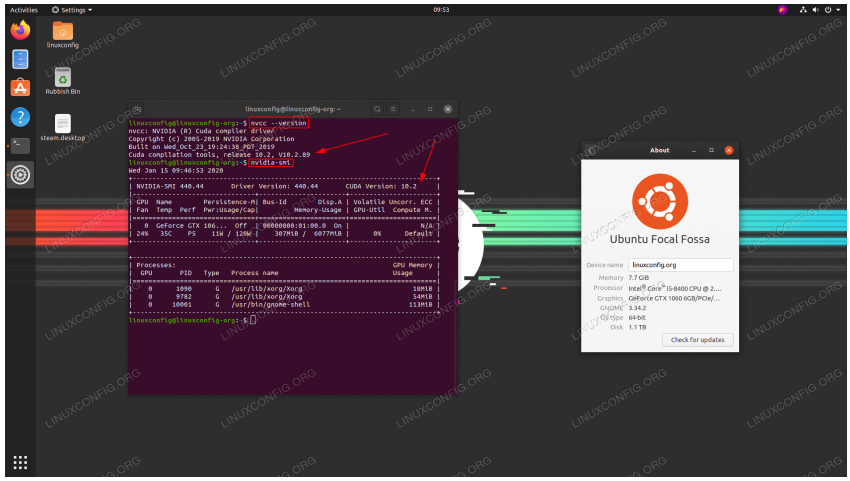
How to install CUDA on Ubuntu 20.04 Focal Fossa Linux

Lubos Rendek Ubuntu 04 May 2020

The Nvidia CUDA toolkit is an extension of the GPU parallel computing platform and programming model. The Nvidia CUDA installation consists of inclusion of the official Nvidia CUDA repository followed by the installation of relevant meta package and configuring path the the executable CUDA binaries.

In this tutorial you will learn:

- How to install CUDA toolkit from Ubuntu Repository
- How to install CUDA toolkit from CUDA repository
- How to compile example CUDA C code and execute program
- How to Check CUDA version



(/images/01-how-to-install-cuda-on-ubuntu-20-04-focal-fossa-linux.png)

CUDA on Ubuntu 20.04 Focal Fossa Linux

Software Requirements and Conventions Used

Software Requirements and Linux Command Line Conventions

Category	Requirements, Conventions or Software Version Used
System	Installed Ubuntu 20.04 (how-to-install-ubuntu-20-04-focal-fossa-desktop) or upgraded Ubuntu 20.04 Focal Fossa (/how-to-upgrade-ubuntu-to-20-04-lts-focal-fossa)
Software	CUDA
Other	Privileged access to your Linux system as root or via the `sudo` command.
Conventions	**#** - requires given linux commands (/linux-commands) to be executed with root privileges either directly as a root user or by use of `sudo` command **$** - requires given linux commands (/linux-commands) to be executed as a regular non-privileged user

How to install CUDA on Ubuntu 20.04 step by step instructions

SUBSCRIBE TO NEWSLETTER

Subscribe to Linux Career NEWSLETTER (https://bit.ly/2X5D30q) and receive latest Linux news, jobs, career advice and tutorials.

How to install CUDA toolkit from Ubuntu Repository

Step 1 Although you might not end up witht he latest CUDA toolkit version, the easiest way to install CUDA on Ubuntu 20.04 (ubuntu-20-04-guide) is to perform the installation from Ubuntu's standard repositories.

To install CUDA execute the following commands:

```
$ sudo apt update
$ sudo apt install nvidia-cuda-toolkit
```

Step 2 All should be ready now. Check your CUDA version:

```
$ nvcc --version
nvcc: NVIDIA (R) Cuda compiler driver
Copyright (c) 2005-2019 NVIDIA Corporation
Built on Sun_Jul_28_19:07:16_PDT_2019
Cuda compilation tools, release 10.1, V10.1.243
```

NOTE

To install latest and more up to date CUDA version see below **How to install CUDA toolkit from CUDA repository** section.

Step 3 Confirm the CUDA toolkit installation by sample CUDA C code compilation. See the bellow **Compile a Sample CUDA code** section.

How to install CUDA toolkit from CUDA repository

Step 1 In case you have not done so yet, make sure that you have installed the Nvdia driver for your VGA. To do so follow our guide on How to install the NVIDIA drivers on Ubuntu 20.04 Focal Fossa Linux. (/how-to-install-the-nvidia-drivers-on-ubuntu-20-04-focal-fossa-linux)

Step 2 Setup Nvida CUDA repository.

NOTE
At the time of writing the Ubuntu 20.04 Cuda driver version is not yet available. From this reason we will resort to the the latest stable version which made for Ubuntu 18.04.

Execute the following commands to enable CUDA repository.

```
$ wget -O /etc/apt/preferences.d/cuda-repositor
$ sudo apt-key adv --fetch-keys https://develop
$ sudo add-apt-repository "deb http://developer
```

Step 3 At this stage all should be ready to install CUDA. Execute the following `apt` command:

```
$ sudo apt install cuda
```

Step 4 Once ready, set your path to point to CUDA binaries:

```
$ echo 'export PATH=/usr/local/cuda/bin${PATH:+
```

Step 5 Check CUDA version to confirm the installation:

```
$ nvcc --version
nvcc: NVIDIA (R) Cuda compiler driver
Copyright (c) 2005-2019 NVIDIA Corporation
Built on Wed_Oct_23_19:24:38_PDT_2019
Cuda compilation tools, release 10.2, V10.2.89
```

Compile a Sample CUDA code

Confirm the installation by compiling an example CUDA C code (https://devblogs.nvidia.com/easy-introduction-cuda-c-and-c/). Save the following code into a file named eg. `hello.cu` :

```
#include <stdio.h>

__global__
void saxpy(int n, float a, float *x, float *y)
{
  int i = blockIdx.x*blockDim.x + threadIdx.x;
  if (i < n) y[i] = a*x[i] + y[i];
}

int main(void)
{
  int N = 1<<20;
  float *x, *y, *d_x, *d_y;
  x = (float*)malloc(N*sizeof(float));
  y = (float*)malloc(N*sizeof(float));

  cudaMalloc(&d_x, N*sizeof(float));
  cudaMalloc(&d_y, N*sizeof(float));

  for (int i = 0; i < N; i++) {
    x[i] = 1.0f;
    y[i] = 2.0f;
  }

  cudaMemcpy(d_x, x, N*sizeof(float), cudaMemcpyHost
  cudaMemcpy(d_y, y, N*sizeof(float), cudaMemcpyHost

  // Perform SAXPY on 1M elements
  saxpy<<<(N+255)/256, 256>>>(N, 2.0f, d_x, d_y);

  cudaMemcpy(y, d_y, N*sizeof(float), cudaMemcpyDevi

  float maxError = 0.0f;
  for (int i = 0; i < N; i++)
    maxError = max(maxError, abs(y[i]-4.0f));
  printf("Max error: %f\n", maxError);

  cudaFree(d_x);
  cudaFree(d_y);
  free(x);
  free(y);
}
```

Next, use `nvcc` the Nvidia CUDA compiler to compile the code and run the newly compiled binary:

```
$ nvcc -o hello hello.cu
$ ./hello
Max error: 0.000000
```

Troubleshooting

At the moment CUDA does not support GCC compiler higher then version 8 when installed from CUDA Ubuntu 18.04 sources. As a result upon the code compilation with the Nvidia CUDA compiler you might receive the following error:

```
In file included from /usr/local/cuda-10.2/bin/
                 from :
/usr/local/cuda-10.2/bin/../targets/x86_64-linu
  138 | #error -- unsupported GNU version! gcc
      |  ^~~~~
```

To comply with the CUDA compiler requirements switch your default GCC compiler (/how-to-switch-between-multiple-gcc-and-g-compiler-versions-on-ubuntu-20-04-lts-focal-fossa) to version 8 or lower.

❮ Prev (/how-to-test-3d-acceleration-fps-on-ubuntu-20-04-focal-fossa)

Next ❯ (/how-to-install-git-on-ubuntu-20-04-lts-focal-fossa-linux)

FIND LATEST LINUX JOBS (https://www.linuxcareers.com/) on LinuxCareers.com

Submit your RESUME (https://www.linuxcareers.com/candidate/register/), create a JOB ALERT (https://www.linuxcareers.com/jobs/alert/) or subscribe to RSS (https://www.linuxcareers.com/jobs/rss/) feed.

LINUX CAREER NEWSLETTER

Subscribe to NEWSLETTER (https://bit.ly/2X5D30q) and receive latest news, jobs, career advice and tutorials.

DO YOU NEED ADDITIONAL HELP?

Get extra help by visiting our LINUX FORUM (https://forum.linuxconfig.org) or simply use comments below.

MORE ON LINUXCONFIG.ORG:

- How to check CUDA version on Ubuntu 20.04 Focal Fossa Linux (/how-to-check-cuda-version-on-ubuntu-20-04-focal-fossa-linux)
- How to uninstall the NVIDIA drivers on Ubuntu 20.04 Focal Fossa Linux (/how-to-uninstall-the-nvidia-drivers-on-ubuntu-20-04-focal-fossa-linux)
- How to install Blender on Ubuntu 20.04 Focal Fossa Linux Desktop (/how-to-install-blender-on-ubuntu-20-04-focal-fossa-linux-desktop)
- How to install the NVIDIA drivers on Ubuntu 20.04 Focal Fossa Linux (/how-to-install-the-nvidia-drivers-on-ubuntu-20-04-focal-fossa-linux)

YOU MAY ALSO BE INTERESTED IN:

Comments and Discussions

7 replies

laszewsk 10 May

Unfortunately it did not yet work for me you documented. I get the warning

dpkg: error processing archive /var/cache/apt/archives/libcublas10_10.2.2.89-1_amd64.deb (–unpack):
trying to overwrite ‘/usr/lib/x86_64-linux-gnu/libnvblas.so.10’, which is also in package libnvblas10:amd64 10.1.243-3
Errors were encountered while processing:
/var/cache/apt/archives/libcublas10_10.2.2.89-1_amd64.deb
E: Sub-process /usr/bin/dpkg returned an error code (1)

Did I miss a step?

hermanfelker 22 May

I had this happen as well. It may be related to an older version of CUDA being installed, but possibly not. Either way I was able to get past the error by running

```
  $ sudo dpkg -i --force-overwrite
/var/cache/apt/archives/libcublas10_10.2
.2.89-1_amd64.deb

  $ sudo apt --fix-broken install
```

You will get some warnings after running the first command but the --fix-broken-install clears that right up.

Cheers!

Tucker_Mirzaev 27 May

sudo apt install nvidia-cuda-toolkit
Reading package lists… Done
Building dependency tree
Reading state information… Done
Some packages could not be installed. This may mean that you have
requested an impossible situation or if you are using the unstable
distribution that some required packages have not yet been created
or been moved out of Incoming.
The following information may help to resolve the situation:

The following packages have unmet dependencies:
nvidia-cuda-toolkit : Depends: nvidia-cuda-dev (= 10.1.243-3) but it is not going to be installed
E: Unable to correct problems, you have held broken packages.

SenthilKathi 5 Jun

Hi, the installation worked well.

But when I run the sample hello.cu code, it get the output as:

Max error: 2.000000

I suppose it should be 0.000000. Is this fine? if so what is the reason for this difference?

1 reply

linamaria11 ▶ SenthilKathi 14 Jun

I'm not sure what the difference is, but when I was getting Max error:2.000000 something was wrong with my nvidia-driver (maybe due to SecureBoot) but running sudo dpkg --configure -a and then running sudo apt install nvidia-cuda-toolkit . This fixed the driver issue and then I got the error of 0.000

Dean_Schulze 26 Jul

These instructions install the latest cuda version which at this time is cuda 11. There is a serious version mismatch with the 440 driver and version 11. Before installing this when I ran nvidia-smi it indicated that is used cuda 10.2. After installing cuda nvidia-smi returns

```
Failed to initialize NVML:
Driver/library version mismatch
```

I read somewhere that Cuda 11 was required when running the 5.3 kernel, so it's not clear that I could downgrade to cuda 10.2. Maybe I need a version 440 driver written with cuda 11, if such a thing exists.

Doing

```
sudo apt remove cuda
```


NEWSLETTER

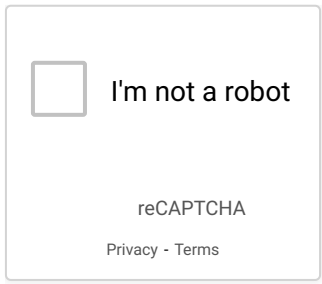
Subscribe to **Linux Career Newsletter** to receive latest news, jobs, career advice and featured configuration tutorials.

Full Name

Email

☐ **GDPR permission**: I give my consent to be in touch with me via email using the information I have provided

in this form for the purpose of news and updates.



Subscribe to Newsletter

WRITE FOR US

LinuxConfig is looking for a technical writer(s) geared towards GNU/Linux and FLOSS technologies. Your articles will feature various GNU/Linux configuration tutorials and FLOSS technologies used in combination with GNU/Linux operating system.

When writing your articles you will be expected to be able to keep up with a technological advancement regarding the above mentioned technical area of expertise. You will work independently and be able to produce at minimum 2 technical articles a month.

APPLY NOW
(https://www.linuxcareers.com/jobs/floss-technical-writer-new-york-city-new-york/1-1/)

CONTACT

web (at) linuxconfig (dot) org

FEATURED LINUX TUTORIALS

- How To enable the EPEL Repository on RHEL 8 / CentOS 8 Linux (/redhat-8-epel-install-guide)
- Bash scripting Tutorial (/bash-scripting-tutorial)

- How to install VMware Tools on RHEL 8 / CentOS 8 (/how-to-install-vmware-tools-on-rhel-8-centos-8)
- Howto mount USB drive in Linux (/howto-mount-usb-drive-in-linux)
- How to install the NVIDIA drivers on Ubuntu 18.04 Bionic Beaver Linux (/how-to-install-the-nvidia-drivers-on-ubuntu-18-04-bionic-beaver-linux)
- How to update Kali Linux (/how-to-update-kali-linux)
- Ubuntu 20.04 Download (/ubuntu-20-04-download)

- How To Upgrade Ubuntu To 20.04 LTS Focal Fossa (/how-to-upgrade-ubuntu-to-20-04-lts-focal-fossa)
- How to install node.js on RHEL 8 / CentOS 8 Linux (/how-to-install-node-js-on-redhat-8-linux)
- How to check CentOS version (/how-to-check-centos-version)
- How to Parse Data From JSON Into Python (/how-to-parse-data-from-json-into-python)
- Check what Debian version you are running on your Linux system (/check-what-debian-version-you-are-running-on-your-linux-system)
- Bash Scripting Tutorial for Beginners (/bash-scripting-tutorial-for-beginners)
- Ubuntu 20.04 Guide (/ubuntu-20-04-guide)
- How to stop/start firewall on RHEL 8 / CentOS 8 (/redhat-8-stop-start-firewall)
- Install gnome on RHEL 8 / CentOS 8 (/install-gnome-on-redhat-8)
- Linux Download (/linux-download)
- How To Upgrade from Ubuntu 18.04 and 19.10 To Ubuntu 20.04 LTS Focal Fossa (/how-to-upgrade-ubuntu-to-20-04-lts-focal-fossa)
- Enable SSH root login on Debian Linux Server (/enable-ssh-root-login-on-debian-linux-server)

LATEST ARTICLES

- How to check an hard drive health from the command line using smartctl (/how-to-check-an-hard-drive-health-from-the-command-line-using-smartctl)
- Netplan network configuration tutorial for beginners (/netplan-network-configuration-tutorial-for-beginners)

- How to create phpinfo.php page (/how-to-create-phpinfo-php-page)
- Linux Complex Bash One-Liner Examples (/linux-complex-bash-one-liner-examples)
- How to check PHP version on Ubuntu (/how-to-check-php-version-on-ubuntu)

- Useful Bash command line tips and tricks examples - Part 3 (/useful-bash-command-line-tips-and-tricks-examples-part-3)
- Advanced Linux Subshells With Examples (/advanced-linux-subshells-with-examples)
- How to change SSH port on Linux (/how-to-change-ssh-port-on-linux)
- Rsync: exclude directory (/rsync-exclude-directory)
- How to kill process by name (/how-to-kill-process-by-name)
- How to set crontab to execute every 5 minutes (/how-to-set-crontab-to-execute-every-5-minutes)
- Nslookup Linux command (/nslookup-linux-command)
- Linux Subshells for Beginners With Examples (/linux-subshells-for-beginners-with-examples)
- How to uninstall package on Ubuntu Linux (/how-to-uninstall-package-on-ubuntu-linux)
- Useful Bash command line tips and tricks examples - Part 2 (/useful-bash-command-line-tips-and-tricks-examples-part-2)

© 2007 - 2020 LinuxConfig.org