

How to install NVIDIA drivers on Kali Linux

This document explains how to install NVIDIA drivers on Kali Linux and supports CUDA, allowing integration with popular intrusion testing tools.

This document explains how to install NVIDIA drivers on Kali Linux and supports CUDA, allowing integration with popular intrusion testing tools.

Prerequisites

First, you'll need to make sure your system is completely upgraded and your card supports CUDA.

Note : GPUs with CUDA calculation capability > 5.0 are recommended, but GPUs with lower capacity will still work.

```
apt update && apt dist-upgrade -y && reboot
```

Determine exactly which GPU is installed and check the kernel modules in use.

```
root @ potassium: ~ # lspci -v
01: 00.0 VGA compatible controller: NVIDIA Corporation GM204 [GeForce GTX 970]
Subsystem: ZOTAC International (MCO) Ltd. GM204 [GeForce GTX 970]
Region 1: Memory at e0000000 (64-bit, prefetchable) [size = 256M]
Capabilities: [60] Power Management version 3
Capabilities: [68] MSI: Enable + Count = 1/1 Maskable- 64bit +
Capabilities: [78] Express (v2) Legacy Endpoint, MSI 00
Ability: [600 v1] Vendor Specific Information: ID = 0001 Rev = 1 Len = 024
Kernel driver in use: nouveau
Module kernel: nouveau
```

Setting

Once the system has restarted, please install **OpenCL ICD Loader, Drivers** and **CUDA** toolkit.

```
apt install -y ocl-icd-libopencl1 nvidia-driver nvidia-cuda-toolkit
```

While installing drivers, the system has created new kernel modules, so need to reboot again.

Verify driver installation

Now that the system is ready to work, the next step is to verify that the drivers are loaded correctly, by running the nvidia-smi tool.

```
root @ potassium: ~ # nvidia-smi
```

```
+-----+
| NVIDIA-SMI 375.26 Driver Version: 375.26 |
+-----+-----+-----+-----+
| GPU Name Persistence-M | Bus-Id Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr: Usage / Cap | Memory-Usage | GPU-Util Compute M. |
+-----+-----+-----+-----+
| 0 GeForce GTX 970 Off | 0000: 01: 00.0 On | N / A |
| 36% 46C P0 47W / 325W | 200MiB / 4036MiB | 0% Default |
+-----+-----+-----+-----+
+-----+
| Processes: GPU Memory |
| GPU PID  Type  Process name  Usage |
+-----+-----+-----+-----+
| 0 692 G / usr / lib / xorg / Xorg 198MiB |
+-----+-----+-----+-----+
```

With the correct display of drivers and GPUs, we can now go into benchmarking. Before continuing, check carefully to make sure that **hashcat** and **CUDA** are working at the same time.

```
root @ potassium: ~ # hashcat -I
OpenCL Info:
Platform ID # 1
Vendor: NVIDIA Corporation
Name: NVIDIA CUDA
Version: OpenCL 1.2 CUDA 8.0.0
Device ID # 1
Type: GPU
Vendor ID: 32
Vendor: NVIDIA Corporation
Name: GeForce GTX 970
Version: OpenCL 1.2 CUDA
Processor (s): 13
Clock: 1240
Memory: 1009/4036 MB allocatable
OpenCL Version: OpenCL C 1.2
Driver Version: 375.26
```

It looks like everything is working, go ahead and run the benchmark test.

Benchmarking

```
root @ potassium: ~ # hashcat -b
OpenCL Platform # 1: NVIDIA Corporation
=====
* Device # 1: Geforce GTX 970, 1009/4095 MB allocatable, 13MCU

Hashtype: MD5
Speed.Dev. # 1 .: 10443.1 MH / s
Hashtype: SHA1
Speed.Dev. # 1 .: 3349.8 MH / s
Hashtype: SHA256
```

```
Speed.Dev. # 1 .: 1321.8 MH / s
```

There are countless configurations to improve speed, not mentioned in this tutorial. We encourage you to read more materials for each specific case.

Resolve problem

During setup, everything can go according to plan, we will install **clinfo** for detailed troubleshooting information.

```
apt install -y clinfo
```

OpenCL Loader downloader

You may need to check for additional packages that may conflict with your settings. First, check if OpenCL Loader is installed. **The NVIDIA OpenCL Loader and OpenCL Loader** will all work on the system.

```
root @ potassium: ~ # dpkg -l | grep -i icd
ii nvidia-egl-icd: amd64 375.26-2 amd64 NVIDIA EGL installable client driver (
ii nvidia-opencl-icd: amd64 375.26-2 amd64 NVIDIA OpenCL installable client dr
ii nvidia-vulkan-icd: amd64 375.26-2 amd64 NVIDIA Vulkan installable client dr
ii ocl-icd-libopencl1: amd64 2.2.11-1
```

If mesa-opencl-icd is installed, run:

```
apt remove mesa-opencl-icd
```

Since the installation of the ICD loader is validated, we can easily determine which loader is currently being used.

```
root @ potassium: ~ # clinfo | grep -i "icd loader"
ICD loader properties
ICD loader Name OpenCL ICD Loader
ICD loader Vendor OCL Icd free software
ICD loader Version 2.2.11
ICD loader Profile OpenCL 2.1
```

As expected, the setup is using the previously installed open source loader. Now, get some details about the system.

Query GPU information

We will use **nvidia-smi** again, but with a much more detailed result.

```
root @ potassium: ~ # nvidia-smi -i 0 -q
Driver Version: 375.26
Attached GPUs: 1
GPU 0000: 01: 00.0
Product Name: GeForce GTX 970
Product Brand: GeForce
Display Mode: Enabled
```

```
Display Active: Enabled
Persistence Mode: Disabled
Accounting Mode: Disabled
Accounting Mode Buffer Size: 1920
Temperature
GPU Current Temp: 47 C
GPU Shutdown Temp: 96 C
GPU Slowdown Temp: 91 C
Clocks
Graphics: 1101 MHz
SM: 1101 MHz
Memory: 3523 MHz
Video: 1012 MHz
Processes
Process ID: 692
Type: G
Name: / usr / lib / xorg / Xorg
Used GPU Memory: 198 MiB
```

Looks like the GPU is being recognized correctly, so use `glxinfo` to determine if 3D Rendering is enabled.

```
root @ potassium: ~ # glxinfo | grep -i "direct rendering"
direct rendering: Yes
```

The combination of these tools will support the troubleshooting process a lot. If you still have problems, we recommend searching for similar settings and any factors that may affect the system.

See more:

1. [How to install Kali Linux on Android using Linux Deploy](#)
2. [How to install Kali Linux dual boot on macOS](#)
3. [How to install VmWare Tools on Kali Linux](#)

You finished reading the article "**How to install NVIDIA drivers on Kali Linux**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.