Cyber Games CTF

Beginner Challenges

Flag Checker

- I am given a python file. I can see that the variable names are nonsense. I decided to go thorugh the program and make names that make much more sense.
- I can see that everything is being encryped with simple ecnryption. Once I rename everything i will write a script to reverse the encryption.
- I simply reversed the code and got it to decrypt.
- **I should have not modified the code. I realized I was missing a good chunk of the information**
Decrypt File

Spider (rev)

Notes

- Windows Executble. It asks some questions and then sense for a debugger. It either is expecting a debugger to run or it won't run with one. I will look at it in ida.
- I am seeing a bunch of windows calls that use fiber objects. I need to figure out what these are.
- I think I see anti debugging with the NTGlobalFlag, I will have to do some research to figure out how to get passed it.
- I need to bypass the NtGlobal by setting the value to eax, I then need to get past the id debugger present section.
- I can also zero the rax register to bypass this. I guess I could patch the binary but I might avoid that for now.
- So far the challenge has me bypassing the isdebebugger present andntglobal check.
- There are some other reg values I need to modify as well, when i do those there is another anti-debug call I need to figure out.
- I now need to figureo out how to bypass all the antidebug interupts.
- I do hit int 3 at the very beginning. I think I need to pass this first.
- I noticed the hint was using output debug string. I need to bypass this. I realized if I put a break point in this function I get a software interupt.
- abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789!@#

KaTeX parse error: Expected 'EOF', got '#' at position 123: …XYZ0123456789!@#_

%^&*()_+-=/.;[]{}

- the matrix manipulation function gets called a bunch
-
- I am going to look at where the flag gets printed out. I wonder if anything interesting happens as I step through them.
- I am stuck and studying threads and fibers, I am not sure where else to look.
  After talking to a fellow intern I realized that I have solved the challenge but I was not paying attention to see the flag.
- I notice that having some break points at certain areas cause the int 3 and messes with the program. I should read into this.
- ** I found the flag in the Debug Console** I realized that I needed to look in the debugger and not standard out. I also noticed that it uses an antidebug techniques to actually check if we are using the debugger. In this case I don't actually want to avoid this call. Very intersting. **RTFM**

Emoticonsole

Notes

- I am given a python .pyc module and a program with the file extension .emo. I will look at the python module with visual studio code.
- The .emo file contains a bunch of emotes in a single line.
- The program was created with python 3.11. I was not able to get it to work with uncompyle6.
- I tried pycdc but the decompyle is imcomplete. I may do some more research on this.
- I think it is reading in the emotes as lines of code. I wonder if I need to write a compiler/intepreter.

```
*Possible Sudo Code*

file.read()
```

- I am going to see if I can manually change the magic byte in the header to spoof it so it uses the python version I am using
- `bytecode given: pip a7 0d`
- I am going to see if I can find the magic number and then edit the hex set at this value. I believe I need the magic for the .pyc byte code for python3.11
- 0D7A possible bytecode (earliest version of python 3.11)
- 0DA7 is what I think it is based off what a decompyle attempt showed me.
- switching th em
- agic to 3.12.3 value gave me some more information, I still have an error message though.
- i changed the magic to see if I get a better deocompiled view but there is unspported op codes.
  Am I suppose to rewrite the decompiled code to make it execute? I don't think so because I have no

clue what the unicode would mean for commands.

- so it does have the correct magic code.
- I may need to see if I could try and run the program on an older python intepreter. Not sure if this would work.
- changinging the python version makes me run the code and I get a `usage:` message.
- When I enter the program it then asks me for the flag. I need to try and figure out how to find this.
- I am given the interpreter but I need to figure out how to read their language.
- By reading the code I could see that I need to perform some sort of xor and it will give me the flag. I tried to look at the emojis for hints.
- The sun starts the program, the moon ends the program/function. The numbers are given in emoji format in decimal form. I see a xor symbol.
- I am trying to think of a way to automate the process. Maybe see if there is a way to print this value.
- SIVUSCG{em0t1on4l_d4m4g3}

Ring a Ding

- I needed to look at the js code of the website. I used a command that prints out how to print every function in the console. I learned that you can use console and change the javascript using google chrome developer tools.

NIST

- I need to look into function pointers. The flag will be the first letter of each function successfully initialized. It is wrapped in the flag it self.
- I will get a message letting me know if there was a successful intialization.