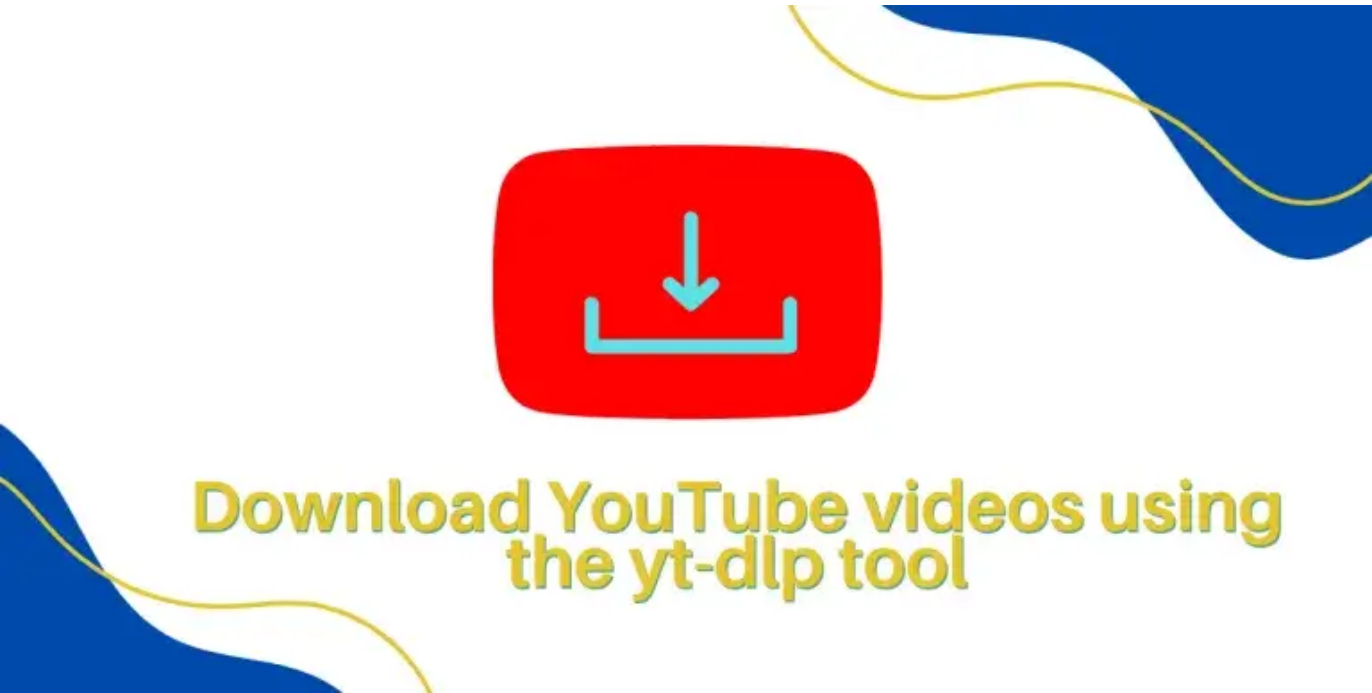




# yt-dlp управление



## Опции

Настройки: ключей. Полный список в [реозитории](#).

Name	Short	Description
<code>--skip-download</code>	`	Полезный ключ, для тестирования набора ключей.
<code>--list-formats</code>	<code>-F</code>	Отобразить список доступных форматов медиа фалов.
<code>--format</code>	<code>-f</code>	Указать необходимый формат ролика
<code>--write-link</code>		Сохранит ссылку на источник

Со сторонних сайтов искать плейлисты в формате `m3u8`.

## Форматы видео

Установка формата медиа файла производится ключом `-f` или `--format`.

Как я понял, самый универсальный формат это **mp4**. Так как он понимается наибольшим числом плееров.

Выбрать лучший **mp4** формат.

```
-f 22/bestvideo[ext=mp4]+bestaudio[ext=m4a]/best[ext=mp4]/best
```

Так же хорош формат **webm**, позволяет сильно сжимать файлы. Правда сам контейнер чуть менее понятен для некоторых проигрывателей. Пока имеет смысл видео формата **webm** упаковывать в **mp4** контейнер.

```
-f bestvideo[height<=720]+bestaudio[ext=webm] --merge-output-format mp4
```

Примеры установки форматов видео в репозитории.

А так же, что при скачивании с **youtube** не обязательно сохранять номер видеоролика. Достаточно названия файла и его расширения.

## Одиночное видео

```
film='link'
yt-dlp -o '%(title)s.%(ext)s' --format
"bestvideo+bestaudio[ext=m4a]/bestvideo+bestaudio/best" --merge-output-format mp4
$film
```

Пример с машей и медведем.

```
yt-dlp -o '%(title)s.%(ext)s' --yes-playlist https://youtube.com/playlist?
list=PLXnIohISHNIur5SkRfv0Lo1YJjw7NwQx6
```

Команды которые я часто использую.

Загрузка видеороликов из списка в файле

```
yt-dlp -f best -a list.txt
```

```
yt-dlp -ct --simulate --batch-file='/path/to/batch-file.txt'
```

## Плейлисты

Программа поддерживает автоматическое скачивание плейлистов, с youtube во всяком случае. И при передаче ссылки на него начнет скачивать все ролики.

Для плейлиста, иногда имеет смысл скачивать так же и номер ролика. Делается это добавлением параметра:

```
-o '%(playlist_index)s.%(title)s.%(ext)s'
```

Полезная опция при скачивании плейлистов Установка начала и конца плейлиста, или шаг для скачки. Опция похожа на слайсы в питоне.

```
-I, --playlist-items ITEM_SPEC Comma separated playlist_index of the items to download. You can specify a range using "[START]:[STOP][:STEP]". For backward compatibility, START-STOP is also supported. Use negative indices to count from the right and negative STEP to download in reverse order. E.g. "-I 1:3,7,-5::2" used on a playlist of size 15 will download the items at index 1,2,3,7,11,13,15
```

Если работа со слайсами не понятна, можно делать по старинке

```
--playlist-start NUMBER      -I NUMBER:  
--playlist-end NUMBER        -I :NUMBER
```

## Скачивание аудио

С кодеком Youtube VP9 этот файл, скорее всего, будет расширение `.webm` или `.opus`.

Первый спороб это скачать нужный формат. Выбрав его после получения всего списка форматов `-F`.

Предположим аудиострим находится под ID 2

```
yt-dlp -f 2 -o 'songname.%(ext)s'
```

Второй способ, доверится автоматическому определению. Выбрав опцию `-x`

```
yt-dlp -f 'ba' -x --audio-format mp3 https://www.youtube.com/watch?v=dQw4w9WgXcQ  
-o '%(id)s.%(ext)s'
```

## Обложка

### Сохранение обложки

Изображения заставки. Изображения скачиваются флагом `--write-thumbnail` Для сохранения изображения определенным именем используется шаблон сохранения

```
yt-dlp --write-thumbnail --convert-thumbnails jpg
```

где:

- `--convert-thumbnails jpg` - конвертировать изображение в формат `jpg`

Путь для сохранения

```
--write-thumbnail -o "thumbnail:%(title)s/folder.%(ext)s"
```

### Вставка обложки в файл

Вставить обложку в файл можно ключем `--embed-thumbnail`

- Поддерживаются контейнеры `mp4`, `mkv`.
- `webm` вставку обложки не поддерживает.

Для вставки изображений в файлы `yt-dlp` требует модуль `mutagen`

```
python3 -m pip install mutagen
```



```
sudo pacman -S python-mutagen
```

## Субтитры

Опции субтитров:

<code>--write-sub</code>	Write subtitle <code>file</code>
<code>--write-auto-sub</code>	Write automatic subtitle <code>file</code> (YouTube only)
<code>--all-sub</code>	Download all the available subtitles of the video
<code>--list-sub</code>	List all available subtitles <code>for</code> the video
<code>--sub-format FORMAT</code>	Subtitle format, accepts formats preference, <code>for</code> example: <code>"srt"</code> or <code>"ass/srt/best"</code>
<code>--sub-lang LANGS</code>	Languages of the subtitles to download (optional) separated by commas, use IETF language tags like <code>'en,pt'</code>

## Примеры загрузки субтитров

Например, чтобы перечислить все субтитры для видео:

```
yt-dlp --list-sub https://www.youtube.com/watch?v=Ye8mB6VsUHW
```

Чтобы загрузить все субтитры, но не видео:

```
yt-dlp --all-sub --skip-download https://www.youtube.com/watch?v=Ye8mB6VsUHW
```

Если видео содержит только автоматически сгенерированные субтитры, то `--all-sub` всё равно не загрузит их, вместо этого используйте:

```
yt-dlp --write-auto-sub --skip-download https://www.youtube.com/watch?v=Ye8mB6VsUHW
```

Скачать `en` субтитры

```
yt-dlp --write-sub --sub-lang en --skip-download "URL"
```

или автогенерировать субтитры `en`

```
yt-dlp --write-auto-sub --skip-download --sub-lang en "URL"
```

## Скачивание с закрытых сервисов

Если нужно скачать видеоролики с закрытых ресурсов. То есть тех, которые доступны только по логин-пароллю. Можно использовать ключи `yt-dlp` есть `--username` и `--password`.

Но я сталывался с той проблемой, что из за смены `api` сервиса команды `--username` и `--password` ломались и в консоль выбрасывалась ошибка. Короче универсальный метот это использовать `cookies` браузера. Этот метот срабатывал у меня всегда. Единственный минус этого метода, что перед началом скачивания нужно залогиниться на нужную страницу, с браузера.

Для пользователей `firefox`:

- логинимся на страницу.
- заходим по ссылке на необходимое видео или плейлист (чтобы убедиться что он открыт для пользователя)
- в консоли вбиваем

```
yt-dlp --cookies-from-browser firefox "url_for_video_or_playlist"
```

Для браузера `chrome` предварительно нужно сделать небольшой танец с бубном. А именно разрешить браузеру `chrome` импортировать `cookies` сторонними программами.

Используем ключ `--disable-features=LockProfileCookieDatabase`

То есть действие выглядит так:

- Запускаем хром через `cmd` (клавиши `Win` + `R`) вставив строку ниже:

```
"C:\Program Files\Chrome\chrome.exe" --disable-features=LockProfileCookieDatabase
```

- Далее все по списку: Логинимся на страницу, проверяем доступность видосов.
- После открываем консоль и вбиваем.

```
yt-dlp --cookies-from-browser chrome "url_for_video_or_playlist"
```

## Сторонние сервисы

Скачивание видео со сторонних сервисов

### Скачивание видео с сервиса vk.com

Для скачивания общедоступных одиночных роликов `vk.com` команда такая же как и для `youtube.com`

*# одиночное видео*

```
yt-dlp "https://vk.com/video-123456789_123456789"
```

*# плейлист*

```
yt-dlp "https://vk.com/video/playlist/-123456789_12"
```

Если необходимо скачать **закрытые** видео с сервиса вконтакте. Или же вы хотите скачать **приватные** плейлисты роликов, доступные только за донаты или по подписке, то необходимо использовать `cookies` из браузера.

В настройках `yt-dlp` есть ключи `--username` и `--password`. Но из-за частой смены `api` сервисом **вконтакте**, они часто не работают. А вот импортирование `cookies` - срабатывает почти всегда.

Итак наши действия:

1. Залогиниться через браузер в `https://vk.com` к примеру это будет браузер `firefox`.
2. Перейти на страницу приватного видео или плейлиста, убедившись, что видео проигрывается в браузере, а плейлист отображается полностью. Предположим адрес приватного плейлиста такой: `https://vk.com/video/playlist/-123456789_12`
3. В командной строке использовать ключ `--cookies-from-browser firefox`

Пример в командной строке будет выглядеть так:

```
yt-dlp -f "bestvideo[ext=mp4]" --cookies-from-browser firefox  
"https://vk.com/video/playlist/-123456789_12"
```

# Файл конфигурации

В случае регулярного использования команд можно задать файл конфигурации `yt-dlp.conf`



## MY TIP

Use this awesome feature option

Пример для скачивания `youtube` роликов через сервис `colab`

yt-dlp.conf

```
# Lines starting with # are comments
# Formats

# Best "mp4"
-f 22/bestvideo[ext=mp4]+bestaudio[ext=m4a]/best[ext=mp4]/best

# Template folder
-P "./videos/"
# Save all videos under YouTube directory in your home directory
-o %(title)s/%(title)s.%(ext)s

# Embed datas to video
--embed-chapters
--embed-thumbnail
--add-metadata

# Convert thumbnails to jpg
--convert-thumbnails "jpg"

# Save thumbnail with 'folder.jpg' name
--write-thumbnail -o "thumbnail:%(title)s/folder.%(ext)s"

# links
# - [yt-dlp - pena](https://github.com/yt-dlp/yt-dlp)
# - [yt-dlp - опции](https://github.com/yt-dlp/yt-dlp#usage-and-options)
# - [yt-dlp - примеры](https://github.com/yt-dlp/yt-dlp#output-template-examples)
```

## links



- [yt-dlp](#) - репа
- [yt-dlp](#) - опции
- [yt-dlp](#) - примеры
- Коллекция [скриптов](#) для yt-dlp
- [плагин](#) для jellyfin

## Обновление

Через `wget`

```
wget https://github.com/yt-dlp/yt-dlp/releases/latest/download/yt-dlp -O  
~/.local/bin/yt-dlp  
chmod a+rx ~/.local/bin/yt-dlp # Make executable
```

Через `curl`

```
curl -L https://github.com/yt-dlp/yt-dlp/releases/latest/download/yt-dlp -o  
~/.local/bin/yt-dlp  
chmod a+rx ~/.local/bin/yt-dlp # Make executable
```

Теги: `yt-dlp` `youtube-dl`

 [Отредактировать эту страницу](#)

Последнее обновление **30 июл. 2024 г.** от **Stavis**