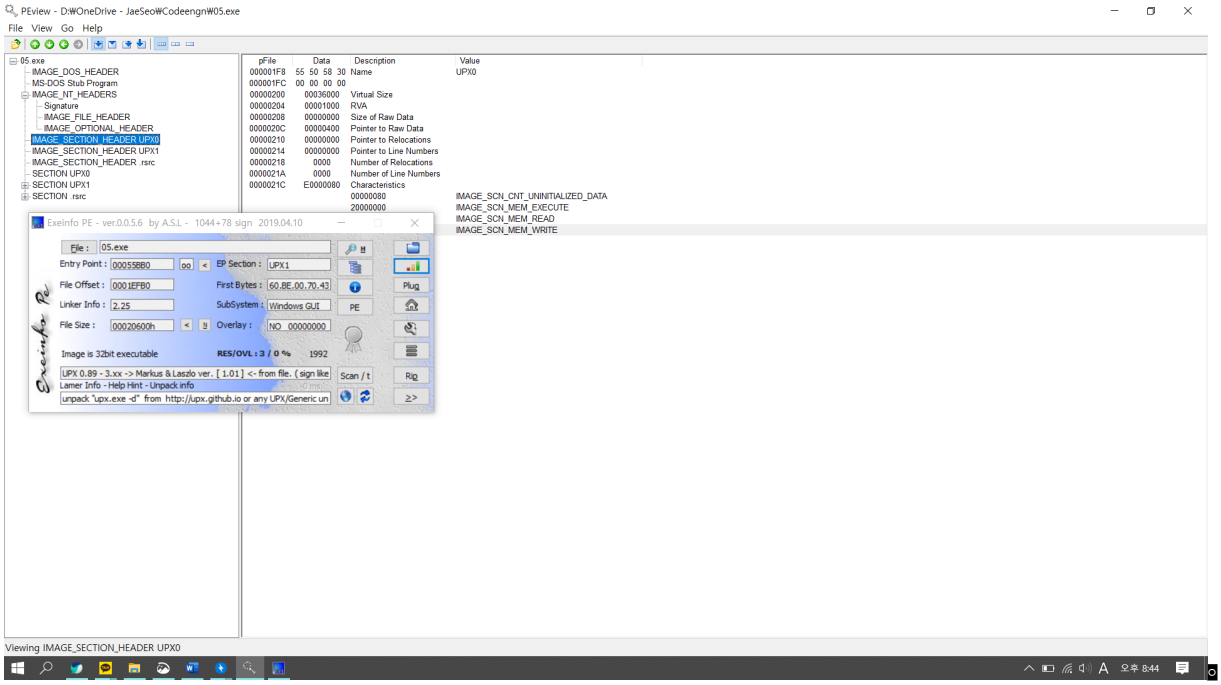
05.exe - 이 프로그램의 등록키는 무엇인가

프로그램을 실행하면 등록키를 입력하는 inputbox와 name을 입력하는 inputbox가 있고 이것을 이용하여 등록을 진행 하는 프로그램으로 보인다.

이제 프로그램 분석을 시작 한다.

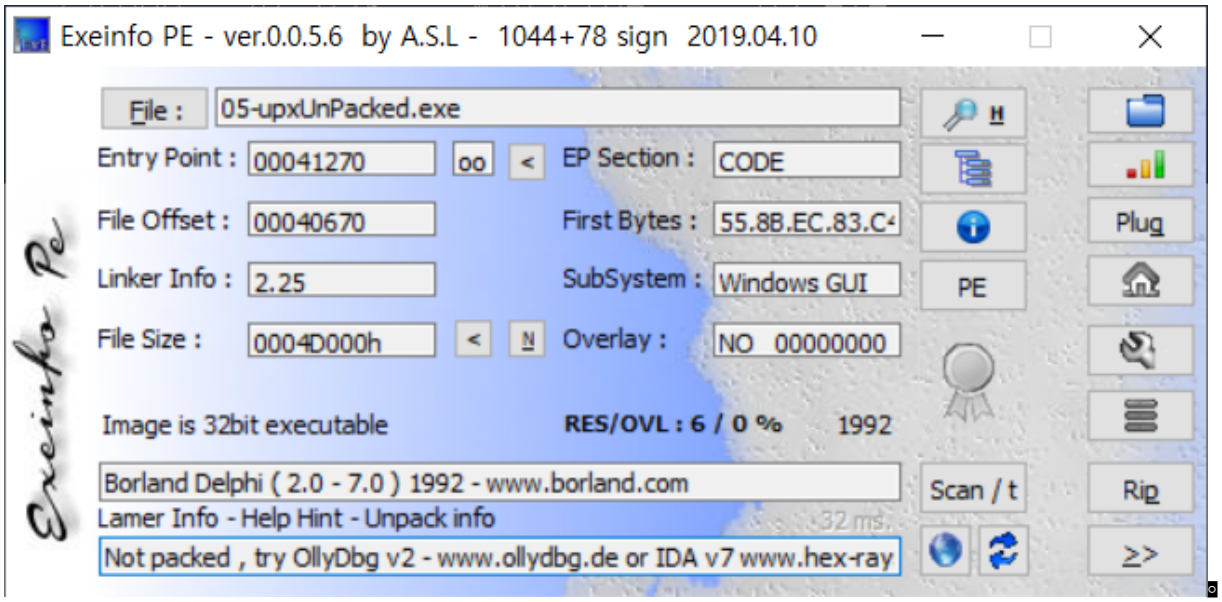
맨 처음 PE분석을 통해 패킹이 되어있는지 확인을 해보는 과정을 진행 해본다.



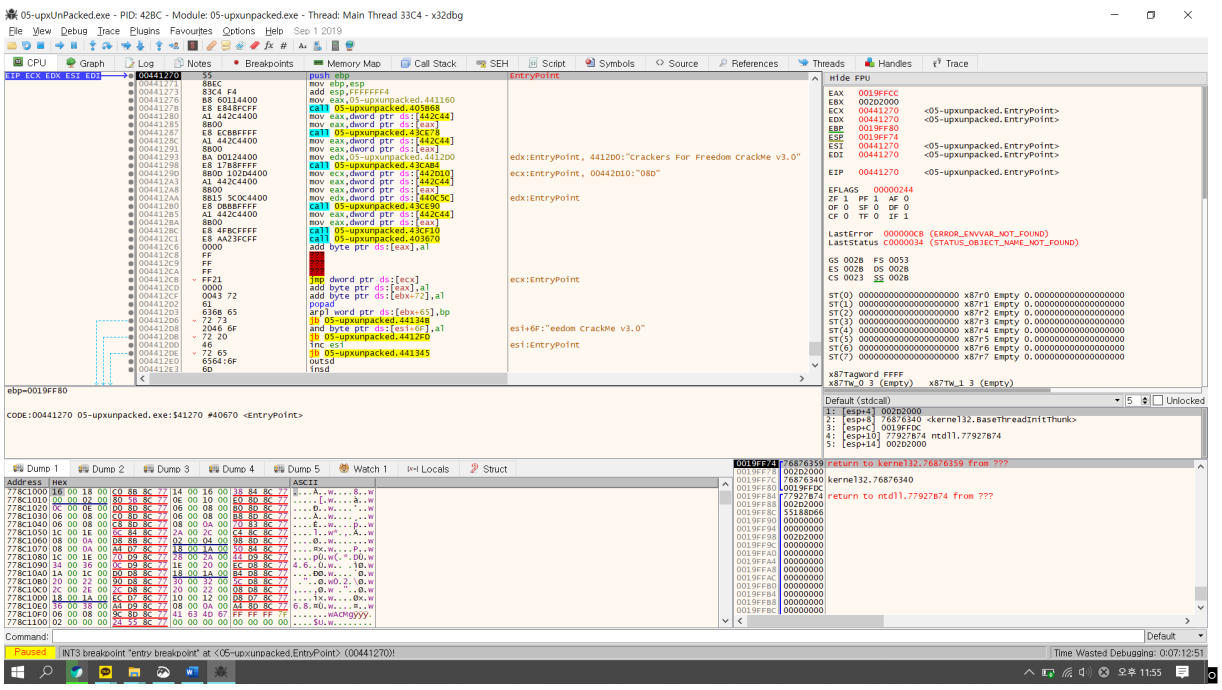
PEview와 Exeinfo PE 툴을 통해 이프로그램은 UPX로 패킹이 되어있는걸 확인 할수 있다.

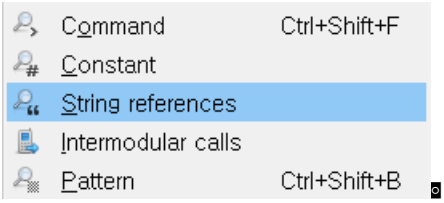
UPX를 사용하여 언패킹을 진행해본다.

```
PS D:₩> .₩upx.exe -d "C:₩Users₩kimja₩Documents₩UPX unpacking₩05.exe"
                       Ultimate Packer for eXecutables
                          Copyright (C) 1996 - 2018
UPX 3.95w       Markus Oberhumer, Laszlo Molnar & John Reiser   Aug 26th 2018

        File size         Ratio      Format      Name
   --------------------   ------   -----------   -----------
    315392 <-     132608   42.05%    win32/pe     05.exe

Unpacked 1 file.
PS D:₩OneDrive - JaeSeo₩KShield Jr₩tools₩Reversing₩upx-3.95-win64>
```

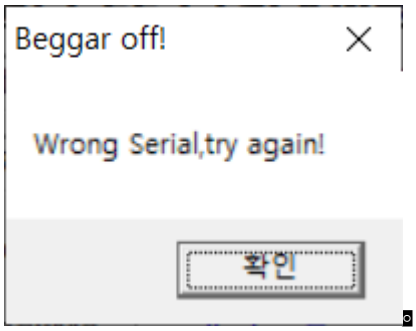


ExeinfoPE를 통해 Unpack이 된 것을 볼수 있다. 이제 X32DBG로 분석을 해본다.

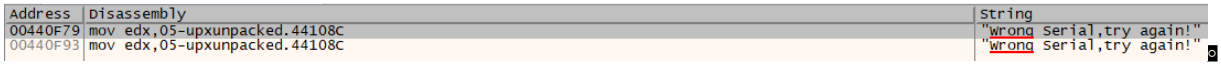




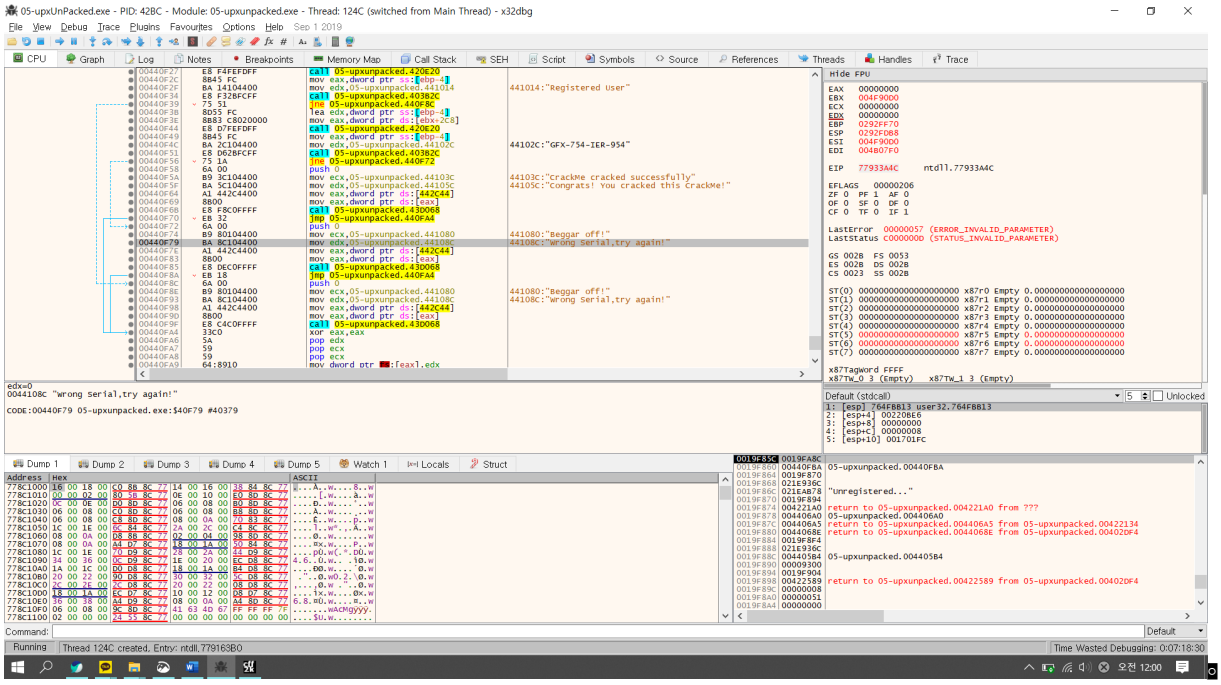
String references 검색으로 통해 등록키로 의심이 되는 값 또는 인증관련 메시지와 관련된 값을 찾는다.



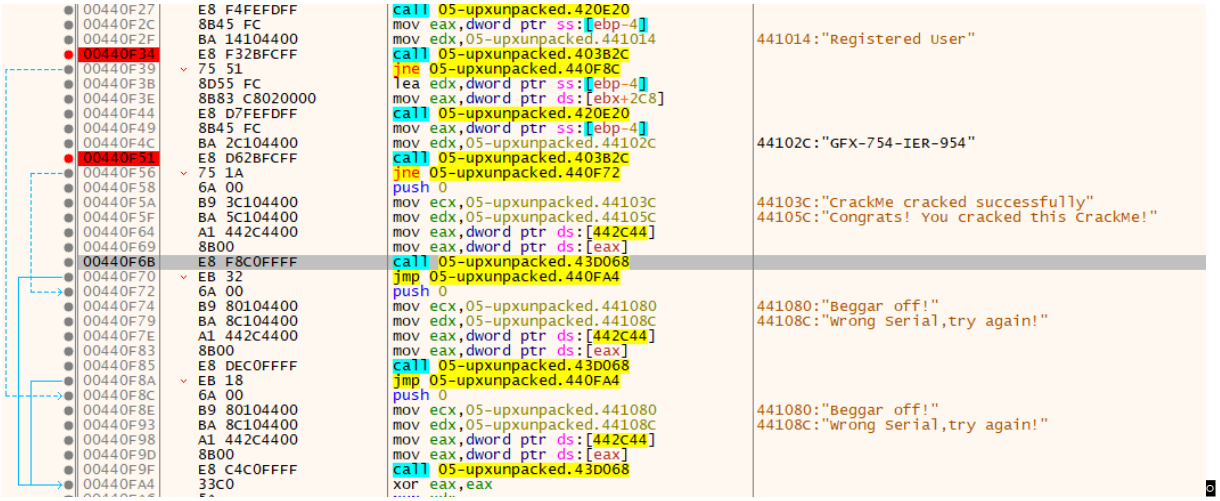
잘못된 입력 값을 입력 했을 때 뜨는 키워드를 통해 검색을 해본다.



그 결과 이러한 값을 찾을수 있는데 들어가서 분석을 해본다.



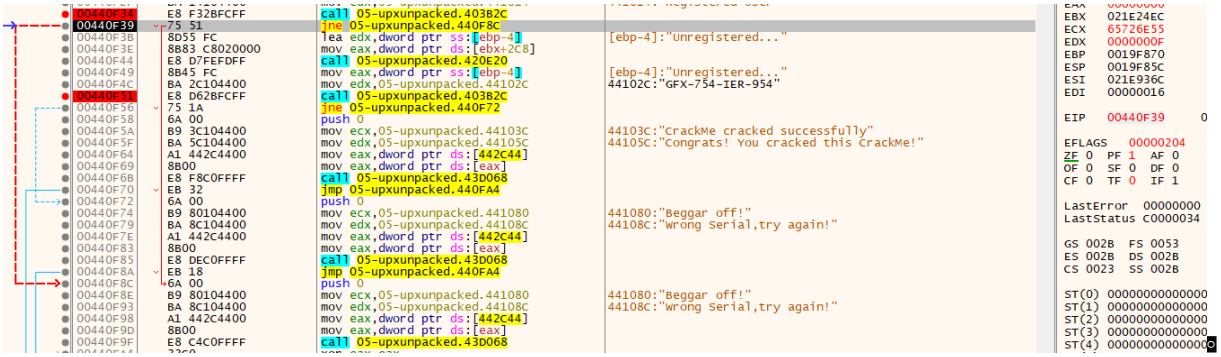
바로 주변에 인증을 성공 했을 때 뜨는 문자열값과 인증키로 의심되는 값을 발견했다



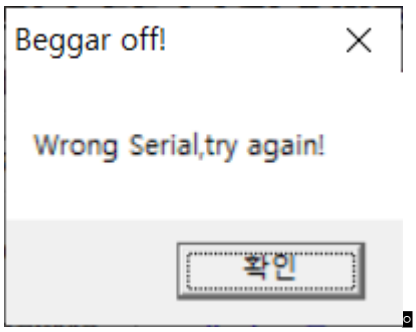
Call를 하고 jne를 하는 부분앞에 Break Point를 설정 하고 입력값을 넣어 작동하는지에 대해 살펴본다.

첫번째에 BreakPoint에서 그전에 입력을 했던 name 값을 eax에 가지고 mov를 통해 name으로 추정되는 문자열을 EDX로 가 가져오는 것 알수 있다.

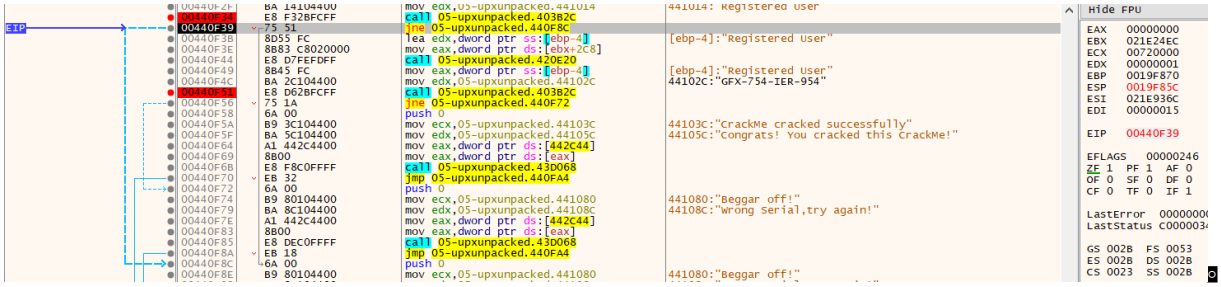
이때 진행을 해서 ZF 플래그가 어떻게 변화하는지를 본다.



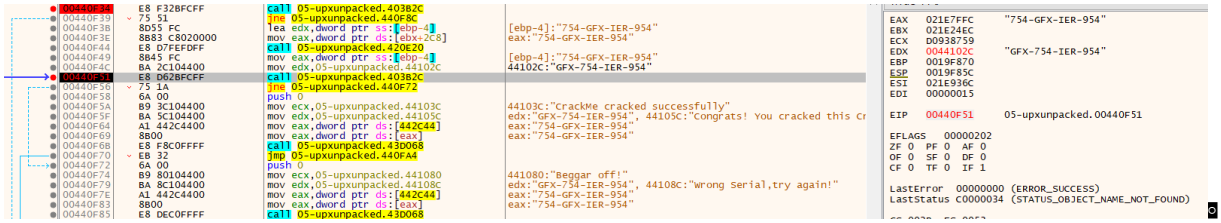
위와 같이 ZF 플래그가 0으로 선언되어 jne를 통해 틀렸다는 문자열이 있는곳으로 점프한는 것을 볼수 있고 또한 계속 실행을 해보면 프로그램에서도 틀렸다는 문자열이 뜨는 것을 볼수가 있다.



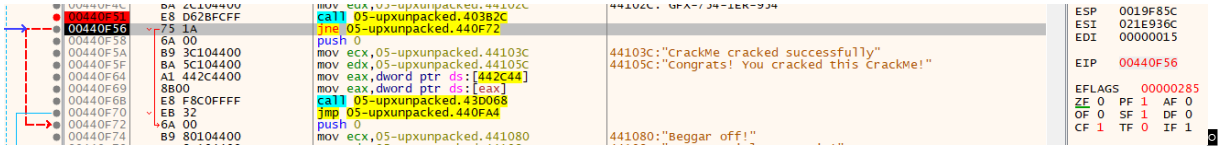
이제 name으로 추정되는 값을 넣어서 정상적으로 지나가는지 확인을 해본다.



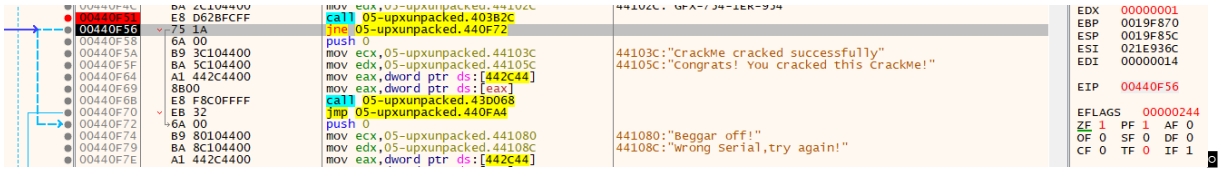
이번에는 ZF 플래그가 1로 되어 jne가 작동하지 않고 넘어가는 것을 볼수 있다.



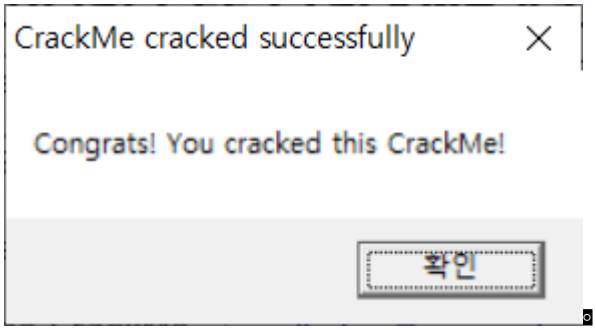
2번째 Break Point에서는 입력한 등록키를 eax로 불러오고 edx에 등록키로 의심이되는 키값을 가져와 call를 하고 있는 모습을 볼수 있다. 이때 그대로 실행을 해보고 ZF 플래그를 관찰 해본다.



이때 ZF 플래그는 0으로 변하여 오류 메시지를 출력하는 곳으로 점프되는 것을 볼수 있다. 이번에는 등록키로 의심되는 키값을 입력해서 다시 시도 하여 본다.



이번에는 ZF 플래그가 1로 되어 JMP를 하지 않고 넘어가는 모습과 성공적으로 Crack를 성공했다는 문자를 볼수 있다.



정답: GFX-754-IER-954