

[首页](#) [HTML](#) [CSS](#) [JAVASCRIPT](#) [JQUERY](#) [BOOTSTRAP](#) [SQL](#) [MYSQL](#) [PHP](#) [PYTHON2](#) [PYTHON3](#) [C](#) [C++](#) [C#](#) [JAVA](#) [本地书签](#)

Linux 教程



Linux 教程

Linux 简介

Linux 安装

Linux 系统启动过程

Linux 系统目录结构

Linux 忘记密码解决方法

Linux 远程登录

Linux 文件基本属性

Linux 文件与目录管理

Linux 用户和用户组管理

Linux 磁盘管理

Linux vi/vim

linux yum 命令

Shell 教程

Shell 教程

Shell 变量

Shell 传递参数

[← Shell 文件包含](#)[Nginx 安装配置 →](#)

Linux sed命令

[Linux 命令大全](#)

Linux sed命令是利用script来处理文本文件。

sed可依照script的指令，来处理、编辑文本文件。

Sed主要用来自动编辑一个或多个文件；简化对文件的反复操作；编写转换程序等。

语法

```
sed [-hnV][-e<script>][-f<script文件>][文本文件]
```

参数说明：

- e**<script>或**--expression**=<script> 以选项中指定的script来处理输入的文本文件。
- f**<script文件>或**--file**=<script文件> 以选项中指定的script文件来处理输入的文本文件。
- h**或**--help** 显示帮助。
- n**或**--quiet**或**--silent** 仅显示script处理后的结果。
- V**或**--version** 显示版本信息。

动作说明：

a：新增，a 的后面可以接字符串，而这些字符串会在新的一行出现(目前的下一行)~

分类导航

[HTML / CSS](#)[JavaScript](#)[服务端](#)[数据库](#)[移动端](#)[XML 教程](#)[ASP.NET](#)[Web Service](#)[开发工具](#)[网站建设](#)[Advertisement](#)

[Shell 数组](#)[Shell 运算符](#)[Shell echo命令](#)[Shell printf命令](#)[Shell test 命令](#)[Shell 流程控制](#)[Shell 函数](#)[Shell 输入/输出重定向](#)[Shell 文件包含](#)

Linux 参考手册

[Linux 命令大全](#)[Nginx 安装配置](#)[MySQL 安装配置](#)

c：取代，c 的后面可以接字符串，这些字符串可以取代 n1,n2 之间的行！

d：删除，因为是删除啊，所以 d 后面通常不接任何咚咚；

i：插入，i 的后面可以接字符串，而这些字符串会在新的一行出现(目前的上一行)；

p：打印，亦即将某个选择的数据印出。通常 p 会与参数 sed -n 一起运行~

s：取代，可以直接进行取代的工作哩！通常这个 s 的动作可以搭配正规表示法！例如 1,20s/old/new/g 就是啦！

实例

在testfile文件的第四行后添加一行，并将结果输出到标准输出，在命令行提示符下输入如下命令：

```
sed -e 4a\newline testfile
```

首先查看testfile中的内容如下：

```
$ cat testfile #查看testfile 中的内容
HELLO LINUX!
Linux is a free unix-type operating system.
This is a linux testfile!
Linux test
```

使用sed命令后，输出结果如下：

```
$ sed -e 4a\newline testfile #使用sed 在第四行后添加新字符串
HELLO LINUX! #testfile文件原有的内容
Linux is a free unix-type operating system.
This is a linux testfile!
Linux test
newline
```

以行为单位的新增/删除

将 /etc/passwd 的内容列出并且列印行号，同时，请将第 2~5 行删除！

```
[root@www ~]# nl /etc/passwd | sed '2,5d'
1 root:x:0:0:root:/root:/bin/bash
6 sync:x:5:0:sync:/sbin:/bin/sync
7 shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
.....(后面省略).....
```

sed 的动作为 '2,5d'，那个 d 就是删除！因为 2-5 行给他删除了，所以显示的数据就没有 2-5 行罗~ 另外，注意一下，原本应该是要下达 sed -e 才对，没有 -e 也行啦！同时也要注意的，sed 后面接的动作，请务必以 " 两个单引号括住喔！

只要删除第 2 行

```
nl /etc/passwd | sed '2d'
```

要删除第 3 到最后一行

```
nl /etc/passwd | sed '3,$d'
```

在第二行后(亦即是加在第三行)加上『drink tea?』字样！

```
[root@www ~]# nl /etc/passwd | sed '2a drink tea'
1 root:x:0:0:root:/root:/bin/bash
2 bin:x:1:1:bin:/bin:/sbin/nologin
drink tea
3 daemon:x:2:2:daemon:/sbin:/sbin/nologin
.....(后面省略).....
```

那如果是要在第二行前

```
nl /etc/passwd | sed '2i drink tea'
```

如果是要增加两行以上，在第二行后面加入两行字，例如『Drink tea or』与『drink beer?』

```
[root@www ~]# nl /etc/passwd | sed '2a Drink tea or .....\  
> drink beer ?'  
1 root:x:0:0:root:/root:/bin/bash  
2 bin:x:1:1:bin:/bin:/sbin/nologin  
Drink tea or .....  
drink beer ?  
3 daemon:x:2:2:daemon:/sbin:/sbin/nologin  
.....(后面省略).....
```

每一行之间都必须要以反斜杠『\』来进行新行的添加喔！所以，上面的例子中，我们可以发现在第一行的最后面就有\存在。

以行为单位的替换与显示

将第2-5行的内容取代成为『No 2-5 number』呢？

```
[root@www ~]# nl /etc/passwd | sed '2,5c No 2-5 number'  
1 root:x:0:0:root:/root:/bin/bash  
No 2-5 number  
6 sync:x:5:0:sync:/sbin:/bin/sync  
.....(后面省略).....
```

透过这个方法我们就能够将数据整行取代了！

仅列出 /etc/passwd 文件内的第 5-7 行

```
[root@www ~]# nl /etc/passwd | sed -n '5,7p'  
5 lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
```

```
6 sync:x:5:0:sync:/sbin:/bin/sync
7 shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
```

可以透过这个 sed 的以行为单位的显示功能，就能够将某一个文件内的某些行号选择出来显示。

数据的搜寻并显示

搜索 /etc/passwd有root关键字的行

```
n1 /etc/passwd | sed '/root/p'
1 root:x:0:0:root:/root:/bin/bash
1 root:x:0:0:root:/root:/bin/bash
2 daemon:x:1:1:daemon:/usr/sbin:/bin/sh
3 bin:x:2:2:bin:/bin:/bin/sh
4 sys:x:3:3:sys:/dev:/bin/sh
5 sync:x:4:65534:sync:/bin:/bin/sync
.... 下面忽略
```

如果root找到，除了输出所有行，还会输出匹配行。

使用-n的时候将只打印包含模板的行。

```
n1 /etc/passwd | sed -n '/root/p'
1 root:x:0:0:root:/root:/bin/bash
```

数据的搜寻并删除

删除/etc/passwd所有包含root的行，其他行输出

```
n1 /etc/passwd | sed '/root/d'
2 daemon:x:1:1:daemon:/usr/sbin:/bin/sh
3 bin:x:2:2:bin:/bin:/bin/sh
.... 下面忽略
#第一行的匹配root已经删除了
```

数据的搜寻并执行命令

搜索/etc/passwd,找到root对应的行,执行后面花括号中的一组命令,每个命令之间用分号分隔,这里把bash替换为blueshell,再输出这行:

```
nl /etc/passwd | sed -n '/root/{s/bash/blueshell/;p;q}'
1 root:x:0:0:root:/root:/bin/blueshell
```

最后的q是退出。

数据的搜寻并替换

除了整行的处理模式之外, sed 还可以用行为单位进行部分数据的搜寻并取代。基本上 sed 的搜寻与替代的与 vi 相当的类似!他有点像这样:

```
sed 's/要被取代的字串/新的字串/g'
```

先观察原始信息,利用 /sbin/ifconfig 查询 IP

```
[root@www ~]# /sbin/ifconfig eth0
eth0 Link encap:Ethernet HWaddr 00:90:CC:A6:34:84
inet addr:192.168.1.100 Bcast:192.168.1.255 Mask:255.255.255.0
inet6 addr: fe80::290:ccff:fea6:3484/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
.....(以下省略).....
```

本机的ip是192.168.1.100。

将 IP 前面的部分予以删除

```
[root@www ~]# /sbin/ifconfig eth0 | grep 'inet addr' | sed 's/^.*addr://g'
192.168.1.100 Bcast:192.168.1.255 Mask:255.255.255.0
```

'^':表示每行开头;

'.':可匹配任何字符

'*':匹配字符串中的单字符或其重复序列

表示行开头任何字符到"address:"被删除

接下来则是删除后续的部分，亦即：192.168.1.100 Bcast:192.168.1.255 Mask:255.255.255.0

将 IP 后面的部分予以删除

```
[root@www ~]# /sbin/ifconfig eth0 | grep 'inet addr' | sed 's/^.*addr://g' | sed 's/Bcast.*$//g'
192.168.1.100
```

多点编辑

一条 sed 命令，删除/etc/passwd第三行到末尾的数据，并把bash替换为blueshell

```
nl /etc/passwd | sed -e '3,$d' -e 's/bash/blueshell/'
1 root:x:0:0:root:/root:/bin/blueshell
2 daemon:x:1:1:daemon:/usr/sbin:/bin/sh
```

-e表示多点编辑，第一个编辑命令删除/etc/passwd第三行到末尾的数据，第二条命令搜索bash替换为blueshell。

直接修改文件内容(危险动作)

sed 可以直接修改文件的内容，不必使用管道命令或数据流重导向！不过，由於这个动作会直接修改到原始的文件，所以请你千万不要随便拿系统配置来测试！我们还是使用下载的 regular_express.txt 文件来测试看看吧！

利用 sed 将 regular_express.txt 内每一行结尾若为 . 则换成！

```
[root@www ~]# sed -i 's/\.$/\!/g' regular_express.txt
```

利用 sed 直接在 regular_express.txt 最后一行加入『# This is a test』

```
[root@www ~]# sed -i '$a # This is a test' regular_express.txt
```

由於 \$ 代表的是最后一行，而 a 的动作是新增，因此该文件最后新增『# This is a test』！

sed 的『-i』选项可以直接修改文件内容，这功能非常有帮助！举例来说，如果你有一个 100 万行的文件，你要在第 100 行加某些文字，此时使用 vim 可能会疯掉！因为文件太大了！那怎么办？就利用 sed 啊！透过 sed 直接修改/取代的功能，你甚至不需要使用 vim 去修订！

[← Shell 文件包含](#)[Nginx 安装配置 →](#)[✎ 点我分享笔记](#)

笔记需要是本篇文章的内容扩展！

[反馈/建议](#)

1 PayPal

[广告 ×](#)

一个账户，收款全球。0 费用
开户，享卖家保障，赢逾 2
亿用户。 [paypal.com](https://www.paypal.com)



2 JAVA 零基础 免费公开课

[×](#)

100G 学习教程、开发工具、
源代码等学习资源，加群获取
wulv5.com



在线实例

- [HTML 实例](#)
- [CSS 实例](#)
- [JavaScript 实例](#)
- [Ajax 实例](#)
- [jQuery 实例](#)
- [XML 实例](#)
- [Java 实例](#)

字符集&工具

- [HTML 字符集设置](#)
- [HTML ASCII 字符集](#)
- [HTML ISO-8859-1](#)
- [HTML 实体符号](#)
- [HTML 拾色器](#)
- [JSON 格式化工具](#)

最新更新

- [Python Set update\(\)](#)
- [Python Set union\(\)](#)
- [Python Set symmetric_difference\(\)](#)
- [Python Set symmetric_difference_update\(\)](#)
- [Python Set remove\(\)](#)
- [Python Set pop\(\)](#)
- [Python Set issue\(\)](#)

站点信息

- [意见反馈](#)
- [免责声明](#)
- [关于我们](#)
- [文章归档](#)



关注微信



Copyright © 2013-2018 菜鸟教程
Reserved. 备案号：闽ICP备13022339号-1

[反馈/建议](#)

All Rights Reserved