

Basic 05

05파일을 실행시키면 다음과 같은 창이 뜬다.



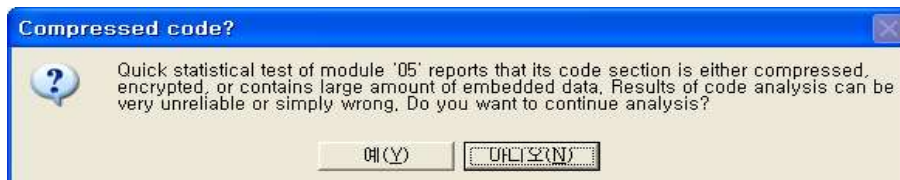
창의 Register now ! 버튼을 누르면



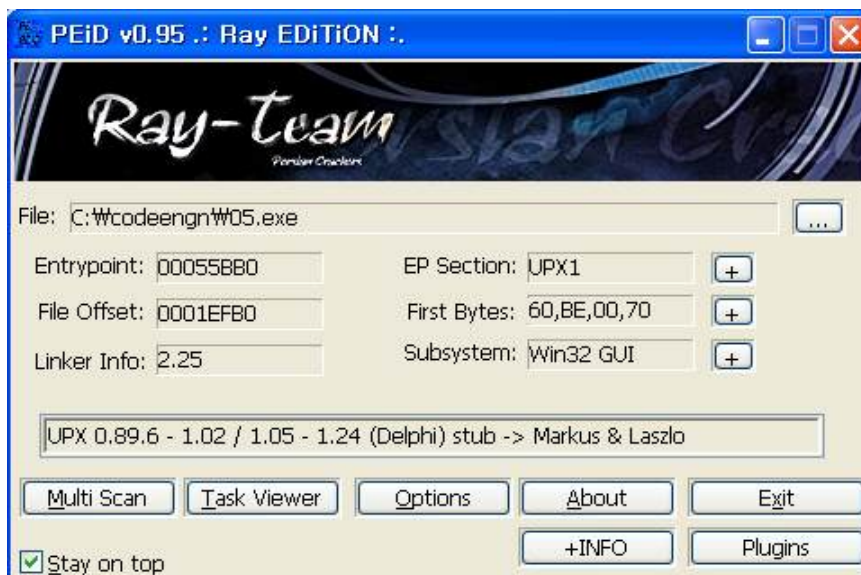
시리얼값이 잘못됐다는 메시지가 뜨게 된다.

이 메시지를 이용하여 시리얼을 찾을 수 있을 것 같다.

OllyDbg를 이용하여 파일을 열면



위와 같은 창이 뜨게 되는데 코드 섹션이 압축되었다는 얘기 같다. PEiD 툴을 사용하여 보면

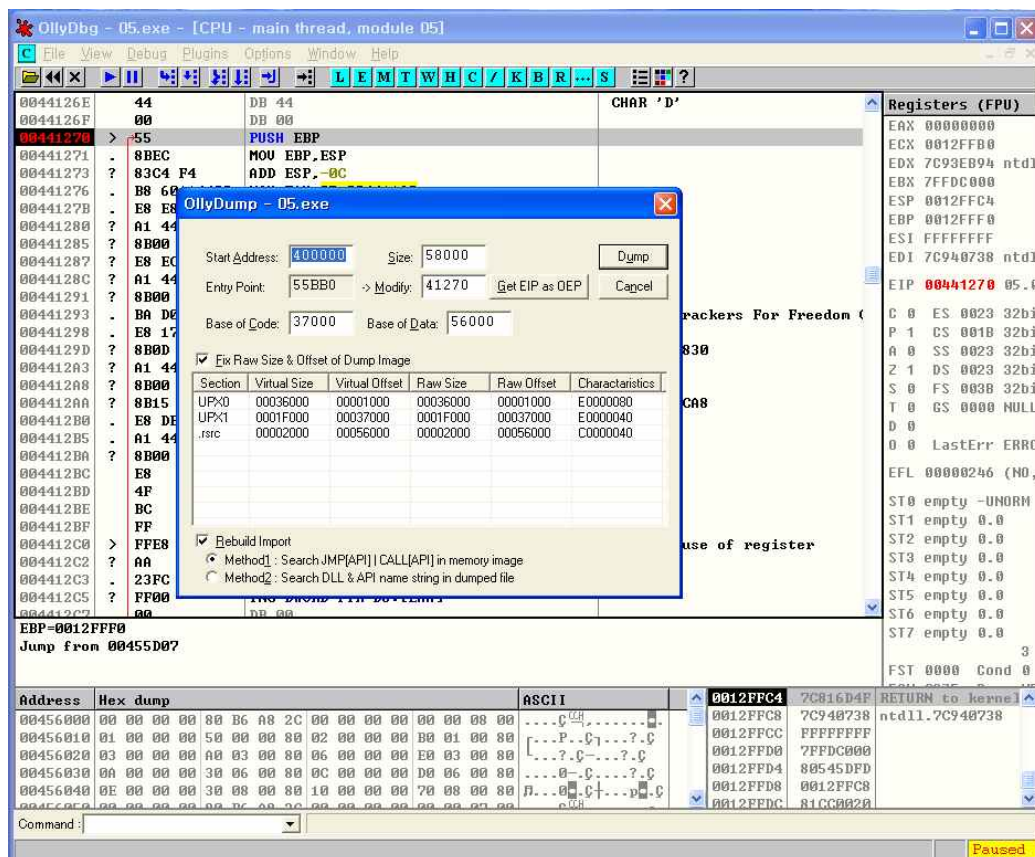


UPX로 패킹되어 있는 것을 볼 수 있다.

패킹 되어 있는 코드는 언패킹하여 보지않으면 제대로 분석 할 수 없으므로 언패킹을 먼저 실행

행한다.

OllyDbg로 다시 파일을 열고 실행되는 라인의 제일 밑 까지 내려보면 JMP 05.441270이라는 부분이 있다. 이 점프가 가르키고 있는 곳이 바로 Original Entry Point라고 볼 수 있다. 이부분에 브레이크를 걸고 디버깅을 한 후, OllyDump라는 플러그인을 사용하여 언팩하여 본다.



Dump버튼을 눌러 언팩팅된 파일을 생성(05\_dump.exe)하였다.

생성된 파일을 PEiD로 확인해 보면



패킹이 해제된 것을 볼 수 있다.

이제 분석을 할 수 있게 되었다.

시리얼이 틀렸을 때 메시지창이 뜨기 때문에 Back to user mode를 이용하여 메시지창이 생성되는 위치를 찾을 수 있다.

0043D131	. 68 ADD14300	PUSH 05_dump.0043D1AD	
0043D136	. 64:FF31	PUSH DWORD PTR FS:[ECX]	
0043D139	. 64:8921	MOV DWORD PTR FS:[ECX],ESP	
0043D13C	. 53	PUSH EBX	
0043D13D	. 57	PUSH EDI	
0043D13E	. 56	PUSH ESI	
0043D13F	. 8B45 FC	MOV EAX,DWORD PTR SS:[EBP-4]	
0043D142	. 8B40 24	MOV EAX,DWORD PTR DS:[EAX+24]	
0043D145	. 50	PUSH EAX	
0043D146	. E8 3191FCFF	CALL <JMP.&USER32.MessageBoxA>	Style Title Text  hOwner MessageBoxA
0043D14B	. 8945 F8	MOV DWORD PTR SS:[EBP-8],EAX	
0043D14E	. 33C0	XOR EAX,EAX	

위치를 보니 MessageBoxA함수가 호출 되었고, 이 함수를 호출하는 함수의 내부에는 시리얼이 참인지 거짓인지를 판단하는 명령어가 없다. 따라서 이 함수를 호출 하기 전에 text에 출력할 내용을 미리 저장한 것 이라고 볼 수 있다. 명령어의 제일 위쪽을 따라가면 다음과 같이 함수 시작점을 찾을 수 있고, 아래에 이 함수를 호출하는 위치를 찾을 수 있다.

0043D068	. 55	PUSH EBP	
0043D069	. 8BEC	MOV EBP,ESP	
0043D06B	. 83C4 B0	ADD ESP,-50	
0043D06E	. 53	PUSH EBX	
0043D06F	. 56	PUSH ESI	
0043D070	. 57	PUSH EDI	
0043D071	. 8BF9	MOV EDI,ECX	
0043D073	. 8BF2	MOV ESI,EDX	
0043D075	. 8945 FC	MOV DWORD PTR SS:[EBP-4],EAX	
0043D078	. 8B5D 08	MOV EBX,DWORD PTR SS:[EBP+8]	
0043D07B	. E8 0C90FCFF	CALL <JMP.&USER32.GetActiveWindow>	[GetActiveWindow
0043D080	. 8945 F4	MOV DWORD PTR SS:[EBP-C],EAX	
0043D083	. 6A 02	PUSH 2	
0043D085	. 8B45 F4	MOV EAX,DWORD PTR SS:[EBP-C]	
0043D088	. 50	PUSH EAX	
0043D089	. A1 AC2B4400	MOV EAX,DWORD PTR DS:[442BAC]	
0043D08E	. 8B00	MOV EAX,DWORD PTR DS:[EAX]	
0043D090	. FFD0	CALL EAX	
0043D092	. 8945 EC	MOV DWORD PTR SS:[EBP-14],EAX	
0043D095	. 6A 02	PUSH 2	

Local calls from 0043D255, 00440EED, 00440F19, 00440F6B, 00440F85, 00440F9F, 004410BA

함수를 호출하는 경로들 중 2번째 주소를 따라가보면 다양한 텍스트 값들이 사용되고 있는 것을 알 수 있다.

그 중 시리얼 키처럼 보이는 텍스트도 발견 할 수 있다.

00440EED	. E8 76C1FFFF	CALL 05_dump.0043D068	
00440EF2	> 8D55 FC	LEA EDX,DWORD PTR SS:[EBP-4]	
00440EF5	. 8B83 C8020000	MOV EAX,DWORD PTR DS:[EBX+2C8]	
00440EFB	. E8 20FFFDFF	CALL 05_dump.00420E20	
00440F00	. 837D FC 00	CMP DWORD PTR SS:[EBP-4],0	
00440F04	. 75 18	JNZ SHORT 05_dump.00440F1E	
00440F06	. 6A 00	PUSH 0	
00440F08	. B9 E80F4400	MOV ECX,05_dump.00440FE8	ASCII "No Serial entered"
00440F0D	. BA FC0F4400	MOV EDX,05_dump.00440FFC	ASCII "Enter a Serial!"
00440F12	. A1 442C4400	MOV EAX,DWORD PTR DS:[442C44]	
00440F17	. 8B00	MOV EAX,DWORD PTR DS:[EAX]	
00440F19	. E8 4AC1FFFF	CALL 05_dump.0043D068	
00440F1E	> 8D55 FC	LEA EDX,DWORD PTR SS:[EBP-4]	
00440F21	. 8B83 C4020000	MOV EAX,DWORD PTR DS:[EBX+2C4]	
00440F27	. E8 F4FEFDFD	CALL 05_dump.00420E20	
00440F2C	. 8B45 FC	MOV EAX,DWORD PTR SS:[EBP-4]	
00440F2F	. BA 14104400	MOV EDX,05_dump.00441014	ASCII "Registered User"
00440F34	. E8 F32BFCFF	CALL 05_dump.00403B2C	
00440F39	. 75 51	JNZ SHORT 05_dump.00440F8C	
00440F3B	. 8D55 FC	LEA EDX,DWORD PTR SS:[EBP-4]	
00440F3E	. 8B83 C8020000	MOV EAX,DWORD PTR DS:[EBX+2C8]	
00440F44	. E8 D7FEFDFD	CALL 05_dump.00420E20	
00440F49	. 8B45 FC	MOV EAX,DWORD PTR SS:[EBP-4]	
00440F4C	. BA 2C104400	MOV EDX,05_dump.0044102C	ASCII "GFX-754-IER-954"
00440F51	. E8 D62BFCFF	CALL 05_dump.00403B2C	
00440F56	. 75 1A	JNZ SHORT 05_dump.00440F72	
00440F58	. 6A 00	PUSH 0	

위 그림에서 프로그램이 시작되고 위쪽 칸에는 Registered User가 입력되고 아래쪽 시리얼

칸에는 GFX-754-IER-954가 입력되어야 할 것 같다. 프로그램을 구동 시켜서 그대로 입력하면 성공 메시지가 뜨는 것을 볼 수 있다.



따라서 문제의 시리얼은 GFX-754-IER-954 이다.