JONSCAFE

WindowsOfOpportunity - HackTheBox University CTF 2023

```
┌──(jons㉿01-20-jonathansebastian)-[~/HTBUniv]
└─$ ./windows
A voice comes from the window... 'Password?'
```

Pertama2, seperti biasanya ngerjain RE, coba cek dulu programnya. Kita run dulu

Dia ternyata minta masukan “password”. Setelah itu kita coba decompile. Disini saya pake IDA Pro

```
int __cdecl main(int argc, const char **argv, const char **envp)
{
  char s[43]; // [rsp+0h] [rbp-30h] BYREF
  char v5; // [rsp+2Bh] [rbp-5h]
  unsigned int i; // [rsp+2Ch] [rbp-4h]

  puts("A voice comes from the window... 'Password?'");
  fgets(s, 42, stdin);
  for ( i = 0; i <= 0x24; ++i )
  {
    v5 = s[i] + s[i + 1];
    if ( v5 != arr[i] )
    {
      puts("The window slams shut...");
      return -1;
    }
  }
  puts("The window opens to allow you passage...");
  return 0;
}
```

Dari hasil tersebut disimpulkan, program itu meminta inputan dimasukkan ke variabel S[43]

S adalah array dengan isi 43.

Dan terjadi operasi looping Dimana operasi tersebut melakukan hal ini:

1. V5 = array s ke i + array s ke (i +1)
2. Melakukan pengecekan, apabila v5 tidak sama dengan arr[i] maka print “windo slam blabalba”

Yang jadi pertanyaan, arr[i] itu isinya apa ?

Kita coba cek lagi pake decompiler kesayangan anda masing-masing

```
.data:0000000000004060 arr             db 9Ch, 96h, 0BDh, 0AFh, 93h, 0C3h, 94h, 60h, 0A2h, 0D1h
.data:0000000000004060                                         ; DATA XREF: main+56↑o
.data:0000000000004060                 db 0C2h, 0CFh, 9Ch, 0A3h, 0A6h, 68h, 94h, 0C1h, 0D7h, 0ACh
.data:0000000000004060                 db 96h, 2 dup(93h), 0D6h, 0A8h, 9Fh, 0D2h, 94h, 0A7h, 0D6h
.data:0000000000004060                 db 8Fh, 0A0h, 0A3h, 0A1h, 0A3h, 56h, 9Eh
.data:0000000000004060 _data           ends
```

Ternyata arr[i] itu isinya seperti itu.

Jadi bisa disimpulkan bahwa program tersebut melakukan pengecekan dengan cara menambahkan s[i] dengan s[i+1] atau index selanjutnya. Setelah itu dikomparasi dengan variabel arr[i] dengan isi seperti di ss sebelumnya.

Kita lakukanr reversing terhadap algoritma tersebut.

Untuk operasi v5 = s[i] + s[i+1] bisa dibalik untuk mencari s[i] yang tidak diketahui.

Karena tidak ada clue terkait nilai array dalam variabel s. asumsi kan S[1] = H (diasumsikan berdasarkan format flag yaitu HTB{flag}). Maka kita hanya perlu mencari nilai s[i+1] dan seterusnya. Dengan matematika sederhana diperoleh

```
s[i + 1] = arr[i] - s[i];
```

dengan s[i] = ‘H’.

untuk nilai arr[i] akan kita coba ubah dari hex ke decimal karena perhitungan tersebut biasanya dilakukan dalam nilai decimal

diperoleh

```
    int arr[] = {156, 150, 189, 175, 147, 195, 148, 96, 162, 209,
                 194, 207, 156, 163, 166, 104, 148, 193, 215, 172,
                 150, 147, 147, 214, 168, 159, 210, 148, 167, 214,
                 143, 160, 163, 161, 163, 86, 158};
```

Maka kita coba susun programnya

```
#include <stdio.h>

int main() {
    char s[37] = {'H'};
    int arr[] = {156, 150, 189, 175, 147, 195, 148, 96, 162, 209,
```

```
                194, 207, 156, 163, 166, 104, 148, 193, 215, 172,
                150, 147, 147, 214, 168, 159, 210, 148, 167, 214,
                143, 160, 163, 161, 163, 86, 158};

    for (int i = 0; i <= 0x24; ++i) {
        s[i + 1] = arr[i] - s[i];

        printf("%c", s[i]); //print dalam bentuk char (%c)
    }

    puts("\nflag");
    return 0;
}
```

Diperoleh flag:

```
HTB{4_d00r_cl0s35_bu7_4_w1nd0w_0p3n5!
```

Tanggal tambahin '}' dan tes, flag correct!

Chall & Solver: https://github.com/jonscafe/ctfs-write-ups/tree/9687fedb533b18cf661b3f5c9d86263ca955ca1b/HTB_Univ2023