# Linux vi/vim

所有的 Unix Like 系统都会内建 vi 文本编辑器，其他的文本编辑器则不一定会存在。

但是目前我们使用比较多的是 vim 编辑器。

## vim

Vim是从 vi 发展出来的一个文本编辑器。代码补完、编译及错误跳转等方便编程的功能特别丰富，在程序员中被广泛使用。

简单的来说， vi 是老式的字处理器，不过功能已经很齐全了。

## vi/vim 的使用



基本上 vi/vim 共分为三种模式，分别是命令模式（Command mode）、输入模式（Insert mode）和底线命令模式（Last line mode）。 这三种模式的作用分别是：

### **命令模式**

用户刚刚启动 vi/vim，便进入了命令模式。

此状态下敲击键盘动作会被Vim识别为命令，而非输入字符。比如我们此时按下i，并不会输入一个字符，i被当作了一个命令。

以下是常用的几个命令：

* i 切换到输入模式，以输入字符。
* x 删除当前光标所在处的字符。
* : 切换到底线命令模式，以在最底一行输入命令。

若想要编辑文本：启动Vim，进入了命令模式，按下i，切换到输入模式。

命令模式只有一些最基本的命令，因此仍要依靠底线命令模式输入更多命令。

### **输入模式**

在命令模式下按下i就进入了输入模式。

在输入模式中，可以使用以下按键：

* 字符按键以及Shift组合，输入字符
* ENTER，回车键，换行
* BACK SPACE，退格键，删除光标前一个字符
* DEL，删除键，删除光标后一个字符
* 方向键，在文本中移动光标
* HOME/END，移动光标到行首/行尾
* Page Up/Page Down，上/下翻页
* Insert，切换光标为输入/替换模式，光标将变成竖线/下划线
* ESC，退出输入模式，切换到命令模式

### **底线命令模式**

在命令模式下按下:（英文冒号）就进入了底线命令模式。

底线命令模式可以输入单个或多个字符的命令，可用的命令非常多。

在底线命令模式中，基本的命令有（已经省略了冒号）：

* q 退出程序
* w 保存文件

按ESC键可随时退出底线命令模式。

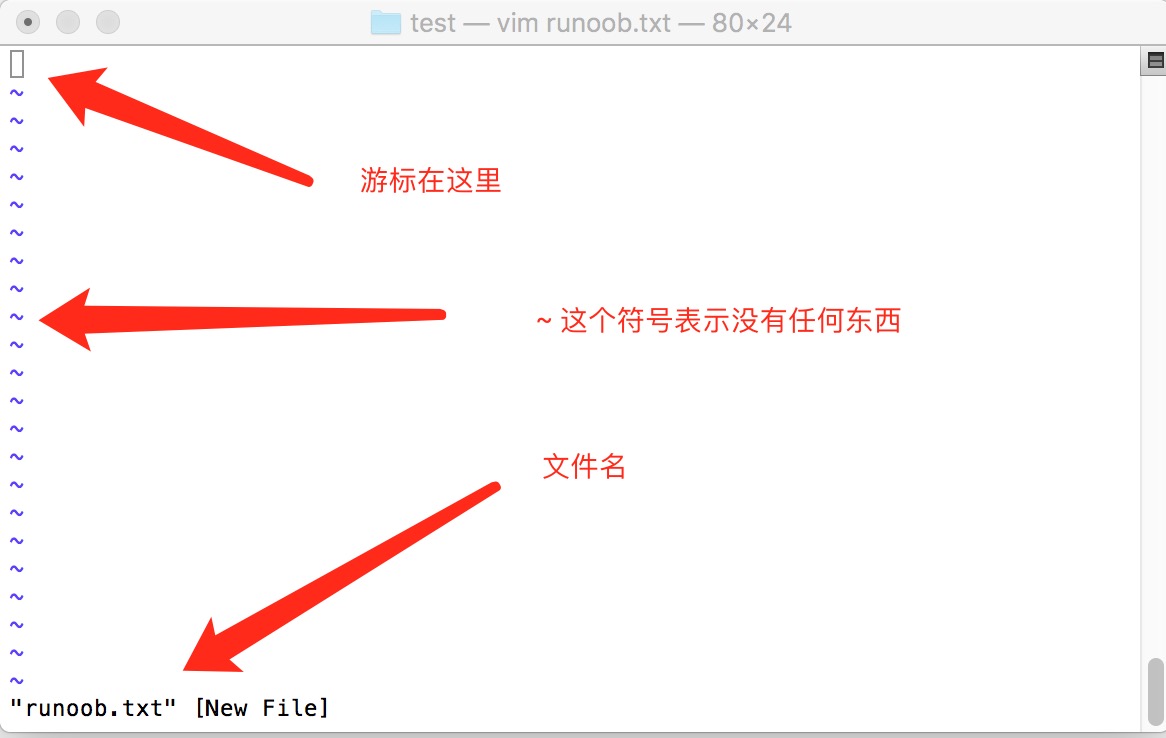
## vi/vim 使用实例

**使用 vi/vim 进入一般模式**

想要使用 vi 来建立一个名为 test.txt 的文件时，可以这样做：

$ vim runoob.txt

直接输入 **vi 文件名** 就能够进入 vi 的一般模式了。注意 vi 后面一定要加文件名，不管该文件存在与否



**按下 i 进入输入模式(也称为编辑模式)，开始编辑文字**

在一般模式之中，只要按下 i, o, a 等字符就可以进入输入模式了

在编辑模式当中，左下角状态栏中会出现 –INSERT- 的字样，那就是可以输入任意字符的提示。

这个时候，键盘上除了 Esc 这个按键之外，其他的按键都可以视作为一般的输入按钮了，可以进行任何的编辑。



**按下 ESC 按钮回到一般模式**

如何退出？按下 Esc 这个按钮即可，左下角的 – INSERT – 会消失不见

**在一般模式中按下 :wq 储存后离开 vi**

存档并离开的指令很简单，输入 :wq 即可

## 系统目录各个文件夹介绍：

/bin        ##二进制文件，系统常规命令

/boot        ##系统启动分区，系统启动时读取的文件

/dev        ##设备文件

/etc        ##大多数配置文件

/home        ##普通用户的家目录

/lib        ##32位函数库

/lib64        ##64位库

/media        ##手动临时挂载点

/mnt        ##手动临时挂载点

/opt        ##第三方软件安装位置

/proc        ##进程信息及硬件信息

/root        ##超级用户家目录

/run        ##临时设备的默认挂载点

/sbin        ##系统管理命令

/srv        ##数据

/var        ##数据

/sys        ##内核相关信息

/tmp        ##临时文件

/usr        ##用户相关设定

## 文件路径

**相对路径：**

（必须有前提条件，其实在系统底层依然是绝对路径执行，只是用户看到的简略移动）

文件相对当前系统位置的一个名称简写

文件名称省略了当前路径的值

只有当前在此目录中时可以使用

文件名称不以/开头

文件名称会自动在字符前加入'pwd'显示的路径

**绝对路径：**

是文件在系统中真实的位置

任何时间都可以精确表示一个文件的名称

文件名称以/开头

pwd            显示当前工作目录

cd               切换当前工作目录

cd /mnt       切换到/mnt目录中

cd -             进入当前目录之前所在目录中

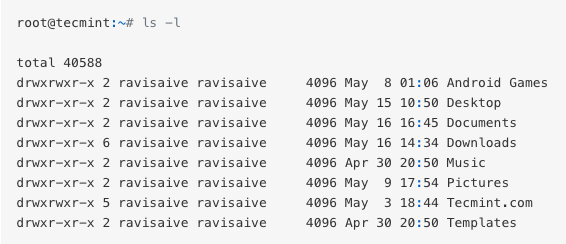
## 一些常用的命令

**1. ls**

ls命令是列出目录内容(List Directory Contents)的意思。运行它就是列出文件夹里的内容，可能是文件也可能是文件夹。



“ls -l”命令以详情模式(long listing fashion)列出文件夹的内容。



"ls -a"命令会列出文件夹里的所有内容，包括以"."开头的隐藏文件。

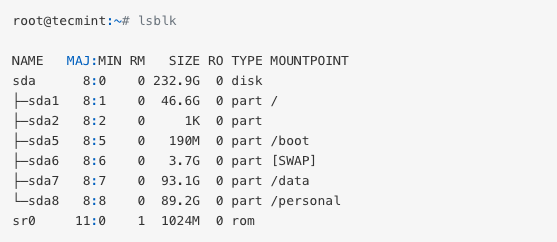


注意：在Linux中，文件以“.”开头的就是隐藏文件，并且每个文件，文件夹，设备或者命令都是以文件对待。**ls -l**命令输出：

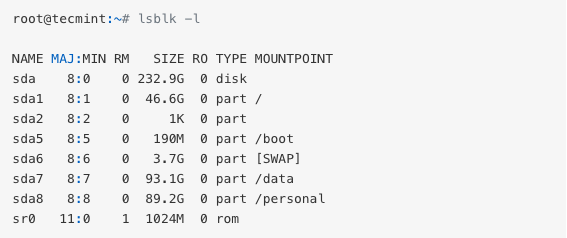
* d (代表了是目录).
* rwxr-xr-x 是文件或者目录对所属用户，同一组用户和其它用户的权限。
* 上面例子中第一个ravisaive 代表了文件文件属于用户ravisaive
* 上面例子中的第二个ravisaive代表了文件文件属于用户组ravisaive
* 4096 代表了文件大小为4096字节.
* May 8 01:06 代表了文件最后一次修改的日期和时间.
* 最后面的就是文件/文件夹的名字

**2. lsblk**

"lsblk"就是列出块设备。除了RAM外，以标准的树状输出格式，整齐地显示块设备。



“lsblk -l”命令以列表格式显示块设备(而不是树状格式)。



**3. history**

“history”命令就是历史记录。它显示了在终端中所执行过的所有命令的历史。



**4. sudo**

“sudo”(super user do)命令允许授权用户执行超级用户或者其它用户的命令。通过在sudoers列表的安全策略来指定。



**5. mkdir**

“mkdir”(Make directory)命令在命名路径下创建新的目录。然而如果目录已经存在了，那么它就会返回一个错误信息"不能创建文件夹，文件夹已经存在了"("cannot create folder, folder already exists")



注意：目录只能在用户拥有写权限的目录下才能创建。

**6. touch**

“touch”命令代表了将文件的访问和修改时间更新为当前时间。touch命令只会在文件不存在的时候才会创建它。如果文件已经存在了，它会更新时间戳，但是并不会改变文件的内容。

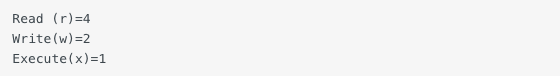


注意：touch 可以用来在用户拥有写权限的目录下创建不存在的文件。

**7. chmod**

“chmod”命令就是改变文件的模式位。chmod会根据要求的模式来改变每个所给的文件，文件夹，脚本等等的文件模式（**权限**）。

在文件（文件夹或者其它，为了简单起见，这里就使用文件）中存在3种类型的权限



如果想给文件只读权限，就设置为'4';只写权限，设置权限为'2';只执行权限，设置为1; 读写权限，就是4+2 = 6, 以此类推。

现在需要设置3种用户和用户组权限。第一个是拥有者，然后是用户所在的组，最后是其它用户。



这里root的权限是 rwx（读写和执行权限），  
所属用户组权限是 r-x (只有读和执行权限, 没有写权限)，  
对于其它用户权限是 -x(只有只执行权限)

为了改变它的权限 为拥有者、用户所在组和其它用户，提供读，写，执行权限。



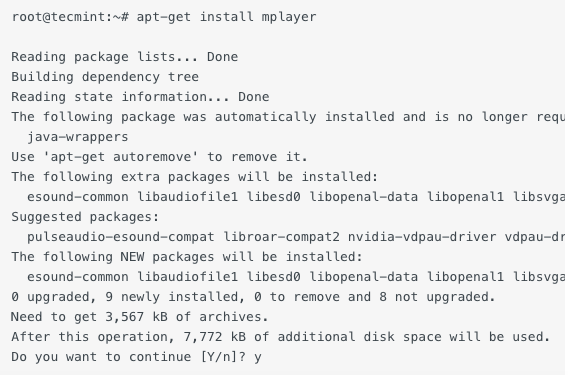
三种都只有读写权限



注意：对于系统管理员和用户来说，这个命令是最有用的命令之一。在多用户环境或者服务器上，对于某个用户，如果设置了文件不可访问，那么这个命令就可以解决，如果设置了错误的权限，那么也就提供了未授权的访问。

**8. apt**

Debian系列以“apt”命令为基础，“apt”代表了Advanced Package Tool。APT是一个为Debian系列系统（Ubuntu，Kubuntu等等）开发的高级包管理器，在Gnu/Linux系统上，它会为包自动地，智能地搜索，安装，升级以及解决依赖。

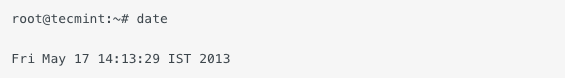


注意：上面的命令会导致系统整体的改变，所以需要root密码（查看提示符为"#"，而不是“$”）.和yum命令相比，Apt更高级和智能。

见名知义，apt-cache用来搜索包中是否包含子包mplayer, apt-get用来安装，升级所有的已安装的包到最新版。

**9. date**

“date”命令使用标准的输出打印当前的日期和时间，也可以深入设置。



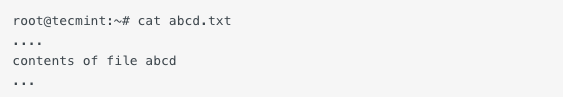


注意：这个命令在脚本中十分有用，以及基于时间和日期的脚本更完美。而且在终端中改变日期和时间，会显得很专业（需要root权限才能操作这个，因为它是系统整体改变）

**10. cat**

“cat”代表了连结（Concatenation），连接两个或者更多文本文件或者以标准输出形式打印文件的内容。





**11. cp**

“copy”，就是复制。它会从一个地方复制一个文件到另外一个地方。



**12. mv**

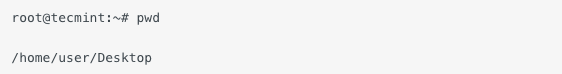
“mv”命令将一个地方的文件移动到另外一个地方去。



注意：mv 命令可以使用通配符。mv需谨慎使用，因为移动系统的或者未授权的文件不但会导致安全性问题，而且可能系统崩溃。

**13. pwd**

“pwd”（print working directory），在终端中显示当前工作目录的全路径。

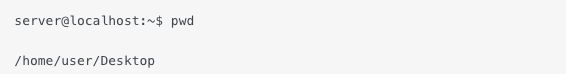


注意： 这个命令并不会在脚本中经常使用，但是对于新手，当从连接到nux很久后在终端中迷失了路径，这绝对是救命稻草。

**14. cd**

最后，经常使用的“cd”命令代表了改变目录。它在终端中改变工作目录来执行，复制，移动，读，写等等操作。





注意： 在终端中切换目录时，cd就大显身手了。“cd ～”会改变工作目录为用户的家目录，而且当用户发现自己在终端中迷失了路径时，非常有用。“cd ..”从当前工作目录切换到(当前工作目录的)父目录。