

实验3-3

**寄生到非代码区段末尾
跳回原入口点**

随便找个具有空洞的非代码区段

cloudmusic.exe	pFile	Data	Description
IMAGE_DOS_HEADER	00000228	2E 72 64 61	Name
MS-DOS Stub Program	0000022C	74 61 00 00	
IMAGE_NT_HEADERS	00000230	0001E7BA	Virtual Size
Signature	00000234	00049000	RVA
IMAGE_FILE_HEADER	00000238	0001E800	Size of Raw Data
IMAGE_OPTIONAL_HEADER	0000023C	00047800	Pointer to Raw Data
IMAGE_SECTION_HEADER .text	00000240	00000000	Pointer to Relocations
IMAGE_SECTION_HEADER .rdata	00000244	00000000	Pointer to Line Numbers
IMAGE_SECTION_HEADER .data	00000248	0000	Number of Relocations
IMAGE_SECTION_HEADER .tls	0000024A	0000	Number of Line Numbers
IMAGE_SECTION_HEADER .rsrc	0000024C	40000040	Characteristics

MS-DOS Stub Program
IMAGE_NT_HEADERS
Signature
IMAGE_FILE_HEADER
IMAGE_OPTIONAL_HEADER
IMAGE_SECTION_HEADER .text
IMAGE_SECTION_HEADER .rdata

0000022C	74 61 00 00	
00000230	0001E7BA	Virtual Size
00000234	00049000	RVA
00000238	0001E800	Size of Raw Data
0000023C	00047800	Pointer to Raw Data
00000240	00000000	Pointer to Relocations
00000244	00000000	Pointer to Line Numbers

寄生位置
 $47800 + 1E7BA = 65FBA$

将入口点RVA（2D9EE）修改为病毒体RVA（ $49000 + 1E7BA = 677BA$ ）

0000012C	00000000	Size of Initialized Data
00000130	0002D9EE	Address of Entry Point
00000134	00001000	Base of Code
00000138	00049000	Base of Data
0000013C	00400000	Image Base



cloudmusic.exe*	
00000130h:	BA 77 06 00 00
00000140h:	00 10 00 00 00
00000150h:	05 00 01 00 00

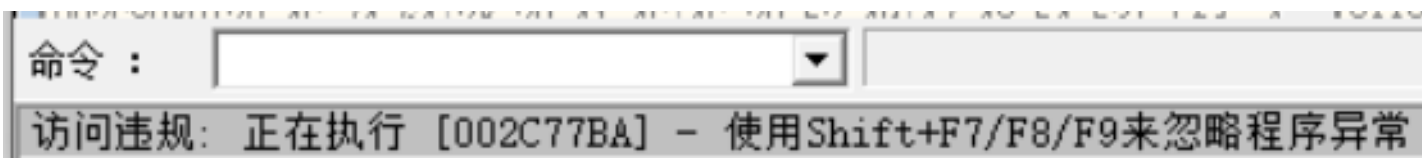
在寄生位置处插入病毒体（**学号最后一位个NOP**）和**一条JMP指令**
 JMP指令在病毒体后，用来跳回原入口点RVA（2D9EE），
 机器码为e9 xx xx xx xx，其偏移量xx xx xx xx为，这里假设学号末尾为5：
 $2D9EE - (49000 + 1E7BA + 5 + 5) = 2D9EE - 677C4 = FF FC 62 2A$

00065fb0h:	62 72 61 72 79 45 78 41 00 00 90 90 90 90 90 e9
00065fc0h:	2A 62 FC FF 00 00 00 00 00 00 00 00 00 00 00 00

用x64dbg加载感染后的程序，可以看见，入口点已经修改为寄生的病毒体部分，后面还有一条JMP指令

地址	HEX 数据	反汇编
002C77BA	90	NOP
002C77BB	90	NOP
002C77BC	90	NOP
002C77BD	90	NOP
002C77BE	90	NOP
002C77BF	- E9 2A62FCFF	JMP cloudmus.0028D9EE

但是，当我们想要去运行代码，跳回原入口点时，却发现无法进行，并提示002C77BA处的指令NOP访问违规！这是为什么呢？



我们来对比原来的.text节和现在病毒寄生的.rdata节的属性

IMAGE_SECTION_HEADER .text	00000218	00000000	Pointer to Relocations
IMAGE_SECTION_HEADER .rdata	0000021C	00000000	Pointer to Line Numbers
IMAGE_SECTION_HEADER .data	00000220	0000	Number of Relocations
IMAGE_SECTION_HEADER .tls	00000222	0000	Number of Line Numbers
IMAGE_SECTION_HEADER .rsrc	00000224	60000020	Characteristics
IMAGE_SECTION_HEADER .reloc			00000020 IMAGE_SCN_CNT_CODE
SECTION .text			20000000 IMAGE_SCN_MEM_EXECUTE
SECTION .rdata			40000000 IMAGE_SCN_MEM_READ

IMAGE_SECTION_HEADER .text	00000240	00000000	Pointer to Relocations
IMAGE_SECTION_HEADER .rdata	00000244	00000000	Pointer to Line Numbers
IMAGE_SECTION_HEADER .data	00000248	0000	Number of Relocations
IMAGE_SECTION_HEADER .tls	0000024A	0000	Number of Line Numbers
IMAGE_SECTION_HEADER .rsrc	0000024C	40000040	Characteristics
IMAGE_SECTION_HEADER .reloc			00000040 IMAGE_SCN_CNT_INITIALIZED_DATA
SECTION .text			40000000 IMAGE_SCN_MEM_READ

我们看见.rdata节并没有内存可执行的属性！（MEM_EXECUTE）
当然代码就无法执行！因此，我们需将.rdata节的属性附加内存可执行
如下图，修改为60000040

cloudmusic.exe
00000240h: 00 00 00 00 00 00 00 00 00 00 00 00 00 40 00 00 60
00000250h: 2E 64 61 74 61 00 00 00 00 0C 50 00 00 00 80 06 00

现在我们可以按**F8**进行单步执行了，执行完**JMP**指令后，可以看见，程序跳回了原入口点处

CPU - 主线程, 模块 - cloudmus		
地址	HEX 数据	反汇编
002577BA	90	NOP
002577BB	90	NOP
002577BC	90	NOP
002577BD	90	NOP
002577BE	90	NOP
002577BF	- E9 2A62FCFF	JMP cloudmus.0021D9EE
002577C4	0000	ADD BYTE PTR DS:[EAX], AL
002577C6	0000	ADD BYTE PTR DS:[EAX], AL
002577C8	0000	ADD BYTE PTR DS:[EAX], AL



CPU - 主线程, 模块 - cloudmus		
地址	HEX 数据	反汇编
0021D9EE	\$ E8 299B0000	CALL cloudmus.0022751C
0021D9F3	. ^ E9 7FFEFFFF	JMP cloudmus.0021D877
0021D9F8	CC	INT3
0021D9F9	CC	INT3
0021D9FA	CC	INT3
0021D9FB	CC	INT3
0021D9FC	CC	INT3
0021D9FD	CC	INT3
0021D9FE	CC	INT3
0021D9FF	CC	INT3
0021DA00	\$ 57	PUSH EDI
0021DA01	. 56	PUSH ESI
0021DA02	. 8B7424 10	MOV ESI, DWORD PTR SS:[ESP+10]
0021DA06	. 8B4C24 14	MOV ECX, DWORD PTR SS:[ESP+14]
0021DA0A	. 8B7C24 0C	MOV EDI, DWORD PTR SS:[ESP+C]