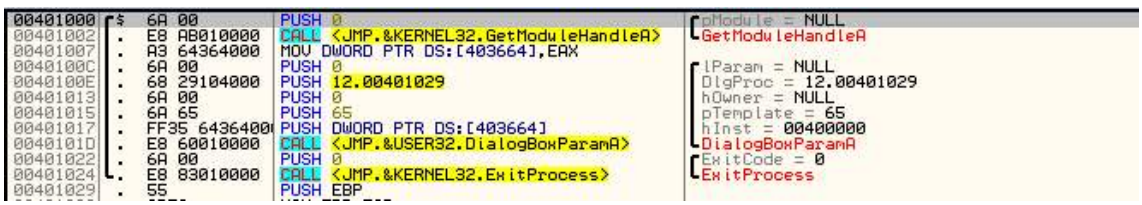
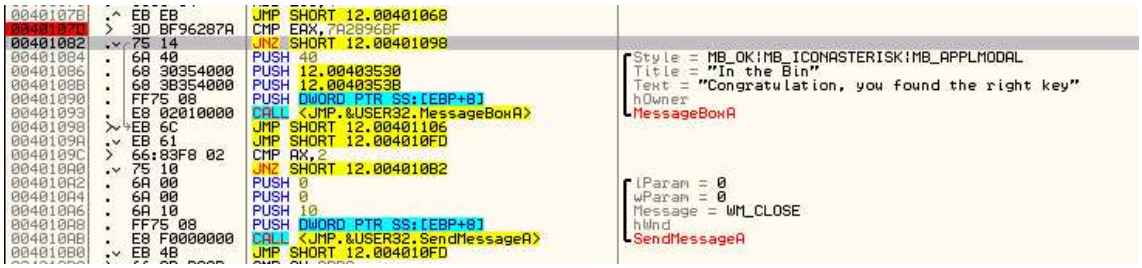
codeengn Basic RCE 12.

Basic RCE L12

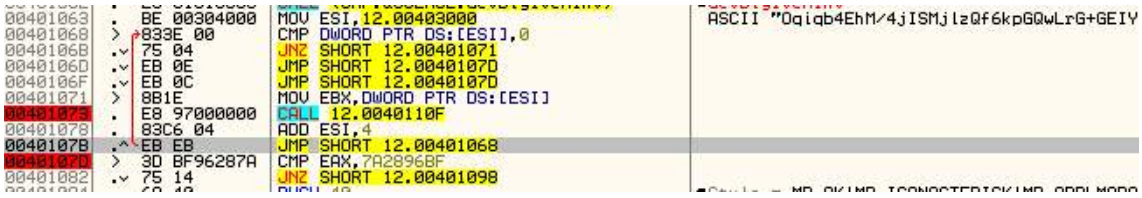
Key를 구한 후 입력하게 되면 성공메시지를 볼 수 있다
이때 성공메시지 대신 Key 값이 MessageBox에 출력 되도록 하려면 파일을 HexEdit로 오픈 한 다음 0x???? ~ 0x???? 영역에 Key 값을 overwrite 하면 된다.
문제 : Key값과 + 주소영역을 찾으시오
Ex) 7777777????????



ep가 4010000이고 루틴이 상당히 단순해 보인다. DialogBoxparamA라는 함수에서 휴지통 아이콘이 나오는 프로그램을 실행하고, 키를 입력할 수 있게 된다. 바로 밑에 그 코드가 있다.

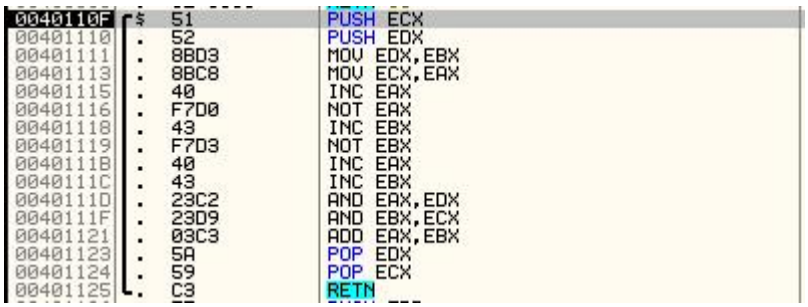


JNZ를 기점으로 위는 성공, 아래는 실패로 보인다. JNZ 앞에 CMP에서는 EAX를 무언가와 비교하고 있다. EAX는 언제 만들어지는지 살펴보자.



반복문이 존재한다. 우선 ESI에 매우 긴 문자를 넣고 이 ESI를 4씩 증가시켜가며 문자열이 끝날 때까지(00이 될 때 까지) 비교한다. 0이 되면 40107D로 점프하게 되고 아까 보았던 CMP를 수행한다. 이 과정에서 40110F에서는 하나의 함수가 계속 반복된다. 함수 안을 살

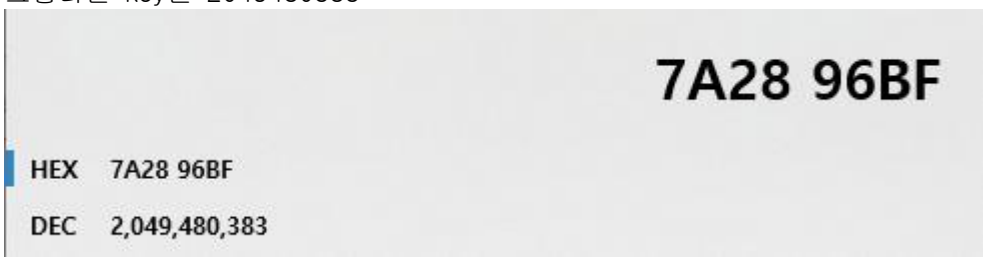
펴보자.



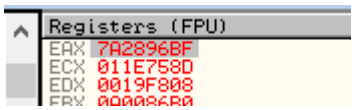
아무 쓰잘데기 없어보이는 괴상한 함수이다. 무언가의 계산을 통해 암호화하는 것처럼 보이기도 한다. 루틴에 들어가기 전에 이것저것 대입해서 분석할 결과, eax는 내가 입력한 정수(16진수로 바꿔서)이고, ebx에는 괴상하게 긴 문자열의 처음 4바이트가 저장된다.("0qiq")

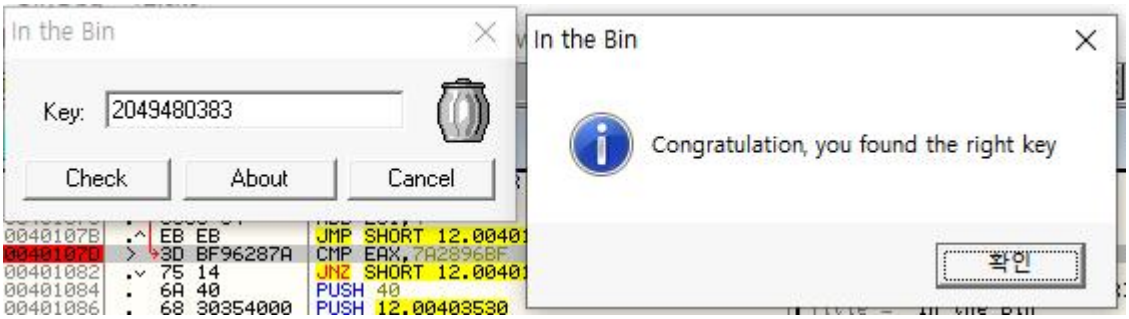
실제 함수를 작동시켜본 결과, 내가 입력한 처음의 정수 값은 암호화 과정에서 하나의 솔트로 작용하는 것 같고, 함수의 알고리즘은 마치 하나의 암호를 만드는 듯하다. 어떡하징.. 1부터 하나하나 대입하면서 차분법을 사용해보자.
1 입력했을 때 eax 1 (목표는 7a2896bf)
2 입력했을 때 eax 2
10 입력했을 때 eax A
64 입력했을 때 eax 40

??? 엄청 복잡해보였는데, 막상 그냥 십진수를 16진수로 바꿔줬을 뿐이다.
그렇다면 key는 2049480383

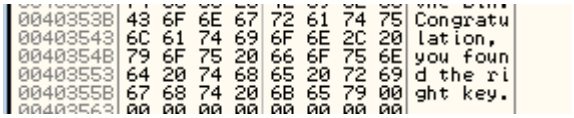


정확히 일치함

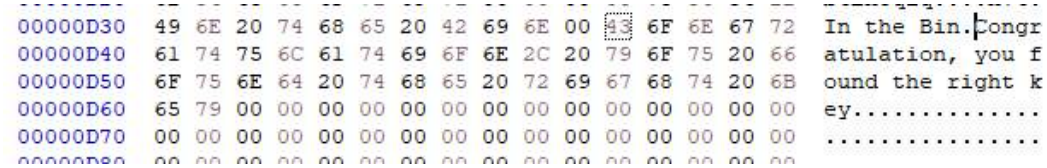




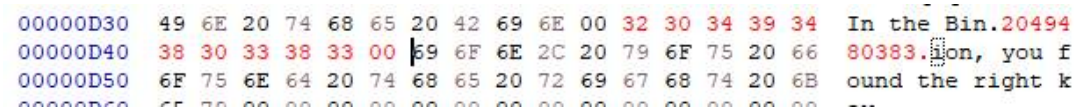
Key는 구했고, 이제 성공 메시지 대신에 key값이 메시지박스에 출력하도록 파일을 수정해 보자.



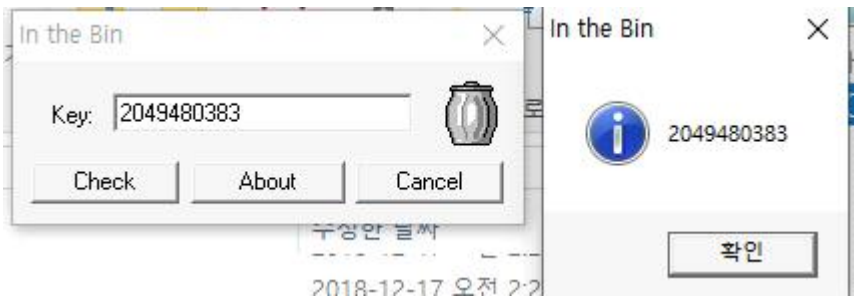
0x40353B 부분부터 Congratu... 로 시작하는 문자열이 있는 것을 확인할 수 있다.



hxd로 열었을 때도, 0x0D3B부터 문자열이 존재하는 것을 확인할 수 있다.



Key 값의 길이가 더 짧기 때문에 마지막에 Null값을 추가해서 파일을 수정하고 실행해보자.



성공! 주소값은 0x0D3B ~ 0x0D45