SDB:NVIDIA the hard way

openSUSE Support Database
Portal - SDB Categories - How to write an article

Situation ⎘

Installing NVIDIA's proprietary driver using their installer (.run) is desired.

This is called "the hard way" because installing via zypp is easier.

[NVIDIA ▾]

Procedure ⎘

Create a system snapshot (https://doc.opensuse.org/documentation/leap/reference/html/book.opensuse.reference
/cha.snapper.html#sec.snapper.manage.create) to rollback to if needed.

```
# snapper create
```

Install the minimum requirements ⎘

The following patterns contain the minimum requirements (https://download.nvidia.com/XFree86/Linux-x86_64/390.25/README
/minimumrequirements.html):

```
# zypper in -t pattern devel_C_C++ devel_kernel
```

The kernel-source package (548M) in devel_kernel is needed according to chapter 4 of the official installation guide
(https://download.nvidia.com/XFree86/Linux-x86_64/390.25/README/installdriver.html); however, you might investigate `--add-this-kernel` (in that same chapter).

Uninstall conflicts ⎘

On Leap 42.3 you need to uninstall the drm-kmp-default package. (https://doc.opensuse.org/release-notes/x86_64/openSUSE/Leap/42.3
/#sec.install.nouveau-extra)

```
# zypper rm drm-kmp-default
```

Get the hardware information ⎘

In a terminal:

```
# lspci | grep VGA
# lscpu | grep Arch
```

Alternatively:

```
# hwinfo --gfxcard | grep Model
# hwinfo --arch
```

There is also **YaST2 -> Hardware Information**.

Download the driver installer ⎘

These all seem to be official sources:

- http://www.nvidia.com/Download/Find.aspx

- http://www.nvidia.com/Download/index.aspx
- http://www.nvidia.com/object/unix.html
- https://www.geforce.com/drivers

Direct links can be found at:

- https://devtalk.nvidia.com/default/topic/533434/linux/current-graphics-driver-releases/
- https://http.download.nvidia.com/XFree86/

Changes in the kernel, Mesa, and Xorg can break driver compatibility, i.e. download the lastest driver before an upgrade.

Boot without nouveau ⚭

Reboot and press E at GRUB2, then add nomodeset and 3 to the group "splash=silent quiet showopts". For example: "splash=silent quiet nomodeset 3 showopts"

nomodeset ensures the nouveau driver (https://nouveau.freedesktop.org/wiki/) won't be loaded.

3 ensures that the system boots to a virtual console (runlevel 3 (https://doc.opensuse.org/documentation/leap/reference /html/book.opensuse.reference/cha.systemd.html#sec.boot.systemd.boot)) instead of to a display manager / X session.

These options will not be saved for the next boot (see **YaST2 -> Boot Loader -> Kernel Parameters** or /etc/default/grub (https://doc.opensuse.org/documentation/leap/reference/html/book.opensuse.reference/cha.grub2.html#vle.grub2.mkconfig)).

Alternatively, you could try booting in IGFX (integrated graphics) mode from BIOS. This requires a VGA connection.

Blacklist nouveau ⚭

The proprietary driver installer will blacklist nouveau; therefore, you can usually skip this section.

The installer creates the file /etc/modprobe.d/nvidia.conf that contains:

```
blacklist nouveau
```

Other versions might also add:

```
options nouveau modeset=0
```

Install the driver ⚭

First, log in as root (the superuser (http://linfo.org/root)).

To view a list of all options:

```
# sh /home/<username>/Downloads/<NVIDIA*.run> -A
```

Normal installation:

```
# sh /home/<username>/Downloads/<NVIDIA*.run>
```

When the installer indicates nouveau might have just been blacklisted, you can continue without rebooting (if you booted as suggested).

Allowing the installer to configure /etc/X11/xorg.conf creates a copy (/etc/X11/xorg.conf.nvidia-xconfig-original) of the current version, which it will restore on uninstallation.

Once the installation is complete, you might need to run:

```
# mkinitrd
```

That will add /etc/modprobe.d/nvidia.conf to the initial ramdisk (https://www.kernel.org/pub/linux/utils/boot/dracut /dracut.html#_definition). To verify, run:

```
# lsinitrd | grep nvidia
```

Restart the computer:

```
# shutdown -r now
```

To check the driver's status:

```
 # hwinfo --gfxcard
 ...
 Driver Info #0:
   Driver Status: nouveau is not active
   Driver Activation Cmd: "modprobe nouveau"
 Driver Info #1:
   Driver Status: nvidia_drm is active
   Driver Activation Cmd: "modprobe nvidia_drm"
 Driver Info #2:
   Driver Status: nvidia is active
   Driver Activation Cmd: "modprobe nvidia"
```

```
# lsmod | grep nvidia
nvidia_drm             49152  1
nvidia_modeset       1097728  4 nvidia_drm
nvidia              14344192  223 nvidia_modeset
ipmi_msghandler        53248  1 nvidia
drm_kms_helper        155648  2 i915,nvidia_drm
drm                   397312  6 i915,drm_kms_helper,nvidia_drm
```

Expert mode ⌗

Alternatively, there is an "expert" installation mode with more questions:

```
# sh /home/<username>/Downloads/<NVIDIA*.run> -e
```

Kernel updates and driver compatibility ⌗

Please note: you will have to repeat these steps whenever you update the kernel. This is necessary because, unlike with the NVIDIA driver from the NVIDIA's openSUSE repos, this driver isn't copied to directories for KMPs which remains valid across kernel updates.

Configuration ⌗

The installer provides two command line utilities:

```
# man nvidia-settings
# man nvidia-xconfig
```

nvidia-settings run without arguments will open a graphical user interface.

Multiple monitors ⌗

Multiple monitors might work without configuration. If not, then see chapters 12-14 of the official documentation (https://download.nvidia.com /XFree86/Linux-x86_64/390.25/README/index.html).

Tearing ⌗

One solution for persistent tearing is:

nvidia-settings -> x server display configuration -> advanced -> force composition pipeline

Warning: Saving this to /etc/X11/xorg.conf via the nvidia-settings GUI might adversely affect loading your desktop.

Alternatively, a simple script can be used to turn it on from the command line. For example:

Create /home/<username>/bin/fcp

```
nvidia-settings --assign CurrentMetaMode="nvidia-auto-select +0+0 { ForceCompositionPipeline = $1 }"
xrefresh
```

Make it executable:

```
chmod +x /home/<username>/bin/fcp
```

Turn it on:

```
fcp 1
```

Turn it off:

```
fcp 0
```

xrefresh can be installed via zypper. It might be needed to avoid a blank screen.

Ctrl+Alt+F1 then Ctrl+Alt+F7 can also help in that situation.

GRUB2 and virtual terminal resolution ⊘

Unlike with nouveau, virtual terminals will be low resolution (https://devtalk.nvidia.com/default/topic/1022918/kernel-mode-setting-does-not-work-in-console-mode-under-linux/).

One way to fix this is with a GRUB2 setting, which will also increase GRUB2's resolution.

yast2 > Boot Loader > Kernel Parameters > Console resolution

Alternatively, to manually achieve the same thing:

From GRUB2 you can press C, and then check the supported resolutions with: vbeinfo

After booting you can edit /etc/default/grub

Change

```
GRUB_GFXMODE="auto"
```

to

```
GRUB_GFXMODE=1920x1080x32
GRUB_GFXPAYLOAD_LINUX=keep
```

Using a resolution you found with vbeinfo. (more on gfxmode (https://www.gnu.org/software/grub/manual/grub/html_node/gfxmode.html))

Then run:

```
# grub2-mkconfig -o /boot/grub2/grub.cfg
```

Plymouth ⊘

Plymouth, the loading screen between GRUB2 and the display manager, might not show.

To see the boot messages instead, uninstall plymouth (or add plymouth.enable=0 (https://www.kernel.org/pub/linux/utils/boot/dracut/dracut.html#_plymouth_boot_splash) to the kernel parameters (https://www.kernel.org/doc/Documentation/admin-guide/kernel-parameters.txt)) and remove the kernel parameters:

```
splash=silent quiet
```

For fewer messages, set the log level:

```
loglevel=5
```

Uninstall and use modesetting or nouveau DDX 🔗

Boot with "nomodeset" and "3", like for installation.

The installer is also used to uninstall:

```
# sh <NVIDIA*.run> --uninstall
```

`/etc/modprobe.d/nvidia.conf` will be deleted.

`/etc/X11/xorg.conf` will be replaced with `/etc/X11/xorg.conf.nvidia-xconfig-original` (a pre-installation copy).

You might need to run `mkinitrd` after uninstalling (more (https://forums.opensuse.org/showthread.php/529709-Is-nouveau-in-the-initial-ramdisk-image)).

The default DDX (https://dri.freedesktop.org/wiki/DDX/) `modesetting` has been provided by the server package since version 1.17.0 in 2015. When package `xf86-video-nouveau` is installed, the `nouveau` DDX should be utilized automagically instead.

See also 🔗

Related articles 🔗

- SDB:NVIDIA drivers
- SDB:NVIDIA using osc

External links 🔗

- The official 390.25 README and Installation Guide (https://download.nvidia.com/XFree86/Linux-x86_64/390.25/README/index.html)
- x11-video-nvidia (http://software.opensuse.org/package/x11-video-nvidia), package that downloads NVidia driver, sets it and DKMS up enabling the automatic update of the nVIDIA proprietary Driver kernel modules upon each kernel upgrade.
- https://nouveau.freedesktop.org/wiki/KernelModeSetting/
- https://en.wikipedia.org/wiki/Mode_setting
- https://en.wikipedia.org/wiki/Direct_Rendering_Manager

Categories: SDB:Hardware │ SDB:HOWTOs │ NVIDIA