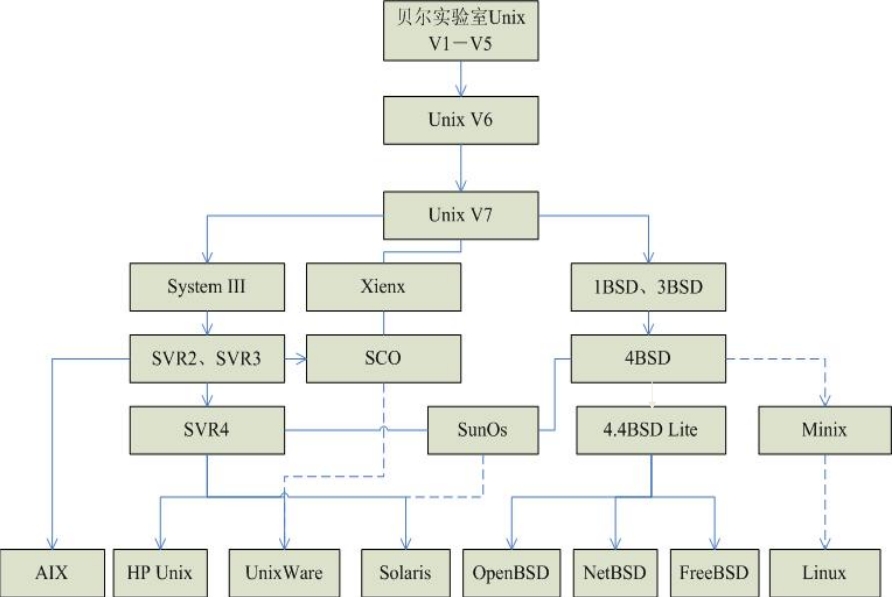
Linux的历史

– Unix

– Minix

– GNU

– BSD



Linux概述

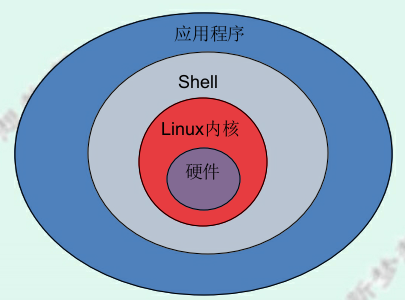
Linux 派生于 UNIX 操作系统。

• 创始人Linus Torvalds

• Linux 的发行遵守 GNU 的通用公共许可证，它是一 个自由软件，内核是源代码公开的。

• Linux 起初为基于 386/486 的 PC 机开发，但现在， Linux 也可以运行在 PC服务器、小型机、大型巨型 机上。

• Linux具有：开放性、多用户、多任务、出色的性能、 良好的用户界面、丰富的网络功能、可靠的安全性、 良好的可移植性、兼容性等特点。



内核：是Linux系统的心脏，管理着磁盘、网卡、CPU等硬件设备，管理进程、 时间等资源的核心程序；

shell：是系统的用户界面，提供了用户与内核交互的接口；

应用程序：像文本编辑器、数据库、java、php等；

基本知识

内核版本-----主版本号.次版本号.修订 次数

• 发布版本之间的区别

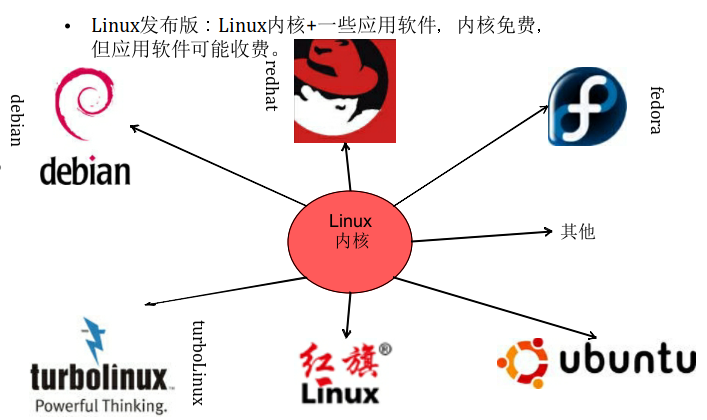
– 次版本号为奇数版本---测试版本，表示新添加了一些功能， 可能不是很稳定；

• 2.5.1

– 次版本号为偶数版本---稳定版本

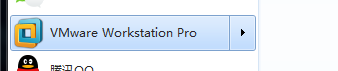
• 2.6.1

– 内核官网：www.kernel.org

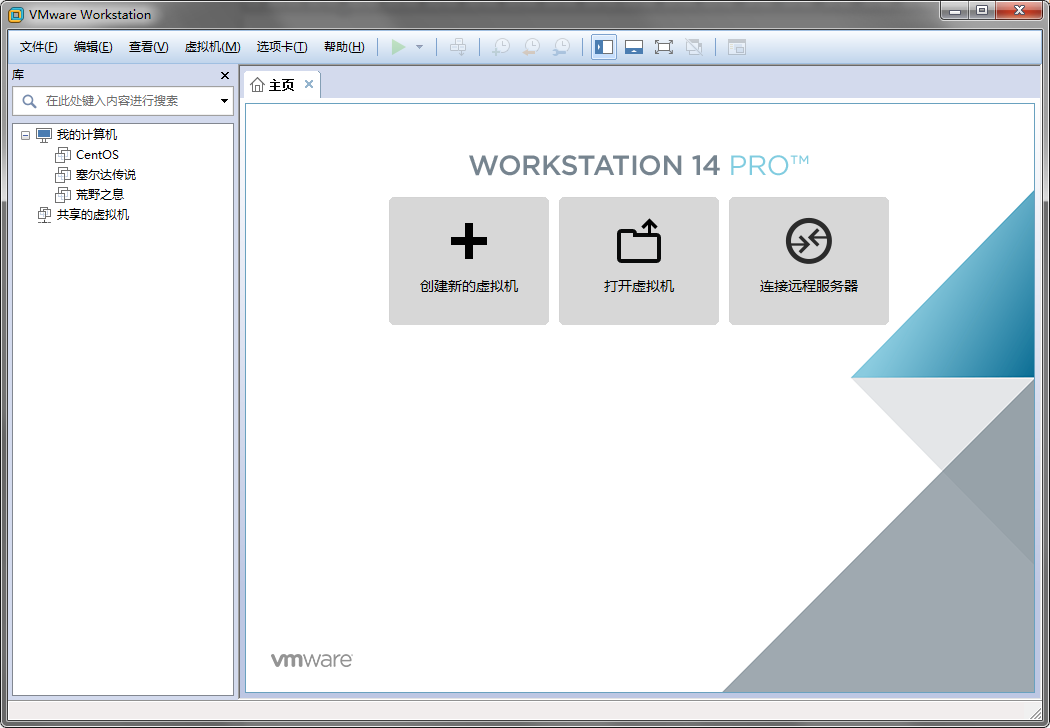


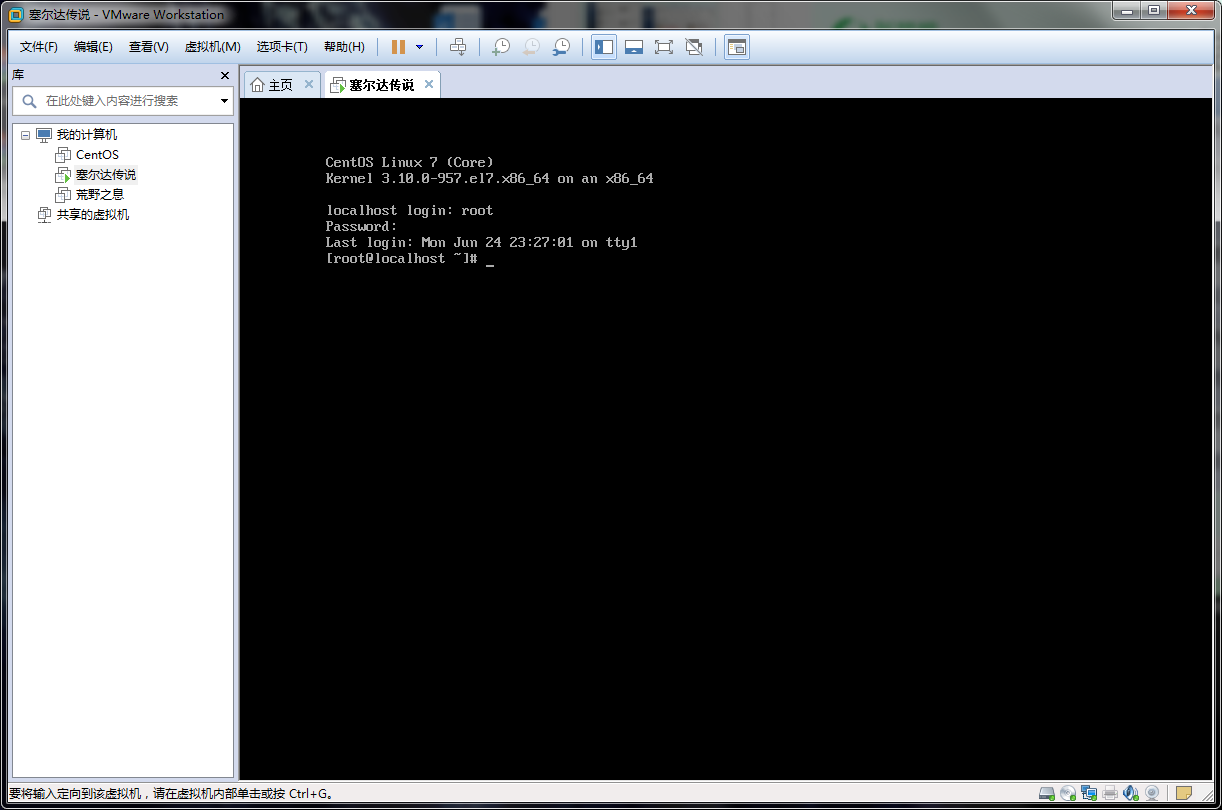
**虚拟机操作**

打开VMware虚拟机软件

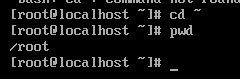


打开虚拟机





返回当前根目录

 通过cd ~来返回当前系统的根目录

cd .. 返回上次目录

cd ~ 转到用户目录

cd - 回到上一次所在目录

cd ../dir 回到上层目录后又进入dir目录

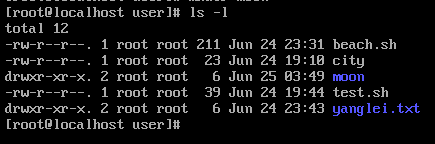
cd / 回到根目录

cd：切换目录命令，可按照绝对路径、相对路径进行切换，如：

cd ~ ; cd /;cd /mnt;cd /usr/local/bin;cd ../a;

进入某个文件夹

建立moon为名字的文件夹

通过ls -l调取当前目录下的所有内容的详情

ls：显示文件和目录列表，可显示当前目录下的文件、文件夹，也可以实现其他路径 下的文件、文件夹，如：

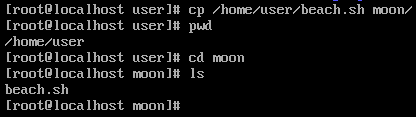
ls -l --按列表显示所有文件

ls -a --显示所有文件，包括隐藏文件（隐藏文件名以.开头）

ls -la --按列表显示所有文件，包括隐藏文件

ls -a /sbin/a\* --显示/sbin/目录下a字母开头的文件；

 pwd显示当前所在的文件夹

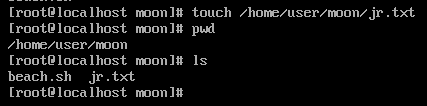


cp：复制文件、文件夹

cp a.txt a1.txt -----将文件a.txt复制成a1.txt

cp /root/a/\* ./ -----将文件夹a下的所有文件（不包含文件夹）复制到根目录下

cp -R /bin/\* ./ ------将文件夹bin目录下的所有文件和目录等拷贝到根目录下



touch：创建新文件（创建文本文件）

touch 1 -----创建文件1

touch file1 file2 -----创建文件 file1 文件file2

注：若遇到命令不熟，则可使用如help cd查看其帮助

\* - 通配符,代表任意字符(0到多个)

? - 通配符,代表一个字符

# - 注释

/ - 跳转符号,将特殊字符或通配符还原成一般符号

| - 分隔两个管线命令的界定

; - 连续性命令的界定

~ - 用户的根目录

$ - 变量前需要加的变量值

! - 逻辑运算中的"非"(not)

/ - 路径分隔符号

>, >> - 输出导向,分别为"取代"与"累加"

' - 单引号,不具有变量置换功能

" - 双引号,具有变量置换功能

` - quote符号,两个``中间为可以先执行的指令

() - 中间为子shell的起始与结束

[] - 中间为字符组合

{} - 中间为命令区块组合

Ctrl+C - 终止当前命令

Ctrl+D - 输入结束(EOF),例如邮件结束的时候

Ctrl+M - 就是Enter

Ctrl+S - 暂停屏幕的输出

Ctrl+Q - 恢复屏幕的输出

Ctrl+U - 在提示符下,将整行命令删除

Ctrl+Z - 暂停当前命令

&& - 当前一个指令执行成功时,执行后一个指令

|| - 当前一个指令执行失败时,执行后一个指令

常用的通配符：

其中最常用的是＊、？、［］和 ‘。下面举几个简单的例子：

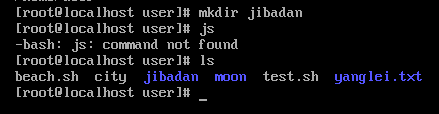
1，ls test\* <== \*表示后面不论接几个字符都接受（没有字符也接受）

2，ls test? <== ?表示后面当且仅当接一个字符时才接受

3，ls test??? <== ???表示一定要接三个字符

4，cp test[1~5] /tmp <== test1, test2, test3, test4, test5若存在，则复制到/tmp目录下

5，cd /lib/modules/' uname -r'/kernel/drivers <== 被 ' ' 括起来的命令先执行

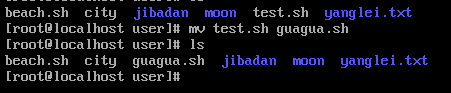


mkdir:创建文件夹

mkdir a --创建文件夹a

mkdir 1 2 --同时创建文件夹1 2

mkdir -p ./1/2/3 --创建多级目录



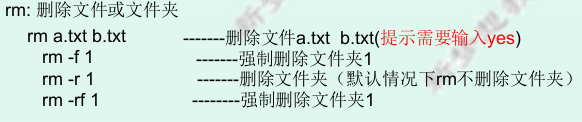
mv：剪切或文件、文件夹重命名

mv a.txt b.txt -----将文件a.txt 重命名为b.txt

mv a.txt /root/a.txt -----将文件a.txt 移动到root目录下

mv 1 2 ------将文件夹1 重命名为 2

mv ./\*.txt /root/ -------将根目录下的所有txt文件移动到root目录下



rmdir:只能删除空文件夹

rmdir 1 ----删除文件夹1

rmdir 2 3 ----删除文件夹2 文件夹3

vi命令

**vi的基本概念：**

　基本上vi可以分为三种状态，分别是命令模式（command mode）、插入模式（Insert mode）和底行模式（last line mode），各模式的功能区分如下：

1) 命令行模式command mode）

　　控制屏幕光标的移动，字符、字或行的删除，移动复制某区段及进入Insert mode下，或者到 last line mode。

2) 插入模式（Insert mode）

　　只有在Insert mode下，才可以做文字输入，按「ESC」键可回到命令行模式。

3) 底行模式（last line mode）

　　将文件保存或退出vi，也可以设置编辑环境，如寻找字符串、列出行号……等。

不过一般我们在使用时把vi简化成两个模式，就是将底行模式（last line mode）也算入命令行模式command mode）。

vi的基本操作

a) 进入vi

　在系统提示符号输入vi及文件名称后，进入vi之后，是处于「命令行模式（command mode）」，您要切换到「插入模式（Insert mode）」才能够输入文字。初次使用vi的人都会想先用上下左右键移动光标，结果电脑一直哔哔叫，把自己气个半死，所以进入vi后，先不要乱动，转换到「插入模式（Insert mode）」再说吧！

b) 切换至插入模式（Insert mode）编辑文件

　　在「命令行模式（command mode）」下按一下字母「i」就可以进入「插入模式（Insert mode）」，这时候你就可以开始输入文字了。

c) Insert 的切换

　　目前处于「插入模式（Insert mode）」，就只能一直输入文字，如果发现输错了字！想用光标键往回移动，将该字删除，就要先按一下「ESC」键转到「命令行模式（command mode）」再删除文字。

d) 退出vi及保存文件

　　在「命令行模式（command mode）」下，按一下「：」冒号键进入「Last line mode」，例如：

: w filename （输入 「w filename」将文章以指定的文件名filename保存）

: wq (输入「wq」，存盘并退出vi)

: q! (输入q!， 不存盘强制退出vi)

3、命令行模式（command mode）功能键

1）. 插入模式

按「i」切换进入插入模式「insert mode」，按"i"进入插入模式后是从光标当前位置开始输入文件；

　　按「a」进入插入模式后，是从目前光标所在位置的下一个位置开始输入文字；

　　按「o」进入插入模式后，是插入新的一行，从行首开始输入文字。

2）. 从插入模式切换为命令行模式

按「ESC」键。

3）. 移动光标

　　vi可以直接用键盘上的光标来上下左右移动，但正规的vi是用小写英文字母「h」、「j」、「k」、「l」，分别控制光标左、下、上、右移一格。

　　按「ctrl」+「b」：屏幕往"后"移动一页。

　　按「ctrl」+「f」：屏幕往"前"移动一页。

　　按「ctrl」+「u」：屏幕往"后"移动半页。

　　按「ctrl」+「d」：屏幕往"前"移动半页。

　　按数字「0」：移到文章的开头。

　　按「G」：移动到文章的最后。

　　按「$」：移动到光标所在行的"行尾"。

　　按「^」：移动到光标所在行的"行首"

　　按「w」：光标跳到下个字的开头

　　按「e」：光标跳到下个字的字尾

　　按「b」：光标回到上个字的开头

　　按「#l」：光标移到该行的第#个位置，如：5l,56l。

4）. 删除文字

　　「x」：每按一次，删除光标所在位置的"后面"一个字符。

　　「#x」：例如，「6x」表示删除光标所在位置的"后面"6个字符。

　　「X」：大写的X，每按一次，删除光标所在位置的"前面"一个字符。

　　「#X」：例如，「20X」表示删除光标所在位置的"前面"20个字符。

　　「dd」：删除光标所在行。

　　「#dd」：从光标所在行开始删除#行

5）. 复制

　　「yw」：将光标所在之处到字尾的字符复制到缓冲区中。

　　「#yw」：复制#个字到缓冲区

　　「yy」：复制光标所在行到缓冲区。

　　「#yy」：例如，「6yy」表示拷贝从光标所在的该行"往下数"6行文字。

　　「p」：将缓冲区内的字符贴到光标所在位置。注意：所有与"y"有关的复制命令都必须与"p"配合才能完成复制与粘贴功能。

6）. 替换

　　「r」：替换光标所在处的字符。

　　「R」：替换光标所到之处的字符，直到按下「ESC」键为止。

7）. 回复上一次操作

　　「u」：如果您误执行一个命令，可以马上按下「u」，回到上一个操作。按多次"u"可以执行多次回复。

8）. 更改

　　「cw」：更改光标所在处的字到字尾处

　　「c#w」：例如，「c3w」表示更改3个字

9）. 跳至指定的行

　　「ctrl」+「g」列出光标所在行的行号。

　　「#G」：例如，「15G」，表示移动光标至文章的第15行行首。

4、Last line mode下命令简介

　　在使用「last line mode」之前，请记住先按「ESC」键确定您已经处于「command mode」下后，再按「：」冒号即可进入「last line mode」。

A) 列出行号

　「set nu」：输入「set nu」后，会在文件中的每一行前面列出行号。

B) 跳到文件中的某一行

　「#」：「#」号表示一个数字，在冒号后输入一个数字，再按回车键就会跳到该行了，如输入数字15，再回车，就会跳到文章的第15行。

C) 查找字符

　「/关键字」：先按「/」键，再输入您想寻找的字符，如果第一次找的关键字不是您想要的，可以一直按「n」会往后寻找到您要的关键字为止。

　「?关键字」：先按「?」键，再输入您想寻找的字符，如果第一次找的关键字不是您想要的，可以一直按「n」会往前寻找到您要的关键字为止。

D) 保存文件

　「w」：在冒号输入字母「w」就可以将文件保存起来。

E) 离开vi

　「q」：按「q」就是退出，如果无法离开vi，可以在「q」后跟一个「!」强制离开vi。

　「qw」：一般建议离开时，搭配「w」一起使用，这样在退出的时候还可以保存文件。

5、vi命令列表

1、下表列出命令模式下的一些键的功能：

h

左移光标一个字符

l

右移光标一个字符

k

光标上移一行

j

光标下移一行

^

光标移动至行首

0

数字"0"，光标移至文章的开头

G

光标移至文章的最后

$

光标移动至行尾

Ctrl+f

向前翻屏

Ctrl+b

向后翻屏

Ctrl+d

向前翻半屏

Ctrl+u

向后翻半屏

i

在光标位置前插入字符

a

在光标所在位置的后一个字符开始增加

o

插入新的一行，从行首开始输入

ESC

从输入状态退至命令状态

x

删除光标后面的字符

#x

删除光标后的＃个字符

X

(大写X)，删除光标前面的字符

#X

删除光标前面的#个字符

dd

删除光标所在的行

#dd

删除从光标所在行数的#行

yw

复制光标所在位置的一个字

#yw

复制光标所在位置的#个字

yy

复制光标所在位置的一行

#yy

复制从光标所在行数的#行

p

粘贴

u

取消操作

cw

更改光标所在位置的一个字

#cw

更改光标所在位置的#个字

2、下表列出行命令模式下的一些指令

w filename

储存正在编辑的文件为filename

wq filename

储存正在编辑的文件为filename，并退出vi

q!

放弃所有修改，退出vi

set nu

显示行号

/或?

查找，在/后输入要查找的内容

n

与/或?一起使用，如果查找的内容不是想要找的关键字，按n或向后（与/联用）或向前（与?联用）继续查找，直到找到为止。

对于第一次用vi，有几点注意要提醒一下：

1、用vi打开文件后，是处于「命令行模式（command mode）」，您要切换到「插入模式（Insert mode）」才能够输入文字。切换方法：在「命令行模式（command mode）」下按一下字母「i」就可以进入「插入模式（Insert mode）」，这时候你就可以开始输入文字了。

2、编辑好后，需从插入模式切换为命令行模式才能对文件进行保存，切换方法：按「ESC」键。

3、保存并退出文件：在命令模式下输入:wq即可！（别忘了wq前面的:）

在屏幕上移动

Vi提供了三个关于光标在全屏幕上移动并且文件本身不发生滚动的命令。它们分别是H、M和L命令。

H命令

该命令将光标移至屏幕首行的行首（即左上角），也就是当前屏幕的第一行，而不是整个文件的第一行。利用此命令可以快速将光标移至屏幕顶部。若在H命令之前加上数字n，则将光标移至第n行的行首。

M命令

该命令将光标移至屏幕显示文件的中间行的行首。即如果当前屏幕已经充满,则移动到整个屏幕的中间行；如果并未充满，则移动到文本的那些行的中间行。利用此命令可以快速地将光标从屏幕的任意位置移至屏幕显示文件的中间行的行首。例如，在上面屏幕显示的情况下（不论光标在屏幕的何处），在命令模式下，输入命令M之后，光标都将移到add这一行的字母a上。

同样值得一提的是，使用命令dM将会删除从光标当前所在行至屏幕显示文件的中间行的全部内容。

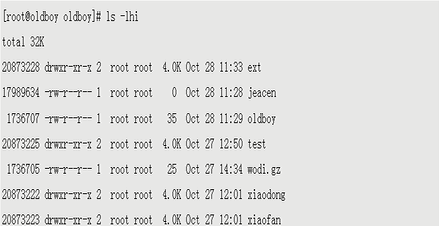
L命令

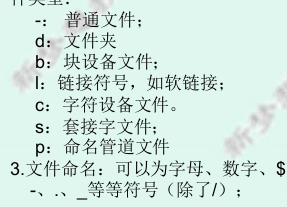
当文件显示内容超过一屏时，该命令将光标移至屏幕上的最底行的行首；当文件显示内容不足一屏时，该命令将光标移至文件的最后一行的行首。可见，利用此命令可以快速准确地将光标移至屏幕底部或文件的最后一行。若在L命令之前加上数字n，则将光标移至从屏幕底部算起第n行的行首。例如，在上面屏幕显示的情况下（不论光标在屏幕的何处），在命令模式下，输入命令3L之后，光标都将移到x++这一行的字母x上。

**文件及权限**

系统中文件或目录的属性主要包括：文件或目录的索引节点（inode）、类型、权限属性、链接数、所归属的用户和用户组、最近修改时间等内容；

看一下执行ls -lhi命令的结果:





第一列：inode索引节点

第二列：文件类型及权限

第三列：硬链接个数

第四列：文件或目录所属的用户（属主）

第五列：文件或目录所属的用户所归属的组；

第六列：文件或目录的大小；

第七列、八、九列：文件或目录的修改时间

第十列：实际的文件名或目录名