

Writeup

CSC CTF Contest 2025



Nama : Marcellino Marvell
Nickname : Krt
NIM : 2702209281
Jurusan : Cyber Security

Table of Contents

Misc	3
The Chest Boards Treasure	3
Forensic	5
Kawat Hiu	5
Skibidi Invasion	6
Cryptography	10
Baby Substituion	10
Aurichia-Docs	11
MY encryption	16
The Secret of Sepuh	19
Reverse Engineering	22
Broken Mario	22
Sigma Checker	26
Welcome	31
Sanity Check	31

Misc

The Chest Boards Treasure

Author : Auric, Redmito

Description

My friend redmito left a note what could it mean?

flag format = CSC{NAME_OF_WINNER_YEAR} example flag :
CSC{renko_varel_2025} all in lowercase letters

Explanation

Pada soal ini diberikan sebuah file bernama notes.txt, dan ketika dibuka itu berisi sebuah teka-teki dan dalam teka-teki tersebut disebutkan kata catur. Jadi berdasarkan description dan teka-teki tersebut saya beranggapan bahwa kita disuruh untuk mencari pemenang dari suatu pertandingan catur.

notes.txt

In the heart of a bustling city, there was a quiet individual known only by the alias, Redmito. Though unassuming, Redmito possessed a peculiar talent—an encyclopedic knowledge of chess that bordered on the mystical. Wherever Redmito went, the topic of chess followed, as though the ancient game had intertwined with their very soul. But there was more to Redmito than just chess.

Redmito had an undeniable love for money and treasure. Whether it was tales of hidden vaults or legendary hoards of gold, Redmito's eyes would light up whenever the conversation steered towards riches. It was as though their passion for chess was rivaled only by an obsession with treasure chests. Some say the gleam in their eye while discussing grandmaster matches was the same as when they spoke of gold.

Whispers began to circulate that Redmito believed chess itself held the key to unlocking a grand treasure, hidden away in some forgotten corner of the world. What others saw as just a game, Redmito saw as a map leading to untold fortune—some say by the end, the game was smothered in gold.

The moves were so legendary, they seemed to defy reality—each calculated step on the chessboard conjured gold out of thin air. It was said that with every perfect combination, shimmering gold would appear, as though the game itself was a portal to hidden riches. Redmito reveled in these tales, convinced that mastering such moves would lead not only to victory on the board but to unimaginable wealth beyond anyone's wildest dreams...

Pada teka-teki tersebut disebutkan kata gold berulang kali, jadi saya langsung saja mencari tau tentang ini dan dengan bantuan GPT dan searching saya menemukan <https://www.chess.com/blog/raync910/the-gold-coins-game> Dan disitu disebutkan bahwa pemenang dari game tersebut adalah frank marshall dan tahunnya adalah 1912. Jadi langsung saja saya coba masukan ke dalam format flagnya CSC{frank_marshall_1912}, dan saya submit. Dan benar saja itu flag nya

Although the **"gold coins game"** occurred this month **105 years ago** (on July 20, 1912), I just recently learned about it. The finish of the game shows the **dramatic power of a knight** to checkmate an enemy king when a trap is set with other pieces.

Frank Marshall vs. Stefen Levitsky

The game was played in an area that today is Poland in 1912 between **Frank Marshall**, who won with black, and **Stefen Levitsky**. At the time, Marshall was the U.S. Chess Champion. Levitsky was a Russian chess master and national champion. The diagram below shows the board position after the 23rd move by white. The **"shower of gold" move** by Black is next.

Flag

CSC{frank_marshall_1912}

Forensic

Kawat Hiu

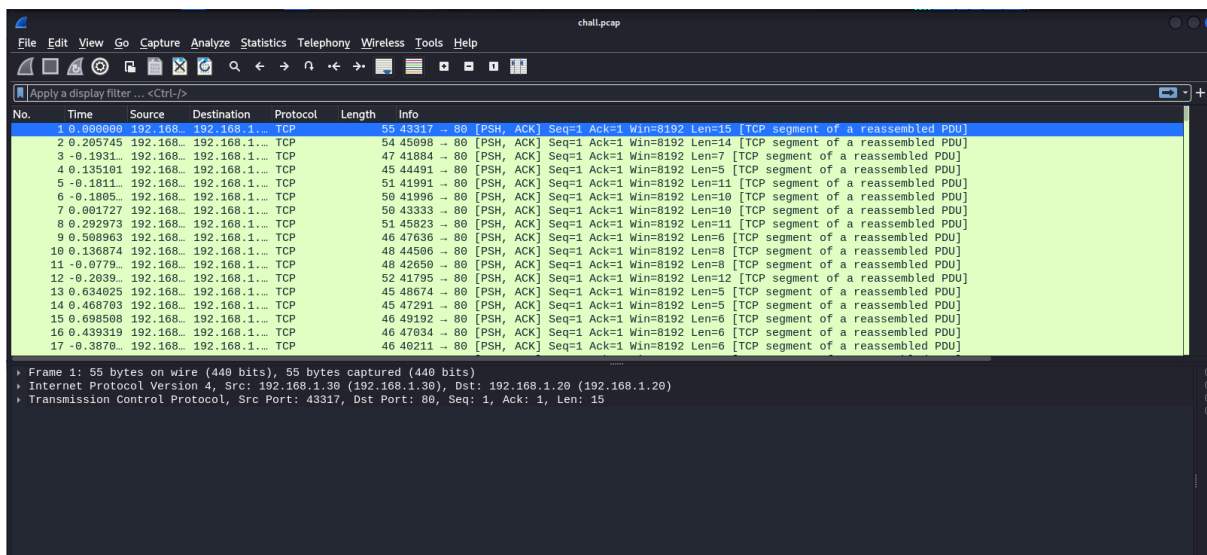
Author : rh17

Description

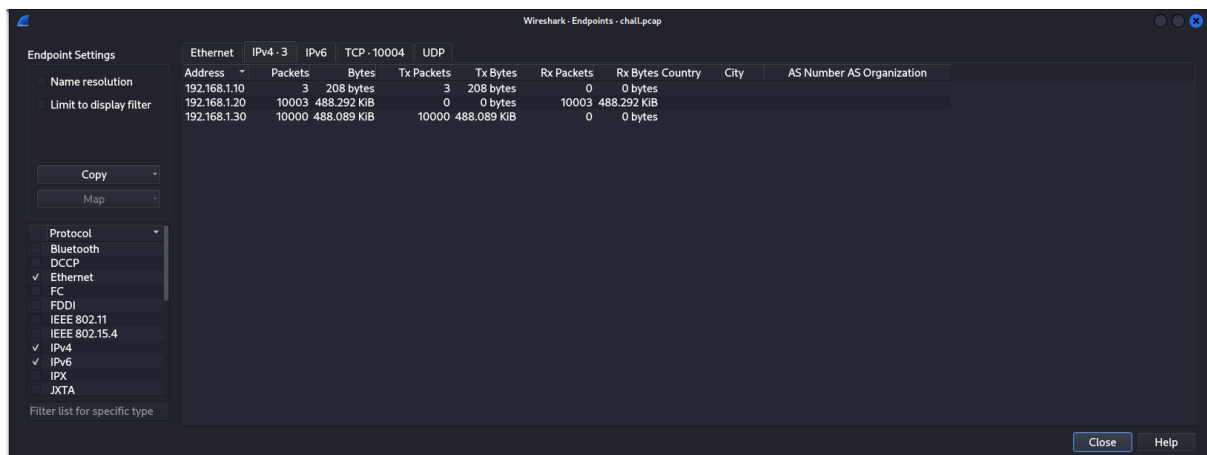
Ada yang aneh di jaringan... katanya sih ada pesan rahasia yang dikirim diem-diem lewat packet TCP. Coba cari ke dalam file .pcap ini dan temukan flag utuh!

Explanation

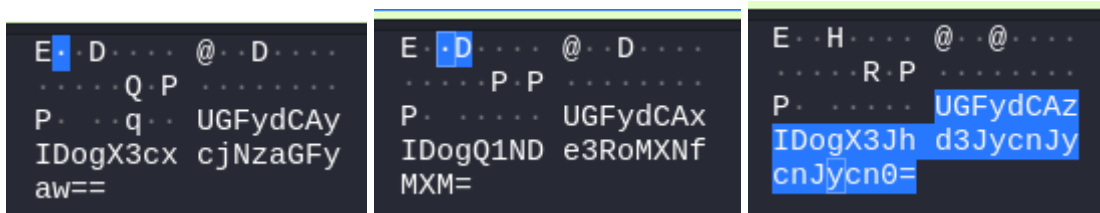
Pada soal ini diberikan sebuah file bernama chall.pcap. Jadi langsung saja saya mencoba untuk membuka filenya di wireshark



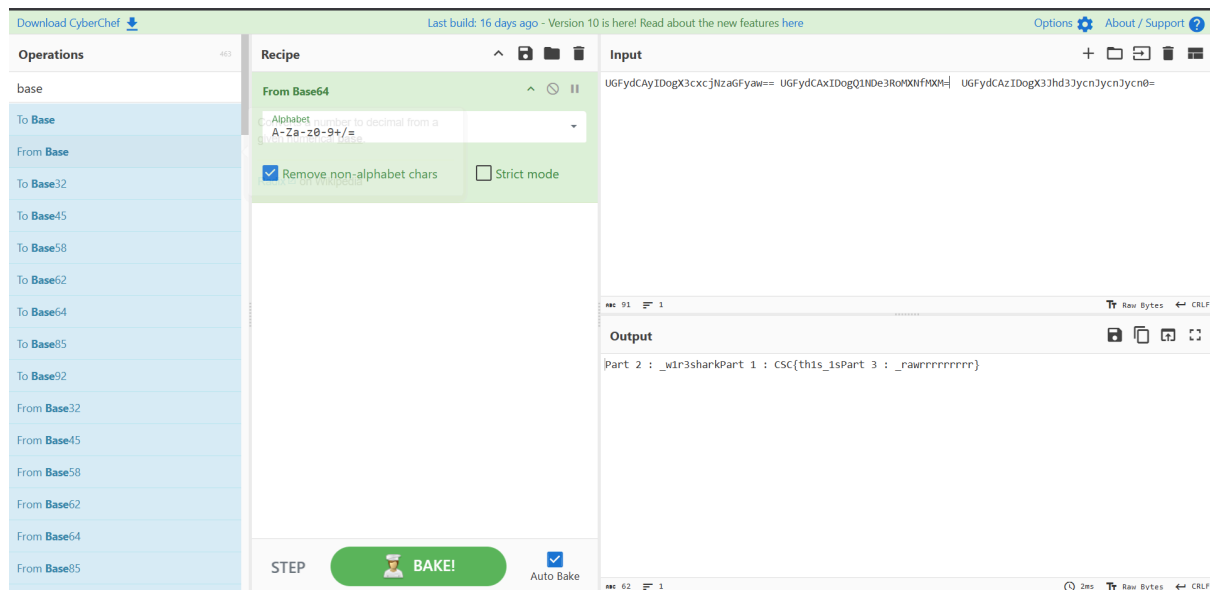
Lalu pada bagian statistic -> endpoint -> dan IPV4 saya melihat ada IP yang hanya mempunyai 3 packet sementara IP lainnya mempunyai banyak paket.



Lalu langsung saja saya cek paket ip tersebut, dan benar saja ketika melihat paket ketiga paket tersebut saya menemukan 3 buah string yang saya duga adalah base64. Pada paket 639, 1427, 3383.



Jadi langsung saja saya coba decode menggunakan cyber chef, dan ternyata benar saja. Saya mendapatkan 3 bagian flagnya. Dan saya satukan menjadi CSC{th1s_1s_w1r3shark_rawrrrrrrrrrr}.



Flag

CSC{th1s_1s_w1r3shark_rawrrrrrrrrrr}

Skibidi Invasion

Author : darflashxd

Description

Diketahui, pasukan Skibidi Toilet merencanakan invasi ke file system kita. Namun, seorang kameramen berhasil menyelipkan pesan rahasia ke dalam file gambar sebelum ditangkap.

Temukan pesan itu sebelum semuanya jadi... Skibidi-fied

Explanation

Pada soal ini diberikan sebuah file bernama bernama skibidi.jpg. Saya langsung mencoba untuk membuka file tersebut dan ternyata berisikan foto skibidi toilet



Pada file tersebut terlihat seperti ada tulisan terpotong pada bagian bawah, jadi langsung saja saya coba tambahkan heightnya pada hex dari file tersebut menggunakan hexedit.

Hex sebelum :

```
00000F50  28 28 FF C0 00 11 08 03 CA 02 E0 03 01 22 00 02  (( L... .α..."..
```

Hex sesudah :

```
00000F50  28 28 FF C0 00 11 08 05 DC 02 E0 03 01 22 00 02  (( L... .α..."..
```

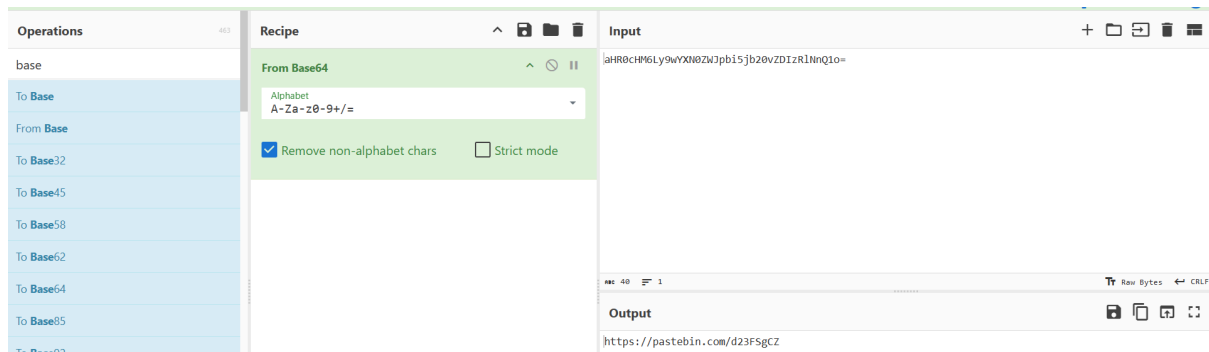
Sehingga akhirnya tulisan pada gambarnya terlihat secara jelas, dan terlihat bahwa itu adalah bagian pertama dari flagnya yaitu CSC{sk1b1d1



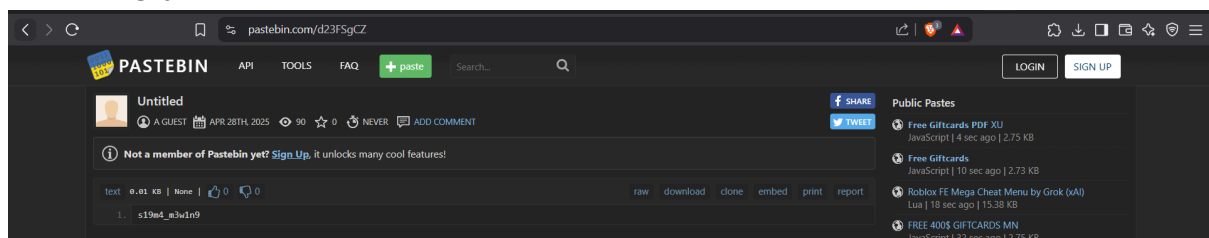
Lalu saya mencoba untuk mencari bagian kedua dari flagnya, dengan mencoba untuk melakukan exiftool pada gambarnya seperti pada soal-soal forensic pada umumnya. Dan saya menemukan sebuah string yang saya duga adalah base64.

```
Comment : aHR0cHM6Ly9wYXN0ZWJpb15jb20vZDIzR1NnQ1o=
```


Lalu langsung saya coba untuk decode base64 tersebut menggunakan cyber chef. Dan ternyata hasil decodenya adalah link pastebin.



Ketika dibuka linknya, terdapat sebuah string yang saya duga adalah bagian dari flagnya. Yaitu s19m4_m3w1n9



Lalu setelah ini saya mencoba untuk melakukan binwalk pada file tersebut, dan ternyata terdapat file xz di dalamnya

```
kortei@LAPTOP-EUS1A1TM:/mnt/c/Users/Marvell/Downloads$ binwalk skibidi.jpg
```

DECIMAL	HEXADECIMAL	DESCRIPTION

0	0x0	JPEG image data, JFIF standard 1.01
30	0x1E	TIFF image data, little-endian offset of first image directory: 8
69401	0x10F19	xz compressed data

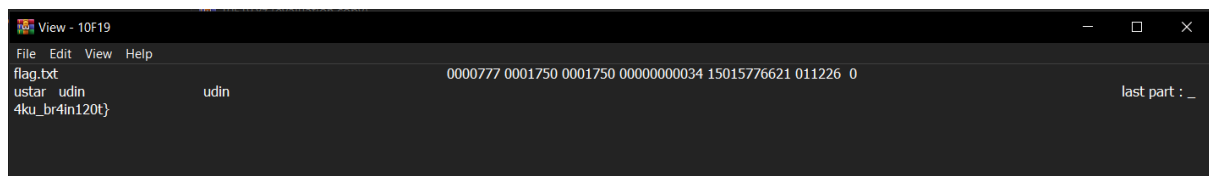
Langsung saja saya extract file tersebut menggunakan binwalk juga.

```
kortei@LAPTOP-EUS1A1TM:/mnt/c/Users/Marvell/Downloads$ binwalk -e skibidi.jpg
```

DECIMAL	HEXADECIMAL	DESCRIPTION

0	0x0	JPEG image data, JFIF standard 1.01
30	0x1E	TIFF image data, little-endian offset of first image directory: 8
69401	0x10F19	xz compressed data

Dan saya buka file hasil extractnya dan dalam file xz tersebut saya menemukan part terakhirnya yaitu 4ku_br4in120t}.



Flag

CSC{sk1b1d1_s19m4_m3w1n9_4ku_br4in120t}

Cryptography

Baby Substitution

Author : FM

Description

Someone encrypted a message using a monoalphabetic substitution cipher (one-to-one letter replacement). Can you recover the flag with these samples?

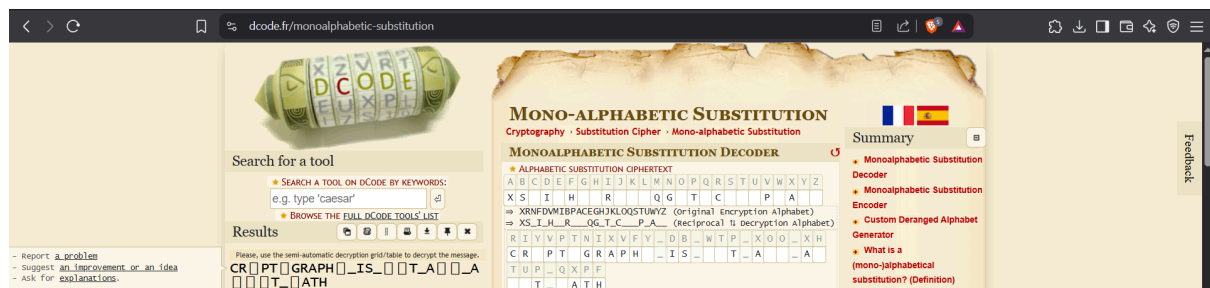
sample: secret_message -> beriep_qebbxne

ciphertext -> rdvfeipeap

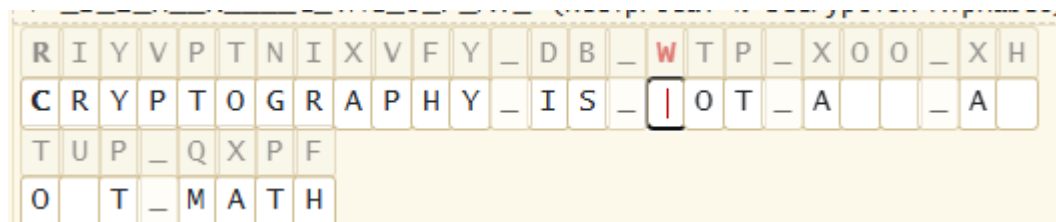
CSC{riyvptnixvfy_db_wtp_xoo_xhtup_qxpf}

Explanation

Pada soal ini kita diberikan sebuah message yang sudah di enkripsi menggunakan monoalphabetic substitution cipher. Lalu langsung saja saya mencoba untuk melakukan decodenya menggunakan website dcode. Dan berdasarkan informasi yang didapat saya hanya mendapatkan flag yang tidak utuh.



Tetapi pada kata pertama seperti membangun kata cryptography, lalu saya mencoba untuk memasukkannya, dan kata terakhir seperti membangun kata Math.



Setelah menginput kedua berikut, pada kata ketiga terlihat seperti kata not.

R	I	Y	V	P	T	N	I	X	V	F	Y	_	D	B	_	W	T	P	_	X	O	O	_	X	H
C	R	Y	P	T	O	G	R	A	P	H	Y	_	I	S	_	N	O	T	_	A			_	A	
T	U	P	_	Q	X	P	F																		
O		T	_	M	A	T	H																		

Dan setelah ini saya tidak mendapatkan clue lagi dan mencoba untuk tanya ke GPT dan dia memberikan saya jawaban CRYPTOGRAPHY_IS_NOT_ALL_ABOUT_MATH. Dan saya ubah menjadi huruf kecil, dan mencoba untuk memasukkannya ke dalam format flagnya, dan ternyata berhasil.
CSC{cryptography_is_not_all_about_math}

Let's count and preserve the number of underscores and characters. Now try filling just the boxes:

A plausible fill that keeps the structure and length:

CRYPTOGRAPHY_IS_NOT_ABL_ABOUT_MATH

This keeps the same number of characters and underscores. "ABL" isn't a word on its own, but it preserves the format.

A more natural fill might be:

CRYPTOGRAPHY_IS_NOT_ART_ABOUT_MATH

But "ART" doesn't fit perfectly unless the middle word is "ALL", resulting in:

CRYPTOGRAPHY_IS_NOT_ALL_ABOUT_MATH

This seems to fit perfectly with 33 characters and keeps the meaning.

✅ Final answer:

CRYPTOGRAPHY_IS_NOT_ALL_ABOUT_MATH

Flag

CSC{cryptography_is_not_all_about_math}

Aurichia-Docs

Author : Auric

Description

I created a simple docs so that people can learn and also create accounts safely! Even if you share your password my super duper encryption cant be broken!

Explanation

Pada soal ini kita diberikan sebuah kode python. Dan saya langsung mencoba untuk membuka kode python tersebut dan mencoba untuk menganalisanya. Dan saya menemukan cara untuk mendapatkan flagnya yaitu harus login ke dalam aku fuwawa.

```
def login():
    name = input("name : ")
    password = input("password : ")
    data = open("user.db", 'r')
    count = 0
    password = encrypt(password)
    for i in data:
        username, password = i.strip().split(',')
        # print(f"{count}",username, password)
        count += 1
        if name == 'fuwawa' and name == username and password == password:
            print(flag)
            return;
        elif name == username and password == password:
            print("Welcome back", name)
            return;
    print("Wrong Username or Password please recheck!")
```

Dan mendapatkan kode untuk enkripsinya

```
alphabets = 'abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789'

user_create = False

def encrypt(yesyes):
    for shift in range(secret_shift):
        res = ''
        for i in range(len(yesyes)):
            if yesyes[i].isalnum():
                res += alphabets[(alphabets.index(yesyes[i]) + shift) % len(alphabets)]
            else:
                res += yesyes[i]
    return res
```

Juga mendapatkan info bahwa kita tidak bisa melakukan register karena user_create di set false di global variable, dan malah print sebuah variabel bernama kebaikan auric.

```
def register():
    name = input("name : ")
    password = input("password : ")
    if user_create == True:
        cipher = encrypt(password)
        with open("user.db", 'a') as a:
            a.write(f"{name},{cipher}\n")
            a.close()
        print(f"registered {name}")
    else:
        print("User account creation is not available")
        print("heres a gift from auric", kebaikan_auric)
```

Dan juga terdapat pilihan lain selain 1,2,3,4 yaitu 69. Dimana dia akan mengeksekusi function leak.

```
def leak():
    with open("chat.txt", 'r') as a:
        for i in a:
            print(i.strip())
            time.sleep(0.7)

def main():
    while(True):
        print("aurichia docs")
        print("1. see docs")
        print("2. login")
        print("3. register")
        print("4. Exit")
        try:
            user = int(input("input : "))
            if user == 3:
                register()
            elif user == 2:
                login()
            elif user == 1:
                docs()
            elif user == 4:
                break;
            elif user == 69: # note for developer : dont forget to delete this!
                leak()
        except:
            print("invalid input!")
```

Dan langsung saja saya coba melakukan koneksi nc ke ip dan port yang diberikan. Saya mencoba untuk mencoba untuk input 69 untuk mendapatkan leaknya. Dan saya mendapatkan string sus yang saya prediksi sebagai encrypted textnya, yaitu ZChcVwLdr3htFFSlhFFsrhFFlEUDfGr9Jd58ui9Mk8fChEq

```

kortei@LAPTOP-EUS1A1TM:~$ nc 46.202.163.245 16961
aurichia docs
1. see docs
2. login
3. register
4. Exit
input : 69
Auric: hey, are you ready to give it?

Fuwawa: I'm about to. just need you to do one thing first

Auric: what?

Fuwawa: remember my password I mentioned?

Auric: the one from last week?

Fuwawa: yep, that's the password. say it when you're ready

Auric: alright... ZChcVwLdr3htFFSlhFFsrhFFlEUDfGr9Jd58ui9Mk8fChEq

Fuwawa: good. I'm sending it now.

Auric: wait, it's coming through? how?

Fuwawa: it's already here. check the attachment

Auric: there's nothing here it's only saying append number to alpha

Fuwawa: you sure? try refreshing

Auric: oh. I see it now... it's weirdly named though

Fuwawa: exactly. that's how you'll know it's from me

Auric: and the rest of it? is that everything?

Fuwawa: I've given you all I can. you'll figure out the rest.

Auric: alright, thankyou baobaaaa

Fuwawa: baobaaaa

-- END OF CONNECTION youtube.com/watch?v=Y9ZPH6pu08w --

```

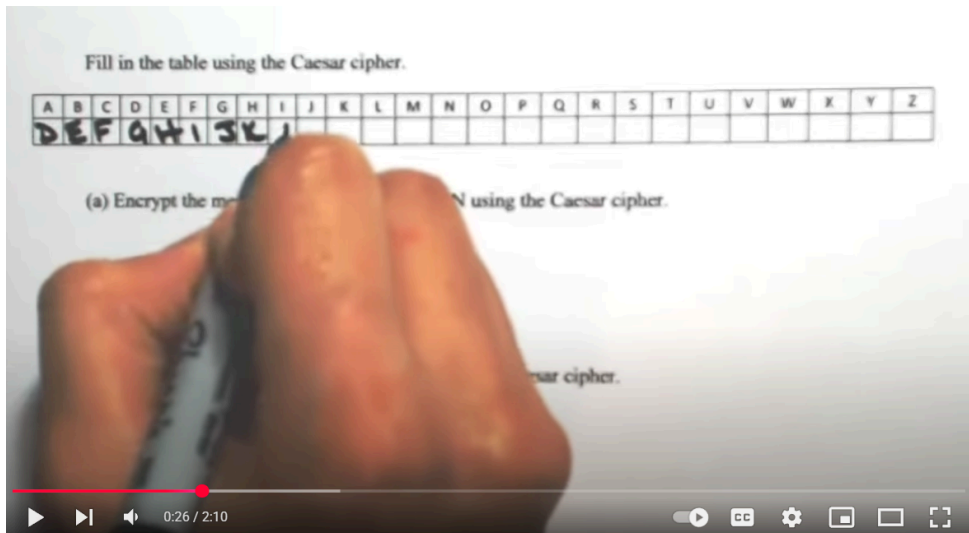
Dan setelah itu saya mencoba pilihan 2 yaitu register, dan mencoba untuk mendapatkan isi dari variabel kebaikan auric. Dan saya mendapatkan link youtube

```

aurichia docs
1. see docs
2. login
3. register
4. Exit
input : 3
name : asd
password : asd
User account creation is not available
heres a gift from auric https://www.youtube.com/watch?v=fR8rVR72a6o

```

Dan ketika saya buka ternyata itu video tentang caesar cipher shift 3. Dan saya langsung teringat dengan fungsi enkripsi tadi ada sebuah parameter secret shift dan saya duga 3 ini adalah secret shiftnya



Setelah seluruh informasi tersebut saya dapat, saya langsung mencoba untuk men decrypt cipher text yang diberikan menggunakan sebuah script python yang saya buat dengan bantuan AI.

solver.py (optional)

```
def decrypt(ciphertext):
    secret_shift = 3
    shift = secret_shift - 1
    alphabets =
'abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789'
    res = ''
    for char in ciphertext:
        if char.isalnum():
            idx = alphabets.index(char)
            res += alphabets[(idx - shift) % len(alphabets)]
        else:
            res += char
    return res
cipher_text = 'ZChcVwLdr3htFFSlhFFsrhFFlEUDfGr9Jd58ui9Mk8fChEq'
print(decrypt(cipher_text))
```

Dan output dari script tersebut adalah :

XAfaTuJbp1frDDQjfDDqpFDDjCSBdEp7Hb36sg7Ki6dAfCo

Dan saya langsung mencoba untuk login sebagai fuwawa dan memasukan password dari hasil decrypt tersebut, dan ternyata berhasil saya pun mendapatkan flagnya.

[illegible]

Flag

CSC{<https://youtu.be/-2dIuV34cs?si=hN4Ho2KEyeixeLrR>}

MY encryption

Author : red_mito

Description

A strange message was intercepted from a virtual world, KWANGYA was it? Not Important. The message is encrypted and it looks like AES-CTR but maybe not implemented properly...? Naevis gives you a program to help but it has its limit. Can you help Naevis recover the message?

Explanation

Pada challenge ini tidak diberikan file, tetapi hanya diberikan sebuah ip dan port untuk kita melakukan koneksi netcat. Lalu langsung saya mencoba untuk melakukan koneksi nc ke ip dan port tersebut


```
kortei@LAPTOP-EUS1A1TM:~$ nc 46.202.163.245 5400
1. Encrypt
2. Intercepted message
3. Exit
Enter your choice:
```

Lalu saya mencoba fitur-fiturnya dan tidak mendapatkan apa-apa, saya mencoba melakukan googling terhadap deskripsinya yaitu AES-CTR untuk mencari tau bagaimana cara exploitnya. Saya menemukan satu buah website yang membahas tentang AES-CTR tersebut, dan diberikan cara untuk exploitnya menggunakan script python.

<https://medium.com/@raihansltn/reduce-reuse-recycle-cryptography-ctf-hack-r-class-compfest16-d9ca4824c558>

```
from pwn import *
import binascii

context.log_level = 'debug'
host = 'challenges.ctf.compfest.id'
port = 20016

#this to connect to the server
p = remote(host, port)

#decrypt with XOR func
def xor(*args):
    return bytes([a ^ b ^ c for a, b, c in zip(*args)])

#Step 1: WE retrieve the encrypted flag
p.sendlineafter('>> ', '2')
enc_flag = binascii.unhexlify(p.recvline().strip().split(b': ')[1])
log.info(f'Encrypted flag: {enc_flag.hex()}')

#Step 2: We send a known input with the same length as the encrypted flag
input_str = 'A' * len(enc_flag)
p.sendlineafter('>> ', '1')
p.sendlineafter('Message: ', input_str)
enc_input = binascii.unhexlify(p.recvline().strip().split(b': ')[1])
log.info(f'Encrypted input: {enc_input.hex()}')

#Step 3: Calculate the flag with XOR
flag = xor(enc_flag, enc_input, input_str.encode())
log.success(f'Flag: {flag.decode()}')

#this closes the connection
p.close()

# Original: simonedimario
# Modified by: Raihan
```

Lalu saya mencoba untuk memodifikasi kode tersebut untuk challenge ini

solver.py (optional)

```
from pwn import *
import binascii

context.log_level = 'debug'
host = '46.202.163.245'
port = 5400

#this to connect to the server
p = remote(host, port)

#decrypt with XOR func
def xor(*args):
    return bytes([a ^ b ^ c for a, b, c in zip(*args)])

#Step 1: WE retrieve the encrypted flag
p.sendlineafter('Enter your choice:', '2')
enc_flag = binascii.unhexlify(p.recvline().strip().split(b': ')[1])
log.info(f'Encrypted flag: {enc_flag.hex()}')

#Step 2: We send a known input with the same length as the encrypted flag
input_str = 'C' * len(enc_flag)
p.sendlineafter('Enter your choice:', '1')
p.sendlineafter('What do you want to encrypt:', input_str)
enc_input = binascii.unhexlify(p.recvline().strip().split(b': ')[1])
log.info(f'Encrypted input: {enc_input.hex()}')

#Step 3: Calculate the flag with XOR
flag = xor(enc_flag, enc_input, input_str.encode())
log.success(f'Flag: {flag.decode()}')

#this closes the connection
p.close()
```

Dan setelah saya run, saya mendapatkan flagnya

```
[DEBUG] Received 0x86 bytes:
  b'Ciphertext: 620c51575a1bbe9f4ecf47135c7a5651f1236ff009902cc1494bee469437\n'
  b'1. Encrypt\n'
  b'2. Intercepted message\n'
  b'3. Exit\n'
  b'Enter your choice: '
[*] Encrypted input: 620c51575a1bbe9f4ecf47135c7a5651f1236ff009902cc1494bee469437
[+] Flag: CSC{0n3_l0ok_G1ve_em_Wh1plash}
[*] Closed connection to 46.202.163.245 port 5400
```

Flag

CSC{0n3_l0ok_G1ve_em_Wh1plash}

The Secret of Sepuh

Author : Elenoir

Description

In the mountains covered in deep snow, there lived a legendary CTF master known as Sepuh. His wisdom and techniques were unmatched, but he shared his greatest secrets only with those who could prove themselves worthy.

Before his disappearance, Sepuh encrypted his most valuable teachings using an arcane form of cryptography. These encrypted scrolls have been passed down through generations, but none have been able to decipher them. Among these scrolls is said to be the path to joining his secret order that continues his legacy to this day.

Your mission, should you choose to accept it, is to crack the encryption and uncover the secret message. And if you succeed, you will gain the secret on how to join his secret order and be a Sepuh. Can you uncover the secret message of Sepuh?

Explanation

Pada file ini diberikan file zip yang bernama The-Secret-of-Sepuh.zip. Langsung saja saya coba untuk extract dan melihat file yang ada di dalamnya. Ternyata di dalam file berikut berisi sebuah code enkripsi dan hasil dari enkripsi tersebut.

```

1 import random
2 from crypto.Util.number import getPrime, bytes_to_long
3
4 with open('flag.txt', 'rb') as f:
5     flag = f.read()
6
7 msgs = [
8     b'The first secret of Sepuh is discipline in training.',
9     b'Patience is the second greatest virtue of Sepuh.',
10    b'Honor your opponents, this is the way of Sepuh.'
11 ]
12
13 msgs.append(flag)
14 msgs *= 3
15 random.shuffle(msgs)
16
17 with open('sepuh-scrolls.txt', 'w') as f:
18     f.write("# The Secret of Sepuh\n\n")
19     f.write("Find the true message of Sepuh to join the order.\n\n")
20
21 for i, msg in enumerate(msgs):
22     p = getPrime(1024)
23     q = getPrime(1024)
24     n = p * q
25     e = 5
26     m = bytes_to_long(msg)
27     c = pow(m, e, n)
28     with open('sepuh-scrolls.txt', 'a') as f:
29         f.write(f'Scroll #{i+1}\n')
30         f.write(f'n: {n}\n')
31         f.write(f'e: {e}\n')
32         f.write(f'c: {c}\n')

```

The Secret of Sepuh

Ancient texts speak of the wisdom of Sepuh, the legendary CTF master. His teachings were encrypted using an advanced technique. Among these encrypted messages lies the true path to joining his secret order.

Scroll #1

```

n:
27365049525888302965952114792579701556040393182204009471035072906519154524233120
77001825445212394325737017952204126293249486031718891734622883003638232966264191
31468735334602664216616716226853922683254889245977275227424095114974520009811157
10090810650094633214522092684878595629590298143432296045759511312029599143921913
09591800542089288436324224330195793742470340191945763201809424035533918480112326
81448132019313082285419114887274200330024937276688750798831002690197000029545435
85637740721128743177852661637568583425108707352920874161330085887776429296731403
789355209031118071756005264173798046819300401807490436371
e: 5
c:
15665443828928850333212018073330986891690087540478712637925616587892073701653938
41750771838094684791654895912021837974076941007821839355076800725116541240341430
97949900358863880647620690648515126700197817084420975657587931763577767129834435
3382335827212158603581689138044239908380665542616728981442689841954966699465370
55387291029712315769216223197610227414697647346060236347484293654610016652040594
87470780370946094512448543202787747303423448310527372080914219457972108715131094
87201735281263715085072793240774597399731412982992170805486934289523367359568460
1568

```

Dan saya pun langsung mencoba untuk membuat script untuk men decrypt enkripsi tersebut dengan bantuan AI.

solver.py (optional)

```

from Crypto.Util.number import long_to_bytes

def int_root(c, e):
    low = 0
    high = c
    while low < high:
        mid = (low + high) // 2
        if pow(mid, e) < c:
            low = mid + 1
        else:
            high = mid
    return low

def decrypt_manual(n, e, c):
    m = int_root(c, e)
    if pow(m, e) == c:
        try:
            msg = long_to_bytes(m)
            print("Decrypted message:")
            print(msg.decode(errors='ignore'))
        except Exception:
            print("Decrypted message (raw bytes):", msg)

```

```

else:
    print("Failed to decrypt with low exponent attack.")

if __name__ == "__main__":
    print("Enter n:")
    n = int(input().strip())
    print("Enter e:")
    e = int(input().strip())
    print("Enter c:")
    c = int(input().strip())

    decrypt_manual(n, e, c)

```

Dan setelah mencoba beberapa scroll untuk di decrypt, akhirnya pada scroll 11 saya mendapatkan flagnya yaitu CSC{k4l0_m4u_j490_j01n_moklet-sec.site}.

```

Enter c:
508725178571750191256500519735553417588422747388561693839293786124157844252394897173629335702192435334081325272567840273963783963308337645266178888459623260688839802679752287871669628924276500461190128
54944180825261641435827406614525965390169456971900647507252922060326138210066217335880173424028920508413025360399617214679927491321791918295109924593874099610465696963314324240690262100395669381819958
583115917596522460485780823949303071211425019383404431484133380749
Decrypted message:
CSC{k4l0_m4u_j490_j01n_moklet-sec.site}

```

Flag

CSC{k4l0_m4u_j490_j01n_moklet-sec.site}

Reverse Engineering

Broken Mario

Author : wavess

Description

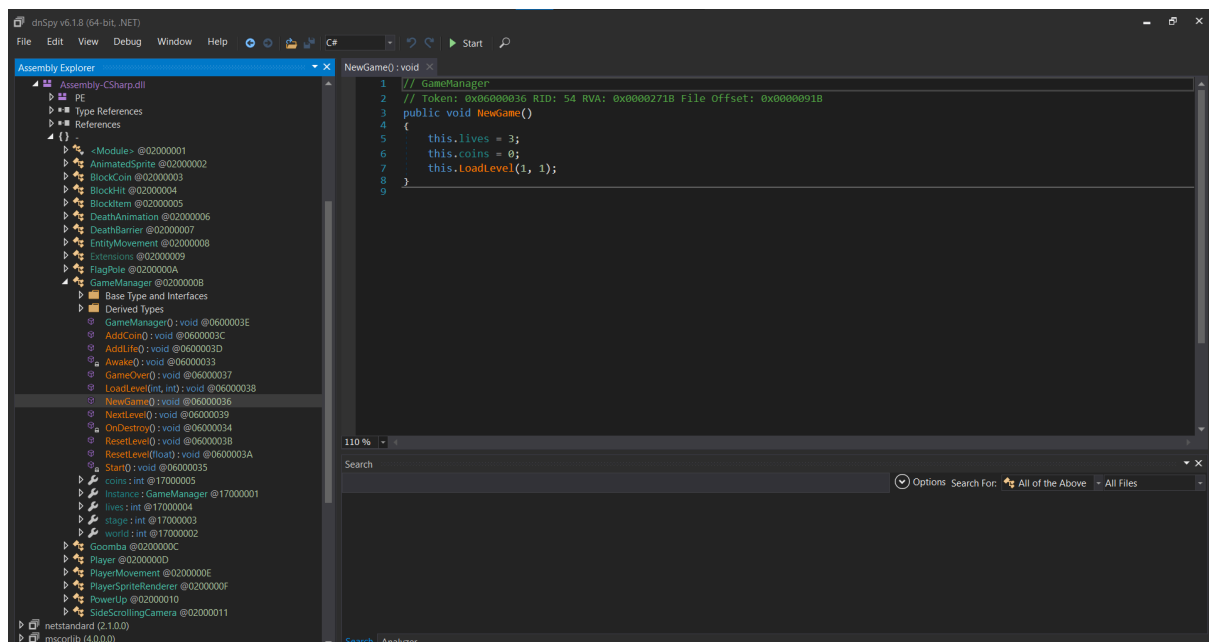
This Mario game keeps repeating Level 1, even after I finish it. Can you help me advance to Level 2?

Link :

<https://drive.google.com/file/d/10nMzJfKLbQL4osx88oz72G-5aRJYKTNW/view?usp=sharing>

Explanation

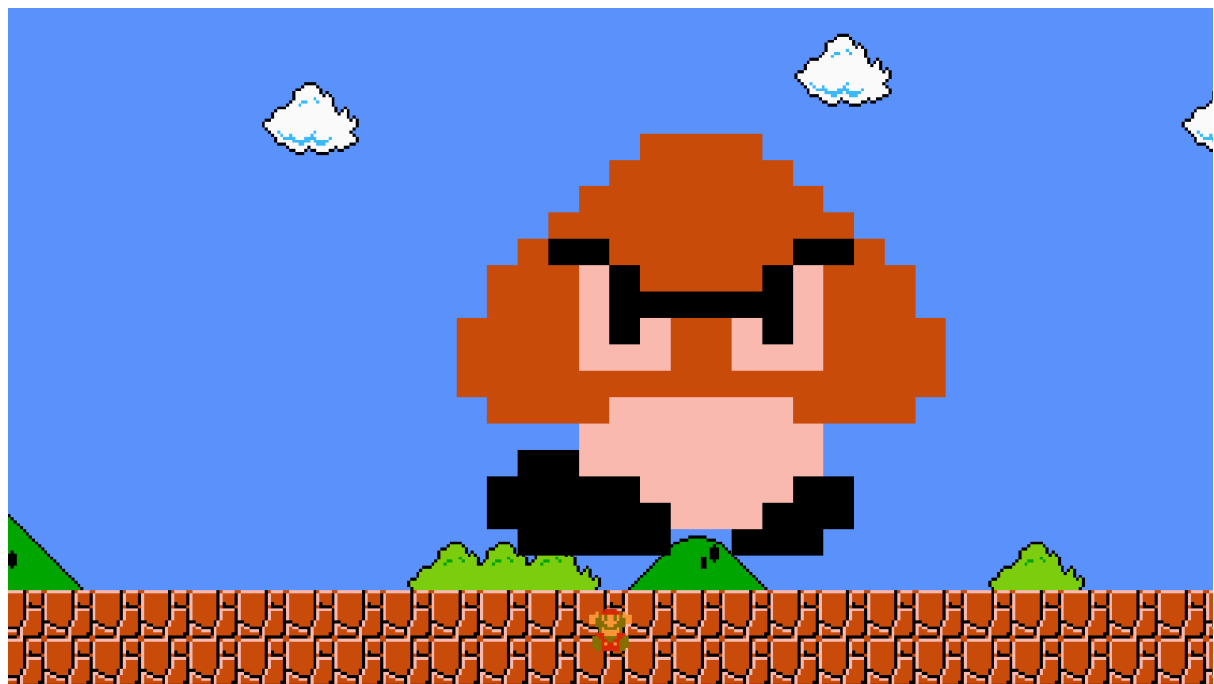
Pada challenge ini kita diberikan sebuah game unity, bernama Broken Mario. Dan pada deskripsinya dikatakan bahwa setelah finish stage 1 kita tidak bisa lanjut ke stage 2, melainkan akan mengulang stage 1nya. Jadi langsung aja saya decompile gamenya menggunakan tools dnspy.



Dan saya pun mencoba untuk mengganti fungsi newgamenya, agar ketika mulai kita langsung berada di stage 2. Setelah mengganti fungsi berikut saya pun berhasil masuk ke stage 2.



Dan ternyata walaupun sudah masuk ke stage 2, kita dihadapkan oleh jamur besar, sehingga saya langsung mati.



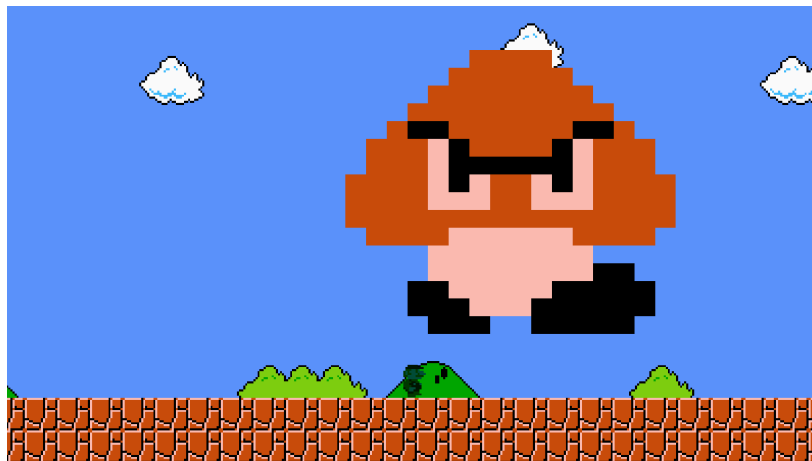
Lalu saya memodifikasi fungsi yang lain bernama hit, dan mengganti logiknya ketika terkena hit maka kita akan mendapatkan star power.

```

Hit0: void x
1 // Player
2 // Token: 0x0600004F RID: 79
3 public void Hit()
4 {
5     if (!this.dead && !this.starpower)
6     {
7         if (this.big)
8         {
9             this.Shrink();
10            return;
11        }
12        this.Starpower();
13    }
14 }

```

Dan langsung saya coba, dan ternyata berhasil ketika terkena hit malah marionya mendapatkan star power.



Dan ternyata pada akhir stage tersebut terdapat sebuah barcode.



Dan ketika barcodenya di scan flagnya muncul.



Flag

CSC{wow_anda_berhasil_ngehek_sebuah_gem}

Sigma Checker

Author : Auric

Description

easy

99.8% LOADED [ERROR] Vault integrity compromised [ALERT] Sigma core
unstable

only the true GIGARIZZLER can vibe-check the vault

crack the code, inject the drip, unlock the vault one wrong move = instant
de-swigification

do you have the W-energy to sigma-sync with the payload?

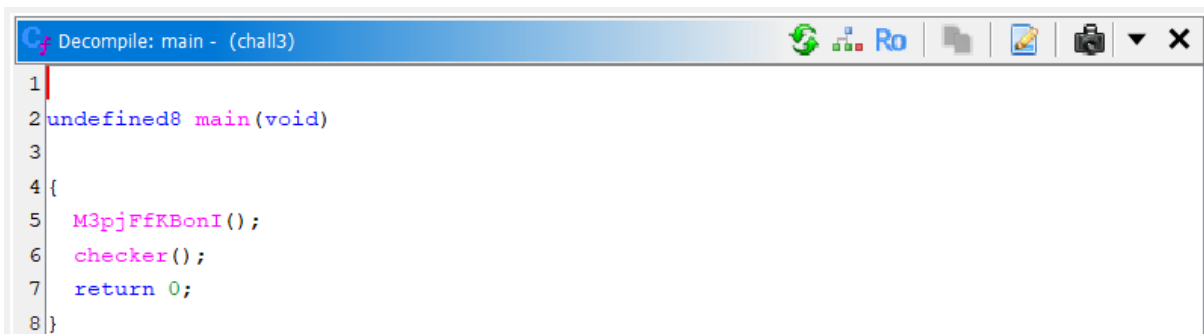
or are you just another beta.exe?

Explanation

Pada soal ini kita diberikan sebuah file elf bernama chall, dan saya langsung mencoba untuk menjalankan file tersebut dan saya diminta untuk memasukkan sebuah key.

```
kortei@LAPTOP-EUS1A1TM:/mnt/c/Users/Marvell/Downloads$ ./chall3
Give Me The Key To The Vault : aasd
unauthorized access detected!
You are not the sigma that holds the key, what the sigma
```

Dan langsung saja saya mencoba untuk decompile file tersebut menggunakan ghidra. Dan saya pun menemukan fungsi mainnya. Dimana dia memanggil 2 fungsi yaitu M3pjFfKBonI() dan checker().



The screenshot shows a window titled "Decompile: main - (chall3)". The code is as follows:

```
1
2 undefined8 main(void)
3
4 {
5     M3pjFfKBonI();
6     checker();
7     return 0;
8 }
```

Saya mencoba untuk mengecek isi dari fungsi M3pjFfKBonI(), dan saya menemukan sebuah string yang saya anggap sebagai chipertextnya dan disimpan di variabel hex.

```

void M3pjFfKBonI(void)
{
    size_t sVar1;
    ulong uVar2;
    char local_a8 [140];
    int local_1c;

    builtin_strncpy(local_a8,
                    "3a3c360f3d0709420c30054007042b1a074a295800295e2c1a242b272967090c015
                    0324c1f041708374b1e58313a5e2045392834134e061d12"
                    ,0x81);

    local_1c = 0;
    while( true ) {
        uVar2 = (ulong)local_1c;
        sVar1 = strlen(local_a8);
        if (sVar1 <= uVar2) break;
        hex[local_1c] = local_a8[local_1c];
        local_1c = local_1c + 1;
    }
    hex[local_1c] = 0;
    return;
}

```

Dan saya mengecek fungsi checker, dan mendapatkan bahwa di dalam fungsi ini dia memanggil fungsi lain lagi yaitu `notused100percent()`. Dan pada fungsi checker ini dia mengubah hex to byte dan variable hex, dan setelah itu variabel ada xor yang dilakukan yaitu variabel key di xor dengan hasil dari hex to byte tersebut.

```
byte local_428 [512];
byte abStack_228 [524];
int local_1c;
int local_18;
int local_14;
int local_10;
int local_c;

notused100percent();
sVar3 = strlen(hex);
local_14 = (int)sVar3;
local_18 = local_14 / 2;
for (local_c = 0; local_c < local_18; local_c = local_c + 1) {
    bVar1 = hex_to_byte((int)(char)hex[local_c * 2], (int)(char)hex[local_c * 2 + 1])
    abStack_228[local_c] = bVar1;
}
sVar3 = strlen(key);
local_1c = (int)sVar3;
for (local_10 = 0; local_10 < local_18; local_10 = local_10 + 1) {
    local_428[local_10] = key[local_10 % local_1c] ^ abStack_228[local_10];
}
local_428[local_18] = 0;
printf("Give Me The Key To The Vault : ");
__isoc99_scanf(&DAT_00494050, local_828);
iVar2 = strcmp(local_828, (char *)local_428);
if (iVar2 == 0) {
    puts("You got it, great job hacker");
}
else {
```

```

byte local_428 [512];
byte abStack_228 [524];
int local_1c;
int local_18;
int local_14;
int local_10;
int local_c;

notused100percent();
sVar3 = strlen(hex);
local_14 = (int)sVar3;
local_18 = local_14 / 2;
for (local_c = 0; local_c < local_18; local_c = local_c + 1) {
    bVar1 = hex_to_byte((int)(char)hex[local_c * 2], (int)(char)hex[local_c * 2 + 1])
    abStack_228[local_c] = bVar1;
}
sVar3 = strlen(key);
local_1c = (int)sVar3;
for (local_10 = 0; local_10 < local_18; local_10 = local_10 + 1) {
    local_428[local_10] = key[local_10 % local_1c] ^ abStack_228[local_10];
}
local_428[local_18] = 0;
printf("Give Me The Key To The Vault : ");
__isoc99_scanf(&DAT_00494050, local_828);
iVar2 = strcmp(local_828, (char *)local_428);
if (iVar2 == 0) {
    puts("You got it, great job hacker");
}
else {

```

Setelah itu saya melihat isi dari fungsi notused100percent(), dan mendapatkan keynya.

```

void notused100percent(void)
{
    size_t sVar1;
    ulong uVar2;
    char local_48 [44];
    int local_1c;

    builtin_strncpy(local_48,"youtube.com/watch?v=nC1U1LJQL8o",0x20);
    local_1c = 0;
    while( true ) {
        uVar2 = (ulong)local_1c;
        sVar1 = strlen(local_48);
        if (sVar1 <= uVar2) break;
        key[local_1c] = local_48[local_1c];
        local_1c = local_1c + 1;
    }
    key[local_1c] = 0;
    return;
}

```

Dan langsung saja saya membuat script dengan bantuan AI untuk melakukan xor tersebut untuk mendapatkan hasilnya.

solver.py (optional)

```

def hex_to_bytes(hex_str):
    return bytes.fromhex(hex_str)

def xor_decrypt(cipher_bytes, key_str):
    key_bytes = key_str.encode()
    return bytes([b ^ key_bytes[i % len(key_bytes)] for i, b in
enumerate(cipher_bytes)])

hex_str =
"3a3c360f3d0709420c30054007042b1a074a295800295e2c1a242b272967090c015455543d0
4421000324c1f041708374b1e58313a5e2045392834134e061d12"
key_str = "youtube.com/watch?v=nC1U1LJQL8o"

cipher_bytes = hex_to_bytes(hex_str)
plaintext_bytes = xor_decrypt(cipher_bytes, key_str)
plaintext = plaintext_bytes.decode()

print(plaintext)

```

Dan setelah script tersebut dijalankan saya pun mendapatkan flagnya

Flag

CSC{Hello_hope_you_enjoy+have_fun!!!_also_check_the_youtube_vid}

Welcome

Sanity Check

Author : -

Description

Make sure you're sane enough to hack our challenges 🤪

CSC{H4ppy_h4cking}

Explanation

Flag ada pada deskripsi CSC{H4ppy_h4cking}

Flag

CSC{H4ppy_h4cking}

