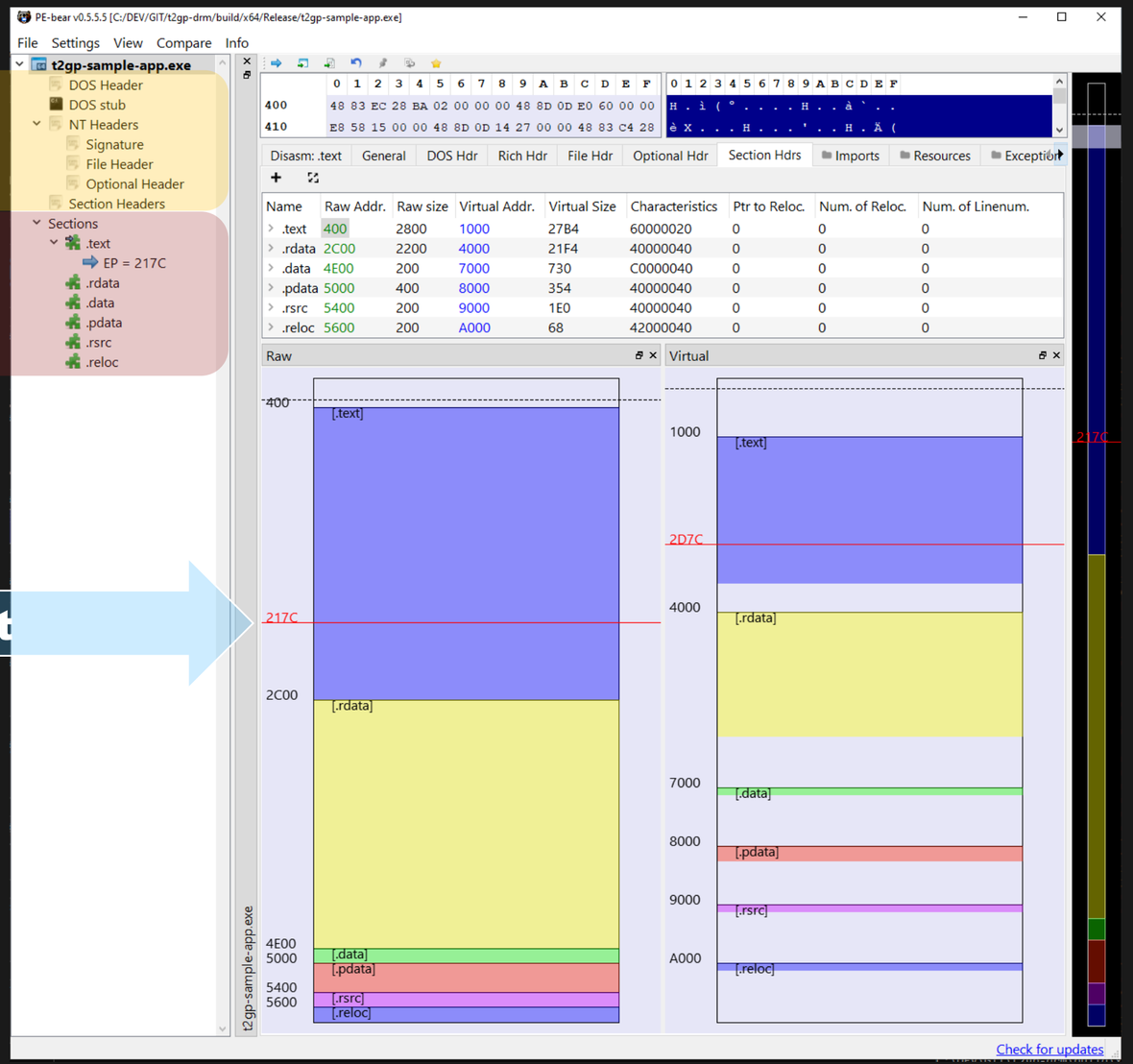
What‘s a Windows Executable file?

Windows x64 Portable Executable format (PE Format)

Headers

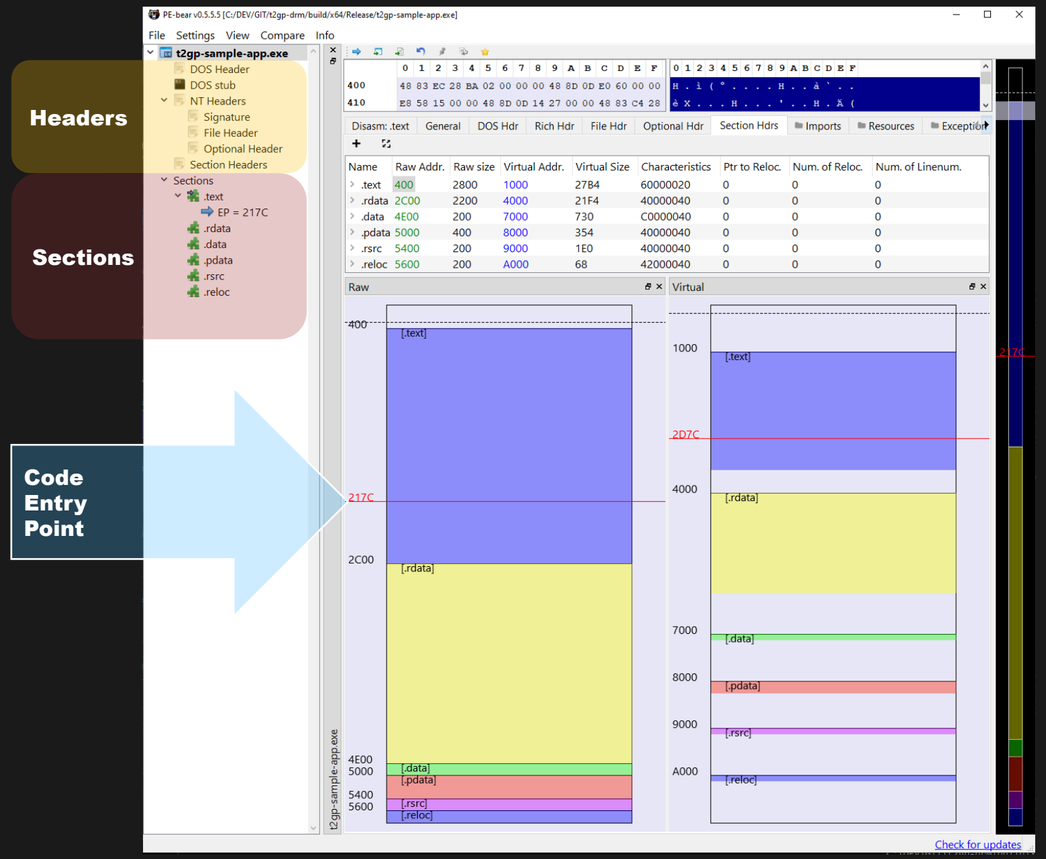
Sections

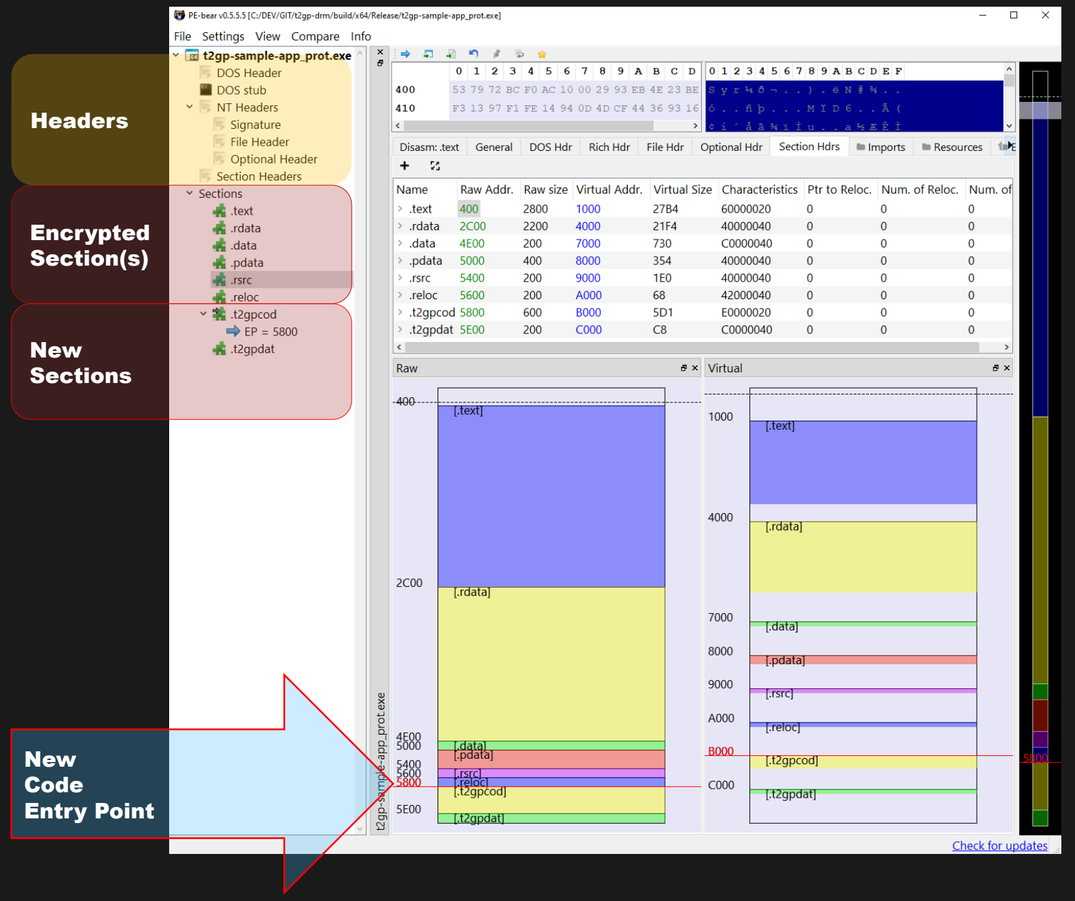
Code Entry Point



How to protect an executable?

Unprotected vs. Protected



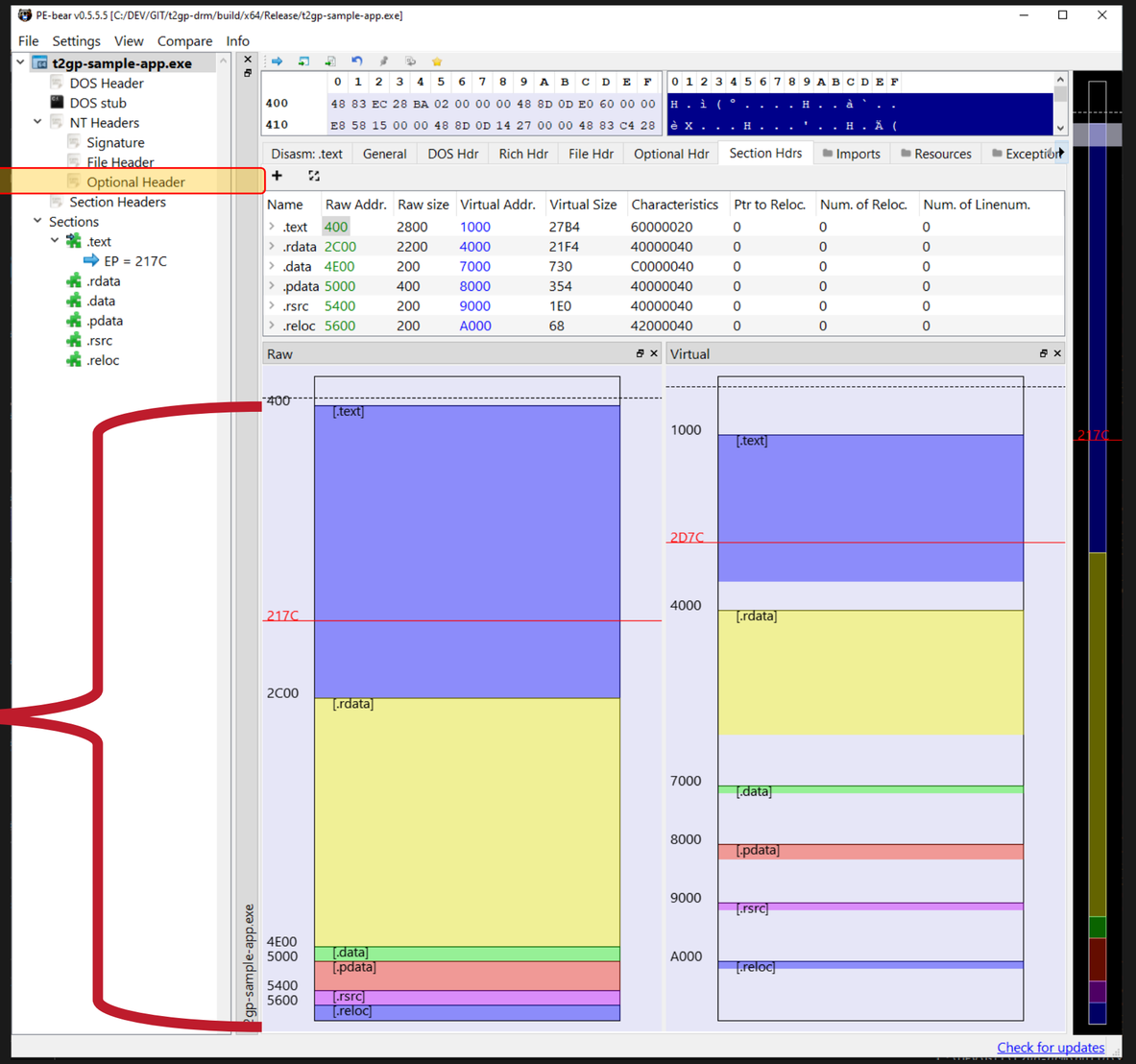


We will start all license veri□cation and decrypting at our **new Code Entry Point!**

TLS crash issues

Adding more Sections Bug

- Adding more sections can be easy
  - Add a new section entry in the header
  - Add the new section into the sections area
- But, what happens if the header is full after you added your new sections?
- It crashes, because the header size needs to be aligned properly!
- The □x:
  - Increase the header size and align it!
  - Add a new section entry in the header
  - Re-Align all existing sections RAW addresses
  - Add the new section into the sections area



TLS crash issues

Code Entry Point

Wait, we have a **Code Entry Point**, so we know were the OS starts calling code for execution?

- Yes and **No!**

Yes: The entry point is the starting point for the OS when the executable file has been loaded!

No: That's not the point of the first code execution of an executable file!

What's going on?

When you execute a file, the OS loader will load the image and doing the following stuff:

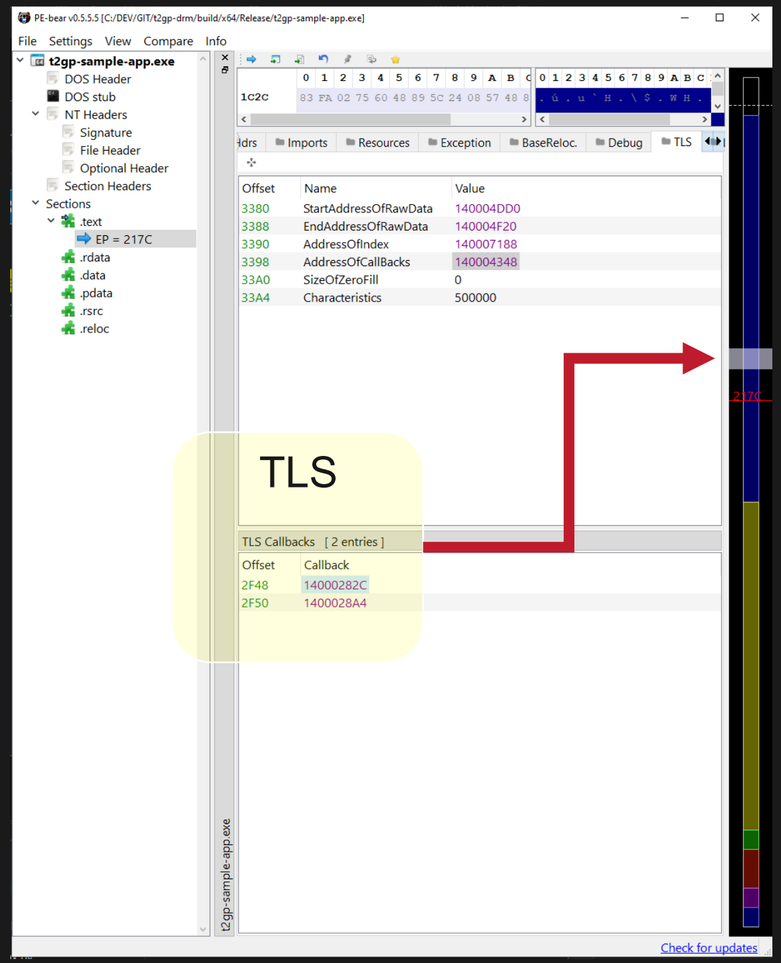
- `// 1. Setup OS APIs`
- `// 2. process relocations`
- `// 3. process image imports`
- `// 4. Initialize TLS`
- `// 5. Call the Code Entry Point`

TLS crash issues

Do not execute encrypted code…

Sounds □ne, so where is the issue?

- `// 4. Initialize TLS` -> will crash, because we are still in the encrypted executable!
- `// 5. Call the Code Entry Point`



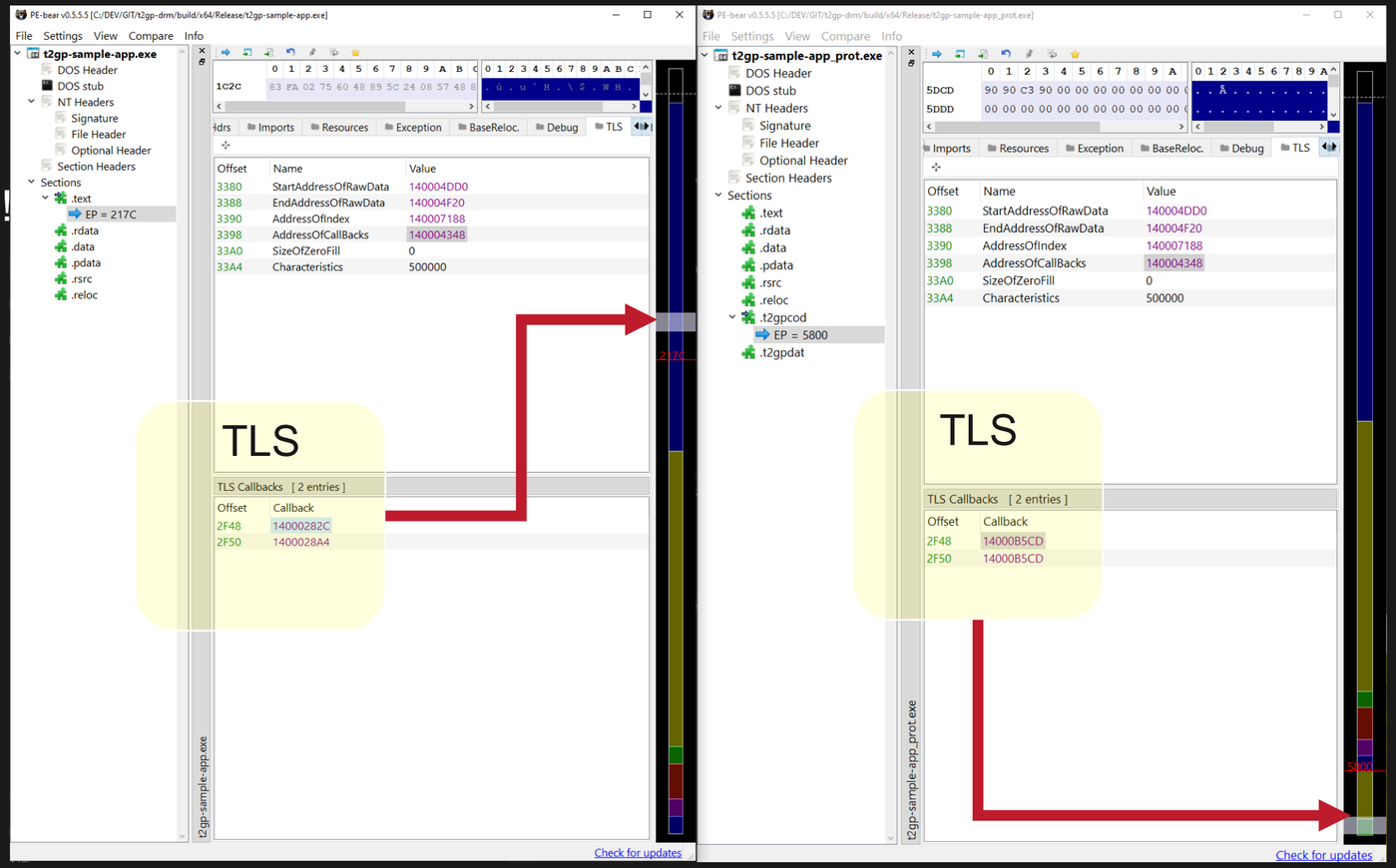
TLS crash issues

Do not execute encrypted code...

- We cannot execute anything in encrypted code
- And we cannot prevent the execution, without modifying the OS DLLs -> We don't want to hack the OS!

The fix:
- We must backup the original callback addresses
- Create new dummy callbacks in
  decrypted sections, that will do anything real,
  until we have decrypted the executable sections!

- Then we can restore the original callbacks and
  manually execute them, **before** calling
  the **original Code Entry Point**!



TLS crash issues

More to come...

- A few more issues may arise, stay tuned for more!