





ffmpeg: quelques trucs utiles

- Création par :  [lagrenouille](#)
- Objet : du tuto ffmpeg
- Niveau requis :  [débutant](#), [avisé](#)
- Commentaires : *Contexte d'utilisation du sujet du tuto.* 
- Débutant, à savoir : [Utiliser GNU/Linux en ligne de commande](#), tout commence là !. 
- Suivi :
[à](#), [tester](#), [à-placer](#)

Introduction

cet article est un condensé de nombreuses commandes que j'ai testé ou utilisés. Il faut parfois adapter.

Installation

```
apt install ffmpeg libavcodec-extra libavcodec-extra-53 imagemagick  
oggvideotools oggvideotools-dbg vorbis-tools flac mpv
```

Outils

ffmpeg : outil de traitement

```
ffprobe : outils d'analyse  
ffplay : lecteur vidéo
```

-pix_fmt (pixel format) - Défini comment sont encodé les pixels, c'est ce qui va permettre de rendre notre vidéo plus légère et davantage compatible avec les lecteurs vidéos existants. Pour ce faire le format "grand public" recommandé est yuv420p qui permet le sous-échantillonnage de la chrominance (plus d'infos sur la page wikipédia yuv)

-crf (constant rate factor) - Cet argument va nous permettre de moduler facilement le ratio poid/qualité. Plus on donne un nombre proche de zéro plus le fichier est proche de sa source et donc lourd. En général en fonction du besoin on attribue des valeurs entre 15 et 25:

```
0 = (Lossless) super lourd, identique à la source.  
10 = lourd, grande qualité  
30 = léger, mauvaise qualité
```

-preset - Détermine la vitesse d'encodage, une vitesse plus lente permet au logiciel d'optimiser le fichier pour gagner encore de la place, les valeurs disponibles sont :

```
ultrafast
superfast
veryfast
faster
fast
medium (default)
slow
slower
veryslow
```

Avant de commencer, il semble judicieux de connaître quel format est capable de gérer FFmpeg.

Pour cela, ouvrez un terminal et saisissez la commande suivante :

\$

```
ffmpeg -formats -hide-banner
```

la réponse peut être assez fournie.

Pour avoir uniquement la liste des codecs que sait utiliser FFmpeg, saisissez les commandes suivante :

```
ffmpeg -codecs
```

```
ffmpeg -filters
```

Pour rechercher les informations le concernant votre vidéo.

```
ffmpeg -i votre-vidéo.mp4
```

Identifier le format d'un fichier multimedia (audio ou video)

```
ffmpeg -i votrefichier 2>&1 | grep -E "Seems|Input|Duration:|Stream"
```

Pour faire un diaporam avec ffmpeg, voir l'article ici :

https://debian-facile.org/utilisateurs:lagrenouille:tutos:ffmpeg-faire-un-diaporama_musical

commandes diverses

****commandes trouvées dans des docs, ffmpeg ou autres, testés pour mes besoins****

Si j'ai des vidéos musicales webm, et je veux un truc plus léger, sans images

```
ffmpeg -i Marlene-Dietrich-Lili_Marleen.webm Marlene-Dietrich-
```

```
Lili_Marleen.mp3
```

si je veux une vidéo mp4 à la place du webm

```
ffmpeg -i Marlene-Dietrich-Lili_Marleen.webm Marlene-Dietrich-Lili_Marleen.mp4
```

Pour redimensionner une vidéo :

```
ffmpeg -i input.mp4 -vf scale=640:360 output.mp4
```

Pour extraire une partie de votre vidéo. “-ss 00:00:30” indique le moment où vous souhaitez récupérer la séquence vidéo. “-t 00:00:30” indique la durée de la vidéo de sortie, nommée “output.mp4”

```
ffmpeg -i input.mp4 -ss 00:00:30 -t 00:00:30 output.mp4
```

Pour fusionner des vidéos, il faut faire avant dans le même répertoire, un fichier texte avec vim ou nano ou autre.

[nom.txt](#)

```
nano join.txt  
file '/chemin/vers/vidéo1.mp4  
file '/chemin/vers/vidéo2.mp4  
file '/chemin/vers/vidéo3.mp4
```

maintenant on peut fusionner.

```
ffmpeg -f concat -safe 0 -i join.txt -c copy output.mp4
```

Compresser une vidéo, est souvent utile pour mettre en ligne, et ainsi alléger le poids du fichier, il existe une multitude de façon de coder cette conversion, j'en mettrai que deux ou trois que j'ai testé.

```
ffmpeg -i mavidéo.mp4 -vf scale=1280:-1 -c:v libx264 -preset veryslow -crf 24 manouvellevidéo.mp4
```

autre compression

```
ffmpeg -i input.mp4 -c:v libx264 -crf 23 -c:a aac -b:a 128k output.mp4
```

Dans cette deuxième commande, “-crf 23” est utilisé pour contrôler la qualité de la vidéo compressée à l'aide de la méthode Constant Rate Factor (CRF). Une valeur CRF plus faible (par exemple 18) produira une meilleure qualité mais un fichier plus volumineux, tandis qu'une valeur plus élevée (par exemple 28) produira une qualité plus faible mais un fichier plus petit. Dans ce cas, une valeur de 23 est utilisée. En outre, “-b:a 128k” fixe le débit audio à 128 kbps (kilobits par seconde). Cela permet de contrôler la quantité de données audio utilisées pour représenter le son. Des débits binaires plus élevés offrent généralement une meilleure qualité audio, mais peuvent augmenter la taille du fichier.

Extraire des images d'une vidéo .

"-r 1" signifie que vous allez extraire une image par seconde avec le nom suivant "image-%2d.png".
(image -%2d signifie que les images seront nommées imageX. X étant un nombre de 0 à 99)

```
ffmpeg -i input.mp4 -r 1 -f image2 image-%2d.png
```

Ajouter une image de couverture à une vidéo, changez couverture.jpg par le nom qui vous va bien.

```
ffmpeg -i input.mp4 -i couverture.jpg -c copy -map 0 -map 1 output.mp4
```

Convertir une vidéo en noir et blanc.

```
ffmpeg -i mavidéo.mp4 -vf "colorchannelmixer=.3 :.4 :.3:0 :.3 :.4 :.3:0 :.3 :.4 :.3" mavidéo_N&B.mp4
```

Pour sous-titrer vos vidéos, allez lire ce site très bien fait:

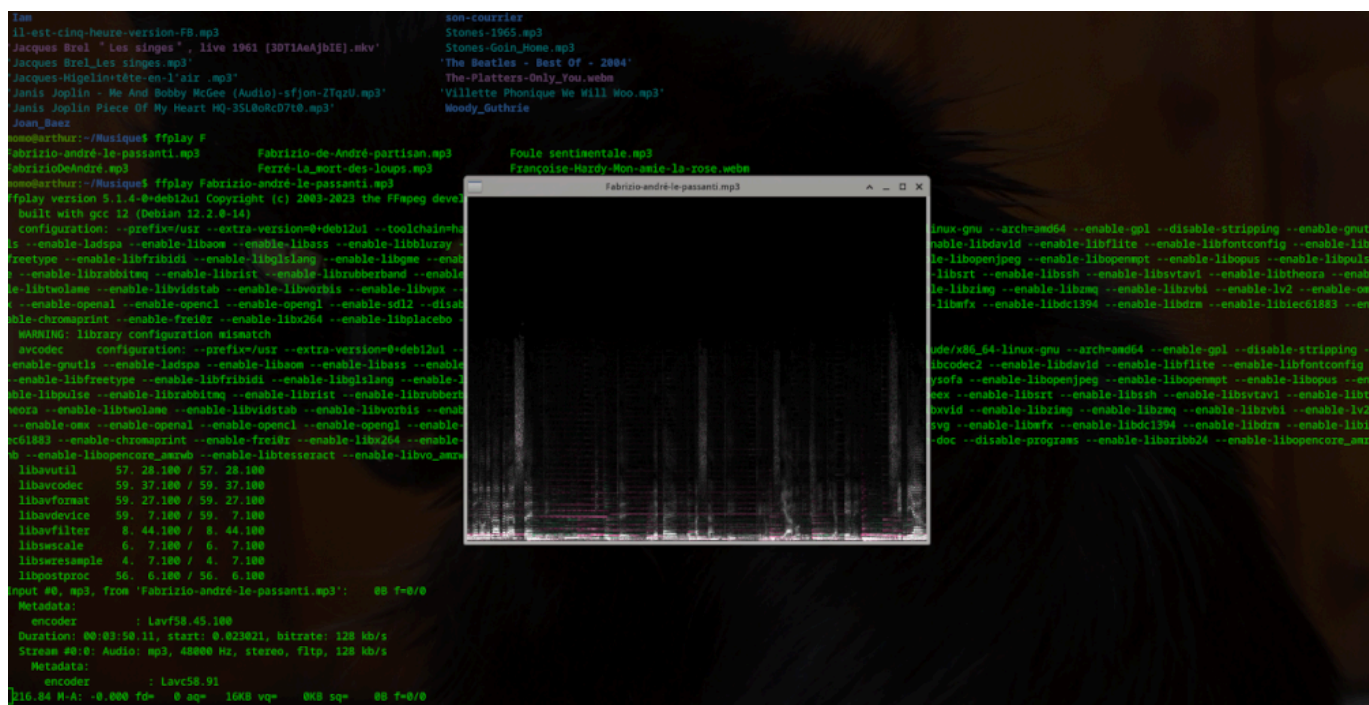
<https://fr.abysale.com/generate-video/how-to-add-subtitles-to-videos-with-ffmpeg>

LA COMMANDE FFPLAY

ffplay est un lecteur multimédia très simple et portable utilisant les bibliothèques FFmpeg .

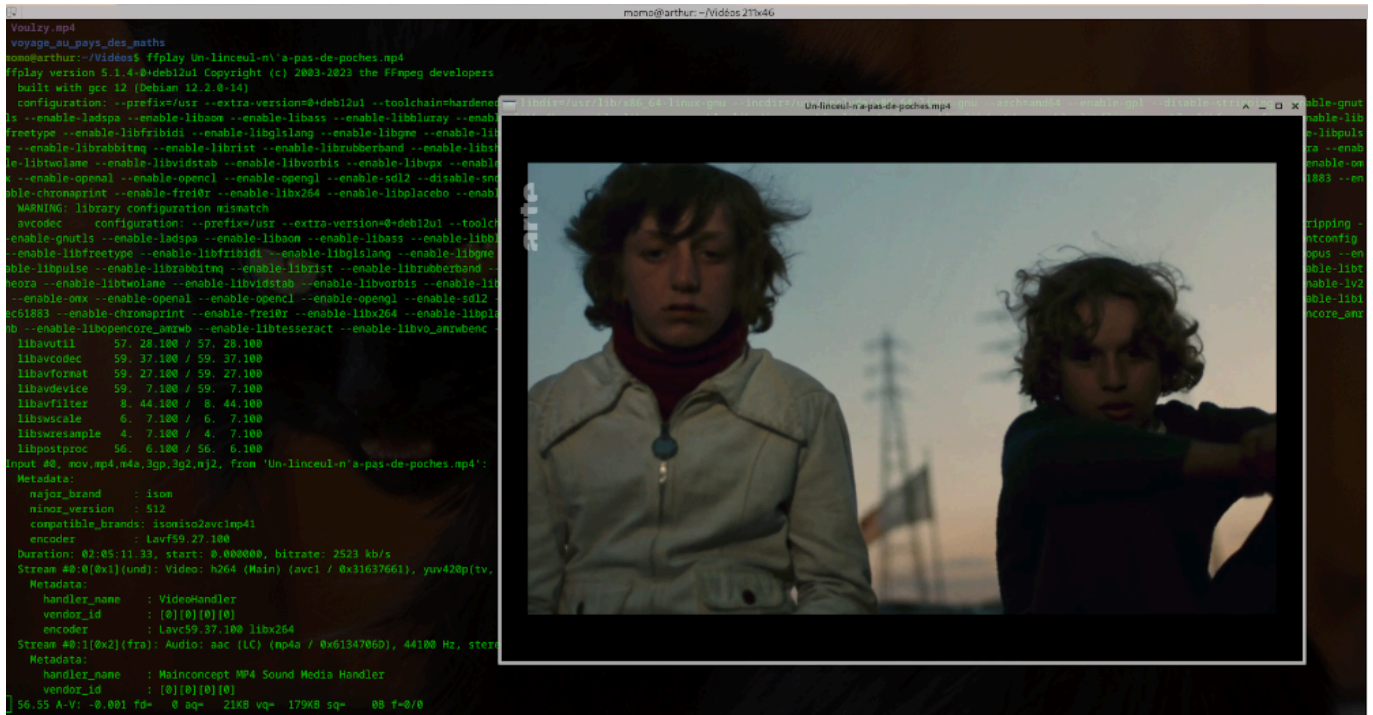
MUSIQUE

```
ffmpeg Fabrizio-andré-le-passanti.mp3
```



VIDÉO

ffplay Un-linceul-n\'a-pas-de-poches.mp4



Pour exécuter plusieurs vidéos ou musiques les unes à la suite des autres, placez vous dans un repertoire avec plusieurs vidéos ou plusieurs musiques.

```
cd /le_chemin_de_votre_repertoire_contenant_les_fichiers_a_lire
```

```
for f in *.wav ; do ffplay -hide_banner -nodisp -autoexit "$f"; done
```

```
for f in *.mp4 ; do ffplay -hide_banner -nodisp -autoexit "$f"; done
```

Ctrl + C pour passer au morceau suivant

Ctrl + Z pour arrêter

Pour ne pas afficher la fenêtre du graph audio, supprimez dans la commande l'option -nodisp.

Voilà, il existe probablement bien d'autres options, je vous laisse les découvrir.

From:

<http://debian-facile.org/> - **Documentation - Wiki**

Permanent link:

<http://debian-facile.org/utilisateurs:lagrenouille:tutos:ffmpeg:quelques-trucs-utiles>

Last update: **04/05/2024 09:06**

