CodeEngn Basic RCE

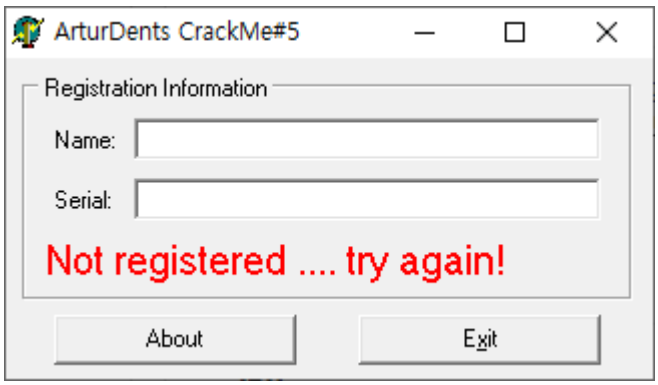
10. Level 10

Basic RCE L10

OEP를 구한 후 '등록성공' 으로 가는 분기점의 OPCODE를 구하시오.
정답인증은 OEP + OPCODE
EX) 00400000EB03

— Author: ArturDents
— File Password: codeengn

* OPCODE : 프로세서가 수행할 연산과 실행할 동작을 정의하는 코드. Hex dump 칸에 있는 코드들이 OPCODE.

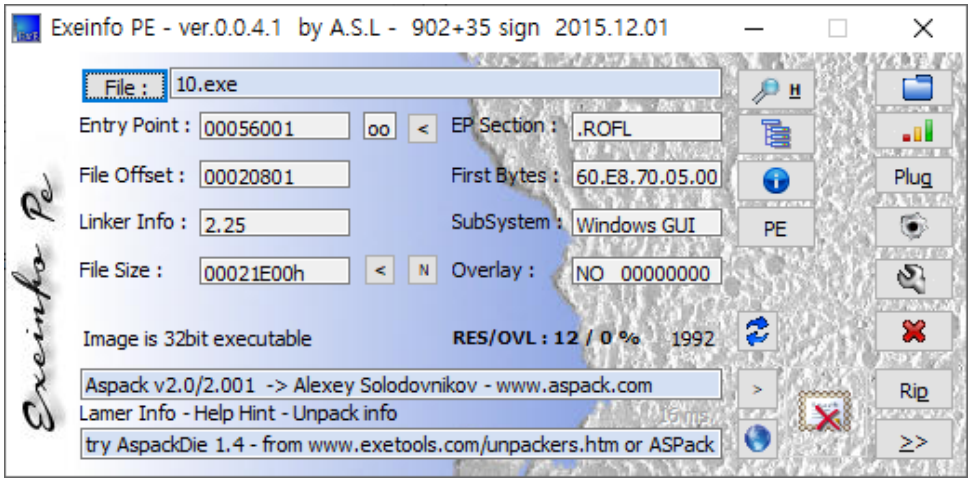


프로그램 실행 화면이다.

이름과 시리얼번호를 입력하여 인증을 받는 프로그램인 것 같다.

입력칸 안에 입력을 해보려 했지만 입력이 되지 않는다.

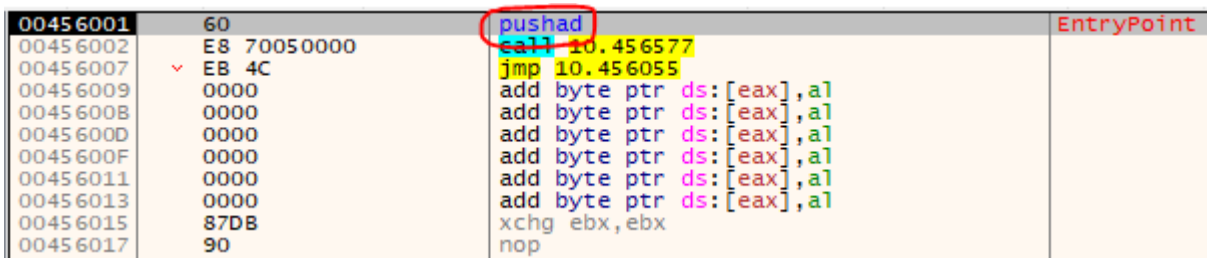
exeinfo로 프로그램 정보를 봐보자.



Aspack 패킹이 되어있다.

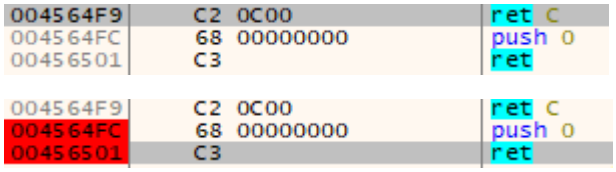
Aspack은 RETN 0C를 찾아 BP를 걸어 실행 후 OEP로 가는 방법으로 언패킹이 가능하다.

* x64dbg는 "ret 0xC" 로 명령어를 찾을 수 있다.

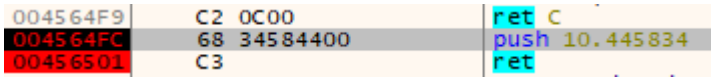


* 디버거를 이용해서 해당파일을 로딩하면 첫 구문에서 pushad를 볼 수 있다. pushad 명령어는 범용 레지스터들에 저장된 값들을 스택에 저장하는 명령임. 패킹되어 있는 대부분의 파일이 첫 구문에서 pushad를 볼 수 있음. 현재 레지스터 값을 스택에 저장하고, 원본코드를 특정 메모리상에 복구시킨 다음 popad를 이용하여 원래의 레지스터값을 복구하여 사용하기 위해 씀.

x64dbg로 언패킹을 해보자.

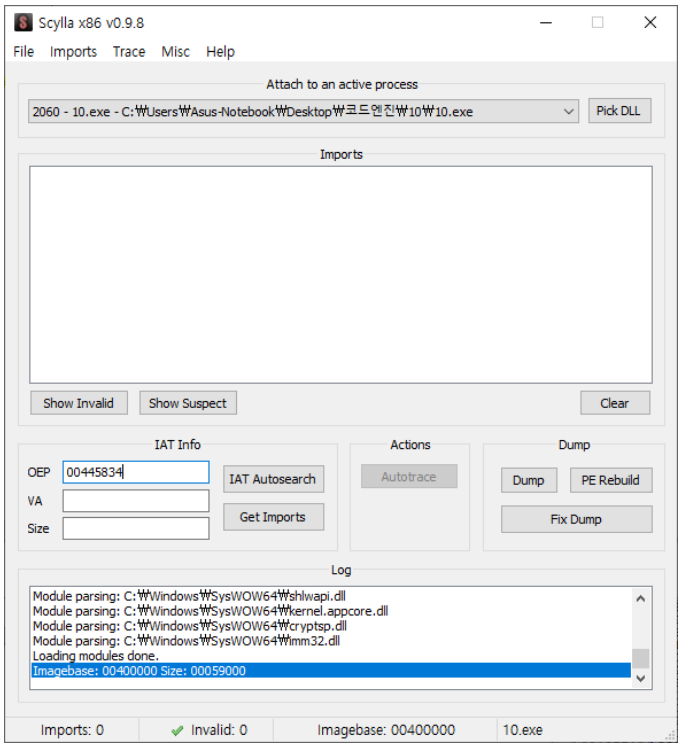


ret 0xC 명령어를 찾은 후 밑 코드 두 줄에 BP를 걸어준다.



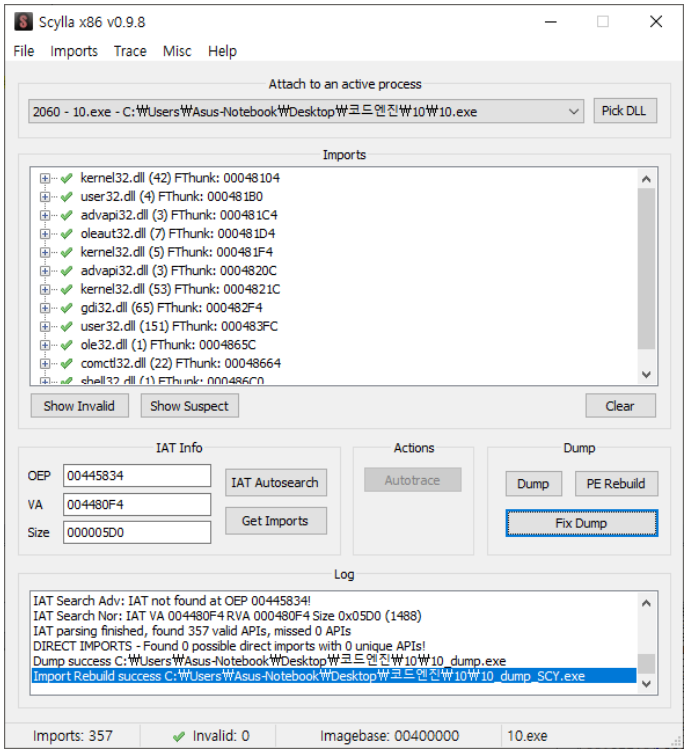
실행하면 OEP 주소로 변경되고 아래 ret을 통해서 OEP 코드가 있는 곳으로 이동.

```
00445834    55          push ebp
00445835    8BEC        mov ebp,esp
00445837    83C4 F4     add esp,FFFFFFF4
0044583A    B8 F4564400 mov eax,10.4456F4
0044583F    E8 0408FCFF call 10.406048
00445844    A1 6C6C4400 mov eax,dword ptr ds:[446C6C]
00445849    8B00        mov eax,dword ptr ds:[eax]
0044584B    E8 F0CCFFFF call 10.442540
00445850    8B0D 386D4400 mov ecx,dword ptr ds:[446D38]   ecx:EntryPoint, 00446D38:"LxD"
00445856    A1 6C6C4400 mov eax,dword ptr ds:[446C6C]
0044585B    8B00        mov eax,dword ptr ds:[eax]
0044585D    8B15 88514400 mov edx,dword ptr ds:[445188]   edx:EntryPoint
00445863    E8 F0CCFFFF call 10.442558
00445868    8B0D 586D4400 mov ecx,dword ptr ds:[446D58]   ecx:EntryPoint, 00446D58:"DxD"
0044586E    A1 6C6C4400 mov eax,dword ptr ds:[446C6C]
00445873    8B00        mov eax,dword ptr ds:[eax]
00445875    8B15 104F4400 mov edx,dword ptr ds:[444F10]   edx:EntryPoint, 00444F10:"\\OD"
0044587B    E8 D8CCFFFF call 10.442558
00445880    A1 6C6C4400 mov eax,dword ptr ds:[446C6C]
00445885    8B00        mov eax,dword ptr ds:[eax]
00445887    E8 4CCDFFFF call 10.4425D8
0044588C    E8 17DFFBFF call 10.4037A8
```



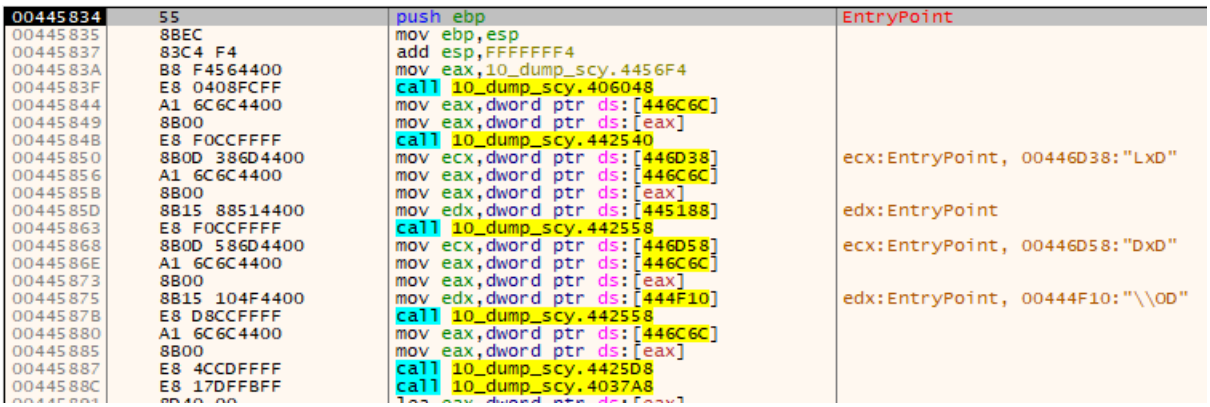
이동 후 "플러그인 > Scylla"

덤프를 저장한다.



EIP와 OEP가 같은지 확인 > IAT Autosearch > Get Imports > Dump > Fix Dump

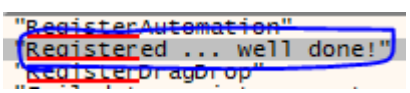
언패킹 완료



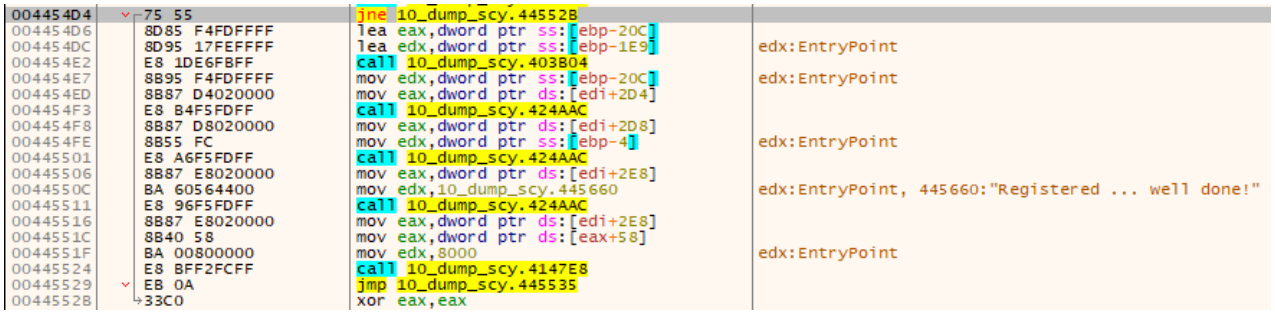
언패킹한 파일을 x64dbg로 깐 상황이다.

EntryPoint가 잘잡히는 것을 확인할 수 있다.

이제 문자열을 검색 후 계속 분석해보자.



"Registered... well done!" 이라는 긍정적 문자열이 보인다.



"Registered... well done!" 문자열이 포함되어있는 점프문은 jne 10_dump_scy.44552B 이다.