

# VYctf2023 wp by Y7syau

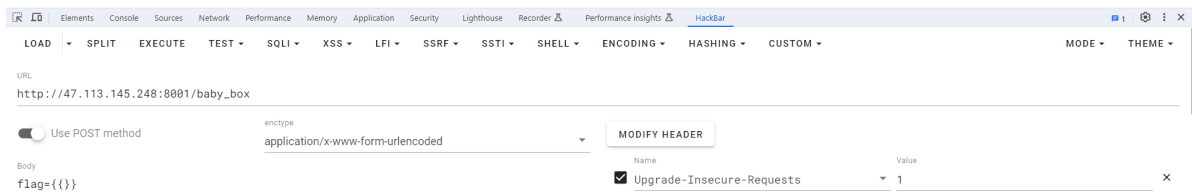
## WEB

### 玩蛇(签到)

抓包即可

请求	Raw	Hex	美化	Raw	Hex	页面渲染
1	GET /python HTTP/1.1		151	case 38: togo=2;		
2	Host: 47.113.145.248:8001		152	break;		
3	Pragma: no-cache		153	case 39: togo=3;		
4	Cache-Control: no-cache		154	break;		
5	Upgrade-Insecure-Requests: 1		155	case 40: togo=4;		
6	User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.0.0 Safari/537.36		156	break;		
7	Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7		157	case 65: togo=5;		
8	Referer: http://121.41.107.43:8000/		158	break;		
9	Accept-Encoding: gzip, deflate		159	case 68: togo=6;		
10	Accept-Language: zh-CN,zh;q=0.9		160	break;		
11	Connection: close		161	}		
12			162	}		
13			163			
			164	function isEat()//吃到食物后长度加1		
			165	{		
			166	if (snake[snakeCount-1].x==foodx&&snake[snakeCount-1].y==foody)		
			167	{		
			168	addfood();		
			169	snakeCount++;		
			170	snake.unshift({		
			171	x:-15,y:-15		
			172	});		
			173	if (snakeCount==102){		
			174	alert('恭喜你拿到了flag: VYctf{Pyth0n_15_thE_be5t_L4ngu4ge}')		
			175	}		
			176	}		
			177			
			178			
			179			
			180			
			181			
			182			
			183			
			184			
			185			
			186			
			187			
			188			
			189			
			190			
			191			
			192			
			193			
			194			
			195			
			196			
			197			
			198			
			199			
			200			
			201			
			202			
			203			
			204			
			205			
			206			
			207			
			208			
			209			
			210			
			211			
			212			
			213			
			214			
			215			
			216			
			217			
			218			
			219			
			220			
			221			
			222			
			223			
			224			
			225			
			226			
			227			
			228			
			229			
			230			
			231			
			232			
			233			
			234			
			235			
			236			
			237			
			238			
			239			
			240			
			241			
			242			
			243			
			244			
			245			
			246			
			247			
			248			
			249			
			250			
			251			
			252			
			253			
			254			
			255			
			256			
			257			
			258			
			259			
			260			
			261			
			262			
			263			
			264			
			265			
			266			
			267			
			268			
			269			
			270			
			271			
			272			
			273			
			274			
			275			
			276			
			277			
			278			
			279			
			280			
			281			
			282			
			283			
			284			
			285			
			286			
			287			
			288			
			289			
			290			
			291			
			292			
			293			
			294			
			295			
			296			
			297			
			298			
			299			
			300			
			301			
			302			
			303			
			304			
			305			
			306			
			307			
			308			
			309			
			310			
			311			
			312			
			313			
			314			
			315			
			316			
			317			
			318			
			319			
			320			
			321			
			322			
			323			
			324			
			325			
			326			
			327			
			328			
			329			
			330			
			331			
			332			
			333			
			334			
			335			
			336			
			337			
			338			
			339			
			340			
			341			
			342			
			343			
			344			
			345			
			346			
			347			
			348			
			349			
			350			
			351			
			352			
			353			
			354			
			355			
			356			
			357			
			358			
			359			
			360			
			361			
			362			
			363			
			364			
			365			
			366			
			367			
			368			
			369			
			370			
			371			
			372			
			373			
			374			
			375			
			376			
			377			
			378			
			379			
			380			
			381			
			382			
			383			
			384			
			385			
			386			
			387			
			388			
			389			
			390			
			391			
			392			
			393			
			394			
			395			
			396			
			397			
			398			
			399			
			400			
			401			
			402			
			403			
			404			
			405			
			406			
			407			
			408			
			409			
			410			
			411			
			412			
			413			
			414			
			415			
			416			
			417			
			418			
			419			
			420			
			421			
			422			
			423			
			424			
			425			
			426			
			427			
			428			
			429			
			430			
			431			
			432			
			433			
			434			
			435			
			436			
			437			
			438			
			439			
			440			
			441			
			442			
			443			
			444			
			445			
			446			
			447			
			448			
			449			
			450			
			451			
			452			
			453			
			454			
			455			
			456			
			457			
			458			
			459			
			460			
			461			
			462			
			463			
			464			
			465			
			466			
			467			
			468			
			469			
			470			
			471			
			472			
			473			
			474			
			475			
			476			

干得漂亮! flag是vyctf{th1s\_is\_c0de9ate\_baby\_b0x}



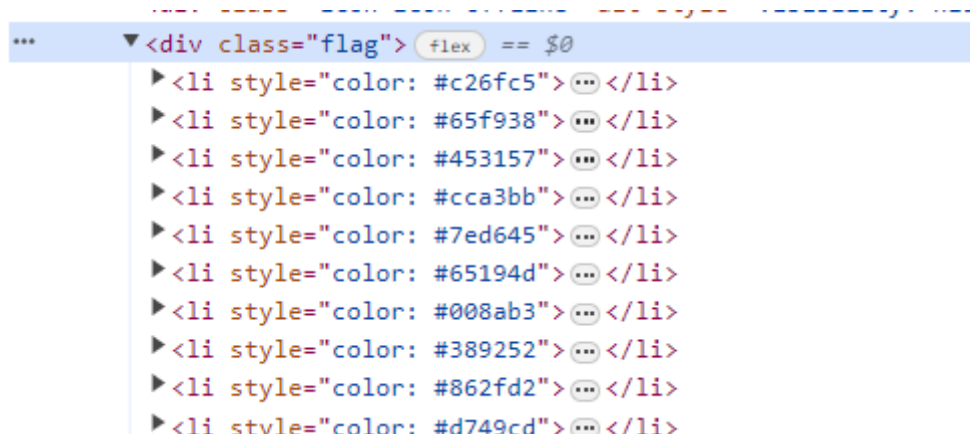
```
if request.method == "POST":  
  
    payload = request.form['flag']  
  
    result = sandbox(payload)
```

```
if len(payload) > 0x8:  
  
    return '这可太长了!'  
  
try:  
  
    to_feed = base64.b64decode(payload)  
  
except:  
  
    return '这可不是base64!'
```

从源码可以得到post传参flag 然后经典{}

## 小恐龙

hint:多余的颜色是不是在暗示什么?



源码有如下 再通过hint尝试把color提取出来 头部一眼十六进制png

```
89504e470d0a1a0a0000000d494844520000001d0000001d0802000000d9f1f05800000009704859
7300002e2300002e230178a53f760000018049444154484b8d55ed1283300c9adedeff95b7f6d221
05a2f387d72f532024be5ee9f9fc9eda1cb37af300e318e18dcfb07d1cc7188f3702d5b8d6b18569
1761c1e9a0311cc1c80c70acce9c9185c4aaa38cb422326489d3c62dee82c2f176b0a6be2e50ad00
4e8d236ab9f8813de3fa67ece136b237b715f6083c7e75c2559516bc655aa845748ec82e9c7ee0e4
f097226847651c0315a4eaf24974a2a43416a13812d355426e0941e1fe9503c26fc565f938453157
cca3bb7ed64565194d008ab3389252862fd2d749cd1bfbe1126bef3b8ff7cd20ec0726156b1f4528
b5172c88cb4b10168157a2389e6d7866eae2d0675057b4a0988eb016863890e104fb39a6c613fbc
c117e24e0daee20614cb8165bcba09227a13f09a86dff82e548a1c3de82b5a5f0e10a2677d392ae
bb99917489ce36b2defe8f852017b717ba94dc9ab2149c1cb7737474cce1fabff9859dee2c0b575d
ad6f2adfe095fbd8cbd8f2121d5beb3fcf86e7ec0bead81585c1ea8b5172d4b1ef82a9df814e3bbe
fa0201257ae732ac524a0000000049454e44ae426082
```

用010导入



得到一个二维码扫码出flag

(misc无处不在

## MISC

### 缺少的专辑(签到)

起始页 × Dream\_of\_Butterfly.png ×

简单改高度

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	0
:0000	89	50	4E	47	0D	0A	1A	0A	00	00	00	0D	49	48	44	52	:
:0010	00	00	00	FA	00	00	00	FA	08	06	00	00	00	88	EC	5A	.



## 这亦是一种图片（足够抽象

hint: 如果看不见图片, 不要心急, 也许图片正以另一种形式存在着, 观察它的名字.

hint: 世界上也许不只有十进制和十六进制

那只有xxd二进制

```
xia@xia: ~  
文件 动作 编辑 查看 帮助  
00000006: 00011010 00001010 00000000 00000000 00000000 00001101 .....  
0000000c: 01001001 01001000 01000100 01010010 00000000 00000000 IHDR..  
00000012: 00000011 11100000 00000000 00000000 00000000 00000000 .....  
00000018: 00000000 00011111 10000000 00000000 00000000 00000000 .....  
0000001e: 00000000 00000000 01111111 11100000 00000000 00000000 .....  
00000024: 00000000 00000000 00000000 00011111 11100000 00000000 .....  
0000002a: 00000000 00000000 00000000 00000000 01110000 00000000 ....p.  
00000030: 00000000 00000000 00000000 00000001 11000000 00000000 .....  
00000036: 00000000 00000000 00000000 11111110 00000000 00000000 .....  
0000003c: 00000000 00000000 00011111 00000000 00000000 00000000 .....  
00000042: 00000000 00000001 11110000 00000000 00000000 00000000 .....  
00000048: 00000000 00111111 00000000 00000000 00000000 00000000 .?....  
0000004e: 00000001 11100000 00000000 00000000 00000000 00000000 .....  
00000054: 00000000 00000000 00000000 00000000 00000000 00000000 .....  
0000005a: 00000000 00000000 00000000 00000000 00000000 00000000 .....  
00000060: 00000001 11000000 00000000 00000000 00000000 00000000 .....  
00000066: 00000000 00111100 00000000 00000000 00000000 00000000 .<....  
0000006c: 00000000 00000111 10000000 00000000 00000000 00000000 .....  
00000072: 00000000 00000000 11111111 11111111 11100000 00000000 .....  
00000078: 00000000 00000001 11000000 00000000 00000000 00000000 .....  
0000007e: 00000000 00000111 00000000 00000000 00000000 00000000 .....  
00000084: 00000000 00011100 00000000 00000000 00000000 00000000 .....  
0000008a: 00000000 11110000 00000000 00000000 00000000 00000000 .....  
00000090: 00000001 10000000 00000000 00000000 00000000 00000000 .....  
00000096: 00000000 00000000 00000000 01111111 10000000 00000000 .....  
0000009c: 00000000 00000000 00000000 11000000 11000000 00000000 .....  
000000a2: 00000000 00000000 00000001 10000000 01000000 00000000 ....@.  
000000a8: 00000000 00000000 00000001 00000000 01000000 00000000 ....@.  
000000ae: 00000000 00000000 00000000 10000000 01000000 00000000 ....@.  
000000b4: 00000000 00000000 00000000 00000000 10000000 00000000 .....  
000000ba: 00000000 00000000 00000000 00000000 00000000 00000000 .....
```

就这样慢慢认

## 雪(snow)

```
D:\misc\snow>.\snow.exe -C D:\misc\snow\12.html  
vyctf{5n0w_15_834u71fu1}
```

## crypto

# 古老的语言(签到)

转换为python直接跑

```
import os
import sys

def interpret_brainfuck(input_file):
    try:
        with open(input_file, 'r', encoding='utf-8') as file:
            program = file.read()
    except FileNotFoundError:
        print('\033[31m[false] \033[0m找不到文件')
        return

    output_file = input_file[:-3] + '.txt' if input_file.endswith('.bf') else 'default.txt'

    with open(output_file, 'w', encoding='utf-8') as new_file:
        memory = [0] * 256
        address = 0
        stack = []
        program_counter = 0

        while program_counter < len(program):
            instruction = program[program_counter]

            if instruction == 'O':
                address += 1
            elif instruction == 'W':
                address -= 1
            elif instruction == '*':
                memory[address] += 1
            elif instruction == '@':
                memory[address] -= 1
            elif instruction == '.':
                data = chr(memory[address])
                new_file.write(data)
                print(data, end='')
            elif instruction == ',':
                input_data = input('')
                memory[address] = ord(input_data[0])
            elif instruction == 'v':
                stack.append(program_counter)
            elif instruction == '~':
                if memory[address] != 0:
                    program_counter = stack[-1]
                else:
                    stack.pop()
            else:
                pass

            program_counter += 1
```

```

•      print("")

if __name__ == "__main__":
    if len(sys.argv) < 2:
        print('\033[31m[false] \033[0m请至少添加一个文件!')
        sys.exit(1)

•      input_file = sys.argv[1]
•      interpret_brainfuck(input_file)

```

```
vYctf{we1c0me_t0_crypt0}
```

## 素数分解

```

# 简单的rsa加密技术

E = 7

N = P * Q

\# N = 2771

phin = (P-1) * (Q-1)

D = pow(E, -1, phin)

\# print(D)

\# D = 1111

PT = open("./flag.ct", "w")

with open("./flag.pt", "r") as file:

    for f in file.read():

•      PT.write(chr((ord(f) ** E) % N))

PT.close()

```

题给了加密 由题素数分解可以直接求出PQ

```

# 已知的值
P = 17
Q = 163
D = 1111
N = P * Q

```

```

\# 解密函数
def rsa_decrypt(ciphertext, D, N):
    plaintext = ""
    for char in ciphertext:
        num = ord(char)
        m = pow(num, D, N)
        plaintext += chr(m)
    return plaintext

\# 读取密文
with open("./flag.ct", "r", encoding="utf-8") as file:
    ciphertext = file.read()

\# 解密
plaintext = rsa_decrypt(ciphertext, D, N)

\# 将解密后的明文写入文件
with open("./decrypted_message.txt", "w", encoding="utf-8", errors="ignore") as output_file:
    output_file.write(plaintext)

print("Decrypted message:", plaintext)

```

```
vyctf{R5a_1s_M0dern_pA55w0rd}
```

## 小小的也很可爱哦

```

P = 487

\# D = ?

E1 = 31

\# E2 = pow(E1, D, P)

E2 = 168

R = 11

def enc(PT, E1, E2, R, P):

    C1 = pow(E1, R, P)

    C2 = ""

    for i in PT:

        • data = (i * pow(E2, R)) % P

```

```

•   C2 += chr(data)

    return c1,c2

with open("./flag.pt","rb") as PT:

    C1,C2 = enc(PT.read(), E1, E2, R, P)

with open("./flag.ct","w") as CT:

    CT.write("C1 is:"+str(C1)+"\nC2 is:"+C2)

print("C1 is:"+str(C1)+"\nC2 is:"+C2)

```

在附件flag.ct中告诉了C1的值 直接通过源代码求私钥

```

P = 487
E1 = 31
C1 = 162
R = 11

def find_all_possible_d_values(P, E1, R, C1):
    possible_d_values = []
    for D in range(P):
        \# 尝试解密
        decrypted_C1 = pow(E1, R * D, P)
        if decrypted_C1 == C1:
            possible_d_values.append(D)
    return possible_d_values

all_possible_D_values = find_all_possible_d_values(P, E1, R, C1)

if all_possible_D_values:
    for D in all_possible_D_values:

        E2 = pow(E1, D, P)

        •   print(f"Found D: {D}")
        •   print(f"Calculated E2: {E2}")
        •   print(f"Decrypted C1: {pow(E1, R * D, P)} (should match C1: {C1})")
        •   print("-" * 20)
    else:
        •   print("No valid D found.")

```

求出D=244然后直接做



```

# 定义解密函数
def decrypt(D, C2, E2, R, P):
    decrypted_data = ""
    for char in C2:
        decrypted_char = (ord(char) * pow(pow(E2, R, P), -D, P)) % P
        decrypted_data += chr(decrypted_char)
    return decrypted_data

\# 已知的参数
P = 487
E1 = 31
E2 = 168
R = 11
D = 244 # 你已知的 D 的值

\# 以二进制模式读取文件内容
file_path = "D:/chrome/elgamal/flag.ct"
with open(file_path, "rb") as file:
    content = file.read()

\# 获取 C2 密文部分
c2_index = content.find(b"C2 is:") # 查找 C2 的位置
if c2_index != -1:
    C2 = content[c2_index + len(b"C2 is:"):].decode().strip()
    \# 解密数据
    decrypted_data = decrypt(D, C2, E2, R, P)
    \# 输出解密结果
    print("Decrypted Data:", decrypted_data)
else:
    print("Error: C2 not found in file.")

\# 打印文件内容的十六进制表示
hex_content = " ".join(format(byte, '02x') for byte in content)
print("File Content (Hex):", hex_content)

```

```
VYctf{ElG4m4l_15_4n_45ymmetr1c_encrypt10n_4lg0r1thm}
```

## iot

### 简单ino(签到)

hint: 什么, 这不只是一个单纯的程序, 也许能从元件上得到灵感

hint: 显示屏大小为16x2

```

// lcd1602:SCL is uno:A5, lcd1602:SDA is uno:A4, lcd1602:VCC is num:V5,
lcd1602:GND is uno:GND.

#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27, 20, 4);

```

```

int flag[20] = {118, 121, 99, 116, 102, 123, 104, 101, 49, 108, 48, 95, 65, 114, 100, 117, 49, 110, 48, 125};
int line[20] = {10, 3, 14, 4, 0, 13, 10, 3, 14, 0, 14, 0, 0, 7, 13, 5, 14, 0, 14, 7};
int i = 0;

void setup() {
  lcd.init();
  lcd.backlight();
  lcd.setCursor(0, 0);
  lcd.print("Hello vYctf!");
}

void loop() {
  delay(1000);
  lcd.clear();
  lcd.print("flag is:");
  lcd.setCursor(line[i], 1);
  lcd.print(flag[i]);
  i++;
}

```

读代码的flag入手 118由ascii对照为v 尝试拿去转ASCII（剔除大于126的）

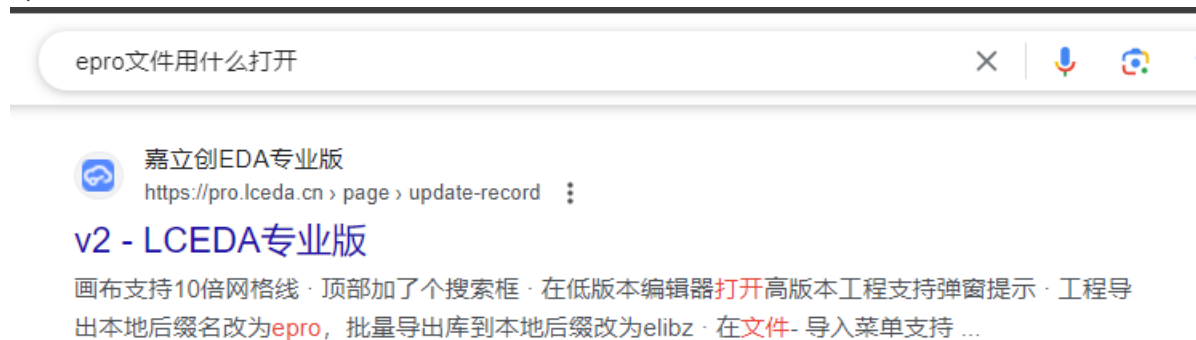
118, 121, 99, 116, 102, 123, 104, 101, 49, 108, 48, 95, 65, 114, 100, 117, 49, 110, 48, 125

vYctf{he1l0\_Ardu1n0}

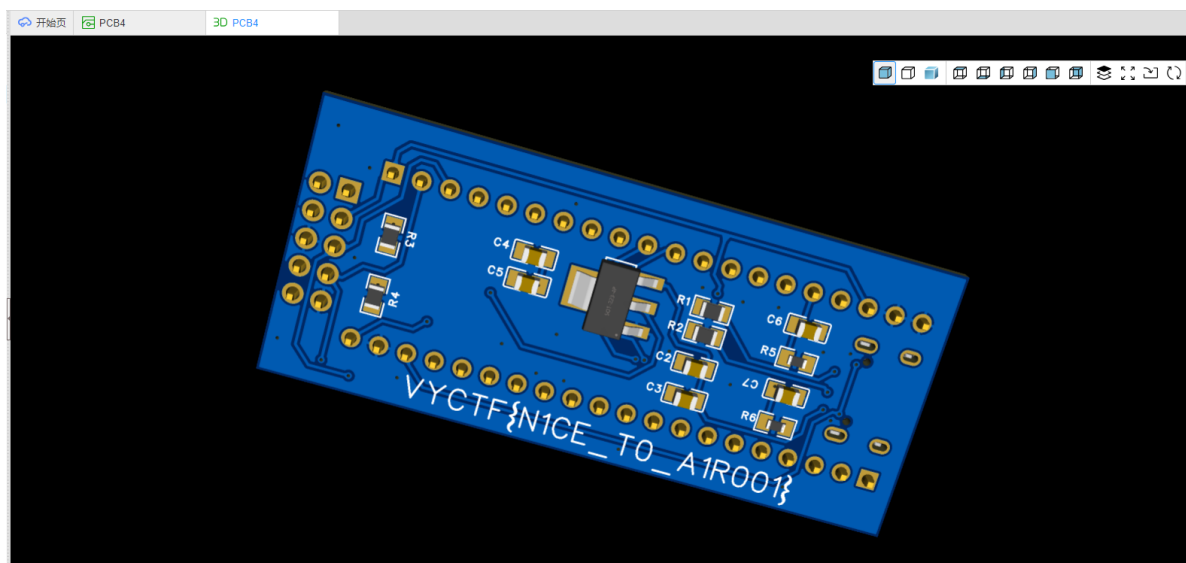
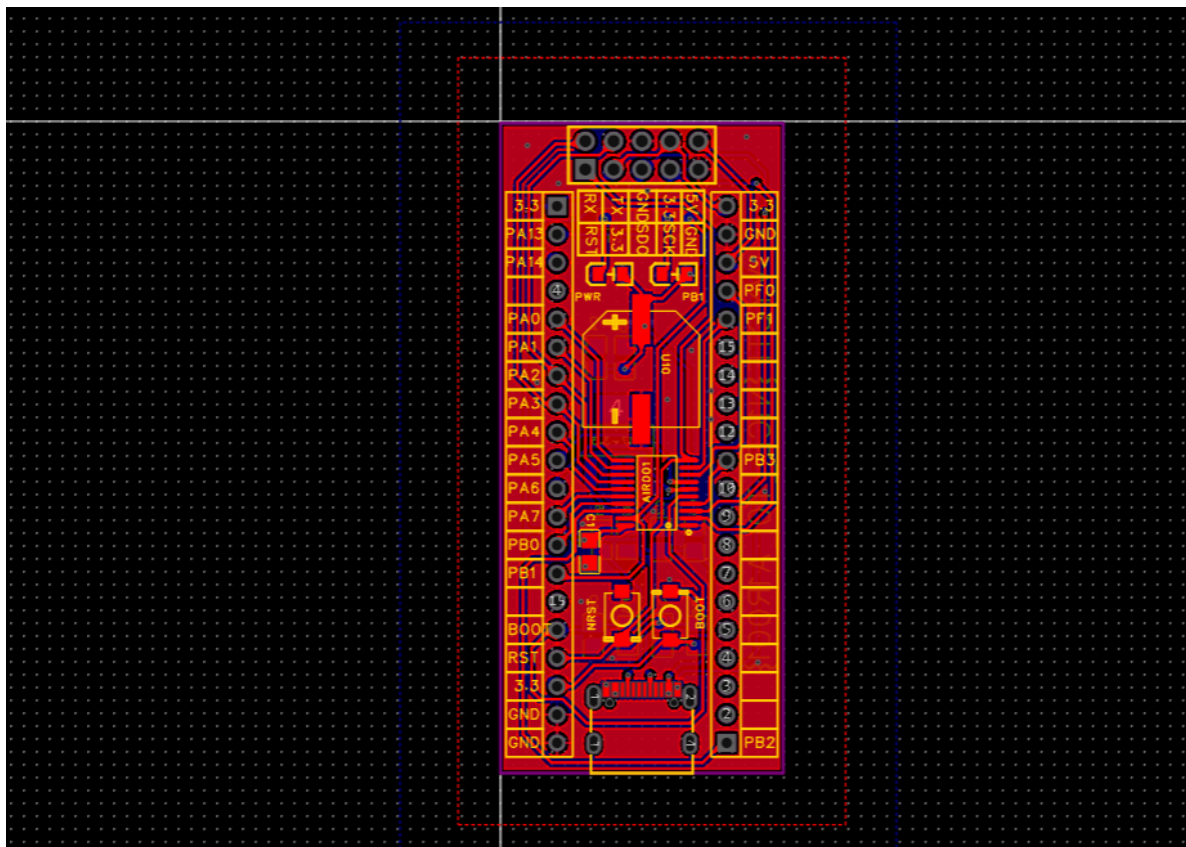
不太懂iot 唉

## Air001

epro文件（（（直接网上搜发现是



就尝试瞎整 正常导入后是个二维的



瞎捣鼓后换成3D展示 得到flag

**reverse**

大家一起和平地玩耍吧(签到)



## base64逆向

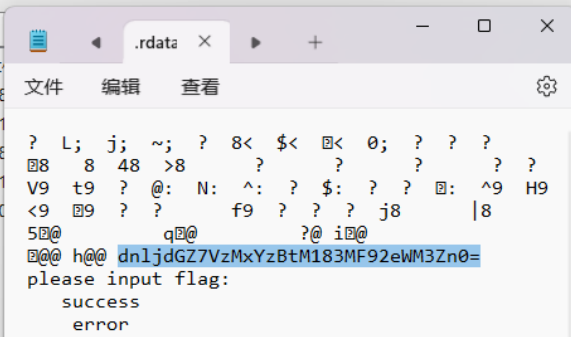
打开文件随便看

文件(F) 编辑(E) 查看(V) 窗口(W) 工具(I) 帮助(H)

添加 提取 测试 复制 移动 删除 信息

D:\chrome\ez\_base64.exe\

名称	大小	压缩后大小
.rsrc	381	381
.data	512	512
.rdata	3 584	3 584
.reloc	512	512
.text	4 608	4 608



文件夹 1

dnIjdGZ7VzMxYzBtM183MF92eWM3Zn0=

疑似base64

编码 (Encode)

解码 (Decode)

↕ 交换

Base64 编码或解码的结果:

vyctf{W31c0m3\_70\_vyc7f}

不太懂re

## 二进制

hint: 注意main函数附近

ida打开f5查看

```
__int64 __fastcall shl_flag(int a1)
{
    return (unsigned int)(a1 >> 1);
}
```

```

unsigned __int64 v8; // [rsp+F8h] [rbp-18h]

v8 = __readfsqword(0x28u);
v6[0] = 236;
v6[1] = 242;
v6[2] = 198;
v6[3] = 232;
v6[4] = 204;
v6[5] = 246;
v6[6] = 166;
v6[7] = 208;
v6[8] = 216;
v6[9] = 190;
v6[10] = 98;
v6[11] = 230;
v6[12] = 190;
v6[13] = 154;
v6[14] = 96;
v6[15] = 236;
v6[16] = 202;
v6[17] = 190;
v6[18] = 232;
v6[19] = 208;
v6[20] = 202;
v6[21] = 190;
v6[22] = 196;
v6[23] = 98;
v6[24] = 220;
v6[25] = 104;
v6[26] = 228;

```

直接遍历数组，除以二，转换为ASCII并打印

```

v6 = [236, 242, 198, 232, 204, 246, 166, 208, 216, 190, 98, 230, 190, 154, 96,
236, 202, 190, 232, 208, 202, 190, 196, 98, 220, 104, 228, 242, 190, 232, 96,
190, 232, 208, 202, 190, 216, 202, 204, 232, 250]

```

```

for num in v6:
    num //= 2
    ascii_char = chr(num)
    print(ascii_char, end='')

print()

```

```

vyctf{Sh1_1s_M0ve_the_b1n4ry_t0_the_left}

```

## virus

### kawaii病毒

hint: 这么kawaii的病毒, 如果依它所想做点什么, 它会不会告诉你flag呢?

hint:

```

int filenum()
{

```

```
WIN32_FIND_DATA findFileData;
HANDLE hFind = FindFirstFile("*.txt", &findFileData);
if (hFind == INVALID_HANDLE_VALUE)
{
    printf("Can't find file");
    return 1;
}
int fileCount = 0;
do {
    if (findFileData.dwFileAttributes & FILE_ATTRIBUTE_DIRECTORY) {
        continue;
    }
    fileCount++;
}
while (FindNextFile(hFind, &findFileData) != 0);
FindClose(hFind);
return fileCount;
}
```

跟着说的来就行 随便搞个目录建个文件再删掉

新加卷 (D:) > chrome > 123				在 123 ...
名称	修改日期	类型	大小	
virus.exe	2023/11/19 22:58	应用程序	138 KB	

