



FFmpeg

[Documentation home](#) > [Workflows / guidelines / tools](#) > [DR Utilities](#)

Last updated: June 20, 2025

FFmpeg is an open-source command-line utility for converting audio and video files.

The Archives uses FFmpeg mainly to manually convert files from one format to another.

A common use case is where audio recordings, video or films have been digitized as preservation copies (wav, mkv), but access copies (mp3, mp4) haven't yet been created (not yet ingested to Archivematica); you need a smaller access copy to provide to a researcher or the donor but don't want to run it through Archivematica.

^ Installation

Install FFmpeg with [Homebrew](#) (Mac users).

- You will need administrator access to your computer to install the software.

Open Terminal and enter the following command:

```
brew install ffmpeg
```

No special configuration is needed.

^ Convert a single audio file to mp3

To convert a file using FFmpeg (example shows an m4a file):

1. Open Terminal.
2. Run the following FFmpeg command, specifying the location of the source file and the new converted file (mp3).

```
ffmpeg -i /path/to/source/file.m4a -ac 2 -ab 192000 /path/to/converted/file.mp3
```

Convert a set of audio files to mp3



In this scenario, you want to batch convert a set of audio files to mp3; example shows a set of wav files. There are two steps:

- A. Get files for conversion into a single target folder.
- B. Run conversion script.

A. Get files for conversion into a single target folder.

If the preservation copies are organized as a set of sub-folders (e.g. one folder for each file / item), use the **ConsolidateItems** Automator app to move all items to a single top-level target folder and delete the sub-folders.

- The app is available at [ITM001 > DR Utilities > ConsolidateItems](#).

This will change the structure of the folder (deletes all sub-folders), so make sure you are working with a copy, not the original preservation set.

1. Download the Automator app, unzip and store somewhere on your computer (e.g. Desktop).
2. Drag the target folder on to the Automator app.
 - Folder will be restructured so that there is just one folder + set of items (all sub-folders deleted and contents moved to top level of target folder).
3. After consolidation, create a sub-folder in the target folder called mp3.

Your folder should now look like this:

```
TargetFolder
  file1.wav
  file2.wav
  file3.wav
  ...
mp3 [empty folder for converted mp3 files]
```

B. Run conversion script

1. Open your Terminal app.
2. Use cd to navigate to the target folder.
 - The easiest way is to enter cd [+ space], then drag the target folder into the Terminal window to get its path.
3. Run the following command:

```
for f in *.wav; do ffmpeg -i "$f" "mp3/${f%.*}.mp3"; done
```

This will create an mp3 for each wav and put it in the **mp3** folder.

Note that on some computers the script fails if you have already created the empty **mp3** folder before running the command.

- On others, it fails if you don't first create the mp3 folder.
- Try it without first creating the folder; if it fails, create the folder and try again.

^ Convert a video file to mp4

To convert a file using FFmpeg (example shows conversion of an mkv file), open Terminal and run the following ffmpeg command, specifying the location of the source file (mkv) and the new converted file (mp4).

```
ffmpeg -i /path/to/source/file.mkv -vcodec libx264 -pix_fmt yuv420p -preset medium -crf 18  
/path/to/converted/file.mp4
```

Notes:

- The easiest way to get file paths is to drag the file into the Terminal window at the appropriate spot in the command.
- For an alternative conversion method, use the [VLC app](#).

To run this script recursively on a set of files:

- Make sure the folder contains no sub-folders, it should have only the set of files to be converted.
- Create a sub-folder called **access** and run the following command (example assumes the files are mkv and should be converted to mp4; change as needed to the required extensions):

```
for f in *.mkv; do ffmpeg -i "$f" -vcodec libx264 -pix_fmt yuv420p -preset medium -crf 18 "access/${f%.*}.mp4"; done
```

^ Links

FFmpeg site:

- <https://www.ffmpeg.org/>