

学习Linux二（创建、删除文件和文件夹命令） - 鸚兒 - 博客园

今天学习了几个命令，是创建、删除文件和文件夹的，在linux里，文件夹是目录，下面说下我学习的命令。

创建文件夹【mkdir】

一、mkdir命令使用权限

所有用户都可以在终端使用 `mkdir` 命令在拥有权限的文件夹创建文件夹或目录。

二、mkdir命令使用格式

格式：`mkdir [选项] DirName`

三、mkdir命令功能

通过 `mkdir` 命令可以实现指定位置创建以 `DirName` (指定的文件名) 命名的文件夹或目录。要创建文件夹或目录的用户必须对所创建的文件夹的父文件夹具有写权限(了解Linux文件-文件夹权限请点击[这里](#))。并且，所创建的文件夹(目录)不能与其父目录(即父文件夹)中的文件名重名，即同一个目录下不能有同名的(区分大小写)。

四、mkdir命令选项说明

命令中的 [选项] 一般有以下两种：

`-m` 用于对新建目录设置存取权限，也可以用 `chmod` 命令进行设置。

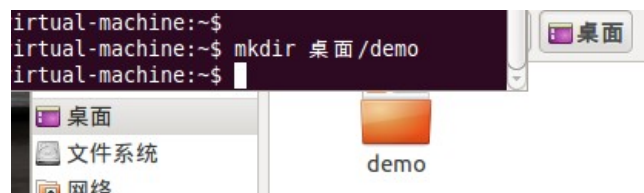
`-p` 需要时创建上层文件夹(或目录)，如果文件夹(或目录)已经存在，则不视为错误。

五、mkdir命令使用举例

例一：在桌面下面创建以“demo”命名的文件夹。

使用以下命令即可。

`mkdir 桌面/demo`



例二：在桌面下面创建以“demo”命名的文件夹，并且给文件夹赋权限，权限为123。

`mkdir 123 桌面/demo`



删除文件夹【rm】

一、rm命令使用权限

所有用户都可以在终端使用 `rm` 命令删除目录。

二、rm命令使用格式

格式：`rm [选项] DirName`

三、rm命令功能

删除档案及目录。

四、rm命令选项说明

命令中的 [选项] 一般有以下几种：

`-i` 删除前逐一询问确认。

`-f` 即使原档案属性设为唯读，亦直接删除，无需逐一确认。

`-r` 将目录及以下之档案亦逐一删除。

五、rm命令使用举例

例一：删除所有C语言程序文档；删除前逐一询问确认。

`rm -i *.c`

例二：将 `Finished` 子目录及子目录中所有档案删除。

`rm -r Finished`

注：在linux没有回收站，在试用rm命令的时候，一定要小心些，删除之后就无法再恢复了。

创建文件【vi】

一、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

二、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

三、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。

四、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

五、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

六、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

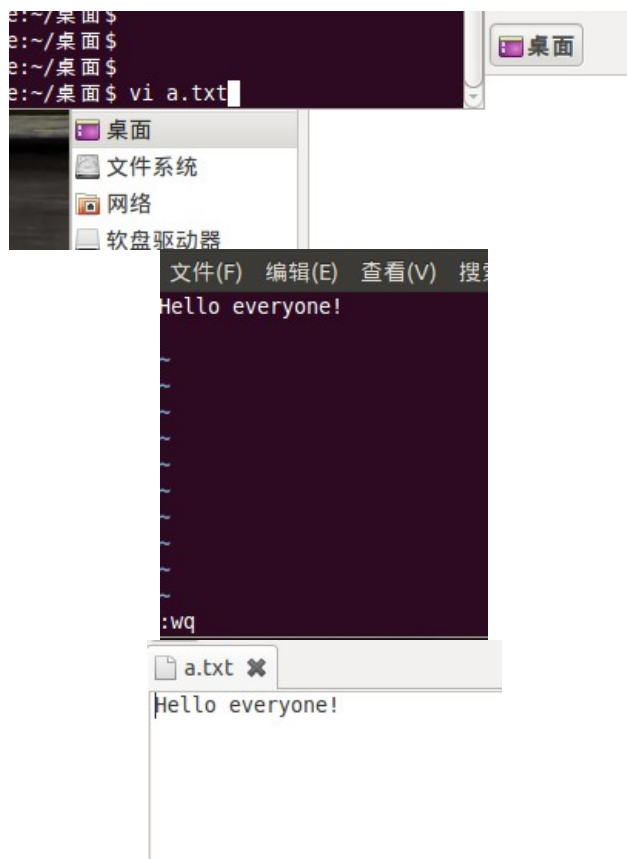
七、vi命令使用举例

例一: 创建文件a.txt。

```
vi a.txt
```

```
Hello everyone!
```

:wq //在退出时, 直接输入:wq会发现退不出去, 退出方法是: 编辑完成后按ESC, 然后输入:q就是退出;还有:wq是保存后退出, 加感叹号是表示强制



修改档案时间【touch】

一、touch命令使用权限

所有用户都可以在终端使用 touch命令。

二、touch命令使用格式

格式: touch [-acfm] [-d <日期时间>] [-r <参考文件或目录>] [-t <日期时间>] [--help] [--version] [文件或目录...] 或 touch [-acfm] [--help] [--version] [日期时间] [文件或目录...]

(touch [-acfm] [-r reference-file] [--file=reference-file] [-t MMDDhhmm[[CC]YY][.ss]] [-d time] [--date=time] [--time={atime,access,use,mtime,modify}] [--no-create] [--help] [--version] file1 [file2 ...])

三、touch命令功能

`touch fileA`，如果fileA存在，使用touch命令可更改这个文件或目录的日期时间，包括存取时间和更改时间；如果fileA不存在，touch命令会在当前目录下新建一个空白文件fileA。

注：使用touch指令可更改文件或目录的日期和时间，包括存取时间和更改时间。文件的时间属性包括文件的最后访问时间，最后修改时间以及最后在磁盘上修改的时间，命令stat显示结果显示了三个时间属性。

四、touch命令选项说明

a 改变档案的读取时间记录。

m 改变档案的修改时间记录。

c 假如目的档案不存在，不会建立新的档案。与 `--no-create` 的效果一样。

f 不使用，是为了与其他 unix 系统的相容性而保留。

r 使用参考档的时间记录，与 `--file` 的效果一样。

d 设定时间与日期，可以使用各种不同的格式。

t 设定档案的时间记录，格式与 `date` 指令相同。`[[CC]YY]MMDDhhmm[.SS]`，CC为年数中的前两位，即“世纪数”；YY为年数的后两位，即某世纪中的年数。如果不给出CC的值，则linux中touch命令参数将把年数CCYY限定在1969--2068之内。MM为月数，DD为天将把年数CCYY限定在1969--2068之内。MM为月数，DD为天数，hh 为小时数(几点)，mm为分钟数，SS为秒数。此处秒的设定范围是0--61，这样可以处理闰秒。这些数字组成的时间是环境变量TZ指定的时区中的一个时间。由于系统的限制，早于1970年1月1日的时间是错误的。

`--no-create` 不会建立新档案。

`--help` 列出指令格式。

`--version` 列出版本讯息。

五、touch命令使用举例

例一：更新file1.txt的存取和修改时间。

```
touch file1.txt
```

例二：如果file1.txt不存在，不创建文件

```
touch -c file1.txt
```

例三：更新file1.txt的时间戳和ref+file相同

```
touch -r ref+file file1.txt
```

例四：设置文件的时间戳为2011年5月18日9点45分10秒

```
ls -l file1.txt
```

```
touch -t 1105190945.10 file1.txt
```

在新建文件时用touch命令可以建立一个空文件，而vi命令则可以直接编辑文件的内容并保存。touch命令在linux中很少用到，我个人还是比较喜欢vi命令，可以直接对文档进行编辑。今天的学习就这些了，每天进步一点点，昨天没写完，今天续写的。

0(∩_∩)0~