



链滴

Linux - 内存和磁盘管理

作者: [douniwan](#)

原文链接: <https://ld246.com/article/1573916622995>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

文件系统

什么是文件系统？详情参考[Linux 文件系统\(一\)---虚拟文件系统VFS---超级块、inode、dentry、file](#)

ext4文件系统的基本结构

- 超级块
- 超级块副本
- I节点 (inode) `ls -li` (查看i节点)
- 数据块 (datablock) `du -h`

超级块

请参考：[超级块的定义](#)

超级快副本

顾名思义，就是超级块的副本。

Inode

理解inode，要从文件储存说起。

文件储存在硬盘上，硬盘的最小存储单位叫做"扇区" (Sector)。每个扇区储存512字节（相当于0.5K）。

操作系统读取硬盘的时候，不会一个个扇区地读取，这样效率太低，而是一次性连续读取多个扇区，一次性读取一个"块" (block)。这种由多个扇区组成的"块"，是文件存取的最小单位。"块"的大小，常见的是4KB，即连续八个 sector组成一个 block。

文件数据都储存在"块"中，那么很显然，我们还必须找到一个地方储存文件的元信息，比如文件的创建者、文件的创建日期、文件的大小等等。这种储存文件元信息的区域就叫做inode，中文译名为"索引点"。

每一个文件都有对应的inode，里面包含了与该文件有关的一些信息。

感觉这里的inode就跟c语言中的指针的功能差不多。

Datablock

datablock，文件的内容都是存放在block中，1个block通常是4K，1个文件（比较大的）可能会占用个block，但如果文件很少（比如0.1K），也会占用1个block，并且剩余空间不能使用。

Inode和数据块操作

首先通过`ls -li`查看文件的信息：

```
huny@huny-PC:~$ ls -li
```

总用量 248

```
1717797 -rw----- 1 huny huny 186 10月 29 08:06 2019-10-29-00-06-53.069-VBoxSVC-12
57.log
1704766 lrwxrwxrwx 1 huny huny 5 11月 14 21:27 afile -> afile
1708491 -rw-r--r-- 2 huny huny 4 11月 14 21:16 afile
1712123 -rw-r--r-- 1 huny huny 4 11月 14 21:17 afile3
1707817 -rw-r--r-- 1 huny huny 8 11月 14 21:20 afile4
2228295 drwxr-xr-x 3 huny huny 4096 10月 19 14:44 Android
2100345 drwxr-xr-x 4 huny huny 4096 10月 29 08:37 AndroidStudioProjects
1708491 -rw-r--r-- 2 huny huny 4 11月 14 21:16 bfile
3023319 drwxrwxrwx 8 huny huny 4096 11月 12 22:22 blog
1703986 drwxr-xr-x 5 huny huny 4096 11月 16 21:05 Desktop
1703960 drwxr-xr-x 7 huny huny 4096 11月 10 18:12 Documents
1703961 drwxr-xr-x 2 huny huny 4096 11月 16 14:58 Downloads
1836292 drwxr-xr-x 10 huny huny 4096 11月 5 22:11 IdeaProjects
1704989 -rw-r--r-- 1 huny huny 167142 11月 12 09:39 java_error_in_STUDIO_18074.log
1703965 drwxr-xr-x 3 huny huny 4096 10月 20 21:21 Music
1714651 -rw----- 1 huny huny 162 11月 14 20:25 nohup.out
1703938 drwxr-xr-x 3 huny huny 4096 10月 20 21:21 Pictures
1707443 lrwxrwxrwx 1 huny huny 36 10月 20 21:34 PlayOnLinux's virtual drives -> /home/
huny/.PlayOnLinux//wineprefix/
1717825 -rw-r--r-- 1 huny huny 0 10月 27 20:23 stale_outputs_checked
3541458 drwxr-xr-x 3 huny huny 4096 10月 28 23:58 Steam
1703982 drwxr-xr-x 2 huny huny 4096 10月 20 21:21 Videos
2230208 drwxr-xr-x 7 huny huny 4096 11月 13 13:43 个人资料
3023275 drwxr-xr-x 2 huny huny 4096 11月 16 16:57 工具
1703953 drwxr-xr-x 2 huny huny 4096 10月 19 01:44 模板
2887058 drwxrwxrwx 2 huny huny 4096 10月 20 22:32 书
huny@huny-PC:~$
```

在每一行输出的最前面的数字，及文件的inode标识，每个不同的文件的inode都是不同的。然后文
权限后面的数字，标识这个文件的硬链接数（inode对应了多少文件名）

这里跟Inode有关的一些知识：

- mv改名不会改变i节点，改的是i节点和文件名的对应关系。
- vim会改变文件的i节点。通过swap文件来操作（为了保证一致性）
- rm实际是i节点和文件名断开。（所以删除文件才会那么快）
- ln让更多的文件名连接到这个文件（硬链接）
- （文件名只存在该父目录下），不能跨越分区使用
- ln -s（符号链接，也叫软链接。相当于windows的快捷方式）（可以跨文件系统使用）

查看系统内存信息

前面[Linux - 进程管理（一）](#)中学过top命令中的输出第四行是关于内存的信息，那么除了top命令方
，还有什么方式可以查看内存信息呢？这时候就可以使用free命令来查看了。

free

用法：

`free [-bkmotV][-s <间隔秒数>]`

参数说明：

- -b 以Byte为单位显示内存使用情况。
- -k 以KB为单位显示内存使用情况。
- -m 以MB为单位显示内存使用情况。
- -h 以合适的单位显示内存使用情况，最大为三位数，自动计算对应的单位值。单位有：

B = bytes
K = kilos
M = megas
G = gigas
T = teras

- -o 不显示缓冲区调节列。
- -s<间隔秒数> 持续观察内存使用状况。
- -t 显示内存总和列。
- -V 显示版本信息。

示例：

```
huny@huny-PC:~$ free
             total        used        free      shared  buff/cache   available
Mem:      16292804    2007700    10698380     305468     3586724    13664372
Swap:      8388604         0      8388604
huny@huny-PC:~$
```

输出解释：

Mem 行(第二行)是内存的使用情况

Swap 行(第三行)是交换空间的使用情况

total 列显示系统总的可用物理内存和交换空间大小

used 列显示已经被使用的物理内存和交换空间

free 列显示还有多少物理内存和交换空间可用使用

hared 列显示被共享使用的物理内存大小

buff/cache 列显示被 buffer 和 cache 使用的物理内存大小

available 列显示还可以被应用程序使用的物理内存大小。

查看磁盘信息

在Linux中如果要查看磁盘信息，可以使用**fdisk**，**df**，**du**命令来查看。

fdisk

关于fdisk的使用详情参考：[linux fdisk 分区、格式化、挂载！](#)

这里我主要是查看各个分区的容量情况。使用`fdisk -l`

```
huny@huny-PC:~$ sudo fdisk -l
[sudo] huny 的密码:
Disk /dev/nvme0n1: 238.5 GiB, 256060514304 bytes, 500118192 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: 02129907-7AE0-4914-8753-E1CA650EBC79

Device            Start      End  Sectors  Size Type
/dev/nvme0n1p1    2048 16779263 16777216   8G Linux swap
/dev/nvme0n1p2 16779264 226496511 209717248 100G Linux filesystem
/dev/nvme0n1p3 226496512 227522559 1026048 501M EFI System
/dev/nvme0n1p4 227522560 500117503 272594944 130G Linux filesystem
```

```
Disk /dev/sda: 477 GiB, 512110190592 bytes, 1000215216 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: 31D4B0FB-2F6B-4161-B3DA-D6A68A38B4D9
```

```
Device      Start      End  Sectors  Size Type
/dev/sda1   2048 1085439 1083392 529M Windows recovery environment
/dev/sda2 1085440 1290239 204800 100M EFI System
/dev/sda3 1290240 1323007 32768 16M Microsoft reserved
/dev/sda4 1323008 1000214527 998891520 476.3G Microsoft basic data
huny@huny-PC:~$
```

从这个输出可以很明显的看出各个分区的状况。

df

用于显示 Linux系统中各文件系统的硬盘使用情况，包括文件系统所在硬盘分区的总容量、已使用的量、剩余容量等。

用法：

`df [选项] [目录或文件名]`

常用参数：

- -a: 显示所有文件系统信息，包括系统特有的 /proc、/sysfs 等文件系统；
- -m: 以 MB 为单位显示容量；
- -k: 以 KB 为单位显示容量，默认以 KB 为单位；
- -h: 使用人们习惯的 KB、MB 或 GB 等单位自行显示容量；

- -T: 显示该分区的文件系统名称;
- -i: 不用硬盘容量显示, 而是以含有 inode 的数量来显示。

不使用任何参数的 df 命令, 默认会将系统内所有的文件系统信息, 以 KB 为单位显示出来。

示例如下:

```
huny@huny-PC:~$ df
文件系统      1K-块   已用   可用 已用% 挂载点
udev          8117948    0 8117948   0% /dev
tmpfs         1629284   1844 1627440   1% /run
/dev/nvme0n1p2 102684472 14399960 83025252  15% /
tmpfs         8146400 228536 7917864   3% /dev/shm
tmpfs         5120     4    5116   1% /run/lock
tmpfs         8146400    0 8146400   0% /sys/fs/cgroup
/dev/nvme0n1p3 512000   6684 505316   2% /boot/efi
/dev/nvme0n1p4 133109576 28142744 98135576  23% /home
tmpfs         1629280   36 1629244   1% /run/user/1000
/dev/sda4     499445756 71739576 427706180  15% /media/huny/881E9BCB1E9BB124
huny@huny-PC:~$
```

输出结果是很显然的, 可以清楚的看到各个文件系统的使用信息。

du

用于显示目录或文件的大小。du会显示指定的目录或文件所占用的磁盘空间。

用法及参数参考: [Linux du命令](#)

常用:

```
du -h 文件/目录 #输出文件系统分区使用的情况, 例如: 10KB, 10MB, 10GB等
du -s 文件/目录 #显示文件或整个目录的大小, 默认单位是KB
du -sh 文件/目录 #查看某一个文件的大小
du -sh * | sort -rh #查看目录下所有文件的大小并按照大小排序
```

df与du的区别

****du(disk usage)****是通过搜索文件来计算每个文件的大小然后累加, du能看到的文件只是一些当前在的, 没有被删除的。他计算的大小就是当前他认为存在的所有文件大小的累加和。

****df(disk free)****通过文件系统来快速获取空间大小的信息, 当我们删除一个文件的时候, 这个文件是马上就在文件系统当中消失了, 而是暂时消失了, 当所有程序都不用时, 才会根据OS的规则释放已经删除的文件, df记录的是通过文件系统获取到的文件的大小, 他比du强的地方就是能够看到已删除的文件, 而且计算大小的时候, 把这一部分的空间也加上了, 更精确了。当文件系统也确定删除该文件后, 这时候du与df就一致了。

df和du统计的数据是不同的: 打个比方, 文件是需要放到文件柜里的, 就算只有一个文件, 也要占一个文件柜。文件柜占用的空间比文件要大。df就是统计使用了多少个文件柜。du则统计实际有多少个文件。这样下来, df算的就大, du就小。

简单地说, df命令是根据该卷的inode使用情况进行统计的, 而du则是累加所有文件的字节数。一个文件就算只有1字节, 也要占用一个inode。

参考文章:

[Linux 文件系统\(一\)---虚拟文件系统VFS----超级块、inode、dentry、file](#)

[Linux free 命令学习](#)