Linux 中 grep 命令的 12 个实践例子

2017-10-07 [Linux爱好者](https://mp.weixin.qq.com/s?__biz=MzAxODI5ODMwOA==&mid=2666541291&idx=1&sn=75f2defc843c1b296802318c5080ab24&chksm=80dcf440b7ab7d56799a44f27d4820e02a7eb1d00ea8522948afe0eb72a534464874698a7d38&scene=0&key=cddc325a4ccd2d65b19f9fdc0102215371214a18d374f0b3a50ad85293b164b735d94e52312de62b3255ab2aab74efcfaf23405ce445f21737bc4cf382ca17ab26cf00f82b43ca6755bca435756efdb7&ascene=14&uin=Nzk3NjIyNQ%3D%3D&devicetype=Windows+8&version=6205051a&pass_ticket=onjmyjVR7lvEPDpxEkme1GKgogCO%2FI1xhjApYgndFhU%3D&winzoom=1##)

（点击上方公众号，可快速关注）

英文：tecmint，翻译：Linux中国/Linux-pdz

linux.cn/article-2250-1.html

如有好文章[投稿，请点击 → 这里了解详情](http://mp.weixin.qq.com/s?__biz=MjM5MzA0OTkwMA==&mid=2651194597&idx=2&sn=28611bbc39416581d1e69accb8766c1e&scene=21#wechat_redirect)

你是否遇到过需要在文件中查找一个特定的字符串或者样式，但是不知道从哪儿开始？那么,就请grep来帮你吧。

grep是每个Linux发行版都预装的一个强有力的文件模式搜索工具。无论何种原因，如果你的系统没有预装它的话，你可以很容易的通过系统的包管理器来安装它（Debian/Ubuntu系中的apt-get和RHEl/CentOS/Fedora系中的yum）。

$ sudo apt-get install grep #Debian/Ubuntu $ sudo yum install grep #RHEL/CentOS/Fedora

我发现使用现实世界中的真实例子让你投身其中是让你接触grep命令的最容易方式。

**1.搜索和寻找文件**

假设你已经在你的电脑上安装了一个全新的Ubuntu，然后你打算卸载Python。你浏览网页寻找教程，但是你发现存在两个不同版本的Python在使用，而你不知道你的Ubuntu安装器到底在你的系统中安装了哪个版本的Python，也不知道它安装了哪些模块。解决这个烦恼只需简单的运行以下命令：

 $ sudo dpkg -l | grep -i python

**输出例子**

   ii  python2.7         2.7.3-0ubuntu3.4 Interactive high-level object-oriented language (version 2.7)

   ii  python2.7-minimal 2.7.3-0ubuntu3.4 Minimal subset of the Python language (version 2.7)

   ii  python-openssl    0.12-1ubuntu2.1  Python wrapper around the OpenSSL library

   ii  python-pam        0.4.2-12.2ubuntu4 A Python interface to the PAM library

首先，我们运行dpkg -l列出你系统上安装的.deb包。接着，我们使用管道将输出结果传输给命令grep -i python，这一步可以简单解释为把结果传输给grep然后过滤出所有含有python的项，并返回结果。–i选项用于忽略大小写,因为 grep 是大小写敏感的。使用选项-i是个好习惯，除非你打算进行更细节的搜索。

**2.搜索和过滤文件**

grep还可以在一个或多个文件里用于搜索和过滤。让我们来看一个这样的情景：

你的Apache网页服务器出现了问题，你不得不从许多专业网站里找一个发帖询问。好心回复你的人让你粘贴上来你的/etc/apache2/sites-available/default-ssl文件内容。假如你能移除掉所有的注释行，那么对你，对帮你的人，以及所有阅读该文件的人，不是更容易发现问题吗？你当然可以很容易的做到！只需这样做就可以了：

 $ sudo grep -v "#" /etc/apache2/sites-available/default-ssl

选项-v是告诉grep命令反转它的输出结果，意思就是不输出匹配的项，做相反的事，打印出所有不匹配的项。这个例子中，有#的是注释行（译注：其实这个命令并不准确，包含“#”的行不全是注释行。关于如何精确匹配注释行，可以了解更多的关于正则表达式的内容。）。

**3.找出所有的mp3文件**

grep命令对于过滤来自于标准输出的结果非常有用。例如，假设你的一个文件夹里面全是各种格式的音乐文件。你要找出艺术家jayZ的所有mp3格式的音乐文件，里面也不要有任何混合音轨。使用find命令再结合管道使用grep就可以完成这个魔法：

 $ sudo find . -name ".mp3" | grep -i JayZ | grep -vi "remix""

在这个例子中，我们使用find命令打印出所有以.mp3为后缀名的文件，接着将其使用管道传递给grep -i过滤和打印出名字为“JayZ”的文件,再使用管道传送给grep -vi以便过滤掉含有“remix”的项。

**4.在搜索字符串前面或者后面显示行号**

另外两个选项是-A和-B之间的切换，是用以显示匹配的行以及行号，分别控制在字符串前或字符串后显示的行数。Man页给出了更加详细的解释，我发现一个记忆的小窍门：-A=after、-B=before。

 $ sudo ifconfig | grep -A 4 etho $ sudo ifconfig | grep -B 2 UP

**5.在匹配字符串周围打印出行号**

grep命令的-C选项和例4中的很相似，不过打印的并不是在匹配字符串的前面或后面的行，而是打印出两个方向都匹配的行（译注：同上面的记忆窍门一样：-C=center，以此为中心）： $ sudo ifconfig | grep -C 2 lo

**6.计算匹配项的数目**

这个功能类似于将grep输出的结果用管道传送给计数器（wc程序），grep内建的选项可以达到同样的目的：

$ sudo ifconfig | grep -c inet6

**7.按给定字符串搜索文件中匹配的行号**

当你在编译出错时需要调试时，grep命令的-n选项是个非常有用的功能。它能告诉你所搜索的内容在文件的哪一行：

$ sudo grep -n "main" setup.py

**8.在所有目录里递归的搜索**

假若你要在当前文件夹里搜索一个字符串，而当前文件夹里又有很多子目录，你可以指定一个-r选项以便于递归的搜索： $ sudo grep -r “function” \*

**9.进行精确匹配搜索**

传递-w选项给grep命令可以在字符串中进行精确匹配搜索（译注：包含要搜索的单词，而不是通配）。例如，像下面这样输入：

$ sudo ifconfig | grep -w “RUNNING”

将打印出含有引号内匹配项的行。另外，你还可以试一下这个：

$ sudo ifconfig | grep -w “RUN”

搜索这个匹配项时，若搜索的东西里面没有这样的一个单独的单词，将什么也不会返回。

**10.在Gzip压缩文件中搜索**

我们还要关注一下grep的衍生应用。第一个是zgrep，这个与zcat很相似，可以用于gzip压缩过的文件。它有与grep相似的命令选项，使用方式也一样：

$ sudo zgrep -i error /var/log/syslog.2.gz

**11.在文件中匹配正则表达式**

egrep是另一个衍生应用，代表着“扩展全局正则表达式”。它可以识别更多的正则表达式元字符，例如at + ? | 和（）。在搜索源代码文件时，egrep是一个非常有用的工具，还有其他的一些零碎代码文件的搜索需要，使得这样的搜索能力成为必需。可以在grep命令中使用选项-E来启用它。

$ sudo grep -E

**12.搜索一个固定匹配字符串**

fgrep用于在一个文件或文件列表中搜索固定样式的字符串。功能与grep -F同。fgrep的一个通常用法为传递一个含有样式的文件给它：

$ sudo fgrep -f file\_full\_of\_patterns.txt file\_to\_search.txt

这仅仅是grep命令的开始，你可能已经注意到，它对于实现各种各样的需求简直是太有用了。除了这种我们运行的这种只有一行的命令，grep还可以写成cron任务或者自动的shell脚本去执行。保持好奇心，试验一下man页的各个选项，为实现你的目的写出一些grep表达式吧。