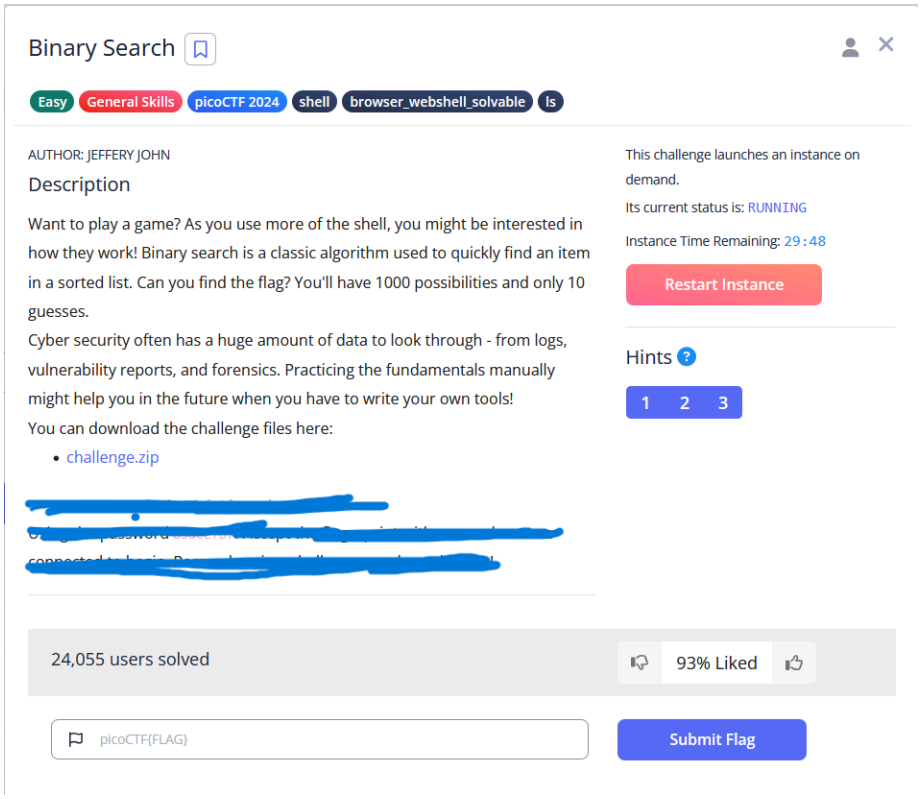
Pico CTF Lucas Jones
I will document my completion of the sections on Pico CTF.



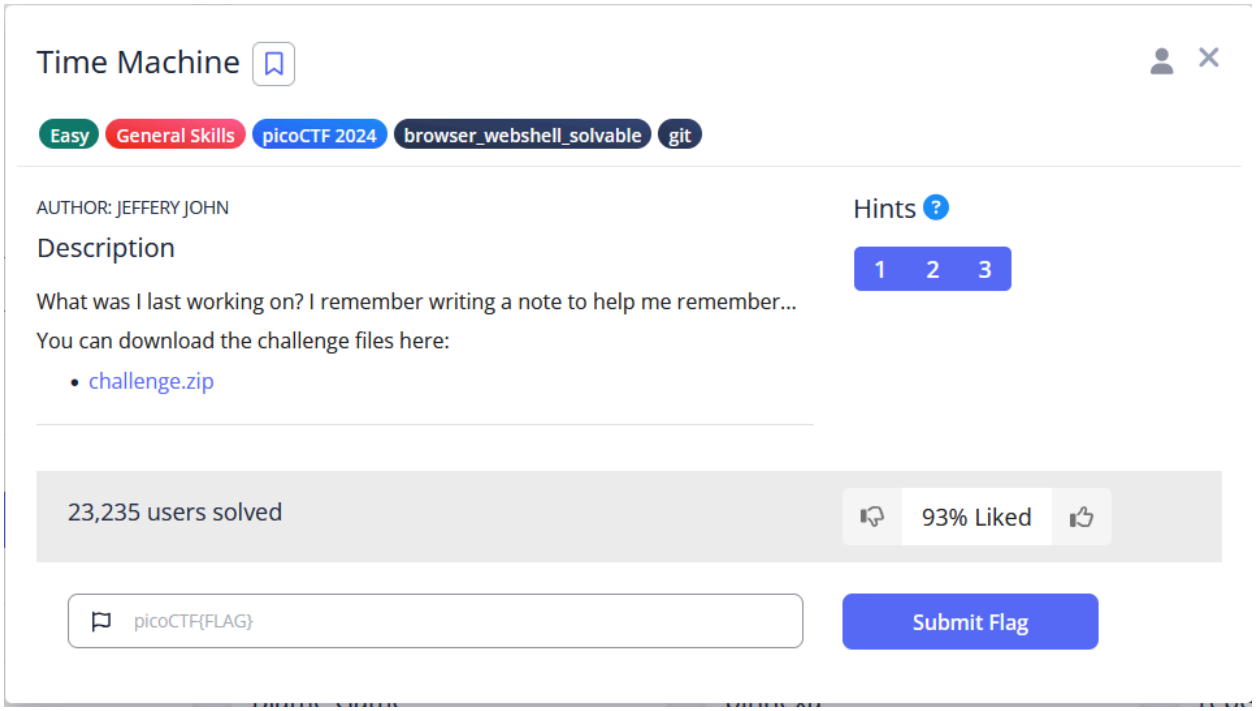
Ok, let's get started working on this one first.

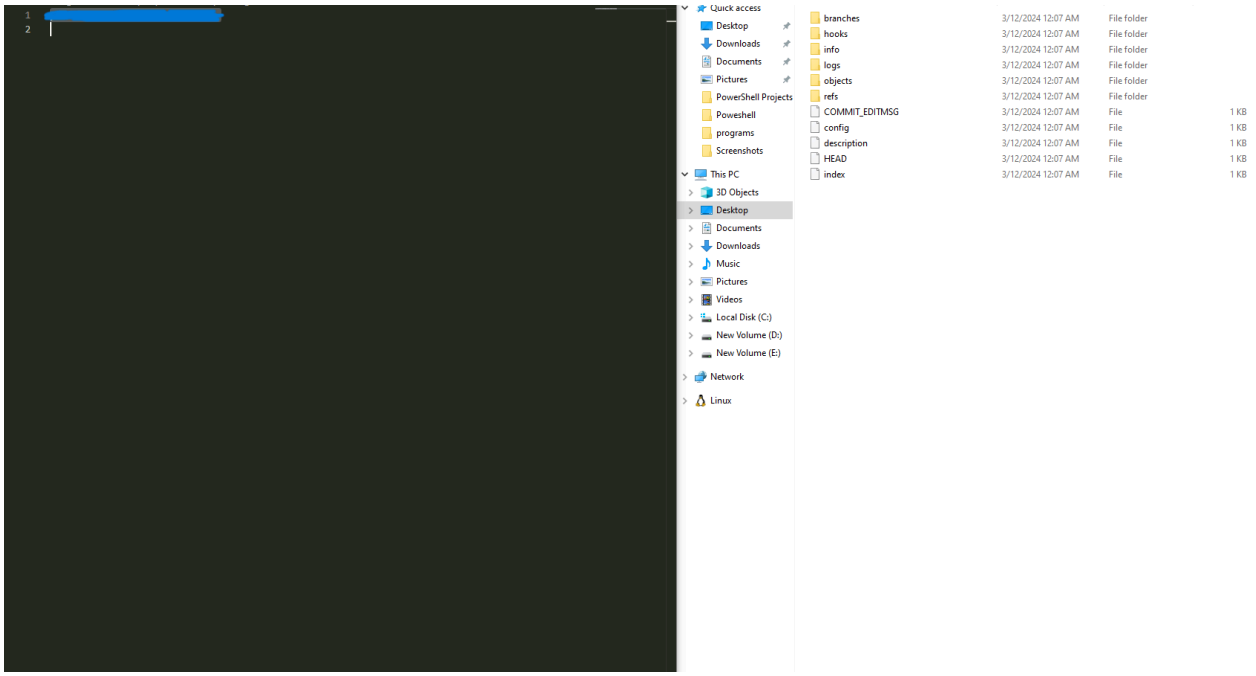
```
selfmadehacker-picoctf@webshell:~$ ssh -p 55851 ctf-player@atlas.picoctf.net
ctf-player@atlas.picoctf.net's password:
Welcome to the Binary Search Game!
I'm thinking of a number between 1 and 1000.
Enter your guess: 815
Higher! Try again.
Enter your guess: 950
Lower! Try again.
Enter your guess: 882
Lower! Try again.
Enter your guess: 818
Higher! Try again.
Enter your guess: 850
Lower! Try again.
Enter your guess: 834
Higher! Try again.
Enter your guess: 842
Higher! Try again.
Enter your guess: 846
Higher! Try again.
Enter your guess: 848
Congratulations! You guessed the correct number: 848
Here's your flag: [redacted]
Connection to atlas.picoctf.net closed.
```

So I found this flag by looking up how binary searching works, trying to find the high and low and get the medium of those two numbers, and then continuing until you hit your flag.

I used x + y / 2 =

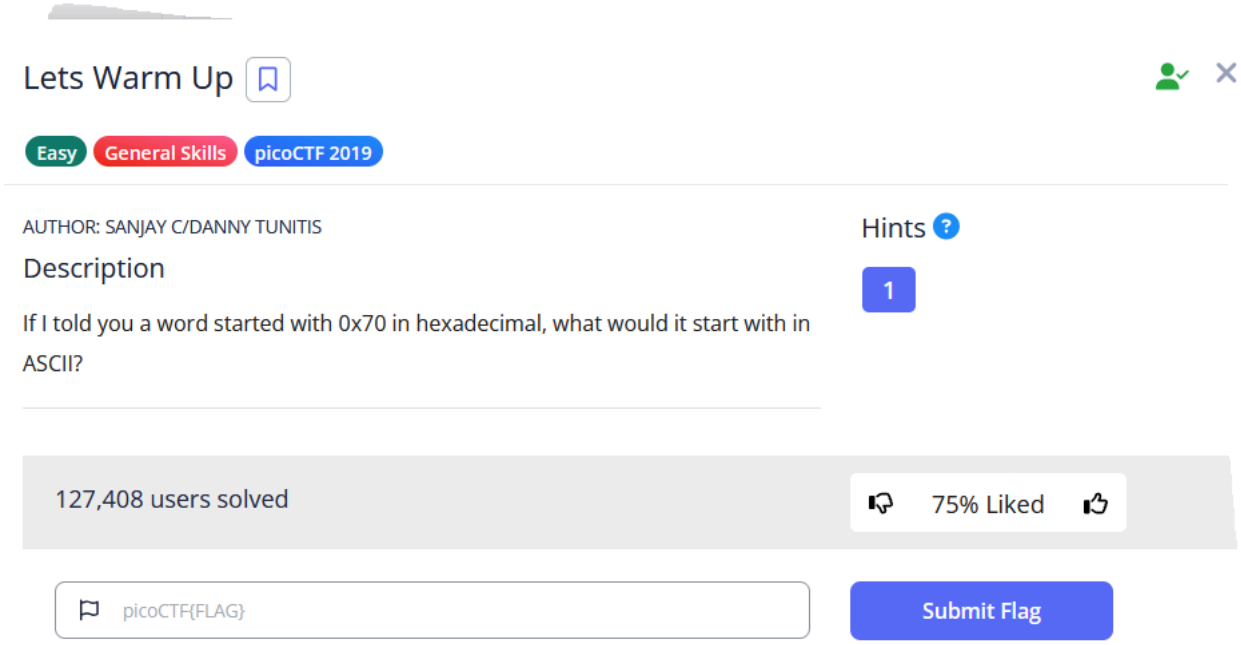
On to the next one!





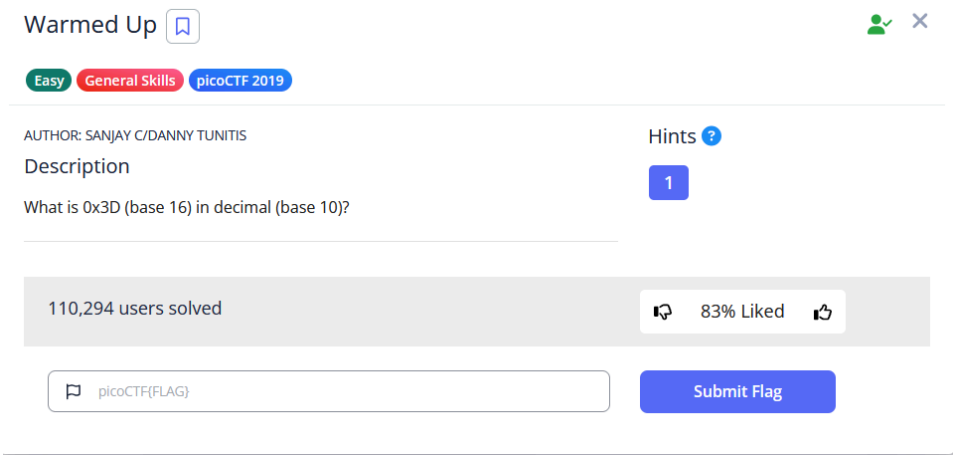
The Time Machine one was finding a file of the comments in the folder and seeing the flag in plain text.

Next one!



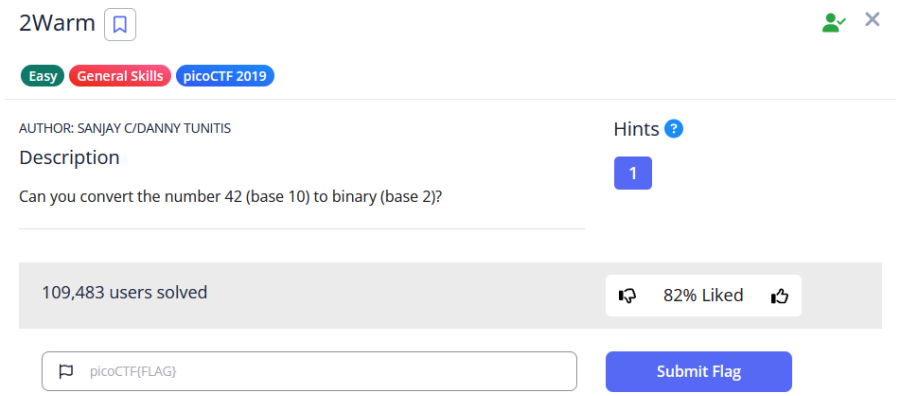
For this one I found an ASCII Table and found the value of 0x70 and put it into the picoCTF format.

Next Challenge!



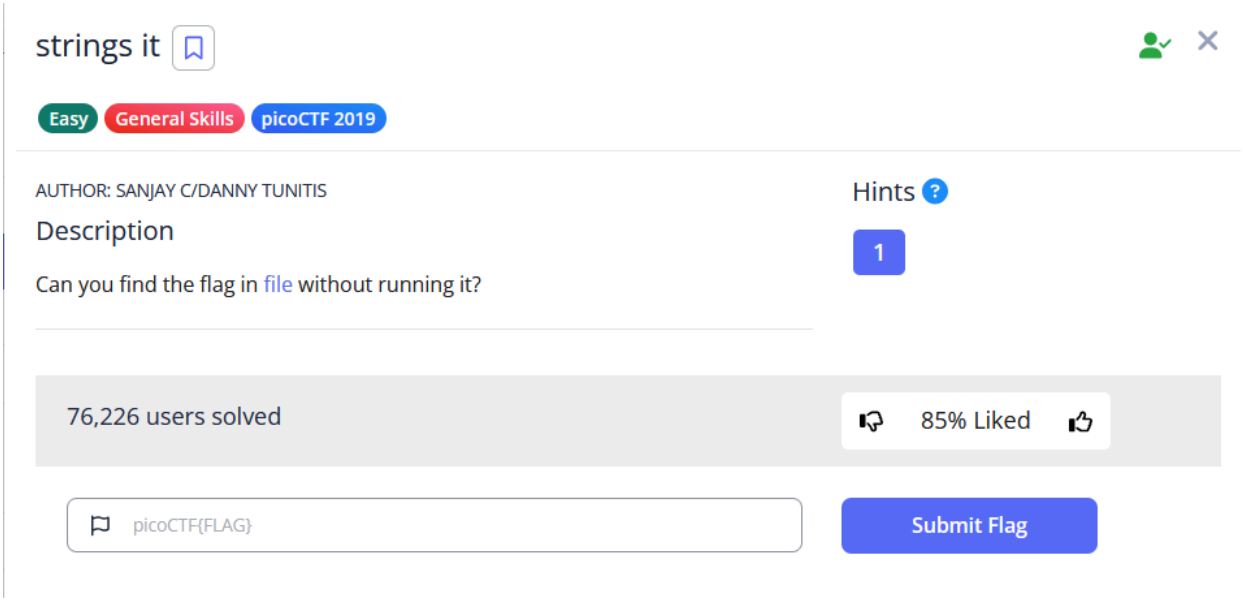
I used the same ASCII Table and found the decimal of 0x3D.

Next challenge!



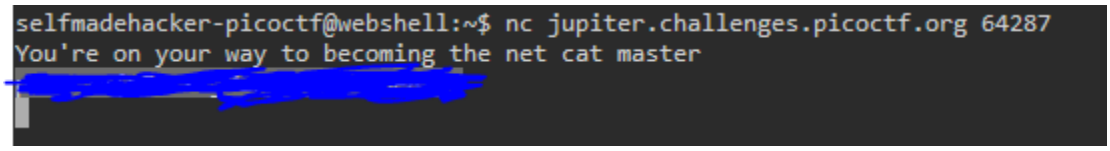
Convert 0x42 into binary, and I found a free service that did that for me.

Next one!



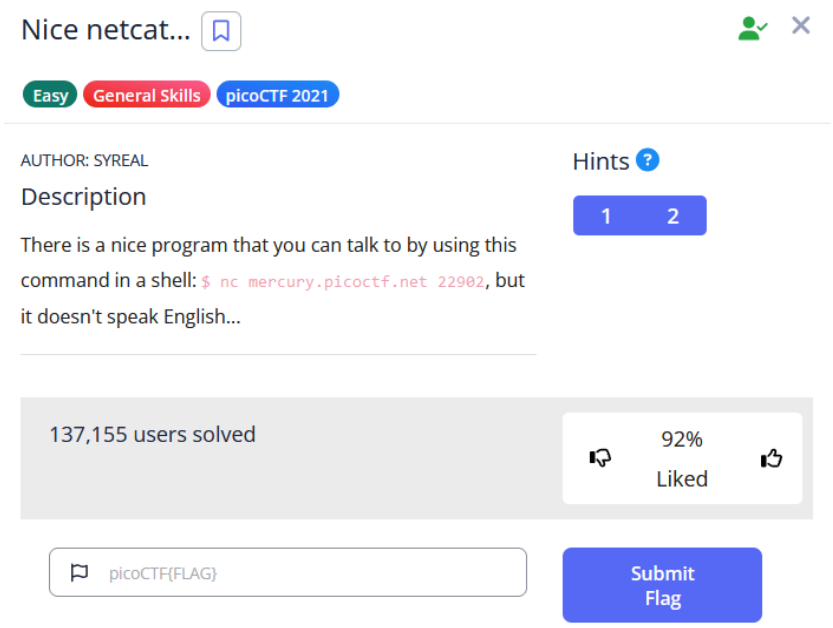
For this one, I opened the file into VS Code looked for the picoCTF flag, and used CTRL-F to find it.

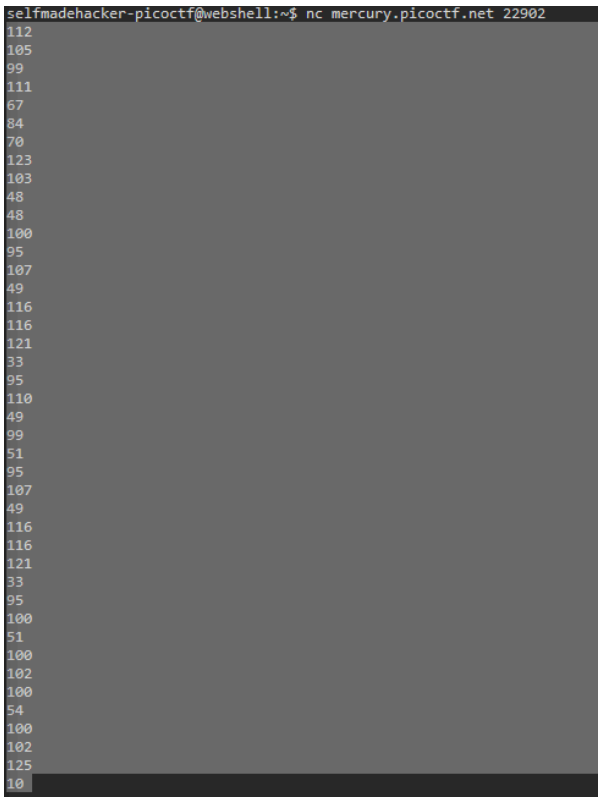
Next One!



Just used netcat syntax and used the login details picoCTF gave me.
Command used - nc jupiter.challenges.picoctf.org 64287

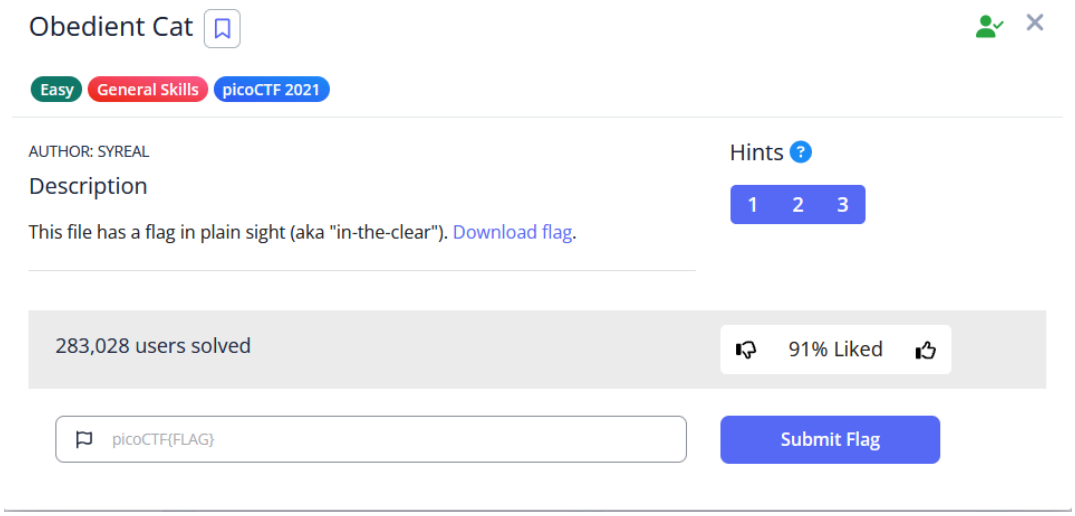
Next One!





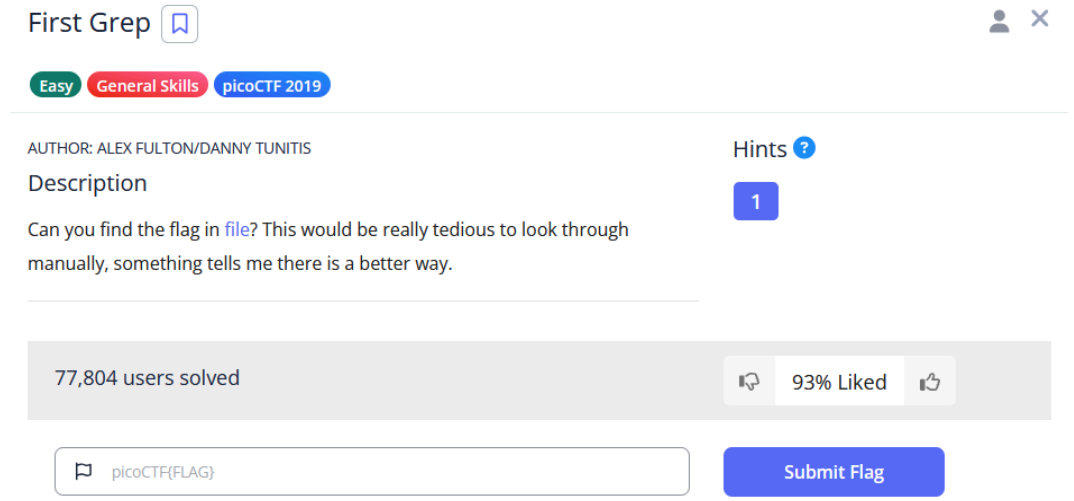
So for this challenge, I used Netcat and the ASCII table to find the flag in decimal.
Command used - nc mercury.picoctf.net 22902

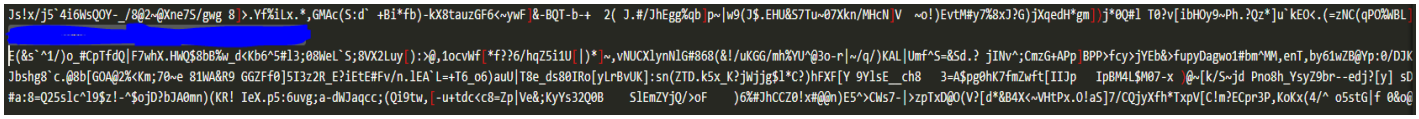
Next one!



This flag was in plain text, which you can find in the document.

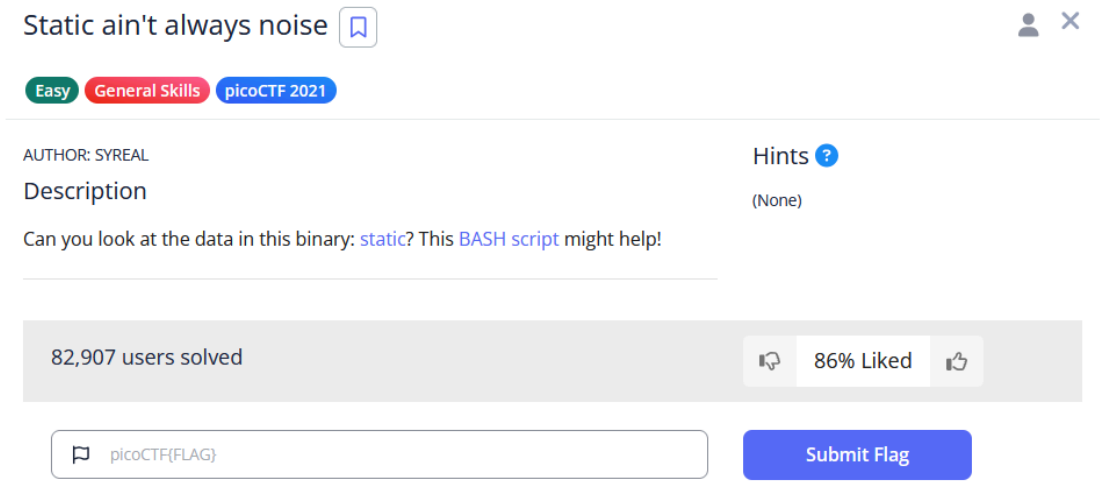
Next One!





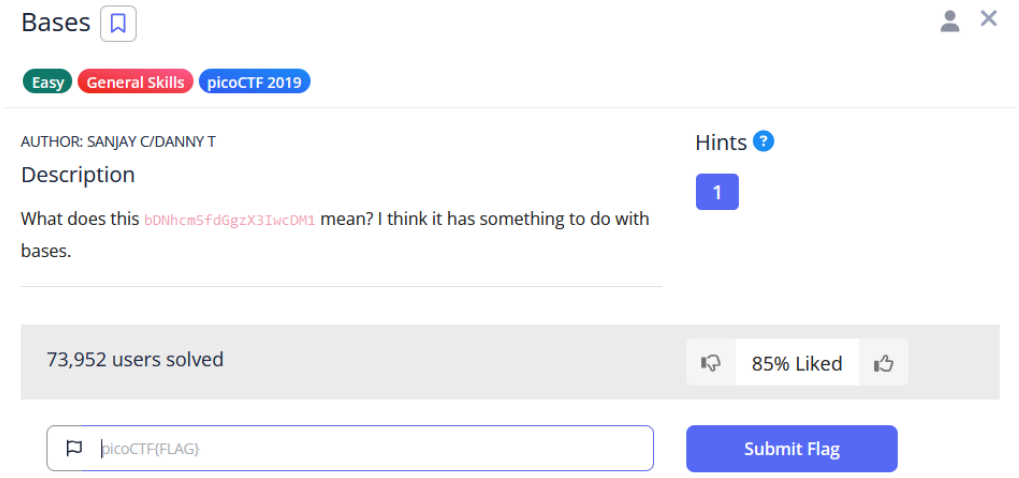
I opened the file and found the flag inside all the information.

Next Challenge!



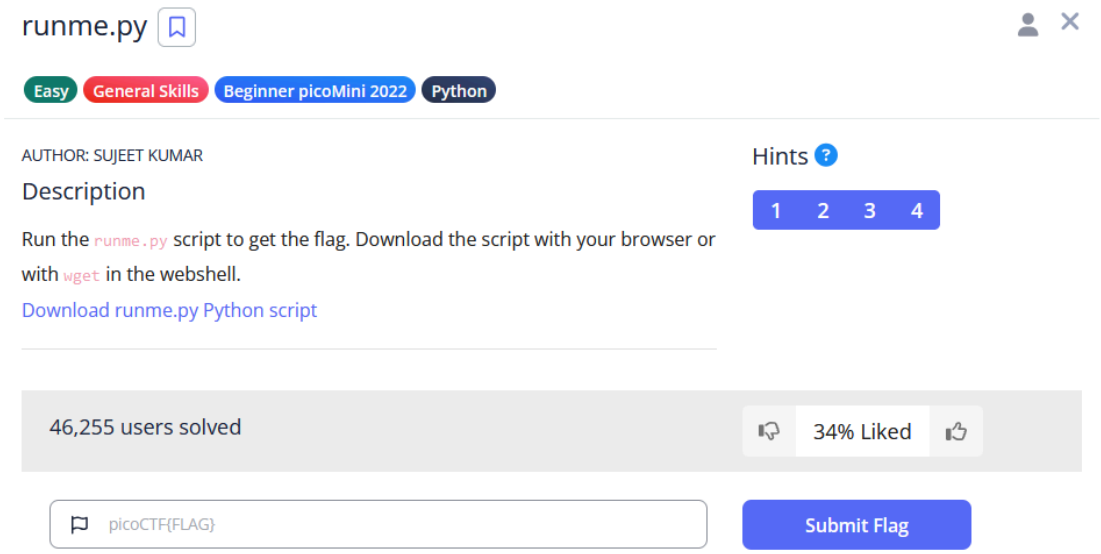

```
Usage: ltdis.sh <program-file>
Bye!
selfmade@DESKTOP-V751NTL:/mnt/c/Users/gameb/Desktop/pico ctf$ ./ltdis.sh <static>
-bash: syntax error near unexpected token `newline'
selfmade@DESKTOP-V751NTL:/mnt/c/Users/gameb/Desktop/pico ctf$ ./ltdis.sh static
Attempting disassembly of static ...
'Disassembly successful! Available at: static.ltdis.x86_64.txt
Ripping strings from binary with file offsets...
Any strings found in static have been written to static.ltdis.strings.txt with file offset
selfmade@DESKTOP-V751NTL:/mnt/c/Users/gameb/Desktop/pico ctf$ '
```

For this challenge, I ran a bash file in Vs Code, and then I ran the bash file with the file named static and looked into the data to find the flag.
Commands used - ./ltdis.sh static
Next one!



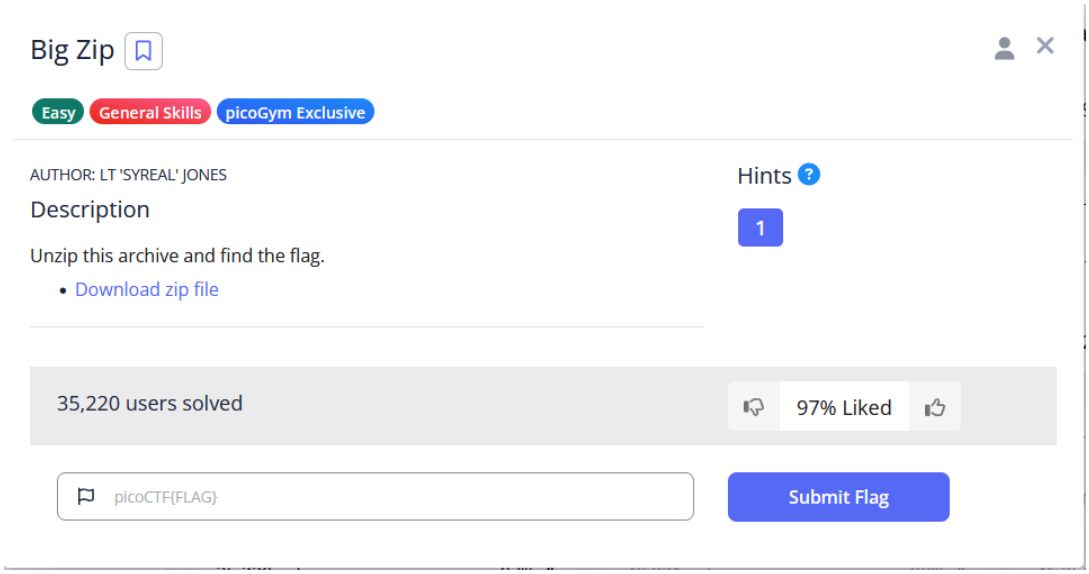
I was able to decode this after using a Base64 decoder. I tried other bases but they did not work for the flag.

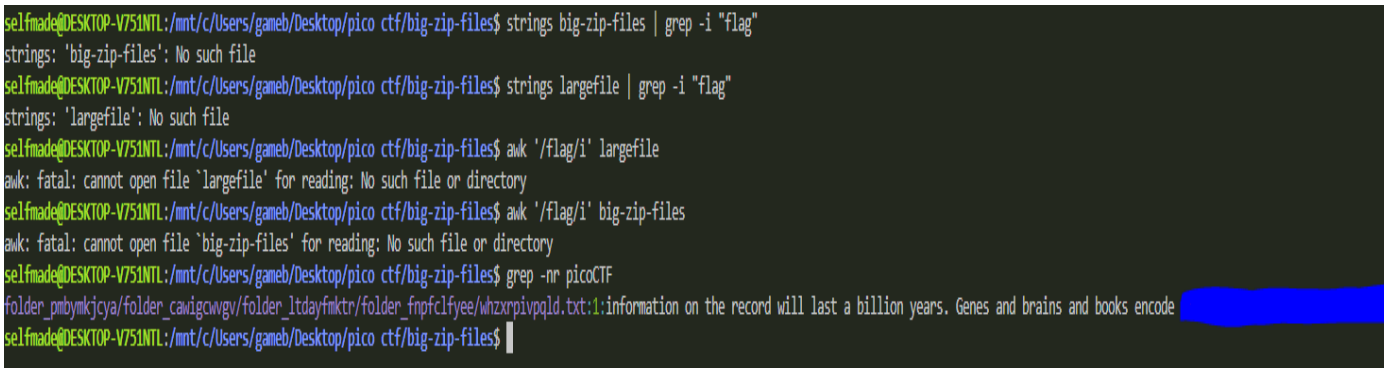
Next Challenge!



I downloaded the Python file and ran it on VS Code to get the flag.
The command used - python3 runme.py

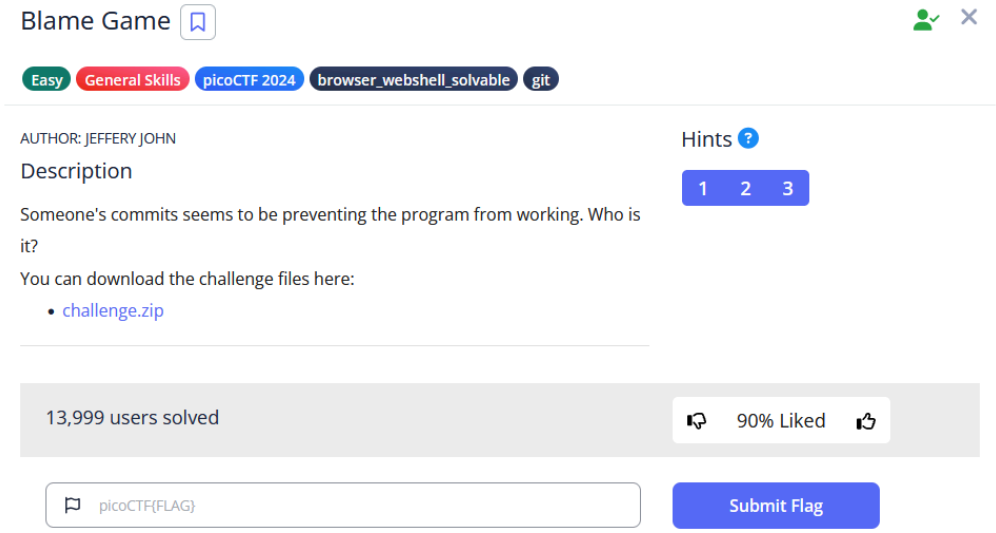
Next Challenge!

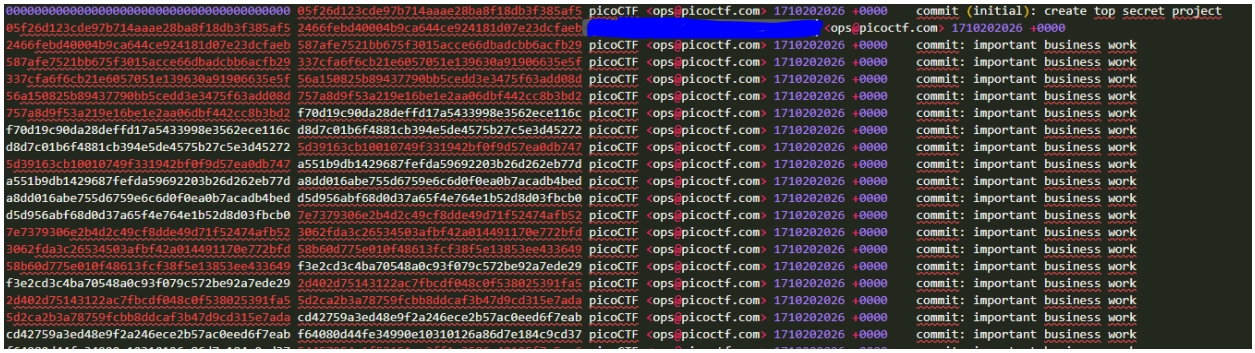




I downloaded the file and opened it with VS Code. I used grep and found the flag.
Commands used -grep -nr picoCTF

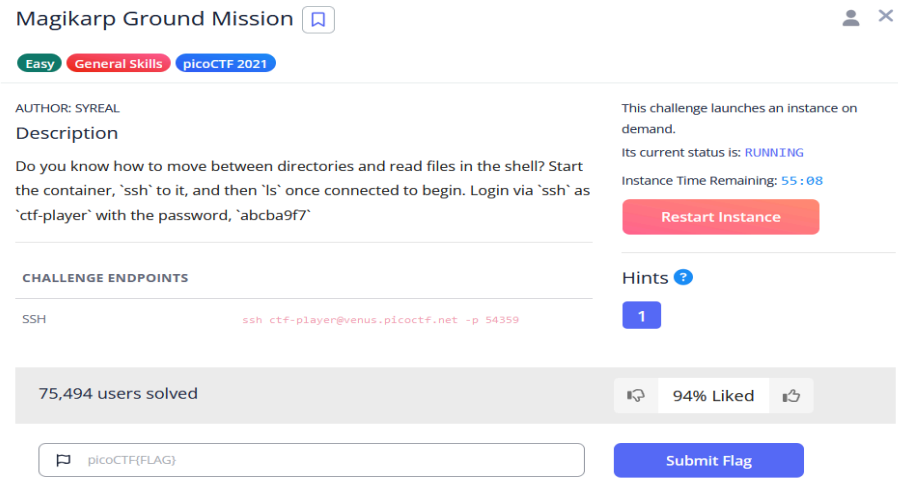
Next Challenge!

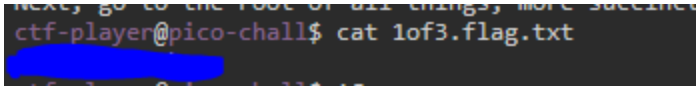


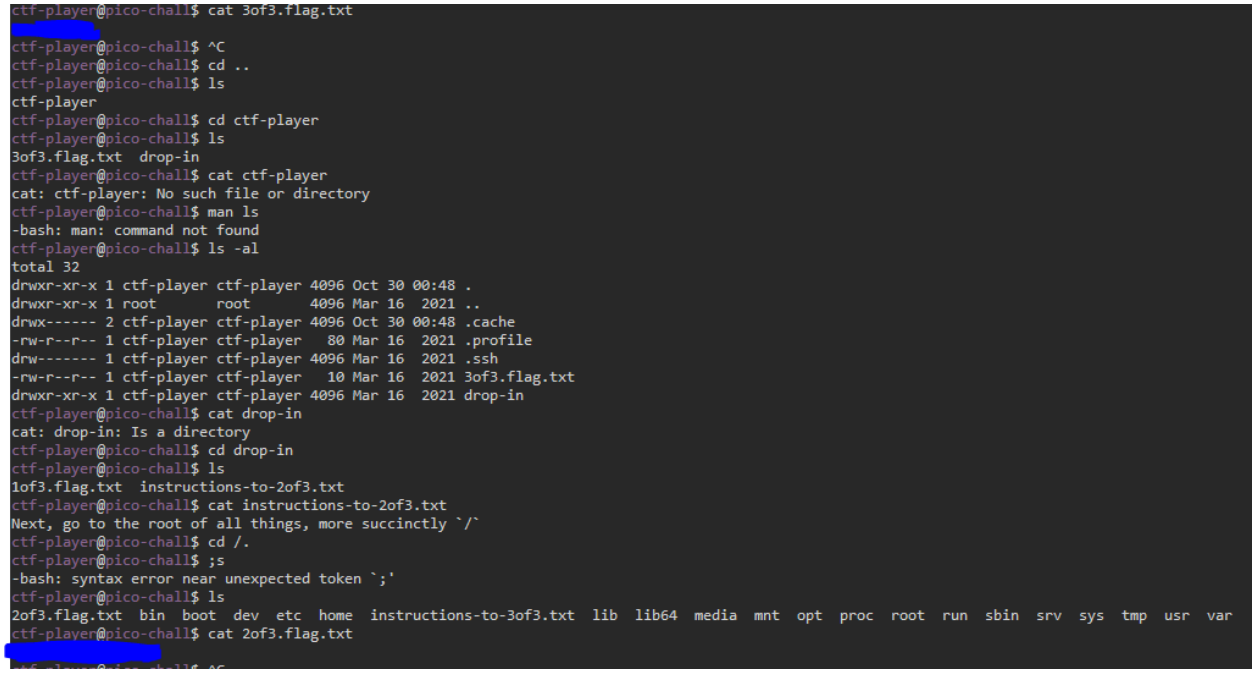


I looked and found the commit where the flag was preventing the program from running.
I did recon and found what file the flag was in.

Next challenge!

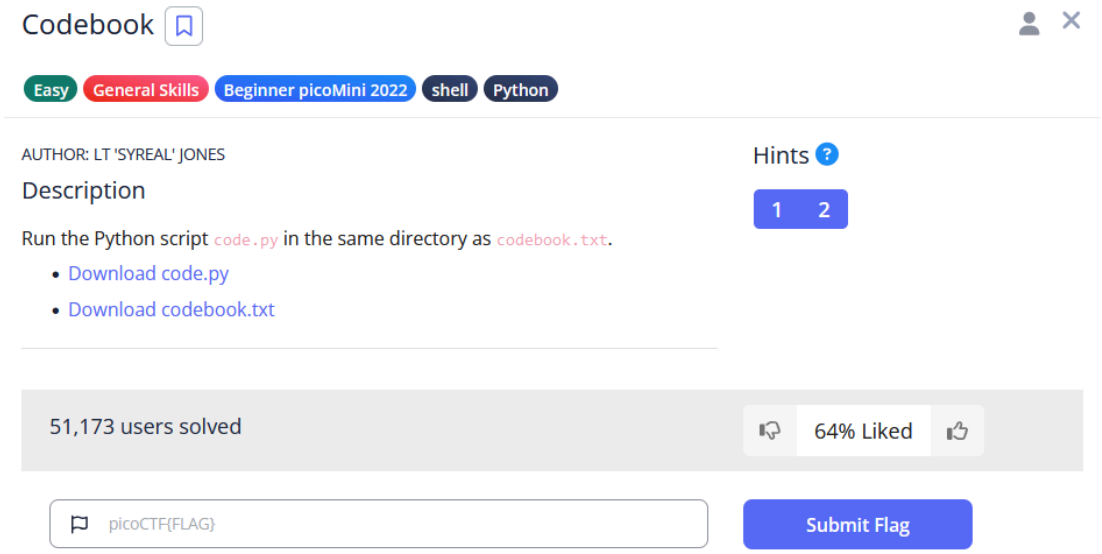


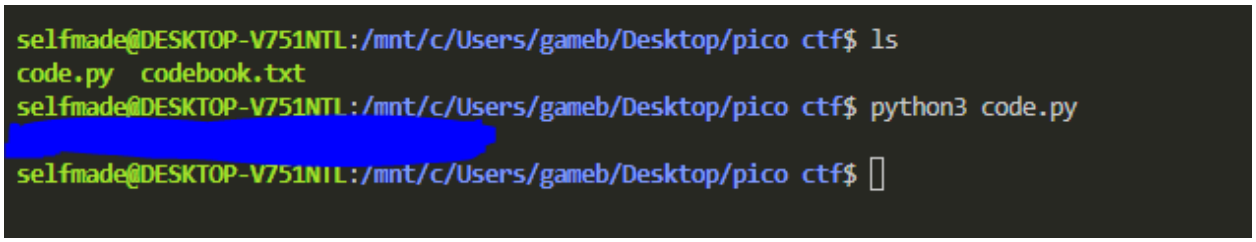




I was able to find all 1-3 parts of the flag.
Commands used were - cd .., cd ~, cd, ls -al, cat, cd /. ,

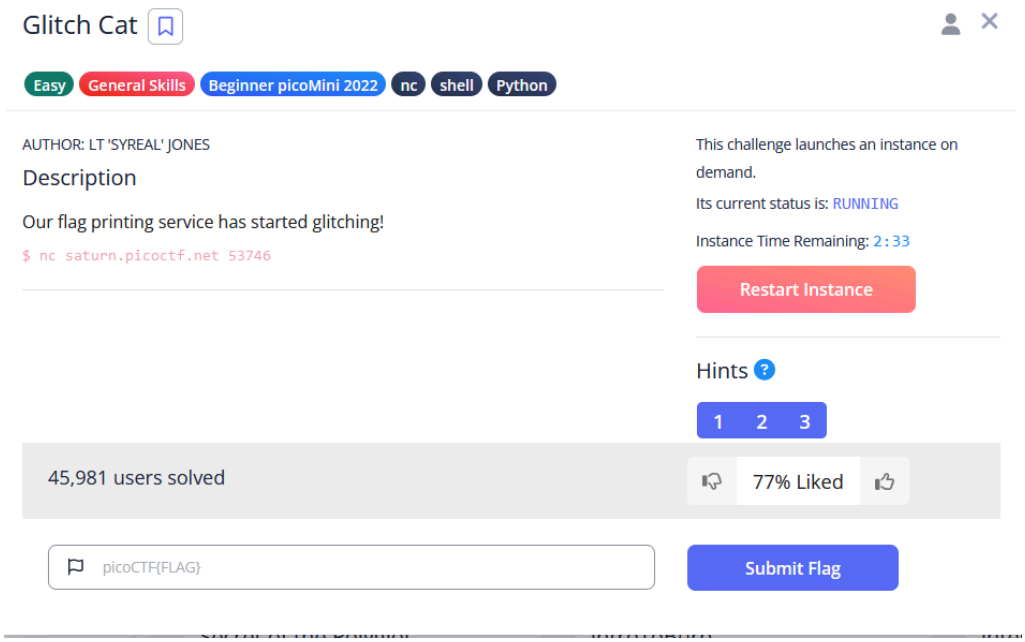
Next one!





I was able to put both of the files inside the same directory and run the Python script.
Command used - Python3 code.py

Next one!

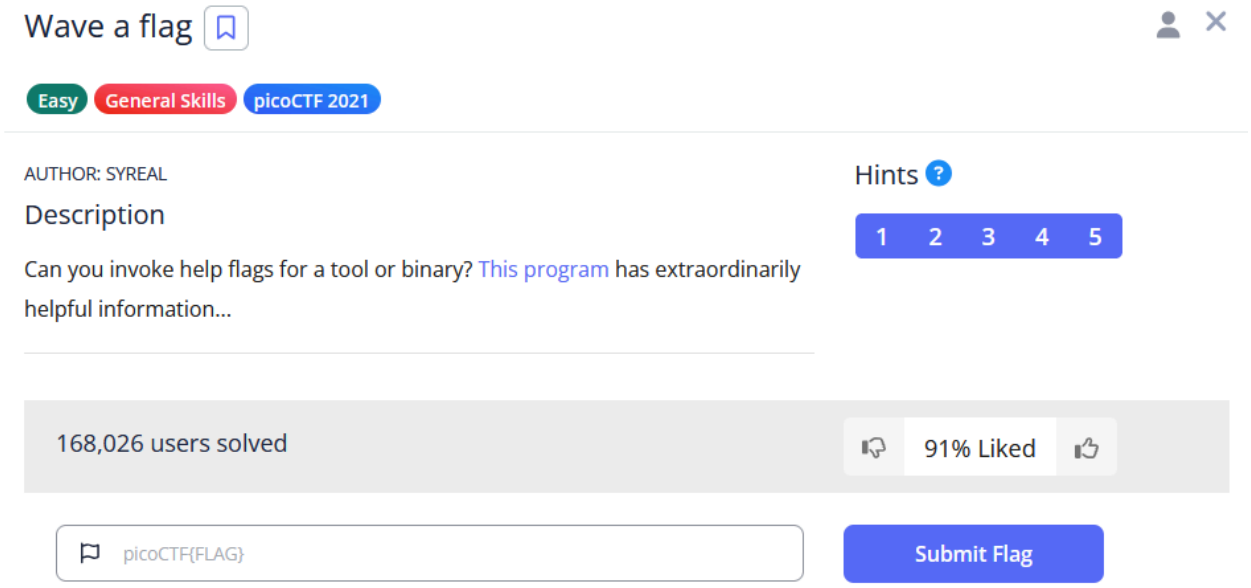


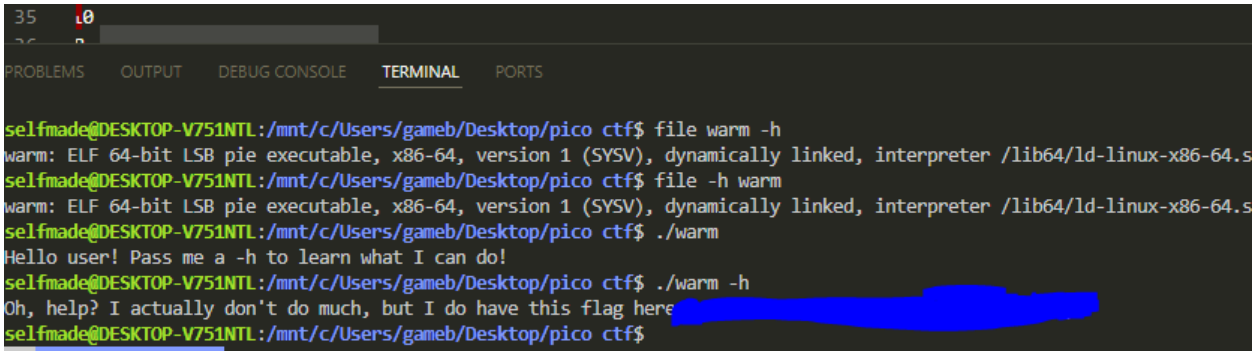

```
Python 3.12.3 (tags/v3.12.3:f6650f9, Apr  9 2024, 14:05:25) [MSC v.1938 64 bit (
AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
print('picoCTF{gl17ch_m3_n07_' + chr(b) + chr(d) + chr(a) + chr($) + chr(&) + ch
r(B) + chr(%) + chr(#) + '}Python 3.12.3 (tags/v3.12.3:f6650f9, Apr  9 2024, 14:
05:25) [MSC v.1938 64 bit (AMD64)] on win32

SyntaxError: invalid syntax
print('picoCTF{gl17ch_m3_n07_' + chr(0x62) + chr(0x64) + chr(0x61) + chr(0x36) +
 chr(0x38) + chr(0x66) + chr(0x37) + chr(0x35) + '}')
```

I discovered that the glitch's output on the nc server was an ASCII number and also used a Python command.
Commands used - print('picoCTF{gl17ch_m3_n07_' + chr(0x62) + chr(0x64) + chr(0x61) + chr(0x36) + chr(0x38) + chr(0x66) + chr(0x37) + chr(0x35) + '}')

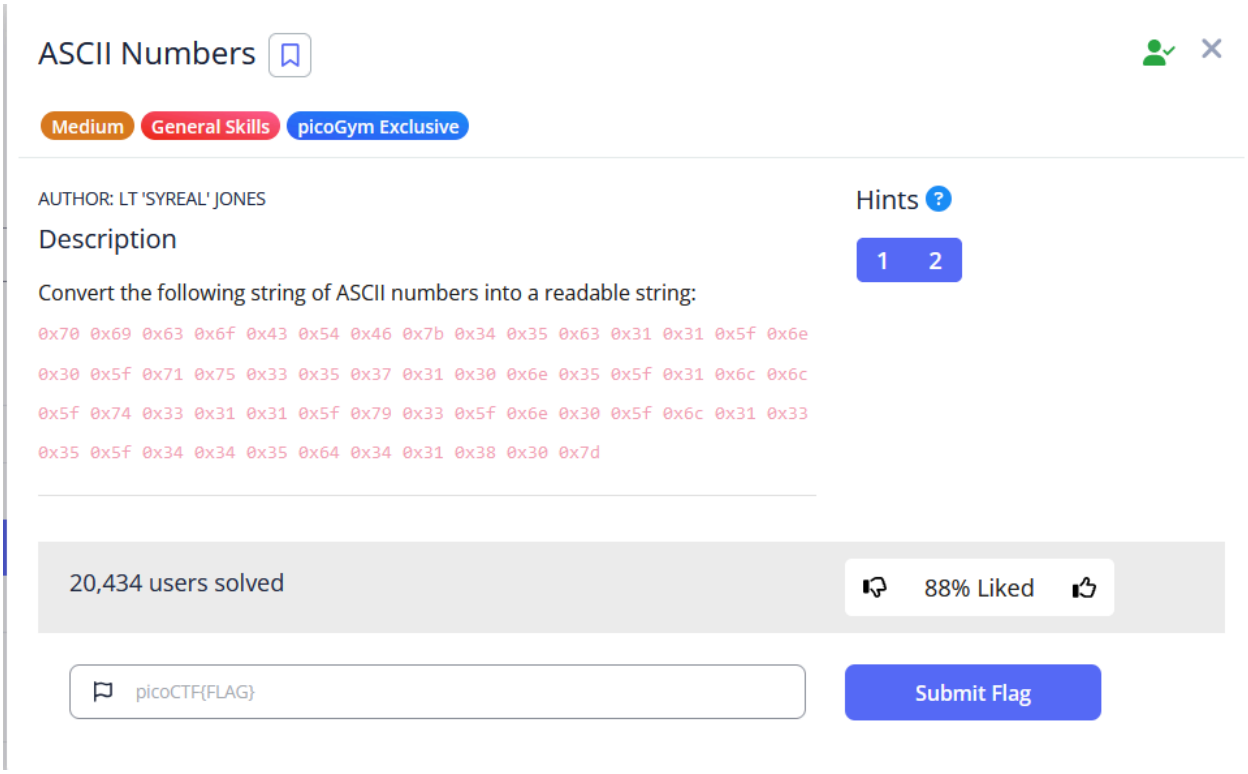
Next one!

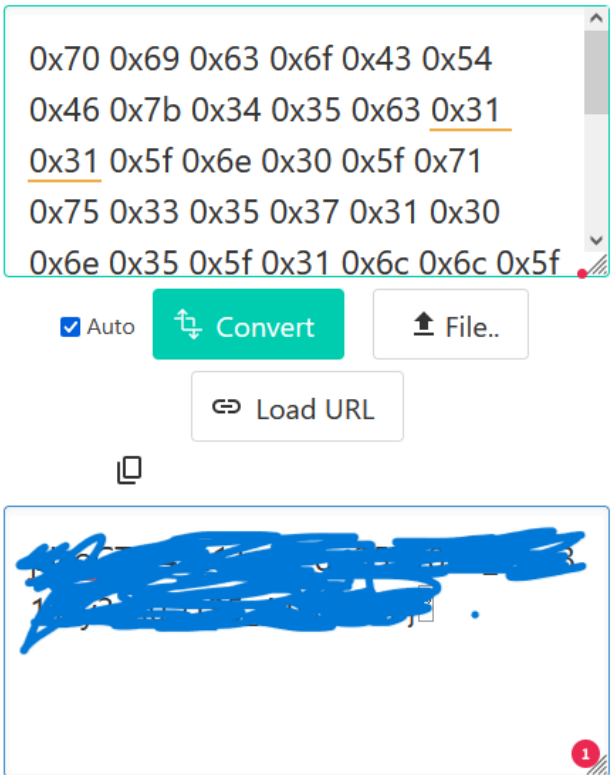




Commands used - ./warm, ./warm -h

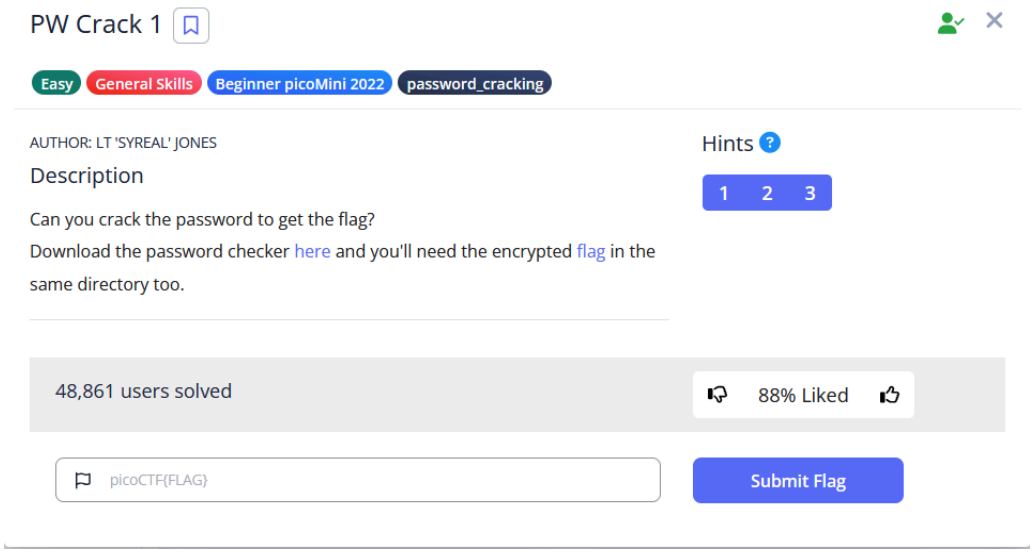
Next one!

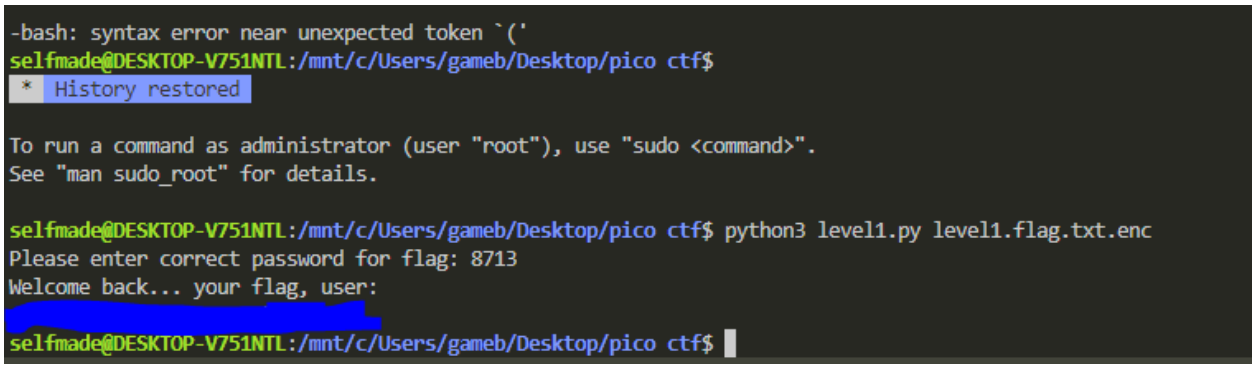




First, I tried to script in Python, but that did not work, so I found an ASCII to translate it and got the flag.

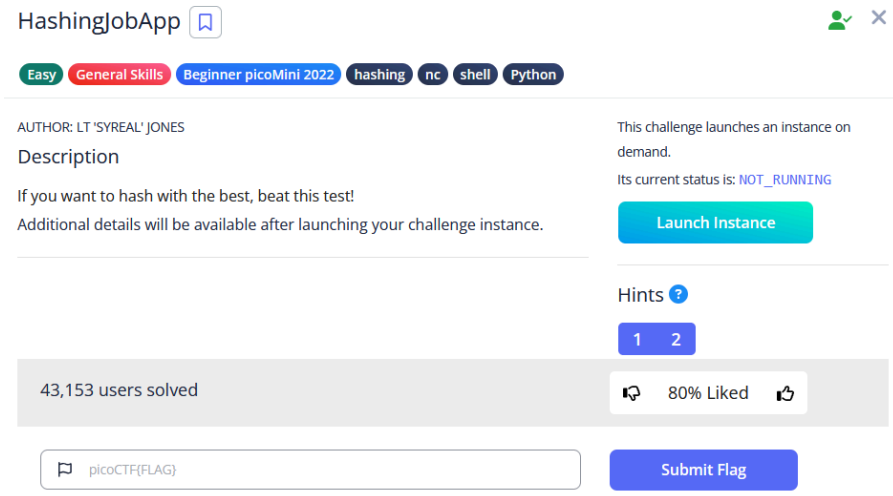
Next challenge!





I found the password inside the Python file and got the flag.
Commands used - Python3 level1.py level1.flag.txt.enc

Next one!



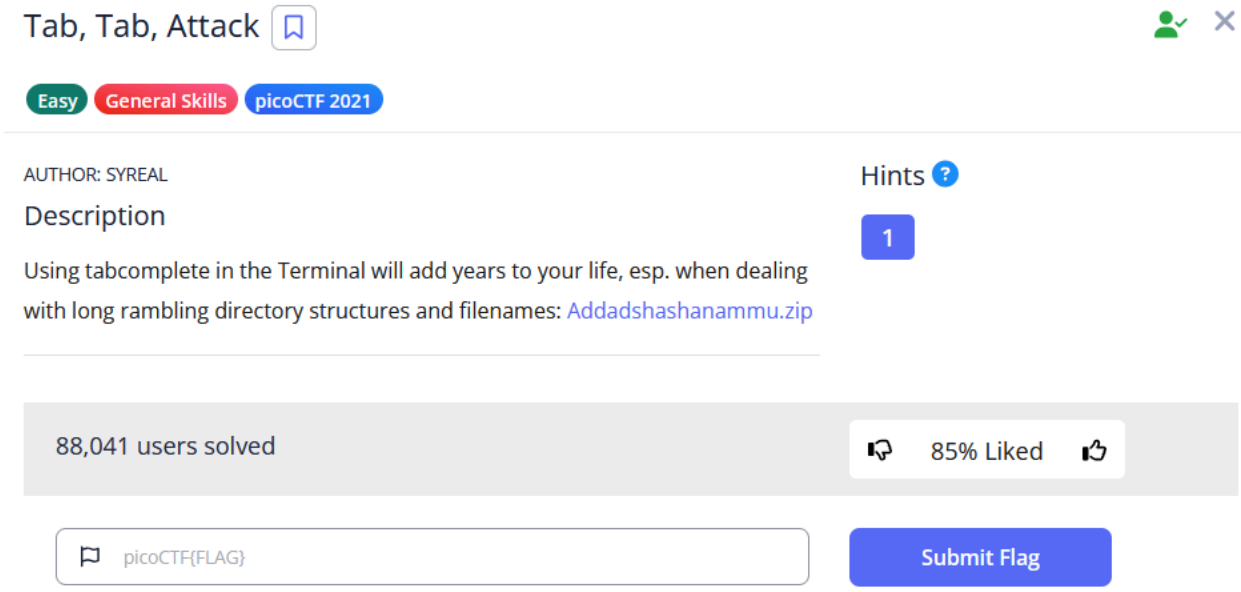
```
Please md5 hash the text between quotes, excluding the quotes: 'armed robbery'
Answer:
986ad0db60b8d3d46281880f693a5926
986ad0db60b8d3d46281880f693a5926
Correct.
Please md5 hash the text between quotes, excluding the quotes: 'computer hackers'
Answer:
1034abc8025edcc22f58c35abc21e36f
1034abc8025edcc22f58c35abc21e36f
Correct.
[redacted]
```

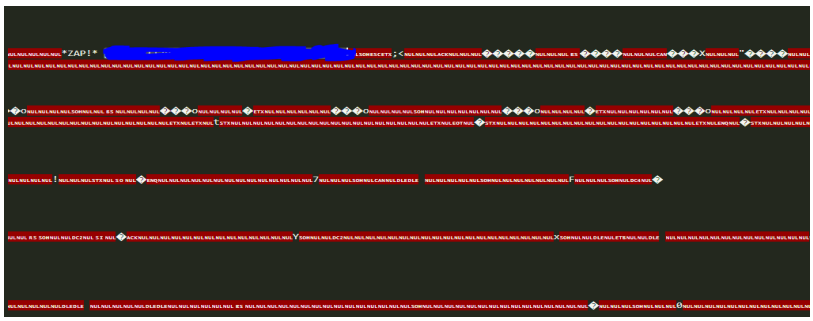
computer hackers

1034abc8025edcc22f58c35abc21e36f

I found an MD5 tool that will encode the strings and I found the flag.

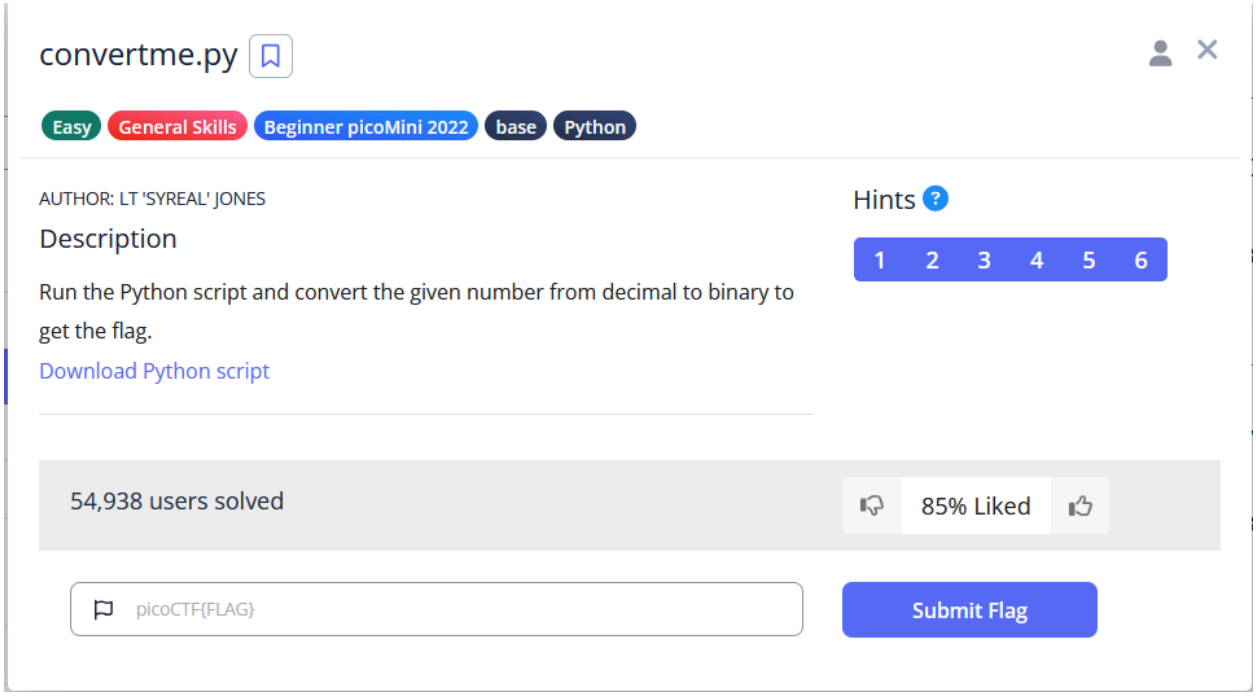
Next one!

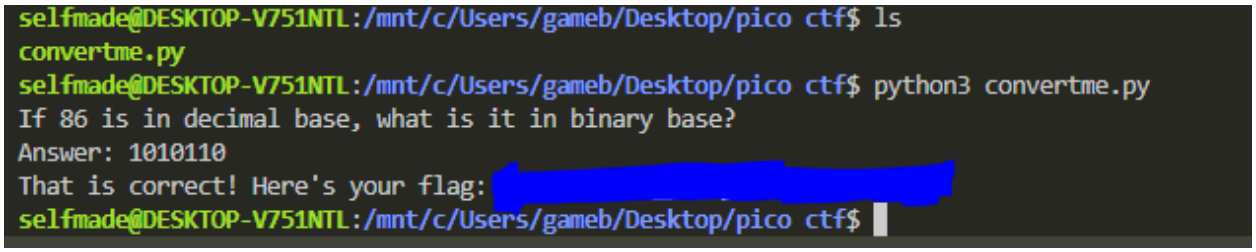




Put the folder into Vs Code and found the flag while looking through it.

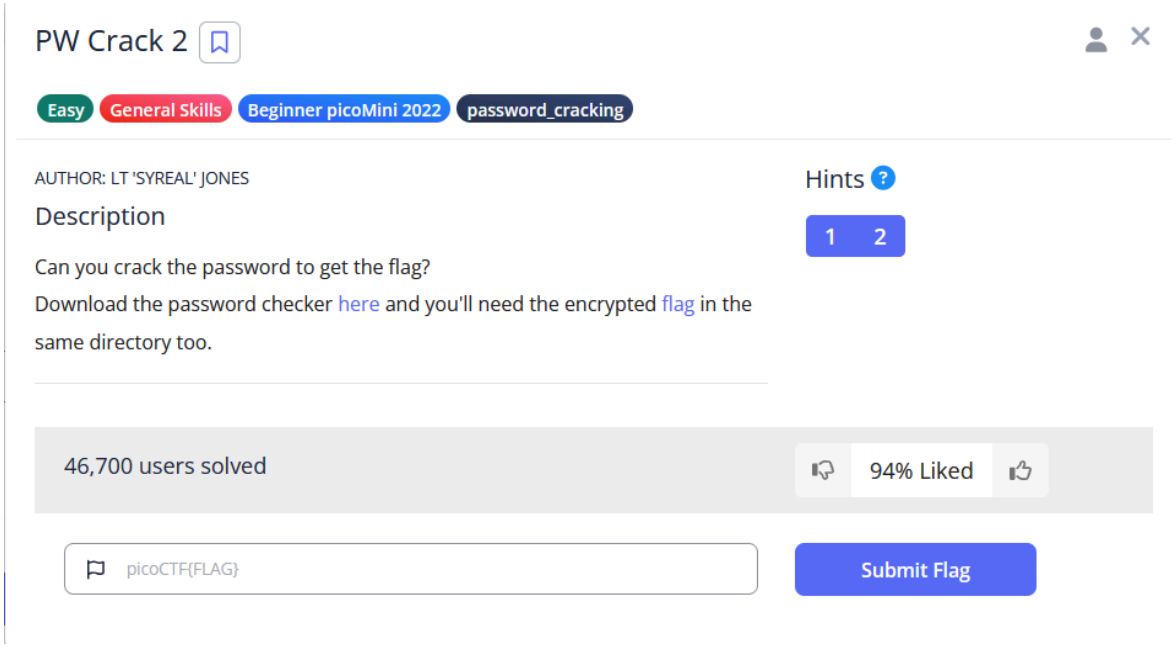
Next one!





Found the decimal base of 86 with the ASCII Table, converted it online, and input it into the file.
Commands used - python3 convertme.py

Next one!

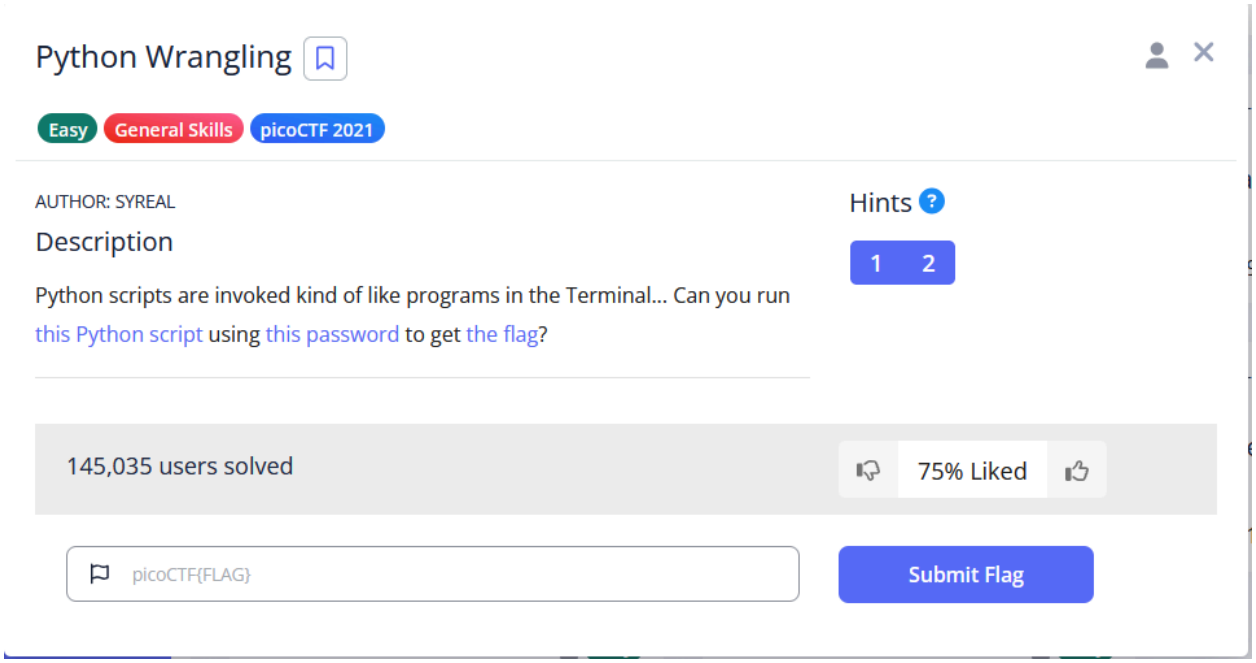


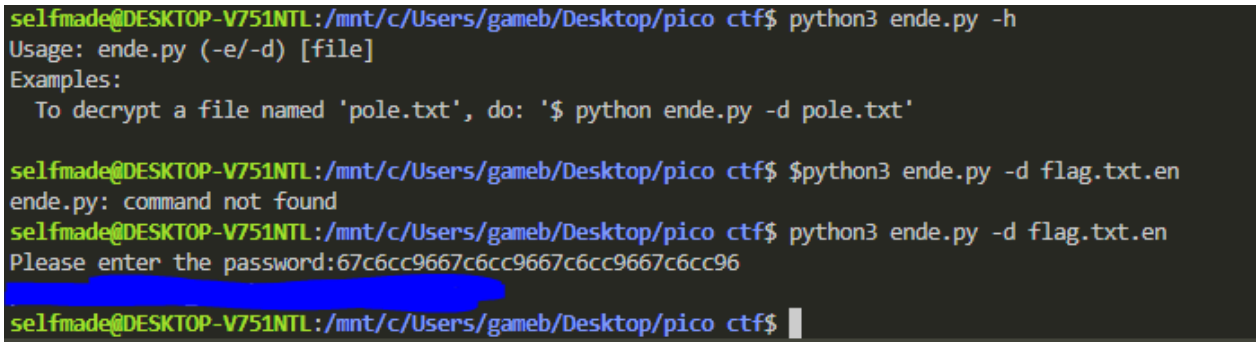

```
>>> print("chr(0x33) + chr(0x39) + chr(0x63) + chr(0x65)")
chr(0x33) + chr(0x39) + chr(0x63) + chr(0x65)
>>> print(chr(0x33) + chr(0x39) + chr(0x63) + chr(0x65))
39ce
>>> exit
Use exit() or Ctrl-D (i.e. EOF) to exit
>>> exit()
selfmade@DESKTOP-V751NTL:/mnt/c/Users/gameb/Desktop/pico ctf$ python3 level2.py level2.flag.txt.enc -d
Please enter correct password for flag: 39ce
Welcome back... your flag, user:
[redacted]
selfmade@DESKTOP-V751NTL:/mnt/c/Users/gameb/Desktop/pico ctf$
```

I noticed that the password was in ASCII to get the correct flag so at first I translated it to text and that did not work. So I opened up Python and printed print(chr(0x33) + chr(0x39) + chr(0x63) + chr(0x65)) and got the flag.

Commands used - python3 level2.py level2.flag.txt.enc -d , print(chr(0x33) + chr(0x39) + chr(0x63) + chr(0x65)).

Next One!





I used the -h header to help me see how to decrypt the file ende.py and I used that to input the password and get the flag.

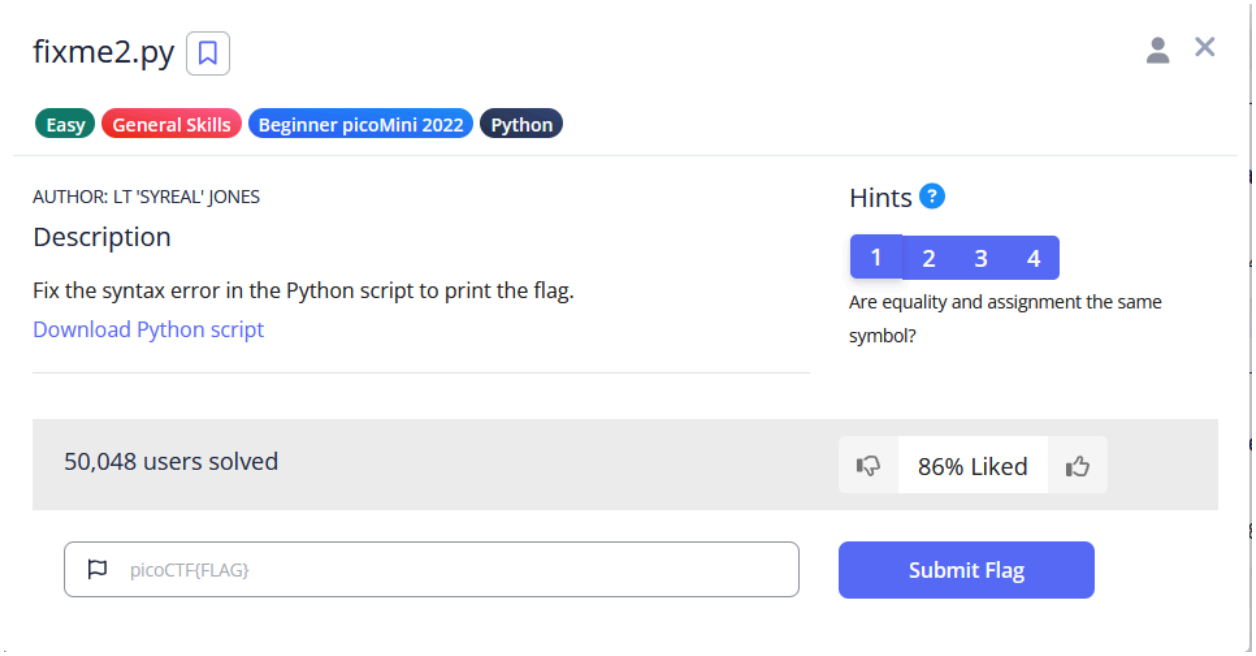
Commands used - python3 ende.py -h, python3 ende.py -d flag.txt.en

Next One!


```
selfmade@DESKTOP-V751NTL:/mnt/c/Users/gameb/Desktop/pico ctf$ python3 fixme1.py
  File "/mnt/c/Users/gameb/Desktop/pico ctf/fixme1.py", line 20
    print('That is correct! Here\'s your flag: ' + flag)
IndentationError: unexpected indent
selfmade@DESKTOP-V751NTL:/mnt/c/Users/gameb/Desktop/pico ctf$ python3 fixme1.py
  File "/mnt/c/Users/gameb/Desktop/pico ctf/fixme1.py", line 20
    print('That is correct! Here\'s your flag: ' + flag)
IndentationError: unexpected indent
selfmade@DESKTOP-V751NTL:/mnt/c/Users/gameb/Desktop/pico ctf$ nano fixme1.py
selfmade@DESKTOP-V751NTL:/mnt/c/Users/gameb/Desktop/pico ctf$ python3 fixme1.py
That is correct! Here's your flag: [redacted]
selfmade@DESKTOP-V751NTL:/mnt/c/Users/gameb/Desktop/pico ctf$
```

At first, I looked at the VS code to see what was causing the errors and could see that it was the indentations. I opened up Nano and saw the indentations causing the errors and deleted them.
Commands used - nano, python3

Next One!




```
selfmade@DESKTOP-V751NTL:/mnt/c/Users/gameb/Desktop/pico ctf$ ls
fixme2.py
selfmade@DESKTOP-V751NTL:/mnt/c/Users/gameb/Desktop/pico ctf$ python3 fixme2.py
  File "/mnt/c/Users/gameb/Desktop/pico ctf/fixme2.py", line 22
    if flag = "":
       ^^^^^^^^^
SyntaxError: invalid syntax. Maybe you meant '==' or ':=' instead of '='?
selfmade@DESKTOP-V751NTL:/mnt/c/Users/gameb/Desktop/pico ctf$ nano fixme2.py
selfmade@DESKTOP-V751NTL:/mnt/c/Users/gameb/Desktop/pico ctf$ nano fixme2.py
selfmade@DESKTOP-V751NTL:/mnt/c/Users/gameb/Desktop/pico ctf$ python3 fixme2.py
That is correct! Here's your flag: [redacted]
selfmade@DESKTOP-V751NTL:/mnt/c/Users/gameb/Desktop/pico ctf$
```

I first took out all the unneeded indentations and opened it in Nano. I saw that the if statement was wrong so I changed it to if == and got the flag.
Commands used - Nano , python3 , ==