

FFmpeg License and Legal Considerations

Legal issues are a constant source of questions and confusion. This is an attempt to clarify the most important issues. The usual disclaimers apply, this is not legal advice.

FFmpeg License

FFmpeg is licensed under the [GNU Lesser General Public License \(LGPL\)](#) version 2.1 or later. However, FFmpeg incorporates several optional parts and optimizations that are covered by [the GNU General Public License \(GPL\) version 2](#) or later. If those parts get used the GPL applies to all of FFmpeg.

Read the license texts to learn how this affects programs built on top of FFmpeg or reusing FFmpeg. You may also wish to have a look at the [GPL FAQ](#).

Note that FFmpeg is not available under any other licensing terms, especially not proprietary/commercial ones, not even in exchange for payment.

License Compliance Checklist

The following is a checklist for LGPL compliance when linking against the FFmpeg libraries. It is not the only way to comply with the license, but we think it is the easiest. There are also a few items that are not really related to LGPL compliance but are good ideas anyway.

Compile FFmpeg without "--enable-gpl" and without "--enable-nonfree".

Use dynamic linking (on windows, this means linking to dlls) for linking with FFmpeg libraries.

Distribute the source code of FFmpeg, no matter if you modified it or not.

Make sure the source code corresponds exactly to the library binaries you are distributing.

Run the command "git diff > changes.diff" in the root directory of the FFmpeg source code to create a file with only the changes.

Explain how you compiled FFmpeg, for example the configure line, in a text file added to the root directory of the source code.

Use tarball or a zip file for distributing the source code.

Host the FFmpeg source code on the same webserver as the binary you are distributing.

Add "This software uses code of [FFmpeg](http://ffmpeg.org) licensed under the [LGPLv2.1](http://www.gnu.org/licenses/old-licenses/lgpl-2.1.html) and its source can be downloaded [here](#)" to every page in your website where there is a download link to your application.

Mention "This software uses libraries from the FFmpeg project under the LGPLv2.1" in your program "about box".

Mention in your EULA that your program uses FFmpeg under the LGPLv2.1.

If your EULA claims ownership over the code, you have to explicitly mention that you do not own FFmpeg, and where the relevant owners can be found.

Remove any prohibition of reverse engineering from your EULA.

Apply the same changes to all translations of your EULA.

Do not misspell FFmpeg (two capitals F and lowercase "mpeg").

Do not rename FFmpeg dlls to some obfuscated name, but adding a suffix or prefix is fine (renaming "avcodec.dll" to "MyProgDec.dll" is not fine, but to "avcodec-MyProg.dll" is).

Go through all the items again for any LGPL external library you compiled into FFmpeg (for example LAME).

Make sure your program is not using any GPL libraries (notably libx264).

Trademark

FFmpeg is a trademark of Fabrice Bellard, originator of the FFmpeg project.

Patent Mini-FAQ

A lot of legal questions surrounding patents arise when discussing multimedia technology. This mini-FAQ attempts to address these issues. Note that much of this discussion is based on precedent, or what has happened in the past under similar circumstances. Very little consideration is given to what could happen. If you use your imagination, you can visualize any dire scenario and cease doing any productive work.

Q: Does FFmpeg use patented algorithms?

A: We do not know, we are not lawyers so we are not qualified to answer this. Also we have never read patents to implement any part of FFmpeg, so even if we were qualified we could not answer it as we do not know what is patented. Furthermore the sheer number of software patents makes it impossible to read them all so no one (lawyer or not) could answer such a question with a definite no, those who do lie. What we do know is that various standards FFmpeg supports contain vague hints that any conforming implementation might be subject to some patent rights in some jurisdictions, examples for such statements are:

For H.264:

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

And for MPEG-4:

The user's attention is called to the possibility that, for some of the processes specified in this part of ISO/IEC 14496, conformance with this specification may require use of an invention covered by patent rights. By publication of this part of ISO/IEC 14496, no position is taken with respect to the validity of this claim or of any patent rights in connection therewith.

Q: Is it safe to use such patented algorithms?

A: Patent laws vary wildly between jurisdictions, and in many countries patents on algorithms are not recognized. Plus the use of patents to prevent the usage of a format or codec on a specific operating system or together with specific other software might violate antitrust laws. So whether you are safe or not depends on where you live and how judges interpret the law in your jurisdiction.

Q: Bottom line: Should I be worried about patent issues if I use FFmpeg?

A: Are you a private user working with FFmpeg for your own personal purposes? If so, there is remarkably little reason to be concerned. Are you using FFmpeg in a commercial software product? Read on to the next question...

Q: Is it perfectly alright to incorporate the whole FFmpeg core into my own commercial product?

A: You might have a problem here. There have been cases where companies have used FFmpeg in their products. These companies found out that once you start trying to make money from patented technologies, the owners of the patents will come after their licensing fees. Notably, MPEG LA is vigilant and diligent about collecting for MPEG-related technologies.