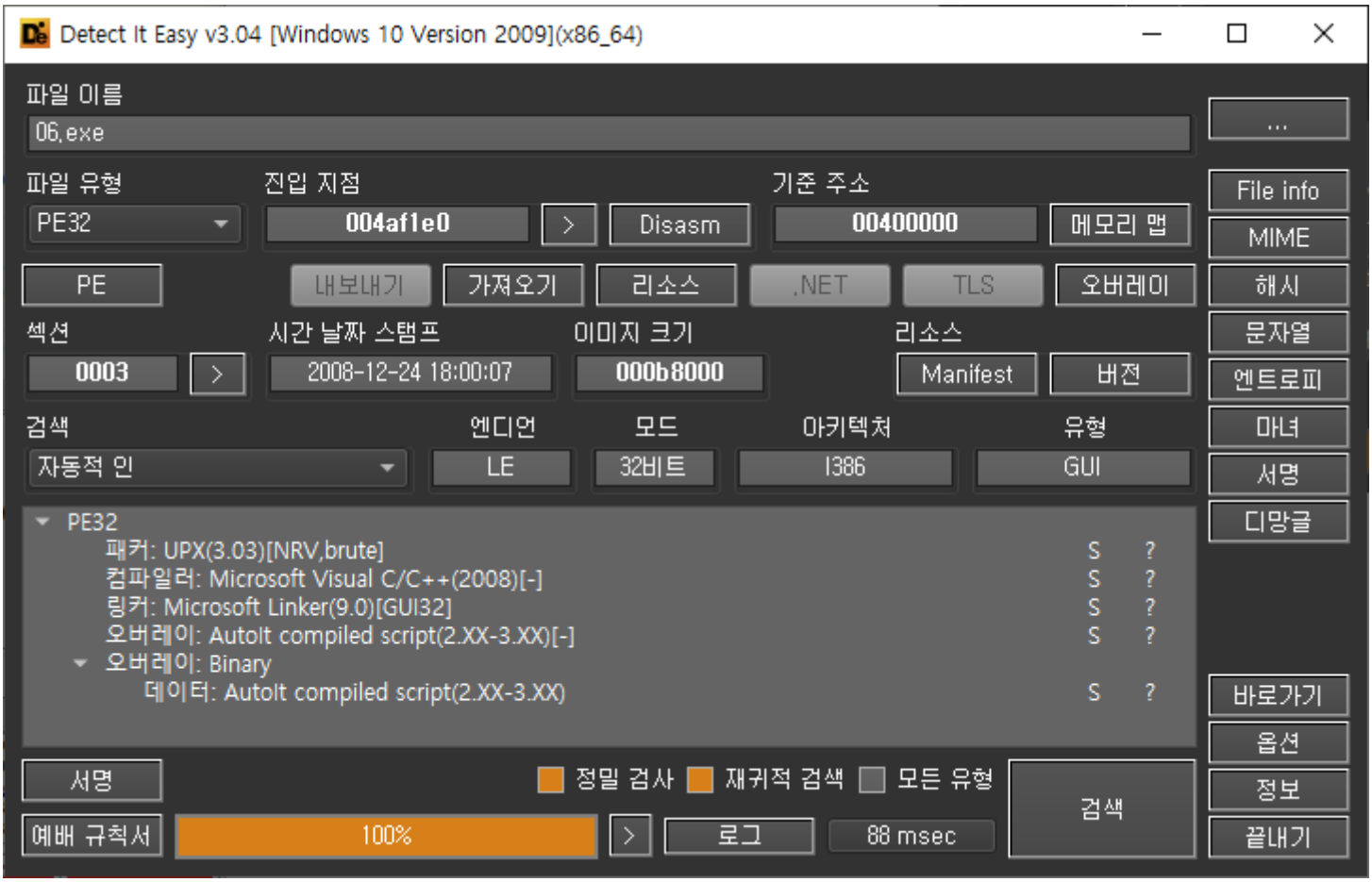
codeengn-advance-L06 풀이

리버싱 문제풀이 / Wonlf / 2022. 5. 9. 21:30

문제는 남은 군생활이 며칠인지 원하고 있다.

Die로 열어본다.

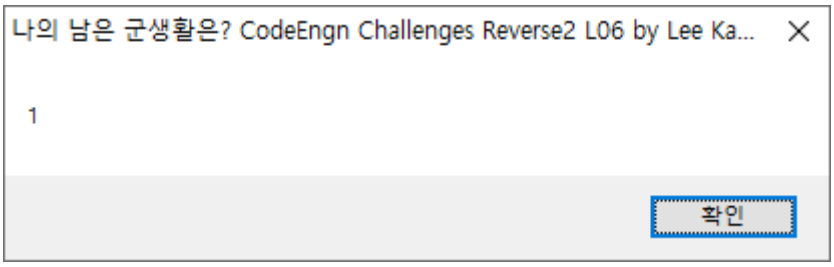


UPX로 패킹이 되어 있다.

언패커로 간단하게 언패킹을 진행해준다.

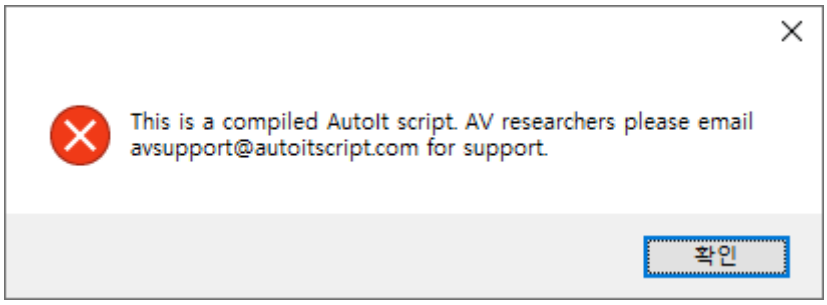
```
                       Ultimate Packer for eXecutables
                          Copyright (C) 1996 - 2020
UPX 3.96w       Markus Oberhumer, Laszlo Molnar & John Reiser   Jan 23rd 2020

        File size         Ratio      Format      Name
   --------------------   ------   -----------   -----------
    613237 <-    290677   47.40%    win32/pe     06.exe

Unpacked 1 file.
```

패킹을 해제한 후, 파일을 실행시켜 본다.



확인을 누르면 1씩 증가하며 새로운 메세지 박스가 뜨게 된다.

프로세스를 죽이고 디버거로 열어본다.



디버거로 열어보니 오류 메시지가 뜬다. 안티 디버깅 기법이 적용 되어 있는 것 같다.

```
0040E961    FF15 20D34700     call dword ptr ds:[<&IsDebuggerPresent>]
0040E967    85C0              test eax,eax                              eax:"カカ灘灘"
0040E969    0F85 6F4F0200     jne <6_patched.sub_4338DE>
0040E96F    884424 0F         mov byte ptr ss:[esp+F],al
```

jne구문을 je로 바꾸어 간단하게 우회하고 패치해준다.

패치한 파일을 디버거로 열어보면, 그냥 실행 했을 때와 같이 1씩 늘어나는 메시지 박스가 뜨게 된다.

```
004338EC call dword ptr ds:[<&MessageBoxA>]    <user32.MessageBoxA>
00438AD1 call dword ptr ds:[<&MessageBoxW>]    <user32.MessageBoxW>
00439B53 call dword ptr ds:[<&MessageBoxW>]    <user32.MessageBoxW>
00439C65 call dword ptr ds:[<&MessageBoxW>]    <user32.MessageBoxW>
0043B2E5 call dword ptr ds:[<&MessageBoxW>]    <user32.MessageBoxW>
00444DB8 call dword ptr ds:[<&MessageBoxW>]    <user32.MessageBoxW>
004536DB call dword ptr ds:[<&MessageBoxW>]    <user32.MessageBoxW>
0045390B call dword ptr ds:[<&MessageBoxW>]    <user32.MessageBoxW>
0045E071 call dword ptr ds:[<&MessageBoxW>]    <user32.MessageBoxW>
```

모든 메시지 박스에 브레이크 포인트를 걸고 확인해본다.

메시지박스에 해당하는 구문을 찾았다.

```
0045E071    FF15 9CD64700     call dword ptr ds:[<&MessageBoxW>]
0045E077    8B7424 4C         mov esi,dword ptr ss:[esp+4C]
0045E07B    8BF8              mov edi,eax
0045E07D    E8 DEDCFAFF       call 6_patched.40BD60
0045E082    8D4C24 20         lea ecx,dword ptr ss:[esp+20]             [esp+20]:L"27"
0045E086    893E              mov dword ptr ds:[esi],edi
0045E088    C746 08 01000000  mov dword ptr ds:[esi+8],1
```

```
dword ptr ss:[esp+20]=[008AF8B8]=03F100E0
```

esp+20 주소에는 증가하는 수가 담겨져 있는 것 같다.

esp+20을 관찰하고 있으면, 언제 또 다른 이벤트가 실행 되는지 알 수 있을 것이다.

```
03F100E0   31 00 00 00 0D F0 AD BA 0D F0 AD BA 0D F0 AD BA  1...ð.°.ð.°.ð.°
```

1이 들어있다.

다음 구문을 F8로 한줄씩 실행시켜보며 확인 하였는데, 굉장히 많은 구문을 거쳐 다음 수를 출력 하였다.

다른 방법으로 찾아보기로 하였다.

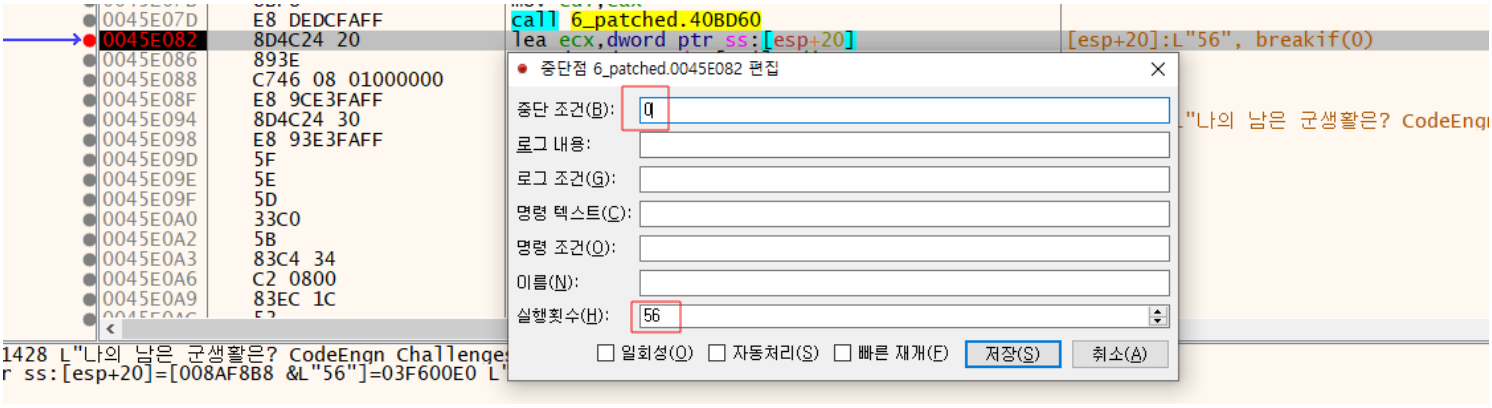
메시지 박스를 실행 할 때마다 확인을 눌러서 다음 구문으로 가면, 시간이 오래 걸리기 때문에 메시지 박스 부분을 없애주겠다.

```
0045E067   EB 0E          jmp 6_patched.45E077
0045E069   8B4C24 30      mov ecx,dword ptr ss:[esp+30]   [esp+30]:L"나의 남
0045E06D   90             nop
0045E06E   90             nop
0045E06F   90             nop
0045E070   90             nop
0045E071   90             nop
0045E072   15 9CD64700    adc eax,<6_patched.&MessageBoxW>
0045E077   8B7424 4C      mov esi,dword ptr ss:[esp+4C]
0045E07B   8BF8           mov edi,eax
0045E07D   E8 DEDCFAFF    call 6_patched.40BD60
0045E082   8D4C24 20      lea ecx,dword ptr ss:[esp+20]   [esp+20]:L"28"
```

메시지 박스를 없애주고 F9로 다른 이벤트가 언제 실행 되는지 확인해보면,

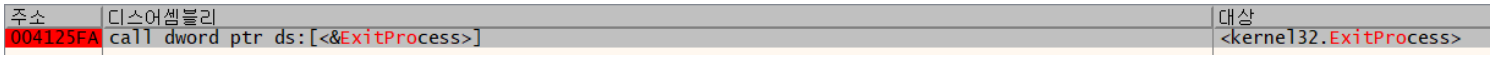
다른 이벤트 없이 어느순간 그냥 프로그램이 종료 되어 버린다.

더욱 편하게 답을 찾기 위해 다른 방법을 생각 했는데, MessageBox가 호출되는 횟수는 메시지 박스에 보여지는 수와 동일 할 것이다.

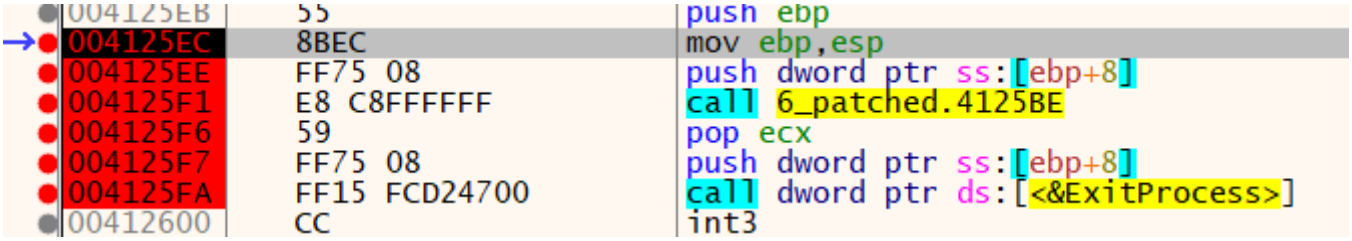


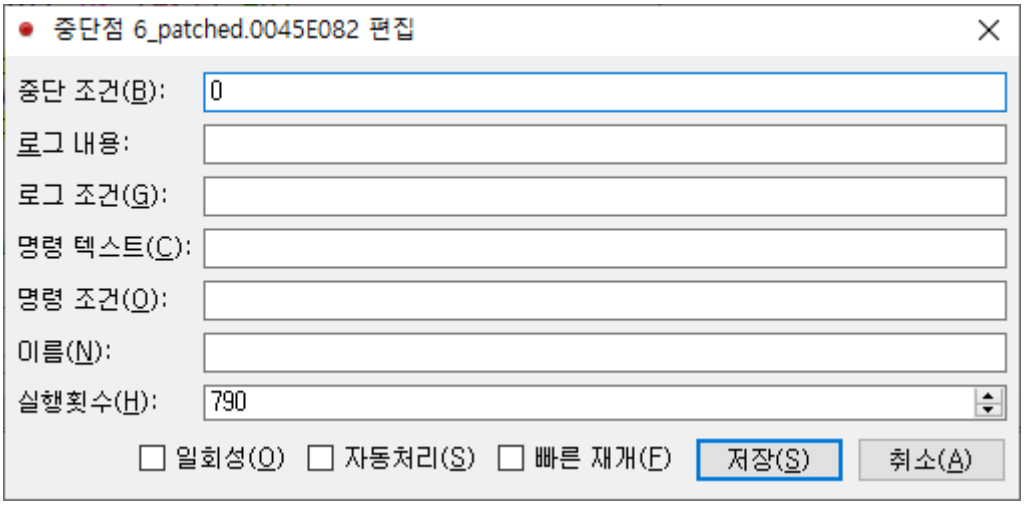
nop으로 처리한 메시지 박스 아래에 브레이크 포인트에 옵션을 주고, 중단 조건이 0이면 브레이크하지 않는다는 것이다 대신 카운트는 올라간다.

프로그램 종료 직전에 브레이크 포인트를 걸어주고 이 함수를 몇번 호출 했는지 알아보자.



종료되는 함수에 브레이크 포인트를 걸어준 다음 F9로 확인해보면,





messagebox구문쪽은 총 790번 실행 되었다. 군 생활이 790일 남은 것일까? MD5로 바꾸어 인증해본다.

2DACE78F80BC92E6D7493423D729448E

정답!