

Advance RCE 1-1

2013년 9월 10일 화요일

오후 4:15

RCE 1. "이 프로그램은 몇 밀리세컨드 후에 종료 되는가? 정답인증은 MD5 해쉬값(대문자) 변환 후 인증하시오."

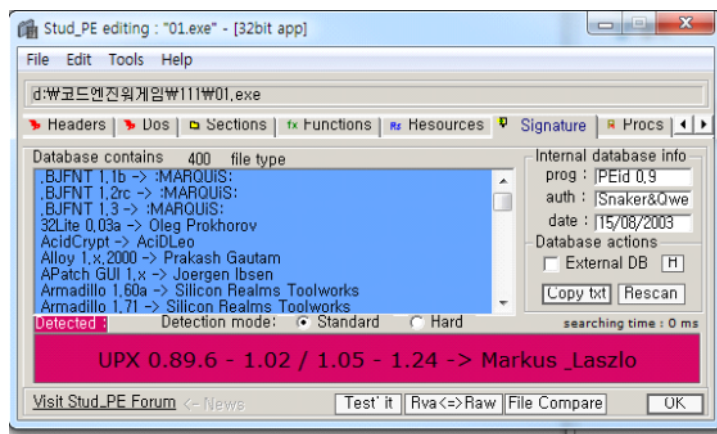


확인버튼을 누르면 종료된다.

문제의도로 보아 다음과 같이 2가지 방식으로 접근할 있다.

- 1.시간관련함수 주위를 중점적으로 분석.
- 2.버튼을 누른 후에 로직을 중점적으로 분석.

1번 방법으로 접근을 하여 문제를 풀게 되었다.



UPX로 패킹되어 있기때문에 언패킹 필요.(메뉴얼 언팩으로 진행하였기 때문에 따로 풀이는 적지 않겠다)

Advance RCE 1-2

2013년 9월 10일 화요일

오후 5:01

004AF1E0	\$ 60	PUSHAD	
004AF1E1	. BE 00004700	MOV ESI,01.00470000	
004AF1E6	. 8DBE 0010F9F1	LEA EDI,DWORD PTR DS:[ESI+FFF91000]	
004AF1EC	. 57	PUSH EDI	
004AF1ED	. 83CD FF	OR EBP,FFFFFFFF	
004AF1F0	~ EB 10	JMP SHORT 01.004AF202	
004AF1F2	. 90	NOP	
004AF1F3	. 90	NOP	
004AF1F4	. 90	NOP	
004AF1F5	. 90	NOP	
004AF1F6	. 90	NOP	
004AF1F7	. 90	NOP	
004AF1F8	> 8A06	MOV AL,BYTE PTR DS:[ESI]	
004AF1FA	. 46	INC ESI	
004AF1FB	. 8807	MOV BYTE PTR DS:[EDI],AL	
004AF1FD	. 47	INC EDI	
004AF1FE	> 01DB	ADD EBX,EBX	
004AF200	~ 75 07	JNZ SHORT 01.004AF209	
004AF202	> 8B1E	MOV EBX,DWORD PTR DS:[ESI]	
004AF204	. 83EE FC	SUB ESI,-4	
004AF207	. 11DB	ADC EBX,EBX	
004AF209	> 72 ED	JB SHORT 01.004AF1F8	
004AF20B	. B8 01000000	MOV EAX,1	
004AF210	> 01DB	ADD EBX,EBX	
004AF212	~ 75 07	JNZ SHORT 01.004AF21B	
004AF214	. 8B1E	MOV EBX,DWORD PTR DS:[ESI]	
004AF216	. 83EE FC	SUB ESI,-4	
004AF219	. 11DB	ADC EBX,EBX	
004AF21B	> 11C0	ADC EAX,EAX	
EBP-008BFF94			
Address	Hex dump	Disassembly	Comment
00401000	0000	ADD BYTE PTR DS:[EAX],AL	008BFF68 00401000 01.00401000
00401002	0000	ADD BYTE PTR DS:[EAX],AL	008BFF6C 00000000
00401004	0000	ADD BYTE PTR DS:[EAX],AL	008BFF70 00000000
			008BFF74 008BFF94

401000에 코드를 언패킹하기 때문에 401000에 코드가 실행 될 때 BP걸리도록 설정 후 진행을 하였고, BP가 걸린 후 호출하는 함수에서 시간관련 함수인 timeGetTime 을 호출 할 때 BP가 걸리도록 하였다. 하지만 BP가 걸리지 않고 프로그램은 종료 되었다. 401000코드를 실행하기 전에 해당 함수를 호출하는 것으로 판단되어, 일단 스레드가 새로 시작될 때 마다 BP 걸리도록 설정 후 흐름을 보았다.

00416243	8BFF	MOV EDI,EDI
00416245	55	PUSH EBP
00416246	8BEC	MOV EBP,ESP
00416248	56	PUSH ESI
00416249	E8 85210000	CALL 01.004183D3
0041624E	E8 7A210000	CALL 01.004183CD
00416253	50	PUSH EAX
00416254	E8 5A210000	CALL 01.004183B3
00416259	85C0	TEST EAX,EAX
0041625B	75 2A	JNZ SHORT 01.00416287
0041625D	8B75 08	MOV ESI,DWORD PTR SS:[EBP+8]
00416260	56	PUSH ESI
00416261	E8 67210000	CALL 01.004183CD
00416266	50	PUSH EAX
00416267	E8 9B210000	CALL 01.00418407

해당 지점이 프로그램이 시작되기 전 지점이며, 메모리 주소도 실행코드 부분이다. 이 지점에서 timeGetTime 함수 호출 시 BP 설정을 하였다.

Advance RCE 1-3

2013년 9월 10일 화요일

오후 5:14

00444C28	880D D1E84800	MOV BYTE PTR DS:[48E8D1],CL	USER32.EnumChildWindows
00444C2E	FF15 24D64700	CALL DWORD PTR DS:[47D624]	
00444C34	D8 28E94800	MOV EAX,01.0048E928	
00444C39	C3	RETN	
00444C3A	53	PUSH EBX	
00444C3B	55	PUSH EBP	
00444C3C	56	PUSH ESI	
00444C3D	57	PUSH EDI	
00444C3E	8B3D 58D74700	MOV EDI,DWORD PTR DS:[47D758]	WINMM.timeGetTime
00444C44	FFD7	CALL EDI	kernel32.Sleep
00444C46	803D D3E84800	CMP BYTE PTR DS:[48E8D3],0	
00444C4D	8BF0	MOV ESI,EAX	
00444C4F	0F84 FF000000	JE 01.00444D54	
00444C55	8B5C24 14	MOV EBX,DWORD PTR SS:[ESP+14]	
00444C59	8B2D 58D14700	MOV EBP,DWORD PTR DS:[47D158]	
00444C5F	FFD7	CALL EDI	
00444C61	3BC6	CMP EAX,ESI	
00444C63	0F83 CF000000	JNB 01.00444D38	
00444C69	2BC6	SUB EAX,ESI	
00444C6B	48	DEC EAX	
00444C6C	E9 C9000000	JMP 01.00444D3A	
00444C71	8B03	MOV EAX,DWORD PTR DS:[EBX]	
00444C73	6A 00	PUSH 0	
00444C75	68 FC864300	PUSH 01.004386FC	
00444C7A	50	PUSH EAX	
00444C7B	C705 28E94900	MOV DWORD PTR DS:[49E928],0	
00444C85	FF15 58D54700	CALL DWORD PTR DS:[47D558]	
00444C8B	A1 28E94900	MOV EAX,DWORD PTR DS:[49E928]	
00444C90	85C0	TEST EAX,EAX	

처음 timeGetTime 함수를 호출하는 부분이다. 이 부분이 문제의 답이 있으며 아래와 같이 프로그램은 흘러간다.

00444C3E	8B3D 58D74700	MOV EDI,DWORD PTR DS:[47D758]	WINMM.timeGetTime
00444C44	FFD7	CALL EDI	kernel32.Sleep
00444C46	803D D3E84800	CMP BYTE PTR DS:[48E8D3],0	
00444C4D	8BF0	MOV ESI,EAX	
00444C4F	0F84 FF000000	JE 01.00444D54	
00444C55	8B5C24 14	MOV EBX,DWORD PTR SS:[ESP+14]	
00444C59	8B2D 58D14700	MOV EBP,DWORD PTR DS:[47D158]	
00444C5F	FFD7	CALL EDI	
00444C61	3BC6	CMP EAX,ESI	
00444C63	0F83 CF000000	JNB 01.00444D38	
00444C69	2BC6	SUB EAX,ESI	
00444C6B	48	DEC EAX	
00444C6C	E9 C9000000	JMP 01.00444D3A	
00444C71	8B03	MOV EAX,DWORD PTR DS:[EBX]	
00444C73	6A 00	PUSH 0	
00444C75	68 FC864300	PUSH 01.004386FC	
00444C7A	50	PUSH EAX	
00444C7B	C705 28E94900	MOV DWORD PTR DS:[49E928],0	
00444C85	FF15 58D54700	CALL DWORD PTR DS:[47D558]	
00444C8B	A1 28E94900	MOV EAX,DWORD PTR DS:[49E928]	
00444C90	85C0	TEST EAX,EAX	
00444C92	0F84 BC000000	JE 01.00444D54	
00444C98	6A 00	PUSH 0	
ESI=005D8F63			
EAX=005E0146			

현재 시스템 시간을 구한 후, 한번 더 시스템 시간을 구한다. 비교를 통해 시간이 흘러 갔으면 다음 로직 진행.

00444D38	2BC6	SUB EAX,ESI
00444D3A	3B43 04	CMP EAX,DWORD PTR DS:[EBX+4]
00444D3D	0F83 2EFFFFFF	JNB 01.00444C71
00444D43	6A 0A	PUSH 0A
00444D45	FFD5	CALL EBP
00444D47	803D D3E84800	CMP BYTE PTR DS:[48E8D3],0
00444D4E	0F85 0BFFFFFF	JNZ 01.00444C5F
00444D54	5F	POP EDI
00444D55	5E	POP ESI
00444D56	5D	POP EBP
00444D57	33C0	XOR EAX,EAX
00444D59	5B	POP EBX
00444D5A	C2 0400	RETN 4
00444D5D	83EC 08	SUB ESP,8
00444D60	56	PUSH ESI
00444D61	57	PUSH EDI
00444D62	8B7C24 24	MOV EDI,DWORD PTR SS:[ESP+24]
00444D66	33F6	XOR ESI,ESI
00444D68	C605 D2E84800	MOV BYTE PTR DS:[48E8D2],0
00444D6F	85FF	TEST EDI,EDI
00444D71	74 31	JE SHORT 01.00444D64
00444D73	C605 D3E84800	MOV BYTE PTR DS:[48E8D3],1
00444D7A	FF15 5CD14700	CALL DWORD PTR DS:[47D15C]
00444D80	894424 08	MOV DWORD PTR SS:[ESP+8],EAX
00444D84	8D4424 24	LEA EAX,DWORD PTR SS:[ESP+24]
00444D88	50	PUSH EAX
00444D89	56	PUSH ESI
00444D8A	8D4C24 10	LEA ECX,DWORD PTR SS:[ESP+10]
00444D8E	51	PUSH ECX
Stack DS:[008BF894]=0000337B		
EAX=000071E3		

흘러간 시간이 얼마인지 구하여(444D38. SUB EAX, ESI) 특정 값(444D3A. [EBX+4])과 비교하여 같으면 프로그램은 종료된다. 따라서 답은 EBX+4 에 있는 값이며 337B 이다.

문제에서 요구하는 값은 337B의 10진수 값인 "13179" 를 MD5 하여 대문자시킨 값.

"DB59260CCE0B871C7B2BB780EEE305DB"