

확인을 누르면 카운트가 증가하는 프로그램입니다.

숫자가 남은 군생활인 것 같습니다.

패킹은 UPX패킹이 되어있어 언패킹 후 진행하도록 하겠습니다.

R Found intermodular calls		
Address	Disassembly	Destination
00444CA9	CALL DWORD PTR DS: [<&USER32.FindWindowExW>]	USER32.759F712B
00447319	CALL DWORD PTR DS: [<&USER32.OpenWindowStationW>]	USER32.759F714C
0043B0A4	CALL EBP	USER32.75A0BB6D
0046E567	CALL DWORD PTR DS: [<&USER32.GetKeyboardLayoutNameW>]	USER32.75A0FA13
00437ECF	CALL DWORD PTR DS: [<&USER32.ExitWindowsEx>]	USER32.75A106C7
004338EC	CALL DWORD PTR DS: [<&USER32.MessageBoxA>]	USER32.75A1EA11
00438AD1	CALL DWORD PTR DS: [<&USER32.MessageBoxW>]	USER32.75A1EA5F
00439B53	CALL DWORD PTR DS: [<&USER32.MessageBoxW>]	USER32.75A1EA5F
00439C65	CALL DWORD PTR DS: [<&USER32.MessageBoxW>]	USER32.75A1EA5F
004440B8	CALL DWORD PTR DS: [<&USER32.MessageBoxW>]	USER32.75A1EA5F
004536D8	CALL DWORD PTR DS: [<&USER32.MessageBoxW>]	USER32.75A1EA5F
00453908	CALL DWORD PTR DS: [<&USER32.MessageBoxW>]	USER32.75A1EA5F
0045E071	CALL DWORD PTR DS: [<&USER32.MessageBoxW>]	USER32.75A1EA5F

메세지박스를 통해 문구를 출력하기 때문에 메세지 박스에 BreakPoint를 설정하고 실행합니다.

CPU - main thread, module 06

0045E05D	. 55	PUSH EBP			
0045E05E	. 53	PUSH EBX			
0045E05F	. E8 F96CFF	CALL 06.0044405D			
0045E064	. 83C4 14	ADD ESP,14			
0045E067	. EB 0E	JMP SHORT 06.0045E077			
0045E069	> 8B4C24 30	MOV ECX,DWORD PTR SS:[ESP+30]			
0045E06D	. 56	PUSH ESI			
0045E06E	. 51	PUSH ECX			
0045E06F	. 55	PUSH EBP			
0045E070	. 53	PUSH EBX			
0045E071	. FF15 9CD6470	CALL DWORD PTR DS:[<USER32.MessageBoxW	Style	00000000	hOwner
0045E077	> 8B7424 4C	MOV ESI,DWORD PTR SS:[ESP+4C]	Title	0034FA70	0034FA70
0045E07B	. 8BF8	MOV EDI,EAX	Text	0034FC48	0034FC48
0045E07D	. E8 DEDCFAFF	CALL 06.00408D60	hOwner	0034FA70	0034FA70
0045E082	. 8D4C24 20	LEA ECX,DWORD PTR SS:[ESP+20]			
0045E086	. 893E	MOV DWORD PTR DS:[ESI],EDI			
0045E088	. C746 08 0100	MOV DWORD PTR DS:[ESI+8],1			
0045E08F	. E8 9CE3FAFF	CALL 06.0040C430			
0045E094	. 8D4C24 30	LEA ECX,DWORD PTR SS:[ESP+30]			
0045E098	. E8 93E3FAFF	CALL 06.0040C430			
0045E09D	. 5F	POP EDI			
0045E09E	. 5E	POP ESI			
0045E09F	. 5D	POP EBP			
0045E0A0	. 33C0	XOR EAX,EAX			
0045E0A2	. 5B	POP EBX			
0045E0A3	. 83C4 34	ADD ESP,34			
0045E0A6	. C2 0800	RETN 8			

Registers (FPU)

EAX	0034F900
ECX	0034FC48
EDX	0034F820
EBX	00000000
ESP	008AF8A0
EBP	0034FA70
ESI	00010000
EDI	008AF950
EIP	0045E071 06.0045E071
C 1	ES 0023 32bit 0(FFFFFFFF)
P 1	CS 001B 32bit 0(FFFFFFFF)
A 1	SS 0023 32bit 0(FFFFFFFF)
Z 0	DS 0023 32bit 0(FFFFFFFF)
S 1	FS 003B 32bit 7FDF000(FFF)
T 0	GS 0000 NULL
D 0	
0 0	LastErr ERROR_SUCCESS (00000000)
EFL	00000297 (NO,B,NE,BE,S,PE,L,LE)
ST0	empty 0.0
ST1	empty 0.0
ST2	empty 0.0
ST3	empty 0.0
ST4	empty 0.0
ST5	empty 0.0
ST6	empty 0.0
ST7	empty 0.0
FST	0100 Cond 0 0 0 1 Err 0 0 0 0 0 0 (I

Address	Hex dump	ASCII
0034FA70	31 00 00 00 00 F0 AD BA	1...濟?
0034FA78	00 F0 AD BA 00 F0 AD BA	.濟?濟?
0034FA80	AB AB AB AB AB AB AB AB	カカカカ

메세지 박스를 통해 문구를 출력하는 것을 알 수 있습니다.

밑으로 실행하면서 함수콜을 전부 뒤져보도록 하겠습니다.

CPU - main thread, module 06

00408F02	> 8B6D 00	MOV EBP,DWORD PTR SS:[EBP]	Cases 1,2,7 of switch 00408EF9	
00408F05	> 8B46 08	MOV EAX,DWORD PTR DS:[ESI+8]	Switch (cases 1..B)	
00408F08	. 83F8 01	CMP EAX,1	Cases 1,2,7 of switch 00408F08	
00408F0B	> 0F85 9A68020	JNZ 06.0042F7AB	Default case of switch 00408EF4	
00408F11	> 8B06	MOV EAX,DWORD PTR DS:[ESI]		
00408F13	> 3BE8	CMP EBP,EAX		
00408F15	> 7C 7E	JL SHORT 06.00408F95		
00408F17	> 8B47 04	MOV EAX,DWORD PTR DS:[EDI+4]		
00408F1A	. 8B4C24 44	MOV ECX,DWORD PTR SS:[ESP+44]		
00408F1E	. 40	INC EAX		
00408F1F	. 8901	MOV DWORD PTR DS:[ECX],EAX		
00408F21	> 8B4424 34	MOV EAX,DWORD PTR SS:[ESP+34]		
00408F25	. 85C0	TEST EAX,EAX		
00408F27	> 0F85 4669020	JNZ 06.0042F873		
00408F2D	> 83FB 08	CMP EBX,8		
00408F30	> 0F84 5469020	JE 06.0042F88A		
00408F36	. 83FB 0A	CMP EBX,0A		
00408F39	> 0F84 7069020	JE 06.0042F8AF		
00408F3F	. 83FB 05	CMP EBX,5		
00408F42	> 0F84 7E69020	JE 06.0042F8C6		
00408F48	. 83FB 0B	CMP EBX,0B		
00408F4B	> 0F84 8369020	JE 06.0042F8D4		
00408F51	. 83FB 0C	CMP EBX,0C		
00408F54	> 0F85 97FEFFFF	JNZ 06.00408DF1		
00408F5A	> E9 97690200	JMP 06.0042F8F6		
00408F5F	> 83E9 02	SUB ECX,2		
00408F62	. 83F9 07	CMP ECX,7		

SS:[01EAF7E0]=00000316
EBP=01EAF7E0
Jumps from 0042F3B5, 0042F44F, 0042F5A6, 0042F659, 0042F70D

Registers (FPU)

EAX	00000001
ECX	00000001
EDX	00000001
EBX	00000003
ESP	008AF8D0
EBP	01EAF7E0
ESI	01EAF78
EDI	01EAF7D0
EIP	00408F02
C 0	ES 0023
P 1	CS 001B
A 0	SS 0023
Z 1	DS 0023
S 0	FS 003B
T 0	GS 0000
D 0	
0 0	LastErr
EFL	00000246
ST0	empty 0.0
ST1	empty 0.0
ST2	empty 0.0
ST3	empty 0.0
ST4	empty 0.0
ST5	empty 0.0
ST6	empty 1.0
ST7	empty 0.0

함수콜을 분석하면서 발견한 부분입니다. EBP에 0x316의 값을 넣는 것을 확인할 수 있습니다.

CPU - main thread, module 06			
00408F02	> 8B6D 00	MOV EBP, DWORD PTR SS:[EBP]	Cases 1,2,7 of switch 00408EF9
00408F05	> 8B46 08	MOV EAX, DWORD PTR DS:[ESI+8]	
00408F08	> 83F8 01	CMP EAX, 1	Switch (cases 1..B)
00408F0B	> 0F85 9A680200	JNZ 06.0042F7A8	
00408F11	> 8B06	MOV EAX, DWORD PTR DS:[ESI]	Cases 1,2,7 of switch 00408F08
00408F13	> 3BE8	CMP EBP, EAX	
00408F15	> 7C 7E	JL SHORT 06.00408F95	
00408F17	> 8B47 04	MOV EAX, DWORD PTR DS:[EDI+4]	Default case of switch 00408EF4
00408F1A	> 8B4C24 44	MOV ECX, DWORD PTR SS:[ESP+44]	
00408F1E	> 40	INC EAX	
00408F1F	> 8901	MOV DWORD PTR DS:[ECX], EAX	
00408F21	> 8B4424 34	MOV EAX, DWORD PTR SS:[ESP+34]	
00408F25	> 85C0	TEST EAX, EAX	
00408F27	> 0F85 46690200	JNZ 06.0042F873	
00408F2D	> 83FB 08	CMP EBX, 8	
00408F30	> 0F84 54690200	JE 06.0042F88A	
00408F36	> 83FB 0A	CMP EBX, 0A	
00408F39	> 0F84 70690200	JE 06.0042F8AF	
00408F3F	> 83FB 05	CMP EBX, 5	
00408F42	> 0F84 7E690200	JE 06.0042F8C6	
00408F48	> 83FB 0B	CMP EBX, 0B	
00408F4B	> 0F84 83690200	JE 06.0042F8D4	
00408F51	> 83FB 0C	CMP EBX, 0C	
00408F54	> 0F85 97FEFFFF	JNZ 06.00408DF1	
00408F5A	> E9 97690200	JMP 06.0042F8F6	
00408F5F	> 83E9 02	SUB ECX, 2	
00408F62	> 83F9 07	CMP ECX, 7	
EAX=00000005 EBP=00000316 Jumps from 0042F7FE, 0042F810, 0042F81C, 0042F828, 0042F832, 0042F83F, 0042F846			
Address	Hex dump	ASCII	
01EA1A78	05 00 00 00 0D F0 AD BA	

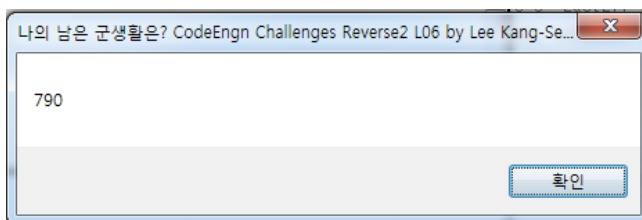
EAX에 ESI주소가 담고있는 값을 이동시킵니다.

ESI의 주소는 0x01EA1A78이고 그 값은 0x5입니다. 이 값은 메세지박스에 출력되는 숫자와 값이 같습니다.

이 값과 EBP에 저장된 값과 비교한 뒤 나머지 연산을 처리하게 됩니다.

CPU - main thread, module 06			
00408F02	> 8B6D 00	MOV EBP, DWORD PTR SS:[EBP]	Cases 1,2,7 of switch 00408EF9
00408F05	> 8B46 08	MOV EAX, DWORD PTR DS:[ESI+8]	
00408F08	> 83F8 01	CMP EAX, 1	Switch (cases 1..B)
00408F0B	> 0F85 9A680200	JNZ 06.0042F7A8	
00408F11	> 8B06	MOV EAX, DWORD PTR DS:[ESI]	Cases 1,2,7 of switch 00408F08
00408F13	> 3BE8	CMP EBP, EAX	
00408F15	> 7C 7E	JL SHORT 06.00408F95	
00408F17	> 8B47 04	MOV EAX, DWORD PTR DS:[EDI+4]	Default case of switch 00408EF4
00408F1A	> 8B4C24 44	MOV ECX, DWORD PTR SS:[ESP+44]	
00408F1E	> 40	INC EAX	
00408F1F	> 8901	MOV DWORD PTR DS:[ECX], EAX	
00408F21	> 8B4424 34	MOV EAX, DWORD PTR SS:[ESP+34]	
00408F25	> 85C0	TEST EAX, EAX	
00408F27	> 0F85 46690200	JNZ 06.0042F873	
00408F2D	> 83FB 08	CMP EBX, 8	
00408F30	> 0F84 54690200	JE 06.0042F88A	
00408F36	> 83FB 0A	CMP EBX, 0A	
00408F39	> 0F84 70690200	JE 06.0042F8AF	
00408F3F	> 83FB 05	CMP EBX, 5	
00408F42	> 0F84 7E690200	JE 06.0042F8C6	
00408F48	> 83FB 0B	CMP EBX, 0B	
00408F4B	> 0F84 83690200	JE 06.0042F8D4	
00408F51	> 83FB 0C	CMP EBX, 0C	
00408F54	> 0F85 97FEFFFF	JNZ 06.00408DF1	
00408F5A	> E9 97690200	JMP 06.0042F8F6	
00408F5F	> 83E9 02	SUB ECX, 2	
00408F62	> 83F9 07	CMP ECX, 7	
DS:[009F1A78]=00000316 EAX=00000001 Jumps from 0042F3B5, 0042F44F, 0042F5A6, 0042F659, 0042F70D, 0042F7B7			
Address	Hex dump	ASCII	
009F1A78	16 03 00 00 0D F0 AD BA	

ESI의 값을 EBP의 값과 똑같이 바꿔서 실행하면



위 그림처럼 출력되고 프로그램이 종료됩니다.