

Forum:[Hack Forums](#)

Thread: [SFED-AES-256 // Encrypt & Decrypt Any File // FREE Source Code](#)

Hackers - February 20, 2024, 7:21 pm

SFED-AES-256

Version 0.0.1

SFED-AES-256 is the newest iteration of my program collection "SFED" (Single File Encryption & Decryption). This is a series of tools I am creating for encryption and decryption for C++/C. This newest version features a mini aes.h/aes.c library that is fully compatible with both C and C++ using standard libs. This is a great way to bring in AES128, AES196, or AES256 to any project without requiring on other dependencies.

Encrypt or decrypt files using AES-256.

Generate a random AES key for encryption.

Allow specifying a key via file, directly as a string, or generate a new one.

Output encrypted/decrypted content to a specified file.

Include the initialization vector (IV) in the encrypted file for decryption

Usage

Download

Windows (.EXE) :

Code <https://www.mediafire.com/file/vihn3x2jioob36n/SFED-AES-256.exe/file>

Source Code

Code <https://github.com/LNodesL/SFED-AES-256>

Would you rather use XOR instead of AES256?

SFEDXOR 0.0.2 : <https://hackforums.net/showthread.php?tid=6264286>

Disclaimer

This is not a crypter. It is a single file encryption & decryption tool. You can use this to lock down your own files. If you are using scripts you can loop through multiple files of yours to encrypt or decrypt them.

Spoiler ([Click to View](#))

To make SFED-AES-256 I utilized an LLM to do two things:

- Take the mini AES library & encrypt/decrypt functions from another project of mine
- Utilize the existing input/output formats of "SFEDXOR 0.0.2", a previous SFED tool of mine

Truth - February 20, 2024, 7:29 pm

Nice, doubt AI will ever crack AES encryption algorithm, can't even crack a captcha yet.

Hackers - February 21, 2024, 4:37 am

(Feb 20, 2024 08:29 PM)Truth Wrote: [\[CITE\]](#) Nice, doubt AI will ever crack AES encryption algorithm, can't even crack a captcha yet.

I doubt it too, but not going to say I'd be surprised if it did. There were rumors it could crack some AES algorithms in past, but haven't heard anything about that in a while.

I think I'm going to spin up a folder encryption program today possibly. I'll use SFED-AES-256 here as the base but it'll encrypt every file in a folder. Just need to figure out how I want to handle re-constructing the folder on decryption but other than that should be an easy program to push out. I have recursive file/folder loading in another program, just need to merge that with SFED basically.

Dirty - February 21, 2024, 4:53 am

(Feb 21, 2024 05:37 AM)nodes Wrote: [\[CITE\]](#) I doubt it too, but not going to say I'd be surprised if it did. There were rumors it could crack some AES algorithms in past, but haven't heard anything about that in a while.

I think I'm going to spin up a folder encryption program today possibly. I'll use SFED-AES-256 here as the base but it'll encrypt every file in a folder. Just need to figure out how I want to handle re-constructing the folder on decryption but other than that should be an easy program to push out. I have recursive file/folder loading in another program, just need to merge that with SFED basically.

I don't really feel like that's an AI issue to begin with, aes256 just requires too much compute. I don't understand how an AI could help with this. Quantum computing is a different discussion I guess, although I think we already have quantum resistant algorithms

ByteShift - February 21, 2024, 7:35 pm

(Feb 21, 2024 05:53 AM)Dirtybox Wrote: [\[CITE\]](#) I don't really feel like that's an AI issue to begin with, aes256 just requires too much compute. I don't understand how an AI could help with this. Quantum computing is a different discussion I guess, although I think we already have quantum resistant algorithms

EDR's and malware analysis is honestly the real issue and when it comes down to it runtime is what is important, they're vm'ing software watching all the registry modifications, files dropped or moved, etc. Most malware isn't working 100% in memory.

as of recently hash identification or signature based identification has kinda faded away now the cloud EDR's are essentially machine learning algorithms and VM's and debuggers, it's actually insane how some of the more highly funded ones work.

In terms of encryption it will ALWAYS be cat and mouse, when you are talking malware at least. For regular obfuscation for enterprise software or whatever these methods are perfect, may throw up false positives depending on the way it's implemented.

Polymorphism seems to be the new wave of cryptography and the only viable methods for long term fud (in my opinion)

fahrenheit451 - March 10, 2024, 12:07 pm

Thank you very much!

Hackers - March 11, 2024, 6:35 am

(Mar 10, 2024 12:07 PM)fahrenheit451 Wrote: [\[CITE\]](#) Thank you very much!

You are welcome, come back here to reply if you have any questions!

dtb1031231 - March 27, 2024, 8:00 am

Thank you so much this helped a lot

salvator - March 27, 2024, 8:26 am

(Feb 20, 2024 08:21 PM)nodes Wrote: [\[CITE\]](#) SFED-AES-256
Version 0.0.1

SFED-AES-256 is the newest iteration of my program collection "SFED" (Single File Encryption & Decryption). This is a series of tools I am creating for encryption and decryption for C++/C. This newest version features a mini aes.h/aes.c library that is fully compatible with both C and C++ using standard libs. This is a great way to bring in AES128, AES196, or AES256 to any project without requiring on other dependencies.

Encrypt or decrypt files using AES-256.

Generate a random AES key for encryption.

Allow specifying a key via file, directly as a string, or generate a new one.

Output encrypted/decrypted content to a specified file.

Include the initialization vector (IV) in the encrypted file for decryption

Usage

Download

Windows (.EXE) :

Code <https://www.mediafire.com/file/vihn3x2jioob36n/SFED-AES-256.exe/file>

Source Code

Code <https://github.com/LNodesL/SFED-AES-256>

Would you rather use XOR instead of AES256?

SFEDXOR 0.0.2 : <https://hackforums.net/showthread.php?tid=6264286>

Disclaimer

This is not a crypter. It is a single file encryption & decryption tool. You can use this to lock down your own files. If you are using scripts you can loop through multiple files of yours to encrypt or decrypt them.

Spoiler [\(Click to View\)](#)

To make SFED-AES-256 I utilized an LLM to do two things:

- Take the mini AES library & encrypt/decrypt functions from another project of mine
 - Utilize the existing input/output formats of "SFEDXOR 0.0.2", a previous SFED tool of mine
- ty man

Hackers - March 27, 2024, 8:38 am

(Mar 27, 2024 08:26 AM)salvator Wrote: [\[CITE\]](#) ty man

No problem. Let me know what you think or if you have any questions at all

Ahmad Mouais - March 27, 2024, 10:28 am

I tried your code on my bitcoin wallet.dat and it didn't work

Hackers - March 27, 2024, 10:29 am

(Mar 27, 2024 10:28 AM)Ahmad Mouais Wrote: [\[CITE\]](#) I tried your code on my bitcoin wallet.dat and it didn't work

Can you share the commands you used to do it? I'll take a look and help you out.

Thanks

Ahmad Mouais - March 27, 2024, 10:36 am

(Mar 27, 2024 10:28 AM)Ahmad Mouais Wrote: [\[CITE\]](#) I tried your code on my bitcoin wallet.dat and it didn't work

Sure bro

```
./SFED-AES-256 -i wallet.dat -o decrypted_wallet.dat -m decrypt
```

(Mar 27, 2024 10:28 AM)Ahmad Mouais Wrote: [\[CITE\]](#) I tried your code on my bitcoin wallet.dat and it didn't work

(Mar 27, 2024 10:29 AM)nodes Wrote: [\[CITE\]](#) Can you share the commands you used to do it? I'll take a look and help you out.

Thanks

```
./SFED-AES-256 -i wallet.dat -o decrypted_wallet.dat -m decrypt
```

Hackers - March 27, 2024, 10:46 am

(Mar 27, 2024 10:36 AM)Ahmad Mouais Wrote: [\[CITE\]](#) Sure bro

```
./SFED-AES-256 -i wallet.dat -o decrypted_wallet.dat -m decrypt
```

```
./SFED-AES-256 -i wallet.dat -o decrypted_wallet.dat -m decrypt
```

Did you use the encrypt function or did you only use decrypt

Remember, your wallet.dat file is most likely a combination of data - not just 1 AES string. This tool only works for pure AES 256 hash encryption & decryption.

Also, you need to know the key for it

example - you would add this to end of your decrypt command, with path to your saved key file:

Code -k keyfile.aes

or you can use -ks for a keystring inline.

If you are trying to crack a wallet you don't own without a key, my tool will never work for that.

north0 - May 18, 2024, 11:41 am

Thanks for the contribution to the community. Keep up the work

Hackers - June 8, 2024, 5:20 pm

(May 18, 2024 11:41 AM)north0 Wrote: [\[CITE\]](#) Thanks for the contribution to the community. Keep up the work

thank you north0, have a nice time! Let me know if you have any questions

Not477 - July 6, 2024, 7:18 am

Hello.

Can I decrypt with this program a zip file?

I try to decrypt a zip file and I got another file, with no extension, what can I do with the new file that I got.

Thank you.

Hackers - July 6, 2024, 7:23 am

(Jul 6, 2024 07:18 AM)Not477 Wrote: [\[CITE\]](#) Hello.

Can I decrypt with this program a zip file?

I try to decrypt a zip file and I got another file, with no extension, what can I do with the new file that I got.

Thank you.

ah, make sure you specifically set the extension when you encrypt/decrypt so it converts properly. If it's truly decrypted and in the original form, then you can just put the ext back in the filename and try to open.

Also something to keep in mind:

- this is for you to encrypt/decrypt your own files and data, not to decrypt someone else's if that's what you're trying to do.

secret_code - July 8, 2024, 12:18 am

(Feb 20, 2024 08:21 PM)nodes Wrote: [\[CITE\]](#) SFED-AES-256

Version 0.0.1

SFED-AES-256 is the newest iteration of my program collection "SFED" (Single File Encryption & Decryption). This is a series of tools I am creating for encryption and decryption for C++/C. This newest version features a mini aes.h/aes.c library that is fully compatible with both C and C++ using standard libs. This is a great way to bring in AES128, AES196, or AES256 to any project without requiring on other dependencies.

Encrypt or decrypt files using AES-256.

Generate a random AES key for encryption.

Allow specifying a key via file, directly as a string, or generate a new one.

Output encrypted/decrypted content to a specified file.

Include the initialization vector (IV) in the encrypted file for decryption

Usage

Download

Windows (.EXE) :

Code <https://www.mediafire.com/file/vihn3x2jioob36n/SFED-AES-256.exe/file>

Source Code

Code <https://github.com/LNodesL/SFED-AES-256>

Would you rather use XOR instead of AES256?

SFEDXOR 0.0.2 : <https://hackforums.net/showthread.php?tid=6264286>

Disclaimer

This is not a crypter. It is a single file encryption & decryption tool. You can use this to lock down your own files. If you are using scripts you can loop through multiple files of yours to encrypt or decrypt them.

Spoiler ([Click to View](#))

To make SFED-AES-256 I utilized an LLM to do two things:

- Take the mini AES library & encrypt/decrypt functions from another project of mine
- Utilize the existing input/output formats of "SFEDXOR 0.0.2", a previous SFED tool of mine

nice job, looks good at first glance, using A256 is a good option instead of xor(as you mentioned it), i wouldn't use xor for anything LOL :), but yeah, what happens in case the process gets corrupted or something is wrong? do you actually provide the user with a backup copy of the data first? maybe a quick backup before trying to operate on. additionally you might also generate a hash from the result just in case...

overall, this util is good man... keep it up, i also like the fact you're using c++ and not some college school PL like python :) so it is obvious that you're trying to push your limits over, i was also creating encryption related utilities in the past, but i've made so many(crypters included) that i got bored eventually of them :), i can see you're into cryptography some way or you're still finishing to learn something maybe? how's your journey being so far? what was the most difficult topic you're covered?

Hackers - July 8, 2024, 6:16 am

(Jul 8, 2024 12:18 AM)secret_code Wrote: [\[CITE\]](#) nice job, looks good at first glance, using A256 is a good option instead of xor(as you mentioned it), i wouldn't use xor for anything LOL :), but yeah, what happens in case the process gets corrupted or something is wrong? do you actually provide the user with a backup copy of the data first? maybe a quick backup before trying to operate on. additionally you might also generate a hash from the result just in case...

overall, this util is good man... keep it up, i also like the fact you're using c++ and not some college school PL like python :) so it is obvious that you're trying to push your limits over, i was also creating encryption related utilities in the past, but i've made so many(crypters included) that i got bored eventually of them :), i can see you're into cryptography some way or you're still finishing to learn something maybe? how's your journey being so far? what was the most difficult topic you're covered?

The answer is no, there is nothing in place for backups or to deal with that issue. It is a flaw for sure.

I was researching encryption/decryption in general for C/C++ because it's something I've done in other languages, but I wanted to learn as much as possible. So from here I went on to mess around with crypters but I got pretty bored - I never really needed to use one for anything so it was just to practice and learn C++ and I moved on to other niches w/ the language.

The most complex thing I did with C++ was probably deconstructing peb/teb, IAT, data sections, etc and re-constructing a PE in memory without dropping a tmp file at all, along with using my own ntldr (and similar) functions, compiletime encrypt / runtime decrypt, and other methods. I had a lot of fun learning how to do that.

secret_code - July 8, 2024, 6:57 am

(Jul 8, 2024 06:16 AM)nodes Wrote: [\[CITE\]](#) The answer is no, there is nothing in place for backups or to deal with that issue. It is a flaw for sure.

I was researching encryption/decryption in general for C/C++ because it's something I've done in other languages, but I wanted to learn as much as possible. So from here I went on to mess around with crypters but I got pretty bored - I never really needed to use one for anything so it was just to

practice and learn C++ and I moved on to other niches w/ the language.

The most complex thing I did with C++ was probably deconstructing peb/teb, IAT, data sections, etc and re-constructing a PE in memory without dropping a tmp file at all, along with using my own ntdll (and similar) functions, compiletime encrypt / runtime decrypt, and other methods. I had a lot of fun learning how to do that.

sounds cool, what's your goal atm? learning more cryptography? btw you should try writing your own ransomware(education purpose only) to learn cryptography more efficiently, im suggesting this because you seem to want to learn cryptography and something malware related as well(correct me if im wrong), well.... write your own ransomware, that will get you pretty setup with what you're trying to learn... i promise ;)

Hackers - July 8, 2024, 8:08 am

(Jul 8, 2024 06:57 AM)secret_code Wrote: [\[CITE\]](#) sounds cool, what's your goal atm? learning more cryptography? btw you should try writing your own ransomware(education purpose only) to learn cryptography more efficiently, im suggesting this because you seem to want to learn cryptography and something malware related as well(correct me if im wrong), well.... write your own ransomware, that will get you pretty setup with what you're trying to learn... i promise ;)

I have a pretty good grasp of cryptography in general, just haven't applied much to C++ outside of crypters, SFED, and a few others I made.

I made a personal locker program that allows you to encrypt a folder recursively, but it's not 'ransomware'.

To be honest I'm not really looking to do much malware related software at all. If anything I would create an anti- software to detect or prevent ransomware, but I have no interest in making the ransomware itself. Now, a program that can lock down a drive and encrypt it, or erase it from a remote server? That's whitehat and a normal security measure for protected devices.

Stanley - November 19, 2024, 12:54 pm

I'm Stanley HF's helper bot. The OP (original poster) of this thread wants everyone to see this page and has elected to bump it to the top of the showthread listing. Since HF doesn't want a ton of LQP (Low Quality Posts) this is the best method of bumping your content. There is a small ?yte fee for each bump. Any OP can bump any of their threads on HF from the linked icon on the top of their first post.

If you haven't already read the first post of this thread we encourage you to do so.

Thanks,
Stanley

Hackers - March 12, 2026, 9:03 pm

This program still works & the source is an easy way to learn more about encryption & decryption.

kriminaltetic - March 14, 2026, 9:02 am

(Feb 20, 2024 08:21 PM)Hackers Wrote: [\[CITE\]](#) SFED-AES-256
Version 0.0.1

SFED-AES-256 is the newest iteration of my program collection "SFED" (Single File Encryption & Decryption). This is a series of tools I am creating for encryption and decryption for C++/C. This newest version features a mini aes.h/aes.c library that is fully compatible with both C and C++ using standard libs. This is a great way to bring in AES128, AES196, or AES256 to any project without requiring on other dependencies.

Encrypt or decrypt files using AES-256.

Generate a random AES key for encryption.

Allow specifying a key via file, directly as a string, or generate a new one.

Output encrypted/decrypted content to a specified file.

Include the initialization vector (IV) in the encrypted file for decryption

Usage

Download

Windows (.EXE) :

Code <https://www.mediafire.com/file/vihn3x2jioob36n/SFED-AES-256.exe/file>

Source Code

Code <https://github.com/LNodesL/SFED-AES-256>

Would you rather use XOR instead of AES256?

SFEDXOR 0.0.2 : <https://hackforums.net/showthread.php?tid=6264286>

Disclaimer

This is not a crypter. It is a single file encryption & decryption tool. You can use this to lock down your own files. If you are using scripts you can loop through multiple files of yours to encrypt or decrypt them.

Spoiler ([Click to View](#))

To make SFED-AES-256 I utilized an LLM to do two things:

- Take the mini AES library & encrypt/decrypt functions from another project of mine
- Utilize the existing input/output formats of "SFEDXOR 0.0.2", a previous SFED tool of mine

fix bro broke

Hackers - March 14, 2026, 9:17 am

(Mar 14, 2026 09:02 AM)krimaltetic Wrote: [\[CITE\]](#) fix bro broke

I can take a look, but can you please explain what error you are receiving? Happy to provide an update or help you personally... Thanks for flagging this problem to me.

krimaltetic - March 14, 2026, 9:48 am

(Mar 14, 2026 09:17 AM)Hackers Wrote: [\[CITE\]](#) I can take a look, but can you please explain what error you are receiving? Happy to provide an update or help you personally... Thanks for flagging this problem to me.

(Mar 14, 2026 09:17 AM)Hackers Wrote: [\[CITE\]](#) I can take a look, but can you please explain what error you are receiving? Happy to provide an update or help you personally... Thanks for flagging this problem to me.

1. It's now detecting threats. 2. It's having output issues; it's getting old.

CL1B - March 14, 2026, 11:13 am

(Feb 20, 2024 08:29 PM)Truth Wrote: [\[CITE\]](#) Nice, doubt AI will ever crack AES encryption algorithm,

can't even crack a captcha yet.
Oh yeah?

<https://hackforums.net/showthread.php?tid=6307937>

<https://hackforums.net/showthread.php?tid=6308695>

by [Yakless](#)

Hackers - March 14, 2026, 11:16 am

(Mar 14, 2026 09:48 AM)kriminaltetic Wrote: [\[CITE\]](#) 1. It's now detecting threats. 2. It's having output issues; it's getting old.

You are making a pretty wild accusation. I'm guessing you compiled the source and then ran a scan on it and it's showing normal, safe warnings about the program using encryption. If you used it to encrypt your own data and then scanned your own encrypted bin/bytes, then it's possible it could get flagged for high entropy. However I can tell you 100% that there is no chance my source or original program here is a threat.

If you do have any bigger issue than what I described above, you should immediately look into your own firewall settings and make sure you have them fully enabled. For example, if you are compiling my code and your final program has malware in it, then there's a good chance your dev machine has been infected with malware that packs itself into final programs during builds/compile time. So check that out if that's the case, you could be unknowingly creating new malware if you're compiling my safe source & ending up with something bad.

I am a trusted member on this forum. I am not distributing anything malicious. There is no hidden, packaged, downloaded, streamed, or crypted malware in this project from me whatsoever. I just want to say that 100% so it's clear I'm not entertaining your accusation here whatsoever.

On my last commit, I updated the version number to 0.0.1

<https://github.com/LNodesL/SFED-AES-256/commit/a6f8409e6b2415c1088d32c3b3de10b6cd6a2738>

You can see that I never updated main.cpp once after the original commit, and the code is clean my guy! All it does is encrypt your input with the settings and functionality described in the README, exactly as-is. You are free to prove me wrong, but once you review the code you will see that there is nothing malicious, and no code that is even capable of running anything, downloading anything, or doing anything out of the ordinary on your computer. It's 100% safe, and there are not very many lines of code to review (143 in main program):

<https://github.com/LNodesL/SFED-AES-256/commit/77ed85448c62acb23a408fd69c68139b998a0658#diff-608d8de3fba954c50110b6d7386988f27295de845e9d7174e40095ba5efcf1bb>

Look for yourself, the program that is out today is the same exact program I released originally. SFED-AES-256 has been used by a number of other users and I have used it in my own projects too.

Git history for main.cpp:

<https://github.com/LNodesL/SFED-AES-256/commits/main/main.cpp>

Lastly, the MediaFire link doesn't even appear to be available anymore. I am still trying to recover a lot of old projects in general, I lost my primary Windows rig unfortunately and lost 100s of projects. Luckily this one is on a few git platforms. I stopped paying for MF ages ago though. I'll have to recompile and share a new link soon in this thread.

Anyways... You are free to tag staff if you think I am doing something weird. I will tag @Charlie Sheen@ and @The Pact@ who both would probably deal with me quickly if I was doing anything malicious at all.

We already have [CL1B](#) here, but lets get this going lol - I'll also tag VIBE group owner @7@, and all these others who can investigate the code or compile it and RE my program to prove my innocence - [Vinavil](#) [HerrWinston](#) [InspectorGadget](#) @Au Jus Demon@ [P??sonl?af](#) [loadman](#) [Klonopin](#) [ttbr0](#)
I do agree this is an old program/codebase though...

~ over & out, thanks for free bump my guy ~

kriminaltetic - March 14, 2026, 11:18 am

(Mar 14, 2026 11:16 AM)Hackers Wrote: [\[CITE\]](#) You are making a pretty wild accusation. I'm guessing you compiled the source and then ran a scan on it and it's showing normal, safe warnings about the program using encryption. If you used it to encrypt your own data and then scanned your own encrypted bin/bytes, then it's possible it could get flagged for high entropy. However I can tell you 100% that there is no chance my source or original program here is a threat.

If you do have any bigger issue than what I described above, you should immediately look into your own firewall settings and make sure you have them fully enabled. For example, if you are compiling my code and your final program has malware in it, then there's a good chance your dev machine has been infected with malware that packs itself into final programs during builds/compile time. So check that out if that's the case, you could be unknowingly creating new malware if you're compiling my safe source & ending up with something bad.

I am a trusted member on this forum. I am not distributing anything malicious. There is no hidden, packaged, downloaded, streamed, or crypted malware in this project from me whatsoever. I just want to say that 100% so it's clear I'm not entertaining your accusation here whatsoever.

On my last commit, I updated the version number to 0.0.1

<https://github.com/LNodesL/SFED-AES-256/commit/a6f8409e6b2415c1088d32c3b3de10b6cd6a2738>

You can see that I never updated main.cpp once after the original commit, and the code is clean my guy! All it does is encrypt your input with the settings and functionality described in the README, exactly as-is. You are free to prove me wrong, but once you review the code you will see that there is nothing malicious, and no code that is even capable of running anything, downloading anything, or doing anything out of the ordinary on your computer. It's 100% safe, and there are not very many lines of code to review (143 in main program):

<https://github.com/LNodesL/SFED-AES-256/commit/77ed85448c62acb23a408fd69c68139b998a0658#diff-608d8de3fba954c50110b6d7386988f27295de845e9d7174e40095ba5efcf1bb>

Look for yourself, the program that is out today is the same exact program I released originally. SFED-AES-256 has been used by a number of other users and I have used it in my own projects too.

Git history for main.cpp:

<https://github.com/LNodesL/SFED-AES-256/commits/main/main.cpp>

Lastly, the MediaFire link doesn't even appear to be available anymore. I am still trying to recover a lot of old projects in general, I lost my primary Windows rig unfortunately and lost 100s of projects. Luckily this one is on a few git platforms. I stopped paying for MF ages ago though. I'll have to recompile and share a new link soon in this thread.

Anyways... You are free to tag staff if you think I am doing something weird. I will tag @Charlie Sheen@ and @The Pact@ who both would probably deal with me quickly if I was doing anything malicious at all.

We already have [CL1B](#) here, but lets get this going lol - I'll also tag VIBE group owner @7@, and all these others who can investigate the code or compile it and RE my program to prove my innocence - [Vinavil](#) [HerrWinston](#) [InspectorGadget](#) @Au Jus Demon@ [P??sonl?af](#) [loadman](#) [Klonopin](#) [ttbr0](#)
I do agree this is an old program/codebase though...

~ over & out, thanks for free bump my guy ~

WD doesn't recognize the extracted EXE file—is that what you're saying?

Hackers - March 14, 2026, 11:31 am

(Mar 14, 2026 11:18 AM)kriminaltetic Wrote: [\[CITE\]](#) WD doesn't recognize the extracted EXE file—is that what you're saying?

I think I understand what's happening here... SFED-AES-256 is NOT a crypter. It is an input/output encryption tool that is for devs to use in other programs or as a standalone CLI tool for string encryption needs. It's a wrapper for encryption/decryption, not a standalone tool that you would use for "crypting" a Windows program. This is more like you're hashing your password and then unhashing it with a key instead of what you are saying. You are talking about a different concept that encrypts the runtime behavior of a PE so that it can still run on Windows after the "crypt".

SFED-AES-256 is not the tool you are looking for, that's not how it works.

You can't run the output from SFED-AES-256. If you encrypt binary, then you would need to decrypt it first and then run it (if using SFED tool). It really is not intended for what I think you are trying to do.