JONSCAFE

Cyberyolk 2023 – under username k.eii

Reverse Engineering Write Ups

Decompiler used: IDA Pro, jadx-gui

IDE: VS Code

PasswordGenerator

```
┌──(jons㉿01-20-jonathansebastian)-[~/download]
└─$ ./PasswordGenerator
Generate a random password!
Guess the password..
Input : tes
Nope.. I don't feel its good enough..
[VaultS3curity$] is better
```

Diberikan ELF PasswordGenerator, coba jalankan dan diminta untuk menebak password.

Decompile, hasil:

```
__int64 __fastcall main(int a1, char **a2, char **a3)

{
  unsigned int v3; // eax
  char s1[104]; // [rsp+0h] [rbp-70h] BYREF
  int v6; // [rsp+68h] [rbp-8h]
  int v7; // [rsp+6Ch] [rbp-4h]

  v7 = 100;
  puts("Generate a random password!");
  sleep(3u);
  v3 = time(0LL);
  srand(v3);
  v6 = rand() % 100;
  printf("Guess the password.. ");


  printf("\nInput : ");
  __isoc99_scanf("%s", s1);
  if ( !strcmp(s1, (&src)[v6]) )
  {
    puts("Yep.. That's perfect!!");
    sub_21B9();
  }
  else
  {
    puts("Nope.. I don't feel its good enough..");
    printf("[%s] is better\n", (&src)[v6]);
  }
  return 0LL;
}
```

Diketahui program melakukan pengecekan dari masukan dalam s1 dengan variabel v6 yang isinya merupakan array acak. Jika kondisi terpenuhi akan memanggil fungsi *sub_21b9().* Daripada capek menebak masukannya, mending cek fungsi yang dipanggil tersebut.

```
int sub_21B9()
```

```
{
  char dest[2]; // [rsp+0h] [rbp-20h] BYREF
  char v2; // [rsp+2h] [rbp-1Eh] BYREF
  char v3; // [rsp+3h] [rbp-1Dh] BYREF
  char v4[2]; // [rsp+4h] [rbp-1Ch] BYREF
  char v5; // [rsp+6h] [rbp-1Ah] BYREF
  char v6; // [rsp+7h] [rbp-19h] BYREF
  char v7[2]; // [rsp+8h] [rbp-18h] BYREF
  char v8[3]; // [rsp+Ah] [rbp-16h] BYREF
  char v9[19]; // [rsp+Dh] [rbp-13h] BYREF

  strcpy(dest, src);
  printf("%.1s%.1s%.1s{", dest, &v6, v8);
  strcpy(dest, off_5110);
  printf("%.3s", dest);
  strcpy(dest, off_5390);
  printf("%.2s", &v3);
  strcpy(dest, off_50E0);
  printf("%.1s_%.2s", &v6, v7);
  strcpy(dest, off_5308);
  printf("%.1s%.1s", v9, v9);
  strcpy(dest, off_52C0);
  printf("%.4s_", v4);
  strcpy(dest, off_5338);
  printf("%.9s_", &v5);
  strcpy(dest, off_51E8);
  return printf("%.2s0931%.2s}", dest, &v2);
}
```

Fungsi tersebut melakukan copy string dan print selama beberapa kali dari variabel src yang dicopy ke variabel dest untuk diprint. Dan tedapat beberapa keterangan assembly. Ada juga variabel off yang sepertinya memiliki isi karena dicopy ke variabel dest. Kita coba cek isi variabel off tersebut.

```
.data:0000000000005068 off_5068        dq offset off_5068      ; DATA XREF: sub_2170+1B↑r
.data:0000000000005068                                         ; .data:off_5068↓o
.data:0000000000005070                 align 20h
.data:0000000000005080 ; char *src
.data:0000000000005080 src             dq offset aC1ph3rabcxyz ; DATA XREF: sub_21B9+8↑r
.data:0000000000005080                                         ; main+95↑o ...
.data:0000000000005080                                         ; "C1PH3RABCXYZ!"
.data:0000000000005088                 dq offset aRand0mstr1ng ; "Rand0mStr1ng$"
.data:0000000000005090                 dq offset aPSsw0rdhCker ; "P@ssw0rdH@cker"
.data:0000000000005098                 dq offset aReadable12345 ; "Readable12345#"
.data:00000000000050A0                 dq offset aPassphrase456 ; "Passphrase$456"
.data:00000000000050A8                 dq offset aStr0ngpSsWord ; "Str0ngP@ss!word"
.data:00000000000050B0                 dq offset aSecretcode987 ; "SecretCode!987"
.data:00000000000050B8                 dq offset a1234secur3code ; "1234Secur3Code"
.data:00000000000050C0                 dq offset aR3adabl3pSsw0r ; "R3adabl3P@ssw0rd"
.data:00000000000050C8                 dq offset aPassgenius789 ; "PassGenius789!"
.data:00000000000050D0                 dq offset aCipher12345  ; "Cipher12345$"
.data:00000000000050D8                 dq offset aAuth3nticTeus ; "Auth3ntic@teUs"
.data:00000000000050E0 ; char *off_50E0
.data:00000000000050E0 off_50E0        dq offset aCr3Tiv3passw0r
.data:00000000000050E0                                         ; DATA XREF: sub_21B9+B2↑r
.data:00000000000050E0                                         ; "Cr3@tiv3Passw0rd"
.data:00000000000050E8                 dq offset a5678secretcode ; "5678SecretCode"
.data:00000000000050F0                 dq offset aPSswordmSter ; "P@sswordM@ster!"
.data:00000000000050F8                 dq offset aHCkm3ifyoucan ; "H@ckM3ifYouCan"
.data:0000000000005100                 dq offset a9876c0mpl3xpSs ; "9876C0mpl3xP@ss"
.data:0000000000005108                 dq offset aRandomize321 ; "Randomize$321"
.data:0000000000005110 ; char *off_5110
.data:0000000000005110 off_5110        dq offset aS3cur1tyl3vel5
.data:0000000000005110                                         ; DATA XREF: sub_21B9+4C↑r
```

```
.data:0000000000005110                                          ; "S3cur1tyL3vel5"
.data:0000000000005118                  dq offset aExpSsw0rd12  ; "ExP@ssw0rd12$"
.data:0000000000005120                  dq offset aStr0ngauth123 ; "Str0ngAuth123!"
.data:0000000000005128                  dq offset aPassw0rdgal0r3 ; "Passw0rdGal0r3"
.data:0000000000005130                  dq offset aUnlockc0d3567 ; "UnlockC0d3$567"
.data:0000000000005138                  dq offset a12345pSsc0de ; "12345P@ssC0de!"
.data:0000000000005140                  dq offset aReadWritable12 ; "Read&Writable12"
.data:0000000000005148                  dq offset aPasscr3Tor789 ; "PassCr3@tor789"
.data:0000000000005150                  dq offset a0p3nsesame456 ; "0p3nSesame!456"
.data:0000000000005158                  dq offset aHCkproof12345 ; "H@ckProof12345"
.data:0000000000005160                  dq offset aC0mpl3xpSswrd ; "C0mpl3xP@sswrd!"
.data:0000000000005168                  dq offset aSecurel0gin567 ; "SecureL0gin567$"
.data:0000000000005170                  dq offset aAccessgT3c0de ; "AccessG@t3C0de"
.data:0000000000005178                  dq offset a123passw0rd45 ; "123Passw0rd!45"
.data:0000000000005180                  dq offset aR3s3tmSt3rcode ; "R3s3tM@st3rCode"
.data:0000000000005188                  dq offset aAuth0r1z3me789 ; "Auth0r1z3Me$789"
.data:0000000000005190                  dq offset aR3adabl3cipher ; "R3adabl3Cipher!"
.data:0000000000005198                  dq offset aPSskey12345  ; "P@sskey12345$"
.data:00000000000051A0                  dq offset aSF3guardc0d3s ; "S@f3GuardC0d3s"
.data:00000000000051A8                  dq offset aCr3T3pass123 ; "Cr3@t3Pass123!"
.data:00000000000051B0                  dq offset aVaults3curity ; "VaultS3curity$"
.data:00000000000051B8                  dq offset aHCkm3now567  ; "H@ckM3Now567!"
.data:00000000000051C0                  dq offset aAuth3nticTe12 ; "Auth3ntic@te12$"
.data:00000000000051C8                  dq offset a1234pSsgal0r3 ; "1234P@ssGal0r3"
.data:00000000000051D0                  dq offset aMTrixsecur1ty ; "M@trixSecur1ty!"
.data:00000000000051D8                  dq offset aOpend00rc0de ; "OpenD00rC0de$"
.data:00000000000051E0                  dq offset aPSsw0rdw1zard ; "P@ssw0rdW1zard"
.data:00000000000051E8 ; char *off_51E8
.data:00000000000051E8 off_51E8         dq offset a5678acc3ssgT3
.data:00000000000051E8                                          ; DATA XREF: sub_21B9+196↑r
.data:00000000000051E8                                          ; "5678Acc3ssG@t3"
.data:00000000000051F0                  dq offset aC0d3mSt3r123 ; "C0d3M@st3r123!"
.data:00000000000051F8                  dq offset aR3s1l13ntpSs ; "R3s1l13ntP@ss$"
.data:0000000000005200                  dq offset aG3n3rT3c0mpl3x ; "G3n3r@t3C0mpl3x"
.data:0000000000005208                  dq offset aAuth0riz3me567 ; "Auth0riz3Me567!"
.data:0000000000005210                  dq offset aPSscr3Tor12  ; "P@ssCr3@tor12$"
.data:0000000000005218                  dq offset aB3secure12345 ; "B3Secure12345!"
.data:0000000000005220                  dq offset aLockKeymSter ; "Lock&KeyM@ster$"
.data:0000000000005228                  dq offset a1234r3s3tc0de ; "1234R3s3tC0de"
.data:0000000000005230                  dq offset aHackm3not567 ; "HackM3Not567!"
.data:0000000000005238                  dq offset aAuth3nticT3now ; "Auth3ntic@t3Now"
.data:0000000000005240                  dq offset aAccessgrNt12 ; "AccessGr@nt12$"
.data:0000000000005248                  dq offset aS3cur1tyvUlt12 ; "S3cur1tyV@ult123"
.data:0000000000005250                  dq offset aPSsw0rdpr0tect ; "P@ssw0rdPr0tect"
.data:0000000000005258                  dq offset a5678pSsc0mb0 ; "5678P@ssC0mb0$"
.data:0000000000005260                  dq offset aRandom1z3c0de ; "Random1z3C0de"
.data:0000000000005268                  dq offset aC0mpl3xauth123 ; "C0mpl3xAuth123!"
.data:0000000000005270                  dq offset aAuth0r1zem3now ; "Auth0r1zeM3Now"
.data:0000000000005278                  dq offset aR3adabl3key567 ; "R3adabl3Key$567"
.data:0000000000005280                  dq offset aPassw0rdl0ck ; "Passw0rdL0ck!"
.data:0000000000005288                  dq offset aS3cur3mTrix12 ; "S3cur3M@trix12$"
.data:0000000000005290                  dq offset aHack3rpr00f123 ; "Hack3rPr00f123"
.data:0000000000005298                  dq offset aCipheredpSswrd ; "CipheredP@sswrd"
.data:00000000000052A0                  dq offset a5678unlockc0de ; "5678UnlockC0de$"
.data:00000000000052A8                  dq offset aPassphrSemSt3r ; "Passphr@seM@st3r"
.data:00000000000052B0                  dq offset aAuth3nticT3me ; "Auth3ntic@t3Me!"
.data:00000000000052B8                  dq offset aR3v3rs3pass123 ; "R3v3rs3Pass123$"
.data:00000000000052C0 ; char *off_52C0
.data:00000000000052C0 off_52C0         dq offset aPSsw0rdf0rtify
.data:00000000000052C0                                          ; DATA XREF: sub_21B9+12C↑r
.data:00000000000052C0                                          ; "P@ssw0rdF0rtify"
.data:00000000000052C8                  dq offset aPr0t3ctme567 ; "Pr0t3ctMe567!"
.data:00000000000052D0                  dq offset aS3cur1tylYers1 ; "S3cur1tyL@yers12"
.data:00000000000052D8                  dq offset aC0mpl3xity123 ; "C0mpl3xity123!"
.data:00000000000052E0                  dq offset aAuth0r1zedacc3 ; "Auth0r1zedAcc3ss"
.data:00000000000052E8                  dq offset aPSsmTr1x567  ; "P@ssM@tr1x567$"
```

```
.data:00000000000052F0                 dq offset aR3adWritec0de ; "R3ad&WriteC0de"
.data:00000000000052F8                 dq offset aHCkm3n0w12    ; "H@ckM3N0w12$"
.data:0000000000005300                 dq offset aSecurepSspr0t3 ; "SecureP@ssPr0t3ct"
.data:0000000000005308 ; char *off_5308
.data:0000000000005308 off_5308        dq offset a5678guardc0d3
.data:0000000000005308                                          ; DATA XREF: sub_21B9+EF↑r
.data:0000000000005308                                          ; "5678GuardC0d3$"
.data:0000000000005310                 dq offset aAuth3nticT3key ; "Auth3ntic@t3Key"
.data:0000000000005318                 dq offset aPSsw0rdsF3ty ; "P@ssw0rdS@f3ty"
.data:0000000000005320                 dq offset aR3s1stm3now567 ; "R3s1stM3Now567"
.data:0000000000005328                 dq offset aC0mpl3xlock12 ; "C0mpl3xLock12$"
.data:0000000000005330                 dq offset aHackpr00f12345 ; "HackPr00f12345!"
.data:0000000000005338 ; char *off_5338
.data:0000000000005338 off_5338        dq offset aSecureg3n3rat0
.data:0000000000005338                                          ; DATA XREF: sub_21B9+161↑r
.data:0000000000005338                                          ; "SecureG3n3raT0r"
.data:0000000000005340                 dq offset a5678passsh13ld ; "5678PassSh13ld"
.data:0000000000005348                 dq offset aAuth3nticTionm ; "Auth3ntic@tionMe"
.data:0000000000005350                 dq offset aPSsw0rdwLl567 ; "P@ssw0rdW@ll567"
.data:0000000000005358                 dq offset aR3adabl3f0rtif ; "R3adabl3F0rtify"
.data:0000000000005360                 dq offset aStr0ngmTrix12 ; "Str0ngM@trix12$"
.data:0000000000005368                 dq offset aHCk3rblock123 ; "H@ck3rBlock123"
.data:0000000000005370                 dq offset aPr0t3ctm3now56 ; "Pr0t3ctM3Now567"
.data:0000000000005378                 dq offset aAuth0riz3m3key ; "Auth0riz3M3Key"
.data:0000000000005380                 dq offset aR3s1l13ntsF3ty ; "R3s1l13ntS@f3ty"
.data:0000000000005388                 dq offset aPassw0rdstr3ng ; "Passw0rdStr3ngth"
.data:0000000000005390 ; char *off_5390
.data:0000000000005390 off_5390        dq offset aS3cur1tysh13ld
.data:0000000000005390                                          ; DATA XREF: sub_21B9+7D↑r
.data:0000000000005390                                          ; "S3cur1tySh13ld12"
.data:0000000000005398                 dq offset aHackm3n0w567 ; "HackM3N0w567$"
.data:0000000000005398 _data           ends
.data:0000000000005398
.bss:00000000000053A0 ;
===========================================================================
```

Dari hasil tersebut bisa kita cek variabel yang mana saja yang dipanggil dan diperoleh:

```
    const char off_5110[] = "C1PH3RABCXYZ!";
    const char off_5390[] = "S3cur1tyL3vel5";
    const char off_50E0[] = "S3cur1tySh13ld12";
    const char off_5308[] = "Cr3@tiv3Passw0rd";
    const char off_52C0[] = "5678GuardC0d3$";
    const char off_5338[] = "P@ssw0rdF0rtify";
    const char off_51E8[] = "SecureG3n3raT0r";
    const char off_5368[] = "5678Acc3ssG@t3";
```

(variabel diatas sudah disesuaikan dengan Bahasa C untuk dimasukkan ke dalam solver)

Bisa disimpulkan bahwa program melakukan print berdasarkan variabel off dengan beberapa keterangan pada fungsi print yaitu

%.1s , %.2s, %.3s dan seterusnya yang berarti melakukan print pada 1 digit pertama, 2 digit pertama dan seterusnya.

Keterangan assembler pada deklarasi variabel berarti memanggil variabel dest yang dideklarasi di awal ditambahkan dengan asm add 0x2h (+2), 0x3h (+3), dan seterusnya.

Maka yang diperlkan untuk membuat solver adalah membuat program yang meniru fitur print tersebut.

solv.c

```
#include <stdio.h>
```

```c
#include <string.h>

void sub_21B9() {
  char dest[100];  // Adjust the size as needed
  char v2; // [rsp+2h] [rbp-1Eh] BYREF
  char v3; // [rsp+3h] [rbp-1Dh] BYREF
  char v4[2]; // [rsp+4h] [rbp-1Ch] BYREF
  char v5; // [rsp+6h] [rbp-1Ah] BYREF
  char v6; // [rsp+7h] [rbp-19h] BYREF
  char v7[2]; // [rsp+8h] [rbp-18h] BYREF
  char v8[3]; // [rsp+Ah] [rbp-16h] BYREF
  char v9[19]; // [rsp+Dh] [rbp-13h] BYREF

    const char off_5110[] = "C1PH3RABCXYZ!";
    const char off_5390[] = "S3cur1tyL3vel5";
    const char off_50E0[] = "S3cur1tySh13ld12";
    const char off_5308[] = "Cr3@tiv3Passw0rd";
    const char off_52C0[] = "5678GuardC0d3$";
    const char off_5338[] = "P@ssw0rdF0rtify";
    const char off_51E8[] = "SecureG3n3raT0r";
    const char off_5368[] = "5678Acc3ssG@t3";

    strcpy(dest, off_5110);
    printf("%.1s%.1s%.1s{", dest, dest + 7, dest + 10);

    strcpy(dest, off_5390);
    printf("%.3s", dest);

    strcpy(dest, off_50E0);
    printf("%.2s", dest + 3);

    strcpy(dest, off_5308);
    printf("%.1s_%.2s", dest + 7, dest + 8);

    strcpy(dest, off_52C0);
    printf("%.1s%.1s", dest + 13, dest + 13);

    strcpy(dest, off_5338);
    printf("%.4s_", dest + 4);

    strcpy(dest, off_51E8);
    printf("%.9s_", dest + 6);

    strcpy(dest, off_5368);
    printf("%.2s0931%.2s}", dest, dest + 2);
}

int main() {
    sub_21B9();
    return 0;
}
```

Flag : CBY{S3cur3_Pa$$w0rd_G3n3raT0r_56093178}

Baby Snake

Diperoleh file Baby_Snake.pyc yang merupakan python compiled. Lakukan decompile file tersebut dengan uncompyle6 atau saya pake situs cina yang nemu di google gara2 uncompyle6 saya eror

Tautan situs cina: https://tool.lu/en_US/pyc/

Diperoleh:

```
#!/usr/bin/env python
# visit https://tool.lu/pyc/ for more information
# Version: Python 3.9

def b(i):
    r = 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/'
    a = []
    n = 0
    d = 0
    y = 0
    for p in i:
        y = y << 8 | p
        d += 8
        if d >= 6:
            l = y >> (d - 6) & 63
            a.append(r[l])
            d -= 6
    return ''.join(a)

def z(x):
    q = []
    for i in range(len(x)):
        o = ord(x[i]) ^ i
        q.append(o)
    return q

if __name__ == '__main__':
    hexa = [
        81, 49, 72, 89, 97, 52, 101, 112, 89, 94, 98, 109, 90, 74, 105, 119,
        69, 32, 43, 80, 90, 82, 92, 77, 64, 41, 46, 108, 74, 91, 39, 84, 68,
        117, 116, 118, 124, 21, 96, 65, 124, 67, 104, 82, 120, 121, 124, 92,
        104, 3, 120, 113, 101, 91, 90, 81, 109, 11, 14, 11, 111, 71, 112, 6]

    s = input('>>')
    m = s.encode()
    t = b(m)
    u = z(t)

    if ''.join(chr(v) for v in u) == ''.join(chr(v) for v in hexa):
        print('Correct!')
    else:
        print('Wrong')
```

pada fungsi main program, dilakukan beberapa tindakan

deklarasi array hexa

deklarasi variabel s yang merupakan pembacaan input

variabel m melakukan encoding pada s

t melakukan fungsi b yang diberikan nilai m, dan

variabel u melakukan fungsi z yang diberikan nilai dari fungsi t.

parameter pengecekan dilakukan dengan cara membandingkan nilai u yang dikonversi ke string dan digabungkan lalu dibandingkan dengan array hexa yang diubah menjadi string. Program ini memeriksa apakah 2 string u dan hexa tadi adalah sama.

Hasil solver.py

```
def reverse_b(encoded_str):
    r = 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/'
    decoded_bytes = []
    d = 0
    y = 0

    for c in encoded_str:
        if c == '=':
            continue

        l = r.index(c)
        y = y << 6 | l
        d += 6

        while d >= 8:
            byte = (y >> (d - 8)) & 255
            decoded_bytes.append(byte)
            d -= 8

    return bytes(decoded_bytes)

def reverse_z(encoded_list):
    original_str = ''
    for i, o in enumerate(encoded_list):
        original_str += chr(o ^ i)
    return original_str

if __name__ == '__main__':
    hexa = [
        81, 49, 72, 89, 97, 52, 101, 112, 89, 94, 98, 109, 90, 74, 105, 119,
        69, 32, 43, 80, 90, 82, 92, 77, 64, 41, 46, 108, 74, 91, 39, 84, 68,
        117, 116, 118, 124, 21, 96, 65, 124, 67, 104, 82, 120, 121, 124, 92,
        104, 3, 120, 113, 101, 91, 90, 81, 109, 11, 14, 11, 111, 71, 112, 6
    ]

    encoded_str = ''.join([chr(c) for c in hexa])
    reversed_z = reverse_z(hexa)
    reversed_b = reverse_b(reversed_z)

    print("flag:", reversed_b.decode())


flag: CBY{W0Ah_Th1S_B4bY_N0T_Ju5T_A_N0rM4l_bABy_Sn4K3}
```

Flag Checker V1

```
Enter the flag: 123
Nope!
```

Hasil decompile:

```
int __cdecl main(int argc, const char **argv, const char **envp)
{
  char Str[56]; // [rsp+20h] [rbp-40h] BYREF
  int v5; // [rsp+58h] [rbp-8h]
  int i; // [rsp+5Ch] [rbp-4h]

  _main();
  printf_0("Enter the flag: ");
  scanf("%s", Str);
  v5 = strlen(Str);
  if ( v5 == 43 )
  {
    for ( i = 0; i < v5; ++i )
    {
      if ( Str[i] != (s0me[i] ^ th1ng[i]) )
        goto LABEL_2;
    }
    printf_0("Yep!\n");
    getchar();
    getchar();
    return 0;
  }
  else
  {
LABEL_2:
    printf_0("Nope!\n");
    getchar();
    getchar();
    return 0;
  }
}
```

Program meminta masukan input dimasukkan ke dalam variabel Str dan melakukan beberapa fungsi pengecekan menentukan apakah input dari Str memiliki Panjang 43. Jika mengembalikan nilai true maka akan dicek kembali, apakah nilai Str[i] dalam looping memiliki nilai yang sama dengan hasil xor dari array s0me[i] dengan th1ng[i].

Untuk membuat solver kita hanya memerlukan nilai array s0me[i] dan th1ng[i]

Dari decompiler diperoleh nilai hex s0me dan th1ng

```
.data:0000000140019020                 public s0me
.data:0000000140019020 ; _DWORD s0me[48]
.data:0000000140019020 s0me            dd 0B3h, 2Ah, 7Ch, 5Dh, 0DDh, 5, 0C6h, 0F0h, 21h, 48h
.data:0000000140019020                                          ; DATA XREF: main+85↑o
.data:0000000140019020                 dd 94h, 11h, 0D0h, 67h, 3Fh, 85h, 0D7h, 1Dh, 3Ah, 0C0h
.data:0000000140019020                 dd 5Bh, 9Bh, 46h, 0D9h, 58h, 0F6h, 6Eh, 8, 0BAh, 0E7h
.data:0000000140019020                 dd 27h, 4Ch, 0A2h, 37h, 81h, 0E3h, 15h, 51h, 8Ch, 34h
.data:0000000140019020                 dd 0CCh, 93h, 42h, 5 dup(0)
.data:00000001400190E0                 public th1ng
.data:00000001400190E0 ; _DWORD th1ng[48]
.data:00000001400190E0 th1ng           dd 0F0h, 68h, 25h, 26h, 85h, 35h, 89h, 82h, 7Eh, 10h, 0A4h
.data:00000001400190E0                                          ; DATA XREF: main+9E↑o
```

```
.data:00000001400190E0                 dd 7Eh, 0BFh, 28h, 0Fh, 0D7h, 88h, 65h, 0Ah, 0B2h, 4, 0AFh
.data:00000001400190E0                 dd 28h, 0BDh, 7, 8Eh, 5Eh, 67h, 0D5h, 0A8h, 48h, 1Eh, 0D0h
.data:00000001400190E0                 dd 68h, 0B5h, 0A4h, 54h, 60h, 0E2h, 6Bh, 0F6h, 0D7h, 3Fh
.data:00000001400190E0                 dd 5 dup(0)
```

Buatlah program yang melakukan xor 2 variabel tersebut dan diperoleh solver.py

```
def xor_arrays(array1, array2):
    result = [a ^ b for a, b in zip(array1, array2)]
    return result

def print_ascii(array):
    ascii_string = ''.join(chr(value) for value in array)
    print(ascii_string)

s0me = [0xB3, 0x2A, 0x7C, 0x5D, 0xDD, 0x05, 0xC6, 0xF0, 0x21, 0x48, 0x94, 0x11,
0xD0, 0x67, 0x3F, 0x85,
        0xD7, 0x1D, 0x3A, 0xC0, 0x5B, 0x9B, 0x46, 0xD9, 0x58, 0xF6, 0x6E, 0x08,
0xBA, 0xE7, 0x27, 0x4C,
        0xA2, 0x37, 0x81, 0xE3, 0x15, 0x51, 0x8C, 0x34, 0xCC, 0x93, 0x42] + [0] *
5

th1ng = [0xF0, 0x68, 0x25, 0x26, 0x85, 0x35, 0x89, 0x82, 0x7E, 0x10, 0xA4, 0x7E,
0xBF, 0x28, 0x0F, 0xD7,
         0x88, 0x65, 0x0A, 0xB2, 0x04, 0xAF, 0x28, 0xBD, 0x07, 0x8E, 0x5E, 0x67,
0xD5, 0xA8, 0x48, 0x1E,
         0xD0, 0x68, 0xB5, 0xA4, 0x54, 0x60, 0xE2, 0x6B, 0xF6, 0xD7, 0x3F]

result_array = xor_arrays(s0me, th1ng)

print("flag:")
print_ascii(result_array)

hasil run:
flag:
CBY{X0Or_X0ooO0R_x0r_4nd_x0ooOoRr_4GA1n_:D}
```

Flag Checker V2

```
Enter the flag: tesst
```

Lakukan decompile

```
int __cdecl main(int argc, const char **argv, const char **envp)
{
  char Str[60]; // [rsp+20h] [rbp-40h] BYREF
  int v5; // [rsp+5Ch] [rbp-4h]

  _main();
  printf_0("Enter the flag: ");
  scanf("%s", Str);
  v5 = strlen(Str);
```

```
  if ( v5 == 40 )
  {
    if ( (unsigned int)check(Str) )
      printf_0("Yep!\n");
    else
      printf_0("Nope!\n");
    getchar();
    getchar();
    return 0;
  }
  else
  {
    printf_0("Nope!\n");
    getchar();
    getchar();
    return 0;
  }
}
```

Program melakukan pengecekan inputan Str yang memiliki Panjang 40 dengan fungsi “check”. Mari kita liat isi fungsi “check”.

```
_BOOL8 __fastcall check(char *a1)
{
  return a1[7] + a1[3] * a1[17] - a1[2] + a1[25] - a1[11] * a1[6] - a1[35] == 5913
      && a1[7] * a1[20] == 10450
      && a1[16] + a1[10] * a1[29] * a1[4] - a1[28] - a1[36] - a1[13] - a1[27] == 757856
      && a1[24] * a1[9] == 5035
      && a1[26] + a1[14] - a1[1] * a1[22] - a1[32] - a1[33] + *a1 * a1[9] == 390
      && *a1 * a1[23] == 7638
      && a1[21] + a1[12] + a1[31] * a1[15] + a1[19] - a1[24] * a1[38] + a1[30] == -3673
      && a1[35] * a1[38] == 6460
      && a1[18] + a1[20] + a1[5] - a1[37] - a1[34] + a1[23] * a1[8] * a1[39] == 1524896
      && a1[36] * a1[16] == 6264
      && a1[16] + a1[3] * a1[28] - a1[2] + a1[9] - a1[7] * a1[14] - a1[21] == -2562
      && a1[15] * a1[31] == 2448
      && a1[20] + a1[39] * a1[8] * a1[35] - a1[12] - a1[30] - a1[27] - a1[5] == 1270376
      && a1[6] * a1[29] == 4940
      && a1[34] + a1[36] - a1[25] * a1[22] - a1[19] - *a1 + a1[1] == -4296
      && a1[22] * a1[30] == 4992
      && a1[33] + a1[31] + a1[26] * a1[6] + a1[11] - a1[23] * a1[15] + a1[10] == -2660
      && a1[5] * a1[3] == 8856
      && a1[13] + a1[32] + a1[37] - a1[17] - a1[24] + a1[4] * a1[2] * a1[38] * a1[29] == 48294989
      && a1[13] * a1[1] == 6270
      && a1[21] + a1[36] * a1[19] - a1[11] + a1[10] - a1[5] * a1[24] - a1[34] == -186
      && a1[33] * a1[2] == 7120
      && a1[14] + a1[4] * a1[8] * a1[32] - a1[7] - a1[31] - a1[28] - a1[30] == 682856
      && a1[37] * a1[14] == 4485
      && a1[33] + a1[17] - a1[20] * a1[23] - a1[1] - a1[16] + a1[3] * a1[27] == 3553
      && a1[4] * a1[11] == 7560
      && *a1 + a1[25] + a1[9] * a1[12] + a1[35] - a1[26] * a1[22] + a1[2] == 739
      && a1[10] * a1[21] == 6650
      && a1[29] + a1[37] + a1[15] - a1[6] - a1[18] + a1[39] * a1[13] * a1[38] == 807579
      && a1[32] * a1[12] == 3876
      && a1[37] + a1[21] * a1[3] - a1[1] + a1[30] - a1[18] * a1[9] - a1[24] == 5889
      && a1[18] * a1[26] == 2448
      && a1[27] + a1[26] * a1[12] * a1[38] - a1[20] - a1[17] - a1[16] - a1[19] == 166178
      && a1[25] * a1[19] == 10146
      && a1[5] + a1[35] - a1[32] * a1[15] - a1[6] - a1[13] + a1[29] * a1[4] == 4352
      && a1[28] * a1[34] == 2706
      && a1[25] + a1[31] + a1[22] * a1[8] + a1[10] - a1[33] * *a1 + a1[11] == 101
      && a1[27] * a1[8] == 12519
      && a1[14] + a1[28] + a1[36] - a1[7] - a1[2] + a1[34] * a1[23] * a1[39] == 470306
      && a1[17] * a1[39] == 10750
      && a1[19] + a1[25] * a1[37] - a1[10] + a1[21] - a1[4] * a1[33] - a1[6] == -3212
      && a1[24] + a1[35] * a1[11] * a1[26] - a1[8] - a1[15] - a1[28] - a1[38] == 410190
      && a1[12] + a1[14] - a1[18] * a1[3] - a1[7] - a1[2] + a1[20] * a1[9] == -1271
      && a1[22] + a1[30] + *a1 * a1[31] + a1[36] - a1[17] * a1[29] + a1[23] == -4429
      && a1[32] + a1[16] + a1[27] - a1[1] - a1[5] + a1[39] * a1[13] * a1[34] == 392038;
}
```

Njir, inimah ga mungkin dihitung sendiri kan. Ternyata ada library python yang sangat powerful yaitu z3. Buat solver dengan z3 untuk melakukan pengecekan terhadap nilai-nilai array a1 yang ada di persamaan matematis pada fungsi “check”.

Solver tersebut melakukan bruteforce terhadap semua nilai ASCII (1-256) dan dicoba ke nilai array a1 yang ada pada fungsi “check”. operasi yang cukup berat jika tidak menggunakan bantuan library z3

Solver.py

```
from z3 import *

def generate_flags(flag_length, possible_values, current_flag=None, index=0):
    if current_flag is None:
        current_flag = [None] * flag_length

    if index == flag_length:
        yield tuple(current_flag)
    else:
        for value in possible_values:
            current_flag[index] = value
            yield from generate_flags(flag_length, possible_values, current_flag,
index + 1)

def check_flag(a1):
    flag_vars = [BitVec(f'flag_{i}', 8) for i in range(len(a1))]

    constraints = [
        flag_vars[7] + flag_vars[3] * flag_vars[17] - flag_vars[2] + flag_vars[25]
- flag_vars[11] * flag_vars[6] - flag_vars[35] == 5913,
        flag_vars[7] * flag_vars[20] == 10450,
        flag_vars[16] + flag_vars[10] * flag_vars[29] * flag_vars[4] -
flag_vars[28] - flag_vars[36] - flag_vars[13] - flag_vars[27] == 757856,
        flag_vars[24] * flag_vars[9] == 5035,
        flag_vars[26] + flag_vars[14] - flag_vars[1] * flag_vars[22] -
flag_vars[32] - flag_vars[33] + flag_vars[0] * flag_vars[9] == 390,
        flag_vars[0] * flag_vars[23] == 7638,
        flag_vars[21] + flag_vars[12] + flag_vars[31] * flag_vars[15] +
flag_vars[19] - flag_vars[24] * flag_vars[38] + flag_vars[30] == -3673,
        flag_vars[35] * flag_vars[38] == 6460,
        flag_vars[18] + flag_vars[20] + flag_vars[5] - flag_vars[37] -
flag_vars[34] + flag_vars[23] * flag_vars[8] * flag_vars[39] == 1524896,
        flag_vars[36] * flag_vars[16] == 6264,
        flag_vars[16] + flag_vars[3] * flag_vars[28] - flag_vars[2] + flag_vars[9]
- flag_vars[7] * flag_vars[14] - flag_vars[21] == -2562,
        flag_vars[15] * flag_vars[31] == 2448,
        flag_vars[20] + flag_vars[39] * flag_vars[8] * flag_vars[35] -
flag_vars[12] - flag_vars[30] - flag_vars[27] - flag_vars[5] == 1270376,
        flag_vars[6] * flag_vars[29] == 4940,
        flag_vars[34] + flag_vars[36] - flag_vars[25] * flag_vars[22] -
flag_vars[19] - flag_vars[0] + flag_vars[1] == -4296,
        flag_vars[22] * flag_vars[30] == 4992,
        flag_vars[33] + flag_vars[31] + flag_vars[26] * flag_vars[6] +
flag_vars[11] - flag_vars[23] * flag_vars[15] + flag_vars[10] == -2660,
        flag_vars[5] * flag_vars[3] == 8856,
        flag_vars[13] + flag_vars[32] + flag_vars[37] - flag_vars[17] -
flag_vars[24] + flag_vars[4] * flag_vars[2] * flag_vars[38] * flag_vars[29] ==
48294989,
        flag_vars[13] * flag_vars[1] == 6270,
        flag_vars[21] + flag_vars[36] * flag_vars[19] - flag_vars[11] +
flag_vars[10] - flag_vars[5] * flag_vars[24] - flag_vars[34] == -186,
        flag_vars[33] * flag_vars[2] == 7120,
        flag_vars[14] + flag_vars[4] * flag_vars[8] * flag_vars[32] - flag_vars[7]
- flag_vars[31] - flag_vars[28] - flag_vars[30] == 682856,
```

```
        flag_vars[37] * flag_vars[14] == 4485,
        flag_vars[33] + flag_vars[17] - flag_vars[20] * flag_vars[23] -
flag_vars[1] - flag_vars[16] + flag_vars[3] * flag_vars[27] == 3553,
        flag_vars[4] * flag_vars[11] == 7560,
        flag_vars[0] + flag_vars[25] + flag_vars[9] * flag_vars[12] +
flag_vars[35] - flag_vars[26] * flag_vars[22] + flag_vars[2] == 739,
        flag_vars[10] * flag_vars[21] == 6650,
        flag_vars[29] + flag_vars[37] + flag_vars[15] - flag_vars[6] -
flag_vars[18] + flag_vars[39] * flag_vars[13] * flag_vars[38] == 807579,
        flag_vars[32] * flag_vars[12] == 3876,
        flag_vars[37] + flag_vars[21] * flag_vars[3] - flag_vars[1] +
flag_vars[30] - flag_vars[18] * flag_vars[9] - flag_vars[24] == 5889,
        flag_vars[18] * flag_vars[26] == 2448,
        flag_vars[27] + flag_vars[26] * flag_vars[12] * flag_vars[38] -
flag_vars[20] - flag_vars[17] - flag_vars[16] - flag_vars[19] == 166178,
        flag_vars[25] * flag_vars[19] == 10146,
        flag_vars[5] + flag_vars[35] - flag_vars[32] * flag_vars[15] -
flag_vars[6] - flag_vars[13] + flag_vars[29] * flag_vars[4] == 4352,
        flag_vars[28] * flag_vars[34] == 2706,
        flag_vars[25] + flag_vars[31] + flag_vars[22] * flag_vars[8] +
flag_vars[10] - flag_vars[33] * flag_vars[0] + flag_vars[11] == 101,
        flag_vars[27] * flag_vars[8] == 12519,
        flag_vars[14] + flag_vars[28] + flag_vars[36] - flag_vars[7] -
flag_vars[2] + flag_vars[34] * flag_vars[23] * flag_vars[39] == 470306,
        flag_vars[17] * flag_vars[39] == 10750,
        flag_vars[19] + flag_vars[25] * flag_vars[37] - flag_vars[10] +
flag_vars[21] - flag_vars[4] * flag_vars[33] - flag_vars[6] == -3212,
        flag_vars[24] + flag_vars[35] * flag_vars[11] * flag_vars[26] -
flag_vars[8] - flag_vars[15] - flag_vars[28] - flag_vars[38] == 410190,
        flag_vars[12] + flag_vars[14] - flag_vars[18] * flag_vars[3] -
flag_vars[7] - flag_vars[2] + flag_vars[20] * flag_vars[9] == -1271,
        flag_vars[22] + flag_vars[30] + flag_vars[0] * flag_vars[31] +
flag_vars[36] - flag_vars[17] * flag_vars[29] + flag_vars[23] == -4429,
        flag_vars[32] + flag_vars[16] + flag_vars[27] - flag_vars[1] -
flag_vars[5] + flag_vars[39] * flag_vars[13] * flag_vars[34] == 392038
    ]

    s = Solver()
    s.add(constraints)
    if s.check() == sat:
        model = s.model()
        result_flag = [model.eval(flag_vars[i]).as_long() for i in range(len(a1))]
        return result_flag
    else:
        return None

def find_valid_flag():
    flag_length = 40
    possible_values = range(256)

    for candidate_flag in generate_flags(flag_length, possible_values):
        result = check_flag(candidate_flag)
        if result:
            print("Valid flag found:", ''.join(map(chr, result)))
            return ''.join(map(chr, result))

    print("No valid flag found.")
    return None
```

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

hasil:

```
Valid flag found: CBY{TH4nk5_Z3_s0lV3r_F0r_Y0uR_h3LP!_:'D}

[Done] exited with code=0 in 1.878 seconds
```

Nomer 5

Diberikan assembly sebagai berikut, flag merupakan hasil dari operasi assembly tersebut

```
nomer 5 ():
        push    rbp
        mov     rbp, rsp
        mov     DWORD PTR [rbp-4], 20
        mov     DWORD PTR [rbp-8], 10
        mov     DWORD PTR [rbp-12], 20
        mov     eax, DWORD PTR [rbp-4]
        imul    eax, DWORD PTR [rbp-8]
        lea     ecx, [rax+2]
        mov     eax, DWORD PTR [rbp-12]
        mov     edx, eax
        sal     eax, 2
        sub     edx, eax
        lea     eax, [rcx+rdx]
        mov     DWORD PTR [rbp-16], eax
        sal     DWORD PTR [rbp-16], 20
        cmp     DWORD PTR [rbp-16], 100000000
        jg      .L2
        mov     eax, DWORD PTR [rbp-16]
        lea     edx, [rax+3]
        test    eax, eax
        cmovs   eax, edx
        sar     eax, 2
        mov     DWORD PTR [rbp-16], eax
        jmp     .L3
.L2:
        cmp     DWORD PTR [rbp-16], 100000000
        jle     .L4
        cmp     DWORD PTR [rbp-16], 500000000
        jg      .L4
        mov     eax, DWORD PTR [rbp-16]
        lea     edx, [rax+7]
        test    eax, eax
        cmovs   eax, edx
        sar     eax, 3
        mov     DWORD PTR [rbp-16], eax
        jmp     .L3
.L4:
        mov     eax, DWORD PTR [rbp-16]
        mov     edx, eax
        shr     edx, 31
        add     eax, edx
        sar     eax
        mov     DWORD PTR [rbp-16], eax
.L3:
        nop
        pop     rbp
        ret
```

assembly tersebut melakukan beberapa fungsi yaitu membuat beberapa variabel, sebut saja var1 = 20, var 2 = 10, var 3 = 20. Dan melakukan beberapa fungsi

hasil = var1 * var2

hasil = hasil + 2

temp = var3

temp = temp – (var3 * 4) dan seterusnya. Kita hanya perlu membuat program yang mirip dengan assembly tersebut untuk menemukan flagnya.

Solver.py

```
#include <stdio.h>

int nomer_5() {
    int result;
    int var1 = 20;
    int var2 = 10;
    int var3 = 20;

    result = var1 * var2;
    result = result + 2;

    int temp = var3;
    temp = temp - (var3 * 4);
    result = result + temp;

    result = result << 20;

    if (result > 100000000) {
        if (result <= 500000000) {
            result = (result + 7) >> 3;
        } else {
            result = (result >> 31) + result;
            result = result >> 1;
        }
    } else {
        result = (result + 3) >> 2;
    }

    return result;
}

int main() {
    int value = nomer_5();
    printf("Result: %d\n", value);
    return 0;
}


Result: 18612224
```

Flag: CBY{18612224}

Where is The Flag

Diberikan Cing-Ucing.apk, lakukan decompile dengan jadx

Diperoleh MainActivity

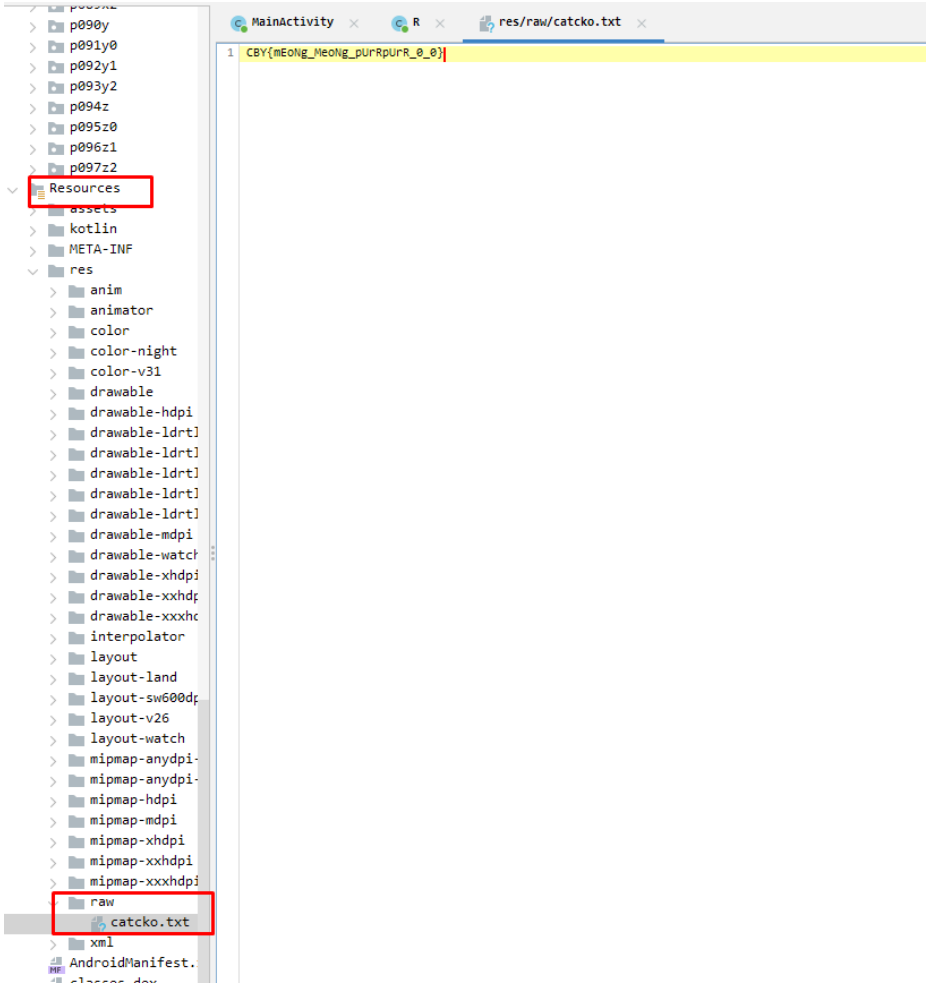
```java
package com.example.where_is_the_flag_0_0;

import android.os.Bundle;
import android.util.Log;
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import p012d.ActivityC0787d;

/* loaded from: classes.dex */
public class MainActivity extends ActivityC0787d {
    @Override // androidx.fragment.app.ActivityC0373q, androidx.activity.ComponentActivity,
p078v.ActivityC1392h, android.app.Activity
    public final void onCreate(Bundle bundle) {
        String str;
        super.onCreate(bundle);
        setContentView(R.layout.activity_main);
        try {
            InputStream openRawResource = getResources().openRawResource(R.raw.catcko);
            BufferedReader bufferedReader = new BufferedReader(new
InputStreamReader(openRawResource));
            StringBuilder sb = new StringBuilder();
            while (true) {
                String readLine = bufferedReader.readLine();
                if (readLine == null) {
                    break;
                }
                sb.append(readLine);
            }
            bufferedReader.close();
            openRawResource.close();
            str = sb.toString();
        } catch (IOException e) {
            e.printStackTrace();
            str = "Flag retrieval failed!";
        }
        StringBuilder sb2 = new StringBuilder();
        for (int i = 0; i < str.length(); i++) {
            char charAt = str.charAt(i);
            if ((charAt >= 'A' && charAt <= 'Z') || (charAt >= 'a' && charAt <= 'z')) {
                charAt = (char) (charAt + 1);
            }
            sb2.append(charAt);
        }
        String sb3 = sb2.toString();
        StringBuilder sb4 = new StringBuilder();
        for (int i2 = 0; i2 < sb3.length(); i2++) {
            char charAt2 = sb3.charAt(i2);
            if ((charAt2 >= 'A' && charAt2 <= 'Z') || (charAt2 >= 'a' && charAt2 <= 'z')) {
                charAt2 = (char) (charAt2 - 1);
            }
            sb4.append(charAt2);
        }
        Log.d("≥^•ω•^≤", sb4.toString());
    }
}
```

Sepertinya ada fungsi yang mencurigakan karena membuka sebuah Resource

```java
InputStream openRawResource = getResources().openRawResource(R.raw.catcko);
```



Flag: CBY{mEoNg_MeoNg_pUrRpUrR_0_0}