

19.02.13 CodeEngn Basic RCE L06

Tree to Tree

## Basic RCE L06

Unpack을 한 후 Serial을 찾으시오.

정답인증은 OEP + Serial

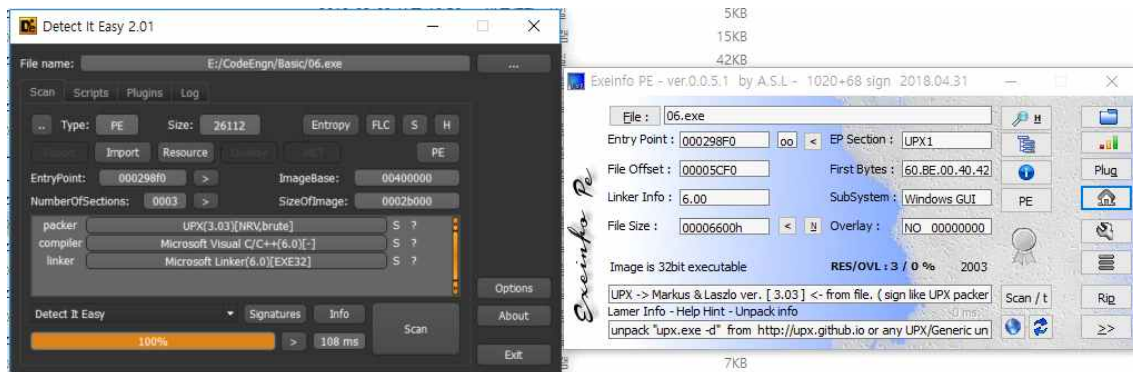
Ex) 00400000PASSWORD

— Author: Raz0r

— File Password: codeengn

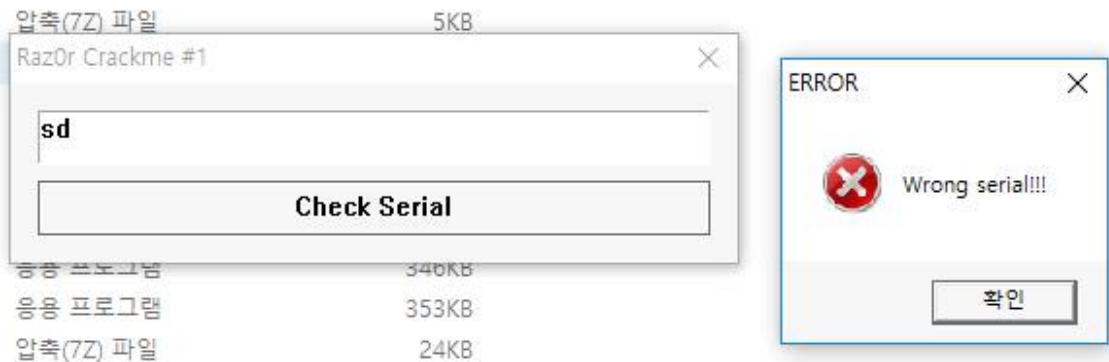


Unpack을 한 후 Serial을 찾는문제



Detect It Easy로 보니 5번문제와 같은 UPX로 팩되어있는거 확인

그냥 실행시켰을 때 모습  
아무 문자나 쳐봤다.



이제 문제 5 unpack한것과 같이 먼저 pushad지점에 breakpoint를 걸고

유형	주소	Module/Label/Exception	상태	디스어셈블러	Hits	Summary
소프트웨어	004298F0	<06.exe, EntryPoint>	One-time	pushad	0	진입점 중단점

popad부분을 찾으려 내려간다.

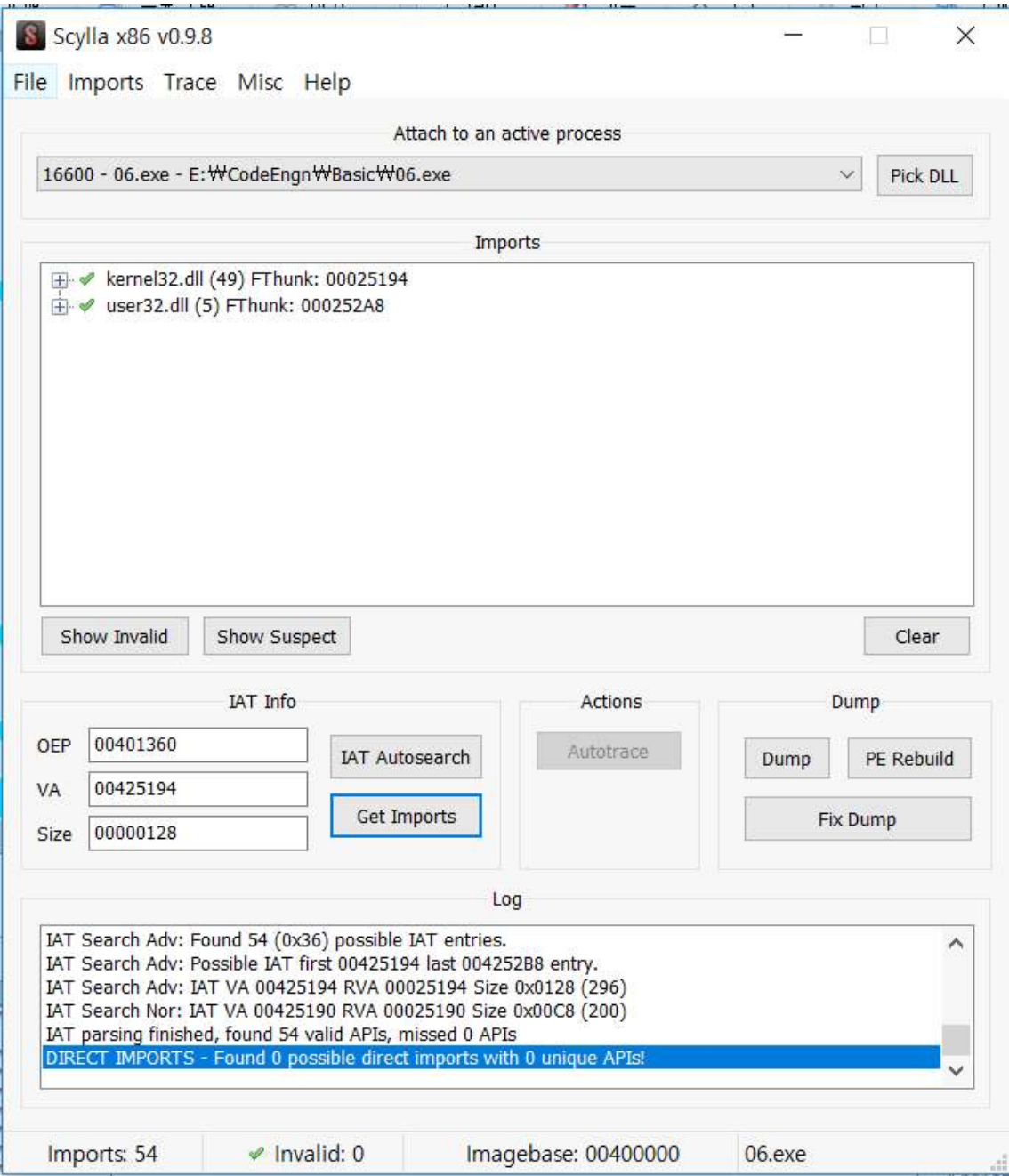
찾고나면 jmp하는곳에 break포인트를 걸고 실행시킨 후

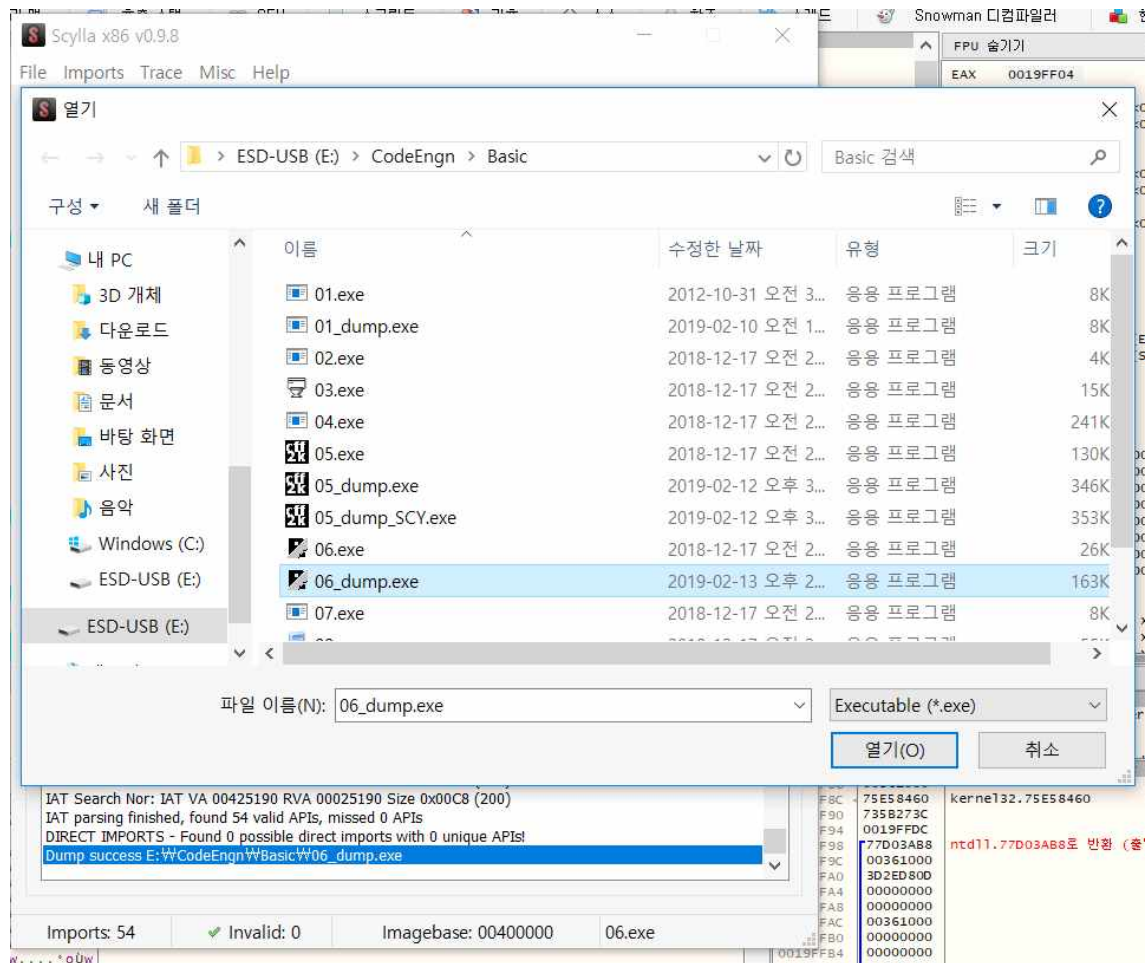
00429A33	54		push esp	
00429A35	53		push eax	
00429A36	57		push ebx	
00429A37	FFD5		push edi	
00429A39	58		call ebp	
00429A3A	61		pop eax	
00429A3B	8D 44 24 80		popad	
00429A3F	6A 00		lea eax, dword ptr ss:[esp-80]	
00429A41	39 C4		push 0	
00429A43	75 FA		cmp esp, eax	
00429A45	83 EC 80		jne 06.429A3F	
00429A48	E9 13 79 FD FF		sub esp, FFFFFFF80	
00429A4D	00 00		jmp 06.401360	
			add byte ptr ds:[eax], al	

한스텝만 실행시키면 OEP에 도달한다.

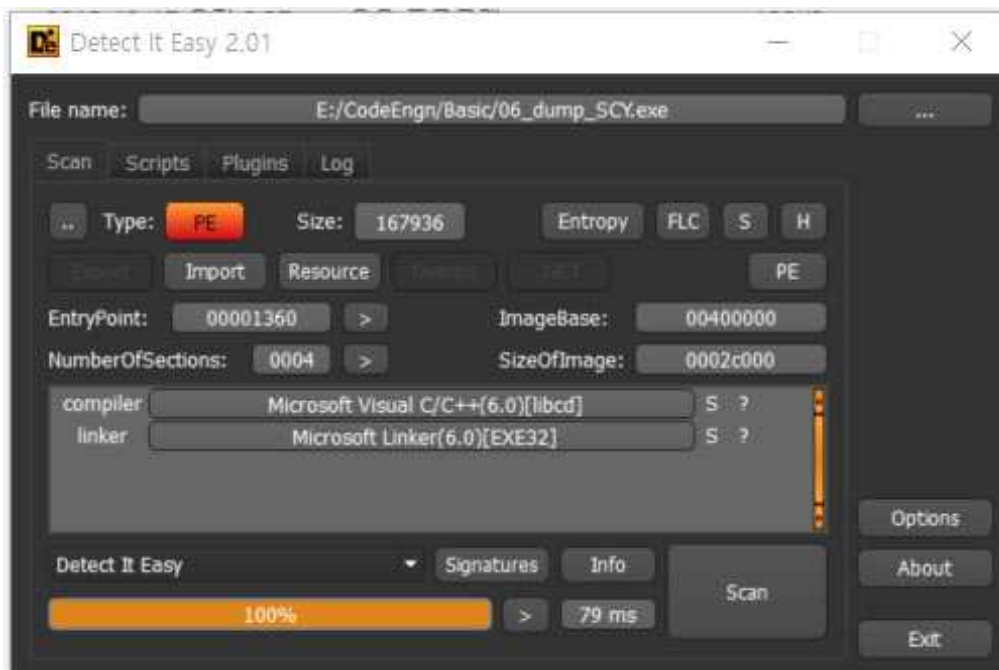
EIP	00401360	55	push ebp	OEP
	00401361	8B EC	mov ebp, esp	
	00401363	6A FF	push FFFFFFFF	
	00401366	68 44 2F 40 00	push 06.420148	
	0040136F	64 A1 00 00 00 00	push 06.402F44	
	00401375	50	mov eax, dword ptr [0]	
	00401376	64 89 25 00 00 00 00	push eax	
	0040137D	53 C4 A4	mov dword ptr [0], esp	
	00401380	53	add esp, FFFFFFFA	
	00401381	56	push ebx	
	00401382	57	push esi	
	00401383	8B 65 E8	push edi	
	00401386	FF 15 B8 51 42 00	mov dword ptr ss:[ebp-18], esp	
	0040138C	A3 6C 36 42 00	call dword ptr ds:[&getVersion]	
	00401391	A1 6C 36 42 00	mov dword ptr ds:[42366C], eax	
			mov eax, dword ptr ds:[42366C]	

Scylla x86을 이용하여 덤프, Fix dump해주면





unpack된 06\_dump\_SCY.exe 파일을 볼수 있다.



이제 Sserial 코드를 찾기위해서

언팩된 실행파일을 실행시키면 정상적인 EP에 Breakpoint가 걸려있다.

유형	주소	Module/Label/Exception	상태	디스어셈블리	Hits	Summary
소프트웨어	00401360	<06_dump_scy.exe.EntryPoint>	One-time	push ebp	0	전입점 중단점

먼저 문자열을 찾아준다.

The screenshot shows the Immunity Debugger interface with the 'Find in Memory' (Ctrl+F) menu open. The 'Find in Strings' (Ctrl+H) option is selected. The search results are displayed in a list, showing memory addresses and their corresponding string values. The address 0019FF84 is highlighted, and the string 'kerne132.75E58484로 반환' is visible. The search criteria are set to '문자열 참조(S)' (String Reference).

Address	String Value
0019FF84	kerne132.75E58484로 반환
0019FF88	00219000
0019FF8C	75E58460
0019FF90	EF851644
0019FF94	0019FFDC
0019FF98	77D03A88
0019FF9C	00219000
0019FFA0	C3174735
0019FFA4	00000000
0019FFA8	00000000
0019FFAC	00219000
0019FFB0	00000000
0019FFB4	00000000
0019FFB8	00000000
0019FFBC	00000000
0019FFC0	C3174735

이번에도 5번문제와같이 문자열에 시리얼 코드가 들어있다.

```
문자열
"AD46DFS547"
"Good Job!"
"You got it ;)"
"ERROR"
"Wrong Serial!!!"
"The value of ESP was not properly saved across a function call. This is usually a result of calling a function declared with one calling convention"
"i386\\chkesp.c"
"user32.dll"
"wsprintfA"
"Second Chance Assertion Failed: File %s, Line %d\n"
```

넣어주면



Clear