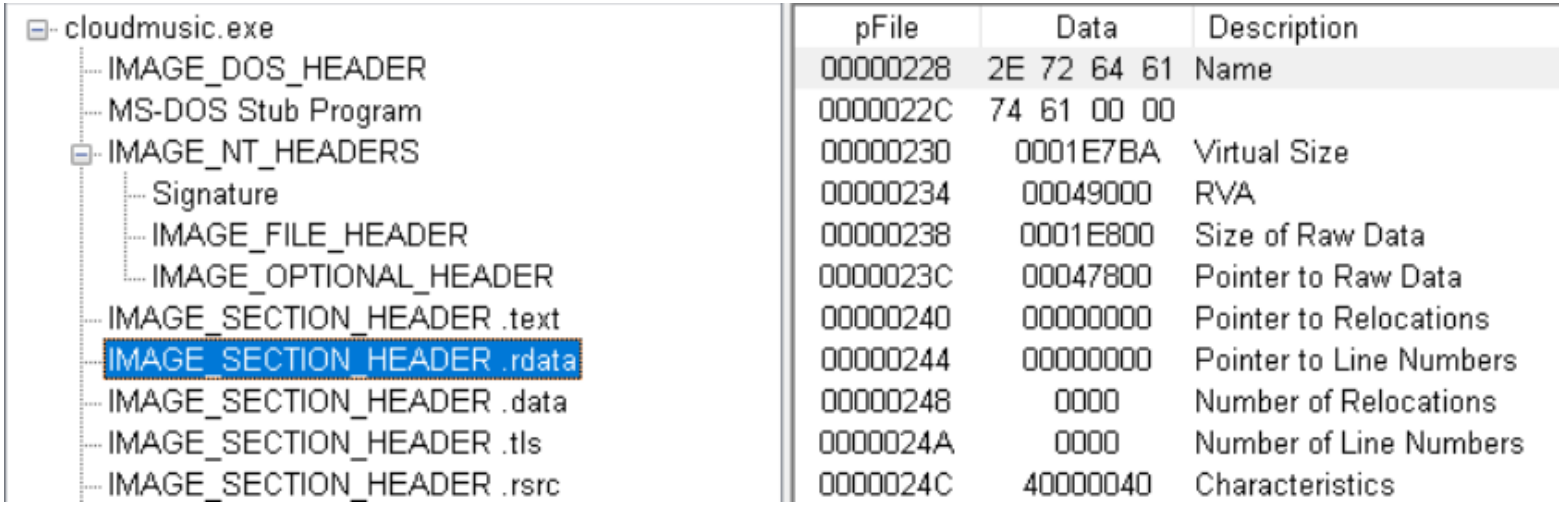
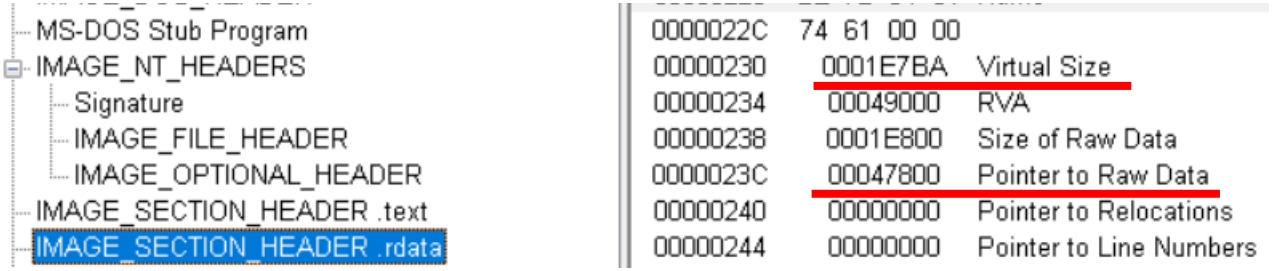
实验3-3

寄生到非代码区段末尾跳回原入口点

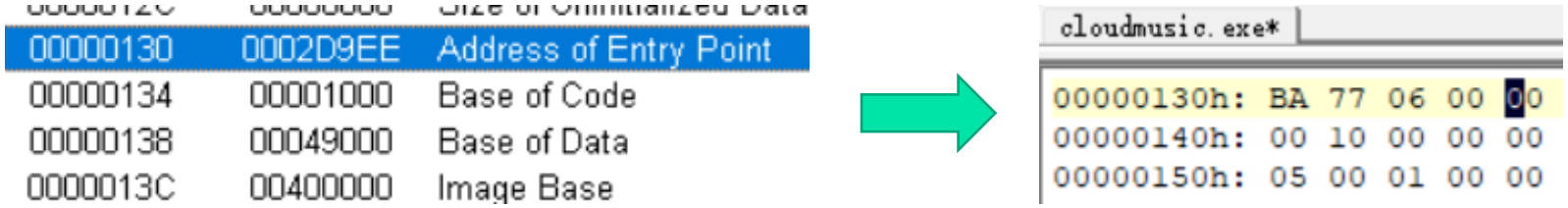
随便找个具有空洞的非代码区段



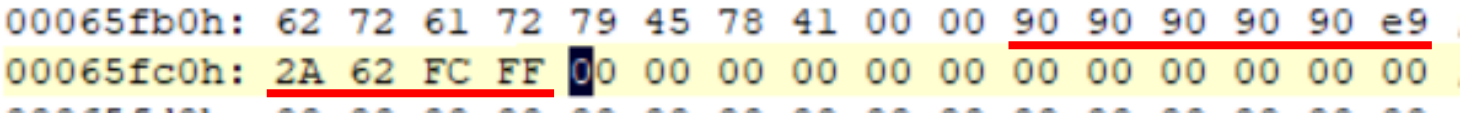


寄生位置
47800 +1E7BA=65FBA

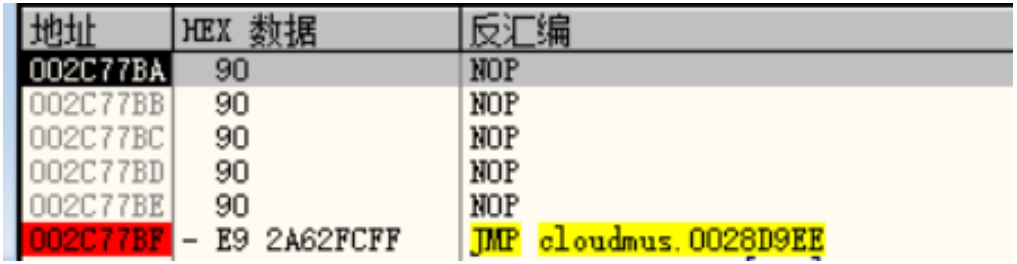
将入口点RVA（2D9EE）修改为病毒体RVA（49000+1E7BA=677BA）



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

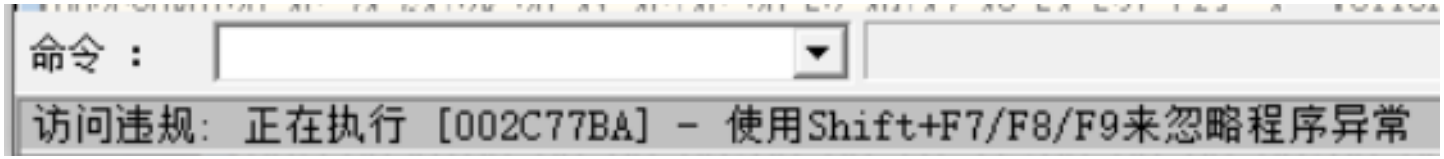


用x64dbg加载感染后的程序，可以看见，入口点已经修改为寄生的病毒体部分，后面还有一条JMP指令

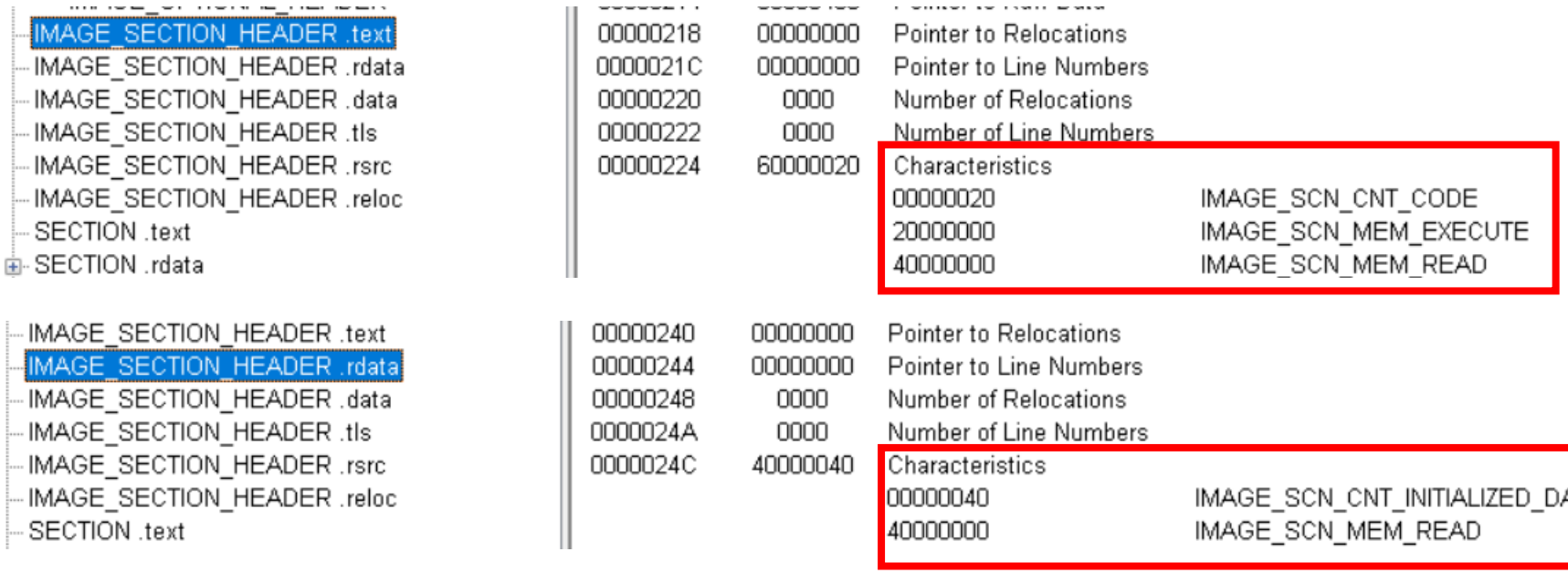


地址	HEX 数据	反汇编
002C77BA	90	NOP
002C77BB	90	NOP
002C77BC	90	NOP
002C77BD	90	NOP
002C77BE	90	NOP
002C77BF	- E9 2A62FCFF	JMP cloudmus.0028D9EE

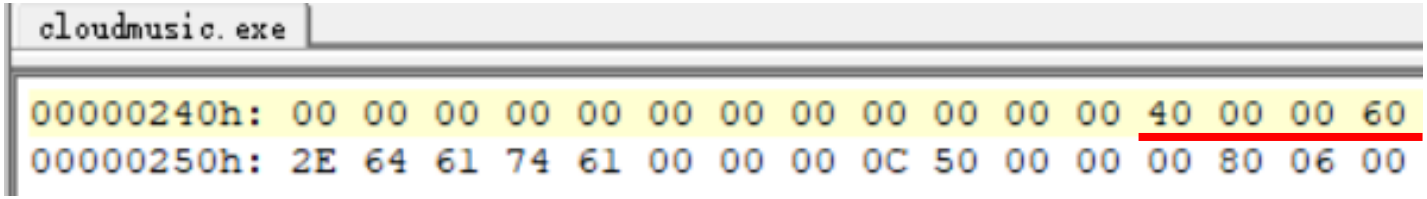
但是，当我们想要去运行代码，跳回原入口点时，却发现无法进行，并提示002C77BA处的指令NOP访问违规！这是什么原因呢？



我们来对比原来的.text节和现在病毒寄生的.rdata节的属性



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



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