Mod Organizer 2 and Cyberpunk 2077

Introduction

This guide was inspired by interactions with the Mod Collection community. I had noticed that someone had a deployable instance of Cyberpunk 2077 with Mod Organizer 2 along with mods that many other people have had so many problems including myself. Sadly, the collection author seemed completely unwilling to share any information or any assistance as to how they configured Mod Organizer 2. After downloading their collection and pulling it apart and rebuilding a new Cyberpunk 2077 instance so many, many times, I was able to determine how they did it or close enough. The shameless shall remain nameless.

This guide is written from the perspective of a Steam user. Feel free to add comments to support other online stores.

Installation

1. Install Cyberpunk 2077 through Steam.
    a. Location Ex. D:\Program Files (x86)\Steam\steamapps\common\Cyberpunk 2077.
    b. This will be called the game root directory.
    c. Cyberpunk 2077 and Mod Organizer 2 need to be installed on the same drive in order for Mod Organizer 2 to “write” to the Cyberpunk 2077 directory.
2. Launch Cyberpunk 2077 and verify that it will run.
3. Quit Cyberpunk 2077.
4. Install Mod Organizer 2 from GitHub.
    a. Location Ex D:\Games\MO2
5. Install Root Builder for MO2 from Nexus
    a. Copy the root builder directories into the plugins directory in D:\Games\MO2\
6. Create a Cyberpunk 2077 mod directory.
    a. Location Ex. D:\Games\CP2077
    b. This will be the location for your downloads, mods, overwrite and profile directories. They can take up a considerable amount of disk space. So, plan ahead.

Configuration

7. Change the following line in your D:\Games\MO2\plugins\basic_games\games\game_cyberpunk2077.py file.
    a. From: GameDataPath = "%GAME_PATH%"
    b. To: GameDataPath = "archive"
    c. We are going to separate the game root directory and the gate data directory. Why…

    We are going to take a small break from the installation to explain the background of a problem that has been dogging users trying to use Mod Organizer 2 with Cyberpunk 2077.

    The default configuration for Cyberpunk 2077 and Mod Organizer 2 configures the game data directory to be the same as the game root directory. While this makes mod installation and organization simple, the problem is many mods like Cyber Engine Tweaks (CET) will not work

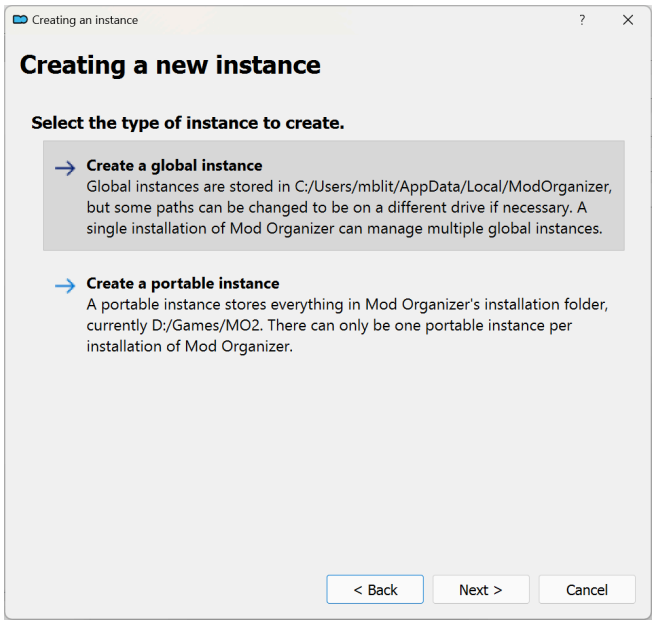
with Mod Organizer 2's Virtual File System (VFS). And if CET and other mods like it don't work, well there isn't much point in modding any further.

The next step was to configure CET to use Root Builder. This seemed a little paradoxical as the default configuration for Cyberpunk 2077 already allows direct access to the game root directory, but it did work and allows CET to function. However, there was a catch that other mods still had problems even when using Root Builder. To be fair to these mod authors, these mods were made when modding was still new for Cyberpunk 2077.

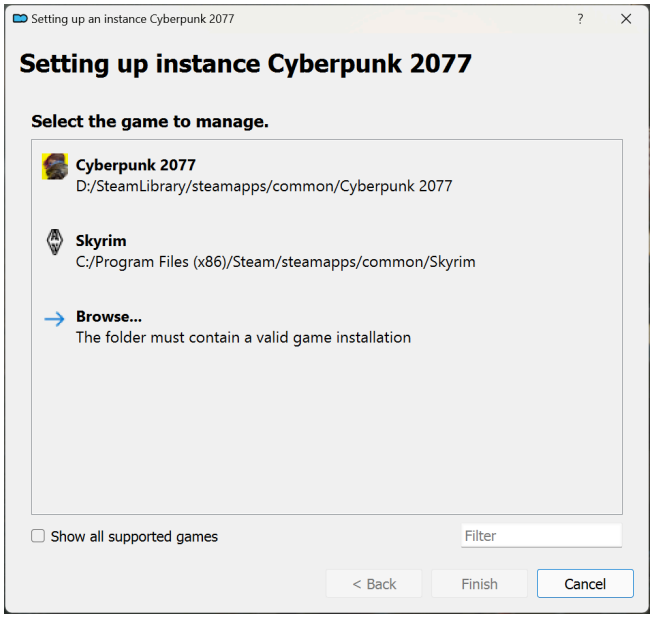
So, someone probably remembering modding Skyrim decided to separate game root and game data directories, e.g. Skyrim and Skyrim\Data. This was the missing step that has seemed to fix 99% of all mods. There are a small number that still have issues. (**Update:** Unable to verify that there are any mods that still have issues.) The game data directory is now Cyberpunk 2077\archive all other non-archive mods will now need to use Root Builder to function. Back to the setup.

8. Launch Mod Organizer 2.

   If it is your first time using Mod Organizer 2, you will get a prompt to set up a new instance. Generally, always choose to create a global instance. If you are already using Mod Organizer 2, then go to File > Manage Instances… to create a new global instance.

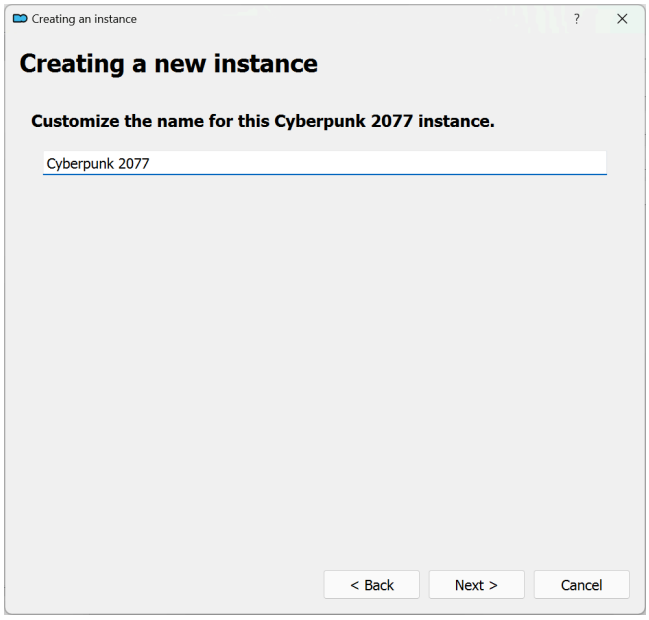


9. Click Create a global instance and click Next.

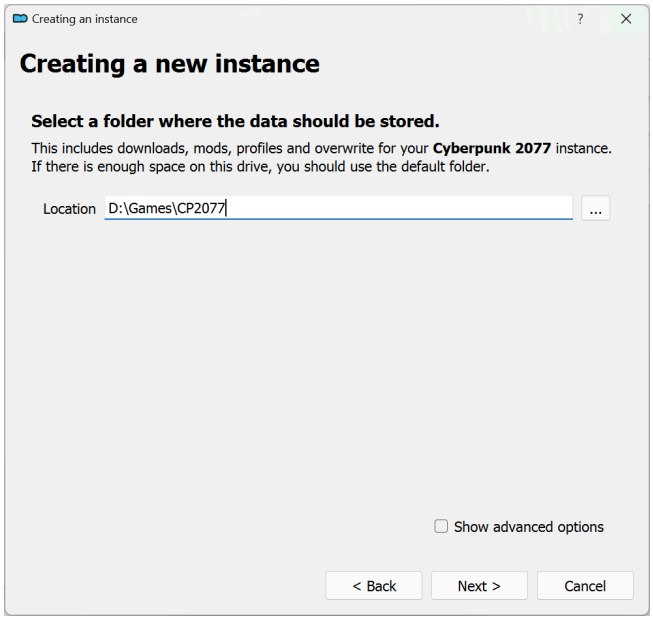


If Mod Organizer 2 does not recognize the Cyberpunk 2077 game, you may have accidentally made a typo in the game_cyberpunk2077.py plugin file. If the Mod Organizer 2's Python plugin for a game has a severe enough error, it will simply not recognize that game. Go back and check your change in step 7 and if necessary, restore the file from the Mod Organizer 2 installation source and try again.

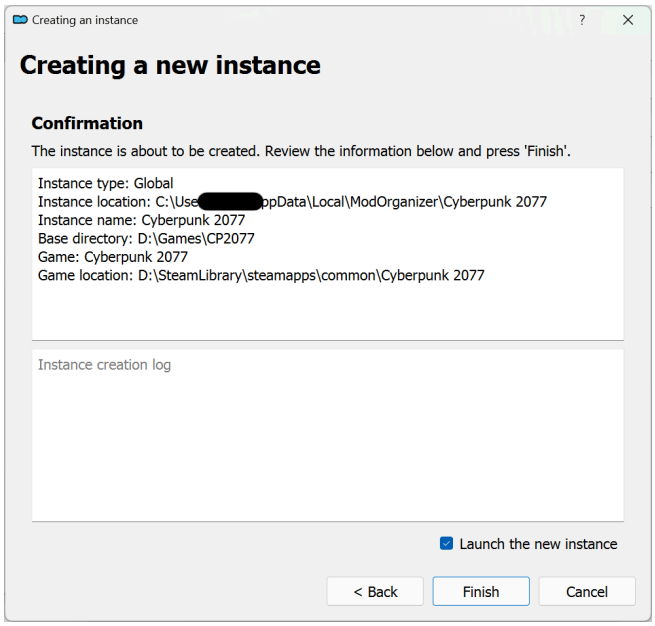
10. Chose Cyberpunk 2077 and Click Next.



11. Give it a name, the default is okay and click Next. (It may be better to name it Cyberpunk 2077 Launcher.)

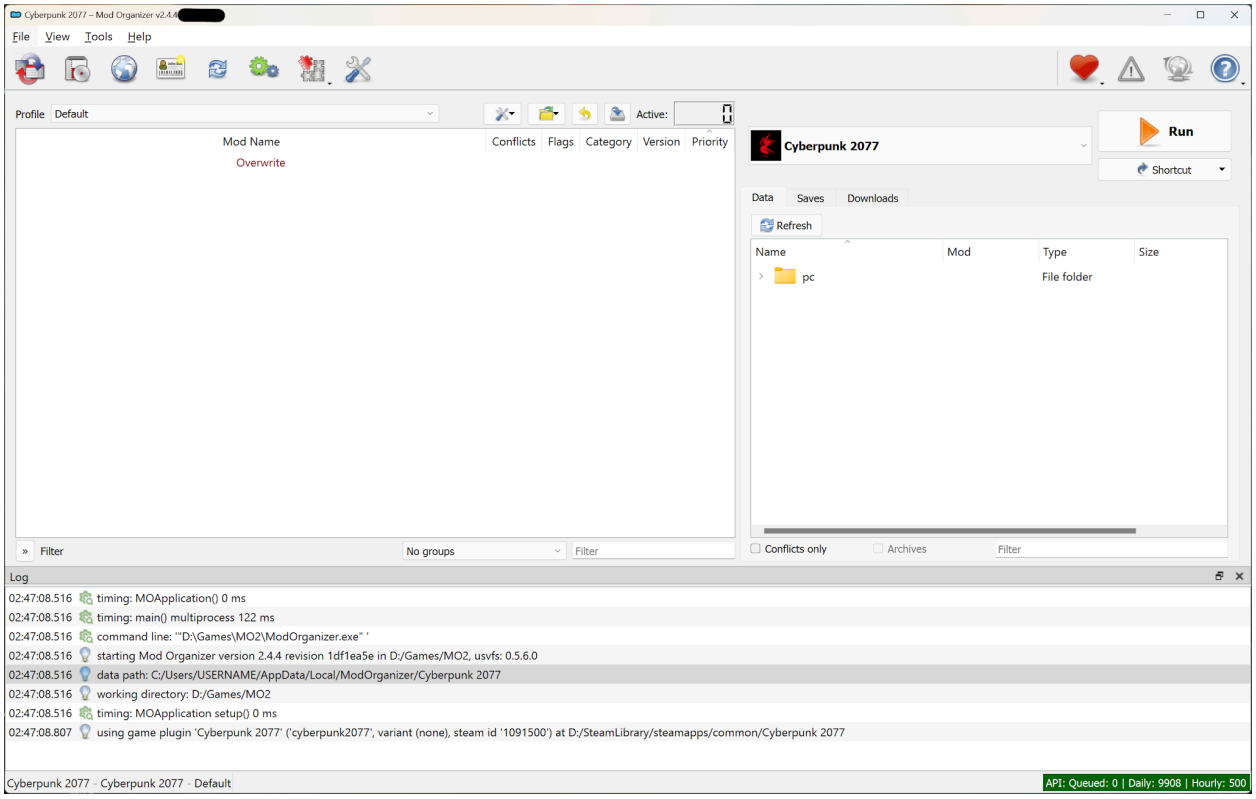


12. Change the location to the directory from step 6. In this case, it is D:\Games\CP2077 and then click Next.



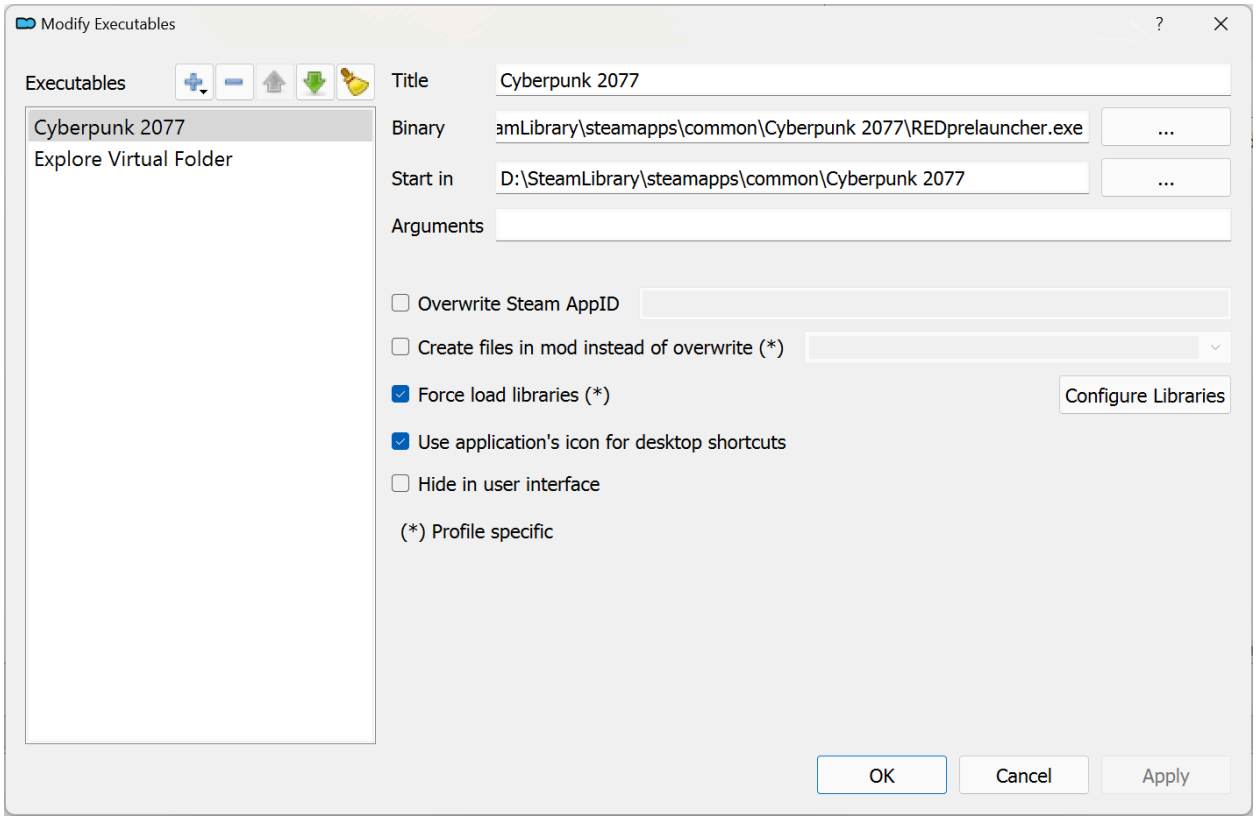
13. Review the confirmation and click Finish.

    Mod Organizer will then restart, and you will then be greeted with the main Mod Organizer 2 window.

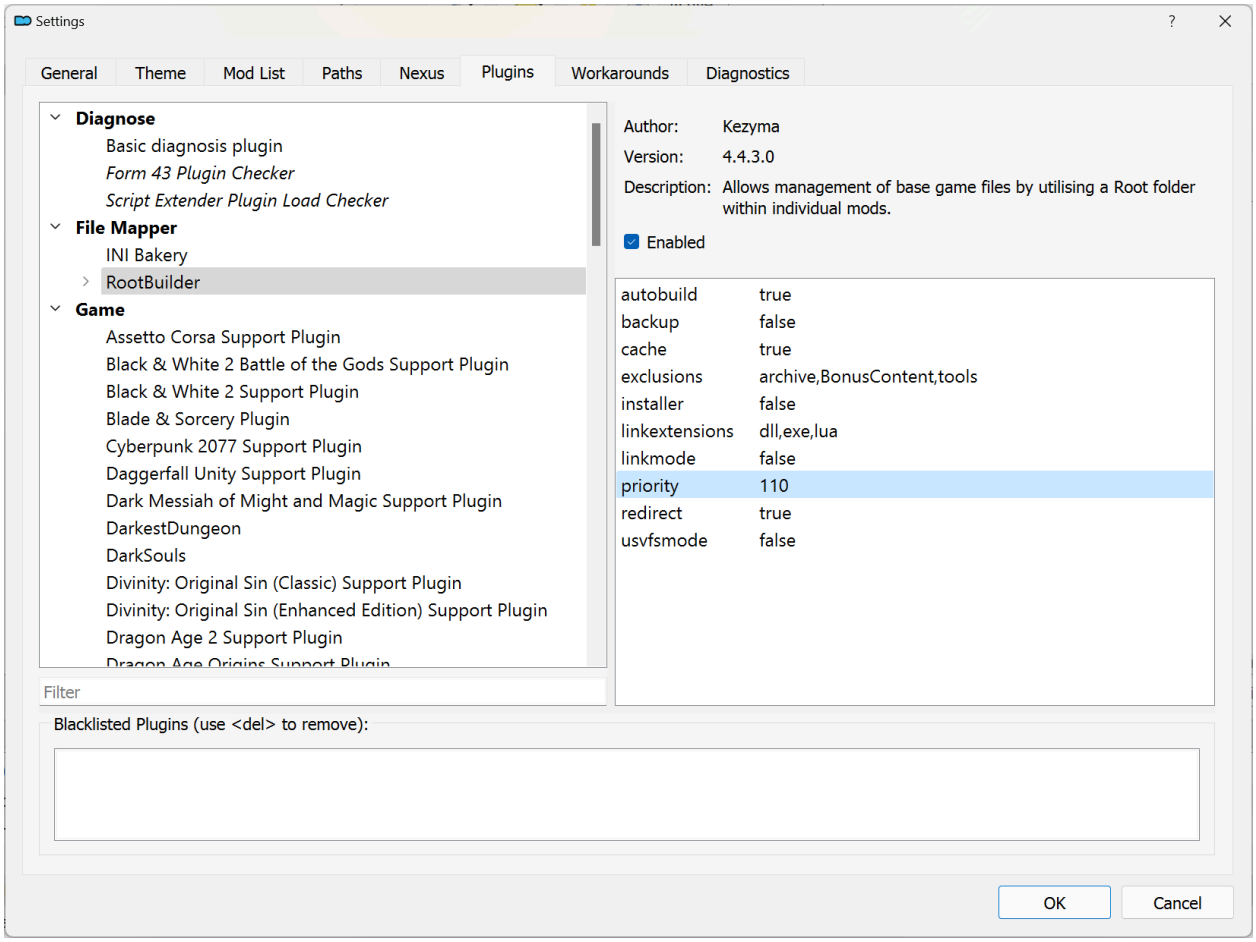


The thing to note is that the Data window on the right will appear to be much more succinct than other examples Mod Organizer 2 builds for Cyberpunk 2077.

14. Click the configurables the executables icon 



15. Check the box to force load libraries if it is not clicked already and then click OK.
16. Click the configure settings and workaround icon. 
    a. Nexus tab
        i. Click the Connect to Nexus button and follow the instructions.
    b. Plugins tab
        i. The only plugin we will configure is for Root Builder.



ii. Set backup to false.
iii. Set exclusions to ”archive,BonusContent,tools”. There are several different exclusion strings I have seen from various sources and no real consensus on which is the best. This is the simplest for now.
iv. Set linkextensions to “dll,exe,lua”. (This may have no effect since we are using Root Builder. It’s a disabled field in another part of Mod Organizer 2.)
v. And then click OK.

You could set installer to true, but I was not impressed with its ability to judge what directory structure to use for mods. There is also another Cyberpunk 2077 plugin on Nexus for Mod Organizer 2. It gets closer to what we need, but it too will often require manual intervention when installing mods because of the way we have configured the game data directory differently.

17. Click the Tool > Root Builder > Root Builder.
    a. Verify the changes made in step 16.b.

We are going to talk about Root Builder for a moment. Root Builder was a plugin originally made for Skyrim. In Skyrim, mods were strictly limited to the game data directory. However, there were things like ENBs (a more powerful version of applications like GShade and ReShade) as well as custom libraries that had to be installed in the Skyrim root directory. Prior to Root

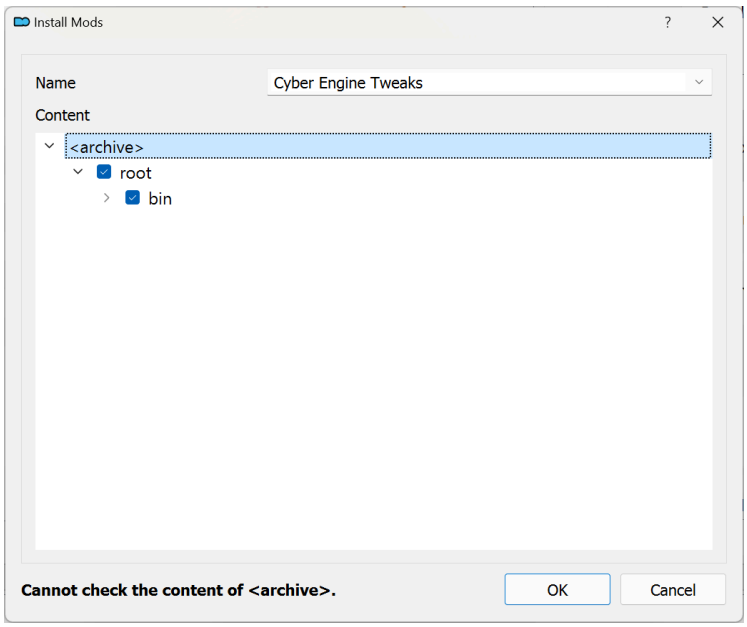
Builder, these had to be managed manually which was a hassle especially for ENBs as they could get very complex. Root Builder allowed a modder to place mods in a special directory called “root” which would then be copied to the Skyrim’s root directory and then deleted when not in use when the game was closed. This was a huge benefit to people with multiple complex ENB setups as now you could use Mod Organizer 2 to manage all those files.

Root Builder had another function which was to back up the root directory. So, in case of a modder error, Skyrim’s executables and libraries could be restored from a save. Root Builder was hard-coded to exclude the Skyrim data directory which contained the bulk of the game’s content to avoid massive backup sizes.

Why is any of this important for Cyberpunk 2077? If the default setting for Root Builder’s exclusions isn’t set to at least “archive”, then any time you click the Root Builder backup button or if backup was set to be enabled, you will be creating 70GB backups on your system as Root Builder has no hard-coded exclusions for any of Cyberpunks 2077’s content. You will also wonder why it takes Cyberpunk 2077 several minutes or more to start or quit as it must copy 70GB every time you start or quit the game. And the last reason is sadly the two functions of Root Builder are tied together. If you exclude the “bin” directory from Root Builder to avoid backing it up, you will also disable mods like CET which are installed in bin.

So in summary, anything in /root mod directory will use Root Builder and anything in any other mod directory will use Mod Organizer 2’s VFS.

18. Install Cyber Engine Tweaks
    a. Download CET from Nexus.
    b. Install CET from the downloads tab.
    c. Right-click on “<archive>” and create a directory and call it “root”.
    d. Drag the bin directory into the root directory.



    e. **This is a first example of a mod that needs Root Builder to work.**

f. Click OK.
g. Launch Cyberpunk 2077.

   The game should start, and you should be prompted to set the CET bind key.

h. Bind the CET hotkey.
i. Disable the intro videos.
j. Quit Cyberpunk 2077.

The next several mods will not be as obvious as to whether they are working as they are modder resources. So will you continue to just install mods until we get to Appearance Change Unlocker.

19. Install redscript
    a. Download redscript from Nexus.
    b. Install redscript from the downloads tab.
    c. Right-click on “<archive>” and create a directory and call it “root”.
    d. Drag the engine and r6 directories into the root directory.
    e. Click OK.
20. Install RED4ext
    a. Download RED4ext from Nexus.
    b. Install RED4ext from the downloads tab.
    c. Right-click on “<archive>” and create a directory and call it “root”.
    d. Drag the bin and red4ext directories into the root directory.
    e. Click OK.
21. Install ArchiveXL
    a. Download ArchiveXL from Nexus.
    b. Install ArchiveXL from the downloads tab.
    c. Right-click on “<archive>” and create a directory and call it “root”.
    d. Drag the r6 and red4ext directories into the root directory.
    e. Click OK.
22. Install TweakXL
    a. Download TweakXL from Nexus.
    b. Install TweakXL from the downloads tab.
    c. Right-click on “<archive>” and create a directory and call it “root”.
    d. Drag the r6 and red4ext directories into the root directory.
    e. Click OK.
23. Install Native Settings UI
    a. Download Native Settings UI from Nexus.
    b. Install Native Settings UI from the downloads tab.
    c. Right-click on “<archive>” and create a directory and call it “root”.
    d. Drag the bin directory into the root directory.
    e. Click OK.
24. Install Codeware
    a. Download Codeware from Nexus.
    b. Install Codeware from the downloads tab.
    c. Right-click on “<archive>” and create a directory and call it “root”.

    d. Drag the r6 and red4ext directories into the root directory.
    e. Click OK.
25. Install Cookedprefabs Nulled
    a. Download Cookedprefabs Nulled from Nexus.
    b. Install Cookedprefabs Nulled from the downloads tab.
    c. Drag the pc directory inside the archive directory into “<archive>”.
    d. This is the first example of an archive mod that doesn’t need Root Builder to work.
    e. Click OK.
26. Install Material and Texture Override
    a. Download Material and Texture Override from Nexus.
    b. Install Cookedprefabs Nulled from the downloads tab.
    c. Right-click on “<archive>” and create a directory and call it “root”.
    d. Drag the mod directory into the root directory.
    e. **This is the first example of a REDmod**.
    f. Click OK.
27. Install Input Loader
    a. Download Input Loader Nulled from Nexus.
    b. Install Cookedprefabs Nulled from the downloads tab.
    c. Right-click on “<archive>” and create a directory and call it “root”.
    d. Drag the red4ext directory into the root directory.
    e. Click OK.
28. Install Appearance Change Unlocker (ACU)
    a. Download Appearance Change Unlocker from Nexus.
    b. Install Cookedprefabs Nulled from the downloads tab.
    c. Right-click on “<archive>” and create a directory and call it “root”.
    d. Drag the bin and r6 directories into the root directory.
    e. Drag the pc directory inside the archive directory into “<archive>”.
    f. This is the first example of a mod that uses archives and Root Builder.
    g. Click OK.
29. Install Virtual Atelier
    a. Download Cookedprefabs Nulled from Nexus.
    b. Install Cookedprefabs Nulled from the downloads tab.
    c. Right-click on “<archive>” and create a directory and call it “root”.
    d. Drag the r6 directory into the root directory.
    e. Click OK.
30. Right-click on the CET mod and click All Mods > Create Separator Above. Call this separator “Core”.
31. Right-click on the Material and Texture Override mod and click All Mods > Create Separator Above. Call this separator “Utilities”.

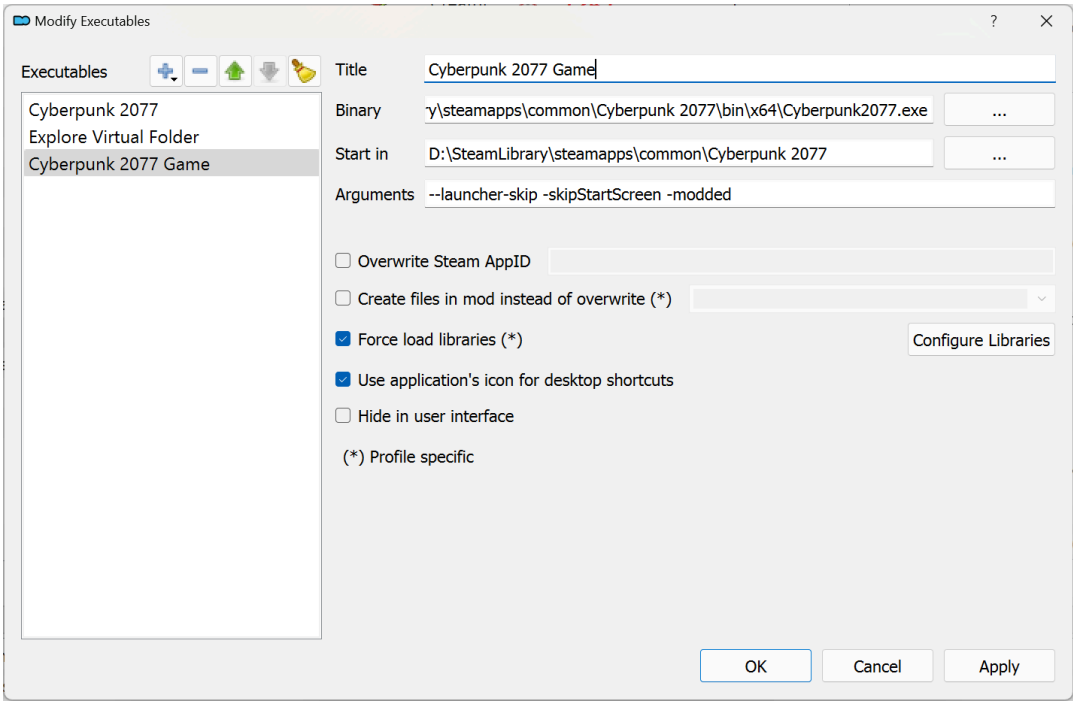
These separators allow you to organize and collapse groups of mods to make managing mods so much easier. While we can debate whether ACU mod should be categorized as an Appearance mod or a Utility mod, the usual trend is that there are many mods on Nexus that are categorized very oddly to say the least. Organize them as you see fit. For some large categories like clothing, I break them up by mod author.

These are the core mods that comprise the core of almost any Cyberpunk 2077 mod build. You have installed one of several different types of mods and should now have enough information to proceed to install whatever mods strike your interest. However, never underestimate a mod author's ability to pack their mod files in the most confusing way possible.

32. You can now start the Cyberpunk 2077 launcher and log into the game.

    The character creator or any mirror should display the ACU is working. Accessing a computer should display that Atelier mod is working (I think. I can't remember if it will show anything if you haven't added any store mods.)

33. Once you have modded Cyberpunk 2077 to your liking, then you can create a new executable menu item for it.
    a. Click the Mod Organizer 2 executable menu drop down list and click <Edit...>.



    b. For Title use "Cyberpunk 2077 Game" or rename the original to "Cyberpunk 2077 Launcher" and this one to "Cyberpunk 2077".
    c. For Binary, navigate to the Cyberpunk2077.exe in \bin\x64 and select it.
    d. For Start in, use "D:\SteamLibrary\steamapps\common\Cyberpunk 2077".
    e. For arguments, use "--launcher-skip -skipStartScreen -modded".
    f. Check the boxes for Force load libraries and Use application icon for desktop shortcuts.
    g. Click OK.

Congratulations, you have successfully completed configuring Cyberpunk 2077 with Mod Organizer 2.

Addendum

Overwrite Directory

You will see new files being created in this special directory. If Mod Organizer 2 is playing nice with Cyberpunk 2077, the only files that you see in the Overwrite directory will be cache files, new files created for store settings (INI files) and log files. You can right-click on the Overwrite directory and create a single mod called “Cache, DB, JSON, INI and Log Files” to store these files all in one place. Do be careful with mods containing your cache files (r6/cache) as mismanaging those is a good way to crash your game.

I would also create a separate mod for Appearance Change Unlocker presets. It sucks to lose a good looking character that wasn’t a part of any save.

I have seen on occasion when Mod Organizer 2 will drop almost the entire contents of the Cyberpunk 2077 root directory into the Overwrite root directory. This causes Cyberpunk to crash and most likely you won’t be able to start Cyberpunk 2077 with Mod Organizer 2 as the game executes are now in the Overwrite root directory and not where they should be. This can be fixed by verifying the Cyberpunk 2077 game in Steam. You might be able to salvage your build if the Overwrite root directory is not cluttered with too many mods. Otherwise, it is new build time. This current build is not duplicating that behavior or at least not yet. I am curious what causes that to happen. (**Update:** All of this is due to RooBuilder not being configured correctly.**)**

File Directory Mapping Summary

This		**Should be this**
/archive/pc/mod	>	/pc/mod
/engine	>	/root/engine
/bin	>	/root/bin
/red4ext	>	/root/red4ext
/r6	>	/root/r6
/mods	>	/root/mods (These are REDmods)

Legacy Mods and REDmods

A REDmod will contain directories that start with /mods. They **should**. Some mod authors don’t always structure their mods correctly. If you see the presence of a info.json file in the mod, then it is a REDmod and you will need to organize it properly. The Material and Texture Override mod is an example of a REDmod. A REDmod install file will have (should have) the following structure.

/mods/<modname>

/mods/<modname>/info.json

/mods/<modname>/archive

/mods/<modname>/archive/modname.archive

A reminder that we will be placing all of this under a new /root/ directory when installing the mod.

In the end, either legacy mods or REDmods will work. If you are familiar with JSON files, you can convert legacy mods yourself to REDmods. Why would you convert a mod? See the section below on Mod Order. If you do this, don't forget that whenever you update the mod later. Otherwise, you will end up with both directories and file structures with the old and new versions of the mod and something that doesn't work as expected.

REDprelauncher

The REDprelaucher downloads the REDlauncher which is installed in C:\Users\username\AppData\Local\Programs\CD Projekt Red\REDlauncher which isn't too unusual for game companies.

REDmods and File Size

There has been a trend on Nexus to pack texture files within REDmods. While this may be okay for smaller texture mods where the mod must take priority over anything else, it does slow down the REDmod tool as it loads those textures to build your mod order and cache files. Will you notice it for a few hundred MB of textures? Probably not. Will you notice it with a few dozen or hundred GB of textures? Probably.

Unless the mods are super important - remember any mod, legacy or REDmod, will take priority over the base game - you can convert these mods back to legacy mods by moving the mods/modname/archives/archivename.archive into the folder /archive/pc/mod and discarding the rest of the mod. This works best for mods that are just texture mods.

Crashing the Game

If you have been using Mod Organizer 2 with Cyberpunk 2077, you may have noticed when closing the game that Mod Organizer 2 is locked due to REDEngineErrorReporting.exe. This doesn't mean that your game has crashed. How error reporting works at a high level in Cyberpunk 2077 is as follows:

If the game crashes:

- Cyberpunk.exe starts either directly by the player or spawned by the REDlauncher.exe process which was started by the REDprelauncher.exe.
- REDEngineErrorReporting.exe attaches to the Cyberpunk.exe process.
- Cyberpunk.exe process continues until an error crosses some threshold or it crashes.
- Cyberpunk.exe process exits.
- REDEngineErrorReporting.exe spawns the CrashReporter.exe and asks the user to send the crash data to CDPR. This is the "Whoa! Cyberpunk 2077 has flatlined!" message.
- REDEngineErrorReporting.exe process exits.
- Player decides whether to send the report or not.
- CrashReporter.exe exits.

If the game doesn't crash:

- Cyberpunk.exe starts either directly by the user or spawned by the REDlauncher.exe process which was started by the REDprelauncher.exe.

- REDEngineErrorReporting.exe attaches to the Cyberpunk.exe process.
- The player quits the game.
- Cyberpunk.exe process exits.
- REDEngineErrorReporting.exe process exits.

So, you will see the REDEngineErrorReporting.exe as the last process locking Mod Organizer 2 session before the session is unlocked. This is normal. Only if you see the "Whoa! Cyberpunk 2077 has flatlined!" error message does there mean there was an actual crash. Now, the threshold for what CDPR considers a crash could be very low such as a mod that tries to write to a log directory that doesn't exist or to a file that is read-only.

Bisecting the Game

Bisecting is a traditional method used for troubleshooting. The process is:

1. Disable half of your mods preferably from the most recently installed with core mods being last.
2. Does your game still crash?
    a. Yes
        i. Disable half of the remaining mods.
        ii. Go to 2.
    b. No
        i. Enable half of the mods that you just disabled.
        ii. Go to 2.

And through a process of elimination you should find the mod (or mods) that are causing the crash to occur. This works for Cyberpunk 2077 because the majority of mods that we use are rather simple mods. In other games like Skyrim where modding is much more complex as modders have much greater access to the game through Bethesda's Creation Kit tool, this approach is less likely to work. The act of disabling half your mods in Skyrim is also just as likely to cause your game to crash as a badly written mod.

It is somewhat nice that we can take a "heavily modded" Cyberpunk 2077 save game and load it in a build without mods and it will load with no problem. We should cherish these days while they last.

Microsoft Sysinternals (Advanced)

A warning first. These are a powerful set of tools (mainly the ones dealing with processes) for gathering data. However, many MMO gaming companies will not like you having these tools and will like you A LOT LESS for having them running while playing their game. They will assume that you are up to no good and may very well ban you from their game for using them. However for single player games (and software development in general), they are useful tools in gathering data for when applications don't work as expected. (If you want god-mode in CyberPunk 2077, just play a netrunner. It's quicker than using these tools.)

For Cyberpunk 2077, tools like Process Monitor can be very useful to see what cyberpunk.exe was doing right before it crashed to the desktop and can help you narrow down the cause of the crash. These tools do generate a lot of data and I do mean a lot. Experience with software development and operating systems is also kind of a requirement to fully understand the data that is being captured.

Mod Order

The (MO2) Problem

Mod Organizer 2 has historically excelled in how it handled mod order and mod conflicts. For Cyberpunk 2077, it's not quite as useful as everything is packed in archive files. So, Mod Organizer 2 will detect if Mod A and Mod B both have the same /pc/mods/modA.archive file and you can order those mods to determine which one that you want to win. However, it won't detect that Mod C with /pc/mod/modC.archive and Mod D with /pc/mod/modD.archive both contain different versions of the samurai_jacket.png texture that will conflict with each other. The winner (modA in this case) will depend on the alphabetical and numerical order of the mod archive file names. So, you might not get the results that you expect.

The Solution(s)

There is a new command line tool for WolvenKit that should detect file conflicts within mods.

For legacy mods, you are limited to adjusting their file archive file names. It is always possible that someone makes a plugin for Mod Organizer 2 to automate this. For example, it would prepend the archive name of every mod with its priority number. But don't hold your breath on it.

For REDmods, you can pass an argument to REDmod mod tool that contains the mod order. However, it only works for REDmods.

Ex. "`redmod deploy -root="D:\Program Files\(x86)\Steam\steamapps\common\Cyberpunk 2077" -mod=modB,modA,modC`"

What the REDmod mod tool does is takes the content of each mod's info.json file and concatenates it into a single file in r6\cache\modded\mods.json in the specified order. So, if your mod doesn't have a info.json file, then REDmod mod tool isn't going to do anything with it.

If you do not supply a mod order list to REDmod, it will order the mods alphabetically like it does with archive files. However, it is based on the mod name within its info.json file which may not be the same as the directory name of the mod in the mods directory.

Questions

Q: Start the game with the Cyberpunk 2077 Launcher (REDprelauncher.exe) or the Cyberpunk 2077 game (Cyberpunk2077.exe)?

A: During the initial build process, you should always use the Cyberpunk 2077 Launcher. The reason for this is that the REDmod.exe mod tool is part of the launcher. If you install or uninstall mods and you don't run the REDmod mod tool, you may get a very unstable build.

If you want to do a clean rebuild on your cache see the section below on cache errors.

A modding best practice is to install five to ten mods at a time. Launch the game to verify that everything works and then proceed to the next set. You don't want to be in the position where the game crashes and it could be any one out of the few hundred mods that you installed. Now, if they are mods like simple clothing mods, then those can be installed in larger numbers all at once.

Once your mod build is stable, then you can skip to just using the Cyberpunk 2077 game itself.

Q: What wins between a conflict between a legacy mod and a REDmod?

A: Hey, that is a pretty good question, probably the REDmod.

Q: What do these red version numbers mean in Mod Organizer 2?

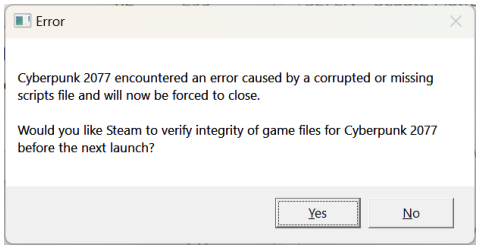
A: This means that the mod may be potentially out of date. The reason for "potentially" is that mod authors don't often keep their version numbers on their mod pages in sync. For example, with Cyber Engine Tweaks, the mod file is flagged with version 1.24.1, but the mod itself is flagged with version 1.24. So, Mod Organizer2 is going to mark it as outdated. It's just a learning experience to know when you need to worry about the version numbers and when you don't need to worry about version numbers.

Mods installed from local archives (i.e. files you obtained from outside of Nexus or where manual download from Nexus) may have no version data. So, you will need to manage that on your own.

Common Errors

r6/cache errors

Example:



Explanation: You have one or more corrupt files in either Cyberpunks r6/cache folder, the Overwrite folder or a mod that has copies of those files.

Solution: This is a little batch file that will delete your cache files, replace them with the original copies of the files and then run REDmod to update the cache and mod order rather than requiring the user to manually run the game launcher. It does require a little preparation.

1. Create a directory D:\Games\Scripts.

   This is a location where you can store scripts for various games that won't be lost if for some reason Cyberpunk 2077 mod directory is deleted.

2. Create a directory D:\Games\Cyberpunk 2077\r6cache

   This is a folder to store a good copy of the r6\cache files that redmod tool needs and thus Cyberpunk 2077 needs to start.

3. Create a directory D:\Games\Cyberpunk 2077\r6cache\modded

   This is a folder to store a copy of the r6\cache\modded files that Cyberpunk 2077 needs to start.

4. Create an empty file called tweakdb.bin in D:\Games\Cyberpunk 2077\r6cache\modded. Yes, REDmod throws an error if this file doesn't exist. It should just throw a warning and create it anew.
5. Delete the existing Cyberpun2077's r6\cache\tweakdb.bin and r6\cache\final.redscripts files or just delete the entire r6\cache directory.
6. Verify CyberPunk 2077's local files in Steam to get a good copy of the r6\cache files.
7. Copy Cyberpun2077's r6\cache\tweakdb.bin and r6\cache\final.redscripts files into D:\Games\Cyberpunk 2077\r6cache. **Whenever the game is updated by CDPR, replace these files with the new versions.**
8. Create the following batch file in NotePad++ (or your editor of choice) and save it to D:\Games\Scripts as CCR_Reset_MO2.bat.

```
@ECHO OFF


REM Cyberpunk 2077 Corrupted Cache Reset
REM By


SET GAME_PATH=%1
SET BASE_DIR=%2


IF %BASE_DIR% == "" (
   ECHO:
   ECHO The base directory argument appears to be missing.
   ECHO:
   GOTO usage
)


IF %GAME_PATH% == %BASE_DIR% (
   ECHO:
   ECHO The game directory and mod base directory appear to
be the same.
   ECHO:
   GOTO usage
)
```

```
REM Delete the r6/cache directory and files from the game
install, overwrite folder and my Cache mod (which I no longer
use but serves as an example if you have a similar
configuration).

REM Mods with spaces in their names are a pain and will need
quotes to surround them.


RMDIR /Q /S %GAME_PATH%\r6\cache

RMDIR /Q /S %BASE_DIR%\overwrite\Root\r6\cache

RMDIR /Q /S %BASE_DIR%\mods\"DB, Cache, INI and Log
Files\Root"\r6\cache


REM Create the r6/cache directory and restore files from the
game install.

REM Update needed to create the cache\modded directory.


MKDIR %GAME_PATH%\r6\cache

MKDIR %GAME_PATH%\r6\cache\modded

COPY /Y %BASE_DIR%\r6cache\final.redscripts
%GAME_PATH%\r6\cache\

COPY /Y %BASE_DIR%\r6cache\tweakdb.bin %GAME_PATH%\r6\cache\

COPY /Y %BASE_DIR%\r6cache\modded\tweakdb.bin
%GAME_PATH%\r6\cache\modded\


REM Now update the cache and mod order with REDmod.


%GAME_PATH%\tools\redmod\bin\redmod.exe deploy
-root=%GAME_PATH%


EXIT


:usage

ECHO:
```

```
ECHO Cyberpunk 2077 Corrupted Cache Reset

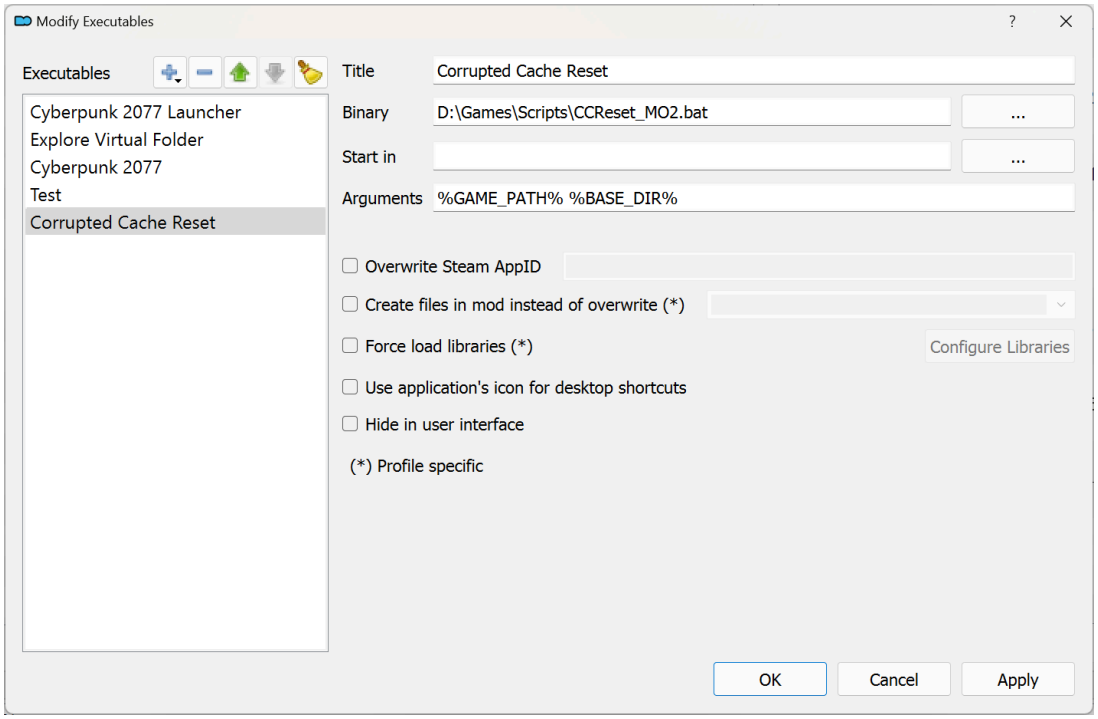
ECHO Is meant to be used within Mod Organizer 2 rather than
on its own.

ECHO Usage: CCReset_MOD "%%GAME_PATH%%" "%%BASE_DIR%%"

ECHO You will need to use the actual values for those
variables rather than the variables names.


EXIT
```

9. Add this batch file as an executable in Mod Organizer 2.



   a. Click the plus sign and click Add from file…
   b. Navigate to the D:\Games\Scripts directory.
   c. Select the CCReset_MO2.bat file.
   d. Click Open
   e. Change the Title to “Corrupted Cache Reset”
   f. Change the Arguments to "D:\SteamLibrary\steamapps\common\Cyberpunk 2077" "D:\Games\Cyberpunk 2077". There is a space between those two values.

   The first value is your game path and the second value is your mod path. Wouldn’t it be nice if we could just use the variables that Mod Organizer uses for those, but we apparently cannot.

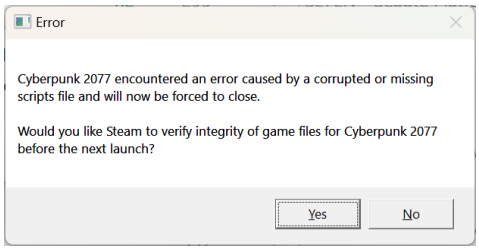
   g. Click OK.

10. You can now select it from the start menu in Mod Organizer 2 to delete your cache and then rebuild it.

The downside to this batch file is that it does wipe out the mod order file r6/cache/modded/mods.json. But since we are Mod Organizer 2 users, we aren't making full use of that file. Yet. If you run this script outside of Mod Organizer 2 with the proper game and base directory, it will build, but won't pick up any mods that you added to the game with Mod Organizer 2.

missing file errors

Example:



Explanation: It looks like the majority of these are due to issues with the r6/cache. I don't think I have seen a case yet due to missing bin files for example. I have seen one case of a user reporting files missing in archives/pc/content. A certain building interior would be all black as the game cannot load missing textures. But no information was given about their configuration.

Solution: Verify files to check everything outside of r6/cache and see the r6/cache section for rebuilding the cache.

redmod.exe deploy errors (mainly archives and scripts)

Example

Explanation: These are usually a result of trying to rebuild a corrupted cache where only some files are missing.

Solution: It's simpler to just delete the existing cache and rebuild it. See r6/cache.

redscript compiler errors

Example

Explanation: In work.

Solution: Unfortunately, there isn't a simple solution to these errors.

1. Verify that all core mods and mods related to the error are up to date.
2. Verify that all core mods and mods related to the error are installed correctly.
3. Clear and rebuild cache. See r6/cache error to rebuild the cache which includes scripts.
4. Verify that one or two mods are not conflicting with each other. Disable one of the mods with the compiler error and test. Does the error still occur?

***.cpp errors**

Example

Explanation: These errors are a result of causing a crash with the game or one of the added libraries (*.dll's) through RED4ext. As a result of the crash, you are seeing a dump that references files from either the game developer's or the mod author's build environment. None of the file references will actually point to files on your system.

Solution: There is no one simple solution to these errors. Updating video card and sound drivers. Making sure Microsoft Visual C++ redistributables are installed. Making sure you meet the hardware requirements for the game. Making sure Windows is up to date. All there are some of the more common solutions. Unlike the other errors, it can be difficult to replicate these.