

Codeengn Challenges Advance RCE LEVEL3 풀이

Reverse2 L03 Start

Author : Vallani

Korea :

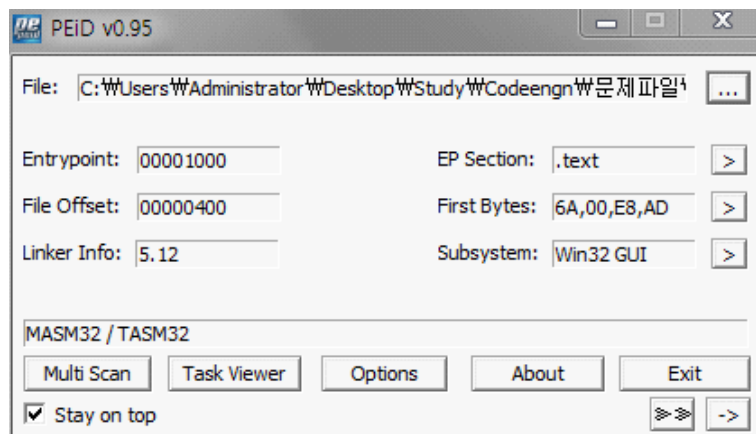
Name이 CodeEngn 일때 Serial은 무엇인가

English :

Find the Serial when the Name is CodeEngn

[Down](#)

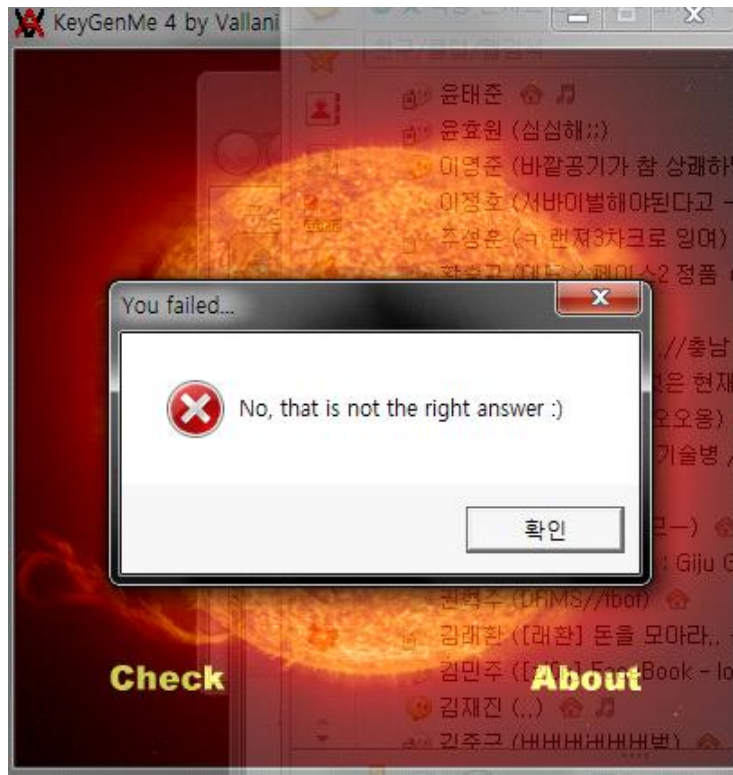
프로그램을 실행시켜보니 이름을 입력하고 그에 맞는 시리얼을 입력 하는 프로그램이었다.
PEID로 프로그램에 대한 정보를 확인해보니,



어셈블리로 코딩된 프로그램이다.

어셈블리로 코딩된 프로그램들은 대개 리버싱에 불필요한게 많이 포함이 안되어있어서 좋다 !

프로그램을 실행 시킨 후 NAME에 CodeEngn과 Serial에 123123을 넣은 후
Check 버튼을 눌렀는데,



다음과 같이 옳은 답이 아니라면서 버튼이벤트가 일어나야 프로그램이 활성화 되는 메시지가 나왔다.

나는 이 문제를 풀기위해 OLLY DBG의 Back to User mode를 사용했다.

Back to User 기능은 프로세스가 뭘했을때 (입력을 받거나 , 버튼 입력을 받을때 등) 이용하면 좋은 기능이다, 특정 이벤트가 일어나기전에(버튼 이벤트같은)Back to User 모드를 설정해주고 이벤트를 수행하면, CALL 명령이 일어난 직후의 위치로 이동 하는 기능이다.

사용 방법은 다음과 같다,

1. 프로그램을 OLLY로 실행이나 attach후 실행을 시킨 뒤
2. 프로그램을 정지된 상태로 만든다.
3. OLLY에서 일시 정지 버튼을 누른 후(F12)
4. Back to user mode(ALT + F9)를 설정해준다
5. 그리고 프로그램에서 이벤트를 발생시킨다. (예를 들어 예, 아니오 버튼 클릭 등)

이렇게 하면 OLLY가 표시하는 주소의 위치가 CALL명령이 일어난 직후로 이동이 된다.

실제로 프로그램에 NAME에 CodeEngn을 넣고 Serial에 123123을 넣은 후 확인 창에서 일어나는 버튼 이벤트를 이용해서 Back to User 기능을 쓰면

00401136	. 6A 00	PUSH 0	hOwner = NULL
00401138	. E8 B3020000	CALL <JMP,&user32.MessageBoxA>	MessageBoxA
0040113D	> E9 8F000000	JMP Reverse2,004011D1	
00401142	. B8 38324000	MOV EAX,Reverse2,00403238	ASCII "CodeEngn"
00401147	. A3 58324000	MOV DWORD PTR DS:[403258],EAX	
0040114C	. 6A 00	PUSH 0	
0040114E	. E8 24010000	CALL Reverse2,00401277	
00401153	. 6A 20	PUSH 20	Count = 20 (32,)
00401155	. 68 64324000	PUSH Reverse2,00403264	Buffer = Reverse2,00403264
0040115A	. 68 E0030000	PUSH 3ED	ControlID = 3ED (1005,)
0040115F	. FF75 08	PUSH DWORD PTR SS:[EBP+8]	hWnd
00401162	. E8 77020000	CALL <JMP,&user32.GetDlgItemTextA>	GetDlgItemTextA
00401167	. FF35 00304000	PUSH DWORD PTR DS:[403000]	<%u> = 0
0040116D	. 68 84304000	PUSH Reverse2,00403084	Format = "%u"
00401172	. 68 84324000	PUSH Reverse2,00403284	s = Reverse2,00403284
00401177	. E8 4A020000	CALL <JMP,&user32.wsprintfA>	wsprintfA
0040117C	. 83C4 0C	ADD ESP,0C	
0040117F	. 33C0	XOR EAX,EAX	
00401181	. A3 00304000	MOV DWORD PTR DS:[403000],EAX	
00401186	. 892D 5C324000	MOV DWORD PTR DS:[40325C],EBP	
0040118C	. 68 64324000	PUSH Reverse2,00403264	String2 = "123123"
00401191	. 68 84324000	PUSH Reverse2,00403284	String1 = "3265754874"
00401196	. E8 25020000	CALL <JMP,&kernel32.lstrcmpA>	lstrcmpA
0040119B	. 99	CDQ	
0040119C	. F7F8	IDIV EAX	
0040119E	. 6A 10	PUSH 10	Style = MB_OK MB_ICONHAND MB_APPLMOD
004011A0	. 68 16314000	PUSH Reverse2,00403116	Title = "You failed,..."
004011A5	. 68 F1304000	PUSH Reverse2,004030F1	Text = "No, that is not the right an
004011AA	. 6A 00	PUSH 0	hOwner = NULL
004011AC	. E8 3F020000	CALL <JMP,&user32.MessageBoxA>	MessageBoxA
004011B1	> E8 1E	JMP SHORT Reverse2,004011D1	
004011B3	. 3D EB030000	CMP EAX,3EB	
004011B8	> 75 17	JNZ SHORT Reverse2,004011D1	
004011BA	. 6A 40	PUSH 40	Style = MB_OK MB_ICONASTERISK MB_APP
004011BC	. 68 21324000	PUSH Reverse2,00403221	Title = "About"
004011C1	. 68 24314000	PUSH Reverse2,00403124	Text = "KeyGenMe 4 by Vallani,PSolut
004011C6	. 6A 00	PUSH 0	hOwner = NULL
004011D1	=Reverse2,004011D1		

다음과 같은 화면이 나온다.

나는 여기서 문자열을 비교해주는 함수에서 답을 얻었다.