

컴퓨터 보안 실습과제 보고서

2012003909 김승현

과제목표

이번과제의 목표는 조교님이 주신 Rena's reversing 응용 프로그램을 리버싱하여 다음을 수행하는 것입니다.

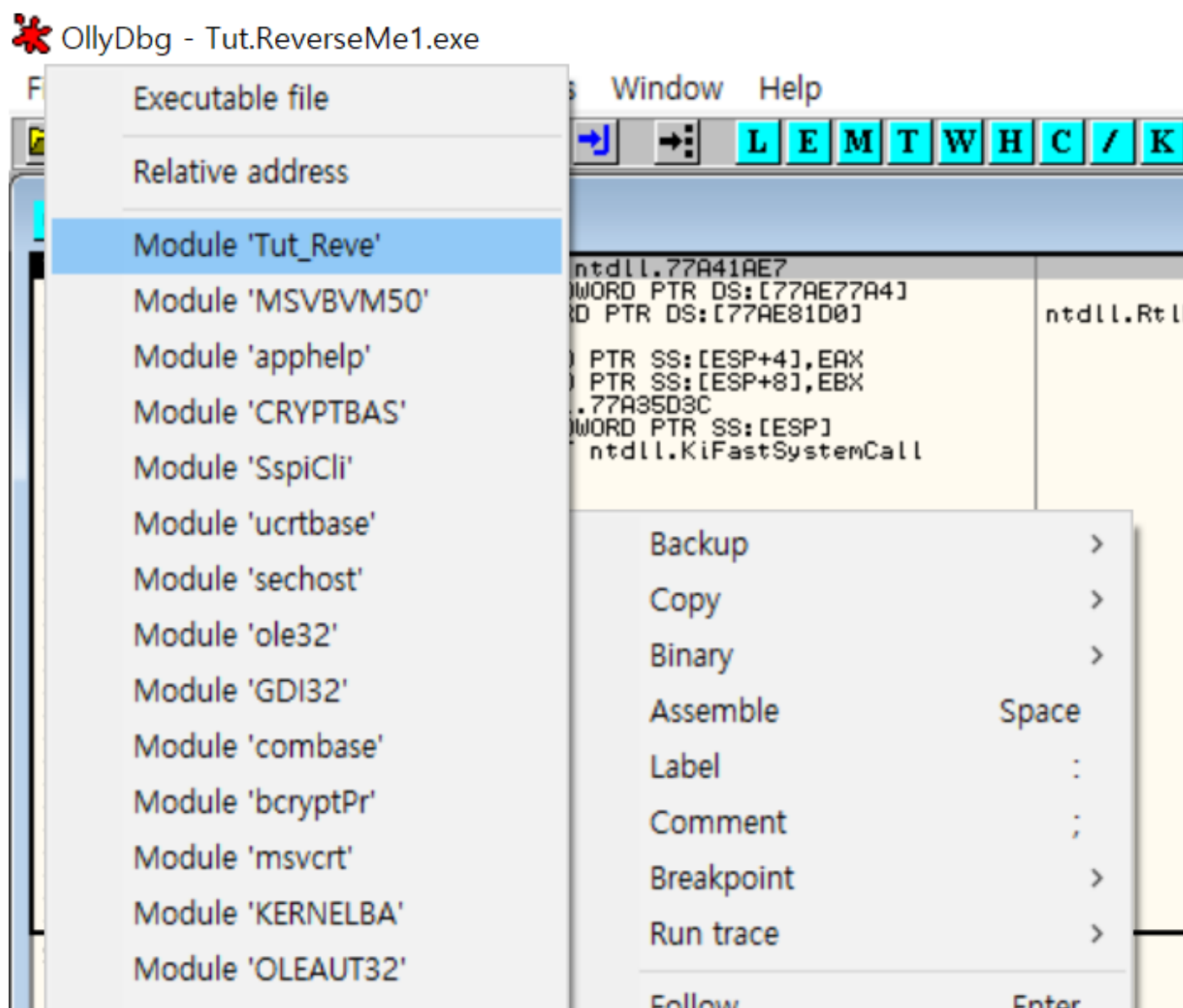
1. 프로그램 실행시 맨 처음에 뜨는 메시지 박스 출력을 없애는 것
2. Registration Code 구하기 (우회하는 방법 제외)

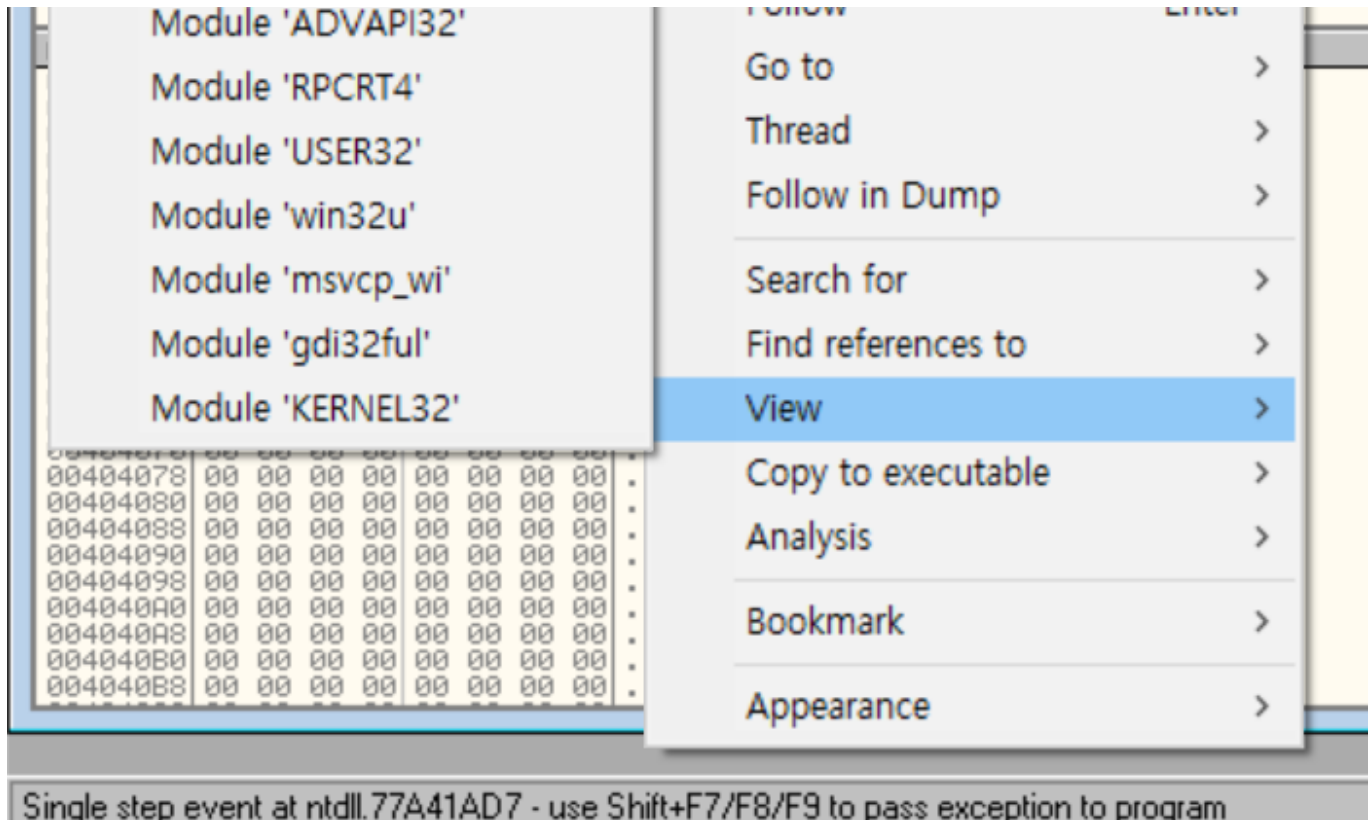
저는 1번 같은 경우 메시지박스 함수 호출하는 부분을 찾아서 아무 일도 일어나지 않도록 고쳐주고

2번 같은 경우 비교하는 부분에서 UNICODE나 Stack에서 찾아내도록 하였습니다

메시지박스 우회

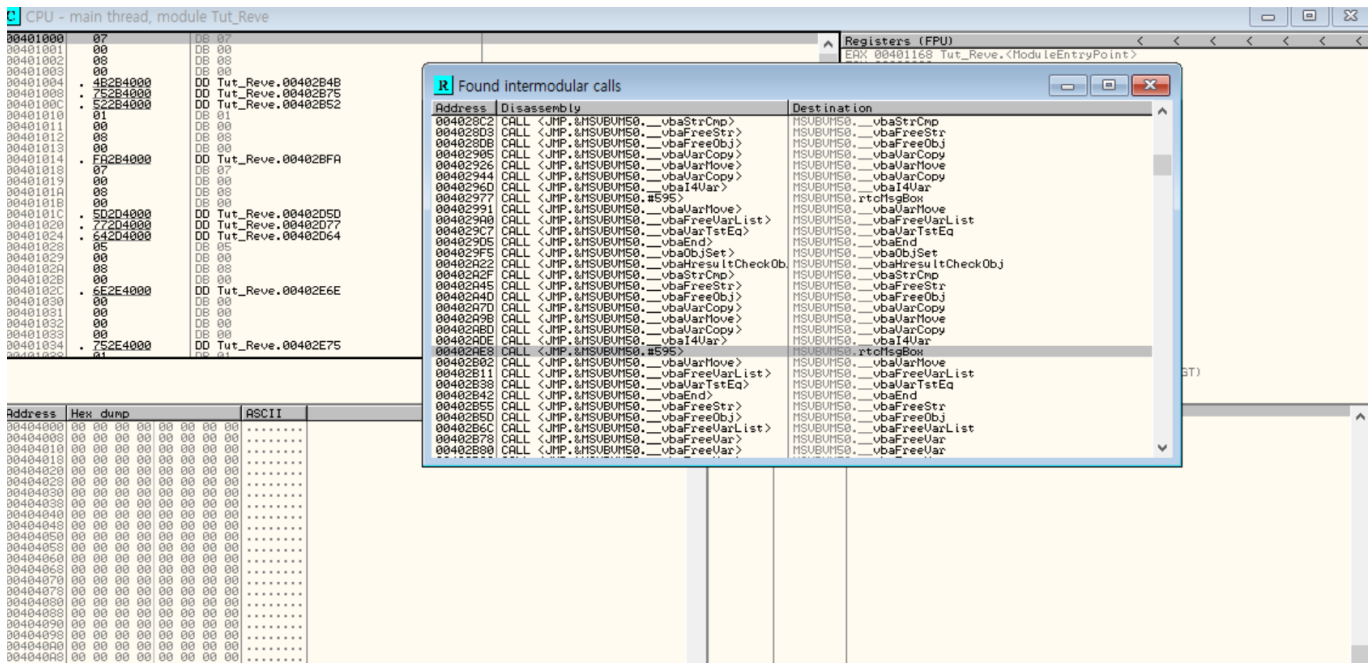
ollydbg로 프로그램을 열고 [View] - [Module Tut_Reve] 메뉴를 이용해서 프로그램의 메인으로 들어갑니다.





그 다음은 메시지박스를 호출 하는 부분을 찾아야 합니다. 이는 [Search for] - [All intermodular calls] 메뉴를 이용해서 찾을 수 있습니다.

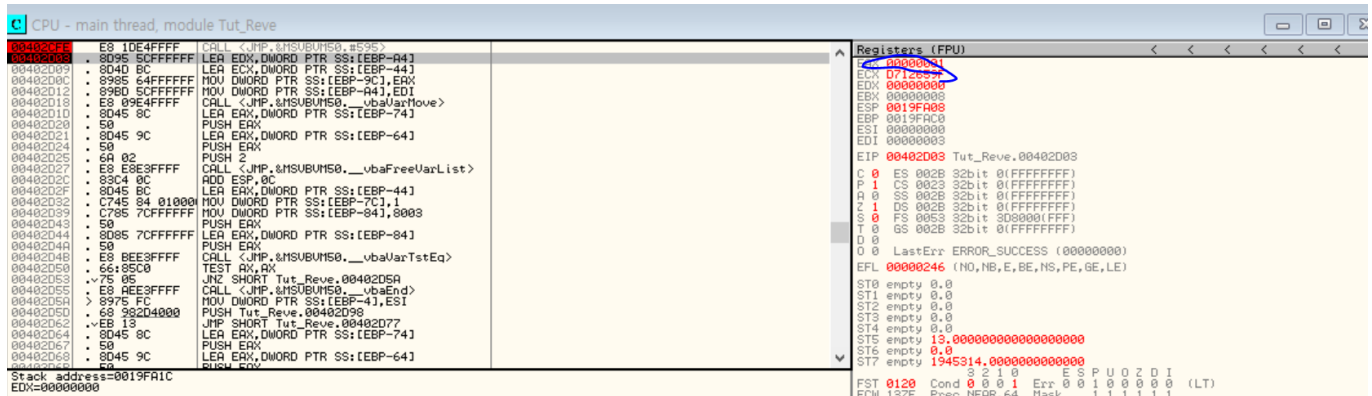
함수 이름중에 'rtcMsgBox'라는 메시지박스 호출 함수이름 같은 부분이 보입니다.



rtcMsgBox 함수 호출 부분으로 가면 CALL 명령어가 보입니다. 저는 처음에 이를 우회하기 위해 NOP 명령어로 채워봤으나, 그러면 실행시 다음으로 넘어갈 수가 없습니다.

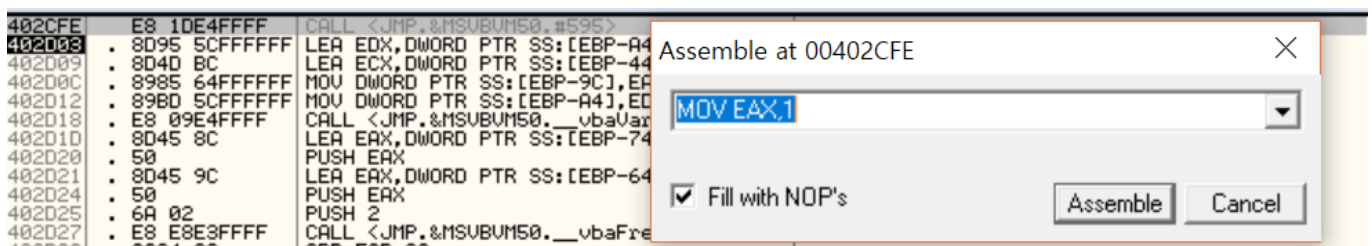
메시지박스가 예/아니오 입력을 받아서 비교처리를 하기 때문이므로 '예' 버튼을 눌렀을 때의 결과를 알아야 합니다

따라서 rtcMsgBox 호출 후 다음 구분에 Breakpoints를 걸어서 실행 후 '예' 버튼을 눌러봅니다.

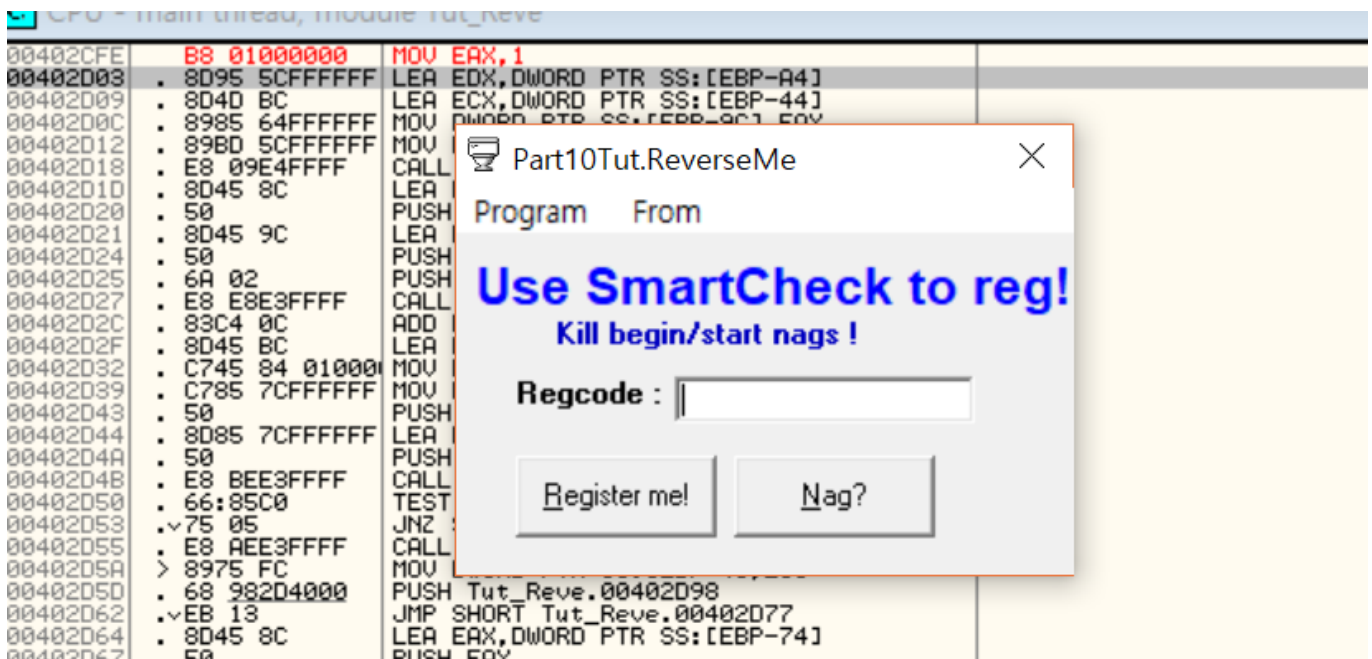


실행 결과 EAX값이 1으로 바뀐 것을 알 수 있습니다.

따라서 CALL 명령어를 MOV EAX,1 로 설정해주었습니다.



실행결과 메시지박스 출력없이 바로 Registration Code 입력하는 폼으로 넘어가는 것을 볼 수 있습니다.



Registration Code 얻기

Registration Code는 생각보다 쉽게 얻을 수 있었습니다.

모듈의 메인에서 [Search for] - [All referenced text strings] 을 통해서 모든 문자열을 찾습니다

```

004022E4 DD Tut_Reve.00401D78 ASCII "Label1"
0040228D PUSH Tut_Reve.00401DDC UNICODE "I'mlena151"
004022F5 MOV DWORD PTR SS:[EBP-84],Tut_Reve.0040 UNICODE "Yep ! You succeeded registering !"
00402934 MOV DWORD PTR SS:[EBP-84],Tut_Reve.0040 UNICODE "Congrats !!!!!"
00402A2A PUSH Tut_Reve.00401DDC UNICODE "I'mlena151"
00402A69 MOV DWORD PTR SS:[EBP-84],Tut_Reve.0040 UNICODE "Sorry ! Wrong registration code !"
00402AA9 MOV DWORD PTR SS:[EBP-84],Tut_Reve.0040 UNICODE "RegCode is wrong!"
00402C85 MOV DWORD PTR SS:[EBP-7C],Tut_Reve.0040 UNICODE "Get rid of all Nags and find the right r
00402CBE MOV DWORD PTR SS:[EBP-7C],Tut_Reve.0040 UNICODE "Nag Screen "
00402D03 LEA EDX,DWORD PTR SS:[EBP-A4] (Initial CPU selection)
00402E28 MOV DWORD PTR SS:[EBP-5C],Tut_Reve.0040 UNICODE "ReverseMe Tutorial Part10 lena151 "
00402F9A PUSH Tut_Reve.00401FEC UNICODE "Visible"
00403060 PUSH Tut_Reve.00401FEC UNICODE "Visible"

```

Registration Code를 잘못 입력 했을 때 뜨는 문구 'Sorry! Wrong registration code!' 부분을 찾아 들어갑니다

Address	Hex dump	ASCII	Registers
00402A0F	7D 16	JGE SHORT Tut_Reve.00402A27	EAX 004011
00402A11	68 A0000000	PUSH 0A0	ECX 000000
00402A16	68 F41D4000	PUSH Tut_Reve.00401DF4	EDX 000000
00402A18	FFB5 50FFFFFF	PUSH DWORD PTR SS:[EBP-B0]	EBX 003250
00402A21	50	PUSH EAX	ESP 0019FF
00402A22	E8 17E7FFFF	CALL <JMP.&MSUBUM50.__vbaHresultCheckOb	ESI 000000
00402A27	FF75 A8	PUSH DWORD PTR SS:[EBP-58]	EDI 000000
00402A2A	68 DC1D4000	PUSH Tut_Reve.00401DDC	EIP 77A41A
00402A2F	E8 16E7FFFF	CALL <JMP.&MSUBUM50.__vbaStrCmp>	C 0 ES 00
00402A34	F7D8	NEG EAX	P 1 CS 00
00402A36	1BC0	SBB EAX,EAX	A 0 SS 00
00402A38	8D4D A8	LEA ECX,DWORD PTR SS:[EBP-58]	Z 1 DS 00
00402A3B	F7D8	NEG EAX	S 0 FS 00
00402A3D	F7D8	NEG EAX	T 0 GS 00
00402A3F	8985 48FFFFFF	MOV DWORD PTR SS:[EBP-B8],EAX	D 0
00402A45	E8 EEE6FFFF	CALL <JMP.&MSUBUM50.__vbaFreeStr>	O 0 LastE
00402A4A	8D4D A4	LEA ECX,DWORD PTR SS:[EBP-5C]	EFL 000002
00402A4D	E8 E0E6FFFF	CALL <JMP.&MSUBUM50.__vbaFreeObj>	ST0 empty
00402A52	66 83BD 48FF	CMPL WORD PTR SS:[EBP-B8],0	ST1 empty
00402A5A	7D 16	JGE SHORT Tut_Reve.00402A27	ST2 empty
00402A60	8D95 74FFFFFF	LEA EDX,DWORD PTR SS:[EBP-8C]	ST3 empty
00402A66	8D4D AC	LEA ECX,DWORD PTR SS:[EBP-54]	ST4 empty
00402A69	C785 7CFFFFFF	MOV DWORD PTR SS:[EBP-84],Tut_Reve.0040	ST5 empty
00402A73	C785 74FFFFFF	MOV DWORD PTR SS:[EBP-8C],8	ST6 empty
00402A7D	E8 AAE6FFFF	CALL <JMP.&MSUBUM50.__vbaVarCopy>	ST7 empty
00402A82	8D95 74FFFFFF	LEA EDX,DWORD PTR SS:[EBP-8C]	
00402A88	8D4D DC	LEA ECX,DWORD PTR SS:[EBP-24]	
00402A8B	C785 7CFFFFFF	MOV DWORD PTR SS:[EBP-84],10	
00402A95	899D 74FFFFFF	MOV DWORD PTR SS:[EBP-8C],EBX	
00402A98	E8 8FE6FFFF	CALL <JMP.&MSUBUM50.__vbaUseHuge>	
00402B47	Tut_Reve.00402B47		FST 0000
			FCW 027F

Address	Hex dump	ASCII
00404000	00 00 00 00 00 00 00 00
00404008	00 00 00 00 00 00 00 00
00404010	00 00 00 00 00 00 00 00
00404018	00 00 00 00 00 00 00 00
00404020	00 00 00 00 00 00 00 00
00404028	00 00 00 00 00 00 00 00
00404030	00 00 00 00 00 00 00 00
00404038	00 00 00 00 00 00 00 00
00404040	00 00 00 00 00 00 00 00

코드 위를 올라다보면 의심스러운 CMP,JE 명령어 부분이 있습니다. 저는 처음에 저번 과제와 마찬가지로 저 부분에서 STACK을 보면 답이 있을 거라고 생각하고 Breakpoint로 스택을 보았습니다

```

0019F028 77A0A27C ntdll.77A0A27C
0019F02C 02689DD0
0019F030 02689DD0
0019F034 74A5D6D0 combase.74A5D6D0
0019F038 0263BD50
0019F03C 74A5D6D0 combase.74A5D6D0
0019F040 0019F054
0019F044 74A5D6EB combase.74A5D6EB
0019F048 02620000 ASCII "fBWN(!"
0019F04C 00000000
0019F050 02689DD0
0019F054 0019F07C
0019F058 7519AA21 OLEAUT32.7519AA21
0019F05C 74B78EB8 combase.74B78EB8
0019F060 02689DD0
0019F064 00000000
0019F068 02689DD0
0019F06C 7519AA21 OLEAUT32.7519AA21

```

의심스러운 문자열 fBWN(! 를 발견했지만 아쉽게도 정답이 아니었습니다.

다시한번 코드를 보았을 때, CMP 부분위에 UNICODE I'mlena151 이라는 문자열이 있는 것을 발견했습니다. 이 역시 수상하여 입력해 보았습니다

The screenshot shows a debugger window with assembly code on the left and a registration nag screen in the foreground. The assembly code includes instructions like `PUSH EAX`, `CALL <JMP.&MSUBUM50.__vbaHresultCheckOb`, `PUSH DWORD PTR SS:[EBP-58]`, `PUSH Tut_Reve.004010DC`, `CALL <JMP.&MSUBUM50.__vbaStrCmp>`, `NEG EAX`, `SBB EAX,EAX`, `LEA ECX,DWORD PTR SS:[EBP-58]`, `NEG EAX`, `NEG EAX`, `MOV DWORD PTR SS:[EBP-B8],EAX`, `CALL <JMP.&MSUBUM50.__vbaFreeStr>`, `LEA ECX,DWORD PTR SS:[EBP-5C]`, `CALL <JMP.&MSUBUM50.__vbaFreeObj>`, `CMP WORD PTR SS:[EBP-B8],0`, `JE Tut_Reve.00402B47`, `LEA EDX,DWORD PTR SS:[EBP-8C]`, `LEA ECX,DWORD PTR SS:[EBP-54]`, `MOV DWORD PTR SS:[EBP-84],Tut_R`, `MOV DWORD PTR SS:[EBP-8C],8`, `CALL <JMP.&MSUBUM50.__vbaVarCop`, `LEA EDX,DWORD PTR SS:[EBP-8C]`, `LEA ECX,DWORD PTR SS:[EBP-24]`, `MOV DWORD PTR SS:[EBP-8C],10`, `MOV DWORD PTR SS:[EBP-8C],EBX`, `CALL <JMP.&MSUBUM50.__vbaVarMov`, `LEA EDX,DWORD PTR SS:[EBP-8C]`, `LEA ECX,DWORD PTR SS:[EBP-34]`, `MOV DWORD PTR SS:[EBP-84],Tut_R`, `MULL DWORD PTR SS:[EBP-8C],2`. The registration nag screen is titled "Part10Tut.ReverseMe" and contains the text "Use SmartCheck to reg!", "Kill begin/start nags!", and a "Regcode" field with the value "I'mlena151". There are buttons for "Register me!" and "Nag?".

Assembly code snippet:

```

00402A21 . 50          PUSH EAX
00402A22 . E8 17E7FFFF CALL <JMP.&MSUBUM50.__vbaHresultCheckOb
00402A27 > FF75 A8     PUSH DWORD PTR SS:[EBP-58]
00402A2A . 68 DC104000 PUSH Tut_Reve.004010DC
00402A2F . E8 16E7FFFF CALL <JMP.&MSUBUM50.__vbaStrCmp>
00402A34 . F7D8       NEG EAX
00402A36 . 1BC0       SBB EAX,EAX
00402A38 . 8D4D A8     LEA ECX,DWORD PTR SS:[EBP-58]
00402A3B . F7D8       NEG EAX
00402A3D . F7D8       NEG EAX
00402A3F . 8985 48FFFF MOV DWORD PTR SS:[EBP-B8],EAX
00402A45 . E8 EEE6FFFF CALL <JMP.&MSUBUM50.__vbaFreeStr>
00402A4A . 8D4D A4     LEA ECX,DWORD PTR SS:[EBP-5C]
00402A4D . E8 E0E6FFFF CALL <JMP.&MSUBUM50.__vbaFreeObj>
00402A52 . 66 83BD 48FF CMP WORD PTR SS:[EBP-B8],0
00402A5A . 7E F84 E70000 JE Tut_Reve.00402B47
00402A60 . 8D95 74FFFF LEA EDX,DWORD PTR SS:[EBP-8C]
00402A66 . 8D4D AC     LEA ECX,DWORD PTR SS:[EBP-54]
00402A69 . C785 7CFFFF MOV DWORD PTR SS:[EBP-84],Tut_R
00402A73 . C785 74FFFF MOV DWORD PTR SS:[EBP-8C],8
00402A7D . E8 AAE6FFFF CALL <JMP.&MSUBUM50.__vbaVarCop
00402A82 . 8D95 74FFFF LEA EDX,DWORD PTR SS:[EBP-8C]
00402A88 . 8D4D DC     LEA ECX,DWORD PTR SS:[EBP-24]
00402A8B . C785 7CFFFF MOV DWORD PTR SS:[EBP-84],10
00402A95 . 899D 74FFFF MOV DWORD PTR SS:[EBP-8C],EBX
00402A98 . E8 86E6FFFF CALL <JMP.&MSUBUM50.__vbaVarMov
00402AA0 . 8D95 74FFFF LEA EDX,DWORD PTR SS:[EBP-8C]
00402AA6 . 8D4D CC     LEA ECX,DWORD PTR SS:[EBP-34]
00402AA9 . C785 7CFFFF MOV DWORD PTR SS:[EBP-84],Tut_R
00402AB3 . C785 74FFFF MOV DWORD PTR SS:[EBP-8C],2

```

Registration Nag Screen:

Part10Tut.ReverseMe

Program From

Use SmartCheck to reg!
Kill begin/start nags !

Regcode : I'mlena151

Register me! Nag?

그 결과 I'mlena151이 registration code라는 것을 알게 되었습니다.