

- [Advertise](#)
- [Linux Services](#)
- [About](#)
- [Archives](#)
- [Contact Us](#)

**High-performance Tablets**  
 All under **US \$100**  
 • 7-inch screen • 3G & dual core  
 • GPS function • 1080P

**All xpress**  
[Shop Now >](#)

- [Home](#)
- [Linux Distro's](#)
- [Open Source](#)
- [Web Servers](#)
  - [Apache](#)
  - [Nginx](#)
  - [Lighttpd](#)
- [Databases](#)
  - [MySQL](#)
  - [MongoDB](#)
  - [CouchDB](#)
- [Free Linux eBooks](#)
- [Linux Commands](#)

 Powered by [Google Translate](#)

Did You Know?

[Write For Us! and Earn \\$\\$\\$s](#)

 Mark Shuttleworth has announced the codename as "**Trusty Tahr**" for **Ubuntu 14.04 LTS**

## 12 Practical Examples of Linux grep Command

 By [Rob Krul](#) Under: [Linux Commands](#) On: **November 1, 2013**

 Download Your Free eBook NOW - [A Newbie's Getting Started Guide to Linux](#)

Have you ever been confronted with the task of looking for a particular string or pattern in a file, yet have no idea where to start looking? Well then, here is **grep** to the rescue!



### 12 Grep Command Examples

**grep** is a powerful file pattern searcher that comes equipped on every distribution of **Linux**. If, for whatever reason, it is not installed on your system, you can easily install it via your package manager (**apt-get** on **Debian/Ubuntu** and **yum** on **RHEL/CentOS/Fedora**).

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

I have found that the easiest way to get your feet wet with **grep** is to just dive right in and use some real world examples.

## 1. Search and Find Files

Let's say that you have just installed a fresh copy of the new **Ubuntu** on your machine, and that you are going to give **Python** scripting a shot. You have been scouring the web looking for tutorials, but you see that there are two different versions of **Python** in use, and you don't know which one was installed on your system by the **Ubuntu** installer, or if it installed any modules. Simply run this command:

```
$ sudo dpkg -l | grep -i python
```

### Sample Output

```
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
```

First, we ran **dpkg -l**, which lists installed **\*.deb** packages on your system. Second, we piped that output to **grep -i python**, which simply states “go to grep and filter out and return everything with ‘python’ in it.” The **-i** option is there to ignore-case, as **grep** is case-sensitive. Using the **-i** option is a good habit of getting into, unless of course you are trying to nail down a more specific search.

## 2. Search and Filter Files

The **grep** can also be used to search and filter within individual files or multiple files. Let's take this scenario:

You are having some trouble with your **Apache Web Server**, and you have reached out to one of the many awesome forums on the net asking for some help. The kind soul who replies to you has asked you to post the contents of your **/etc/apache2/sites-available/default-ssl** file. Wouldn't it be easier for you, the guy helping you, and everyone reading it, if you could remove all of the commented lines? Well you can! Just run this:

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

The **-v** option tells **grep** to invert its output, meaning that instead of printing matching lines, do the opposite and print all of the lines that don't match the expression, in this case, the **#** commented lines.

## 3. Find all .mp3 Files Only

The **grep** can be very useful for filtering from **stdout**. For example, let's say that you have an entire folder full of music files in a bunch of different formats. You want to find all of the **\*.mp3** files from the artist **JayZ**, but you don't want any of the remixed tracks. Using a **find command** with a couple of **grep** pipes will do the trick:

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

In this example, we are using **find** to print all of the files with a **\*.mp3 extension**, piping it to **grep -i** to filter out and prints all files with the name “**JayZ**” and then another pipe to **grep -vi** which filters out and does not print all filenames with the string (in any case) “**remix**”.

1. [35 Practical Examples of Linux Find Command](#)

## 4. Display Number of Lines Before or After Search String

Another couple of options are the **-A** and **-B** switches, which displays the matched line and number of lines either that come before or after the search string. While the man page gives a more detailed explanation, I find it easiest to remember the options as **-A = after**, and **-B = before**:

```
$ sudo ifconfig | grep -A 4 eth0
$ sudo ifconfig | grep -B 2 UP
```

## 5. Prints Number of Lines Around Match

The **grep**'s **-C** option is similar, but instead of printing the lines that come either before or after the string, it prints the lines in either direction:

```
$ sudo ifconfig | grep -C 2 lo
```

## 6. Count Number of Matches

Similar to piping a **grep** string to word count (**wc** program) **grep**'s built-in option can perform the same for you:

```
$ sudo ifconfig | grep -c inet6
```

## 7. Search Files by Given String

The **-n** option for **grep** is very useful when debugging files during compile errors. It displays the line number in the file of the given search string:

```
$ sudo grep -n "main" setup.py
```

## 8. Search a string Recursively in all Directories

If you would like to search for a string in the current directory along with all of the subdirectories, you can specify the **-r** option to search recursively:



2

[Share](#)[Leave a Comment](#)[Local Restaurant](#)[Scripting](#)[File system backup](#)[Web hosting servers](#)[Database](#)[Regular Expression](#)[Backup Utilities](#)[Free ebook pdf](#)[infolinks](#)[« Previous Post](#)[10 Lesser Known Linux Commands – Part 2](#)[Next Post »](#)**Related Post(s):**

1. [12 TOP Command Examples in Linux](#)
2. [How to Reset Forgotten Root Password in RHEL/CentOS and Fedora](#)
3. [12 Useful “df” Commands to Check Disk Space in Linux](#)
4. [Pydf an Alternative “df” Command to Check Disk Usage in Different Colours](#)
5. [20 Advanced Commands for Linux Experts](#)
6. [15 Useful “ifconfig” Commands to Configure Network Interface in Linux](#)

**Leave a Reply**

<input type="text"/>	Name (Required)
<input type="text"/>	Mail (will not be published) (Required)
<input type="text"/>	Website
<div></div>	

[Submit Comment](#)

- [Twitter 1070](#) [Twitter](#)
- [Facebook 17351](#) [Facebook](#)
- [Google+ 921](#) [Google+](#)
- [RSS 1905](#) [Subscribers](#)

Become A **TecMint** Subscriber to receive latest **Updates**. Enter Your Email Addr [Signup!](#)

- [Popular](#)
- [Latest](#)
- [Comments](#)
- [Tags](#)



- [Install Apache 2.2.15, MySQL 5.5.34 & PHP 5.5.4 on RHEL/CentOS 6.4/5.9 & Fedora 19-12](#) 113 Comments



- [Wine 1.7.3 Released – Install in RHEL, CentOS and Fedora](#) 108 Comments



- [Install Cacti \(Network Monitoring\) on RHEL/CentOS 6.3/5.8 and Fedora 17-12](#) 103 Comments



- [CentOS 6.4 Step by Step Installation Guide with Screenshots](#) 88 Comments



- [CentOS 6.3 Step by Step Installation Guide with Screenshots](#) 84 Comments



- [Google Chrome 29 Released – Install on RHEL/CentOS 6 and Fedora 19/15](#) 70 Comments

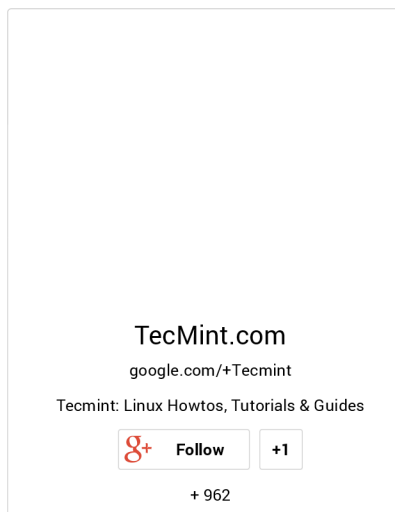
:: Advertise ::



:: Download Free Linux eBooks ::

- [Introduction to Linux – A Hands on Guide](#)
- [The GNU/Linux Advanced Administration](#)
- [Securing & Optimizing Linux: The Hacking Solution](#)
- [Linux Command Line Cheat Sheet](#)
- [A Newbie's Getting Started Guide to Linux](#)
- [Create Your Own Linux System from Scratch](#)
- [Linux Shell Scripting Cookbook – Second Edition](#)
- [Linux Bible – The Complete Tutorial Resource](#)

:: Follow Us ::



## :: About ::

TecMint.com is a website that publishes practical and useful out-of-the-box articles for aspirant like you and me. We seek to present exceptional, remarkable tips, tutorials, and resources that the modern web professional will appreciate.

## :: Our Services ::

[Know More](#)

We offer wide range of Linux Web Hosting and Management Services includes Linux hosting, WordPress hosting, Joomla Hosting, CMS hosting, Website migration and Custom solutions, making us a one-stop destination for all your possible hosting needs at fair minimum rates.

## :: Advertise ::

[Submit Order](#)

TecMint.com is visited by tens of thousands of Linux users and has a excellent reputation in the search engine ranking. Most of the traffic comes from Google organic search (80%). Spread your messages or products to an engaged readers by advertising with us.

[Home](#) | [Privacy Policy](#) | [Copyright Policy](#)

© 2012-2013 All Rights Reserved.

[Advertise Now](#)

- [25 Hardening Security Tips for Linux Servers](#)
- [60 Commands of Linux : A Guide from Newbies to System Administrator](#)
- [15 Command Line Tools to Monitor Linux Performance](#)
- [5 Best Practices to Secure and Protect SSH Server](#)
- [18 Tar Command Examples in Linux](#)
- [20 Linux YUM \(Yellowdog Updater, Modified\) Commands](#)
- [25 Useful Basic Commands of APT-GET and APT-CACHE](#)
- [20 Funny Commands of Linux or Linux is Fun in Terminal](#)
- [35 Practical Examples of Linux Find Command](#)
- [10 Linux Distributions and Their Targeted Users](#)