关机』的意思

如果各位老铁是后面购买了云服务器，尽量不要使用shutdown来关机，因为云服务器一般都要保证服务器一直在线，使用exit或者ctrl+d退出即可

重要的几个热键[Tab],[ctrl]-c, [ctrl]-d

- [Tab]按键---具有『命令补全』和『档案补齐』的功能
- [Ctrl]-c按键---让当前的程序『停掉』
- [Ctrl]-d按键---通常代表着：『键盘输入结束(End Of File, EOF 戒 End OfInput)』的意思；另外，他也可以用来取代exit

以下命令作为扩展-了解:

- ◆ **安装和登录命令**：login、shutdown、halt、reboot、install、mount、umount、chsh、exit、last;
- ◆ **文件处理命令**：file、mkdir、grep、dd、find、mv、ls、diff、cat、ln;
- ◆ **系统管理相关命令**：df、top、free、quota、at、lp、adduser、groupadd、kill、crontab;
- ◆ **网络操作命令**：ifconfig、ip、ping、netstat、telnet、ftp、route、rlogin、rcp、finger、mail、nslookup;
- ◆ **系统安全相关命令**：passwd、su、umask、chgrp、chmod、chown、chattr、sudo ps、who;
- ◆ **其它命令**：tar、unzip、gunzip、unrarj、mtools、man、unendcode、uudecode。