Crack me 문제 보고서

|  |  |
| --- | --- |
| 문제명 | Duelist’s Crack me 3 |
| 파일명 | due-cm3.exe |
| 풀이자 | 서치원(2011003155) |
| 풀이 기간 | 2016/12/26 ~ 2017/01/01 |
| 문제 내용 | check box 조합 중 알맞은 조합을 찾아라. |

문제 풀이:

문제 파일을 실행 시 창 하나가 뜨게됩니다.



여러 개의 check box가 있고 Check 버튼이 있어 클릭시 정답 여부를 체크합니다. 이 문제에서 해야할 일은 알맞은 check box의 조합을 찾는 것 입니다.



Check 버튼을 눌렀을 시의 루틴입니다. 초기에 ESI, EDX를 초기화 하고 주소 40215E, 402162를 초기화해 이 2 주소값을 사용함을 알 수 있습니다.

그 후 401127 ~ 401160 루프를 시작하는데 4020FE 주소부터 순서대로 값을 읽어옵니다.



4020FE 부터의 주소값입니다. 마지막이 4D이고 루프 조건에 4D를 읽을때 빠져나가는것이 있으므로 402110 주소까지 읽게됩니다.

주소값을 읽었으면 IsDigButtonChecked 메소드를 호출하는데 이 메소드의 역할은 ButtonID를 가진 버튼이 check 되었는지 확인하는 메소드입니다. 여기서 ButtonID를 ECX register로 전해주는데 어느 버튼이 실제 buttonID를 가지는지 알아낼 필요가 있습니다. 문제의 Tip에는 이를 알아내기 위해 resource editor를 사용하라고 되어있습니다.



resource hacker 툴을 이용해 각 button의 ID를 확인한 화면입니다. 해당 프로그램은 십진수로 표시하기 때문에 이를 hex code로 변환하고 이를 4020FE 주소값과 대응되는 표로 확인해보면



아래와 같이 나오게 됩니다. 즉, 3번 버튼을 체크하였다면 ECX가 0일때 체크됨을 확인할 것입니다. 여기서 체크된 버튼을 만나면 계산 루틴을 실행하는데 먼저 현재 index의 주소 값을 가져온 후, ESI가 1 증가되었으므로 다음 값을 가져온 후, 이 두 값을 곱하고 거기에 현재 ESI(index)를 곱해 저장된 값과 더해 새로 저장합니다. 즉, 3번 버튼이 체크되어 있었다면 ESI가 0일때 확인 되므로 0x16 \* 0x49 \* 1 이 더해집니다.

이 루프를 끝까지 처리한 후 계산된 값을 불러와서 검사를 하게 되는데 계산된 값에 0x4D를 곱한 후 0xF35466 과 비교함으로 체크되어 더해진 값들은 0xF35466 / 0x4D = 0x328FE(=207102) 입니다.

이를 만족하는 체크박스의 조합을 계산해 대입하면



이렇게 정답 화면이 출력되게 됩니다.