jamk

Reverse Engineering

Lab 01

Timo Lehosvuo, M3426

Report
Reverse Engineering, Marko Silokunnas
20.2.2021
ICT

jamk | Jyväskylän ammattikorkeakoulu
University of Applied Sciences

Sisällys

1 Lab 01

I started reverse engineering the file by trying to understand what the main function does:

```
.text:080484C0
.text:080484C0                push    ebp
.text:080484C1                mov     ebp, esp
.text:080484C3                sub     esp, 28h        ; char *
.text:080484C6                mov     eax, [ebp+arg_4]
.text:080484C9                mov     ecx, [ebp+arg_0]
.text:080484CC                lea     edx, aInsertPassword ; "Insert password: "
.text:080484D2                mov     [ebp+var_4], 0
.text:080484D9                mov     [ebp+var_8], ecx
.text:080484DC                mov     [ebp+var_C], eax
.text:080484DF                mov     [ebp+var_10], 0
.text:080484E6                mov     [esp+28h+var_28], edx
.text:080484E9                call    _printf
.text:080484EE                lea     ecx, aD         ; "%d"
.text:080484F4                lea     edx, [ebp+var_10]
.text:080484F7                mov     [esp+28h+var_28], ecx
.text:080484FA                mov     [esp+28h+var_24], edx
.text:080484FE                mov     [ebp+var_14], eax
.text:08048501                call    ___isoc99_scanf
.text:08048506                mov     ecx, [ebp+var_10]
.text:08048509                mov     [esp+28h+var_28], ecx
.text:0804850C                mov     [ebp+var_18], eax
.text:0804850F                call    check_password
.text:08048514                xor     eax, eax
.text:08048516                add     esp, 28h
.text:08048519                pop     ebp
.text:0804851A                retn
.text:0804851A main           endp
```

Figure 1: Main function.

After understanding what the function does, I realized that I am not going to find the answer here, so I focused my attention to the “check_password” function:

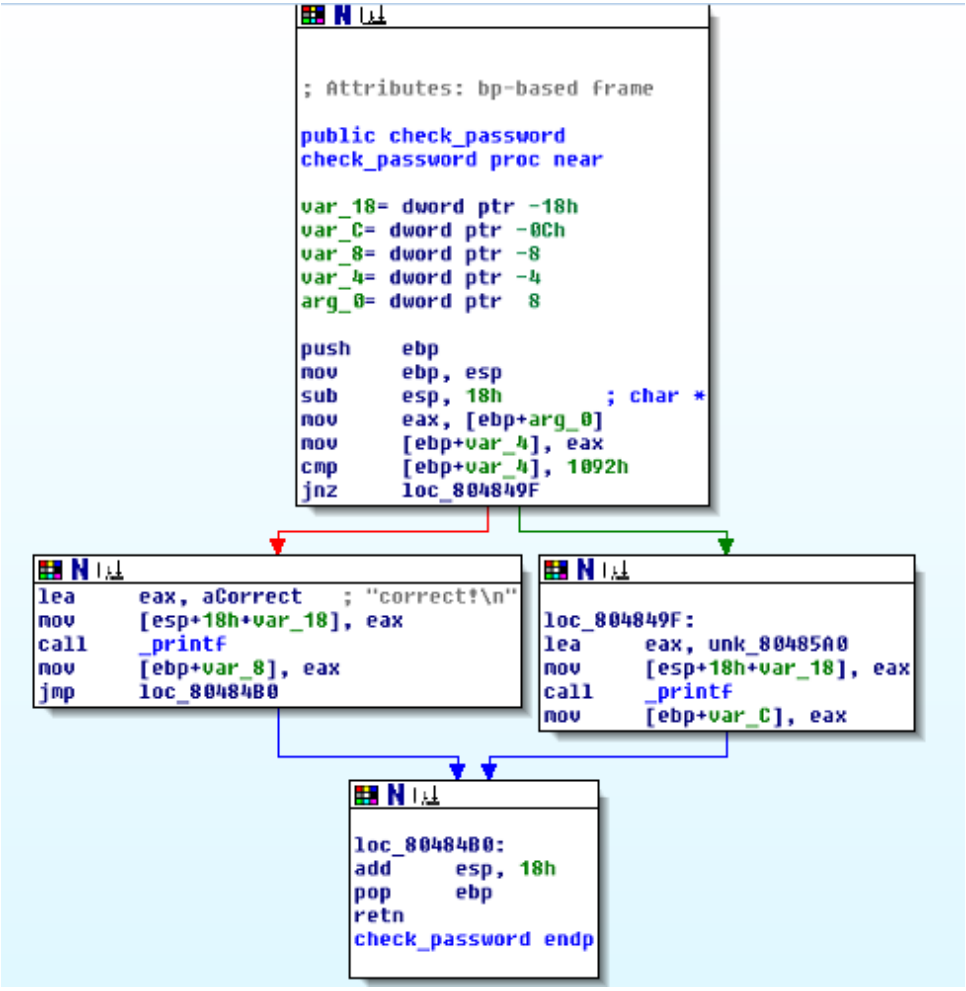

Figure 2: check_password function.

Looking through the function the “cmp” instruction caught my eye. Since it compares the value “1092h” to “[ebp+var_4] I figured that the value “1092h” must contain the password:

Figure 3: "cmp" instruction.

I tested if the password is “1092h” but that did not work so I changed the value to decimal and got “4242”:

Figure 4: Decimal value to 1092h.

At first, I was skeptical since who has only numbers as a password and I let it be and went to look for other clues. After an hour or so I came back just to realize I was right all along:

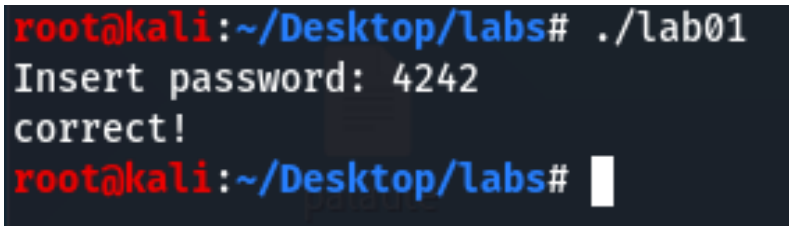

Figure 5: Password check.

2. Time spent

Setting up FlareVM:	4 hours 15 minutes
Solving the lab:	2 hours 45 minutes
Total:	7 hours