

Von Franz Mertens

Medien Downloads und
Weiterverarbeitung mit
Youtube-dl / yt-dlp und
Ffmpeg

<https://www.transform2agile.de>

Voraussetzung

- Aktuelles Linux System
- Python 3
- youtube-dl aus Repository oder von <https://www.youtube-dl.org>
- Ffmpeg

youtube-dl – Erste Schritte

- Die Software wird über Terminal ausgeführt
- Ob die Installation funktioniert, sieht man z.B. wenn man `youtube-dl -U` eingibt.
- Ist die Installation aus dem Repository, kommt der Hinweis, dass man auf das Repository angewiesen ist
- Bei einer Installation über die Links der Homepage kommt entweder eine Information zu einer neuen Version oder, dass alles aktuell ist.

Beispielmeldung bei aktueller Installation ohne Repository

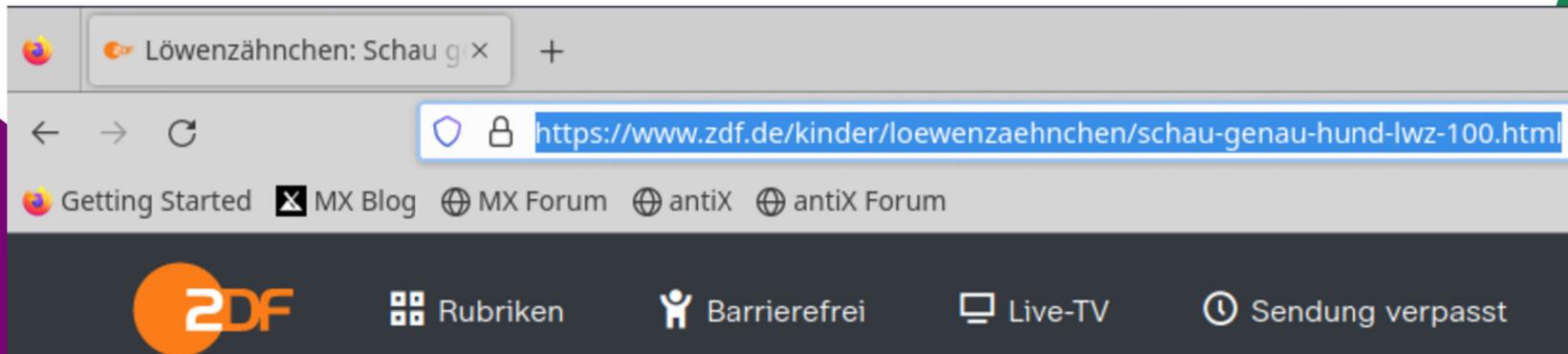
```
Terminal
Datei Bearbeiten Ansicht Terminal Reiter Hilfe
(base) franz@mx:~
$ youtube-dl -U
youtube-dl is up-to-date (2021.12.17)
(base) franz@mx:~
$
```

Meldung bei Installation über das Repository

```
franz@debbie:~/Videos$ youtube-dl -U  
It looks like you installed youtube-dl with a package manager, pip, setup.py or  
a tarball. Please use that to update.
```

Erster Download – einfaches Video

- Wechsele in das Zielverzeichnis
- Syntax für den Anfang:
youtube-dl <Medienlink>
- Den Videolink nehmen wir aus der Adressleiste
- <https://www.zdf.de/kinder/loewenzaehnchen/hamster-104.html>



Reales Beispiel

Wir wechseln zuerst in unser Zielverzeichnis

```
(base) franz@mx:~  
$ cd /home/franz/Videos
```

Nun geben wir youtube-dl gefolgt vom Link ein.

```
(base) franz@mx:~/Videos  
$ youtube-dl https://www.zdf.de/kinder/loewenzaehnchen/schau-genau-hund-lwz-100.html  
[ZDF] schau-genau-hund-lwz-100: Downloading webpage  
[ZDF] schau-genau-hund-lwz-100: Downloading JSON content  
[ZDF] schau-genau-hund-lwz-100: Downloading JSON metadata  
[ZDF] 180608_schau_genau_hund_lwz: Downloading m3u8 information  
[ZDF] 180608_schau_genau_hund_lwz: Downloading m3u8 information  
[hlsnative] Downloading m3u8 manifest  
[hlsnative] Total fragments: 21  
[download] Destination: Schau genau - Hund-180608_schau_genau_hund_lwz.mp4  
[download] 100% of 58.06MiB in 00:06  
[ffmpeg] Fixing malformed AAC bitstream in "Schau genau - Hund-180608_schau_genau_hund_lwz.mp4"  
(base) franz@mx:~/Videos
```

Beispiel 1

- Video 1 zeigt den Download

Probe

Nun die Probe was sehen wir im Zielverzeichnis?

```
[Franz@mx:~/Videos]$ dir  
Schau\ genau\ -\ Hund-180608_schau_genau_hund_lwz.mp4
```

Ausgabeformat definieren bei Videos

- Nun liegt hier die Datei schon im mp4 Format vor
- Wenn ich die Datei in einem anderen Format haben möchte und diese in einem anderen Format vorliegt gibt es die Option -f "Videoformat"
- `youtube-dl -F <Link>` zeigt die verfügbaren Formate an

```
$ youtube-dl -F https://www.zdf.de/kinder/loewenzaehnen/schau-genau-hund-lwz-100.html
```

- Für eine mp4 Datei sieht der Download so aus:

```
(base) franz@mx:~/Videos
$ youtube-dl -f "mp4" https://www.zdf.de/kinder/loewenzaehnen/schau-genau-hund-lwz-100.html
[795] schau-genau-hund-lwz-100: Downloading video
[mp4] Fixing metadata for video in 'schau-genau-hund-lwz-100.mp4'
(base) franz@mx:~/Videos
$ dir
Schau\ genau\ -\ Hund-180608_schau_genau_hund_lwz.mp4
```

Beispiel 2

- Video 2 zeigen

Nur Tonspur extrahieren

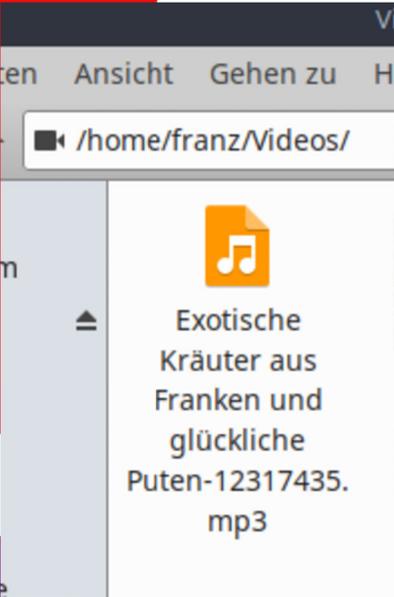
- Manchmal möchte man nur die Tonspur, z.B. um Musik eines Konzerts auf einem Mp3 Player mitzunehmen oder um Geschichten für die Kinder im Auto zu haben
- -x zeigt an, dass Audio extrahiert werden soll
- --audio-format "mp3" zeigt an in welchem Format die Audiodatei abgelegt werden soll.
- --audio-format funktioniert nur in Kombination mit -X
- Folgende Audioformate werden unterstützt: "best", "aac", "flac", "mp3", "m4a", "opus", "vorbis", "wav"

```
(base) Franz@mx:~/Videos  
$ youtube-dl -x --audio-format "mp3" https://www.ardmediathek.de/video/Y3JpZDovL2JyLmRlL3ZpZGVvLzNlYzk1Y2NkLWRiNmYtNDRiMS04NDNiLTJiNDQxZTNiYjczMg
```

Beispiel 3

- Video 3

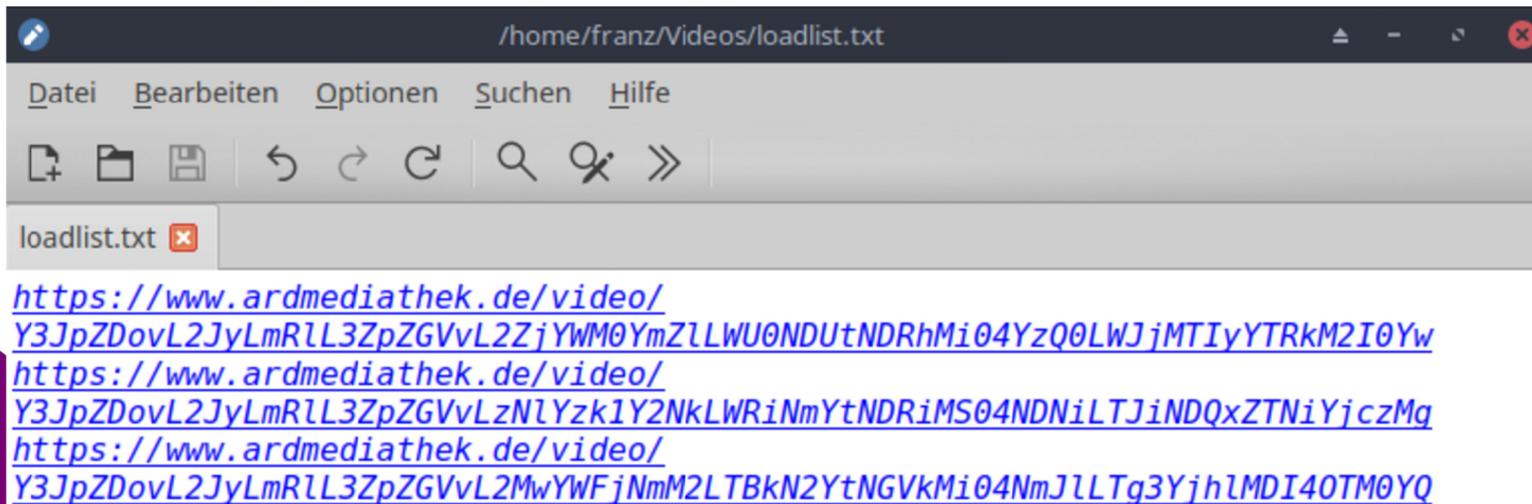
Probe ob es geklappt hat



```
(base) franz@mx:~/Videos
$ dir
Exotische\ Kräuter\ aus\ Franken\ und\ glückliche\ Puten-12317435.mp3
(base) franz@mx:~/Videos
```

Mehrere Videos am Stück downloaden

- Textdatei mit den Links anlegen und auf diese Datei verweisen -a, --batch-file FILE
- Pro Zeile ein Link und keine sonstigen Zeichen



The screenshot shows a text editor window titled "/home/franz/Videos/loadlist.txt". The window has a menu bar with "Datei", "Bearbeiten", "Optionen", "Suchen", and "Hilfe". Below the menu bar is a toolbar with icons for file operations (new, open, save, undo, redo, refresh, search, cut, copy, paste) and a right-pointing arrow. The main text area contains three lines of blue, underlined URLs from armediathek.de:

```
https://www.ardmediathek.de/video/  
Y3JpZDovL2JyLmRlL3pZGVvL2ZjYWm0YmZlLWU0NDUtNDRhMi04YzQ0LWJjMTIyYTRkM2I0Yw  
https://www.ardmediathek.de/video/  
Y3JpZDovL2JyLmRlL3pZGVvLzNlYzk1Y2NkLWRiNmYtNDRiMS04NDNiLTJiNDQxZTNiYjczMg  
https://www.ardmediathek.de/video/  
Y3JpZDovL2JyLmRlL3pZGVvL2MwYWYfjNmM2LTBkn2YtNGVvMi04NmJlLTg3YjhLMDI4OTM0YQ
```

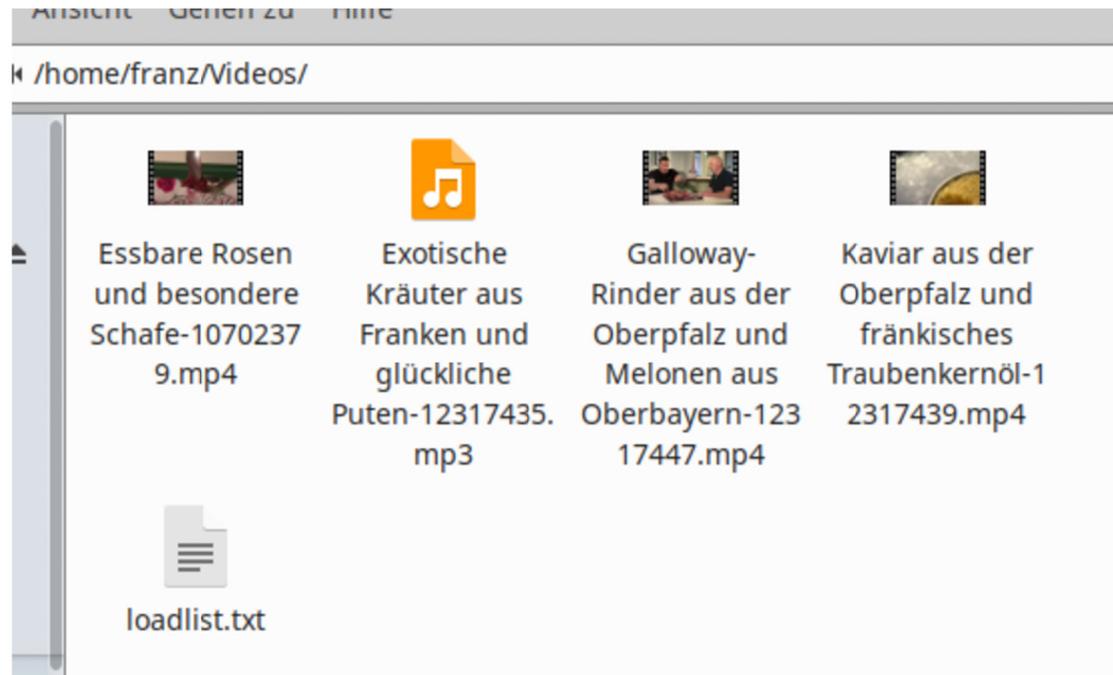
Befehl für den Download einer Liste

```
(base) franz@mx:~/Videos  
$ youtube-dl -a loadlist.txt -f "mp4"
```

Beispiel 4

- Video 4 zeigen

Überprüfung des Downloads



yt-dlp

- Fork von youtube-dl
- Aufgrund von Änderungen bei Youtube Headern funktioniert bei Youtube aktuell yt-dlp besser als youtube-dl, da dort die Patches bereits eingerichtet wurden
- Link zur Installationsanleitung und Dokumentation:
- <https://github.com/yt-dlp/yt-dlp/wiki/Installation>
- <https://gist.github.com/DiegoFleitas/c940d4b94d6b92b55a7084afe84bf571>

yt-dlp

- Download einer Youtube Playlist
- Bitte keine Shortlinks verwenden!

```
yt-dlp -x --audio-format "mp3"  
https://www.youtube.com/watch?  
v=oP5PIYrymHY&list=PLRLpZ0b4vWpF7rXZ-TFCTo6XIPA_Yop9B  
--yes-playlist
```

```
yt-dlp -x --audio-format "mp3" https://www.youtube.com/watch?  
v=oP5PIYrymHY&list=PLRLpZ0b4vWpF7rXZ-TFCTo6XIPA_Yop9B --yes-  
playlist --playlist-start 2
```

Beispiel 5

- Video5 zeigen

Video in Einzelbilder zerlegen

```
ffmpeg -i video.mov -vf fps=1/2 -s 604x353 image_%05d.jpg
```

<https://thomaskekeisen.de/de/blog/ffmpeg-videos-bilder-frames-jpg-png/>

Ende

- Ich wünsche viel Vergnügen bei den weiteren Vorträgen heute.