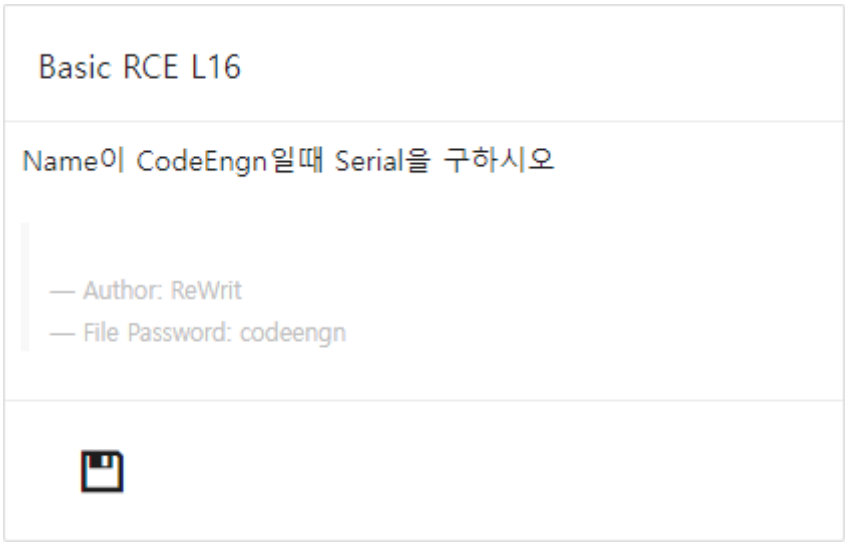
codeengn-basic-L16 풀이

리버싱 문제풀이 / Wonlf / 2022. 4. 19. 17:16

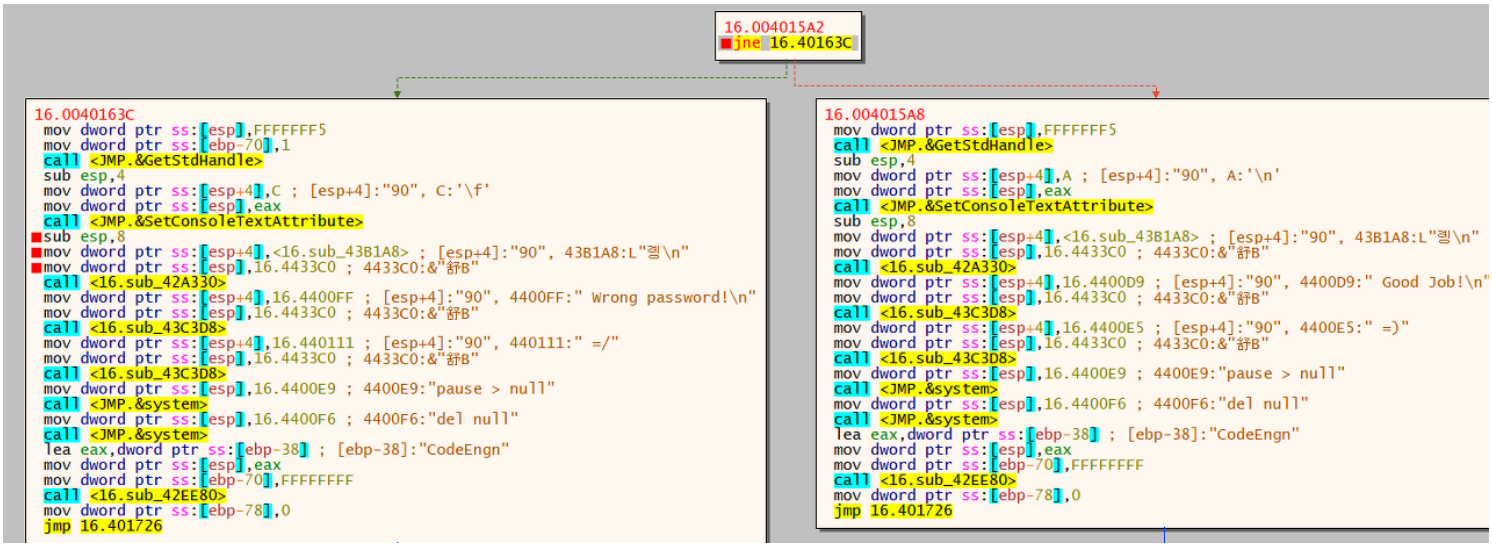


문제는 Name이 CodeEngn일 때, Serial을 원한다.

Die로 확인해보니 특이사항은 보이지 않는다.

디버거로 열어본다.

```
0040159F    3B45 C4          cmp eax,dword ptr ss:[ebp-3C]
004015A2    0F85 94000000    jne 16.40163C
004015A8    C70424 F5FFFFFF  mov dword ptr ss:[esp],FFFFFFF5
004015AF    E8 8CF60000      call <JMP.&GetStdHandle>
004015B4    83EC 04          sub esp,4
004015B7    C74424 04 0A000000 mov dword ptr ss:[esp+4],A                    [esp+4]:"90", A:'\n'
004015BF    890424           mov dword ptr ss:[esp],eax
004015C2    E8 89F60000      call <JMP.&SetConsoleTextAttribute>
004015C7    83EC 08          sub esp,8
004015CA    C74424 04 A8B14300 mov dword ptr ss:[esp+4],<16.sub_43B1A8>      [esp+4]:"90", 43B1A8:L"곌\n"
004015D2    C70424 C0334400  mov dword ptr ss:[esp],16.4433C0                4433C0:&"舒B"
004015D9    E8 528D0200      call <16.sub_42A330>
004015DE    C74424 04 D9004400 mov dword ptr ss:[esp+4],16.4400D9            [esp+4]:"90", 4400D9:" Good Job!\n"
004015E6    C70424 C0334400  mov dword ptr ss:[esp],16.4433C0                4433C0:&"舒B"
004015ED    E8 E6AD0300      call <16.sub_43C3D8>
004015F2    C74424 04 E5004400 mov dword ptr ss:[esp+4],16.4400E5            [esp+4]:"90", 4400E5:" =)"
004015FA    C70424 C0334400  mov dword ptr ss:[esp],16.4433C0                4433C0:&"舒B"
00401601    E8 D2AD0300      call <16.sub_43C3D8>
00401606    C70424 E9004400  mov dword ptr ss:[esp],16.4400E9                4400E9:"pause > null"
0040160D    E8 BEF30000      call <JMP.&system>
00401612    C70424 F6004400  mov dword ptr ss:[esp],16.4400F6                4400F6:"del null"
00401619    E8 B2F30000      call <JMP.&system>
0040161E    8D45 C8          lea eax,dword ptr ss:[ebp-38]                   [ebp-38]:"CodeEngn"
00401621    890424           mov dword ptr ss:[esp],eax
00401624    C745 90 FFFFFFFF mov dword ptr ss:[ebp-70],FFFFFFFF
0040162B    E8 50D80200      call <16.sub_42EE80>
00401630    C745 88 00000000 mov dword ptr ss:[ebp-78],0
00401637    E9 EA000000      jmp 16.401726
0040163C    C70424 F5FFFFFF  mov dword ptr ss:[esp],FFFFFFF5
00401643    C745 90 01000000 mov dword ptr ss:[ebp-70],1
0040164A    E8 F1F50000      call <JMP.&GetStdHandle>
0040164F    83EC 04          sub esp,4
00401652    C74424 04 0C000000 mov dword ptr ss:[esp+4],C                    [esp+4]:"90", C:'\f'
```



문자열 찾기와 그래프로 보기를 사용해 성공으로 가는 구문과 실패로 가는 구문을 찾았다. jne 구문이 이루어지기 전 비교연산을 찾아보겠다.

```
cmp eax, dword ptr ss:[ebp-0x3C]
```

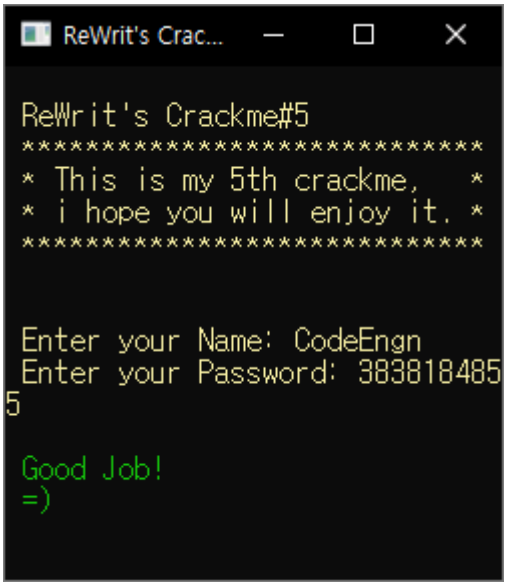
구문이 보인다.

eax와 ebp-0x3C에 있는 값을 비교하여 같다면 점프를 하지 않기 때문에 성공으로 가려면 같아야 한다는 것을 알 수 있다.


```
eax=3039
dword ptr ss:[ebp-3C]=[0070FEEC]=E4C60D97
```

eax에는 내가 입력한 password의 값 12345를 16진수로 변환한 0x3039가 들어있고
ebp-0x3C에는 0xE4C60D97이 들어있다.

시리얼에 이 값을 10진수로 바꾸어 입력해주면..



정답인것 같다. 페이지에 정답 인증을 해준다.

끝

시리얼에 문자열을 넣으면 특정 구문을 통해 eax에 이상한 값이 들어가게 된다.
serial에는 디버깅하기 편하게 숫자를 먼저 넣어보는것으로 하자.