리버싱 핵심원리

저자	이승원
표지	
URL	
상태	읽는 중
완료	☐

25장 PE 패치를 이용한 DLL 로딩

DLL을 ‘실행 중인 프로세스’에 강제로 인젝션 하는 방법에 살펴봤다

이번엔 아예 대상 프로그램의 ‘실행 파일을 직접 수정’하여 DLL을 강제로 로딩하는 방법

이 방법은 한 번 적용해 놓으면 (별도의 인젝션 과정 없이) 프로세스가 시작할 때마다 원하는 DLL을

로딩하게 만들 수 있음. 일종의 크랙(crack)이라고 생각하면 됨

- 실습 목표

TextView.exe. 파일을 직접 수정하여 실행 시 myhack3.dll을 로딩하도록 만드는 것

TextView.exe 실행 화면

- myhack3.dll의 소스코드

DLLMain()의 기능은 단순히 사용자 스레드를 실행시키는 것

그 스레드에서 DownloadURL()과 DropFile() 함수를 호출하여 작업을 수행

DownloadURL() 함수는 szURL에 명시된 인터넷 파일을 다운받아 szFile 경로에 저장하는 기능을 수행 , 예제는 구글에서 index.html 파일을 받아옴

DropFile() 함수는 다운받은 index.html 파일을 TextView_Patch.exe 프로세스에 드롭시켜 그 내용을 보여줌

```
DWORD WINAPI ThreadProc(LPVOID lParam) {
	TCHAR szPath[MAX_PATH] = { 0, };
	TCHAR* p = NULL;

	GetModuleFileName(NULL, szPath, sizeof(szPath));

	if (p = _tcsrchr(szPath, L'\\')) {
		_tcscpy_s(p + 1, wcslen(DEF_INDEX_FILE) + 1, DEF_INDEX_FILE);
		if (DownloadURL(DEF_URL, szPath))
			DropFile(szPath);
	}
	return 0;
}
```

// 스레드: 프로세스 내에서 실행되는 실행 단위

```
#ifdef __cplusplus
extern "C" {
#endif
	__declspec(dllexport) void dummy() {
		return;
	}
#ifdef __cplusplus
}
#endif
```

dummy() 함수는 myhack3.dll 파일에서 외부로 서비스하는 Export 함수

(기능 X)

익스포트하는 이유는 myhack3.dll을 TextView.exe 파일의 임포트 테이블에 추가시킬 수 있도록 형식적인 완전성을 제공하기 위해서임

형식적인 완전성을 위해 익스포트 함수를 최소한 하나 이상 제공해야함

// #ifdef 참고

#ifdef __cplusplus

덧셈 뺄셈 기능을 C++로 만들었다고 합시다.

```
//헤더파일 중 일부
 int add(int a, int b); //예시기 때문에 매우 단순하게 표현
 int minus(int a, int b);
```

그런데 이 기능을 C 프로젝트에도 갖다 쓰고 싶어요.
즉 어떤 기능을 만들었으면 이 기능은 C컴파일러에서도, C++컴파일러에서도 호환이 가능하도록 짜는게 좋겠죠?
그런데 C와 C++을 혼합해서 사용할 경우 문제가 발생할 수 있는데 얘네들 linking 방식이 다르기 때문이예요. 이를 해결하기 위해 'extern'을 씁니다.
정확히는 C++에서 선언한 전역변수나 함수를 C언어에서 사용하고 싶을 때 사용하는 키워드가 extern "C" 입니다. 반대는 그냥 extern. (name mangling이라던가 깊은 얘기는 extern "C"포스팅에서 자세히 알아보아요)

```
//헤더파일 중 일부
 int add(int a, int b);  // C++ 형식으로 링킹됨

 extern "C" {
   int minus(int a, int b);  //C언어 형식으로 링킹됨
 }
```

이 extern "C"없이 C++컴파일러로 컴파일 하면 C++형식으로 링킹되기 때문에 C에서 모듈을 호출해서 쓰면 추후 문제가 발생할 수 있어요. 그래서 C 링킹방식을 써! 라는 의미로 extern "C'를 붙여 컴파일 한 후 갖다쓰는 겁니다.
근데 이 extern "C"는 C++ 컴파일러에서만 지원이 됩니다. 그래서 C컴파일러로 컴파일하면 에러가 나요 ㅎㅎ

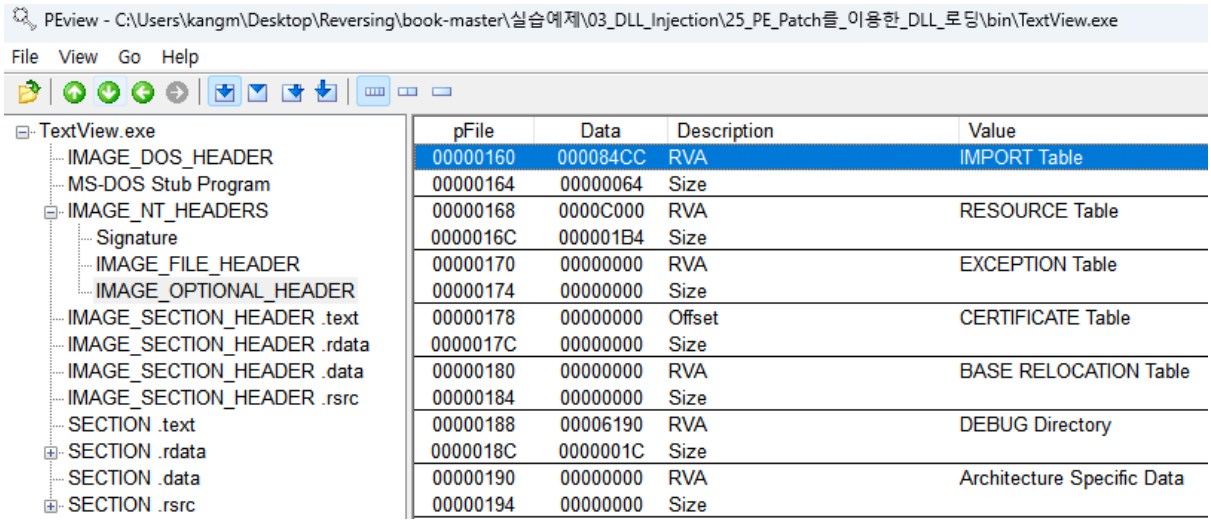
```
//헤더파일 중 일부
#ifdef __cplusplus  //c++일 경우에만 extern "C" {} 가 적용됨
extern "C" {
#endif
 int add(int a, int b);
 int minus(int a, int b);
#ifdef __cplusplus
}
#endif
```

이를 해결하기 위해 #ifdef를 사용합니다. #ifdef __cplusplus는 C++일 경우에만 범위에 있는 소스를 컴파일 하라는 의미예요. ㅎㅎ (짧게 설명하려 했는데 길어졌네요;;)

- **TextView.exe 파일 패치 준비 작업**

IDT에는 PE 파일에서 임포트하는 DLL에 대한 정보들이 구조체 리스트 형식으로 저장, 이 리스트의 마지막에 myhack3.dll 추가하면 됨

먼저 여유 공간부터 확인



TextView.exe의 IDT 주소를 확인

(PE 헤더의 IMAGE_OPTIONAL_HEADER 구조체에서 IMPORT Table RVA 값이

바로 IDT의 RVA)

TextView.exe의 IDT는 '.rdata' 섹션에 존재

IDT는 IMAGE_IMPORT_DESCRIPTOR(IID) 구조체 배열로 이루어져 있고, 배열의 마지막은 NULL 구조체로 끝남

임포트하는 DLL 파일 하나당 IID 구조체가 하나씩 필요

(IID 구조체 하나의 크기는 14바이트)

전체 IID의 영역은 RVA:84CC~852F(전체 크기는 14 * 5 = 64)

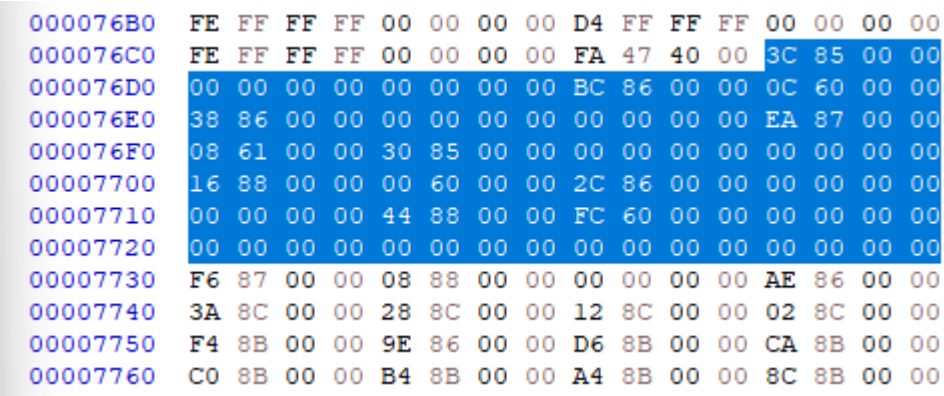
RVA	Data	Description	Value
000084CC	0000853C	Import Name Table RVA	
000084D0	00000000	Time Date Stamp	
000084D4	00000000	Forwarder Chain	
000084D8	000086BC	Name RVA	KERNEL32.dll
000084DC	0000600C	Import Address Table RVA	

IDT의 파일 옵셋은 76CC

pFile	Data	Description	Value
000076CC	0000853C	Import Name Table RVA	

IDT는 file offset으로 76CC~772F 범위에 있으며 NULL 구조체를 포함하여 총 5개의 IID 구조체가 있고 전체 크기는 64

myhack3.dll의 구조체를 추가시킬 자리가 없다는 걸 알 수 있



- IDT 이동

IDT 전체를 다른 넓은 위치로 옮긴 후 새로운 IID를 추가

(1) 파일의 다른 빈 영역을 찾는다.

(2) 파일 마지막 섹션의 크기를 늘린다.

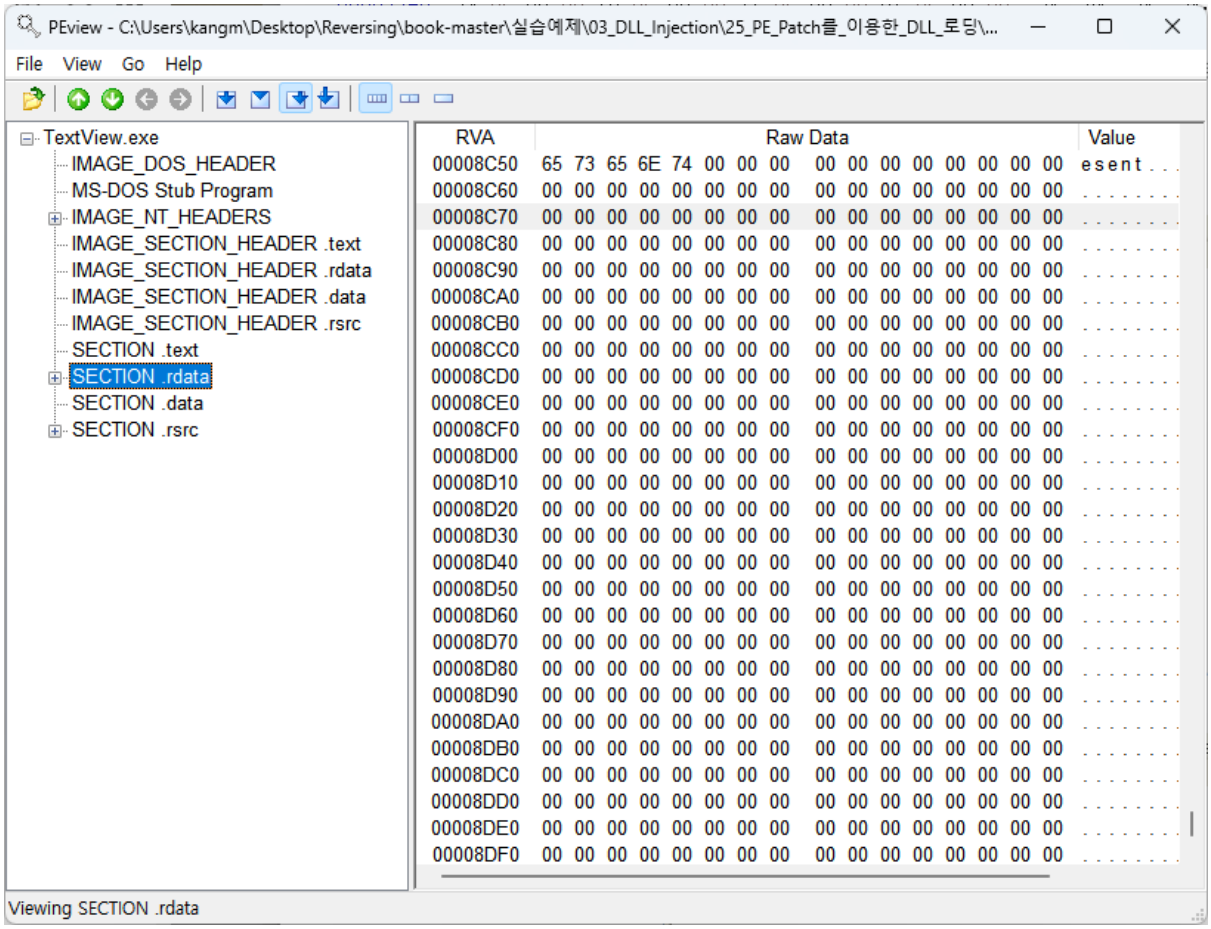
(3) 파일 끝에 새로운 섹션을 추가한다.

실습에서는 (1)을 사용

.rdata 끝부분에 NULL-Padding 영역이 있다

사용하기 전에 메모리에 로딩되는 부분인지 확인해야 함

// 섹션 헤더에 명시된 영역만큼만 메모리에 로딩됨

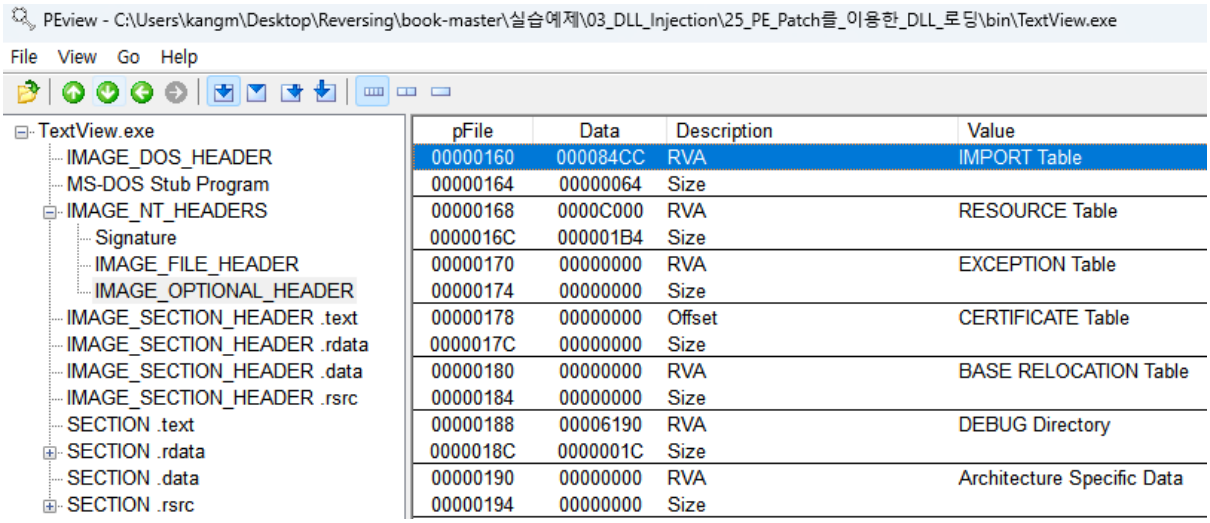


.rdata 섹션의 파일 크기는 2E00이지만 매핑되는 크기는 2C56이기에

나머지 영역의 크기 1AA(2E00-2C56)로 이 위치에 IDT 재구성 가능!

- TextView.exe 패치 작업

IMAGE_OPTIONAL_HEADER의 IMPORT Table 구조체 멤버는 IDT의 위치(RVA)와 크기를 알려주는데 RVA: 8C80 , Size: 78로 변경




```
00000150  00 00 00 00 10 00 00 00 00 00 00 00 00 00 00 00
00000160  80 8C 00 00 78 00 00 00 00 C0 00 00 B4 01 00 00
00000170  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

기존 IDT 복사 후 덮어쓰기

또 다른 자료구조(INT, Name, IAT)를 가리키는 구조체 멤버들 추가

코드 25.5 IMAGE_IMPORT_DESCRIPTOR

```
typedef struct _IMAGE_IMPORT_DESCRIPTOR {
    union {
        DWORD   Characteristics;
        DWORD   OriginalFirstThunk;  // 00008D00 => RVA to INT
    };
    DWORD   TimeDateStamp;           // 0
    DWORD   ForwarderChain;          // 0
    DWORD   Name;                    // 00008D10 => RVA to DLL Name
    DWORD   FirstThunk;              // 00008D20 => RVA to IAT
} IMAGE_IMPORT_DESCRIPTOR;
```

	RVA	RAW
INT	8D00	7F00
Name	8D10	7F10
IAT	8D20	7F20

표 25.2 INT, Name, IAT

```
00007E80  3C 85 00 00 00 00 00 00 00 00 00 00 BC 86 00 00
00007E90  0C 60 00 00 38 86 00 00 00 00 00 00 00 00 00 00
00007EA0  EA 87 00 00 08 61 00 00 30 85 00 00 00 00 00 00
00007EB0  00 00 00 00 16 88 00 00 00 60 00 00 2C 86 00 00
00007EC0  00 00 00 00 00 00 00 00 44 88 00 00 FC 60 00 00
00007ED0  00 8D 00 00 00 00 00 00 00 00 00 00 10 8D 00 00
00007EE0  20 8D 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

그 이후 아래 값을 추가함

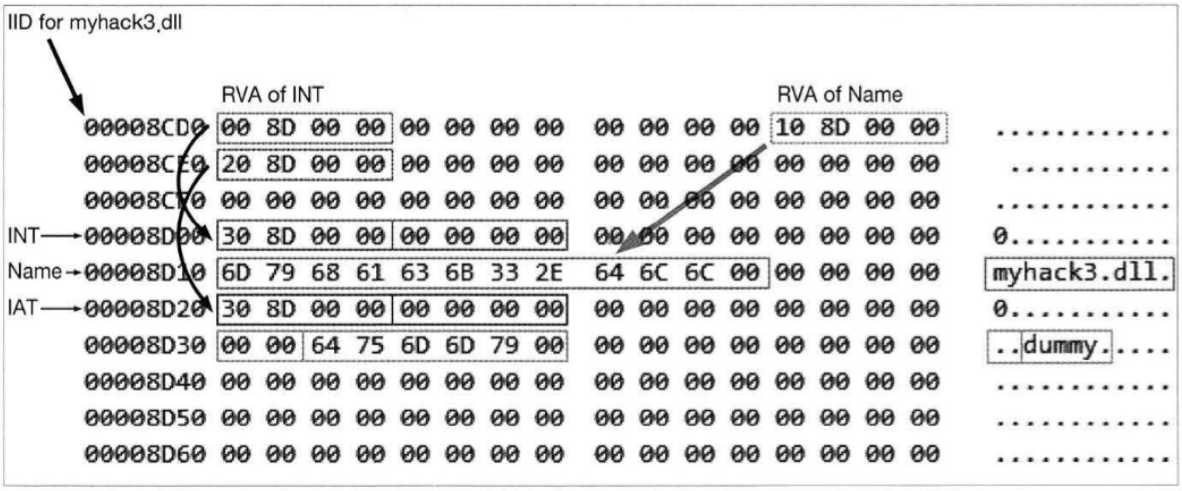
//PE파일이 메모리에 로딩되었을 때 각 섹션에서 메모리의 주소(RVA)와 파일 옵셋을 잘 매핑해야 하는데 이러한 매핑을 'RVA to RAW'라고 부름

```
00007F00  30 8D 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00007F10  6D 79 68 61 63 6B 33 2E 64 6C 6C 00 00 00 00 00
00007F20  30 8D 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00007F30  00 00 64 75 6D 6D 79 00 00 00 00 00 00 00 00 00
```

PEView로 열어 RAV 보기 옵션을 통해 확인

8CD0 주소에 myhack3.dll을 위한 IID 구조체가 존재.

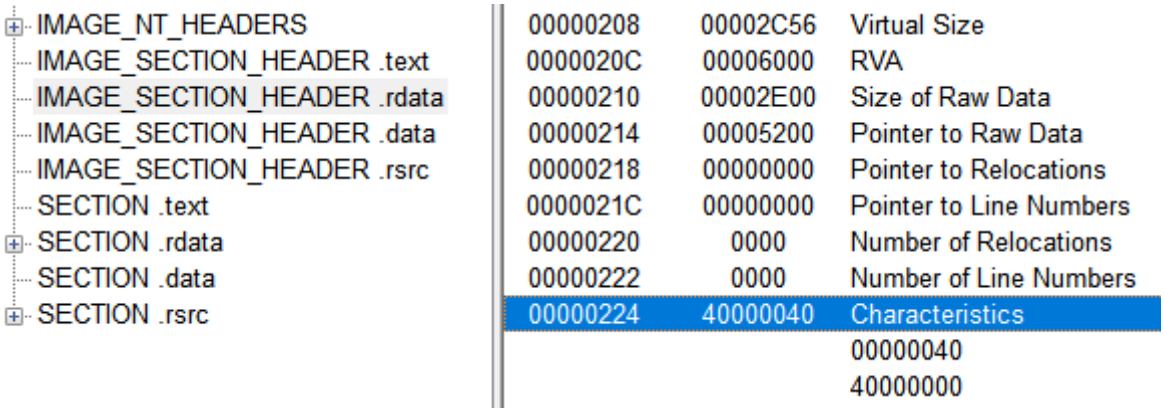
3개 중요 멤버에 입력된 값들의 의미는 실제 INT, Name, IAT의 포인터 역할



- IAT 섹션의 Characteristics 변경

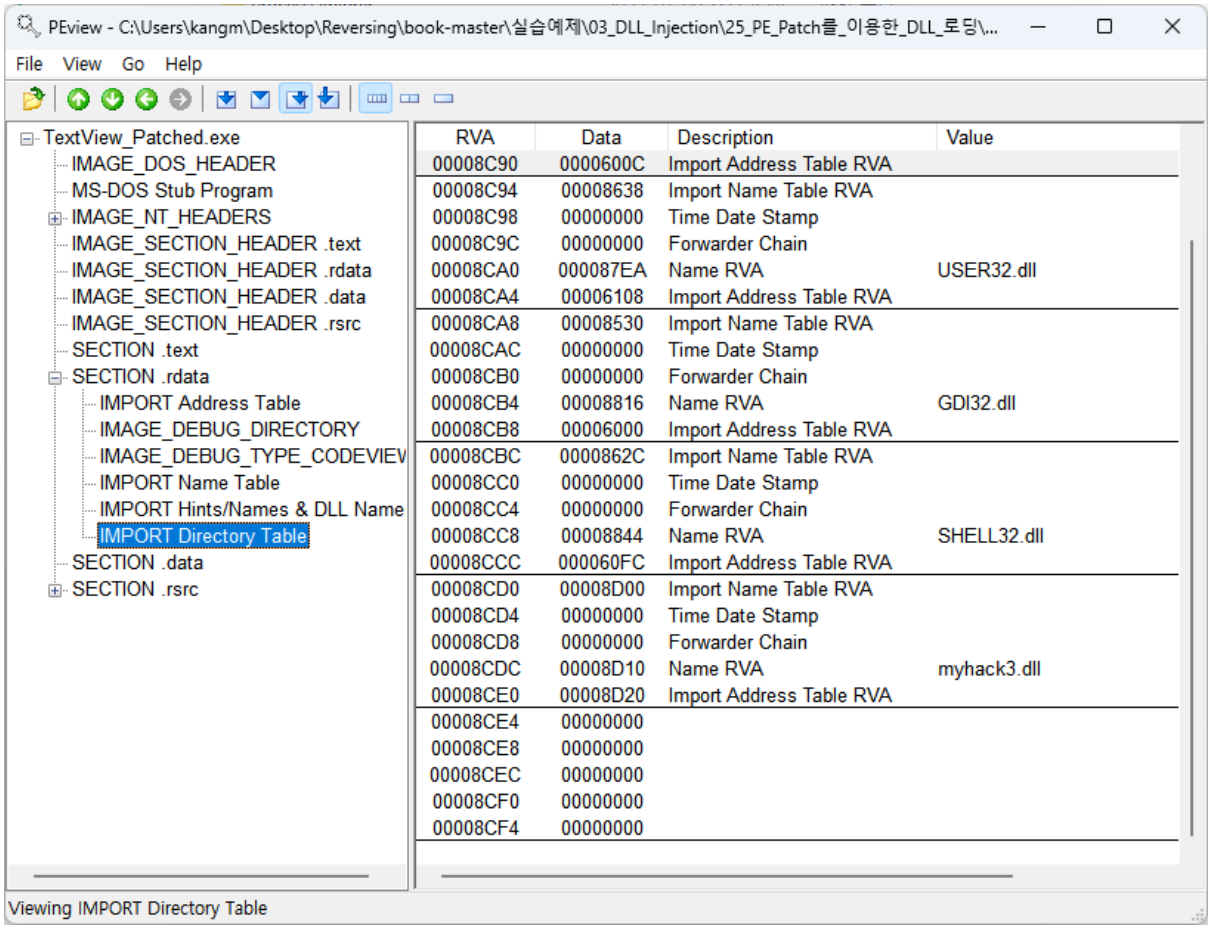
IAT는 PE 로더에 의해서 메모리에 로딩될 때 실제 함수 주소를 덮어쓰기에
해당 섹션은 반드시 WRITE 속성을 가져야함

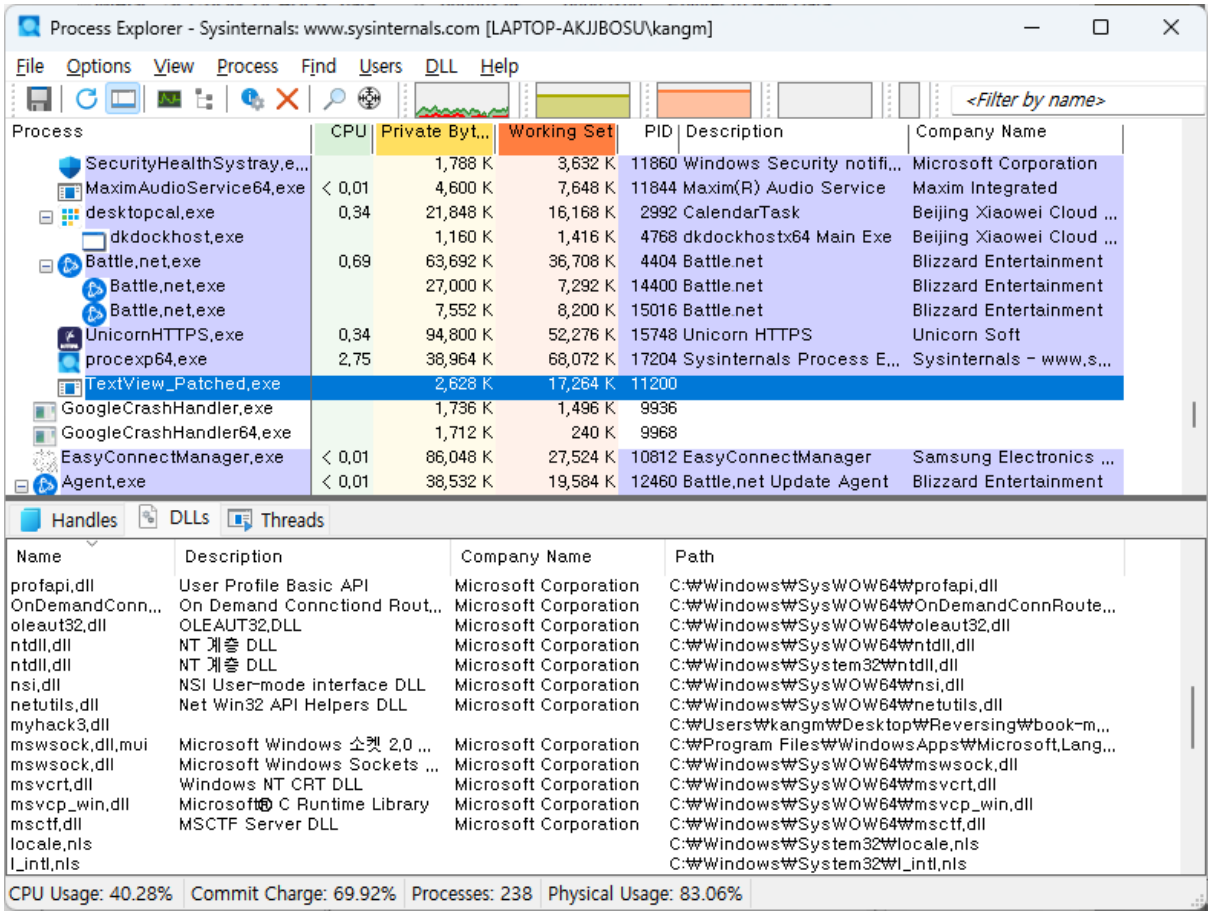
40000040 값에 IMAGE_SCN_MEM_WRITE 값을 추가하면 최종 값은 C0000040출력

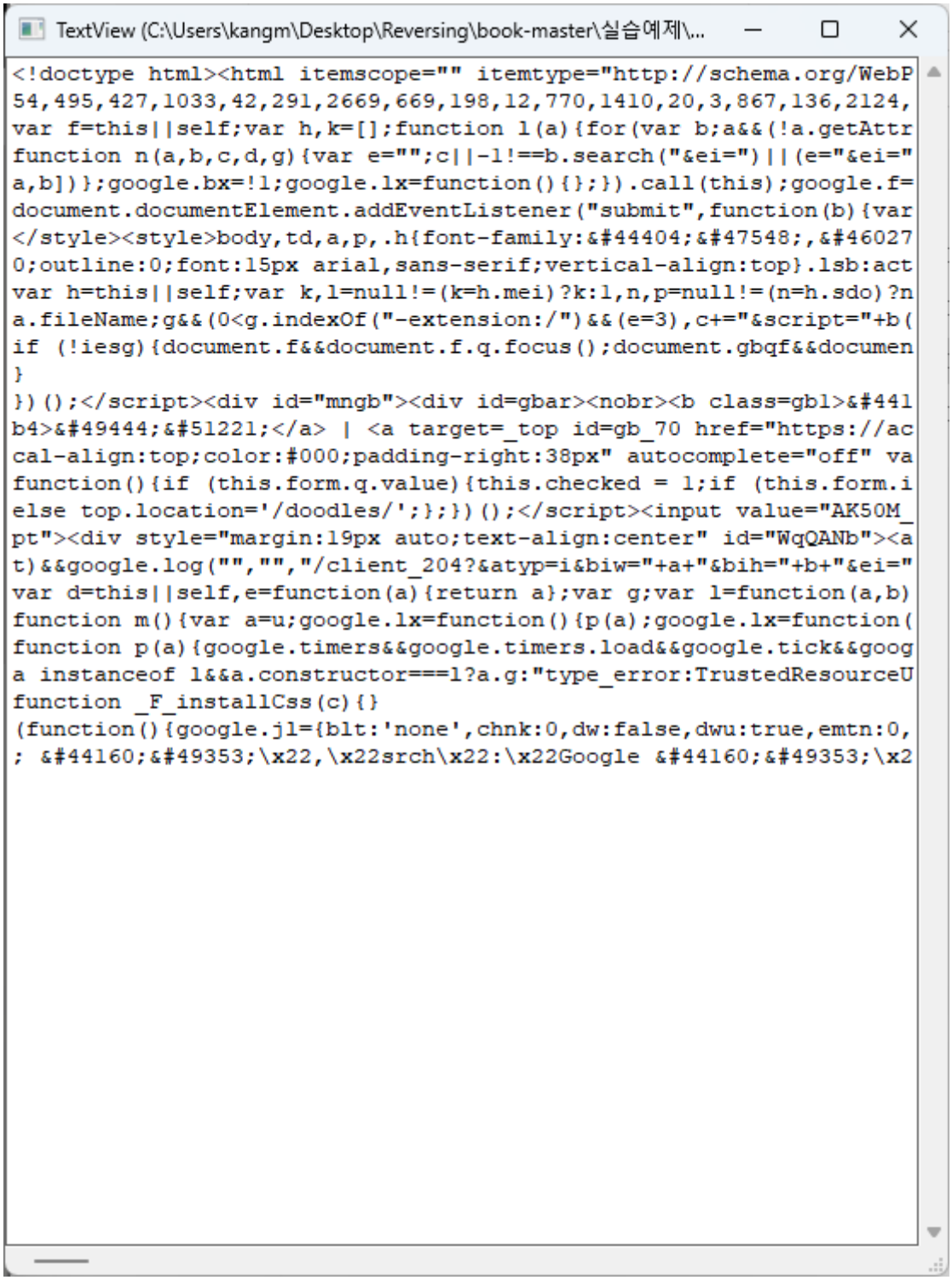


00000224 C0000040 Characteristics

- 결과 확인


PEview - C:\Users\kangm\Desktop\Reversing\book-master\실습예제\03_DLL_Injection\25_PE_Patch를_이용한_DLL_로딩\...
File View Go Help
TextView_Patched.exe
IMAGE_DOS_HEADER
MS-DOS Stub Program
IMAGE_NT_HEADERS
IMAGE_SECTION_HEADER .text
IMAGE_SECTION_HEADER .rdata
IMAGE_SECTION_HEADER .data
IMAGE_SECTION_HEADER .rsrc
SECTION .text
SECTION .rdata
IMPORT Address Table
IMAGE_DEBUG_DIRECTORY
IMAGE_DEBUG_TYPE_CODEVIEV
IMPORT Name Table
IMPORT Hints/Names & DLL Name
IMPORT Directory Table
SECTION .data
SECTION .rsrc
RVA Data Description Value
00008C90 0000600C Import Address Table RVA
00008C94 00008638 Import Name Table RVA
00008C98 00000000 Time Date Stamp
00008C9C 00000000 Forwarder Chain
00008CA0 000087EA Name RVA USER32.dll
00008CA4 00006108 Import Address Table RVA
00008CA8 00008530 Import Name Table RVA
00008CAC 00000000 Time Date Stamp
00008CB0 00000000 Forwarder Chain
00008CB4 00008816 Name RVA GDI32.dll
00008CB8 00006000 Import Address Table RVA
00008CBC 0000862C Import Name Table RVA
00008CC0 00000000 Time Date Stamp
00008CC4 00000000 Forwarder Chain
00008CC8 00008844 Name RVA SHELL32.dll
00008CCC 000060FC Import Address Table RVA
00008CD0 00008D00 Import Name Table RVA
00008CD4 00000000 Time Date Stamp
00008CD8 00000000 Forwarder Chain
00008CDC 00008D10 Name RVA myhack3.dll
00008CE0 00008D20 Import Address Table RVA
00008CE4 00000000
00008CE8 00000000
00008CEC 00000000
00008CF0 00000000
00008CF4 00000000
Viewing IMPORT Directory Table


Process Explorer - Sysinternals: www.sysinternals.com [LAPTOP-AKJJBOSU\kangm]
File Options View Process Find Users DLL Help
<Filter by name>
Process CPU Private Byt... Working Set PID Description Company Name
SecurityHealthSystray.e... 1,788 K 3,632 K 11860 Windows Security notifi... Microsoft Corporation
MaximAudioService64.exe < 0,01 4,600 K 7,648 K 11844 Maxim(R) Audio Service Maxim Integrated
desktopcal.exe 0,34 21,848 K 16,168 K 2992 CalendarTask Beijing Xiaowei Cloud ...
dkdockhost.exe 1,160 K 1,416 K 4768 dkdockhostx64 Main Exe Beijing Xiaowei Cloud ...
Battle.net.exe 0,69 63,692 K 36,708 K 4404 Battle.net Blizzard Entertainment
Battle.net.exe 27,000 K 7,292 K 14400 Battle.net Blizzard Entertainment
Battle.net.exe 7,552 K 8,200 K 15016 Battle.net Blizzard Entertainment
UnicornHTTPS.exe 0,34 94,800 K 52,276 K 15748 Unicorn HTTPS Unicorn Soft
procexp64.exe 2,75 38,964 K 68,072 K 17204 Sysinternals Process E... Sysinternals - www.s...
TextView_Patched.exe 2,628 K 17,264 K 11200
GoogleCrashHandler.exe 1,736 K 1,496 K 9936
GoogleCrashHandler64.exe 1,712 K 240 K 9968
EasyConnectManager.exe < 0,01 86,048 K 27,524 K 10812 EasyConnectManager Samsung Electronics ...
Agent.exe < 0,01 38,532 K 19,584 K 12460 Battle.net Update Agent Blizzard Entertainment
Handles DLLs Threads
Name Description Company Name Path
profapi.dll User Profile Basic API Microsoft Corporation C:₩Windows₩SysWOW64₩profapi.dll
OnDemandConn... On Demand Connctiond Rout... Microsoft Corporation C:₩Windows₩SysWOW64₩OnDemandConnRoute...
oleaut32.dll OLEAUT32.DLL Microsoft Corporation C:₩Windows₩SysWOW64₩oleaut32.dll
ntdll.dll NT 계층 DLL Microsoft Corporation C:₩Windows₩SysWOW64₩ntdll.dll
ntdll.dll NT 계층 DLL Microsoft Corporation C:₩Windows₩System32₩ntdll.dll
nsi.dll NSI User-mode interface DLL Microsoft Corporation C:₩Windows₩SysWOW64₩nsi.dll
netutils.dll Net Win32 API Helpers DLL Microsoft Corporation C:₩Windows₩SysWOW64₩netutils.dll
myhack3.dll C:₩Users₩kangm₩Desktop₩Reversing₩book-m...
mswsock.dll.mui Microsoft Windows 소켓 2.0 ... Microsoft Corporation C:₩Program Files₩WindowsApps₩Microsoft.Lang...
mswsock.dll Microsoft Windows Sockets ... Microsoft Corporation C:₩Windows₩SysWOW64₩mswsock.dll
msvcrt.dll Windows NT CRT DLL Microsoft Corporation C:₩Windows₩SysWOW64₩msvcrt.dll
msvcp_win.dll Microsoft® C Runtime Library Microsoft Corporation C:₩Windows₩SysWOW64₩msvcp_win.dll
msctf.dll MSCTF Server DLL Microsoft Corporation C:₩Windows₩SysWOW64₩msctf.dll
locale.nls C:₩Windows₩System32₩locale.nls
l_intl.nls C:₩Windows₩System32₩l_intl.nls
CPU Usage: 40.28% Commit Charge: 69.92% Processes: 238 Physical Usage: 83.06%


TextView (C:\Users\kangm\Desktop\Reversing\book-master\실습예제\...
<!doctype html><html itemscope="" itemtype="http://schema.org/WebP
54,495,427,1033,42,291,2669,669,198,12,770,1410,20,3,867,136,2124,
var f=this||self;var h,k=[];function l(a){for(var b;a&&(!a.getAttr
function n(a,b,c,d,g){var e="";c||-1!==b.search("&ei=")||(e="&ei="
a,b])};google.bx=!1;google.lx=function(){};}).call(this);google.f=
document.documentElement.addEventListener("submit",function(b){var
</style><style>body,td,a,p,.h{font-family:굴림,&#46027
0;outline:0;font:15px arial,sans-serif;vertical-align:top}.lsb:act
var h=this||self;var k,l=null!=(k=h.mei)?k:1,n,p=null!=(n=h.sdo)?n
a.fileName;g&&(0<g.indexOf("-extension:/")&&(e=3),c+="&script="+b(
if (!iesg){document.f&&document.f.q.focus();document.gbqf&&documen
}
})();</script> <nobr><b class=gb1>&#441
b4>설정 | <a target=_top id=gb_70 href="https://ac
cal-align:top;color:#000;padding-right:38px" autocomplete="off" va
function(){if (this.form.q.value){this.checked = 1;if (this.form.i
else top.location='/doodles/';};})();</script><input value="AK50M_
pt"> <a
t)&&google.log("","","/client_204?&atyp=i&biw="+a+"&bih="+b+"&ei="
var d=this||self,e=function(a){return a};var g;var l=function(a,b)
function m(){var a=u;google.lx=function(){p(a);google.lx=function(
function p(a){google.timers&&google.timers.load&&google.tick&&goog
a instanceof l&&a.constructor===l?a.g:"type_error:TrustedResourceU
function _F_installCss(c){}
(function(){google.jl={blt:'none',chnk:0,dw:false,dwu:true,emtn:0,
; 검색\x22,\x22srch\x22:\x22Google 검색\x2