

Avoir Nvenc sur Ubuntu 20.04 LTS



Publié le 22 novembre 2022 • Mis à jour le 29 décembre 2022
Accueil • Blog • Brèves • Avoir Nvenc sur Ubuntu 20.04 LTS

Table des matières

Compilation party 😊

Mise en place de Nvenc pour OBS

Si vous avez encore une Ubuntu 20.04 ou inférieure qui traîne, sachez que ce n'est pas bien car la 22.04 LTS existe. Toutefois, cette version de la distribution ne prend pas en charge Nvenc pour OBS. C'est frustrant de faire fonctionner le GPU plein pot alors qu'on a une carte graphique Nvidia qui peut le faire.

Comment faire en sorte que FFmpeg exporte les symboles nécessaires sur l'Ubuntu cosmic/disco (et distributions similaires) afin qu'OBS et MPV puissent utiliser NVENC et NVDEC (anciennement appelé CUVID) sur les architectures Fermi, Maxwell, Kepler, Pascal, Volta et Turing et comment utiliser l'accélération matérielle dans Chromium.

Ce guide vise les cartes les plus modernes compatibles avec NVENC et NVDEC, mais il contient également des instructions pour les cartes utilisant des pilotes hérités (378, 390 et 390).

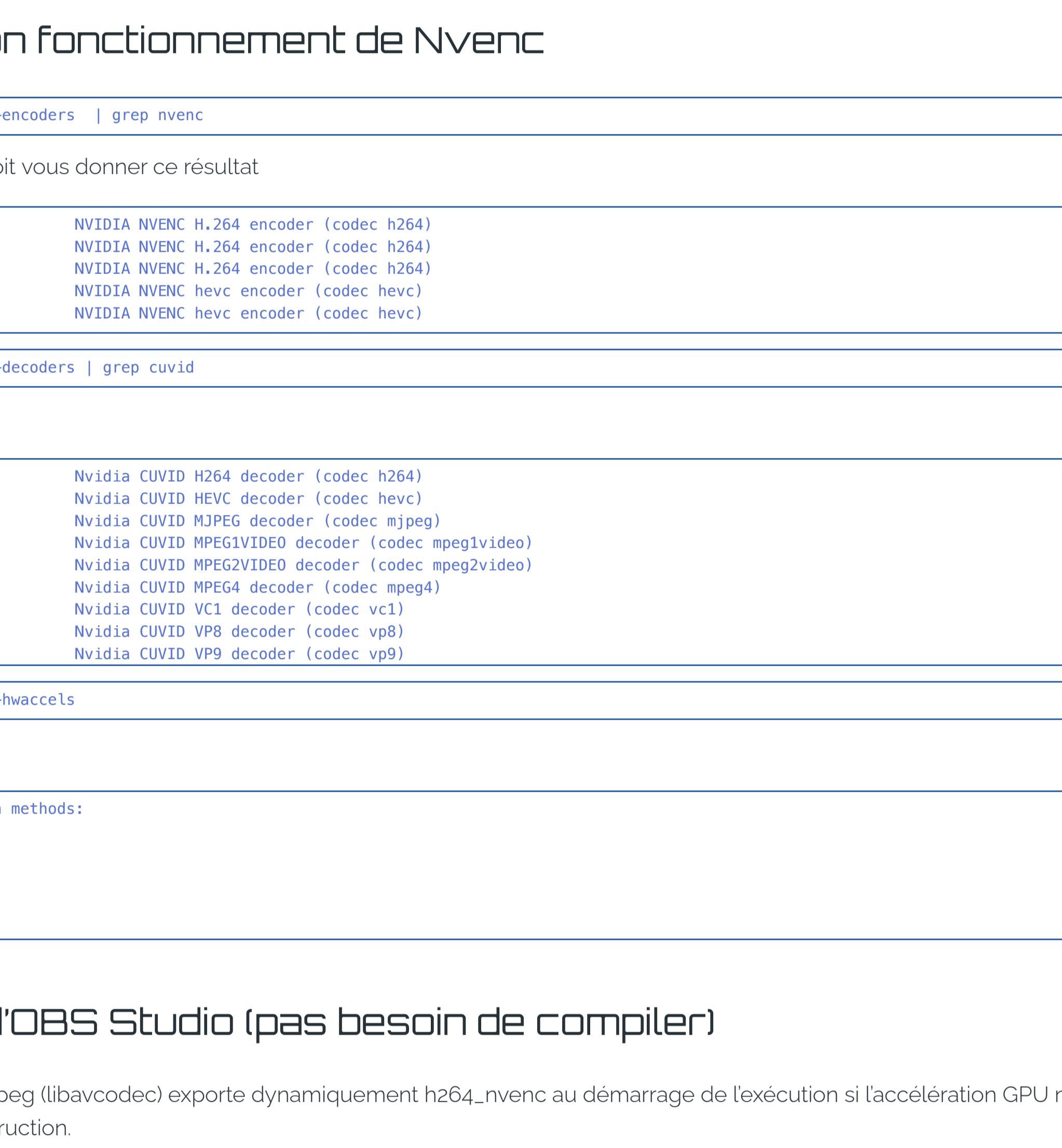
Catégories Nvidia soutenues : <https://developer.nvidia.com/video-encode-decode-gpu-support-matrix>



Installation des pilotes Nvidia

Pour commencer nous allons mettre en place les pilotes Nvidia propriétaire.

```
sudo apt install ppa-purge # Par sécurité
sudo add-apt-repository ppa:graphics-drivers/ppa
# veuillez vérifier le support pour votre GPU - nvidia-driver-(390,396,410,415,418,438,435,440) disponible afin que votre serveur d'affichage n'échoue pas !
# Nous utiliserons 495.44
# Peut-être avez-vous besoin des versions 32 bits de certaines bibliothèques, ajoutez ":i386" au nom du package (par exemple : libnvidia-gl-438:i386)
sudo apt install nvidia-driver-510 xserver-xorg-video-nvidia-510 nvidia-utils-510 nvidia-kernel-common-510 nvidia-dkms-510 nvidia-compute-
```



Installation des paquets nv-codec-headers

On prépare les codecs d'en-tête venant de chez VLC !

```
sudo apt install make git
mkdir ~/devel/v && cd ~/devel/v
git clone https://git.videolan.org/git/ffmpeg/nv-codec-headers.git
cd nv-codec-headers
#change to older SDK branch here if you use a legacy driver – one of: git checkout sdk/8.0,1,2
make && sudo make install
```

La version la plus récente du SDK Video Codec est 9.0.x mais ci-dessous se trouve une liste des anciennes branches SDK pour les pilotes hérités.

```
FFmpeg version of headers required to interface with Nvidia's codec APIs.
Corresponds to Video Codec SDK version 9.1.23.

Minimum required driver versions:
Linux: 435.21 or newer
Windows: 436.15 or newer
```

Liste des branches git pour les anciens pilotes

Si vous utilisez des pilotes hérités, voici les branches pour eux – après avoir cloné le cd du référentiel git et entré l'une des trois lignes suivantes avant l'installation (en entrant make && sudo make install). Vous êtes curieux de savoir ce qui se passera si vous utilisez les derniers en-têtes SDK avec un pilote hérité ? L'éditeur de liens/chargeur ne trouvera pas le point d'entrée ou la fonction (pointeur) ou tenu et votre application échouera.

Toutes ces trois branches prennent en charge les fonctionnalités facultatives de CUDA 10 à partir du pilote 410.48 sous Linux et 411.31 sous Windows.

```
git checkout sdk/8.2 # for Linux 396.24 or newer | for Windows 397.93 or newer
git checkout sdk/8.1 # for Linux 398.25 or newer | for Windows 398.77 or newer
git checkout sdk/8.0 # for Linux 379.39 or newer | for Windows 379.66 or newer
```

Compilation de FFmpeg

FFmpeg détectera automatiquement les en-têtes libavcodec – extrait de /configure --help :

```
The following libraries provide various hardware acceleration features:
--disable-amf           disable AMF video encoding code [autodetect]
--enable-cuda            enable CUDA video decoding code [autodetect]
--enable-cudvid          enable CUDA video acceleration features that require the CUDA SDK [no]
--enable-cuda-ncu         disable Nvidia CUDA support [autodetect]
--enable-cuda-nvenc       disable Nvidia DirectShow video acceleration code [autodetect]
--enable-cuda-nvdec       disable Nvidia DirectShow video decoders [autodetect]
--enable-cuda-nvhw       disable Nvidia NVENC video acceleration code [autodetect]
--enable-cuda-nvhwdec    disable Nvidia NVENC video decoders [autodetect]
--enable-cuda-nvhwenc    disable Nvidia NVENC video encoders [autodetect]
--enable-cuda-nvhwvid    disable Nvidia NVENC video video acceleration code [autodetect]
```

Si la compilation standard ne correspond pas à vos besoins (vous avez besoin de lier des bibliothèques spécifiques/vous n'avez pas besoin de certaines bibliothèques ou vous souhaitez activer/réactiver des fonctionnalités spécifiques), vous pouvez modifier les règles de construction dans ~/devel/ffmpeg/ffmpeg-413/debian/rules. Installez le paquet nv-codec-headers activera libavcode.vdpau_nvenc_nvdec_cuda_cudvid_NVDEC est juste un changement de marque de CUVID. CUDA ne signifie pas le code CUDA-SDK – juste une accélération matérielle via CUDA.

Passons à la compilation et à l'installation de ffmpeg

```
sudo apt build-dep ffmpeg
sudo apt install libavcodec-extra
cd ~/devel/v/ffmpeg
sudo apt source ffmpeg
sudo apt build-dep ffmpeg
# If you get errors like "at least my distro has problems when unpacking source – so we change ownership"
cd ffmpeg-4.2.6 # cd ffmpeg-x.x.x (x.x.x represents the version number)
debbuild -b --no-sign --jobs=try$((`lscpu`/2))
cd ..
rm -rf libavcodec-extra libavfilter-extra*
```

Test du bon fonctionnement de Nvenc

```
ffmpeg -hide_banner -encoders | grep nvenc
```

Cette commande doit vous donner ce résultat

```
V....._nv24_nvenc   NVIDIA NVENC H.264 encoder (codec h264)
V....._nvenc_nvenc NVIDIA NVENC H.264 encoder (codec h264)
V....._nv24_nvdec   NVIDIA NVENC H.264 decoder (codec h264)
V....._nvenc_nvdec NVIDIA NVENC HEVC encoder (codec hevc)
V....._nv24_nvdec   NVIDIA NVENC HEVC encoder (codec hevc)
```

```
ffmpeg -hide_banner -decoders | grep nvdec
```

D'où vous donner :

```
V....._nv24_nvdec   Nvidia CUDA H.264 decoder (codec h264)
V....._nv24_nvdec   Nvidia CUDA HEVC decoder (codec hevc)
V....._nv24_nvdec   Nvidia CUDA MP4VIDEODECODER decoder (codec mpgvideo)
V....._nv24_nvdec   Nvidia CUDA MP4VIDEODECODER decoder (codec mpgevideo)
V....._nv24_nvdec   Nvidia CUDA MP4DECODEC decoder (codec mp3)
V....._nv24_nvdec   Nvidia CUDA VCL decoder (codec vcl)
V....._nv24_nvdec   Nvidia CUDA VP3 decoder (codec vp3)
V....._nv24_nvdec   Nvidia CUDA VP9 decoder (codec vp9)
```

```
ffmpeg -hide_banner -hwaccel
```

D'où vous donner

```
Hardware acceleration methods:
vdpau
cuda
vaapi
drm
cuvid
```

Installation d'OBS Studio (pas besoin de compiler)

OBS détecte si FFmpeg (libavcodec) exporte dynamiquement h264_nvenc au démarrage de l'exécution si l'accélération GPU n'a pas été désactivée au moment de la construction.

Installation d'OBS Studio à partir des paquets officiels

```
sudo add-apt-repository ppa:obsproject/obs-studio
sudo apt update
sudo apt install obs-studio
```

Maintenant, vous devriez avoir un OBS entièrement fonctionnel avec une accélération matérielle ! Le reste de ce guide est facultatif (pour les personnes qui souhaitent tirer le meilleur parti de leur GPU).

Construisez MPV pour utiliser NVDEC pour le décodage vidéo

Compilation et installation avec la prise en charge de Nvdec

```
sudo apt build-dep mpv
mkdir -p ~/devel/mpv
cd ~/devel/mpv
sudo apt source mpv
sudo chown -R $USER: .
cd mpv-0.32.0 # cd mpv-x.x.x (x.x.x represents version)
debbuild -b --no-sign --jobs=try$((`lscpu`/2))
cd ..
rm -rf libavcodec-extra libavfilter-extra*
```

Puis vous pouvez lire la vidéo avec mpv

```
mpv --demux=raw-v4l2
Playing: Sons_Home_Movie.mkv
(+/-) Video: --vid=0 (+) (h264 1280x720 29.97fps)
(+/-) Audio: --aid=1 (+) (aac 2ch 44100Hz)
Info: [mpv] [mpv] This line is very important, you can turn on debug output with --v
AO: [pulse] 44100Hz stereo 2ch float
VD: [gpu] 1280x720 => 1710x720 cuda[nv12]
```

Pour conclure

Cela fait des années que j'utilise cette méthode pour obtenir nvenc sur Ubuntu et OBS Studio. Ceci n'est plus nécessaire sur Ubuntu 22.04, nvenc étant pris en charge nativement. De quoi faire de beau enregistrement pour DonF & 6BORG

[Source](#)

Informations sur l'article

Rédaction

Ecrit par Gaëtan LEGAY le 22 novembre 2022 à 16:33

Mis à jour le 29 décembre 2022 à 00:18

Catégories & Étiquettes

Brèves, Hors Séries

Nvidia, OBS Studio, Ubuntu



Gaëtan LEGAY

20 ans de Linux et de Web • Créeur Web & App, One Page, Vitrine & E-commerce • D'applications PWA pour Android / iOS / Windows / MacOS & Linux • Team à DonF

Partage

[Facebook](#) [Twitter](#) [LinkedIn](#) [Telegram](#) [WhatsApp](#) [Email](#)

Commentaires

S'abonner ▾

Connexion

0 COMMENTAIRES

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

61

62

63

64

65

66

67

68

69

70