

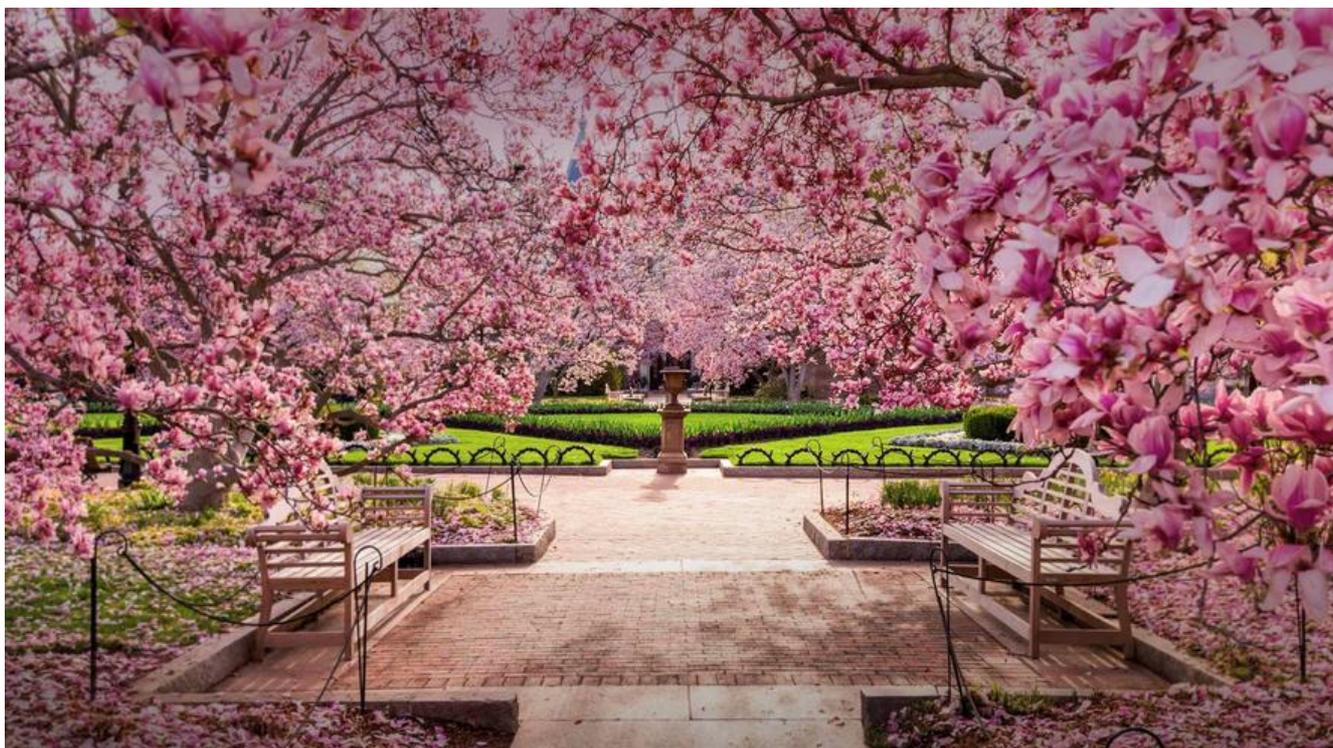
Linux stat 命令：显示文件或文件系统的详细信息

作者：[Leif160519](#)

原文链接：<https://ld246.com/article/1587462250596>

来源网站：[链滴](#)

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



在 Linux 中，文件有访问时间、数据修改时间、状态修改时间这三个时间，而没有创建时间。stat 是看文件详细信息的命令，而且可以看到文件的这三个时间，其基本信息如下。

- 命令名称：stat。
- 英文原意：display file or file system status。
- 所在路径：/usr/bin/stat。
- 执行权限：所有用户。
- 功能描述：显示文件或文件的详细信息。

命令格式

`$ stat [选项]文件名或目录名`

选项：

- -f: 查看文件所在的文件系统信息，而不是查看文件的信息。

常见用法

【例 1】查看文件的详细信息。

```
$ stat anaconda-ks.cfg
文件: "anaconda-ks.cfg"
大小: 1345      块: 8      IO 块: 4096 普通文件
设备: fd00h/64768d Inode: 100663363 硬链接: 1
权限: (0777/-rwxrwxrwx) Uid: ( 0/ root) Gid: ( 0/ root)
最近访问: 2020-04-17 09:26:37.883156715 +0800
最近更改: 2018-12-20 18:53:15.861514535 +0800
```

最近改动: 2019-09-10 16:00:36.809506647 +0800
创建时间: -

【例 2】查看文件系统信息。

如果使用"-f"选项, 就不再是查看指定文件的信息, 而是查看这个文件所在文件系统的信息, 例如:

```
$ stat -f anaconda-ks.cfg
文件: "anaconda-ks.cfg"
ID: fd0000000000 文件名长度: 255 类型: xfs
块大小: 4096 基本块大小: 4096
块: 总计: 13100800 空闲: 10144718 可用: 10144718
Inodes: 总计: 26214400 空闲: 26009834
```

【例 3】三种时间的含义。

查看系统当前时间, 如下:

2020年 04月 21日 星期二 17:33:55 CST

再查看 bds 文件的三种时间, 可以看到, 和当前时间是有差别的, 如下:

```
$ stat test.sh
文件: "test.sh"
大小: 153 块: 8 IO 块: 4096 普通文件
设备: fd00h/64768d Inode: 100663421 硬链接: 1
权限: (0644/-rw-r--r--) Uid: ( 0/ root) Gid: ( 0/ root)
最近访问: 2020-04-20 11:52:15.180435780 +0800
最近更改: 2020-04-20 11:52:13.617422056 +0800
最近改动: 2020-04-20 11:52:13.617422056 +0800
创建时间: -
#这些时间都是最后一次修改文件的时间, 并不是当前时间
```

而如果用 cat 命令读取一下这个文件, 就会发现文件的访问时间 (Access Time) 变成了 cat 命令的执行时间, 如下:

```
$ cat test.sh
$ stat test.sh
文件: "test.sh"
大小: 153 块: 8 IO 块: 4096 普通文件
设备: fd00h/64768d Inode: 100663421 硬链接: 1
权限: (0644/-rw-r--r--) Uid: ( 0/ root) Gid: ( 0/ root)
最近访问: 2020-04-21 17:39:24.381776419 +0800
#只有访问时间变为了cat命令的执行时间, 其他两个时间没有改
最近更改: 2020-04-20 11:52:13.617422056 +0800
最近改动: 2020-04-20 11:52:13.617422056 +0800
创建时间: -
```

而如果用 echo 命令向文件中写入点数据, 那么文件的数据修改时间 (Modify Time) 就会发生改变 但是文件数据改变了, 系统会认为文件的状态也会改变, 所以状态修改时间 (Change Time) 也会随 改变, 如下:

```
$ echo 111 > test.sh
$ stat test.sh
文件: "test.sh"
大小: 4 块: 8 IO 块: 4096 普通文件
```

```
设备: fd00h/64768d Inode: 100663421 硬链接: 1
权限: (0644/-rw-r--r--) Uid: ( 0/ root) Gid: ( 0/ root)
最近访问: 2020-04-21 17:39:24.381776419 +0800
#这个时间还是上次修改时间
最近更改: 2020-04-21 17:40:49.433494860 +0800
最近改动: 2020-04-21 17:40:49.433494860 +0800
#而这两个时间变为了 echo 命令的执行时间
创建时间: -
```

而如果只修改文件的状态（比如改变文件的所有者），而不修改文件的数据，则只会更改状态修改时（ChangeTime），如下：

```
$ chown nobody test.sh
$ stat test.sh
 文件: "test.sh"
 大小: 4      块: 8      IO 块: 4096 普通文件
设备: fd00h/64768d Inode: 100663421 硬链接: 1
权限: (0644/-rw-r--r--) Uid: ( 99/ nobody) Gid: ( 0/ root)
最近访问: 2020-04-21 17:39:24.381776419 +0800
最近更改: 2020-04-21 17:40:49.433494860 +0800
# 前两个时间还是之前修改的时间
最近改动: 2020-04-21 17:41:58.220075909 +0800
#而状态修改时间变为了chown命令的执行时间
创建时间: -
```

而如果用 touch 命令再次触摸这个文件，则这个文件的三个时间 都会改变。touch 命令的作用就是样的，大家记住即可。如下：

```
$ touch test.sh
$ stat test.sh
 文件: "test.sh"
 大小: 4      块: 8      IO 块: 4096 普通文件
设备: fd00h/64768d Inode: 100663421 硬链接: 1
权限: (0644/-rw-r--r--) Uid: ( 99/ nobody) Gid: ( 0/ root)
最近访问: 2020-04-21 17:42:54.314549744 +0800
最近更改: 2020-04-21 17:42:54.314549744 +0800
最近改动: 2020-04-21 17:42:54.314549744 +0800
#三个时间都会变为touch命令的执行时间
创建时间: -
```

参考：

- [Linux stat命令：显示文件或文件系统的详细信息](#)