

How To: Use 'apt-cache' to Find Homepage for Your Installed Apps

It can come in pretty handy to know for certain the homepage for the applications you have installed. You can do this with 'apt-cache'. I'll show you how. This is a pretty easy article to follow and just another tool to add to your toolbox.

NOTE: This is only valid for systems that use apt. As the title indicates, it requires 'apt-cache'. Without apt-cache, this page will do you no good. None good. That's how much it will do you. None!

Why would you want to know the homepage – and, more so, the preferred homepage? For starters, in the days of GitHub and everyone forking, and awkward application names that aren't easily searched for, it's hard to know which site is the correct one.

Maybe you want to report a bug? Maybe you want to request a feature? Maybe you want to make a donation? Maybe you just want to thank the author for writing such awesome software? Maybe you want to know where the homepage is because you need support and you're not sure where to turn to?

There are all sorts of reasons why you might want to know the homepage of a piece of software. It's actually something that's important. It's also something your system already knows and will happily show you if you know the proper magical incantation.

Like many other articles, you're gonna want the terminal for this. Let's go ahead and get that opened by using your keyboard and pressing CTRL + ALT + T.

Got your terminal emulator open? Good! Let's start with the command.

```
[code]apt-cache show inxi[/code]
```

If you do not have 'inxi' installed, feel free to use a different application. Note that you do not need to use sudo for this. Not all apt commands require sudo. You only need sudo when you're actually doing administrative tasks. See? I saved you some typing!

Anyhow, in the text output from the above command you'll see a line that starts with "Homepage:". If you hadn't already guessed it, that's the line that tells you the authors homepage. This, of course, only works on installed applications. For sanity and space sake, it's not like your system has all that information downloaded for all the possible packages. Thus, it works on naught but the apps you have already installed.

So, let's go ahead and make the command a little more precise. We'll pipe the output through grep and get rid of the cruft we don't actually need. In that same terminal, go ahead and enter:

```
[code]apt-cache show inxi | grep Homepage[/code]
```

NOTE: The command contains a capitalized letter H because Linux is often case-sensitive and is certainly case-sensitive in this case. If you don't believe me, try it with a lowercase h!

But wait, there's more!

Not only is there homepage information in there, there's sometimes some useful nuggets of information in there. If you have LibreOffice installed, go ahead and check (skip the pipe and grepping) to see what the output is. Inside, it has a ton of additional information, including listing ways that you can

extend LibreOffice by installing more software.

And there you have it. You can now easily find the homepage for the applications you have installed. Should you need to contact the author, check for information, or just see if they did anything else, you now know how to do that. It's a little hidden nugget that most folks don't seem to know. Well, now they do...

Yay! You made it all the way to the bottom. You deserve a treat. Seeing as you've already got the terminal open, and seeing as we're dealing with apt-cache, let's just get some pretty neat stats with:

```
[code]apt-cache stats[/code]
```

That's it and thanks for reading. I appreciate the audience and am happy that I finally am putting some effort into this project. I've been meaning to do this for years, but something always got in the way. If you want to get notified when new articles are posted, just scroll up and sign up for the newsletter. I promise not to send you any commercial emails and I won't give any of your private data away.

USN-4895-1: Squid vulnerabilities

Alex Rousskov and Amit Klein discovered that Squid incorrectly handled certain Content-Length headers. A remote attacker could possibly use this issue to perform an HTTP request smuggling attack, resulting in cache poisoning. This issue only affected Ubuntu 20.04 LTS.

(CVE-2020-15049)

Jianjun Chen discovered that Squid incorrectly validated certain input. A

remote attacker could use this issue to perform HTTP Request Smuggling and

possibly access services forbidden by the security controls.

(CVE-2020-25097)

USN-4894-1: WebKitGTK vulnerabilities

A large number of security issues were discovered in the WebKitGTK Web and

JavaScript engines. If a user were tricked into viewing a malicious

website, a remote attacker could exploit a variety of issues related to web

browser security, including cross-site scripting attacks, denial of service

attacks, and arbitrary code execution.

How To: List USB Device Info From The Terminal (lsusb)

In this article we will learn to use the terminal to check what USB devices are connected to your computer. We will be using the 'lsusb' command for this one. This article should be both quick and easy.

You may be interested in the three previous articles. In those, we covered 'lshw', 'lscpu', and 'lspci'. Seeing as we've got a good thing going, we'll go ahead and cover 'lsusb' this time. Like the rest, the function becomes evident by the name, and the man page describes it as thus:

lsusb – list USB devices

You shouldn't have to install anything. Your distro should already have the ability to run this command. So, without further ado, let's open your terminal by pressing CTRL + ALT + T. Once you have it open, you can give the command a try.

```
[code]lsusb[/code]
```

This one doesn't need any elevated permissions (see further information below), you can run it as a regular user. As you see from the output, all your connected USB devices should be listed and enumerated. Quite frankly, that's about all you'll ever really need to do with the command – but there is indeed more.

So, what else can you do with it? Well, for starters you can see it in tree view. That may not seem important – but wait for it, there's another use coming up and the tree view is great for seeing what's attached to what hub without digging behind your computer. Let's go ahead and use it with the tree switch:

```
[code]lsusb -t[/code]
```

That's easy enough and informative enough. Like I said, this is useful if you have USB hubs and want to see what's connected to what. But, that's still not the neatest thing you can do with 'lsusb'. Let's go back to the original command.

```
[code]lsusb[/code]
```

Take a look at the output. For example, you might see

something like this:

```
[code]Bus 001 Device 005: ID 0bda:8178 Realtek Semiconductor Corp. RTL8192CU 802.11n WLAN Adapter[/code]
```

Now, I want you to look at those first two sections, 'bus' and 'device'. Those are actually pretty useful. You just need to know how to use them. Keep them in the same order, and add them to the following command:

```
[code]lsusb -D /dev/bus/usb/<bus #>/<device #>[/code]
```

So, using my example above, you end up with a command that looks like:

```
[code]lsusb -D /dev/bus/usb/001/005[/code]
```

See, the -D switch is for Device and, curiously, the manual says that you need to be root to use it. However, I've not ever needed root to do so. Allow me to quote again:

Do not scan the /dev/bus/usb directory, instead display only information about the device whose device file is given. The device file should be something like /dev/bus/usb/001/001. This option displays detailed information like the v option; you must be root to do this.

Try as I might, I can't see a lick of difference when I run the command without root. So, I have no idea why the manual says I should, or that I must. If anyone knows why, please leave a comment. (Contrary to popular opinion, I do not in fact know everything!)

Anyhow, that last use of lsusb is the most informative. The output is too long for me to bother copy/pasting here, though padding the article would be nice. So, go ahead and run it for yourself. All you need to do is check the bus and device numbers, insert them properly into the path, and you'll get a ton of info you probably didn't know was available.

Once again, thanks for reading. Be sure to scroll up, look right, and sign up for the newsletter. You'll get a lovely little notification every time there's an article published. Feel free to leave a comment and let me know why it says root is required. As near as I can tell, that isn't necessary?

[CentOS-announce]

CESA-2021:0996 Important

CentOS 7 thunderbird Security Update

CentOS Errata and Security Advisory 2021:0996 Important
Upstream details at :
<https://access.redhat.com/errata/RHSA-2021:0996>

The following updated files have been uploaded and are currently

syncing to the mirrors: (sha256sum Filename)

x86_64:

412317b2522f388f60a8b9846d99020fa2c884e8557b0552ad09b4218e9780
3d thunderbird-78.9.0-3.el7.centos.x86_64.rpm

Source:

9b3ff2329273f188e644f9e8fb481e12ff32397fac7f7f9b4a689aa99d8452
9b thunderbird-78.9.0-3.el7.centos.src.rpm

—

Johnny Hughes

CentOS Project { <http://www.centos.org/> }

irc: hughesjr, #[hidden email]

Twitter: @JohnnyCentOS

CentOS-announce mailing list

[hidden email]

<https://lists.centos.org/mailman/listinfo/centos-announce>