linux下Vi编辑器命令大全(网上找滴)  
2007-04-23 18:09/根目录下的文件夹  
输入”/”,回车  
里面有很多文件夹,如:etc,home,lib,mnt等等  
etc:存放用户名和密码  
home:每个用户有个文件夹,存放在这里  
lib:操作系统文件等  
mnt:挂扣用的  
opt:java工具  
bea:web编程的一些东西  
sbin,bin,usr:这三个文件夹都是存放unix的命令的  
tmp:临时文件夹  
常用的操作:  
查看当前目录:pwd  
修改密码:passwd  
更换模式:csh,bash  
远程登陆:telnet 服务器ip  
回到上级目录:cd ..  
查看该目录下的文件夹:ls  
打开文件:vi /文件夹/文件  
文件显示的格式:root:x:0:root…:/root;/bin/bash 用户名,加密,序号,组,默认路径  
退出:esc shift+: q! quit exit  
清空:clear  
显示当前目录所有文件:ls –a  
显示当前目录所有文件和详细信息:ls –al  
新建文件:vi 文件名  
文件的权限:chmod g –(+)r w x u g o s   
新建文件(不能编辑)更新文件的访问时间,如果存在文件,更新,不存在,建立:touch 文件名  
创建目录:mkdir 目录名/目录名  
创建多个目录:mkdir 目录名 目录名…..  
在主目录创建下目录:mkdir ~/目录名  
创建多级目录:mkdir –p 目录名/目录名  
在覆盖文件前做个提示:cp –I  
文件和文件内容间的拷贝：cp  
复制目录：cp –r 目录　目录  
复制文件：cp 文件　文件  
移动文件：mv 要移动的文件　要移动到的目录  
移动目录：mv -fi要移动的目录　要移动到的目录  
修改文件或目录的名字：mv 文件　名字  
删除文件：rm rm –I提示  
删除目录中的所有内容包括目录：rm –r 要删除的文件夹  
连接：软连接：相当于创建快捷方式，作用：修改linux内核会用到他 in  
硬连接：如果删除源文件，被连接的文件内容还存在  
帮助：man 命令  
补全名字：table  
显示日历：cal  
输入：＞  
输出：＜  
查看文件内容：cat 例子：cat<文件>文件  
分屏显示：ls –l/目录｜more  
合并到一个新的文件：cat a.txt b.txt  
设置环境变量：  
csh中　.cshrc source cshrc  
bash中　.profile 执行..profile或者source .profile  
  
本编辑器是所有计算机系统中最常用的一种工具。UNIX下的编辑器有ex,sed和vi等，其中，使用最为广泛的是vi，而vi命令繁多，论坛里好像这方面的总结不多，以下稍做总结，以资共享！渴望更正和补充！  
  
进入vi的命令  
vi filename :打开或新建文件，并将光标置于第一行首  
vi +n filename ：打开文件，并将光标置于第n行首  
vi + filename ：打开文件，并将光标置于最后一行首  
vi +/pattern filename：打开文件，并将光标置于第一个与pattern匹配的串处  
vi -r filename ：在上次正用vi编辑时发生系统崩溃，恢复filename  
vi filename....filename ：打开多个文件，依次进行编辑  
  
移动光标类命令  
h ：光标左移一个字符  
l ：光标右移一个字符  
space：光标右移一个字符  
Backspace：光标左移一个字符  
k或Ctrl+p：光标上移一行  
j或Ctrl+n ：光标下移一行  
Enter ：光标下移一行  
w或W ：光标右移一个字至字首  
b或B ：光标左移一个字至字首  
e或E ：光标右移一个字至字尾  
) ：光标移至句尾  
( ：光标移至句首  
}：光标移至段落开头  
{：光标移至段落结尾  
nG：光标移至第n行首  
n+：光标下移n行  
n-：光标上移n行  
n$：光标移至第n行尾  
H ：光标移至屏幕顶行  
M ：光标移至屏幕中间行  
L ：光标移至屏幕最后行  
0：（注意是数字零）光标移至当前行首  
$：光标移至当前行尾  
  
屏幕翻滚类命令  
Ctrl+u：向文件首翻半屏  
Ctrl+d：向文件尾翻半屏  
Ctrl+f：向文件尾翻一屏  
Ctrl＋b；向文件首翻一屏  
nz：将第n行滚至屏幕顶部，不指定n时将当前行滚至屏幕顶部。  
  
插入文本类命令  
i ：在光标前  
I ：在当前行首  
a：光标后  
A：在当前行尾  
o：在当前行之下新开一行  
O：在当前行之上新开一行  
r：替换当前字符  
R：替换当前字符及其后的字符，直至按ESC键  
s：从当前光标位置处开始，以输入的文本替代指定数目的字符  
S：删除指定数目的行，并以所输入文本代替之  
ncw或nCW：修改指定数目的字  
nCC：修改指定数目的行  
  
删除命令  
ndw或ndW：删除光标处开始及其后的n-1个字  
do：删至行首  
d$：删至行尾  
ndd：删除当前行及其后n-1行  
x或X：删除一个字符，x删除光标后的，而X删除光标前的  
Ctrl+u：删除输入方式下所输入的文本  
  
搜索及替换命令  
/pattern：从光标开始处向文件尾搜索pattern  
?pattern：从光标开始处向文件首搜索pattern  
n：在同一方向重复上一次搜索命令  
N：在反方向上重复上一次搜索命令  
：s/p1/p2/g：将当前行中所有p1均用p2替代  
：n1,n2s/p1/p2/g：将第n1至n2行中所有p1均用p2替代  
：g/p1/s//p2/g：将文件中所有p1均用p2替换  
  
选项设置  
all：列出所有选项设置情况  
term：设置终端类型  
ignorance：在搜索中忽略大小写  
list：显示制表位(Ctrl+I)和行尾标志（$)  
number：显示行号  
report：显示由面向行的命令修改过的数目  
terse：显示简短的警告信息  
warn：在转到别的文件时若没保存当前文件则显示NO write信息  
nomagic：允许在搜索模式中，使用前面不带“\”的特殊字符  
nowrapscan：禁止vi在搜索到达文件两端时，又从另一端开始  
mesg：允许vi显示其他用户用write写到自己终端上的信息  
  
最后行方式命令  
：n1,n2 co n3：将n1行到n2行之间的内容拷贝到第n3行下  
：n1,n2 m n3：将n1行到n2行之间的内容移至到第n3行下  
：n1,n2 d ：将n1行到n2行之间的内容删除  
：w ：保存当前文件  
：e filename：打开文件filename进行编辑  
：x：保存当前文件并退出  
：q：退出vi  
：q!：不保存文件并退出vi  
：!command：执行shell命令command  
：n1,n2 w!command：将文件中n1行至n2行的内容作为command的输入并执行之，若不指定n1，n2，则表示将整个文件内容作为command的输入  
：r!command：将命令command的输出结果放到当前行  
  
寄存器操作  
"?nyy：将当前行及其下n行的内容保存到寄存器？中，其中?为一个字母，n为一个数字  
"?nyw：将当前行及其下n个字保存到寄存器？中，其中?为一个字母，n为一个数字  
"?nyl：将当前行及其下n个字符保存到寄存器？中，其中?为一个字母，n为一个数字  
"?p：取出寄存器？中的内容并将其放到光标位置处。这里？可以是一个字母，也可以是一个数字  
ndd：将当前行及其下共n行文本删除，并将所删内容放到1号删除寄存

VI的使用  
--------------------------------------------------------------------------------  
一、插入文本  
┌──┬────────────┐  
│命令│描述│  
├──┼────────────┤  
│i │在当前字符前插入文本│  
├──┼────────────┤  
│I │在行首插入文本 │  
├──┼────────────┤  
│a │在当前字符后添加文本│  
├──┼────────────┤  
│A │在行末添加文本│  
├──┼────────────┤  
│o │在当前行后面插入一空行│  
├──┼────────────┤  
│O │在当前行前面插入一空行│  
├──┼────────────┤  
│R │以改写方式输入文本│  
└──┴────────────┘  
二、移动光标  
┌─────┬───────────┐  
│命令│描述│  
├─────┼───────────┤  
│j或下箭头 │向下移动一行│  
├─────┼───────────┤  
│k或上箭头 │向上移动一行│  
├─────┼───────────┤  
│h或左箭头 │左移一个字符│  
├─────┼───────────┤  
│l或右箭头 │右移一个字符│  
├─────┼───────────┤  
│w │右移一个词│  
├─────┼───────────┤  
│W │右移一个以空格分隔的词│  
├─────┼───────────┤  
│b │左移一个词│  
├─────┼───────────┤  
│B │左移一个以空格分隔的词│  
├─────┼───────────┤  
│0 │移到行首│  
│Ctrl-F│向前翻页│  
├─────┼───────────┤  
│Ctrl-B│向后翻页│  
├─────┼───────────┤  
│nG│到第n行 │  
├─────┼───────────┤  
│G │到最后一行│  
└─────┴───────────┘  
三、替换文本  
┌─────┬──────┐  
│命令│描述│  
├─────┼──────┤  
│$ │到行尾│  
├─────┼──────┤  
│( │到句子的开头│  
├─────┼──────┤  
│) │到句子的末尾│  
├─────┼──────┤  
│{ │到段落的开头│  
├─────┼──────┤  
│} │到段落的末尾│  
└─────┴──────┘  
  
四、删除文本  
┌───┬───────────┐  
│命令│描述 │  
├───┼───────────┤  
│r │替换一个字符 │  
├───┼───────────┤  
│c │修改文本直到按下Esc健 │  
├───┼───────────┤  
│cw│修改下一个词 │  
├───┼───────────┤  
│cnw │修改接下来的n个词 │  
└───┴───────────┘  
五、文本编辑  
┌──┬──────────────────────┐  
│命寺│描述│  
├──┼──────────────────────┤  
│yy│将一行文本移到缺省缓冲区中 │  
├──┼──────────────────────┤  
│yn│将下一个词移到缺省缓冲区中 │  
├──┼──────────────────────┤  
│ynw │将后面的n个词移到缺省缓冲区中 │  
├──┼──────────────────────┤  
│p │如果缺省缓冲区中包含一行文本，则在当前│  
││行后面插入一个空行井将缺省缓冲区中的声│  
││容粘贴到这一行中；如果缺省缓冲区中包含│  
││多个词，把这些词粘贴到光标的右边．│  
├──┼──────────────────────┤  
│P │如果缺省缓冲区中包含一行文本，则正当前 │  
│ │行前面插入一个空行井将缺省缓冲区中的内 │  
││容粘贴到这一行中；如果缺省缓冲区中包含 │  
│ │多个词，把这些词粘贴到光标的左边 │  
└──┴──────────────────────┘  
六、保存退出  
┌───────────┬───────────────┐  
│命令│描述│  
├───────────┼───────────────┤  
│zz│保存并退出│  
├───────────┼───────────────┤  
│:w filename │写入文件 │  
├───────────┼───────────────┤  
│:W│写入文件│  
├───────────┼───────────────┤  
│:x│保存(如果当前文件修改过)并退出│  
├───────────┼───────────────┤  
│:q! │不保存文件，直接退出 │  
├───────────┼───────────────┤  
│:q│退出vi│  
└───────────┴───────────────┘  
VI常用技巧  
  
VI命令可以说是Unix/Linux世界里最常用的编辑文件的命令了，但是因为它的命令集众多，很多人都不习惯使用它，其实您只需要掌握基本命令，然后加以灵活运用，就会发现它的优势，并会逐渐喜欢使用这种方法。本文旨在介绍VI的一些最常用命令和高级应用技巧。  
  
一、基本命令介绍

---- 1．光标命令  
k、j、h、l——上、下、左、右光标移动命令。虽然您可以在Linux中使用键盘右边的4个光标键，但是记住这4个命令还是非常有用的。这4个键正是右手在键盘上放置的基本位置。  
nG——跳转命令。n为行数，该命令立即使光标跳到指定行。  
Ctrl+G——光标所在位置的行数和列数报告。  
w、b——使光标向前或向后跳过一个单词。  
---- 2．编辑命令  
i、a、r——在光标的前、后以及所在处插入字符命令(i=insert、a=append、r=replace)。  
cw、dw——改变(置换)/删除光标所在处的单词的命令 (c=change、d=delete)。  
x、d$、dd——删除一个字符、删除光标所在处到行尾的所有字符以及删除整行的命令。  
---- 3．查找命令  
---- /string、?string——从光标所在处向后或向前查找相应的字符串的命令。  
---- 4．拷贝复制命令  
---- yy、p——拷贝一行到剪贴板或取出剪贴板中内容的命令。  
  
二、常见问题及应用技巧  
  
---- 1．在一个新文件中读/etc/passwd中的内容，取出用户名部分。  
---- vi file  
---- :r /etc/passwd 在打开的文件file中光标所在处读入/etc/passwd  
---- :%s/:.\*//g 删除/etc/passwd中用户名后面的从冒号开始直到行尾的所有部分。  
---- 您也可以在指定的行号后读入文件内容，例如使用命令“:3r /etc/passwd”从新文件的第3行开始读入 /etc/passwd的所有内容。  
---- 我们还可以使用以下方法删掉文件中所有的空行及以#开始的注释行。  
---- #cat squid.conf.default | grep -v ^$ | grep -v ^#  
  
---- 2．在打开一个文件编辑后才知道登录的用户对该文件没有写的权限，不能存盘，需要将所做修改存入临时文件。  
---- vi file  
---- :w /tmp/1 保存所做的所有修改，也可以将其中的某一部分修改保存到临时文件，例如仅仅把第20～59行之间的内容存盘成文件/tmp/1，我们可以键入如下命令。  
---- vi file  
---- :20,59w /tmp/1  
  
---- 3．用VI编辑一个文件，但需要删除大段的内容。  
---- 首先利用编辑命令“vi file”打开文件，然后将光标移到需要删除的行处按Ctrl+G显示行号，再到结尾处再按Ctrl+G，显示文件结尾的行号。  
---- :23,1045d 假定2次得到的行号为23和1045，则把这期间的内容全删除，也可以在要删除的开始行和结束行中用ma、mb命令标记，然后利用“:a,bd”命令删除。  
  
---- 4．在整个文件的各行或某几行的行首或行尾加一些字符串。  
---- vi file  
---- :3,$s/^/some string / 在文件的第一行至最后一行的行首插入“some string”。  
---- :%s/$/some string/g 在整个文件每一行的行尾添加“some string”。  
---- :%s/string1/string2/g 在整个文件中替换“string1”成“string2”。  
---- :3,7s/string1/string2/ 仅替换文件中的第3行到第7行中的“string1”成“string2”。  
---- 注意: 其中s为substitute，%表示所有行，g表示global。  
  
---- 5．同时编辑2个文件，拷贝一个文件中的文本并粘贴到另一个文件中。  
---- vi file1 file2  
---- yy 在文件1的光标处拷贝所在行  
---- :n 切换到文件2 (n=next)  
---- p 在文件2的光标所在处粘贴所拷贝的行  
---- :n 切换回文件1  
  
---- 6．替换文件中的路径。  
---- 使用命令“:%s#/usr/bin#/bin#g”可以把文件中所有路径/usr/bin换成/bin。也可以使用命令“:%s//usr/bin//bin/g”实现，其中“”是转义字符，表明其后的“/”字符是具有实际意义的字符，不是分隔符。