## HW-0x08 Writeup

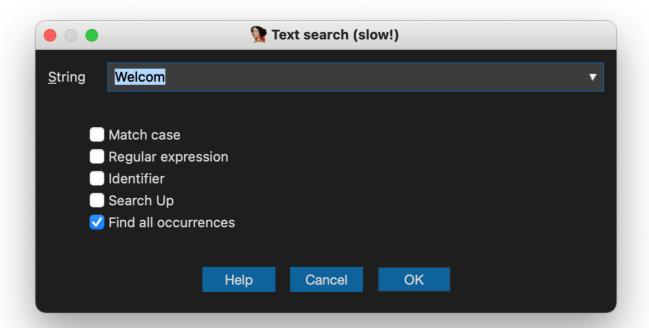
School/Grade: 交大資科工所 碩一

Student ID: 309551004 (王冠中)

ID: aesophor

## wishMachine (250 pts)

我們可以搜尋 "Welcom" 字串來找到它在哪裡被使用到,而經過搜尋可以發現是 sub\_400B8D()。



Address	Function	Instruction
sub_400B8D:5 (400B8D)	sub_410BB0("Welcom to t	

```
1__int64 sub_400B8D()
2{
3    __int64 result; // rax
4
5    sub_410BB0("Welcom to the wish machine");
6    sub_410BB0("Give me 1000 coins, and I'll make your wish come true");
7    sub_410BB0("Each coin has a serial number on it, show me the number");
8    result = sub_44A300(0);
9    if ( result == -1 )
10        sub_40F130(0LL, 0LL);
11    return result;
12}
```

但這個程式會呼叫 ptrace 這個 syscall,導致我們在正常的情況下無法用 gdb 跟進去 debug。有兩個方法,一個是 patch binary,另外一個方法是讓 呼叫 ptrace 的那個 syscall 指令執行完之後,就用 gdb 把 rax 改成 0,這樣就可以 bypass ptrace

接下來同樣用字串搜尋,去搜尋 "One by one" 字串,可以找到 sub\_400BE6()。這個 loop 會跑 1000 次,每次都問你第 i 個金幣的 serial number,只要讓 loop 順利跑過就會得到 flag。

```
for (i = 0; i \le 999; ++i)
151
152
       printf((unsigned __int64)"One by one, What is the serial number of coin%d ?");
memset(&v2, OLL, 70LL);
153
154
       scanf((unsigned __int64)"%70s");
155
156
       func1((__int64)&v2);
       func2(&v3, &v2);
158
     }
     printf((unsigned __int64)"Ok, the flag is %s");
159
160
     result = 0LL;
     if ( __readfsqword(0x28u) != v74 )
161
       sub_44BD80();
162
163
     return result;
164}
```

func1: 做一連串的運算,最後又 call 一個 function pointer

```
2 {
    __int64 result; // rax
   signed int i; // [rsp+1Ch] [rbp-4h]
   for ( i = 0; i \le 69; i += step )
      store_arg1_addr = a1;
      str_offset = *((_DWORD *)&unk_6D5114 + 10 * base);
10
      step = *((DWORD *)&unk_6D5118 + 10 * base);
11
      s1 = *((_DWORD *)&unk_6D511C + 10 * base);
12
      s2 = secret_arr[10 * base];
13
     func offset = *((DWORD *)\&unk 6D5110 + 10 * base);
     call_func = *((_QWORD *)&unk_6D5100 + 5 * base) + func_offset;
14
15
     ((void (*)(void))call_func)();
16
     ++base;
     result = (unsigned int)step;
17
18
19
   return result;
20}
```

func2:跟 main() 裡的東西進行 XOR

```
1_BYTE *__fastcall func2(__int64 a1, __int64 a2)
2{
3    _BYTE *result; // rax
4    signed int i; // [rsp+1Ch] [rbp-4h]
5
6    for ( i = 0; i <= 69; ++i )
7    {
8       result = (_BYTE *)(i + a1);
9       *result ^= *(_BYTE *)(i + a2);
10    }
11    return result;
12}</pre>
```

接下來實際跑跑看,看看 call\_func 這個 function pointer 會是什麼,實際跑起來會發現只有 5 種結果。這邊的 function pointer 其實就是在檢查你提供的 serial number 是否正確。

5 種可能: 0x4011d6、0x40102d、0x401138、0x4010c8、0x400fbe, 這邊我分別把他們重命名為 check1() ~ check5()

```
int64 check1()
 2 {
     _int64 result; // rax
    int v1; // [rsp+Ch] [rbp-14h]
    int i; // [rsp+10h] [rbp-10h]
    int v3; // [rsp+14h] [rbp-Ch]
    signed int v4; // [rsp+18h] [rbp-8h]
    signed int j; // [rsp+1Ch] [rbp-4h]
10
    for (i = 0; ; ++i)
11
12
      result = (unsigned int)step;
13
      if (i >= step)
14
        break;
15
      v3 = 0;
      v4 = 1;
16
17
      for (j = 0; j < *(char *)(store_arg1_addr + str_offset + i); ++j)
18
19
        v1 = v3 + v4;
20
        v3 = v4;
21
        v4 = v1;
22
      }
23
      if ( v1 != *((_DWORD *)&store_arg1_addr + i + 4LL + 1) )
24
        sub_40F130(0LL);
25
26
    return result;
27}
```

```
2 {
     _int64 result; // rax
    int v1; // [rsp+4h] [rbp-Ch]
    int i; // [rsp+8h] [rbp-8h]
    signed int j; // [rsp+Ch] [rbp-4h]
   for (i = 0; ; ++i)
10
      result = (unsigned int)step;
      if (i >= step)
11
12
        break;
      v1 = 0;
13
14
      for (j = 0; j < *(char *)(store_arg1_addr + str_offset + i); ++j)
15
16
        if (j & 1)
17
          v1 += 2;
18
        else
19
          v1 += 11;
20
      if ( v1 != *((_DWORD *)\&store_arg1_addr + i + 4LL + 1) )
21
22
        sub_40F130(0LL);
23
24
    return result;
25}
```

```
_int64 check3()
 2 {
      _int64 result; // rax
    signed int v1; // [rsp+4h] [rbp-Ch]
    int i; // [rsp+8h] [rbp-8h]
 6
    signed int j; // [rsp+Ch] [rbp-4h]
    for (i = 0; ; ++i)
10
      result = (unsigned int)step;
11
      if (i >= step)
12
        break;
13
      v1 = -88035316;
14
      for (j = 0; j < *(char *)(store_arg1_addr + str_offset + i); ++j)
15
16
        if (j & 1)
17
          v1 -= 120;
18
        else
19
          v1 -= 30600;
20
21
      if (v1 !=*((DWORD *)\&store\_arg1\_addr + i + 4LL + 1))
22
        sub_40F130(0LL);
23
24
    return result;
25}
```

```
1__int64 check4()
2 {
3     __int64 result; // rax
4    int i; // [rsp+Ch] [rbp-4h]
5
6    for ( i = 0; ; ++i )
7    {
8       result = (unsigned int)step;
9       if ( i >= step )
10            break;
11       if ( *((_DWORD *)&store_arg1_addr + i + 4LL + 1) != |(*(char *)(store_arg1_addr + str_offset sub_40F130(0LL);
13    }
14    return result;
15}
```

他們的共同點是會從 input 抓某個 byte 去比較。這邊我們可以把會用到的 memory 直接 dump 出來,然後模仿這些 function 的邏輯,就可以逆推出正確的 input 來通過檢查了。

```
$ objdump -s --start-address=0x6d5100 --stop-address=0x8a1008 ./wishMachine
```

這邊可以寫個 script 去整理剛剛 dump 出來的 memory data,把他們整理成 signed int array。接下來把上面 5 個 check function 照貼到一個 .c 檔案裡面自己進行逆推。注意在逆推 input 的時候要注意變數的 type 要和原本程式的邏輯一致,否則如果遇到 signed integer overflow 的狀況就會推錯。剩下的就是模仿程式邏輯找出 1000 次 loop 所需的正確 input 即可拿到 flag。

```
-- trimmed --
One by one, What is the serial number of coin997 ?
One by one, What is the serial number of coin998 ?
One by one, What is the serial number of coin999 ?
Ok, the flag is
FLAG{7hes3_func710n_ptrs_g1v3_m3_l0t_0000f_f4n_I_w4n7_m00r3_11!!l1!|!}
```

flag: FLAG{7hes3\_func710n\_ptrs\_g1v3\_m3\_l0t\_0o0of\_f4n\_I\_w4n7\_m00r3\_11!!l1!|!}

## SecureContainProtect (250 pts)

這題要先解數讀,透過網路上的 sudoku solver 可以解出來:

1112171513161419j5h7h1h2h8hhh4h9j6115141111813j6h9h8hh3h2h4hh1j3161918111211j4hh5h9h6hh7h8h2j51211917141311j7hh9h6h2hhh3h4j711613111811512z

底下可以看到 byte\_202E00 、 byte\_202020 是兩個長度都是 k 的 array。程式會 byte-wise 將 (user input) XOR (byte\_202E00) XOR (byte\_202020),然後 v6 持續累加,只要讓下圖中 line 39 的 if statement 成立,使得 v6 == 257498 就會印出 flag。

```
if ( v2 == -194062 \&\& v3 == 14763 )
29
     v6 = 0;
      printf("%3161s%159s", byte_2020E0, byte_202D40);
       _isoc99_scanf("%39s", s);
      for (k = 0; k \le 6014; ++k)
34
        v0 = byte_202E00[k];
        v1 = byte_202020[k % 81];
        byte_202E00[k] = v1 ^ s[k % strlen(s)] ^ v0;
36
        v6 += byte_202E00[k];
      if (v6 == 257498)
39
        puts(byte_202E00);
     else
        puts(
          "\n"
          "\n"
          "You are not the agent....\n"
46
          "Are you trying to steal our secret?\n"
          "A well-trained killer is coming to your place for a cup of tea...");
     exit(0);
    printf("\n\nWrong, you are bad at sudoku.");
50
    exit(0);
```

最後要輸入 Action Code。這邊我突然想到 ascii art 的特色是:用到的字不多,很多一樣的 char 會頻繁出現。

所以我讓 ascii art 跟 sudoku 的答案做 XOR,然後把結果拿去 XOR 所有的 printable character, 會看到下面這個結果:

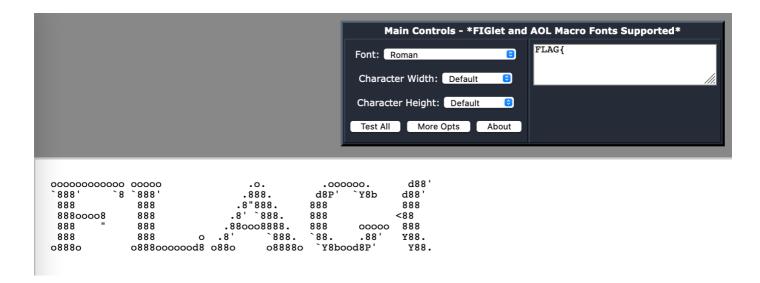
```
Key = 1e: ......Z....PJaQXam}n....T.SLGNJ
Key = 1f: ...../[....QK`PY`l|o....U.RMFOK
Key = 20: +*,=6?;;'*.d ,:"*nt_of_SCP-2521j*mrypt
Key = 21: *+-<7>:::&+.e!-;#+ou^ng^RBQ,3430k+lsxqu
```

0x20 (whitespace) 有: nt of SCP-2521j\*mrypt

```
Key = 6e: edbsxqu^uid^*nbtld :.!(....c|{|.$d#<7>:
Key = 6f: decrypt_the_+ocume!;. )....b}z}~%e"=6?;
Key = 70: {z|mfok@kwz@4p|jrz>$.?6....}beba:z=") $
```

Ox6f(o)有: decrypt\_the+ocume

這邊很像 action code,經過網路上的 ASCII art generator 可以發現 Roman 這個字型是由大量的 0x20 與 0x6f 組成的



只要我們將第一行拿去 XOR 就可以拿到 action code: decrypt\_the\_document\_of\_SCP-2521

```
Hello, Agent. No one would play this boring game, except you.
So we put the next order for you in this game.
Now, It's the last step.Input your action code.
ACTION CODE:00000000000 00000
               `888
                                                                 d88'
                                                                                    `888
                888
                                                                          .00000
                                                                                     888
                                                                                                                            .00.
 88800008
                                                                                                                888P"Y88bP"Y88b
                888
                                  .8' `888.
 888
                888
                                .880008888.
                                                888
                                                                 888
                                                                        888
                                                                                           888
                                                                                                                888
                                                                                                                              888
                                                                                                                                       `88..8
                                                                                            888
                                                                                                                                         888
                888
                                                         .88'
                                                                        888
                                                                                                                888
                                                                                                                       888
                                                                                                                              888
                                                 `Y8bood8P'
08880
              o880 o00000d8 o880
                                        088880
                                                                        `Y8bod8P'
                                                                                   08880 08880 00000000000 08880 08880 08880
                                                                                                                                     'Y8P'
                             .0000.
                                              .08
                                                                 0000
                                                                                                         .0000.
                                             "888
                                 Y8b
                                                                  888,
                                                                                                       d8P'`
                                                                                                             Y8b
                                  888
                                                                                                      888
                                                                                                              888
               .00000000 888
                                        .0000888
                                                                  888
                                                                       .00.
                                                                                 .00000.
                                                                                            .00000.
                                  888 488
                                                                  888P"Y88h
                                                                               188h
                                                                                     '88h d88'
             888
                   `88h
                                                                                                 `88h 888
                                                                                                              888
                          888
                                             `888
             888
                    888
                          888
                                  888 888
                                             888
                                                                  888
                                                                         888
                                                                               888
                                                                                      888 888
                                                                                                 888 888
                                                                                                              888
              `88bod8P'
                           `88b d88'
                                      888 888 888 888
`Y8bod88P" oooooooooo o888o o888o
                                                                               888
                                                                                      888 888
                                                                                                 888
                                                                                                       `88b
                                                                                                            88b
                            Y8bd8P'
                                                                                `Y8bod8P'
0000000000
              `8000000.
                                                                                           `Y8bod8P
                                                                                                        48bd8A,
                     YD
              "Y88888P
                                                                        .08
                                                                      "888"
                                                  "888"
                                              .0000888
                                                                  .0000888
                                                                                           0000
           0000 0000
                                                                                                          .00000
d88'
      `88b
                                             d881
                                                   888,
                                                          888
                                                                d88'
                                                                       888
                                                                                             88.
                                                                                                        d88'
              `888.]88..8'
`888'`888'
                                                                                              88..8
      888
                                             888
                                                   888
                                                          888
                                                                888
                                                                       888
                                                                                                        888
                                                                                                               888
      888
                                                   888
                                                          888
                                                                888
                                                                       888
                                                                                               888
                                                                                                               888
`Y8bod8P
                                             `Y8bod88P"
                              0000000000
                                  .08
                                 "888"
                                                                                                      .08
                             .0000888
                                                                                                    .088800
                                                               0000 d8b 0000
                                                                                  000 00.00000.
0000
      0000
                                         .00000.
                                                     .00000.
                                        d88' `88b d88'
888
       `888
                           d881
                                  888
                                                               `888""8P
                                                                           `88.
                                                                                 .8'
                                                                                        888'
                                                                                              `88b
                                                                                                      888
 888
       888
                                  888
                                        888000888 888
                                                                888
                                                                                               888
                           888
                                                                                        888
                                                                                                      888
                                        888 .0 888
'Y8bod8P' 'Y8
       888
                                  888
                                                           .08
                                                                                        888
                                                                                               888
                                                                                                      888
 888
                           888
                                                                888
                                                                              888
 `V88V"V8P'
                           `Y8bod88P"
                                                    `Y8bod8P'
                                                                                        888bod8P
                                                                                                      "888"
                                                              d888h
            0000000000
                                                                                        888
                                                                         `Y8P
                                                                                       08880
                                                                         `Y8b dP'
                                                     .000000.
dP' ``
                                                             .00. .000000.
`Y8b dP' ``
                                                                                            `88b
                       0000
                                     080
                .08
                       888
                                                                                      18Y
                                                                                             `88b
                                           .0000.0
d88( "8
              . 088800
                       888
                                                     880
                                                             .d8P
                                                                  880
                                                                          .d8P
                                                                               880
                                                                                      .d8P
                                                                                              888
                        888P"Y88b
                888
                                     `888
                                                           .d8P
                                                                       .d8P
                                                                                    .d8P
                                                                                               88>
                                                         `88'
                                                                                  `88'
                888
                        888
                                             "Y88b.
                                                                                              888
                                                )88b
                                                                                             .88P
                "888" o888o o888o o888o 8""888P'
                                                         Y8P
                                                                      Y8P
                                                                                   Y8P
                                                                                            .88P
/home/aesophor/CTF/nctu-secure-programming-2020-fall/hw8-reverse/3-SecureContainProtect [git::master *] [aesophor@sqlab] [13:55]
0 1 python 2 docker 3 zsh
```

輸入該 action code 後即可獲得 flag: FLAG{oh\_my\_g0d\_hoo0ow\_did\_you\_decrypt\_this???}