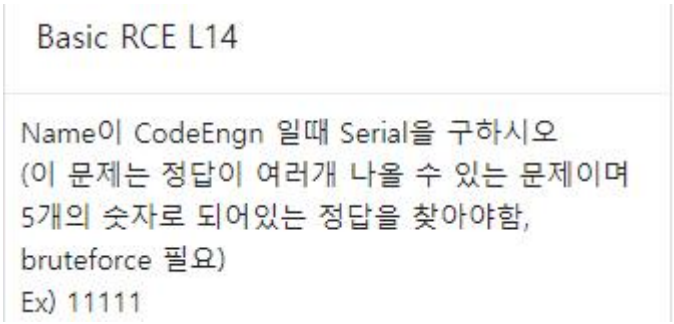
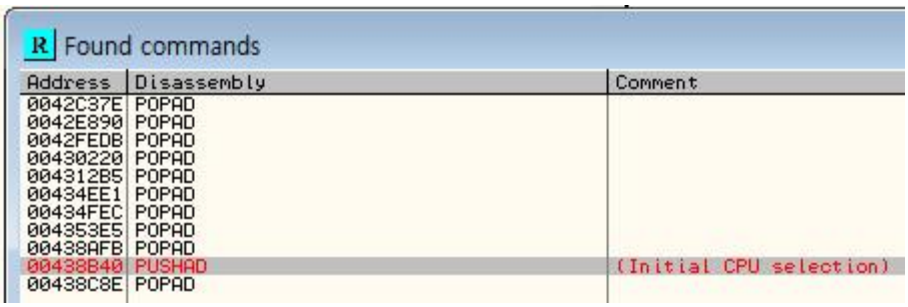
Codeengn Basic RCE 14.exe



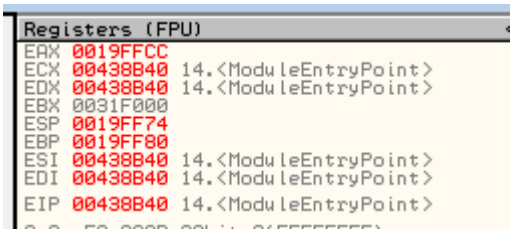
시작이 PUSHAD , 시작할 때 패킹되었다는 알림. 실행압축되었다고 생각하고 압축 풀어보자.

POPAD를 찾아보자.

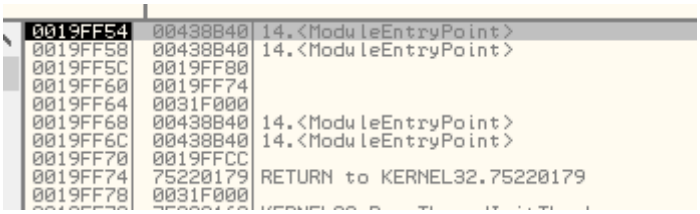


너무 많네. 그러면 PUSHAD에서 스택에 PUSH한 부분에 하드웨어 브레이크 포인트를 걸어보자. (내가 PUSHAD한 레지스터를 다시 POP할 때 해당 부분에 접근하게 되므로 위에 많은 POPAD 중에서 진짜 뭐가 중요한 것인지 알게 됨.)

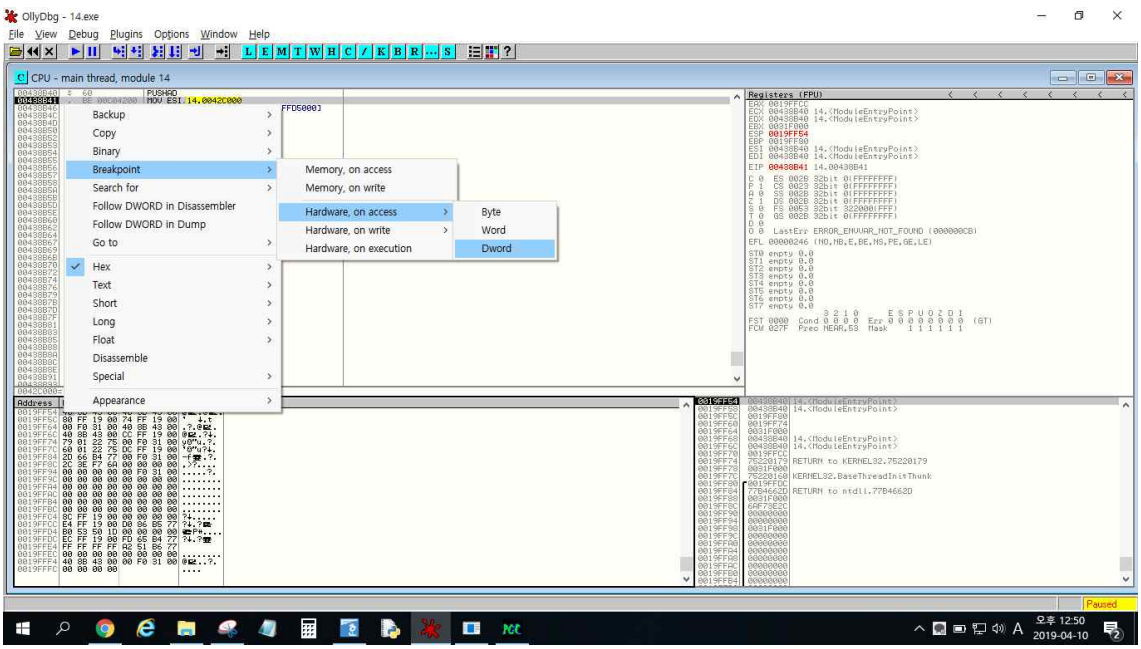
PUSHAD 하기 전의 레지스터.

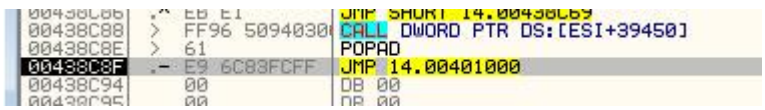


PUSHAD하고 난 후의 스택. EAX부터 상위주소에 차곡차곡 쌓여서 EDI가 가장 낮은 주소로 쌓인 것을 볼 수 있다. 스택은 상위주소부터 쌓여서 쌓이면 쌓일수록 낮은 주소로 내려가므로 만약에 POPAD를 하게 되면 가장 먼저 빠지는 레지스터는 19FF54에 있는 레지스터이다. 여기다가 브레이크포인트를 걸자.

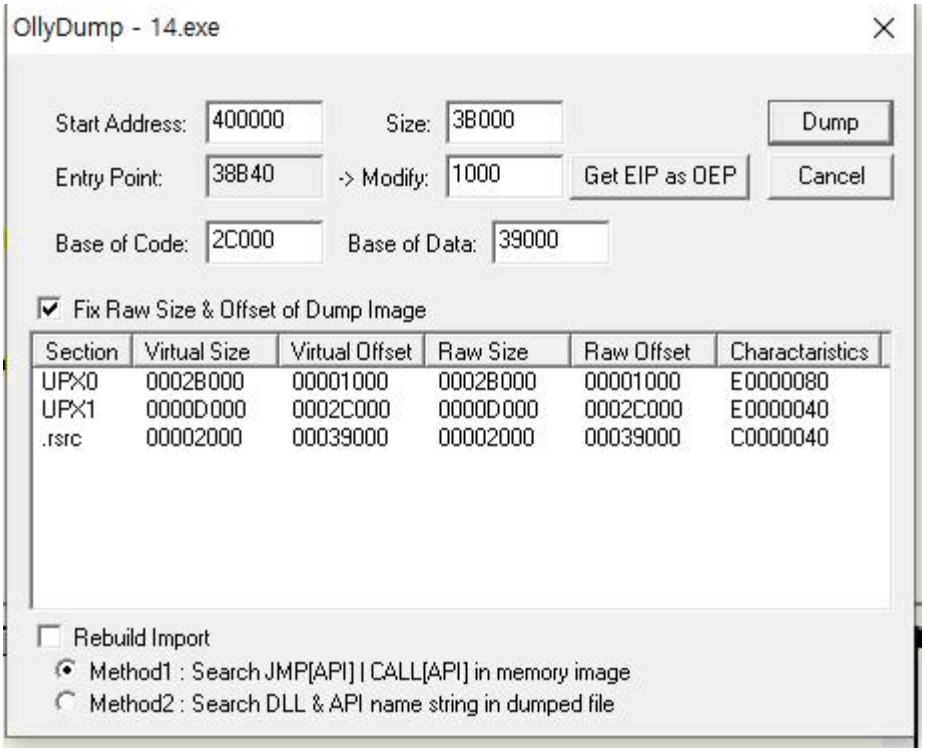


아래는 하드웨어 브레이크를 거는 모습.

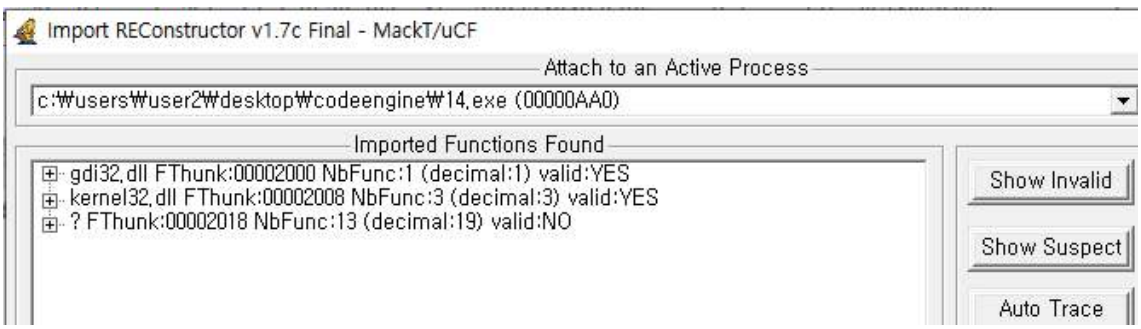




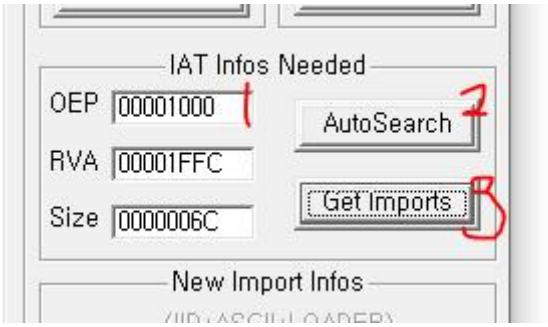
F9로 실행하다보면, 여기서 멈춘다. 실행압축을 다 풀어낸 이후에 JMP로 OEP로 점프하는 모습! 그렇다면 401000에서 덤프를 떠서 원래 파일을 만들어보자.



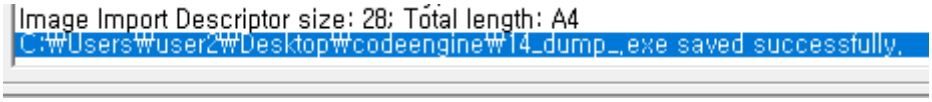
올리덤프로도 리빌드 할 수 있긴한데, 왠지 모르지만 다들 그렇게 안하는 듯하다. 여튼 이렇게 설정하고 저장. 14_dump.exe를 저장하고 실행하면, dump 플러그인이 완전히 IAT(Import Address table)까지 바르게 가져오지 못해서 이런 에러가 발생한다. 그래서 ImportREC 라는 툴을 이용해서 원래 프로그램의 정상적인 IAT를 가져와 dump를 떠놓은 프로그램에 덮어씌우기로 한다.



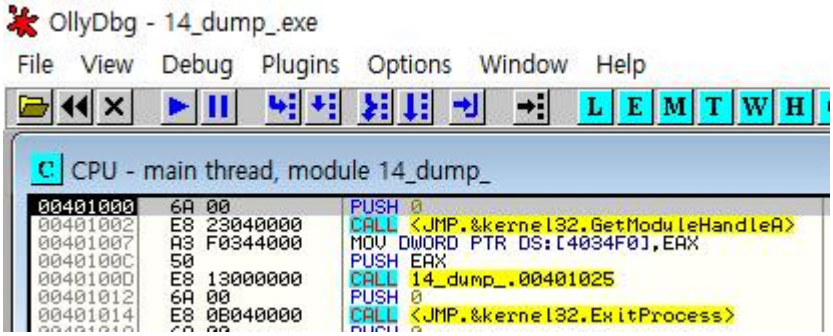
우선 상단 바에서 14.exe(디버거랑 별개로 새로 실행)를 선택하고,



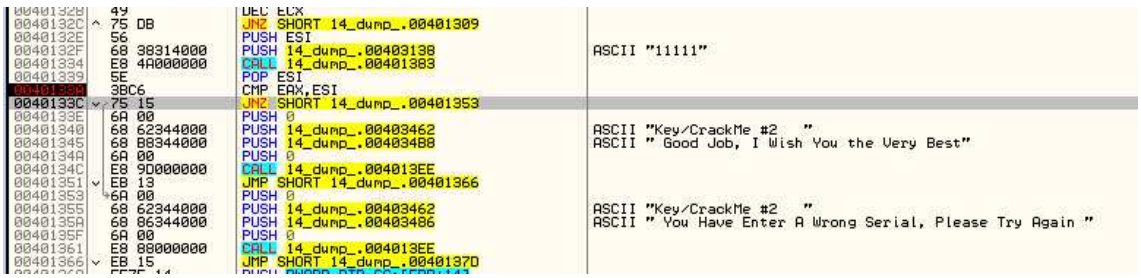
1에 실행 압축하고 JMP 했던 401000에서 imagebase를 제외한 1000을 입력하고 2, 3을 순서대로 누르자! 그리고 맨 밑에 Fix Dump를 눌러 아까 떠놓은 덤프를 선택하면 끝!



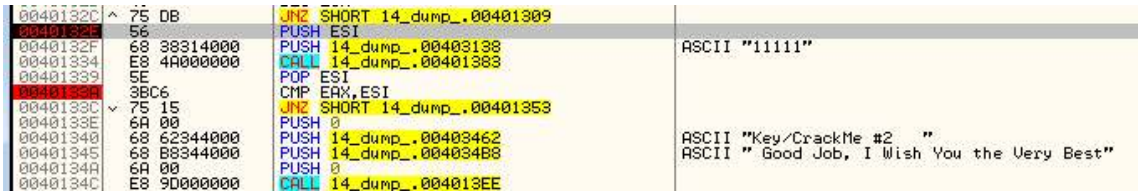
잘 실행된다.



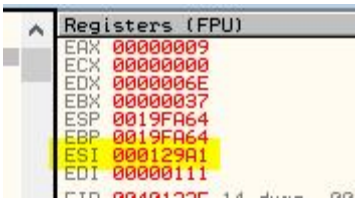
이제 문제를 본격적으로 풀어보자.



성공 메시지와 실패 메시지가 있는데, JNZ에서 비교를 해서 같지 않으면 실패가 된다. 비밀번호로 11111을 입력해서 CALL 위에 인자로 11111이 들어가 있는 것으로 보인다. 그렇다면 그 위에 ESI는 내가 입력해야할 패스워드가 아닐까 추측해보자.

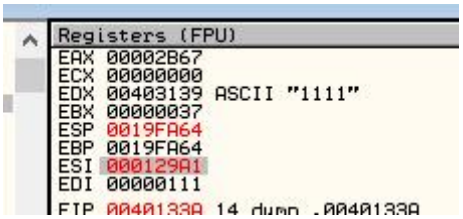


PUSH ESI에 bp 설정.

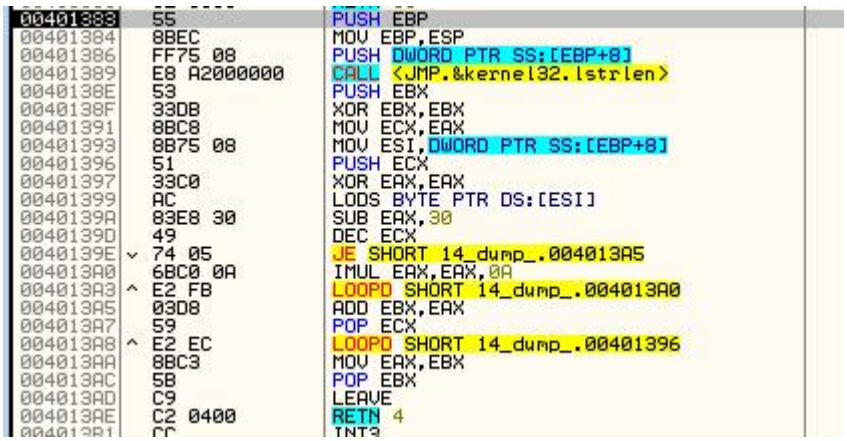


129A1과 11111을 단순히 비교해서 패스워드를 129A1이라고 추측했으나, 빗나갔다.

다음으로 CMP EAX, ESI 부분에서 다양한 비밀번호를 넣어도 ESI는 129A1으로 고정이라는 사실을 발견했다.



그러면 내가 입력한 문자열에 기반해서 EAX를 만든다는 것인데, EAX를 만드는 그 CALL 부분을 살펴보자.



실제 숫자를 대입하여 계속 실행하면서 경과를 살펴보자.
나는 '12345'를 대입했는데 '1'은 '₩x31'이고 0x0040139A에서 30을 빼주니 그대로 1이 된다. 이하의 구문들에 의해 1에다가 0x0A(10)을 4번 곱해주게 되고 10000(십진수 기준)이 된다.
'2' 는 '₩x32'이고 30을 빼주니 그대로 2가 되고 2에다가 10을 3번 곱해줘서 2000이 된다.

이런 식으로 10000 + 2000 + 300 + 40 + 5 = 12345 가 되어 내가 입력한 숫자를 그대로 EAX로 출력하고 있음을 알 수 있다. (물론 십진수가 아니라 16진수로)

따라서 ESI에 고정된 129A1을 10진수로 바꾸어 패스워드로 입력해보자.


KEY/CRACK-ME #2 - PENGAIY
CodeEngn
76193
CHECK
ABOUT
oc_H
Key/CrackMe #2
Good Job, I Wish You the Very Best
확인

성공!