codeengn-basic-L07 풀이

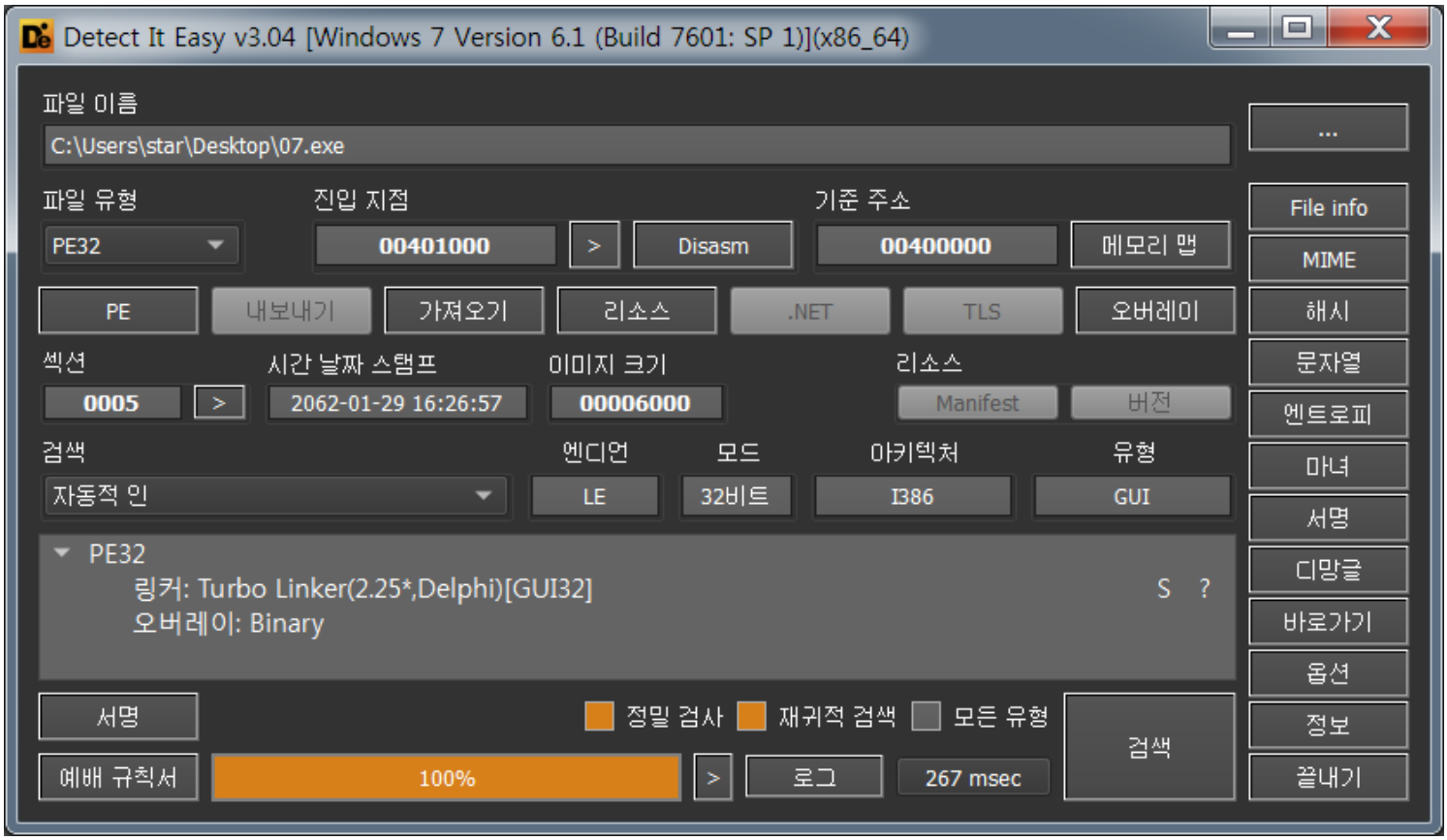
리버싱 문제풀이 / Wonlf / 2022. 3. 20. 17:23

Basic RCE L07

컴퓨터 C 드라이브의 이름이 CodeEngn 일경우 시리얼이 생성될때 CodeEngn은 'ß어떤것'으로 변경되는가

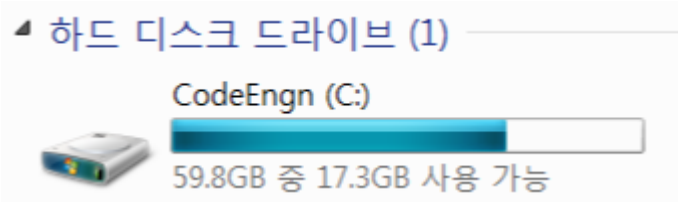
— Author: abex
— File Password: codeengn

C드라이브의 이름이 CodeEngn일 때, CodeEngn문자열이 무엇으로 변하는지 알아내야한다.

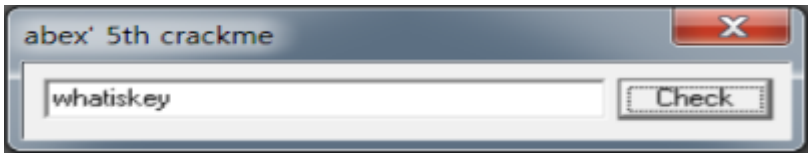


Die로 열어보니 딱히 패킹이 되어있지는 않은듯 하다. 바로 디버깅을 해본다.

문제에서는 C드라이브의 이름이 CodeEngn일때를 가정하고 있으니 변경해준다.

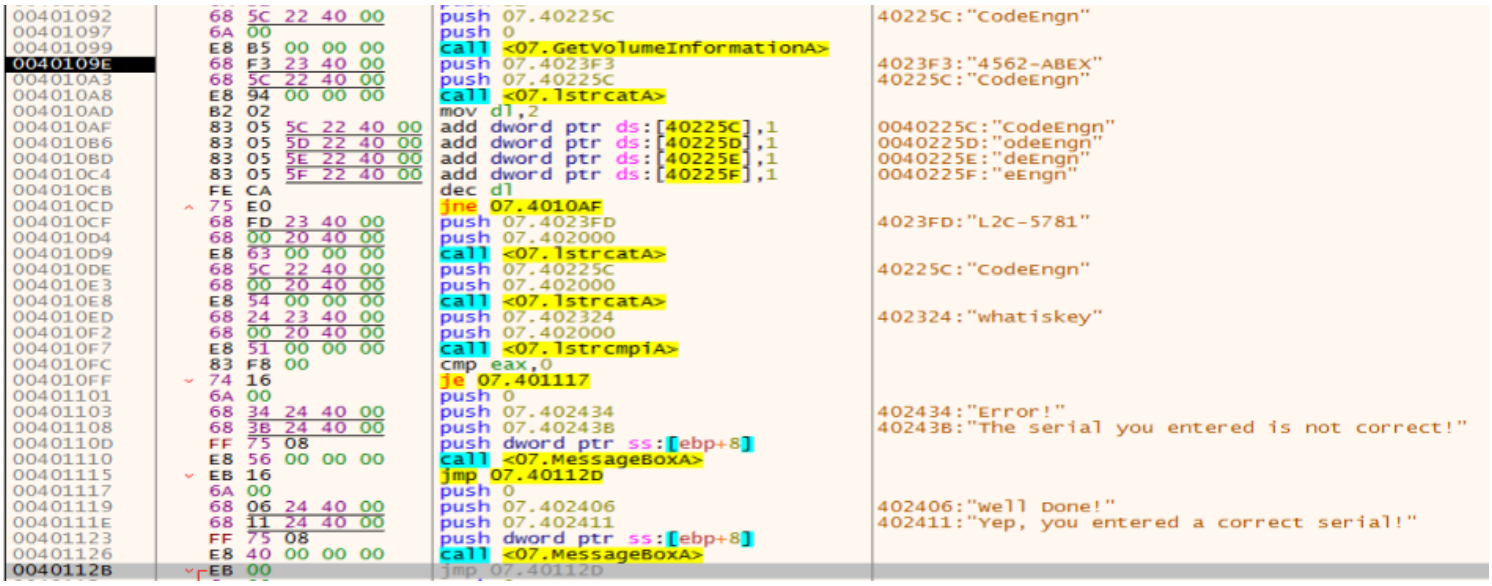


변경하고 디버거로 열어서 key를 입력하고 이 문자열이 변경되는 구문을 찾아본다.

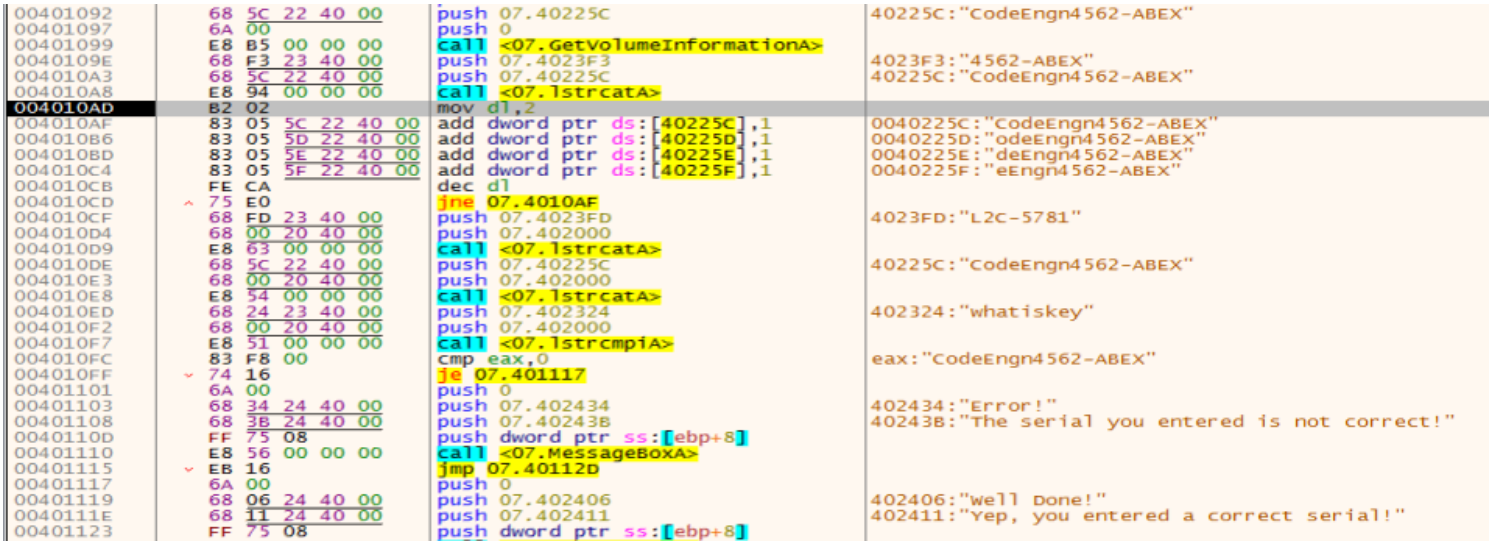



```
00401078    E8 F4 00 00 00      call <07.GetDlgItemTextA>
0040107D    6A 00               push 0
0040107F    6A 00               push 0
00401081    68 C8 20 40 00      push 07.4020C8
00401086    68 90 21 40 00      push 07.402190
0040108B    68 94 21 40 00      push 07.402194
00401090    6A 32               push 32
00401092    68 5C 22 40 00      push 07.40225C
00401097    6A 00               push 0
00401099    E8 B5 00 00 00      call <07.GetVolumeInformationA>
0040109E    68 F3 23 40 00      push 07.4023F3                  4023F3:"4562-ABEX"
004010A3    68 5C 22 40 00      push 07.40225C
004010A8    E8 94 00 00 00      call <07.lstrcatA>
004010AD    B2 02               mov dl,2
004010AF    83 05 5C 22 40 00   add dword ptr ds:[40225C],1
004010B6    83 05 5D 22 40 00   add dword ptr ds:[40225D],1
004010BD    83 05 5E 22 40 00   add dword ptr ds:[40225E],1
004010C4    83 05 5F 22 40 00   add dword ptr ds:[40225F],1
004010CB    FE CA               dec dl
004010CD  ^ 75 E0               jne 07.4010AF
004010CF    68 FD 23 40 00      push 07.4023FD                  4023FD:"L2C-5781"
004010D4    68 00 20 40 00      push 07.402000
004010D9    E8 63 00 00 00      call <07.lstrcatA>
004010DE    68 5C 22 40 00      push 07.40225C
004010E3    68 00 20 40 00      push 07.402000
004010E8    E8 54 00 00 00      call <07.lstrcatA>
004010ED    68 24 23 40 00      push 07.402324                  402324:"whatiskey"
004010F2    68 00 20 40 00      push 07.402000
004010F7    E8 51 00 00 00      call <07.lstrcmpiA>
004010FC    83 F8 00            cmp eax,0
004010FF  v 74 16               je 07.401117
00401101    6A 00               push 0
00401103    68 34 24 40 00      push 07.402434                  402434:"Error!"
00401108    68 3B 24 40 00      push 07.40243B                  40243B:"The serial you entered is not correct!"
0040110D    FF 75 08            push dword ptr ss:[ebp+8]
00401110    E8 56 00 00 00      call <07.MessageBoxA>
00401115  v EB 16               jmp 07.40112D
00401117    6A 00               push 0
00401119    68 06 24 40 00      push 07.402406                  402406:"Well Done!"
0040111E    68 11 24 40 00      push 07.402411                  402411:"Yep, you entered a correct serial!"
00401123    FF 75 08            push dword ptr ss:[ebp+8]
00401126    E8 40 00 00 00      call <07.MessageBoxA>
0040112B  v EB 00               jmp 07.40112D
```

이쪽 부분인것 같다. 육안상으로 특정 메모리 주소에 key로 추정되는 문자열이 보인다. 이것들이 어떻게 쓰이는지 F8로 하나씩 실행시켜 보았다.

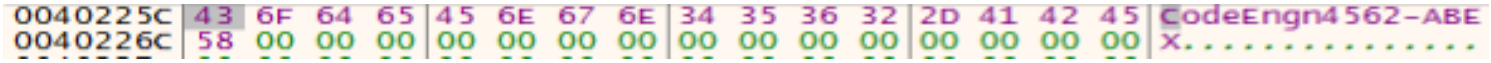


GetVolumeInformationA 함수를 실행시키니, 40225C 주소에 C드라이브의 이름인 "CodeEngn"이 저장되었다.
더 내려가본다.



lstrcatA 함수까지 실행시키니 C드라이브의 이름이 들어있던 주소에 C드라이브의 이름과 특정 문자열이 붙어 다시 저장되었다. 그리고 또 실행시켜보면, dl 레지스터에 2를 넣는다. 그리고...

add명령어를 실행하게 되는데...
현재 40225C에는 이렇게 저장이 되어 있지만 add구문 4번을 도는데, 특정 주소에 있는 값에 1씩 더하라는 명령어이니



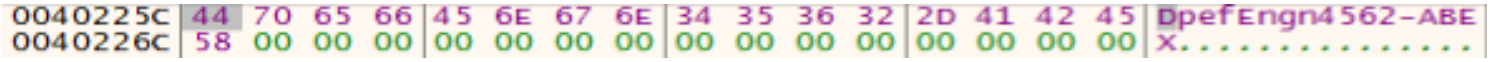
현재 메모리의 값

(40225C) 43 + 1 = 44,

(40225D) 6F + 1 = 70,

(40225E) 64 + 1 = 65,

(40225F) 65 + 1 = 66,

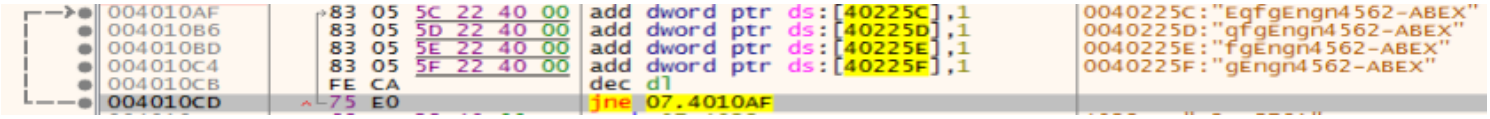


연산 후 메모리의 값

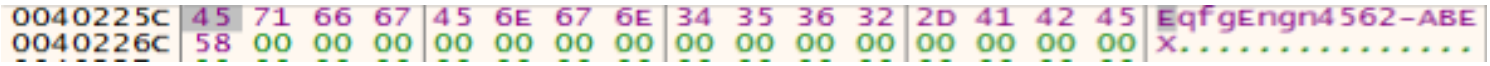
한번 연산을 하고 나면 JNE구문이 있는데

JNE는 같지 않을때 점프를 하는 경우도 있지만, ZeroFlag가 0일 때 점프하는 경우가 있다. 이번에는 후자의 경우이다.

add를 하기 전, mov dl, 2 구문이 있었다. dl안에는 2가 들어있고 add 4번이 지나고 난 뒤에 dec명령어로 1로 만들었다 ZF는 계속 0이고 JNE에서는 ZF가 0이니 특정 주소로 점프를 하게 되니까 add 4번을 하는 주소로 점프하게 된다. 즉 for문 같은 행위라는것. 그래서 2번 add를 하게 되면.

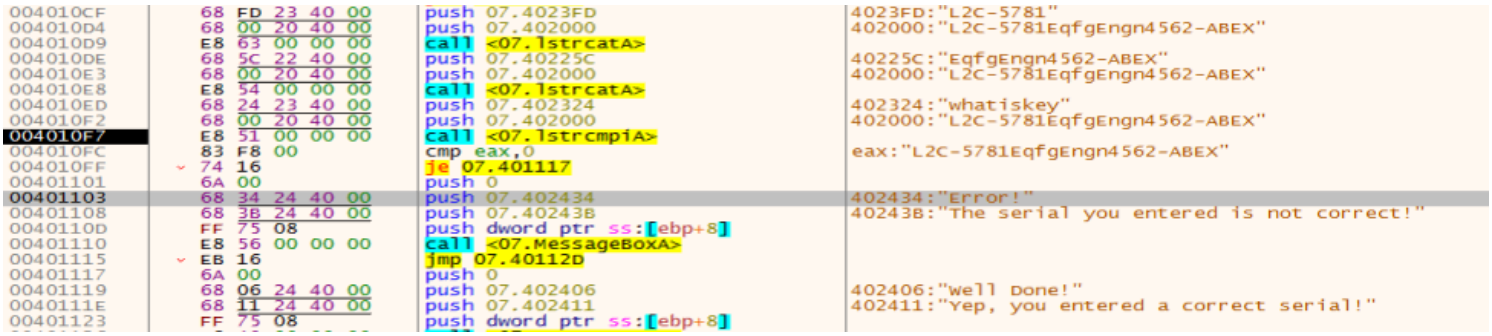


이런식으로 점프하는 곳에 화살표로 표시됨.

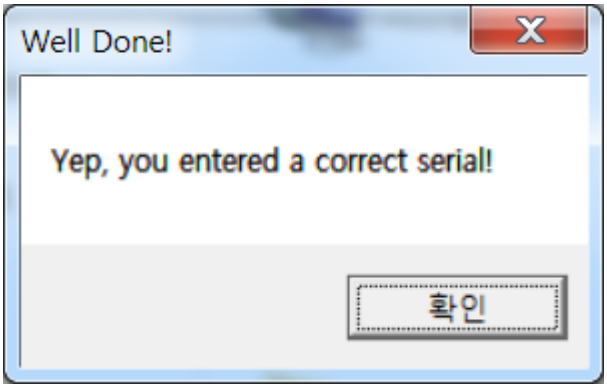


이렇게 값이 변하게 된다. 여기서 이제 또 dec dl을 하게 되는데 그럼 dl에 있는 값이 0이 되게 되고 특정 연산의 값이

0이 됐을 때, ZF는 1로 세팅되게 되니 더이상 JNE를 하게 되는 조건을 벗어난 것이다. 그래서 점프를 하지 않게 되고 아래 구문으로 내려가게 된다.



아래 구문은 정말 쉽게도 특정 주소에 있는 값과 위에서 여러가지 방법으로 변환한 문자열을 합치고, 입력한 값과 비교하여 키인지 아닌지 판단하는 구문으로 되어 있다. 그래서 저기 적혀있는 시리얼로 추정되는 문자열을 입력해주면



정답인듯 하다. 하지만 문제가 원하는 키는 C드라이브의 이름이 CodeEngn일 때 키에서 CodeEngn부분이 어떻게 바뀌는지 알려달라고 했으니 정답은 "EqfgEngn"