

[Home](#)

[Open Source](#)

[Technology](#)

[Linux](#)

[Unix](#)

[Free Videos and EBooks](#)

[Donate](#)



AUDIO/VIDEO

FOLLOW



[Backup tools](#)

[Cloud](#)

[Database](#)

[Directory servers](#)

[Mobile](#)

[Reviews](#)

[Security](#)

[Tips and Tricks](#)

[Trouble shooting](#)

[Virtualization](#)

20 FFmpeg

Commands

For

Beginners

BY [SK](#) · PUBLISHED MAY 10,
2017 · UPDATED OCTOBER 27,
2017



In this guide, I will be explaining how to use FFmpeg multimedia framework to do various audio and video transcoding, conversion operations with examples. Just in case, you haven't installed FFmpeg already, refer the following guide.

■ **Install FFmpeg in Linux**

The typical syntax of the FFmpeg command is:

```
ffmpeg [global_o  
        {[output_file_o
```

We are now going to see some important and useful FFmpeg commands.

1. Getting audio/video

TO SEARCH, TYPE AND HIT ENTER



FREE EBOOK OF THE WEEK



file information

To display your media file details, run:

```
ffmpeg -i video.
```

Sample output:

```
ffmpeg version 3
built with gcc
configuration:
libavutil 55. 5
libavcodec 57.
libavformat 57.
libavdevice 57.
libavfilter 6.
libavresample 3
libswscale 4. 6
libswresample 2
libpostproc 54.
Input #0, mov,mp
Metadata:
major_brand : i
minor_version :
compatible_bran
encoder : Lavf5
Duration: 00:43
Stream #0:0(und
Metadata:
handler_name :
Stream #0:1(und
Metadata:
```



FOLLOW US ON FACEBOOK

[Follow us on Facebook](#)

FOLLOW US ON GOOGLE PLUS

```
handler_name :
At least one out
```

As you see in the above output, FFmpeg displays the media file information along with FFmpeg details such as version, configuration details, copyright notice, build and library options etc.

If you don't want to see the FFmpeg banner and other details, but only the media file information, use **-hide_banner** flag like below.

```
ffmpeg -i video.
```

Sample output:

```

[tk@tk ~]$ ffmpeg -i video.mp4 -hide_banner
Input #0: mov,mp4,m4a,3gp,3g2,mj2, from 'video.mp4':
Metadata:
  major_brand      : isom
  minor_version    : 512
  compatible_brands: isomiso2avc[mp4]
  encoder         : Lavf57.22.100
  duration: 00:42:18.00, start: 0.000000, bitrate: 1020 kb/s
  Stream #0:0(und): Video: h264 (High) (avc1 / 0x31637681), yuv420p, 1280x714
  [SAR 1071:1072 DAR 120:127], 899 kb/s, 23.98 fps, 23.98 tbr, 24k tbn, 47.95 tbc (default)
Metadata:
  handler_name      : VideoHandler
  Stream #0:1(und): Audio: aac (LC) (mp4a / 0x61347080), 48000 Hz, stereo, flt
  [SAR 132 kb/s (default)]
Metadata:
  handler_name      : SoundHandler
[tk@tk ~]$
  
```

See? Now, it displays only

OSTechNix

[google.com/+SenthilkumarP](https://www.google.com/+SenthilkumarP)

Подписаться

416 подписчиков

FOLLOW US ON TWITTER

My Tweets



How To Run A Linux Command Every X Seconds Forever

Orbital Apps – A New Generation Of Linux applications

How To Improve Application Startup Time In Linux

The Numfmt Command Tutorial With Examples For Beginners

A Beginners Guide To Cron Jobs

the media file details.



Recommended

Download – [Free Guide:](#)

[“Spotify Music](#)

[Streaming: The](#)

[Unofficial Guide”](#)

2. Converting video files to different formats

FFmpeg is powerful audio and video converter, thus It's possible to convert media files between different formats. Say for example, to convert mp4 file to avi file, run:

```
ffmpeg -i video.1
```

Similarly, you can convert media files to any format of your choice.

For example, to convert youtube flv format videos to mpeg format, run:

```
ffmpeg -i video.
```

If you want to preserve the quality of your source video file, use '-qscale 0' parameter:

```
ffmpeg -i input.
```

To check list of supported formats by FFmpeg, run:

```
ffmpeg -formats
```

3. Converting video files to audio files

To convert a video file to audio file, just specify the output format as .mp3, or .ogg, or any other audio formats.

The above command will convert input.mp4 video file to output.mp3 audio file.

```
ffmpeg -i input.
```

Also, you can use various audio transcoding options to the output file as shown below.

```
ffmpeg -i input.
```

Here,

- **-vn** – Indicates that we have disabled video recording in the output file.
- **-ar** – Set the audio frequency of the output file. The common values used are 22050, 44100, 48000 Hz.
- **-ac** – Set the number of audio channels.
- **-ab** – Indicates the audio bitrate.
- **-f** – Output file format. In our case, it's mp3 format.

4. Change resolution

of video files

Suppose, you want to set a particular resolution to a video file, you can use following command:

```
ffmpeg -i input.1
```

Or,

```
ffmpeg -i input.1
```

The above command will set the resolution of the given video file to 1280×720.

Similarly, to convert the above file to 640×480 size, run:

```
ffmpeg -i input.1
```

Or,

```
ffmpeg -i input.1
```


This trick will help you to scale your video files to smaller display devices such as tablets and mobiles.

5. Compressing video files

It is always better to reduce the media files size to lower size to save the harddisk's space.

The following command will compress and reduce the output file's size.

```
ffmpeg -i input.
```

Please note that you will lose the quality if you try to reduce the video file size. You can lower that crf value to 23 or lower if 24 is too aggressive.


You could also transcode the audio down a bit and make it stereo to reduce the size by including the following options.

```
-ac 2 -c:a aac -
```



Recommended

Download – [Free Guide:](#)

[“PLEX, a Manual: Your
Media, With Style”](#) 

6. Compressing Audio files

Just compressing video files, you can compress audio files using **-ab** flag in order to save some disk space.

Let us say you have an audio file of 320 kbps bitrate. You want to compress it by changing the bitrate to any lower value like below.

```
ffmpeg -i input.
```

The list of various available audio bitrates are:

1. 96kbps
2. 112kbps

3. 128kbps
4. 160kbps
5. 192kbps
6. 256kbps
7. 320kbps

7. Removing audio stream from a media file

If you don't want to a audio from a video file, use **-an** flag.

```
ffmpeg -i input.
```

Here, 'an' indicates no audio recording.

The above command will undo all audio related flags, because we don't audio from the input.mp4.

8. Removing video stream from a media file

Similarly, if you don't want video stream, you could easily remove it from the media file using

'vn' flag. vn stands for no video recording. In other words, this command converts the given media file into audio file.

The following command will remove the video from the given media file.

```
ffmpeg -i input.
```

You can also mention the output file's bitrate using '-ab' flag as shown in the following example.

```
ffmpeg -i input.
```

9. Extracting images from the video

Another useful feature of FFmpeg is we can easily extract images from a video file. This could be very useful, if you want to create a

photo album from a video file.

To extract images from a video file, use the following command:

```
ffmpeg -i input.
```

Here,

- **-r** – Set the frame rate. I.e the number of frames to be extracted into images per second. The default value is 25.
- **-f** – Indicates the output format i.e image format in our case.
- **image-%2d.png** – Indicates how we want to name the extracted images. In this case, the names should start like image-01.png, image-02.png, image-03.png and so on. If you use %3d, then

the name of images will start like image-001.png, image-002.png and so on.

10. Cropping videos

It is somewhat similar to change the resolution of the video file. let us say you want to a video with size 300×300. You could do that using command:

```
ffmpeg -i input.
```

Please note that cropping videos will affect the quality. Do not do this unless it is necessary.

11. Convert a specific portion of a video

Sometimes, you might want to convert only a specific portion of the video file to different format. Say for example, the following command will convert the first 50 seconds of given

video.mp4 file to
video.avi format.

```
ffmpeg -i input.
```

Here, we specify the the
time in seconds. Also, it
is possible to specify the
time in **hh.mm.ss**
format.

12. Set the aspect ratio to video

You can set the aspect
ration to a video file
using **-aspect** flag like
below.

```
ffmpeg -i input.
```

The commonly used
aspect ratios are:

- 16:9
- 4:3
- 16:10
- 5:4
- 2:21:1
- 2:35:1
- 2:39:1

13. Adding poster image to audio files

You can add the poster images to your files, so that the images will be displayed while playing the audio files. This could be useful to host audio files in Video hosting or sharing websites.

```
ffmpeg -loop 1 -
```

14. Cut a video file to smaller clip

To trim down a video to smaller clip, we could the command like below.

```
ffmpeg -i input.mp4 -ss  
00:00:50 -codec copy -t  
50 output.mp4
```

Here,

- `-s` – Indicates the starting time of the video clip. In our example, starting time is the 50th

second.

- **-t** – Indicates the total time duration.

15. Split video files into multiple parts


Some websites will allow you to upload only a specific size of video. In such cases, you can split the large video files into multiple smaller parts like below.

```
ffmpeg -i input.
```

Here, **-t 00:00:30** indicates a part that is created from the start of the video to the 30th second of video. **-ss 00:00:30** shows the starting time stamp for the video. It means that the 2nd part will start from the 30th second and will continue up to the end of the original video file.



Recommended

Download – [Free Guide:](#)
[“How to Start Your Own
Successful Podcast”](#) 

16. Joining multiple video parts into one

FFmpeg will also join the multiple video parts and create a single video file.

Create **join.txt** file that contains the exact paths of the files that you want to join. All files should be same format (same codec). The path name of all files should mentioned one by one like below.

```
/home/sk/myvideo  
/home/sk/myvideo  
/home/sk/myvideo  
/home/sk/myvideo
```

Now, join all files using command:

```
ffmpeg -f concat
```

The above command will join part1.mp4, part2.mp4, part3.mp4, and part4.mp4 files into a single file called "output.mp4".

17. Add subtitles to a video file

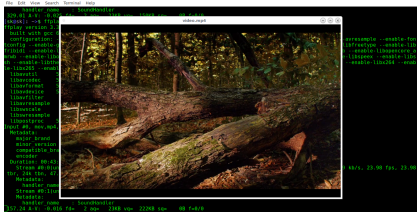
We can also add subtitles to a video file using FFmpeg. Download the correct subtitle for your video and add it your video as shown below.

```
ffmpeg -i input.m
```

18. Preview or test video or audio files

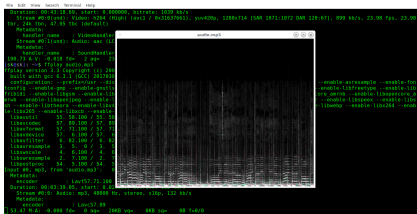
You might want to preview to verify or test whether the output file has been properly transcoded or not. To do so, you can play it from your Terminal with command:

```
ffplay video.mp4
```



Similarly, you can test the audio files as shown below.

```
ffplay audio.mp3
```



19. Increase/decrease video playback speed

FFmpeg allows you to adjust the video playback speed. To increase the video playback speed, run:

```
ffmpeg -i inputv
```

To decrease playback

speed, run:

```
ffmpeg -i inputv
```

20. Getting help

In this guide, I have merely covered the most commonly used FFmpeg commands. FFmpeg has a lot more different options to do various advanced functions. To learn more about it, refer the man page.









```
man ffmpeg
```

And, that's all. I hope this guide will help you to getting started with FFmpeg. If you find this guide useful, please share it on your social, and professional networks.

Cheers!

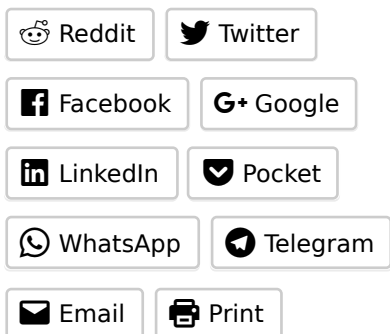
**Thanks for
stopping by!**

**Help us to
help you:**

- Subscribe to
our Email
Newsletter :
[Sign Up
Now](#) 
- Support
OSTechNix :
[Donate Via
PayPal](#) 
- Download free
E-Books and
Videos :
[OSTechNix on
TradePub](#) 
- Connect with
us:
[Facebook](#) 
| [Twitter](#)  |
[Google
Plus](#)  |
[LinkedIn](#)  |
[RSS feeds](#) 

**Have a Good
day!!**

Share this:



Tags:

Audio and Video converters

Audio Conversion FFmpeg

FFmpeg Commands Linux

Video conversion

PREVIOUS	NEXT
STORY	STORY
How	Install
To	Complete
Securely	Fedora
And	Wallpaper
< Permanently	Packs >
Delete	(From
Your	Fedora
Data	8 to
In	Fedora
Linux	25)

YOU MAY ALSO LIKE...



Trizen How Crypt
 – A To omat
 Light Auto or –
 weigh matic An
 t AUR ally Open
 Packa Orga Sourc
 ge nize e
 Mana Simila Client
 ger r Type -side
 For Of Encry
 Arch- Files ption
 Based In Tool
 Syste Speci For
 ms fic Your
 FEBRUARY Folde Cloud
 15, 2018 rS AUGUST
 MARCH 15,
 17, 2017
 2017

3 RESPONSES

 **Comments** 3

 **Pingbacks** 0



IJK

🕒 May 12, 2017 at 4:12
 am

The -sameq option is
 absent in recent
 versions of ffmpeg.

Reply

SK

🕒 May 12, 2017

at 12:35 pm

Thanks for the
heads up. I
removed it
now. Much
appreciated.

Reply

**IJK**

🕒 May

13, 2017

at 4:52

am

qscale

0

seems

to do

pretty

much

the

same

thing

though.

Reply

LEAVE A REPLY

Comment**Name *****Email ***

Website

Post Comment

☐ **Notify me of
follow-up comments by
email.**

☐ **Notify me of new
posts by email.**

EMAIL NEWSLETTER

**Subscribe to our
mailing list**

TODAY'S MOST READ

How To Find The Size Of A
Directory In Linux
Orbital Apps - A New
Generation Of Linux
applications
How to clear Command
line history in Linux

WHO WE ARE

— [About](#)

— [Contact Us](#)

— [Donate](#)

[Sign Up Now](#)

**We respect your
privacy**

Install and configure DNS
server in Ubuntu 16.04

LTS

Best Texting Apps That
Don't Require Your Phone
Number

How To Install Yaourt In
Arch Linux

Allow Or Deny SSH Access
To A Particular User Or
Group In Linux

How To Install NodeJS On
Linux

20 FFmpeg Commands
For Beginners

How To Install Arch Linux
Latest Version

[Privacy Policy](#)

[Terms and Conditions](#)



OSTechNix © 2018. All Rights Reserved.

This site is licensed under CC BY-NC 4.0.