

1,du

统计目录（或文件）所占磁盘空间的大小。

用法: `du -sh path` 或 `du -sm path`

```
w_yxl@drone-004:~/608_lbs/trunk/apps/ios$ du -sh ABT_HD1.0/
103M    ABT_HD1.0/
w_yxl@drone-004:~/608_lbs/trunk/apps/ios$ du -sm ABT_HD1.0/
103     ABT_HD1.0/
```

2,ll

将当前目录的文件升序或降序列举出来。

以列表的方式显示文件名、链接数目、所有者、组、大小（按字节）和每个文件最近一次修改时间。

用法: `ll -t` 是升序, `ll -t | tac` 就是降序

```
w_yxl@drone-004:~/608_lbs/trunk/apps/ios$ ll
total 16
drwxr-xr-x  4 w_yxl w_yxl 4096 2013-02-05 10:27 ./
drwxr-xr-x 16 w_yxl w_yxl 4096 2013-02-05 10:27 ../
drwxr-xr-x 12 w_yxl w_yxl 4096 2013-02-27 15:32 ABT_HD1.0/
drwxr-xr-x  6 w_yxl w_yxl 4096 2013-02-27 15:32 .svn/
w_yxl@drone-004:~/608_lbs/trunk/apps/ios$
```

3,ls

`-ls` 列出当前文件夹下所有内容

`$ls -o` 列出当前文件夹中所有内容，含详细信息，但不列出 group

`$ls -l` 同上，含 group 信息

```

yangxiaolong@ubuntu01:/usr$ ls -o
总用量 60
drwxr-xr-x  2 root 20480 2013-03-05 15:22 bin
drwxr-xr-x  2 root  4096 2011-01-21 04:36 games
drwxr-xr-x  4 root  4096 2013-03-05 15:21 include
drwxr-xr-x 47 root 12288 2013-03-05 15:22 lib
drwxr-xr-x 10 root  4096 2013-03-05 15:11 local
drwxr-xr-x  2 root  4096 2013-03-05 15:22 sbin
drwxr-xr-x 99 root  4096 2013-03-05 15:22 share
drwxrwsr-x  4 root  4096 2013-03-05 15:13 src
drwxr-xr-x  2 root  4096 2013-03-06 11:08 test
yangxiaolong@ubuntu01:/usr$ ls -l
总用量 60
drwxr-xr-x  2 root root 20480 2013-03-05 15:22 bin
drwxr-xr-x  2 root root  4096 2011-01-21 04:36 games
drwxr-xr-x  4 root root  4096 2013-03-05 15:21 include
drwxr-xr-x 47 root root 12288 2013-03-05 15:22 lib
drwxr-xr-x 10 root root  4096 2013-03-05 15:11 local
drwxr-xr-x  2 root root  4096 2013-03-05 15:22 sbin
drwxr-xr-x 99 root root  4096 2013-03-05 15:22 share
drwxrwsr-x  4 root src  4096 2013-03-05 15:13 src
drwxr-xr-x  2 root root  4096 2013-03-06 11:08 test

```

\$ls -a 列出当前文件夹中所有内容，包含以“.”开头的文件

\$ls -t 按更改时间排序

\$ls -v 按版本先后排序

```

aolong@ubuntu01:/usr$ ls -a -a 列出所有文件，保护 . 的文件
.  bin  caca.txt  games  include  lib  local  sbin  share  src  test
aolong@ubuntu01:/usr$ ls -t -t 列出时按照修改时间排序，caca.txt是我刚刚修改的文件
caca.txt  test  sbin  bin  lib  share  include  src  local  games
aolong@ubuntu01:/usr$ ls -v
caca.txt  games  include  lib  local  sbin  share  src  test

```

4、mkdir

在当前目录下，创建一个文件夹（目录）

`mkdir dir1 dir2` 同时创建 `dir1` 和 `dir2` 两个目录

`mkdir -p dir1/dir2` 递归生成目录 `dir1` 和 `dir2`，`dir2` 是 `dir1` 的子目录

```
yangxiaolong@ubuntu01:~$ ls
caca  changeMod  redirection  wawa
yangxiaolong@ubuntu01:~$ mkdir dir1/dir2
mkdir: 无法创建目录"dir1/dir2": 没有那个文件或目录
yangxiaolong@ubuntu01:~$ mkdir -p dir1/dir2
yangxiaolong@ubuntu01:~$ ls
caca  changeMod  dir1  redirection  wawa
                                -p 递归生成目录/子目录
```

`rmdir` 删除目录命令

用于删除空目录，格式：`rmdir [选项] [目录名]`

使用这个只能删除目录，不能删除文件，并且目录必须为空，否则使用 `rm -r` 目录 命令

参数：`-p` 递归删除空目录

```
yangxiaolong@ubuntu01:~$ ls
caca  changeMod  dir1  redirection  wawa
yangxiaolong@ubuntu01:~$ rmdir dir1/
rmdir: 删除 "dir1/" 失败: 目录非空
yangxiaolong@ubuntu01:~$ rmdir -p dir1/
rmdir: 删除 "dir1/" 失败: 目录非空 递归删除失败
```

5、touch

在当前目录下，创建一个文件，此命令无法创建文件夹。

6、chmod

更改文件/文件夹的权限。

`$chmod [Mode] [dir]`，其中 `Mode` 形如“755”或“777”等。

`$chmod [Mode] [file]`

`$chmod -R [Mode] [dir]`，递归形式，即将目标文件夹内所有文件均改变权限（跟 `rm -r` 递归删除一个道理）

注意，递归时，参数 `R` 必须为大写：`chmod -R 755 iOS`

`Mode` 还有另一种表达方式，“755”即为“-rwxr-xr-x”。

```

yangxiaolong@ubuntu01:~/changeMod$ ls -l
总用量 4
-rw-r--r-- 1 yangxiaolong yangxiaolong 82 2013-03-06 14:37 mod.c
yangxiaolong@ubuntu01:~/changeMod$ chmod 777 mod.c 改变权限
yangxiaolong@ubuntu01:~/changeMod$ ls -l
总用量 4
-rwxrwxrwx 1 yangxiaolong yangxiaolong 82 2013-03-06 14:37 mod.c

```

7、rm

`-rm [file]` 删除文件/文件夹

`$rm -f [file]` 强行删除，忽略不存在的文件，无提示

`$rm -r [file]` 递归删除所有内容，一般用于删除某一目录（文件夹）

强制递归删除一个文件夹内所有内容时：`rm -rf ios`

```

yangxiaolong@ubuntu01:~$ rm -f dir1
rm: 无法删除"dir1": 是一个目录
yangxiaolong@ubuntu01:~$ rm -r dir1
yangxiaolong@ubuntu01:~$ ls
caca  changeMod  dir2  redirection  wawa

```

8、cp

拷贝

`$cp [options] [source] [destination]` （中间都有空格）

其中[options]可以为-f（强行拷贝）或-r（递归拷贝）

```

yangxiaolong@ubuntu01://home/yangxiaolong/changeMod$ ls
aa  bb  cc  mod.c
yangxiaolong@ubuntu01://home/yangxiaolong/changeMod$ cp -r bb cc/backu
把文件夹bb拷贝到cc/backup/bbback

```

9、mv

重命名或移动

`$mv [options] [source] [destination]`

[options]常用: -f(强行移动/重命名), -i(移动/重命名前尝试), -u(更新)

例如

`$mv wwwroot/cgi-bin .` 将/cgi-bin 目录移动到当前目录下

```
yangxiaolong@ubuntu01://home/yangxiaolong/changeMod/cc$ ls
backup ddd reca.c 移动前，目标文件夹的内容
yangxiaolong@ubuntu01://home/yangxiaolong/changeMod/cc$ cd ../bb/
yangxiaolong@ubuntu01://home/yangxiaolong/changeMod/bb$ ls
bba.c bbbbbb bbb.c bbc.c bbd.c ff kk new 要移动的文件夹
yangxiaolong@ubuntu01://home/yangxiaolong/changeMod/bb$ mv new/ ../cc/
yangxiaolong@ubuntu01://home/yangxiaolong/changeMod/bb$ ls
bba.c bbbbbb bbb.c bbc.c bbd.c ff kk 移动成功，当前文件夹没有了
yangxiaolong@ubuntu01://home/yangxiaolong/changeMod/bb$ cd ../cc/
yangxiaolong@ubuntu01://home/yangxiaolong/changeMod/cc$ ls
backup ddd new reca.c 移动后的目标文件夹内容
```

`$mv cronfile.txt myfile.txt` 将 cronfile.txt 重命名为 myfile.txt

```
yangxiaolong@ubuntu01://home/yangxiaolong/changeMod/cc$ ls
backup ca.c ddd
yangxiaolong@ubuntu01://home/yangxiaolong/changeMod/cc$ mv ca.c reca.c
yangxiaolong@ubuntu01://home/yangxiaolong/changeMod/cc$ ls
backup ddd reca.c 重命名
```

10、find

查找文件

`find $HOME -mtime 0` 在 home 目录中查找所有 24 小时内修改过的文件。如果不加\$HOME，则是查找当前目录下时间段之内修改过的文件。

`find [filepath] -perm 775` 在当前目录或 path 目录下，查找所有权限为 775 的文件。

11、sudo

`sudo` 可以让你拥有执行 `root` 权限的能力，在需要执行的操作前加上 `sudo` 就好，只是在执行此命令时拥有了 `root` 权限，而非永久，如果想一直拥有 `root` 权限，相当于切换 `root` 用户，则是：`sudo su` 命令，会提示输入 `password`，正确后切换到 `root` 用户。（管理员为`#`，普通用户`$`）

12、cat

`cat` 是查看文件信息的命令，

1, `cat` [选项] 文件名 `-n` 是输出进行编号显示。

2, `cat > 新文件名` 利用输出重定向把 `cat` 命令的屏幕输出信息写入一个新的文件中。

结束编辑是 `ctrl+d`

3, `cat 文件 1 文件 2 >文件 3(新文件名)` 合并文件。

如：

13、more

逐屏显示文件内容

`more` 命令是在每个满屏之后，在屏幕底部提示以显示内容占全部内容的百分比。

回车：显示下一行 空格：显示下一屏 `q` 键：退出。

```
yangxiaolong@ubuntu01:~/caca$ ls
a.c
yangxiaolong@ubuntu01:~/caca$ man ls > lshelp
yangxiaolong@ubuntu01:~/caca$ ls
a.c lshelp
yangxiaolong@ubuntu01:~/caca$ more lshelp

-d, --directory
    list directory entries instead of

-D, --dired

--More-- (18%) 回车 下一行 空格 下一屏 q退出
```

14、grep

在指定文件中检索出匹配关键字的信息内容，它可以方便地搜索文件，能够在不打开文件而搜索文件中的文本字符串。

参数：`-i` 忽略字符大小写 `-n` 在显示符合的字符串之前，标出该行的行号。

如：

```
yangxiaolong@ubuntu01:/etc$ grep -n w passwd -参数 关键字 文件
10:news:x:9:9:news:/var/spool/news:/bin/sh
13:www-data:x:33:33:www-data:/var/www:/bin/sh
yangxiaolong@ubuntu01:/etc$ cat passwd|grep -n w
10:news:x:9:9:news:/var/spool/news:/bin/sh 通过管道符号
13:www-data:x:33:33:www-data:/var/www:/bin/sh
```

15、sort

对文件进行排序与合并，把所有指定文件的行一起进行排序。比较的依据是从文件的每一行提取一个或多个排序关键字进行的。

sort [选项] [文件列表]

sort -n file 对 file 文件按每行第一个字段进行数值排序

```
yangxiaolong@ubuntu01:/boot$ ls -l|sort -n 每行第一个字段数值排序
drwxr-xr-x 2 root root 12288 2013-03-05 15:11 lost+found
drwxr-xr-x 3 root root 4096 2013-03-05 15:23 grub
-rw-r--r-- 1 root root 116492 2011-01-11 09:36 config-2.6.32-28-gener
-rw-r--r-- 1 root root 1200 2011-01-11 09:38 vmcoreinfo-2.6.32-28-g
```

sort -r file 对 file 文件按每行第一个字符进行反向排序

```
yangxiaolong@ubuntu01:/boot$ ls -l|sort -r 每行第一个字段反向排序
总用量 15264
-rw-r--r-- 1 root root 8704520 2013-03-05 15:22 initrd.img-2.6.32-28-g
-rw-r--r-- 1 root root 655316 2011-01-11 09:36 abi-2.6.32-28-generic-
-rw-r--r-- 1 root root 4172992 2011-01-11 09:36 vmlinuz-2.6.32-28-gene
```

sort -k 5 file 对 file 文件按每行第五个字段为关键字进行排序

例如下图是。把当前目录下的所有文件按照文件大小，由大到小排序。

```
yangxiaolong@ubuntu01:/boot$ ls -l|sort -n -r -k 5
-rw-r--r-- 1 root root 8704520 2013-03-05 15:22 initrd.img-2.6.32-28-g
-rw-r--r-- 1 root root 4172992 2011-01-11 09:36 vmlinuz-2.6.32-28-gene
-rw-r--r-- 1 root root 1732138 2011-01-11 09:36 System.map-2.6.32-28-g
-rw-r--r-- 1 root root 655316 2011-01-11 09:36 abi-2.6.32-28-generic-
-rw-r--r-- 1 root root 160280 2010-03-23 17:37 memtest86+.bin
-rw-r--r-- 1 root root 116492 2011-01-11 09:36 config-2.6.32-28-gener
```

总之，记住三个参数：-n 按照数值 -r 反方向 -k 指定数字 行下的某一字段

16、chown

每个文件或目录都有它的所有者，即属主，同时属于某个特定的组，该组称为文件的属组。可以同个 **chown** 命令改变文件的属主和组，前提是拥有相应的权限。

格式： **chown** [选项] [所有者][: 组] 文件名称（目录）

【注意：】**chown** 是修改文件或目录的属主或组（root&user），**chmod** 是修改文件的权限（RWX）

```
root@ubuntu01:/home/yangxiaolong/redirection# ls -l k*
-rw-r--r-- 1 yangxiaolong yangxiaolong 103 2013-03-08 13:30 k1
-rw-r--r-- 1 yangxiaolong yangxiaolong  43 2013-03-08 13:30 k2
-rw-r--r-- 1 yangxiaolong yangxiaolong 146 2013-03-08 13:30 k3
root@ubuntu01:/home/yangxiaolong/redirection# chown root k1
root@ubuntu01:/home/yangxiaolong/redirection# ls -l k* 更改文件所有者
-rw-r--r-- 1 root yangxiaolong 103 2013-03-08 13:30 k1
-rw-r--r-- 1 yangxiaolong yangxiaolong  43 2013-03-08 13:30 k2
-rw-r--r-- 1 yangxiaolong yangxiaolong 146 2013-03-08 13:30 k3
root@ubuntu01:/home/yangxiaolong/redirection# chown :root k2
root@ubuntu01:/home/yangxiaolong/redirection# ls -l k2 更改文件属组
-rw-r--r-- 1 yangxiaolong root 43 2013-03-08 13:30 k2
root@ubuntu01:/home/yangxiaolong/redirection# chown root:root k3
root@ubuntu01:/home/yangxiaolong/redirection# ls -l k3 更改文件属主及属组
-rw-r--r-- 1 root root 146 2013-03-08 13:30 k3
```