

Audio Tools for Just Cause 3

Author: Lord Zapharos (www.lordzapharos.com)

Date: 24 February 2025

I. Description

This is the *definitive* set of tools for extracting and modifying audio in Just Cause 3.

By following this guide, you will learn how to create your own mods, change out music, and—for the first time ever—extract and replace the game’s voice lines. These tools support all nine languages offered by Just Cause 3.

II. Credits

I cannot claim sole credit for this suite of tools. The works of multiple talented authors have been combined into this suite, alongside my own scripts and analysis of course. In particular, I would like to thank the following, and I claim no copyright or credit for their own hard work:

- **python-fsb5** by *HearthSim*
 - <https://github.com/HearthSim/python-fsb5>
- **oggvorbis2fsb5** by *uyjulian*
 - <https://github.com/uyjulian/oggvorbis2fsb5>
- **Fmod Bank Tools** by *Wouldubeinta*
 - <https://www.nexusmods.com/rugbyleaguelive3/mods/2>
- **Modified Gibbed’s Tools** by *RICK / neam*
 - <https://videogamemods.com/justcause3/mods/modified-gibbeds-tools/>
- **DLC Packer** by *neam*
 - <https://videogamemods.com/justcause3/mods/dlc-packer-dlc-mod-tool/>
- **Updated Hash Dictionary** by *LukeJC, UnknownMiscreant, Lexal, and SK83RJOSH*
 - <https://videogamemods.com/justcause3/mods/updated-hash-dictionary-for-jc-tools-1000795/>

Where applicable, I have included the original licenses alongside these tools.



III. Table of Contents

Due to several site closures and relocations, the state of modding for Just Cause 3 can be a bit confusing and even intimidating to newcomers. As such, this guide covers several basic topics in addition to the audio tools themselves—to ensure we are all on the same page.

- I. Description
- II. Credits
- III. Table of Contents
- IV. Where to Get Mods
- V. Setting Up Just Cause 3 for Modding
- VI. Audio Prerequisites
- VII. Extracting Game Audio
- VIII. Compiling Audio Files
- IX. Working with Audio Banks
- X. Appendix A: Filename Hashes
- XI. Appendix B: Known English Audio Files

If you are already familiar with modding Just Cause 3, please feel free to skip ahead.

IV. Where to Get Mods

The largest site for Just Cause 3 mods (<https://justcause3mods.com/>) suffered several closures over the years, and it seems to be down permanently as of 2025. Unfortunately, many mods were hosted exclusively on that site, and are now permanently lost.

The best alternative for Just Cause 3 can be found at <https://videogamemods.com/justcause3>. The community isn't the best at removing old or broken content, however. Check the comments before you download something that no longer works!

Most items on <https://gamebanana.com/games/8348> are reposts of the *videogamemods* site.

I cannot fully recommend <https://www.nexusmods.com/justcause3>, as there just aren't enough mods being posted there, and many are very out-of-date.

Avoid sites like <https://www.gamewatcher.com>, which often link back to the original (defunct) modding community sites, and have a very bad habit of not giving credit to authors.

V. Setting Up Just Cause 3 for Modding

Before you do anything else, you must set up Just Cause 3 to recognize the mods you install. This is pretty easy, but can be confusing to newcomers.

A. Launch Options

Regardless of how you play the game, you must set its launch options to ONE of the following items. Strangely, the option you need to seems to be dependent on your individual computer and/or operating system (for me, only the middle one works). Pick one, test it, and try a different option if it doesn't work.

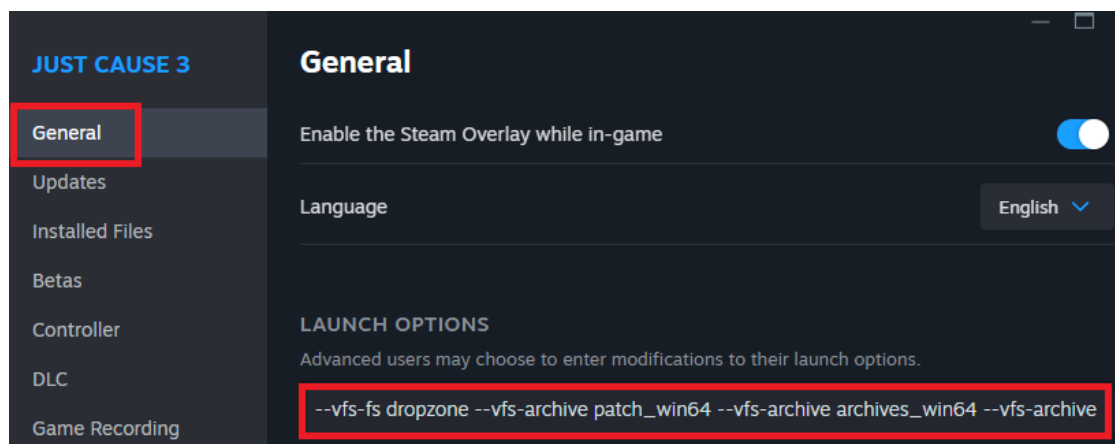
```
--vfs-fs dropzone --vfs-archive patch_win64 --vfs-archive archives_win64 --vfs-archive dlc_win64 --vfs-fs
--vfs-fs dropzone --vfs-archive patch_win64 --vfs-archive archives_win64 --vfs-archive dlc_win64 --vfs-fs.
--vfs-fs dropzone --vfs-archive patch_win64 --vfs-archive archives_win64 --vfs-archive dlc_win64 --vfs-fs .
```

Note: the bottommost option has a space between vfs-fs and the ending period. The middle option does not have this space. The top option does not have an ending period at all.

To actually apply these options to your game, do one of the following:

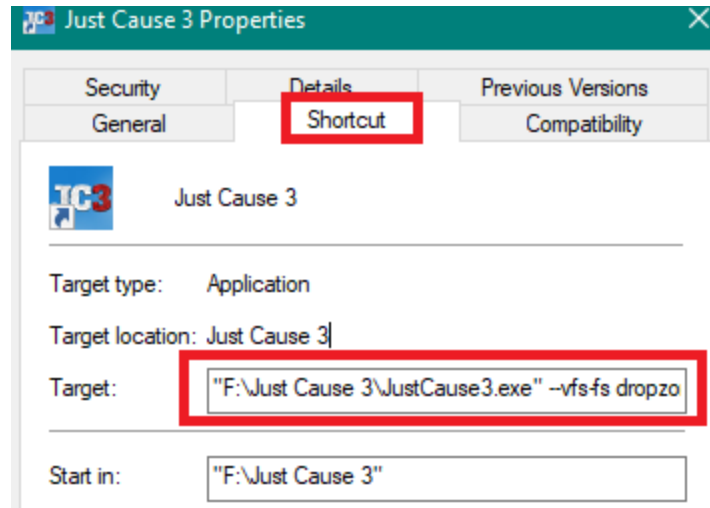
If playing on Steam:

1. In Steam, right-click on your game and select Properties.
2. In the window that appears, go to the General tab and copy one of the launch options from above into the Launch Options field, replacing anything that is already there:



If playing on desktop / via shortcut:

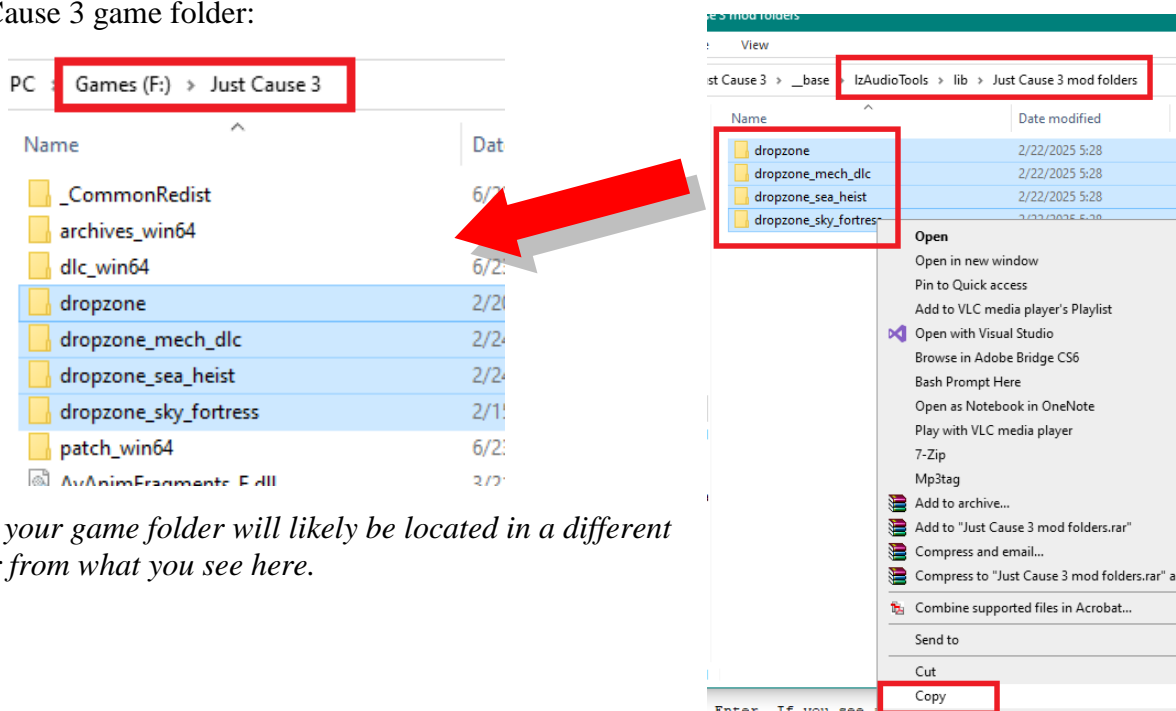
1. If not already done, create a shortcut to the game. *Steam shortcuts don't work for this; you must right-click on JustCause3.exe and select Create Shortcut from the context menu.*
2. Right-click on the shortcut and select Properties from the context menu.
3. Navigate to the Shortcut tab (if not already there) and edit the Target box like so:



Make sure there is a space between the items `JustCause3.exe` and `--vfs-fs`, and don't forget the quotes. You must always launch the game using this modified shortcut.

B. Dropzone Folders

In the *IzAudioTools* folder in which you found this guide, go into the *lib* folder and open *Just Cause 3 mod folders*. Copy all of the *dropzone* folders you see there, then paste them into your Just Cause 3 game folder:

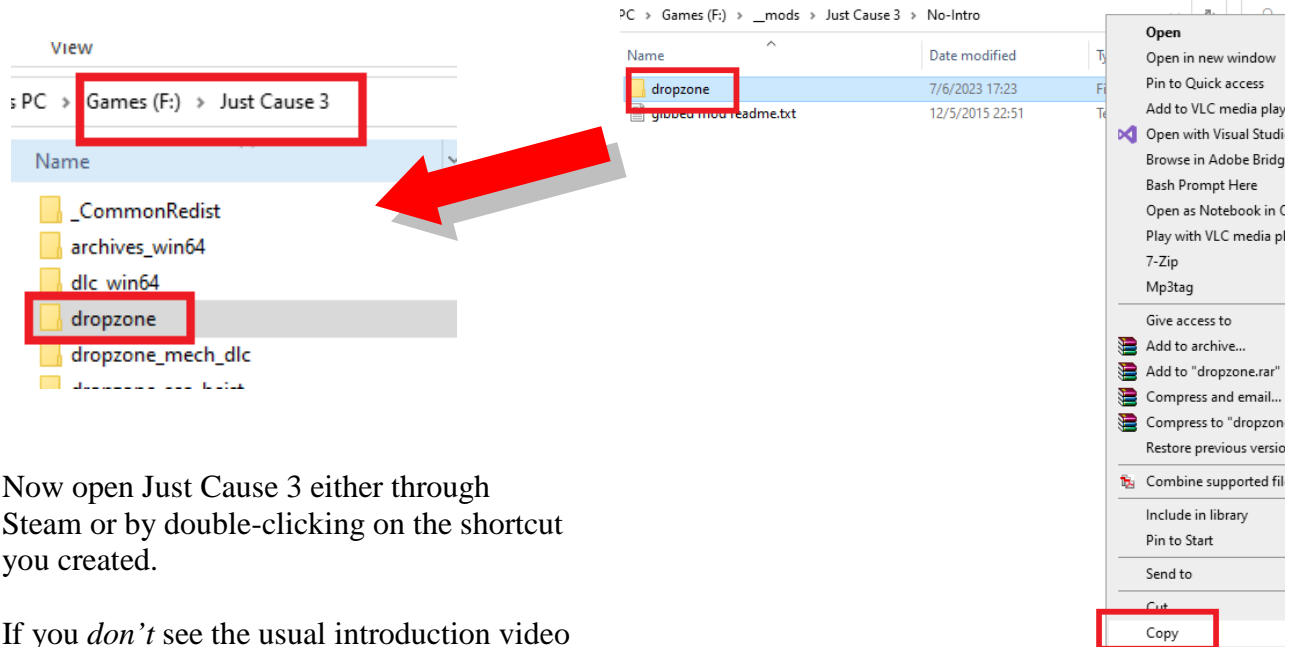


Note: your game folder will likely be located in a different folder from what you see here.

C. Installing Mods (Regular Game)

First, download a simple mod you can easily test. I recommend *No Intro* by Bob Beathard, found here (<https://videogamemods.com/justcause3/mods/no-intro-updated/>).

Unzip the mod and right-click on the *dropzone* folder you see, then select Copy. Paste it into your Just Cause 3 folder, just like in the previous step:



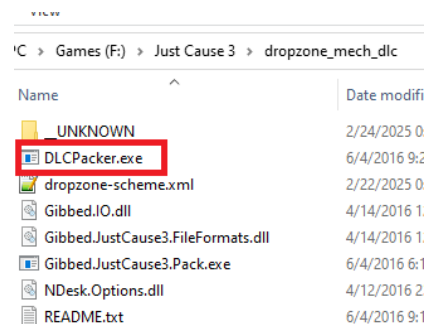
Now open Just Cause 3 either through Steam or by double-clicking on the shortcut you created.

If you *don't* see the usual introduction video (with the raised glass, towel, and explosion), then congratulations! You have just successfully installed your first mod!

This is the same procedure you will need to use for installing most other mods, including the audio mods you create yourself by following this guide.

D. Installing Mods (DLC)

Installing mods for the game's DLC is almost the same procedure. However, you will notice that the DLC folders contain some extra items (if you copied them from the *lzAudioTools\lib* folder):



To install any mods for DLC, do the following:

1. Copy mods into the appropriate *dropzone* folder, just like you would normally do with the main game.
2. Double-click on *DLCPacker.exe* in that mod folder to run it.
3. That's it!

Note that you will not see any messages or popups from the DLCPacker. Just double-click on it, wait for a second or two, and you are done.

The reason we need the DLCPacker is that Just Cause 3 does not permit the loading of loose files for DLCs, unlike with the regular game. This application takes the files you place there and automatically repacks them into the game's own data archives. Pretty nifty!

Note 1: the DLCPacker only needs to be run ONCE any time you install or uninstall mods. In other words, you do not need to run it separately for each DLC folder. Any time you make changes to anything, run the DLCPacker just once before you launch the game.

Note 2: the version of DLCPacker I have included in this suite supports the Bavarium Sea Heist DLC, unlike other packages floating around on the Internet. If you already have the DLCPacker, you can add support yourself by editing the dropzone-scheme.xml, or you can simply copy my XML files to your DLC folders.

E. Uninstalling Mods (Regular Game)

Removing mods from the main game is extremely easy: just remove the files you added.

F. Uninstalling Mods (DLC)

This is also very easy. Remove the files you added, then run *DLCPacker.exe* afterwards.

G. Mod Management and Conflicts

All mods you installed will essentially be dumped into the same folder; there is no easy way to manage different mods by default. When you remove a mod, make sure you are removing the right files!

In general, mods do not conflict unless they alter the same files. You will know this to be the case if Windows asks you whether you would like to overwrite files when copying them to the *dropzone* folders. If this occurs, examine the files carefully. There are some utilities that attempt to make mod-merging easier, if you want to try them (not tested or covered by this guide).

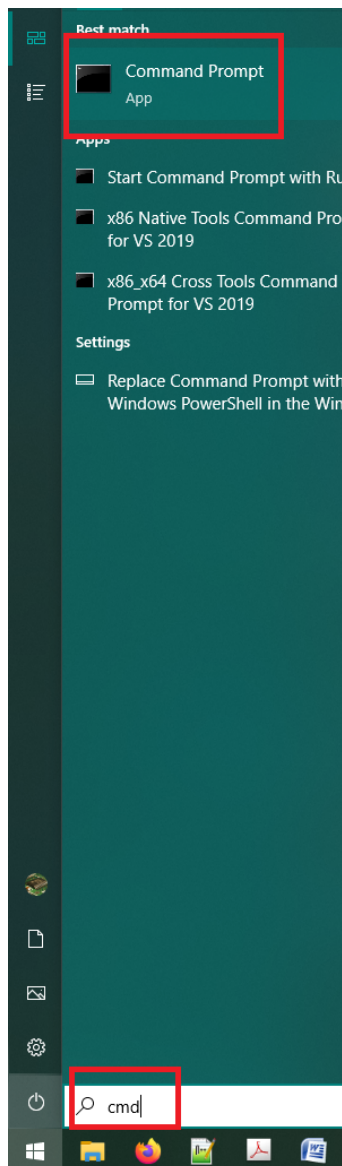
The loose-file format of Just Cause 3 makes mod conflicts less likely than in some other games.

VI. Audio Prerequisites

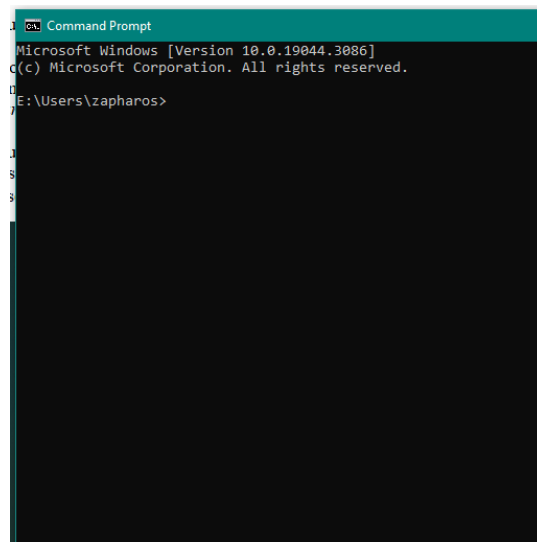
You will need a copy of **Python 3.5+** to run the scripts in this tool suite.

If you don't already have Python, download it from: <https://www.python.org/downloads/>. After doing so, run the installer just like anything else. *If you are prompted whether to add Python to your PATH, make sure you choose Yes! This will make it easier to follow the next steps.*

You will also need some basic understanding of the Command Prompt (*if you already know how to use this, skip to the next section*). To open it, press the Windows key or open the Start menu and type **cmd** (or **command prompt**; either of these will work). Select the Command Prompt or press Enter when it appears:



You should be presented with a black window like the following:



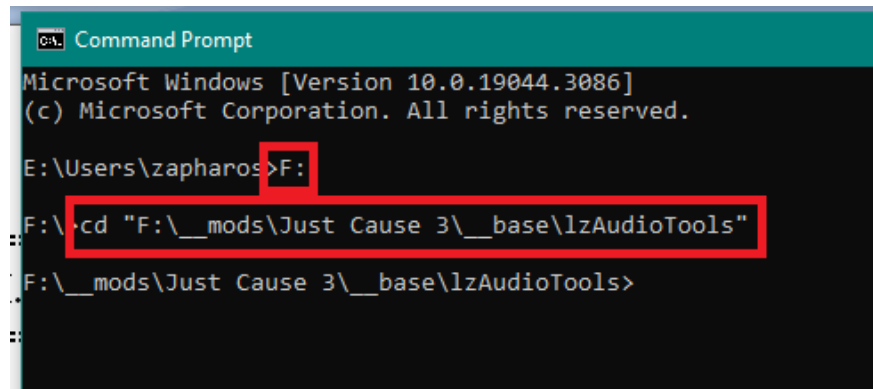
On the left-hand side, you will notice a little prompt—it probably says something like **C:\Users\myname>**. **This is where you are** on the file system (like the search bar at the top of Windows Explorer, without the icons and little arrows).

We can change the current directory by typing the command **cd**, followed by the place we want to go, and then pressing Enter.

If you have multiple drives, you can change to another one by typing the other drive letter e.g. **E:** or **F:**, and then pressing Enter.

When the place we go has spaces in the name, it is often necessary to wrap them with double-quotes **"** and **"**.

Illustrating all of these is the following, in which I change first to my **F:** drive and then navigate to the *lzAudioTools* folder. Note that as I do so, the little prompt to the left of the **>** changes to show where I am at each step:

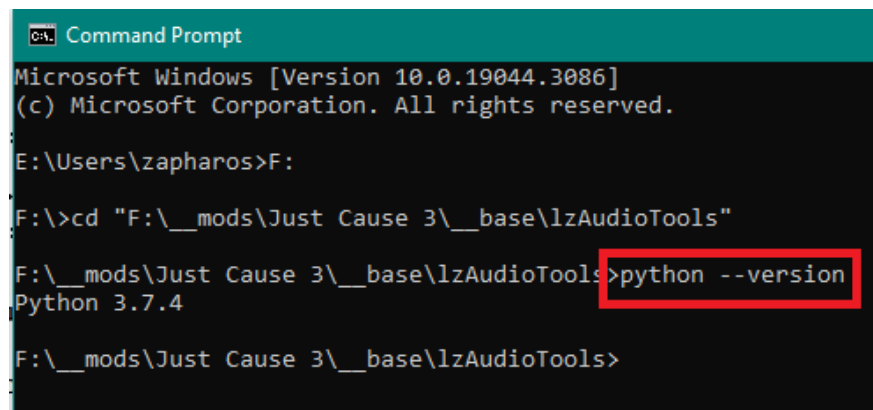
A screenshot of a Windows Command Prompt window. The title bar says "Command Prompt". The text inside shows the user at the prompt "E:\Users\zapharos>". They type "F:" and the prompt changes to "F:\>". Then they type "cd "F:__mods\Just Cause 3__base\lzAudioTools"" and the prompt changes to "F:__mods\Just Cause 3__base\lzAudioTools>". Red boxes highlight the "F:" command and the full "cd" command line.

```
Microsoft Windows [Version 10.0.19044.3086]
(c) Microsoft Corporation. All rights reserved.

E:\Users\zapharos>F:
F:\>cd "F:\__mods\Just Cause 3\__base\lzAudioTools"
F:\__mods\Just Cause 3\__base\lzAudioTools>
```

When you work with the audio tools in this suite, you will need to navigate to the *lzAudioTools* folder in the Command Prompt by using **cd**. Give it a try! Your folder will probably be in a different location, but the process of navigating to it remains essentially the same.

For the next step, let's check our version of Python. Type **python --version** in the Command Prompt and press Enter. You should see a message print out similar to the following (make sure your version is at least 3.5). Mine says *Python 3.7.4*:

A screenshot of a Windows Command Prompt window, continuing from the previous one. The prompt is "F:__mods\Just Cause 3__base\lzAudioTools>". The user types "python --version" and the output "Python 3.7.4" is displayed. A red box highlights the "python --version" command.

```
Microsoft Windows [Version 10.0.19044.3086]
(c) Microsoft Corporation. All rights reserved.

E:\Users\zapharos>F:
F:\>cd "F:\__mods\Just Cause 3\__base\lzAudioTools"
F:\__mods\Just Cause 3\__base\lzAudioTools>python --version
Python 3.7.4
F:\__mods\Just Cause 3\__base\lzAudioTools>
```

If you instead see a message that Python is not a recognized program or application, that means Python wasn't correctly added to your PATH. You can fix that by following the guide here: <https://www.pythoncentral.io/add-python-to-path-python-is-not-recognized-as-an-internal-or-external-command/>.

Once you have Python installed and have navigated to the *lzAudioTools* folder, you can proceed to the next section.

VII. Extracting Game Audio

Now for the fun stuff! Open a Command Prompt and navigate to the *lzAudioTools* folder you unzipped. Type the following command and press Enter: `python extract.py`.

You will see a huge wall of text print out. Scroll back up to the top and take a closer look:

```
F:\_mods\Just Cause 3\_base\lzAudioTools python extract.py

=====
Lord Zapharos Audio Extractor for Just Cause 3
=====

This utility extracts and converts sound archives for Just Cause 3 (and possibly
other Avalanche games) into playable audio files. Converted files will be in OGG
format (you can play them with VLC or edit them with Audacity).

You do NOT require the DLC to use this tool. When performing a full extraction,
any DLC audio will be placed in separate folders, and the main game audio will
be placed in a folder called 'main'.

USAGE: extract.py [options] path\to\file\folder\archive\or\game [outputFolder]

If an output folder is not given, all files will be extracted to the working
directory. The path\to\file\folder\archive\or\game argument is REQUIRED and must
either be a (1) path to a folder, if -r is given; (2) path to the game folder,
if -g is given, or (3) path to a single audio file or archive file, if neither
-r nor -g are given. The output MUST be a file if neither -r nor -g are given
and the input file is not an archive. In this case, if the output is omitted,
the converted file will be placed next to the input.
```

There are three main scripts in the *lzAudioTools* folder. Each of them will display a complete set of instructions when you run them. Included in these instructions are:

1. An explanation of what the script does.
2. Complete documentation on the available options.
3. A set of related examples.

This section deals with extracting audio files from the game so you can edit or listen to them. The following sections explain how to reinsert modified files back into the game.

You have four basic choices when extracting and converting audio:

	Option	What It Does
Full Game	<code>-g</code>	Extracts all audio files for the entire game and all DLC (over 135,000 files!). Requires between 11-23 GB of free space. This is the simplest option for first-timers.
Single File		Converts a single audio file into a playable format.
Single Archive		Extracts all audio files from a single game archive; for example, from <i>archive_win64\game56.arc</i> . Can be useful when you don't have space or time to extract the entire game.
Folder	<code>-r</code>	Converts all supported files in the provided folder into a playable format. Rarely used except when inspecting other mods.

By default, all of these options will also convert the extracted audio streams (into OGG files), which you can then play with VLC or edit with Audacity. If for any reason you *don't* want to convert the game's audio, you can provide the `-e` option to extract it and do nothing else.

If you don't specify an output folder, the script will create one for you automatically (for a single file, the output will be placed alongside).

If you don't have space for 23 GB of files, give the `-d` option to remove the extracted (non-playable) audio files after converting them into a playable format. This will reduce the disk space needed for a full extraction to just under 11 GB. Audio banks are always preserved.

Audio conversion can take a bit of time. This script is multithreaded by default, but users with exceptionally powerful CPUs may benefit from the `-t 16` option to speed things up.

If you don't understand or care about any of that, here is the command you should use:

```
python extract.py -g -b "F:\Just Cause 3"
```

Make sure you substitute your own game path for *F:\Just Cause 3*, then press Enter. The script will begin working; keep in mind that audio extraction and conversion may **take a very long time** (30-40 minutes for fast CPUs; up to 2-3 hours for slower processors and/or spindle drives). While the tool is working, don't type anything or close the window—just let it work!

You will see some progress messages while the tool is doing its thing:

```
F:\_mods\Just Cause 3\_base\lzAudioTools>python extract.py -g -b "F:\Just Cause 3"

Beginning full game extraction (this may take a VERY LONG time!)
Extracting F:\Just Cause 3\archives_win64\game0.tab (this may take a while)
Extracting F:\Just Cause 3\archives_win64\game1.tab (this may take a while)
Extracting F:\Just Cause 3\archives_win64\game7.tab (this may take a while)
Extracting F:\Just Cause 3\archives_win64\game8.tab (this may take a while)
Extracting F:\Just Cause 3\archives_win64\game52.tab (this may take a while)
```

Once everything is finished, the Command Prompt will display a completion message with some additional instructions. You can then go the *out* folder that was created for you in the *lzAudioTools* folder (or to whatever output destination you chose). Within this folder, you will see three folders:

imes (F:) > _mods > Just Cause 3 > _base > _scratch > _audio		
Name	Date modified	Type
_placeChangesHere	2/24/2025 9:02	File
converted	2/22/2025 16:35	File
src	2/22/2025 16:35	File

_placeChangesHere is where you will place modified files for reinsertion into the game (see next section).

converted contains the playable audio.

src contains original files and audio banks.

As an example, let's take a look at the *converted\main* folder. It should be filled with a bunch of *gameX* folders and *gameX_header.dat* files like so:

Name	Date modified	Type
game56	2/22/2025 16:04	File folder
game57	2/22/2025 16:07	File folder
game58	2/22/2025 16:08	File folder
game59	2/22/2025 16:12	File folder
game60	2/22/2025 16:12	File folder
game61	2/22/2025 16:15	File folder
game62	2/22/2025 16:16	File folder
game63	2/22/2025 16:19	File folder
game64	2/22/2025 16:19	File folder
game65	2/22/2025 16:19	File folder
game66	2/22/2025 16:23	File folder
game67	2/22/2025 16:23	File folder
game68	2/22/2025 16:23	File folder
game69	2/22/2025 16:26	File folder
game70	2/22/2025 16:27	File folder
game71	2/22/2025 16:30	File folder
game0_header.dat	2/22/2025 15:57	DAT File
game1_header.dat	2/22/2025 15:57	DAT File
game7_header.dat	2/22/2025 15:57	DAT File
game8_header.dat	2/22/2025 15:57	DAT File
game52_header.dat	2/22/2025 16:00	DAT File
game53_header.dat	2/22/2025 16:01	DAT File

NEVER, EVER TOUCH THE HEADER.DAT FILES! Do not change them, rename them, delete them, or move them. These files are required to reinsert audio files back into your game. If you tamper with them, you will have to do a full game extraction—from *scratch*—to regenerate them. So please, just leave these files alone.

*If you extracted just a single file, you will also see a header.dat file. Leave this alone too. When extracting a single archive, you won't have a main folder, but you will have a header.dat file in the output folder. As before: **leave it alone!***

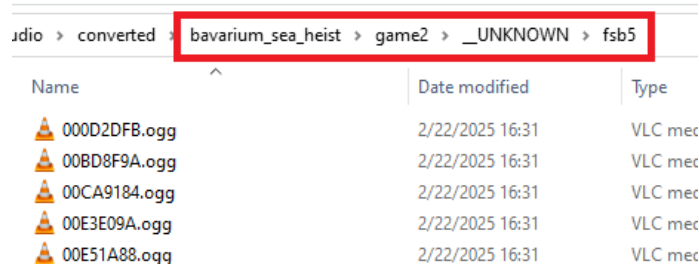
Now let's look a bit deeper at the *game56* folder. You will have to click a few times before you get to the actual audio files (*game56\sound\dialogue\eng*, to be precise):






Name	Date modified	Type
aaa_dialog_test_island_dima.ogg	2/22/2025 16:04	VLC media fi
aaa_dialog_test_island_mario.ogg	2/22/2025 16:04	VLC media fi
aaa_dialog_test_island_mario_clean.ogg	2/22/2025 16:04	VLC media fi
aaa_dialog_test_island_mario_cleanandra...	2/22/2025 16:04	VLC media fi
aaa_dialog_test_island_mario_cutscene.o...	2/22/2025 16:04	VLC media fi
aaa_dialog_test_island_mario_enemy_dial...	2/22/2025 16:04	VLC media fi
aaa_dialog_test_island_mario_mission_cri...	2/22/2025 16:04	VLC media fi
aaa_dialog_test_island_mario_radio.ogg	2/22/2025 16:04	VLC media fi

The files here are OGG format, which is playable with VLC and many other media players (though unfortunately, not always by the default player Windows installs). You can usually just double-click the files and select your audio player to open them.

If you have trouble, download VLC from <https://www.videolan.org/> and try that instead.

The files in *game56* are pretty easy to understand and play. However, not all files are nearly so nicely named. For example, take a look at *bavarium_sea_heist\game2_UNKNOWN\fsb5*:



Name	Date modified	Type
 000D2DFB.ogg	2/22/2025 16:31	VLC mec
 00BD8F9A.ogg	2/22/2025 16:31	VLC mec
 00CA9184.ogg	2/22/2025 16:31	VLC mec
 00E3E09A.ogg	2/22/2025 16:31	VLC mec
 00E51A88.oqq	2/22/2025 16:31	VLC mec

The files in this folder (and sadly, in many others too) have completely nonsensical names. Worse yet, the languages are all jumbled up: one file might be English, the next in the list Spanish, and the third German or Portuguese! The reason for this is discussed in the Appendix. For now, just bear in mind that you **may need to do some digging** to find the audio files you want.

All of the loose OGG files you find here represent *voiced* lines i.e. those with translations. For music and sound effects, take a look at the section on audio banks.

VIII. Compiling Audio Files

At some point, you will probably want to change or replace some of the voiced lines in game. But you can't just drop OGG files into your *dropzone* folder—Just Cause 3 won't recognize them like that.

This is what the compilation Python script is for. You can run it via `python compile.py` (don't forget to take a look at the instructions!), like so:

```
F:\_mods\Just Cause 3\_base\lzAudioTools>python compile.py

=====
Lord Zapharos Audio Compiler for Just Cause 3
=====

This utility allows you to repack any modified audio files back into your game,
or simply convert them for distribution to others. You must have previously run
extract.py BEFORE using this tool!

Please note that fmod banks CANNOT be compiled by this tool. You must use the
Fmod Bank Tools in the lib folder to compile those files.

USAGE: compile.py [OPTIONS] path\to\modified\file\or\files [outputFolder]

If an output folder is not given, all compiled audio files will be placed in the
current working directory in a subfolder called '__compiled'. If a full game
folder is given, the appropriate dropzone\ folders will be automatically created
```

As with extraction, you have a few basic options. The process is a bit simpler, though:

	What It Does
Game/Folder	Compiles all of the audio files in the provided folder into a format usable by Just Cause 3.
Single File	Compiles a single audio file by itself. You must have run <i>extract.py</i> on a single file prior to using this option i.e. you can't compile just a single file from a full game extraction.

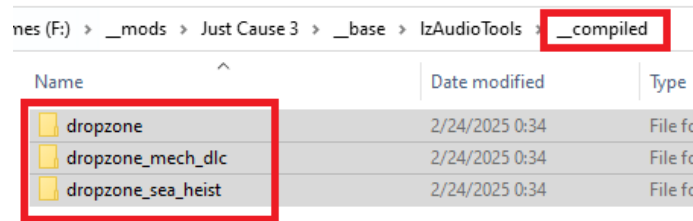
Thanks to all those *header.dat* files, compilation is a LOT faster and more streamlined than the extraction process. The way it works is like this:

First, place all modified files in the `__placeChangesHere` folder that was previously created for you during extraction. **Maintain the same relative folder structure**, but strip the *gameX* parts.

Example: if a file was originally in `converted\main\game56\sound\dialogue\eng`, the modified version should be placed in `__placeChangesHere\main\sound\dialogue\eng`. Notice how the folder structure is almost the same, except for *game56*?

With that done, run `python compile.py out` (or whatever you chose as your output folder), and let the tool do its job. You will see some sporadic messages as the files are recompiled.

By default, the compiled files will be placed in `IzAudioTools__compiled` (but you can choose a different output location if you want). The script will create the needed *dropzone* folders for you automatically like so:



As with any mod, copy these into your Just Cause 3 game folder and run the DLCPacker as needed. Then boot up your game and enjoy the results!

The *dropzone* folders will only be created if you previously extracted the entire game. If you only extracted a single archive, you must place the contents of `__compiled` into the appropriate *dropzone* folder yourself. This is because the script cannot know whether a single archive came from a DLC or from the main game. It's still an easy process—just copy and paste!

By default, the compiler does not overwrite any files already in `__compiled`. To change this behavior, use the `-o` option.

Compilation will not work if you tampered with the header.dat files! If these files became corrupted or unusable, you will need to perform another extraction to regenerate them.

Never put any invalid OGG files in `__placeChangesHere`! Although my script is smart enough to avoid processing files of the wrong type, the underlying converter does not like bad inputs and may cause Windows Explorer to freeze up if given a bad OGG file. This really only occurs if you provide a corrupted/unplayable file, or if you do something silly like rename a photo to `*.ogg` and put it in the `__placeChangesHere` folder (why would you do that, anyways?).

The `__placeChangesHere` folder is quite convenient. If you want to generate another copy of it e.g. when making multiple mods, or if you didn't create this folder during extraction, you can run `createModFolders.py` to make a new set.

Assuming you extracted files to `out` and want a new set of empty folders `__newChangesHere`, just run the following:

```
python createModFolders.py out __newChangesHere
```

You will have to select this new folder manually when you next run `compile.py` (you can find the instructions for how to do this by running `python compile.py --help`).

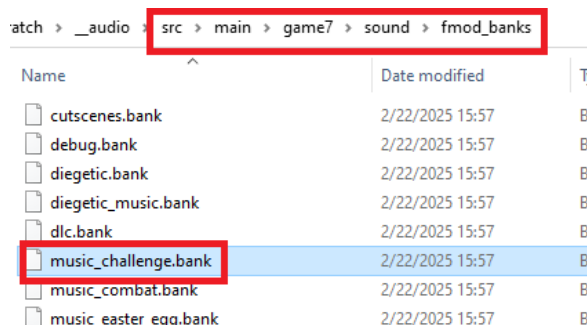
IX. Working with Audio Banks

Audio banks contain **non-localized audio**: **music**, **sound effects**, and so on. The Python scripts extract banks from the game archives for you, but do not manipulate them otherwise. As such, we will use **Fmod Bank Tools** to work with banks.

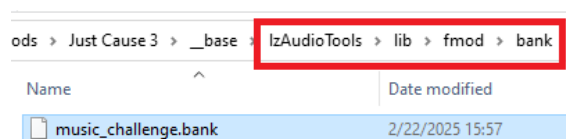
Note: only the main game appears to contain any archives with banks: *game0*, *game1*, *game7*, and *game8*. There is, however, a *dlc.bank* located in *game7* (thus, the DLC sounds may have been “rolled up” into the main game). A few banks contain spoken lines, possibly for demo use.

A. Extracting Banks

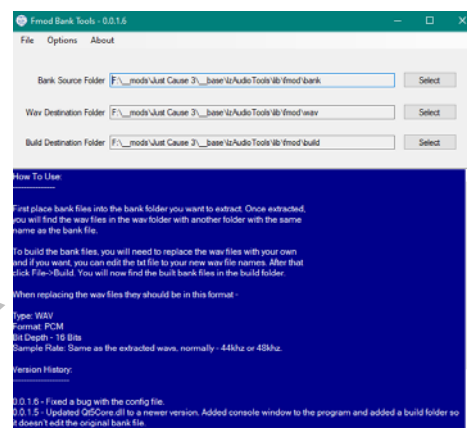
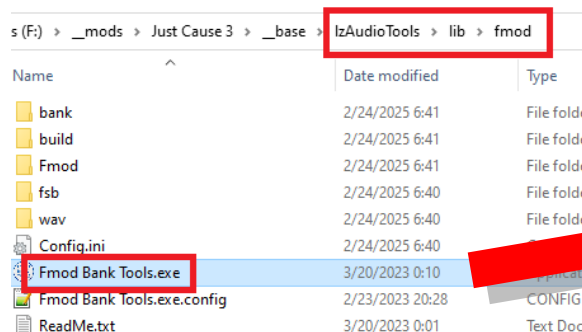
To get started, go to the folder you extracted and look at *src\main\game7\sound\fmod_banks* (note: this will only be found in the *src* folder; not the *converted* folder)



As an example, let's copy *music_challenge.bank* to *lzAudioTools\lib\fmod\bank*:

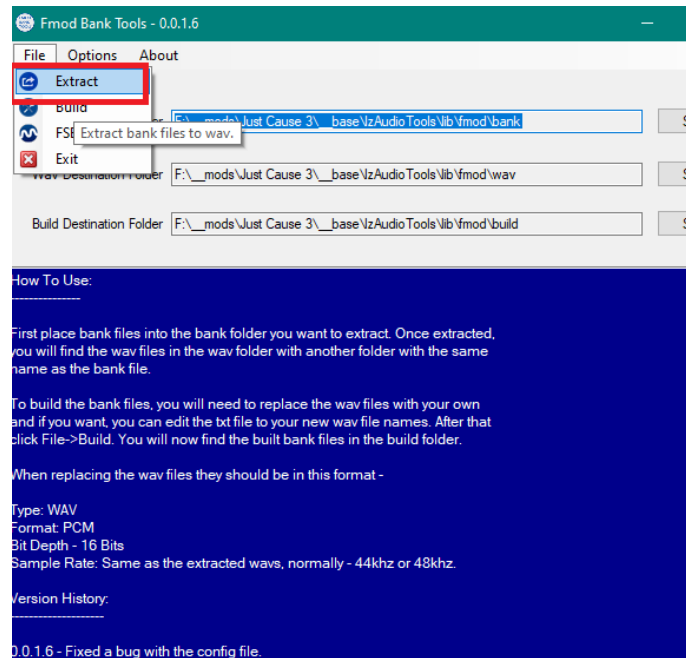


Now, go up one level and double-click on *Fmod Bank Tools.exe* to open it. A window should open looking something like the image on the right:



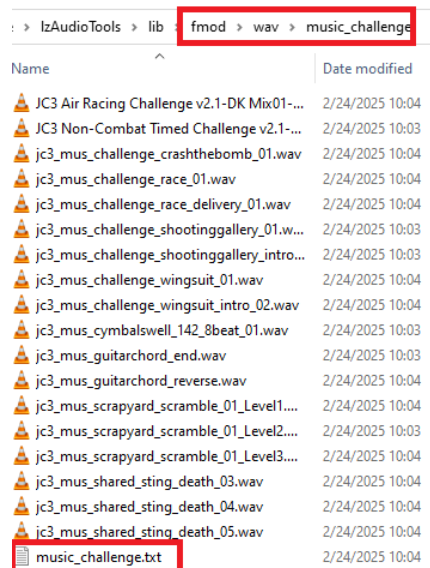
Fmod Bank Tools can be a bit flaky! Many people have reported issues getting the tool to work at all. The version included in this suite has been personally tested by me on Windows 10 and works correctly (though I had to scavenge files from a few different versions to make it so). If you encounter issues, skip ahead to the Troubleshooting section.

Otherwise, make sure all the displayed paths look right; then click on File, followed by Extract:



Allow the tool to extract the BANK file (you can also place multiple banks in the *bank* folder at a time if you want). Depending on the size and number of files, this process can take a while.

When the tool is finished, take a look at the *wav* folder. You should see a subfolder with the same name as the BANK file (*music_challenge* in our case). Within this folder, you can find the playable audio files, in WAV format.



You should also see a file *music_challenge.txt*. Just like with those *header.dat* files—**NEVER TOUCH THE TEXT FILE IN THIS FOLDER!** Do not open it, edit it, or move it. Doing so will prevent you from rebuilding your banks later, and is the **#1 cause of problems** when editing banks.

To avoid the #2 problem, **never rename files in this folder!** If you do, Fmod Bank Tools will be unable to recompile your audio files and, quite unhelpfully, *will not tell you why*.

Members of the modding community have literally wasted *hours* trying to debug this tool just because of an accidentally renamed or moved file. Don't make the same mistake!

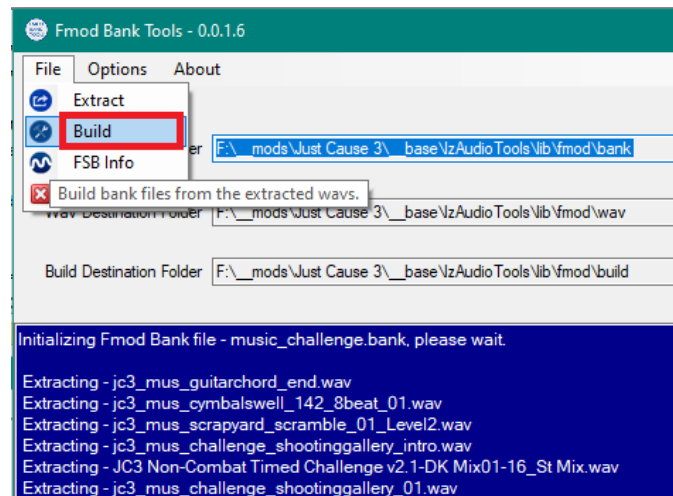
At this point, you can edit the WAV files to be whatever you want. You can substitute different music, your own music, or whatever—just make sure you don’t change those filenames!

You must save any edited WAV files directly in the *wav* folder, overwriting the originals. As such, **I strongly recommend making a backup of this folder** before you begin. Otherwise, any mistake—no matter how small—requires that you re-extract the entire audio bank again, which can be very annoying.

To “remove” an audio file from the game, you must make it silent (do NOT delete it!). Please **follow the tutorial in Section D**, as Fmod Bank Tools will not compile all types of silent files.

B. Compiling Banks

When you have finished editing your WAV files, open up Fmod Bank Tools and choose the Build option from the File menu:



The tool will begin the compilation process, placing the resulting BANK file in the *build* folder you see displayed in the main window. You should see some output in the bottom half of the window while this process is running. **If you don’t see anything at all**, and 2 minutes or so have elapsed, please consult the Troubleshooting section.

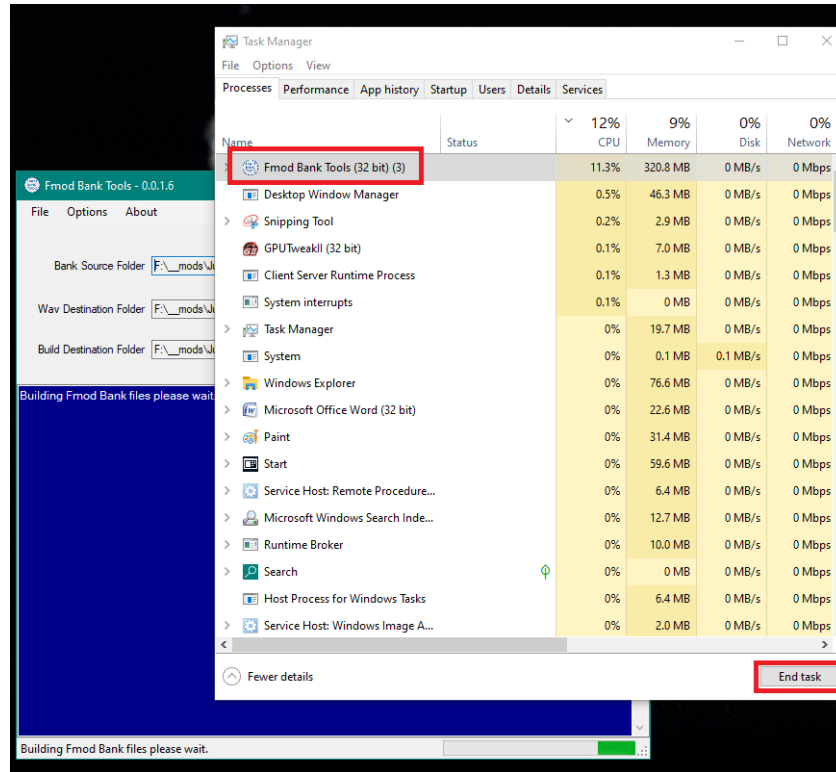
Once your audio bank has been compiled, you can copy it to the *dropzone* folder just like any other mod (in *dropzone\sound\fmdbanks*, to be precise). The changes should take effect when you launch the game. There are no DLC banks, so you don’t need to run the DLCParser.

C. Troubleshooting

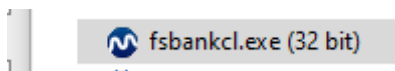
Many people have reported that Fmod Bank Tools **will not compile** their audio banks. On the surface, it looks like an infinite loop: the tool produces no output, but you can often hear your CPU running or see high usage in the Task Manager. Thankfully, this problem is often fixable.

Closing the application doesn't work after an error occurs, so we need to kill the process. To do that, open the Task Manager via the key combination Ctrl+Alt+Escape or Ctrl+Alt+Delete.

Select Fmod Bank Tools from the list and choose End Task like so:



If CPU usage remains high, look for any occurrences of *fsbankcl.exe*, and end those tasks too:



Now that Fmod Bank Tools has been closed, we can try to fix the problem. There are several known causes:

Cause	Resolution
WAV file(s) were renamed.	Make sure the names <u>exactly</u> match those contained by the text file. <i>Check for spaces!</i>
Files were added or removed from the folder, or audio files are not all in WAV format.	The folder contents must include <u>exactly</u> those files listed in the text file, and the text file itself. Remove all other files and subfolders.
WAV file(s) have zero duration.	Occurs when trying to make silent audio files. Follow the instructions in Section D for a fix.
The text file is corrupted or was edited.	Replace the text file with its original version. You may need to re-extract the bank for this.
WAV file(s) are corrupted.	Replace the file with one that plays correctly.

For certain files, you may notice that **extracting an audio bank and immediately rebuilding it** produces a BANK file of very different size. For example, consider *dialog.bank*, which is normally 6947 KB. If you build this audio bank without modifying any of the WAV files, you get a file size around 1200-1300 KB instead:

ods > Just Cause 3 > _base > IzAudioTools > lib > fmod > bank				Search
Name		Date modified	Type	Size
dialog.bank	Original File	2/22/2025 15:57	BANK File	6,947 KB

ods > Just Cause 3 > _base > IzAudioTools > lib > fmod > build				Search
Name		Date modified	Type	Size
dialog.bank	Rebuilt Original File?	2/25/2025 4:58	BANK File	1,229 KB

We didn't modify anything, so what happened?

Don't worry! Fmod Bank Tools is working correctly. The issue is that not all audio streams in BANK files are compressed the same way. Some files are highly compressed, others less so, and some—like *dialog.bank*—are not compressed at all. But Fmod Bank Tools *always* compresses streams into FSB5, and this makes some regular game files smaller than they normally are.

Fortunately, this behavior does not seem to affect game playback in any way.

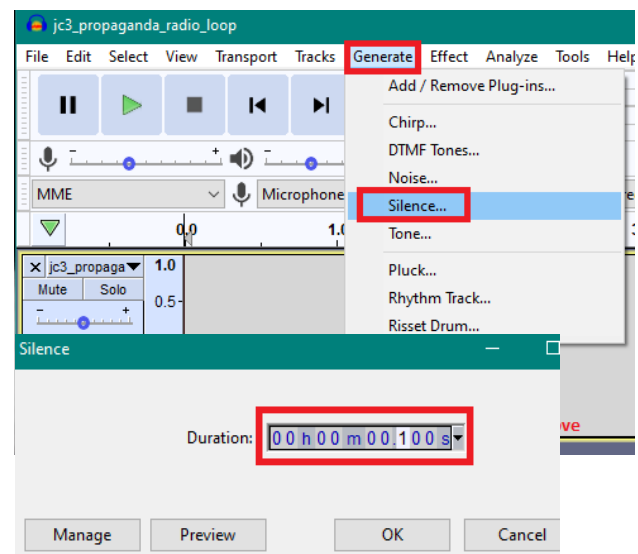
D. Silencing/Removing Audio Bank Tracks

You cannot “remove” audio from a bank by deleting the file. Instead, you must make it *silent*. Preferably, you want this silence to be for as short a duration as possible so it won't affect any other game sounds.

To make an audio file silent in Audacity, first select everything via Ctrl+A, and then press Delete to remove the data.

Then choose Generate → Silence from the menu. In the resulting box, choose exactly 0.1 seconds of silence, then choose OK and save the file.

Warning! Whenever a track has a length of less than 0.1 seconds, Fmod Bank Tools may fail to compile it. If you cannot compile your file, it is usually because you deleted the audio, but did not replace it with at least 0.1 seconds of silence.



The exact threshold for this behavior is not known; however, I have confirmed that 0.1 seconds is a safe amount.

=====

X. Appendix A: Filename Hashes

=====

So what *are* all those weird names with the letters and numbers? And why are they so jumbled?

Basically, what’s going on here is that the Just Cause 3 archives—which contain all of the audio files you have been seeing in this guide—**don’t actually contain any filenames**.

Instead, the archives refer to each file by a *hash*. This 32-bit hexadecimal string (i.e. 0-9, A-F) refers uniquely to that file, but only in the relevant archive e.g. *game53.arc*. When the game was packaged for released, a special function (called a *hash function*) converted the “real name” e.g. *sound\dialogue\eng\audiotapes_010.fsb5* into a simpler, shorter form e.g. *0C7A9861.fsb5*.

But to put it simply, the hash function itself was not shipped along with the game, making it *very, very* hard for modders to get the “real name” back from the hash. The fact that we have any names at all is due to the selfless work of many unnamed heroes in our modding community.

You may have noticed the `__UNKNOWN` folder in some of the previous sections. The tool that unpacks Just Cause 3 files has a small database of known hashes and associated heuristics. This allows it to substitute the “real name” for some of the files it unpacks. Any files for which a real name is not available get dumped into the `__UNKNOWN` folder instead.

Because hashing produces an essentially random name, **languages no longer have subfolders** and the **audio files are not clustered** by their original order. This is what makes it so frustrating to figure out which file does what—you simply have to listen to them *all*.

A further issue is that across different archives, the names of wholly different files may *conflict* (the professional term for this is *hash collision*, if you’re interested).

Why is this a problem?

If you ever try to compile *C0C3EA0A.fsb5* using my Python script, you will get an error (or more precisely, nothing will happen). This is because that name refers to two separate files in different *game* archives. The reason we can’t compile them? Because there is only one *dropzone* folder, and you can’t put two files with the same name in the same folder. As a result, **these two files can never be safely modified or compiled. Never!**

The only way to fix this problem is to reverse engineer the hashes to find the true name of each file. We know the hash algorithm and how to brute-force names, but *everything* in the game is hashed—not just audio files. That means decoding at least 500,000 files (possibly more), which takes a lot of CPU power. No one wants to do it, and few people have the hardware to do it.

Just Cause 3 seems to prefer selecting files by their real name, falling back to the hashed name in *dropzone* only as a last resort.

XI. Appendix B: Known English Audio Files

Following from the previous section, the only way to figure out what most audio files contain is to listen to them—all **135,000** of them!

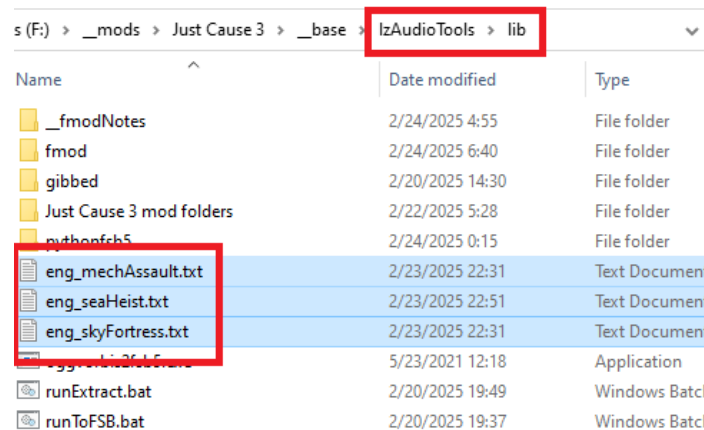
This is made even worse by the nine languages in which Just Cause 3 is offered. Don't believe me? Go to *converted\bavarium_sea_heist\game2__UNKNOWN\fsb5*, pop the first hundred files into a VLC playlist, and listen to them nonstop. It's painful.

Really painful.

And *that* is the only way we can figure out which file does what, short of brute-forcing the hash.

Thankfully for English speakers, I have already done part of the work for the game's DLCs. After more days than I would care to count, I created a **list of all English audio files for the Bavarium Sea Heist, Mech Assault, and Sky Fortress DLCs** (much of the game's other English audio is, thankfully, already named in *main\game56* and *main\game57*).

You can find these files in *lzAudioTools\lib*, and you can open them with any text editor:



If you want an English audio file for any DLC, you generally only need to examine the ones listed in the above files, which will save you quite some time (speakers/transcriptions not given).

This work helps non-English speakers too. It is relatively easy to write a Bash or Powershell script that removes the listed English audio files from your *converted* folder. As such, you only need to go through *eight* languages to find your files instead of nine—or put another way, that's **2,000 fewer files you need to listen to!**

If anyone would like to add to or clarify these lists, I'd be happy to incorporate your changes into this suite (and give you due credit). The more knowledge we have, the better mods we all make!