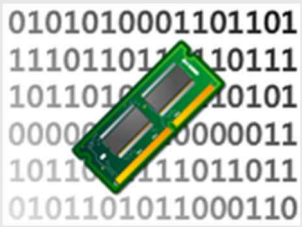
0x01. RVA (Relative Virtual Address)

VA (Virtual Address)는 **memory 에 mapping 된 절대 주소**를 뜻합니다.

RVA 는 이름 그대로 **어느 기준(ImageBase)으로부터의 상대 주소**를 뜻합니다.



VA 와 RVA 의 관계는 다음과 같습니다.

RVA + ImageBase = VA

PE header 내의 많은 정보들이 RVA 형태로 되어있습니다.

RVA 를 사용하는 이유는 relocation 때문입니다.

Dll 같은 경우 memory 에 mapping 하려는 주소에 이미 다른 library 가 있는 경우

Relocation 을 통해 빈 공간에 mapping 하게 됩니다.

0x02. RAW

RAW 는 **disk 상의 file 에서 주소**를 뜻합니다.

File 상에서의 **offset** 이라고도 부릅니다.

0x03. RVA to RAW

PE file 이 memory 에 load 될 때 RVA 와 offset 을 mapping 할 수 있습니다.

RVA 를 통해 offset 을 구할 때 사용하는 비례식과 값들은 다음과 같습니다.

RAW – PointerToRawData = RVA – VirtualAddress

RAW = RVA – VirtualAddress + PointerToRawData

RAW : File 상에서의 offset

RVA : Memory 상에서의 RVA

VirtualAddress : Offset 을 찾으려는 RVA 가 속해있는 section 의 RVA

PointerToRawData : Offset 을 찾으려는 RVA 가 속해있는 section 의 offset

0x04. Practice

PEview 를 이용해 직접 offset 과 RVA 를 변환 해 보겠습니다.

PEview 로 열어볼 파일은 reversing 입문자 용 문제인 abex' crackme 1 입니다.

해당 프로그램에서 load 하는 dll 중 하나인

KERNEL32.dll 의 이름을 가리키고 있는 RVA 를 offset 으로 변환하겠습니다.

계산식을 하나씩 채워나가도록 하겠습니다.

RAW = RVA – VirtualAddress + PointerToRawData

RVA 는 offset 을 찾으려는 값을 넣으면 됩니다.

pFile	Data	Description	Value
00000A00	0000303C	Import Name Table RVA	
00000A04	00000000	Time Date Stamp	
00000A08	00000000	Forwarder Chain	
00000A0C	00003064	Name RVA	KERNEL32.dll
00000A10	00003050	Import Address Table RVA	

KERNEL32.dll 의 RVA 는 0x00003064 입니다.

RAW = 0x00003064 – VirtualAddress + PointerToRawData

VirtualAddress 는 RVA 가 속해있는 section 의 RVA 입니다.

0x00003064 는 .idata section 에 속해있습니다.

pFile	Data	Description	Value
00000248	2E 69 64 61	Name	.idata
0000024C	74 61 00 00		
00000250	00001000	Virtual Size	
00000254	00003000	RVA	
00000258	00000200	Size of Raw Data	
0000025C	00000A00	Pointer to Raw Data	
00000260	00000000	Pointer to Relocations	
00000264	00000000	Pointer to Line Numbers	
00000268	0000	Number of Relocations	
0000026A	0000	Number of Line Numbers	
0000026C	C0000040	Characteristics	
		00000040	IMAGE_SCN_CNT_INITIALIZED_DATA
		40000000	IMAGE_SCN_MEM_READ
		80000000	IMAGE_SCN_MEM_WRITE

.idata section 의 RVA 는 0x00003000 입니다.

RAW = 0x00003064 – 0x00003000 + PointerToRawData

PointerToRawData 는 RVA 가 속해있는 section 의 offset 입니다.

pFile	Data	Description	Value
00000248	2E 69 64 61	Name	.idata
0000024C	74 61 00 00		
00000250	00001000	Virtual Size	
00000254	00003000	RVA	
00000258	00000200	Size of Raw Data	
0000025C	00000A00	Pointer to Raw Data	
00000260	00000000	Pointer to Relocations	
00000264	00000000	Pointer to Line Numbers	
00000268	0000	Number of Relocations	
0000026A	0000	Number of Line Numbers	
0000026C	C0000040	Characteristics	
		00000040	IMAGE_SCN_CNT_INITIALIZED_DATA
		40000000	IMAGE_SCN_MEM_READ
		80000000	IMAGE_SCN_MEM_WRITE

.idata section 의 offset 은 0x00000A00 입니다.

RAW = 0x00003064 – 0x00003000 + 0x00000A00

계산하면 RAW 는 0x00000A64 입니다.

Disk 상에서 0x00000A64 를 찾아가봅니다.

pFile	Raw Data	Value
00000A40	8C 30 00 00 00 00 00 00 9A 30 00 00 00 00 00 00	.0.....0.....
00000A50	7C 30 00 00 8C 30 00 00 00 00 00 00 9A 30 00 00	|0...0.....0..
00000A60	00 00 00 00 4B 45 52 4E 45 4C 33 32 2E 64 6C 6C	....KERNEL32.dll
00000A70	00 55 53 45 52 33 32 2E 64 6C 6C 00 00 00 47 65	.USER32.dll...Ge
00000A80	74 44 72 69 76 65 54 79 70 65 41 00 00 00 45 78	tDriveTypeA...Ex
00000A90	69 74 50 72 6F 63 65 73 73 00 00 00 4D 65 73 73	itProcess...Mess
00000AA0	61 67 65 42 6F 78 41 00 00 00 00 00 00 00 00 00	ageBoxA.........
00000AB0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	....
00000AC0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	....

KERNEL32.dll string 이 있는 것을 확인할 수 있습니다.