## ls命令

ls 命令不仅可以查看 linux 文件夹包含的文件而且可以查看文件权限(包括目录、文件夹、文件权限)查看目录信息等等。

命令格式：ls [选项][目录名]

常用参数

-l ：列出长数据串，包含文件的属性与权限数据等

-a ：列出全部的文件，连同隐藏文件（开头为.的文件）一起列出来（常用）

-d ：仅列出目录本身，而不是列出目录的文件数据

-h ：将文件容量以较易读的方式（GB，kB等）列出来

-R ：连同子目录的内容一起列出（递归列出），等于该目录下的所有文件都会显示出来

## cd命令

最基本的命令语句，其他的命令语句要进行操作，都是建立在使用 cd 命令上的。用于切换当前目录至dirName。

命令格式：cd [目录名]

## pwd命令

查看"当前工作目录"的完整路径。

命令格式：pwd [选项]

常用参数：

-P :显示实际物理路径，而非使用连接（link）路径

-L :当目录为连接路径时，显示连接路径

## mkdir命令

用来创建指定的名称的目录，要求创建目录的用户在当前目录中具有写权限，并且指定的目录名不能是当前目录中已有的目录。

命令格式：mkdir [选项] 目录

常用参数

-m, --mode=模式，设定权限<模式> (类似 chmod)，而不是 rwxrwxrwx 减 umask

-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;

-v, --verbose 每次创建新目录都显示信息

--help 显示此帮助信息并退出

--version 输出版本信息并退出

## rm命令

删除一个目录中的一个或多个文件或目录，如果没有使用- r选项，则rm不会删除目录。如果使用 rm 来删除文件，通常仍可以将该文件恢复原状。

命令格式：rm [选项] 文件

常用参数

-f, --force 忽略不存在的文件，从不给出提示。

-i, --interactive 进行交互式删除

-r, -R, --recursive 指示rm将参数中列出的全部目录和子目录均递归地删除。

-v, --verbose 详细显示进行的步骤

--help 显示此帮助信息并退出

--version 输出版本信息并退出

## rmdir命令

该命令从一个目录中删除一个或多个子目录项，删除某目录时也必须具有对父目录的写权限。

命令格式：rmdir [选项] 目录

常用参数

-p 递归删除目录dirname，当子目录删除后其父目录为空时，也一同被删除。如果整个路径被删除或者由于某种原因保留部分路径，则系统在标准输出上显示相应的信息。

-v, --verbose 显示指令执行过程

## mv命令

可以用来移动文件或者将文件改名（move (rename) files）。当第二个参数类型是文件时，mv命令完成文件重命名。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。

命令格式：mv [选项] 源文件或目录 目标文件或目录

常用参数

-b ：若需覆盖文件，则覆盖前先行备份

-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

-u ：若目标文件已经存在，且 source 比较新，才会更新(update)

-t ：--target-directory=DIRECTORY move all SOURCE arguments into DIRECTORY，即指定mv的目标目录，该选项适用于移动多个源文件到一个目录的情况，此时目标目录在前，源文件在后

## cp命令

将源文件复制至目标文件，或将多个源文件复制至目标目录。

命令格式：cp [选项] 源文件 目录 或 cp [选项] -t 目录 源文件

常用参数

-t --target-directory 指定目标目录

-i --interactive 覆盖前询问（使前面的 -n 选项失效）

-n --no-clobber 不要覆盖已存在的文件（使前面的 -i 选项失效）

-f --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

-u --update 使用这项参数之后，只会在源文件的修改时间较目的文件更新时，或是对应的目的文件并不存在，才复制文件

## cat 命令

用来显示文件内容，或者将几个文件连接起来显示，或者从标准输入读取内容并显示，它常与重定向符号配合使用。

命令格式：cat [选项] [文件]

常用参数

-A, --show-all 等价于 -vET

-b, --number-nonblank 对非空输出行编号

-e 等价于 -vE

-E, --show-ends 在每行结束处显示 $

-n, --number 对输出的所有行编号,由1开始对所有输出的行数编号

-s, --squeeze-blank 有连续两行以上的空白行，就代换为一行的空白行

-t 与 -vT 等价

-T, --show-tabs 将跳格字符显示为 ^I

-v, --show-nonprinting 使用 ^ 和 M- 引用，除了 LFD 和 TAB 之外

## more命令

more 命令和 cat 的功能一样都是查看文件里的内容，但有所不同的是more可以按页来查看文件的内容，还支持直接跳转行等功能。

命令格式：more [-dlfpcsu ] [-num ] [+/ pattern] [+ linenum] [file ... ]

常用参数

+n 从笫n行开始显示

-n 定义屏幕大小为n行

+/pattern 在每个档案显示前搜寻该字串（pattern），然后从该字串前两行之后开始显示

-c 从顶部清屏，然后显示

-d 提示“Press space to continue，’q’ to quit（按空格键继续，按q键退出）”，禁用响铃功能

-l 忽略Ctrl+l（换页）字符

-p 通过清除窗口而不是滚屏来对文件进行换页，与-c选项相似

-s 把连续的多个空行显示为一行

-u 把文件内容中的下画线去掉

操作指令

Enter：向下n行，需要定义。默认为1行

Ctrl+F：向下滚动一屏

空格键：向下滚动一屏

Ctrl+B：返回上一屏

= ：输出当前行的行号

：f ：输出文件名和当前行的行号

V ：调用vi编辑器

!命令 ：调用Shell，并执行命令

q ：退出more

## less命令

less 与 more 类似，但使用 less 可以随意浏览文件，而 more 仅能向前移动，却不能向后移动，而且 less 在查看之前不会加载整个文件。

命令格式：less [参数] 文件

常用参数

-b <缓冲区大小> 设置缓冲区的大小

-e 当文件显示结束后，自动离开

-f 强迫打开特殊文件，例如外围设备代号、目录和二进制文件

-g 只标志最后搜索的关键词

-i 忽略搜索时的大小写

-m 显示类似more命令的百分比

-N 显示每行的行号

-o <文件名> 将less 输出的内容在指定文件中保存起来

-Q 不使用警告音

-s 显示连续空行为一行

-S 行过长时间将超出部分舍弃

-x <数字> 将“tab”键显示为规定的数字空格

操作命令

/字符串：向下搜索“字符串”的功能

?字符串：向上搜索“字符串”的功能

n：重复前一个搜索（与 / 或 ? 有关）

N：反向重复前一个搜索（与 / 或 ? 有关）

b 向后翻一页

d 向后翻半页

h 显示帮助界面

Q 退出less 命令

u 向前滚动半页

y 向前滚动一行

空格键 滚动一行

回车键 滚动一页

[pagedown]：向下翻动一页

[pageup]：向上翻动一页

## head命令

head 用来显示档案的开头至标准输出中，默认 head 命令打印其相应文件的开头 10 行。

命令格式：head [参数] [文件]

常用参数

-q 隐藏文件名

-v 显示文件名

-c<字节> 显示字节数

-n<行数> 显示的行数

## tail命令

显示指定文件末尾内容，不指定文件时，作为输入信息进行处理。常用查看日志文件。

命令格式：tail [必要参数] [选择参数] [文件]

常用参数

-f 循环读取

-q 不显示处理信息

-v 显示详细的处理信息

-c<数目> 显示的字节数

-n<行数> 显示行数

--pid=PID 与-f合用,表示在进程ID,PID死掉之后结束.

-q, --quiet, --silent 从不输出给出文件名的首部

-s, --sleep-interval=S 与-f合用,表示在每次反复的间隔休眠S秒

## vim命令

vim编辑器是所有Unix及Linux系统下标准的编辑器，它的强大不逊色于任何最新的文本编辑器;vi也是Linux中最基本的文本编辑器,vim就是vi的升级版。

### 启动vim

在命令行窗口中输入以下命令即可

vim

直接启动vim

vim filename 打开vim并创建名为filename的文件

### vim的模式

正常模式（按Esc或Ctrl+[进入） 左下角显示文件名或为空

插入模式（按i键进入） 左下角显示--INSERT--

可视模式（不知道如何进入） 左下角显示--VISUAL--

导航命令

% 括号匹配

### 插入命令

i 在当前位置生前插入

I 在当前行首插入

a 在当前位置后插入

### 查找命令

/text　查找text，按n健查找下一个，按N健查找前一个。

?text　查找text，反向查找，按n健查找下一个，按N健查找前一个。

:set hlsearch　高亮搜索结果，所有结果都高亮显示，而不是只显示一个匹配。

:set nohlsearch　关闭高亮搜索显示

:nohlsearch　关闭当前的高亮显示，如果再次搜索或者按下n或N键，则会再次高亮。

:set incsearch　逐步搜索模式，对当前键入的字符进行搜索而不必等待键入完成。

:set wrapscan　重新搜索，在搜索到文件头或尾时，返回继续搜索，默认开启。

:set nu 显示行号

### 替换命令

ra 将当前字符替换为a，当期字符即光标所在字符。

s/old/new/ 用old替换new，替换当前行的第一个匹配

### 撤销和重做

u 撤销（Undo）

U 撤销对整行的操作

Ctrl + r 重做（Redo），即撤销的撤销。

### 删除命令

x 删除当前字符

3x 删除当前光标开始向后三个字符

X 删除当前字符的前一个字符。X=dh

dl 删除当前字符， dl=x

dh 删除前一个字符

dd 删除当前行

10d 删除当前行开始的10行。

D 删除当前字符至行尾。D=d$

d$ 删除当前字符之后的所有字符（本行）

### 拷贝和粘贴

yy 拷贝当前行

nyy 拷贝当前后开始的n行，比如2yy拷贝当前行及其下一行。

p 在当前光标后粘贴,如果之前使用了yy命令来复制一行，那么就在当前行的下一行粘贴。

shift+p 在当前行前粘贴

### 退出命令

:wq 保存并退出

ZZ 保存并退出

:q! 强制退出并忽略所有更改

:e! 放弃所有修改，并打开原来文件。

### 帮助命令

:help or F1 显示整个帮助

:help xxx 显示xxx的帮助，比如 :help i, :help CTRL-[（即Ctrl+[的帮助）。

:help 'number' Vim选项的帮助用单引号括起

:help 特殊键的帮助用<>扩起

:help -t Vim启动参数的帮助用-

：help i\_ 插入模式下Esc的帮助，某个模式下的帮助用模式\_主题的模式

帮助文件中位于||之间的内容是超链接，可以用Ctrl+]进入链接，Ctrl+o（Ctrl + t）返回

其他非编辑命令