Reverse Engineering

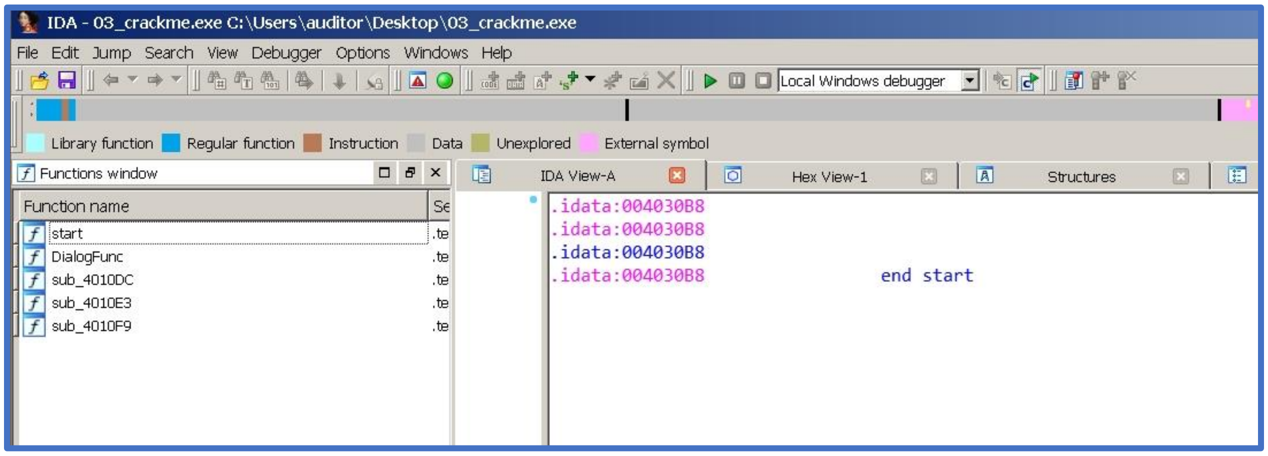
Akhilesh Bandi

05/08/2022

03_crackme.exe

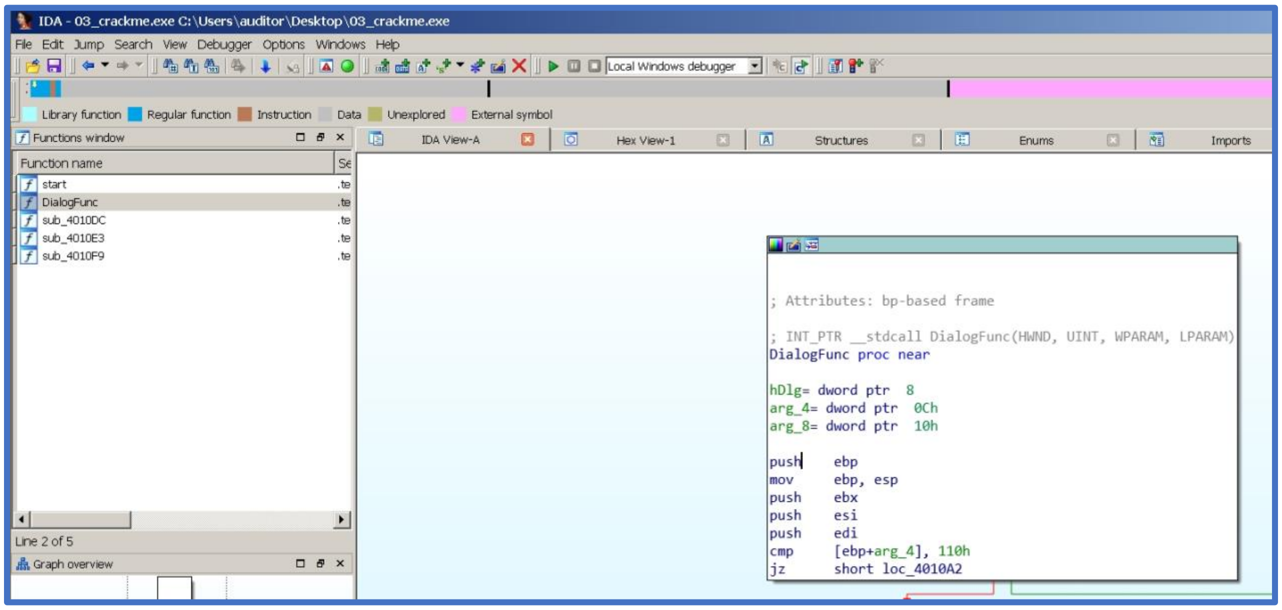
Trial Methods :

Opening the .exe in IDA Disassembler



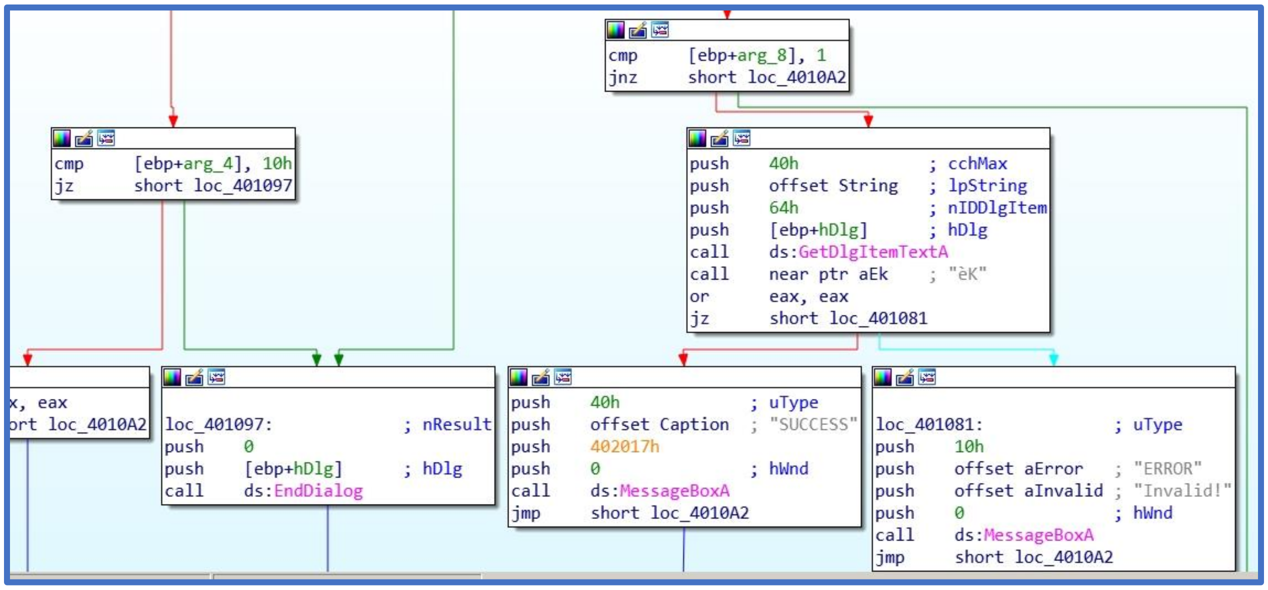
```
.idata:004030B8
.idata:004030B8
.idata:004030B8
.idata:004030B8                 end start
```

Start Function of 03_crackme.exe in IDA Disassembler



```
; Attributes: bp-based frame

; INT_PTR __stdcall DialogFunc(HWND, UINT, WPARAM, LPARAM)
DialogFunc proc near

hDlg= dword ptr  8
arg_4= dword ptr  0Ch
arg_8= dword ptr  10h

push    ebp
mov     ebp, esp
push    ebx
push    esi
push    edi
cmp     [ebp+arg_4], 110h
jz      short loc_4010A2
```

DialogFunc in 03_crackme.exe



```
cmp     [ebp+arg_8], 1
jnz     short loc_4010A2
```

```
cmp     [ebp+arg_4], 10h
jz      short loc_401097
```

```
push    40h             ; cchMax
push    offset String   ; lpString
push    64h             ; nIDDlgItem
push    [ebp+hDlg]      ; hDlg
call    ds:GetDlgItemTextA
call    near ptr aEk    ; "èK"
or      eax, eax
jz      short loc_401081
```

```
x, eax
ort loc_4010A2
```

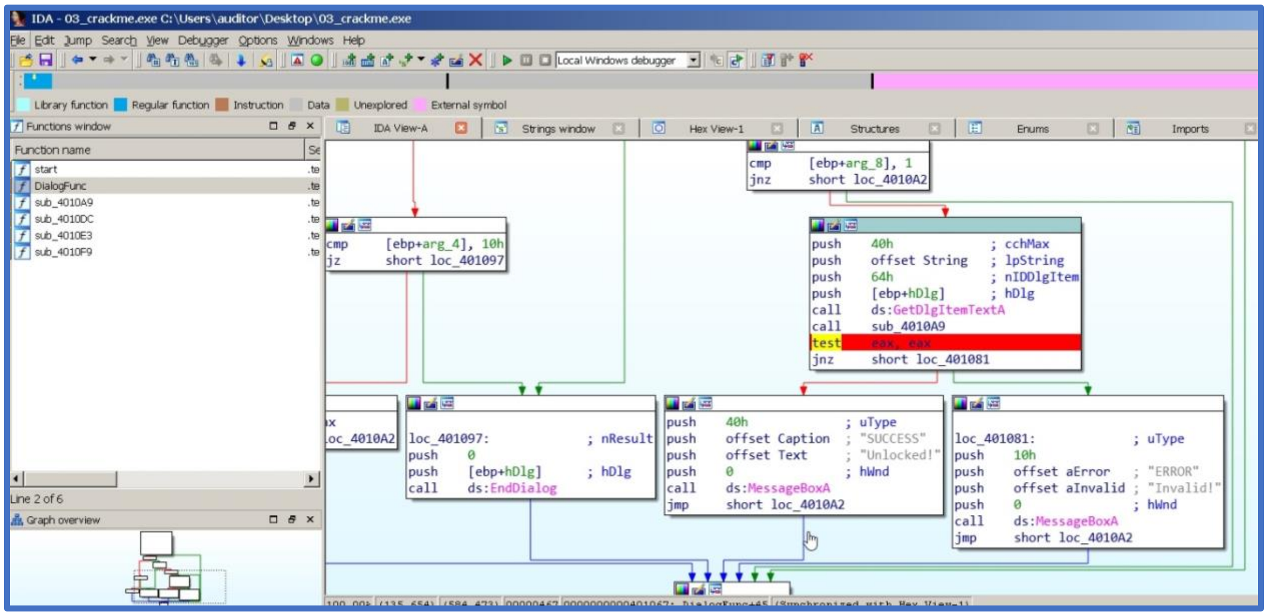
```
loc_401097:                 ; nResult
push    0
push    [ebp+hDlg]          ; hDlg
call    ds:EndDialog
```

```
push    40h             ; uType
push    offset Caption  ; "SUCCESS"
push    402017h
push    0               ; hWnd
call    ds:MessageBoxA
jmp     short loc_4010A2
```

```
loc_401081:                 ; uType
push    10h
push    offset aError   ; "ERROR"
push    offset aInvalid ; "Invalid!"
push    0               ; hWnd
call    ds:MessageBoxA
jmp     short loc_4010A2
```

Observing the Jump when not zero

In x86 assembly code JNZ and JZ are the same which check for the Zero flag to be not set. We can use the IDA disassembler to manually set the zero flag to 1 and check if the program is continuing.



IDA - 03_crackme.exe C:\Users\auditor\Desktop\03_crackme.exe

```
cmp     [ebp+arg_8], 1
jnz     short loc_4010A2
```

```
cmp     [ebp+arg_4], 10h
jz      short loc_401097
```

```
push    40h             ; cchMax
push    offset String   ; lpString
push    64h             ; nIDDlgItem
push    [ebp+hDlg]      ; hDlg
call    ds:GetDlgItemTextA
call    sub_4010A9
test    eax, eax
jnz     short loc_401081
```

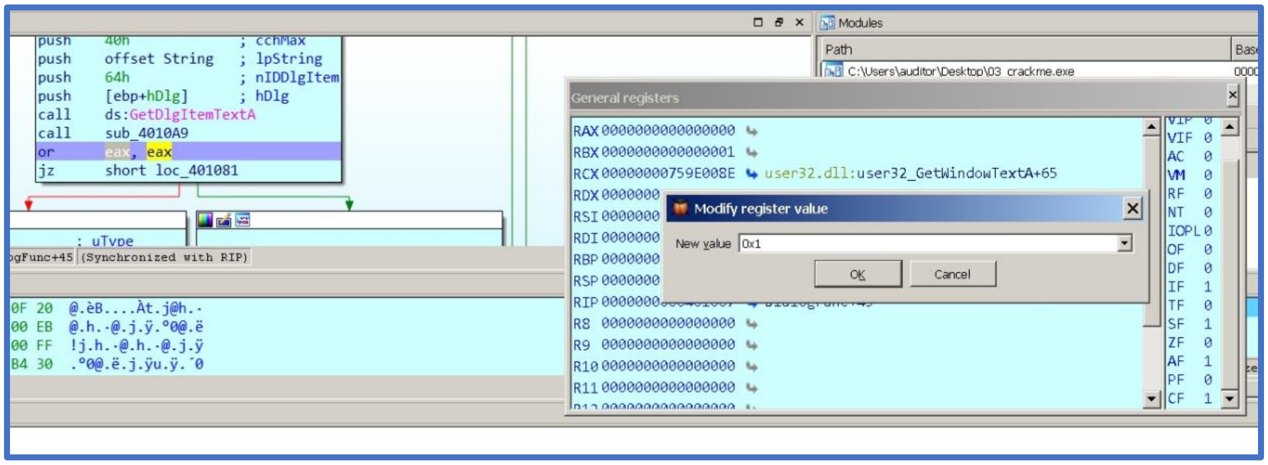
```
x
oc_4010A2
```

```
loc_401097:                 ; nResult
push    0
push    [ebp+hDlg]          ; hDlg
call    ds:EndDialog
```

```
push    40h             ; uType
push    offset Caption  ; "SUCCESS"
push    offset Text     ; "Unlocked!"
push    0               ; hWnd
call    ds:MessageBoxA
jmp     short loc_4010A2
```

```
loc_401081:                 ; uType
push    10h
push    offset aError   ; "ERROR"
push    offset aInvalid ; "Invalid!"
push    0               ; hWnd
call    ds:MessageBoxA
jmp     short loc_4010A2
```

Setting a Break point above the jump and debugging the program


```
push    40h             ; cchMax
push    offset String   ; lpString
push    64h             ; nIDDlgItem
push    [ebp+hDlg]      ; hDlg
call    ds:GetDlgItemTextA
call    sub_4010A9
or      eax, eax
jz      short loc_401081
```

```
0F 20  @.èB....Àt.j@h.·
00 EB  @.h.·@.j.ÿ.°0@.ë
00 FF  !j.h.·@.h.·@.j.ÿ
B4 30  .°0@.ë.j.ÿu.ÿ.´0
```

```
RAX 0000000000000000
RBX 0000000000000001
RCX 00000000759E008E  user32.dll:user32_GetWindowTextA+65
RDX 0000000
RSI 0000000
RDI 0000000
RBP 0000000
RSP 0000000
RIP 0000000
R8  0000000000000000
R9  0000000000000000
R10 0000000000000000
R11 0000000000000000
```

```
VIF  0
AC   0
VM   0
RF   0
NT   0
IOPL 0
OF   0
DF   0
IF   1
TF   0
SF   1
ZF   0
AF   1
PF   0
CF   1
```

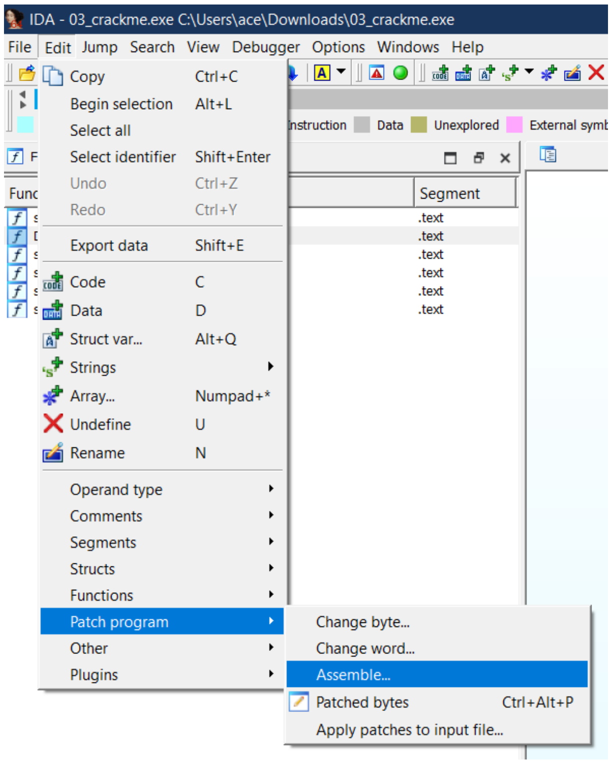
Modify register value

New value 0x1

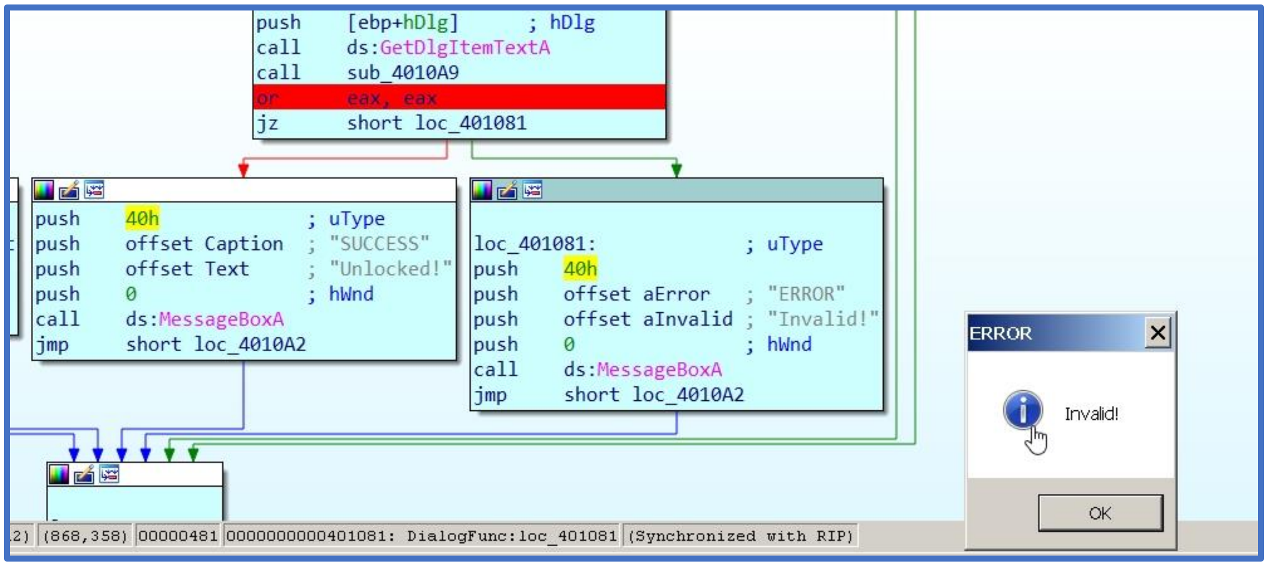
Modules
Path
C:\Users\auditor\Desktop\03_crackme.exe
General registers
OK
Cancel
; uType
gFunc+45 (Synchronized with RIP)

Changing the ZF flag to 1 in the debugger.

The process still resulted in an error window



Trying to Patch the Program



```
push    [ebp+hDlg]      ; hDlg
call    ds:GetDlgItemTextA
call    sub_4010A9
or      eax, eax
jz      short loc_401081
```

```
push    40h               ; uType
push    offset Caption    ; "SUCCESS"
push    offset Text       ; "Unlocked!"
push    0                 ; hWnd
call    ds:MessageBoxA
jmp     short loc_4010A2
```

```
loc_401081:               ; uType
push    40h
push    offset aError     ; "ERROR"
push    offset aInvalid   ; "Invalid!"
push    0                 ; hWnd
call    ds:MessageBoxA
jmp     short loc_4010A2
```

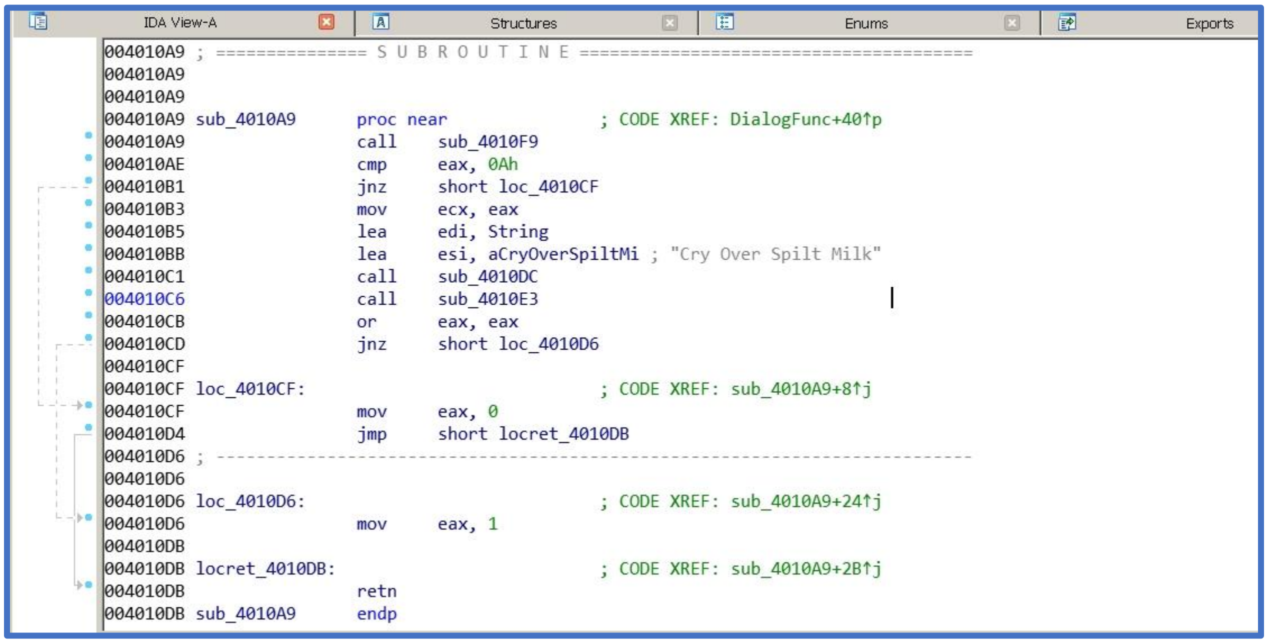
```
.2) (868,358) 00000481 0000000000401081: DialogFunc:loc_401081 (Synchronized with RIP)
```

Trying to change the Error sequence to resemble the success sequence.

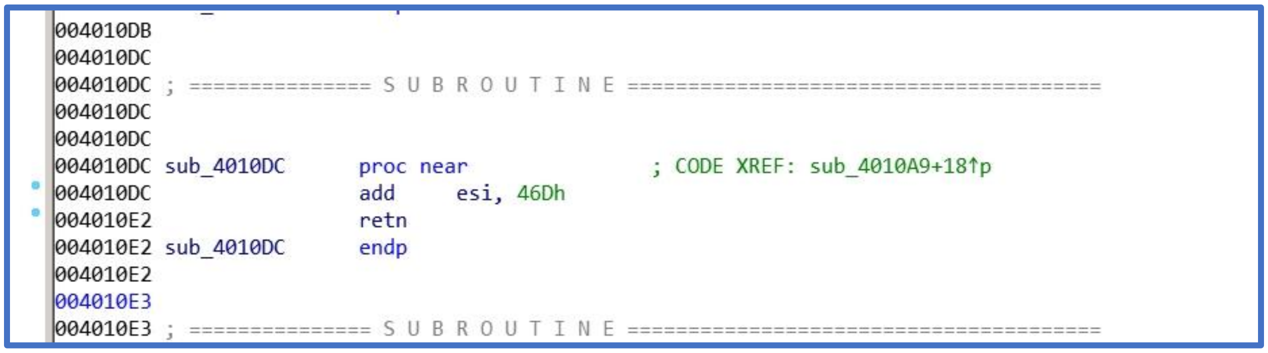
Able to change the symbol by pushing 40h but not everything else.

Final Successful Method :

Following each individual Sub-Routine

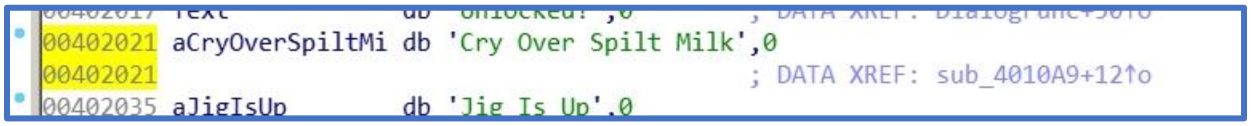


```
004010A9 ; =============== S U B R O U T I N E =======================================
004010A9
004010A9
004010A9 sub_4010A9      proc near               ; CODE XREF: DialogFunc+40↑p
004010A9                 call    sub_4010F9
004010AE                 cmp     eax, 0Ah
004010B1                 jnz     short loc_4010CF
004010B3                 mov     ecx, eax
004010B5                 lea     edi, String
004010BB                 lea     esi, aCryOverSpiltMi ; "Cry Over Spilt Milk"
004010C1                 call    sub_4010DC
004010C6                 call    sub_4010E3
004010CB                 or      eax, eax
004010CD                 jnz     short loc_4010D6
004010CF
004010CF loc_4010CF:                             ; CODE XREF: sub_4010A9+8↑j
004010CF                 mov     eax, 0
004010D4                 jmp     short locret_4010DB
004010D6 ; ---------------------------------------------------------------------------
004010D6
004010D6 loc_4010D6:                             ; CODE XREF: sub_4010A9+24↑j
004010D6                 mov     eax, 1
004010DB
004010DB locret_4010DB:                          ; CODE XREF: sub_4010A9+2B↑j
004010DB                 retn
004010DB sub_4010A9      endp
```



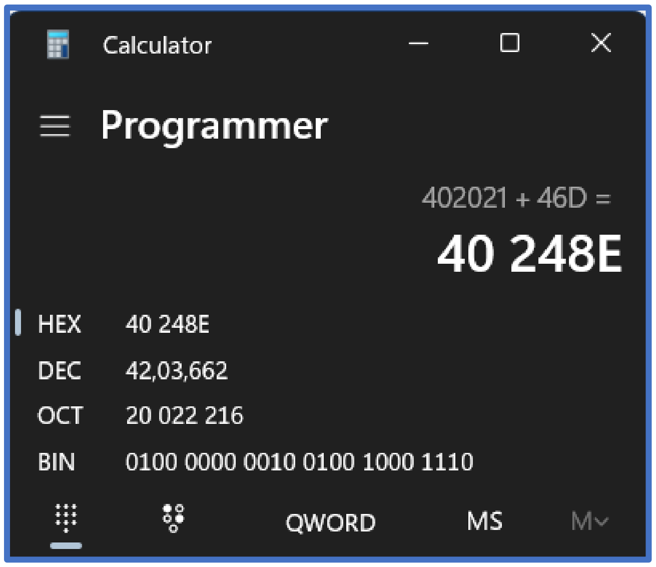
```
004010DB
004010DC
004010DC ; =============== S U B R O U T I N E =======================================
004010DC
004010DC
004010DC sub_4010DC      proc near               ; CODE XREF: sub_4010A9+18↑p
004010DC                 add     esi, 46Dh
004010E2                 retn
004010E2 sub_4010DC      endp
004010E2
004010E3
004010E3 ; =============== S U B R O U T I N E =======================================
```

The ESI pointer used as source pointer for string operations is set to position of 46D



```
00402021 aCryOverSpiltMi db 'Cry Over Spilt Milk',0
00402021                                         ; DATA XREF: sub_4010A9+12↑o
00402035 aJigIsUp        db 'Jig Is Up',0
```

It is starting at 402021



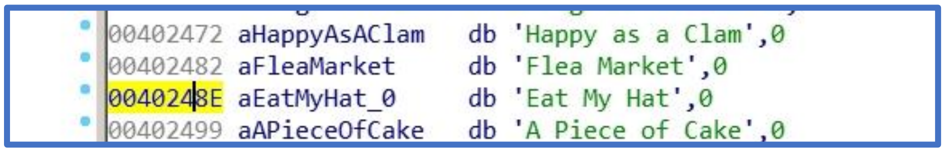
Calculator

Programmer

402021 + 46D =

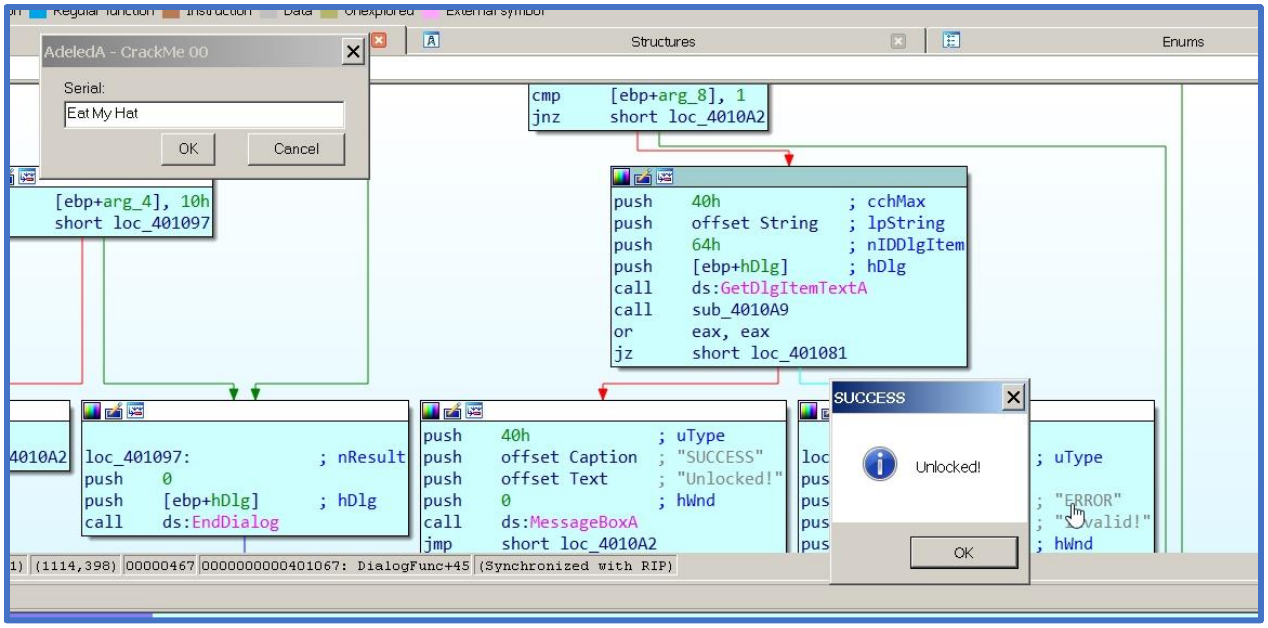
40 248E

```
HEX   40 248E
DEC   42,03,662
OCT   20 022 216
BIN   0100 0000 0010 0100 1000 1110
```



```
00402472 aHappyAsAClam   db 'Happy as a Clam',0
00402482 aFleaMarket     db 'Flea Market',0
0040248E aEatMyHat_0     db 'Eat My Hat',0
00402499 aAPieceOfCake   db 'A Piece of Cake',0
```

402021 + 46D will lead to 40248E



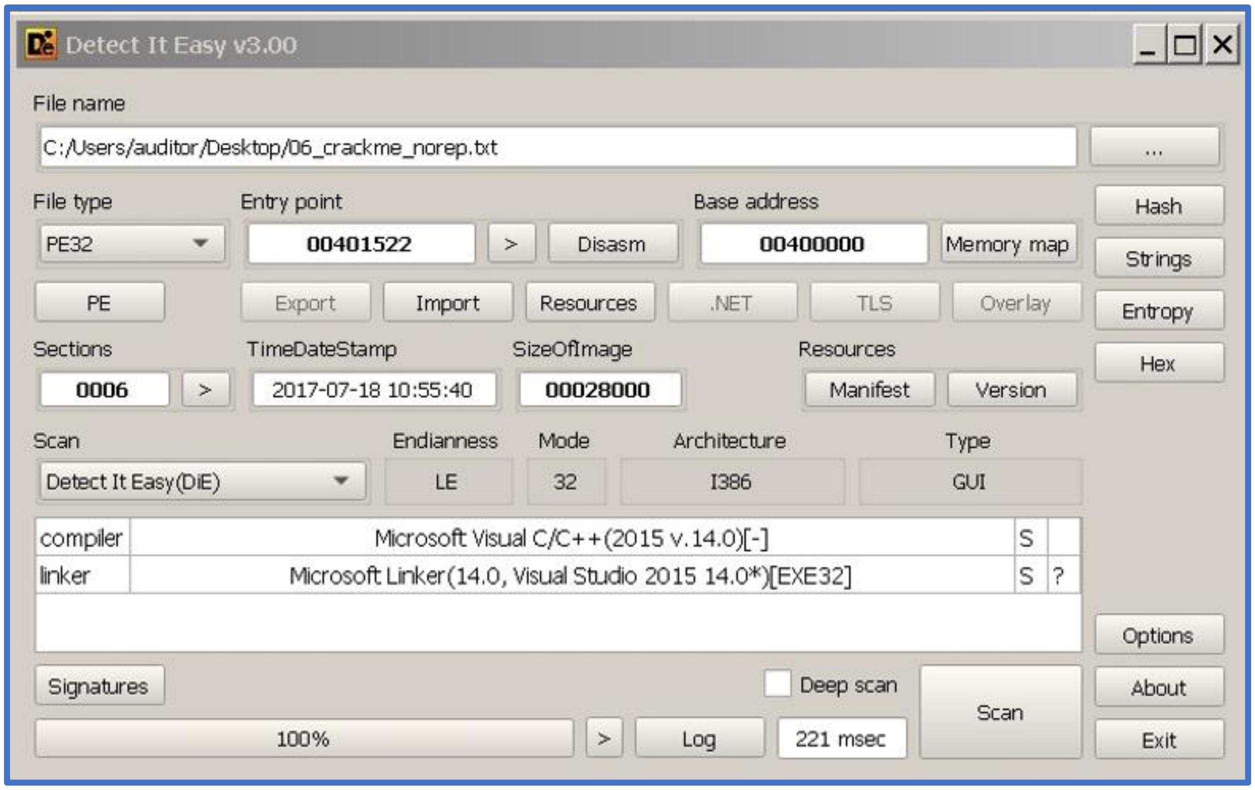
```
cmp     [ebp+arg_8], 1
jnz     short loc_4010A2

push    40h             ; cchMax
push    offset String   ; lpString
push    64h             ; nIDDlgItem
push    [ebp+hDlg]      ; hDlg
call    ds:GetDlgItemTextA
call    sub_4010A9
or      eax, eax
jz      short loc_401081

        [ebp+arg_4], 10h
        short loc_401097

loc_401097:             ; nResult
push    0
push    [ebp+hDlg]      ; hDlg
call    ds:EndDialog

push    40h             ; uType
push    offset Caption  ; "SUCCESS"
push    offset Text     ; "Unlocked!"
push    0               ; hWnd
call    ds:MessageBoxA
jmp     short loc_4010A2

; uType
; "ERROR"
; "Invalid!"
; hWnd
```

```
1) (1114,398) 00000467 0000000000401067: DialogFunc+45 (Synchronized with RIP)
```

Using the string at 40248E Eat my Hat turned out to be the correct password

06_crackme_norep.txt

Opening the file in detect it easy


Detect It Easy v3.00

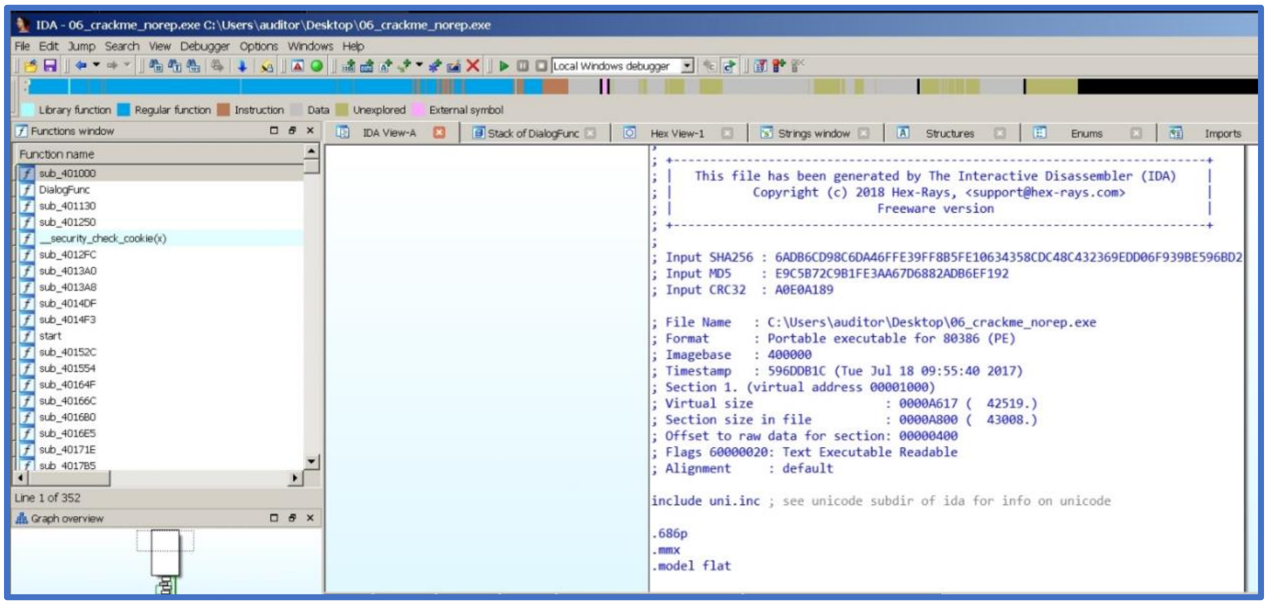
File name: C:/Users/auditor/Desktop/06_crackme_norep.txt

File type	PE32
Entry point	00401522
Base address	00400000
Sections	0006
TimeDateStamp	2017-07-18 10:55:40
SizeOfImage	00028000
Scan	Detect It Easy(DiE)
Endianness	LE
Mode	32
Architecture	I386
Type	GUI

compiler	Microsoft Visual C/C++(2015 v.14.0)[-]	S	
linker	Microsoft Linker(14.0, Visual Studio 2015 14.0*)[EXE32]	S	?

Deep scan: [ ] — 100% — 221 msec

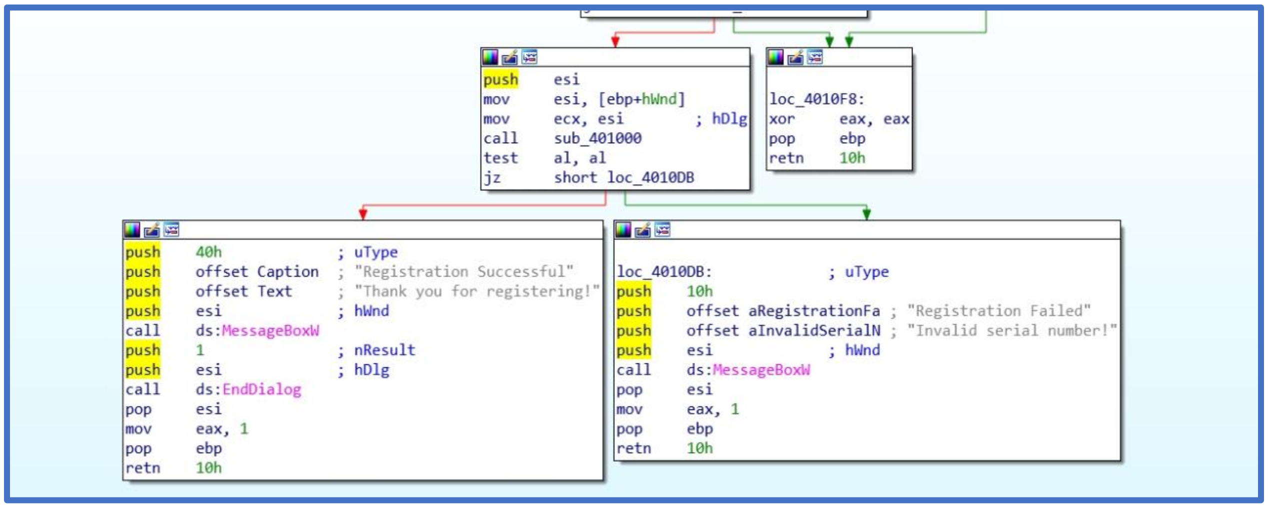
Observed that it’s a 32 bit executable


IDA - 06_crackme_norep.exe C:\Users\auditor\Desktop\06_crackme_norep.exe

```
Function name
sub_401000
DialogFunc
sub_401130
sub_401250
__security_check_cookie(x)
sub_4012FC
sub_4013A0
sub_4013A8
sub_4014DF
sub_4014F3
start
sub_40152C
sub_401554
sub_40164F
sub_40166C
sub_4016B0
sub_4016E5
sub_40171E
sub_4017B5
Line 1 of 352
```

```
; +-------------------------------------------------------------------------+
; |   This file has been generated by The Interactive Disassembler (IDA)    |
; |           Copyright (c) 2018 Hex-Rays, <support@hex-rays.com>           |
; |                            Freeware version                             |
; +-------------------------------------------------------------------------+
;
; Input SHA256 : 6ADB6CD98C6DA46FFE39FF8B5FE10634358CDC48C432369EDD06F939BE596BD2
; Input MD5    : E9C5B72C9B1FE3AA67D6882ADB6EF192
; Input CRC32  : A0E0A189

; File Name   : C:\Users\auditor\Desktop\06_crackme_norep.exe
; Format      : Portable executable for 80386 (PE)
; Imagebase   : 400000
; Timestamp   : 596DDB1C (Tue Jul 18 09:55:40 2017)
; Section 1. (virtual address 00001000)
; Virtual size                  : 0000A617 (  42519.)
; Section size in file          : 0000A800 (  43008.)
; Offset to raw data for section: 00000400
; Flags 60000020: Text Executable Readable
; Alignment     : default

include uni.inc ; see unicode subdir of ida for info on unicode

.686p
.mmx
.model flat
```

Renamed the file to .exe and launched in IDA Disassembler



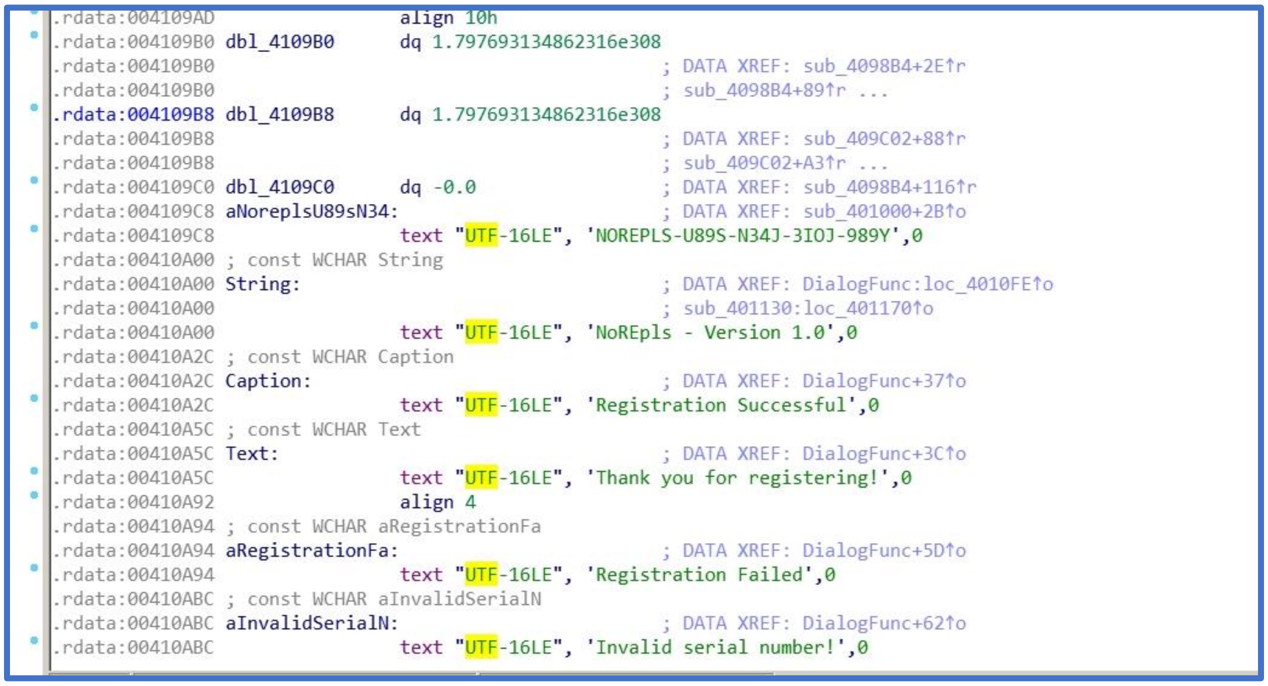
```
push    esi
mov     esi, [ebp+hWnd]
mov     ecx, esi        ; hDlg
call    sub_401000
test    al, al
jz      short loc_4010DB
```

```
loc_4010F8:
xor     eax, eax
pop     ebp
retn    10h
```

```
push    40h             ; uType
push    offset Caption  ; "Registration Successful"
push    offset Text     ; "Thank you for registering!"
push    esi             ; hWnd
call    ds:MessageBoxW
push    1               ; nResult
push    esi             ; hDlg
call    ds:EndDialog
pop     esi
mov     eax, 1
pop     ebp
retn    10h
```

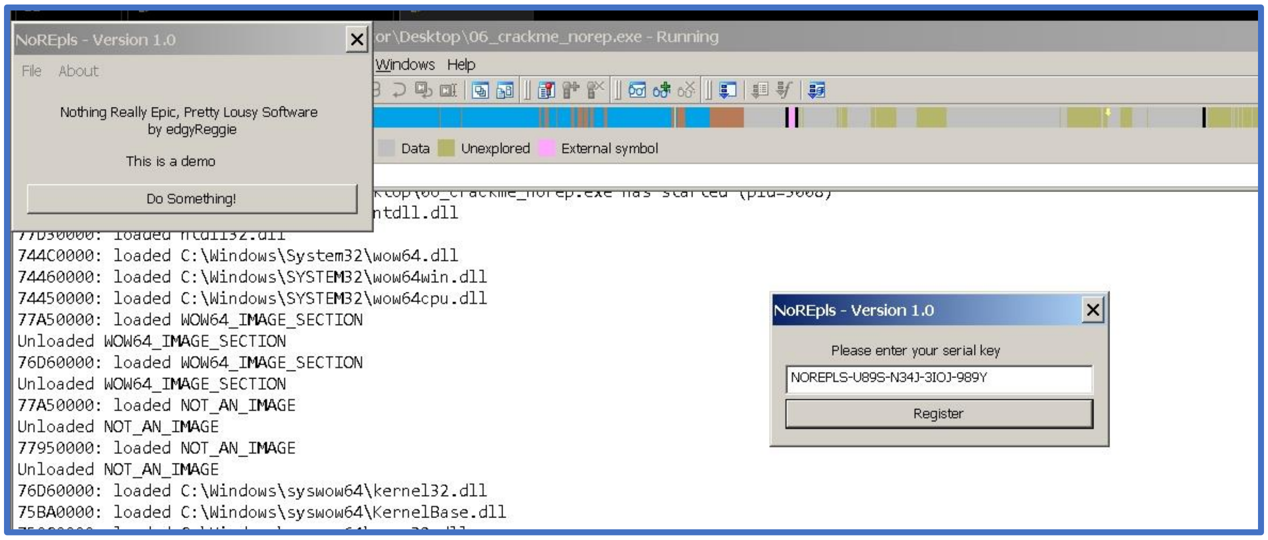
```
loc_4010DB:             ; uType
push    10h
push    offset aRegistrationFa ; "Registration Failed"
push    offset aInvalidSerialN ; "Invalid serial number!"
push    esi             ; hWnd
call    ds:MessageBoxW
pop     esi
mov     eax, 1
pop     ebp
retn    10h
```

Observed the conditional Flow in Graph View



```
.rdata:004109AD                 align 10h
.rdata:004109B0 dbl_4109B0      dq 1.797693134862316e308
.rdata:004109B0                                         ; DATA XREF: sub_4098B4+2E↑r
.rdata:004109B0                                         ; sub_4098B4+89↑r ...
.rdata:004109B8 dbl_4109B8      dq 1.797693134862316e308
.rdata:004109B8                                         ; DATA XREF: sub_409C02+88↑r
.rdata:004109B8                                         ; sub_409C02+A3↑r ...
.rdata:004109C0 dbl_4109C0      dq -0.0                 ; DATA XREF: sub_4098B4+116↑r
.rdata:004109C8 aNoreplsU89sN34:                        ; DATA XREF: sub_401000+2B↑o
.rdata:004109C8                 text "UTF-16LE", 'NOREPLS-U89S-N34J-3IOJ-989Y',0
.rdata:00410A00 ; const WCHAR String
.rdata:00410A00 String:                                 ; DATA XREF: DialogFunc:loc_4010FE↑o
.rdata:00410A00                                         ; sub_401130:loc_401170↑o
.rdata:00410A00                 text "UTF-16LE", 'NoREpls - Version 1.0',0
.rdata:00410A2C ; const WCHAR Caption
.rdata:00410A2C Caption:                                ; DATA XREF: DialogFunc+37↑o
.rdata:00410A2C                 text "UTF-16LE", 'Registration Successful',0
.rdata:00410A5C ; const WCHAR Text
.rdata:00410A5C Text:                                   ; DATA XREF: DialogFunc+3C↑o
.rdata:00410A5C                 text "UTF-16LE", 'Thank you for registering!',0
.rdata:00410A92                 align 4
.rdata:00410A94 ; const WCHAR aRegistrationFa
.rdata:00410A94 aRegistrationFa:                        ; DATA XREF: DialogFunc+5D↑o
.rdata:00410A94                 text "UTF-16LE", 'Registration Failed',0
.rdata:00410ABC ; const WCHAR aInvalidSerialN
.rdata:00410ABC aInvalidSerialN:                        ; DATA XREF: DialogFunc+62↑o
.rdata:00410ABC                 text "UTF-16LE", 'Invalid serial number!',0
```

Noticed a comparison string in Text View



Nothing Really Epic, Pretty Lousy Software
by edgyReggie

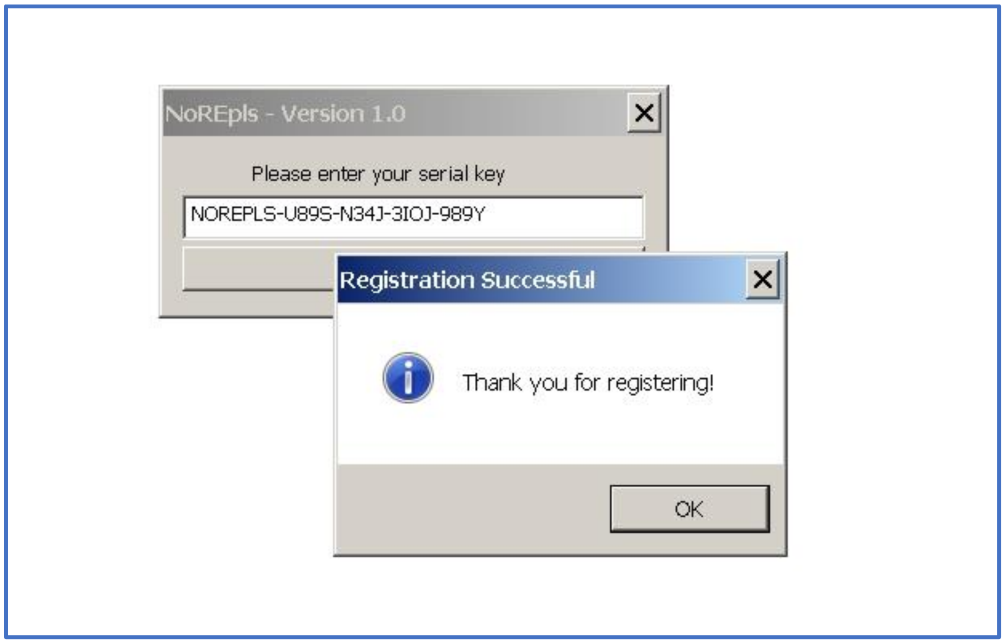
This is a demo

Please enter your serial key

NOREPLS-U89S-N34J-3IOJ-989Y

```
ntdll.dll
744C0000: loaded C:\Windows\System32\wow64.dll
74460000: loaded C:\Windows\SYSTEM32\wow64win.dll
74450000: loaded C:\Windows\SYSTEM32\wow64cpu.dll
77A50000: loaded WOW64_IMAGE_SECTION
Unloaded WOW64_IMAGE_SECTION
76D60000: loaded WOW64_IMAGE_SECTION
Unloaded WOW64_IMAGE_SECTION
77A50000: loaded NOT_AN_IMAGE
Unloaded NOT_AN_IMAGE
77950000: loaded NOT_AN_IMAGE
Unloaded NOT_AN_IMAGE
76D60000: loaded C:\Windows\syswow64\kernel32.dll
75BA0000: loaded C:\Windows\syswow64\KernelBase.dll
```

Entering the found String into the program



Please enter your serial key

NOREPLS-U89S-N34J-3IOJ-989Y

Thank you for registering!

Successfully found the Serial Key