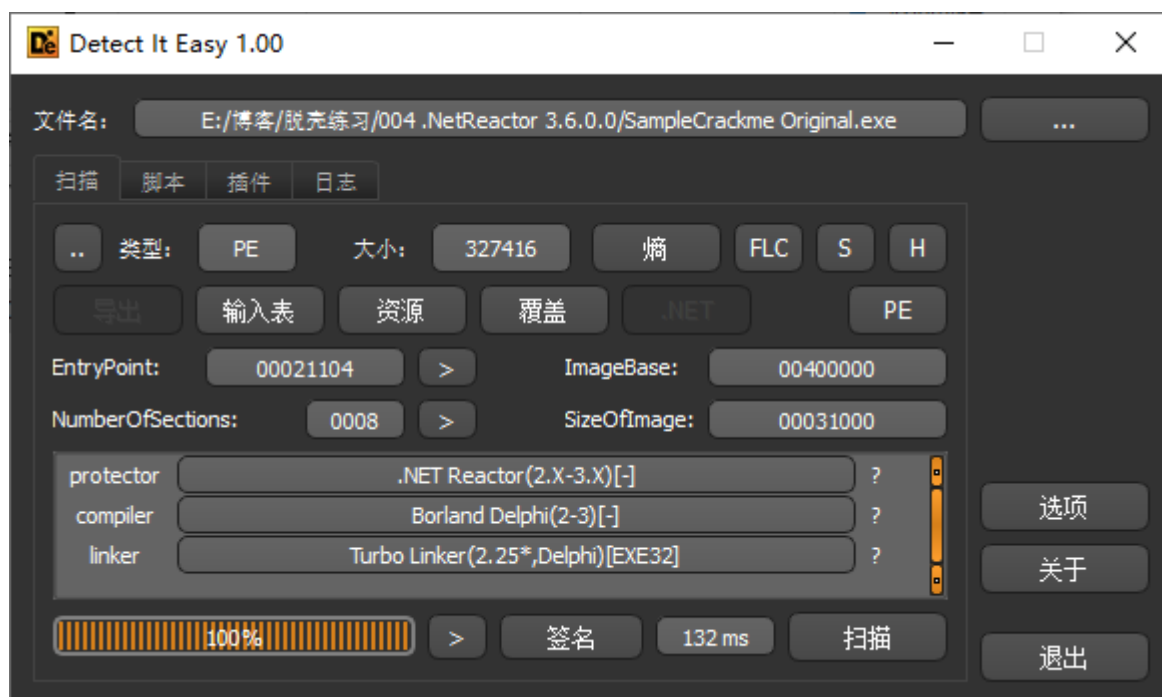


前言  
查壳  
脱壳  
修复目标程序

## 前言

最近一直在看脱壳的相关资料，看到了Tuts4you社区脱壳脚本的教程，这个壳我感觉很不错，挺有意思的，于是打算将内容整理下分享出来。

## 查壳



这个壳是.NetReactor 3.6.0.0的版本。根据作者的介绍，这个壳只是一个包装器，它包装目标程序，然后将其全部解包到内存中执行。但是这是一种不安全的方法，因为有人可以将内存中的目标程序转储回文件并完全恢复程序集。这个壳的重点在于转储之后的修复，需要对PE文件有一定的了解。

## 脱壳

接下来直接载入OD，F9让程序运行起来。

吾爱破解 - SampleCrackme Original.exe - [LCG - 主线程, 模块 - win32u]

文件(F) 查看(V) 调试(D) 插件(P) 选项(T) 窗口(W) 帮助(H) [+] 快捷菜单 Tools BreakPoint->

运行

| 地址       | HEX 数据      | 反汇编                        | 注释 |
|----------|-------------|----------------------------|----|
| 75502BEC | C3          | ret                        |    |
| 75502BED | 8D49 00     | lea ecx,dword ptr ds:[ecx] |    |
| 75502BF0 | B8 10100000 | mov eax,0x1010             |    |
| 75502BF5 | BA 60795075 | mov edx,win32u.75507960    |    |
| 75502BFA | FFD2        | call edx                   |    |
| 75502BFC | C2 0800     | ret 0x8                    |    |
| 75502BFF | 90          | nop                        |    |
| 75502C00 | B8 11100000 | mov eax,0x1011             |    |
| 75502C05 | BA 60795075 | mov edx,win32u.75507960    |    |
| 75502C0A | FFD2        | call edx                   |    |
| 75502C0C | C2 0800     | ret 0x8                    |    |
| 75502C0F | 90          | nop                        |    |
| 75502C10 | B8 12100000 | mov eax,0x1012             |    |
| 75502C15 | BA 60795075 | mov edx,win32u.75507960    |    |

返回到 54C8A188 (System\_W.54C8A188)

Sample Crackme

Name

Serial

| 地址       | HEX 数据  | UNICODE |
|----------|---|---------|
| 00422000 | 00 00 00 00 00 00 00 00 02 8D 40 00             | ..... 修 |
| 00422010 | 88 DC 40 00 00 23 41 00 9C DF 40 00 8C 0F 41 00 | @A @A   |

接着调出内存窗口，为了锁定目标程序被解压的位置，因为这个crackme实际上也是作者写的，所以选择通过搜索关键字串的方法，搜索Crackme

吾爱破解 - SampleCrackme Original.exe - [Memory map]

文件(F) 查看(V) 调试(D) 插件(P) 选项(T) 窗口(W) 帮助(H) [+] 快捷菜单 Tools BreakPoint->

运行

| 地址       | 大小       | 属主       | 区段 | 包含     | 类型   | 访问 | 初始访问 | 已映射为 |
|----------|----------|----------|----|--------|------|----|------|------|
| 00010000 | 00010000 |          |    |        | Map  | RW | RW   |      |
| 00020000 | 00002000 |          |    |        | Priv | RW | RW   |      |
| 00030000 | 00004000 |          |    |        | Map  | R  | R    |      |
| 00040000 | 0001A000 |          |    |        |      |    |      |      |
| 0008F000 | 00011000 |          |    |        |      |    |      |      |
| 000A1000 | 000EF000 |          |    |        |      |    |      |      |
| 00190000 | 00010000 |          |    |        |      |    |      |      |
| 001A0000 | 00004000 |          |    |        |      |    |      |      |
| 001B0000 | 00002000 |          |    |        |      |    |      |      |
| 001C0000 | 00001000 |          |    |        |      |    |      |      |
| 002F1000 | 00004000 |          |    |        |      |    |      |      |
| 002F5000 | 00001000 |          |    |        |      |    |      |      |
| 002FF000 | 00002000 |          |    |        |      |    |      |      |
| 00301000 | 00003000 |          |    |        |      |    |      |      |
| 00304000 | 00003000 |          |    |        |      |    |      |      |
| 00307000 | 00003000 |          |    |        |      |    |      |      |
| 0030A000 | 00003000 |          |    |        |      |    |      |      |
| 0030D000 | 00003000 |          |    |        |      |    |      |      |
| 00310000 | 00001000 |          |    |        |      |    |      |      |
| 00400000 | 00001000 | SampleCr |    | PE 文件头 | Imag | R  | RWE  |      |

输入要查找的二进制字符串

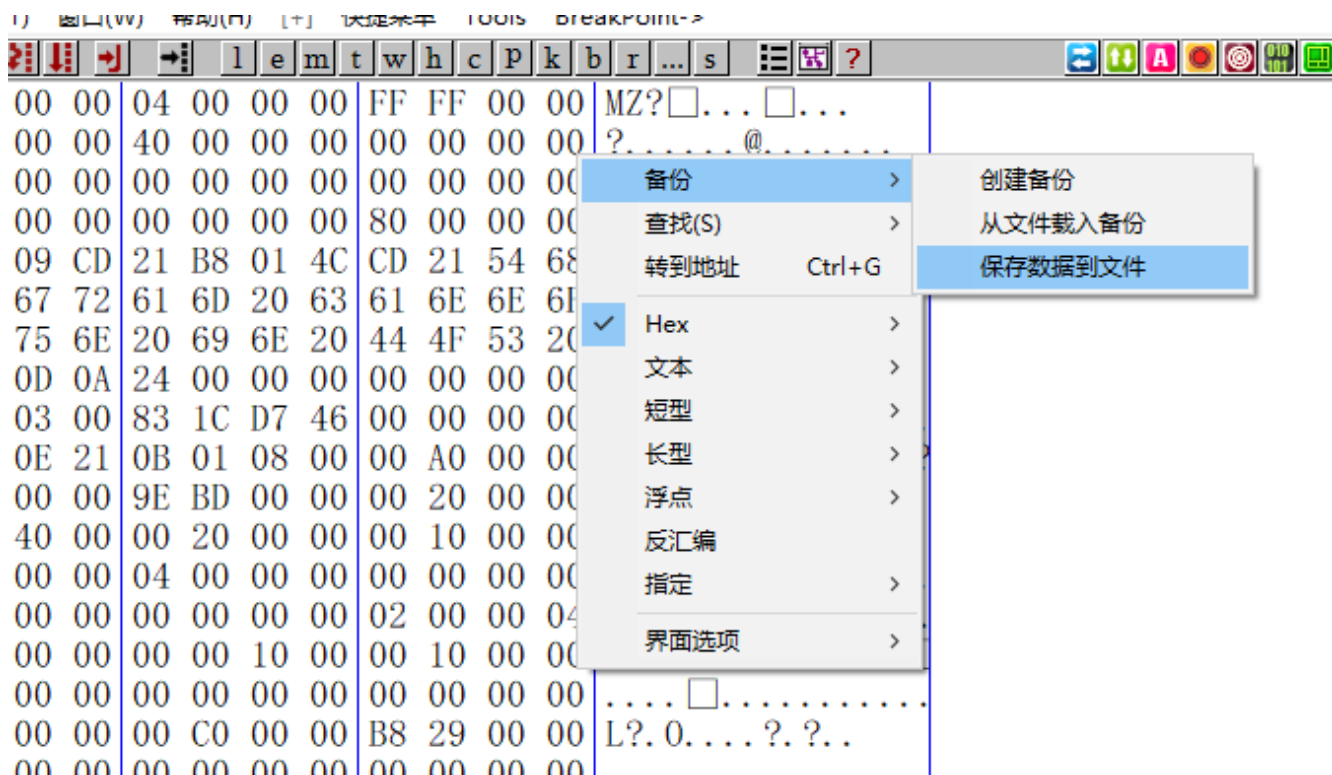
ASCII

UNICODE

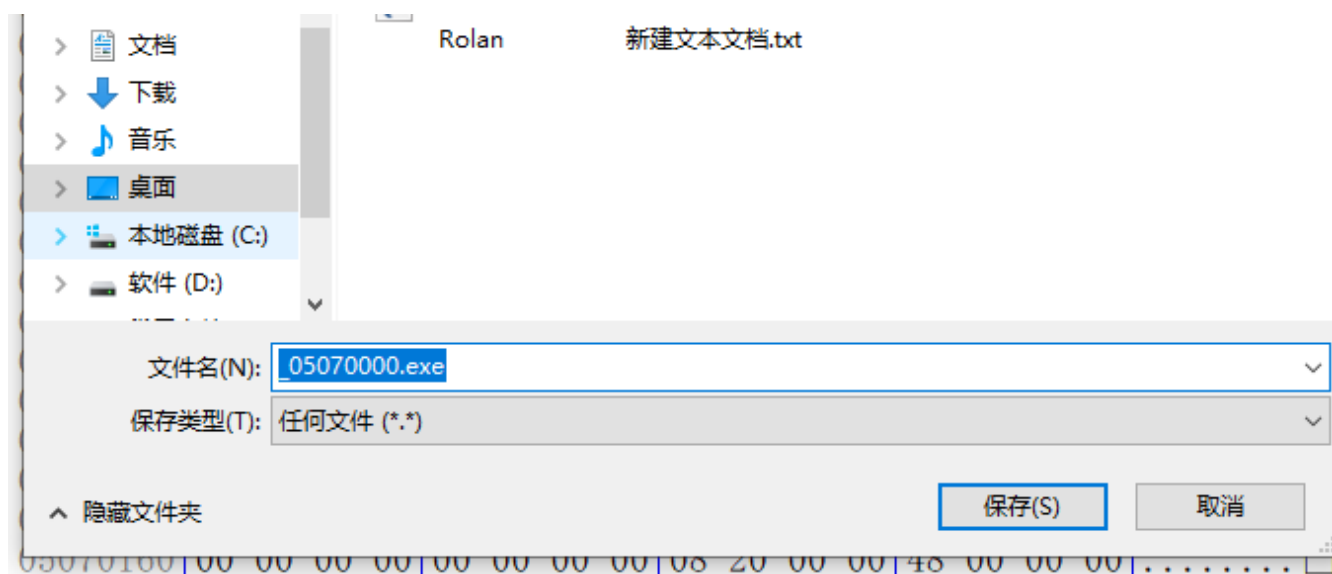
HEX +07

☒ 整个块 ☐ 区分大小写





接着右键->备份->保存数据到文件。



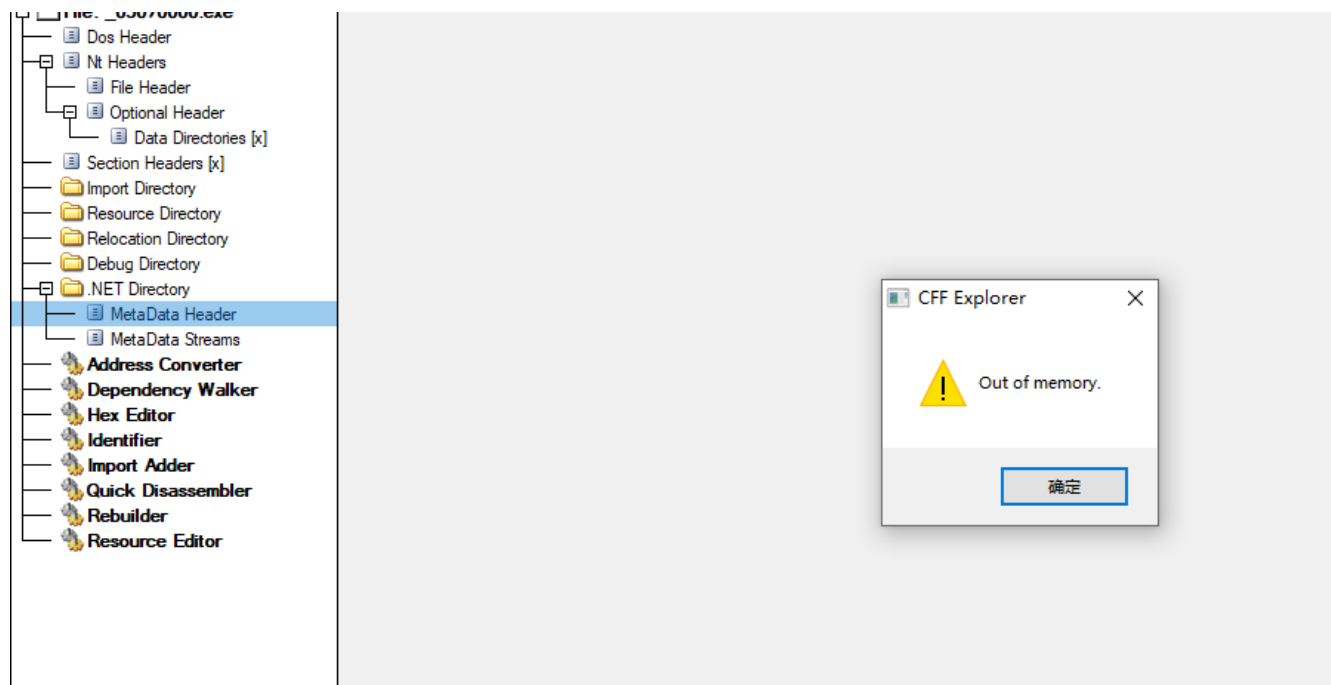
选择保存类型为任何文件，并修改后缀名为exe。



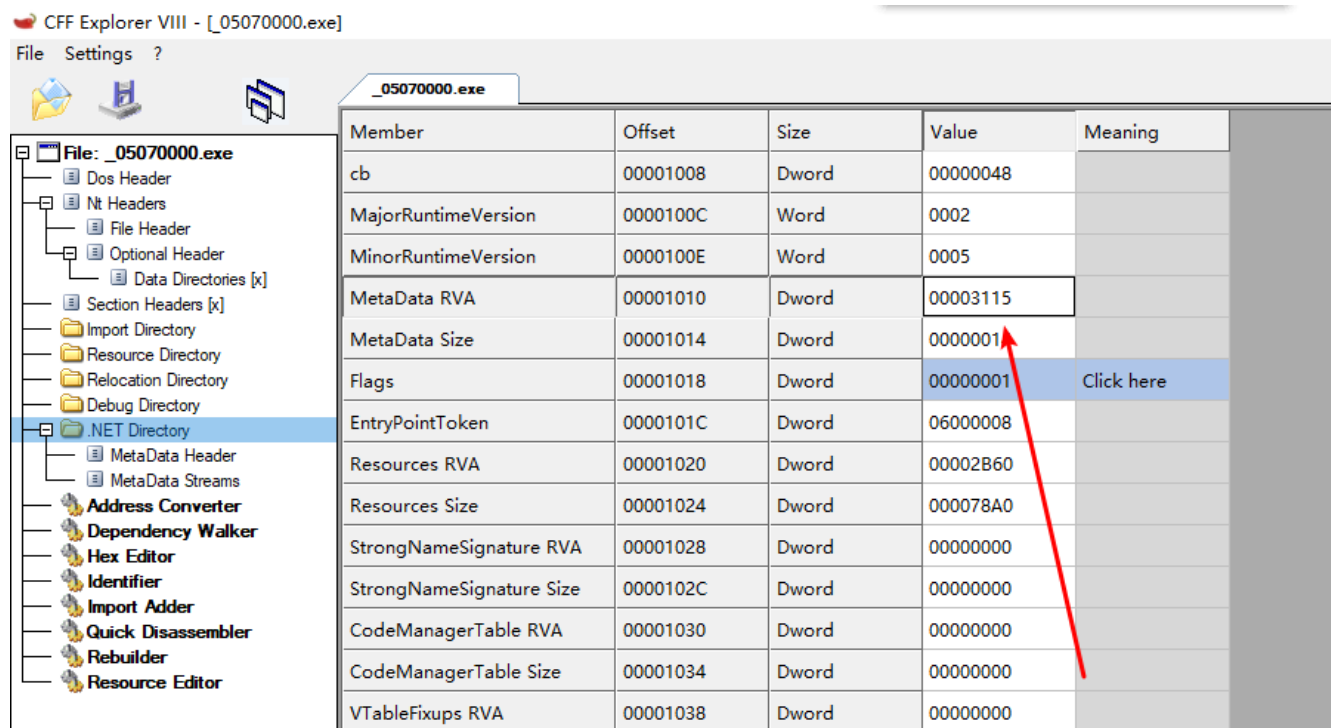
接着测试运行一下，弹出一个错误框，无法在电脑上运行。这很正常，因为直接dump下来的文件在PE头总是会出现问题。因为我是在本机上跑的，如果是W7的话应该是显示不是有效的W32程序。

# 修复目标程序

接下来用CFF Explorer这款PE工具来修复一下目标程序。



载入目标程序，点击Header部分，错误提示为Out of memory。



接下来进入到Driectory部分，修复MetaData Header的错误。我们需要修复这个错误的RVA和Size，Size明显是错的，太小了。

File: \_05070000.exe

Dos Header

Nt Headers

File Header

Optional Header

Data Directories [x]

Section Headers [x]

Import Directory

Resource Directory

Relocation Directory

Debug Directory

.NET Directory

MetaData Header

MetaData Streams

Address Converter

Dependency Walker

Hex Editor

Identifier

Import Addr

Quick Disassembler

Rebuilder

Resource Editor

RVA

File Offset

Find

String

BSJB

Find

Match Case

Unicode

Reset

Hex

Find

| Offset   | 0  | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 8  | 9  | A  | B  | C  | D  | E  | F  |   |
|----------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|---|
| 00000000 | 4D | 5A | 90 | 00 | 03 | 00 | 00 | 00 | 04 | 00 | 00 | 00 | FF | FF | 00 | 00 | M |
| 00000010 | B8 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 40 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | . |
| 00000020 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | . |
| 00000030 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 80 | 00 | 00 | 00 | . |
| 00000040 | 0E | 1F | BA | 0E | 00 | B4 | 09 | CD | 21 | B8 | 01 | 4C | CD | 21 | 54 | 68 | ! |
| 00000050 | 69 | 73 | 20 | 70 | 72 | 6F | 67 | 72 | 61 | 6D | 20 | 63 | 61 | 6E | 6E | 6F | i |
| 00000060 | 74 | 20 | 62 | 65 | 20 | 72 | 75 | 6E | 20 | 69 | 6E | 20 | 44 | 4F | 53 | 20 | t |
| 00000070 | 6D | 6F | 64 | 65 | 2E | 0D | 0D | 0A | 24 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | m |
| 00000080 | 50 | 45 | 00 | 00 | 4C | 01 | 03 | 00 | 83 | 1C | D7 | 46 | 00 | 00 | 00 | 00 | P |
| 00000090 | 00 | 00 | 00 | 00 | E0 | 00 | 0E | 21 | 0B | 01 | 08 | 00 | 00 | A0 | 00 | 00 | . |
| 000000A0 | 00 | 40 | 00 | 00 | 00 | 00 | 00 | 00 | 9E | BD | 00 | 00 | 00 | 20 | 00 | 00 | . |
| 000000B0 | 00 | C0 | 00 | 00 | 00 | 00 | 40 | 00 | 00 | 20 | 00 | 00 | 00 | 10 | 00 | 00 | . |

接着来到Address Converter部分，点击这个放大镜，查找字符串BSJB。至于为什么搜索这么一串字符串，作者给出的解释是这个字符串的Offset就是要修复的Meta Data的Offset。(我也是一脸蒙蔽 这解释有点太勉强了吧)

Find

String

BSJB

Find

Match Case

Unicode

Reset

Hex

Find

Status: String found

| Offset   | 0  | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 8  | 9  | A  | B  | C  | D  | E  | F  | Ascii              |
|----------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|--------------------|
| 00009400 | 42 | 53 | 4A | 42 | 01 | 00 | 01 | 00 | 00 | 00 | 00 | 0C | 00 | 00 | 00 | 00 | BSJB ... ...       |
| 00009410 | 76 | 32 | 2E | 30 | 2E | 35 | 30 | 37 | 32 | 37 | 00 | 00 | 00 | 00 | 05 | 00 | v2.0.50727...      |
| 00009420 | 6C | 00 | 00 | 00 | 68 | 07 | 00 | 00 | 23 | 7E | 00 | 00 | D4 | 07 | 00 | 00 | l...h ...#~... ... |
| 00009430 | C4 | 09 | 00 | 00 | 23 | 53 | 74 | 72 | 69 | 6E | 67 | 73 | 00 | 00 | 00 | 00 | Ä...#Strings...    |
| 00009440 | 98 | 11 | 00 | 00 | 60 | 04 | 00 | 00 | 23 | 55 | 53 | 00 | F8 | 15 | 00 | 00 | ... ...#US... ...  |
| 00009450 | 10 | 00 | 00 | 00 | 23 | 47 | 55 | 49 | 44 | 00 | 00 | 00 | 08 | 16 | 00 | 00 | ...#GUID...  ...   |
| 00009460 | C8 | 02 | 00 | 00 | 23 | 42 | 6C | 6F | 62 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | È...#Blob... ...   |
| 00009470 | 02 | 00 | 00 | 01 | 57 | 15 | A2 | 01 | 09 | 01 | 00 | 00 | 00 | FA | 01 | 33 | ...W e ...ú 3      |
| 00009480 | 00 | 16 | 00 | 00 | 01 | 00 | 00 | 00 | 3F | 00 | 00 | 00 | 07 | 00 | 00 | 00 | ... ...?... ...    |
| 00009490 | 14 | 00 | 00 | 00 | 16 | 00 | 00 | 00 | 0E | 00 | 00 | 00 | 4F | 00 | 00 | 00 | ... ...!... ...    |
| 000094A0 | 4F | 00 | 00 | 00 | 0F | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | ... ...!... ...    |

接着点击查找，找到了0x9400这个地址，那么Meta Data的Offset就是0x9400。



|             |          |
|-------------|----------|
| VA          | 0040A400 |
| RVA         | 0000A400 |
| File Offset | 00009400 |

接着把9400输入到Offset中，会自动计算出我们要的RVA是0xA400。

|                      |                          |          |       |          |            |
|----------------------|--------------------------|----------|-------|----------|------------|
| File Settings ?      |                          |          |       |          |            |
| _05070000.exe        |                          |          |       |          |            |
| File: _05070000.exe  | Member                   | Offset   | Size  | Value    | Meaning    |
| Dos Header           | cb                       | 00001008 | Dword | 00000048 |            |
| Nt Headers           | MajorRuntimeVersion      | 0000100C | Word  | 0002     |            |
| File Header          | MinorRuntimeVersion      | 0000100E | Word  | 0005     |            |
| Optional Header      | MetaData RVA             | 00001010 | Dword | 0000A400 |            |
| Data Directories [x] | MetaData Size            | 00001014 | Dword | 00000014 |            |
| Section Headers [x]  | Flags                    | 00001018 | Dword | 00000001 | Click here |
| Import Directory     | EntryPointToken          | 0000101C | Dword | 06000008 |            |
| Resource Directory   | Resources RVA            | 00001020 | Dword | 00002B60 |            |
| Relocation Directory | Resources Size           | 00001024 | Dword | 000078A0 |            |
| Debug Directory      | StrongNameSignature RVA  | 00001028 | Dword | 00000000 |            |
| .NET Directory       | StrongNameSignature Size | 0000102C | Dword | 00000000 |            |
| MetaData Header      | CodeManagerTable RVA     | 00001030 | Dword | 00000000 |            |
| MetaData Streams     |                          |          |       |          |            |
| Address Converter    |                          |          |       |          |            |
| Dependency Walker    |                          |          |       |          |            |
| Hex Editor           |                          |          |       |          |            |
| Identifier           |                          |          |       |          |            |
| Import Adder         |                          |          |       |          |            |
| Quick Disassembler   |                          |          |       |          |            |

回到Directory部分，将正确的RVA填入。至于Size我们可以根据一个公式计算得出:  $\text{MetaDataSize} = \text{Import Directory RVA} - \text{MetaDataRVA}$ , Import Directory RVA的值如下图：

File Settings ?

File: \_05070000.exe

- Dos Header
- Nt Headers
  - File Header
  - Optional Header
  - Data Directories [x]
- Section Headers [x]
- Import Directory
- Resource Directory
- Relocation Directory
- Debug Directory
- .NET Directory
  - MetaData Header
  - MetaData Streams
- Address Converter
- Dependency Walker
- Hex Editor
- Identifier
- Import Adder
- Quick Disassembler
- Rebuilder
- Resource Editor

| Member                    | Offset   | Size  | Value    | Section |
|---------------------------|----------|-------|----------|---------|
| Export Directory RVA      | 000000F8 | Dword | 00000000 |         |
| Export Directory Size     | 000000FC | Dword | 00000000 |         |
| Import Directory RVA      | 00000100 | Dword | 0000BD4C | .text   |
| Import Directory Size     | 00000104 | Dword | 0000004F |         |
| Resource Directory RVA    | 00000108 | Dword | 0000C000 | .rsrc   |
| Resource Directory Size   | 0000010C | Dword | 000029B8 |         |
| Exception Directory RVA   | 00000110 | Dword | 00000000 |         |
| Exception Directory Size  | 00000114 | Dword | 00000000 |         |
| Security Directory RVA    | 00000118 | Dword | 00000000 |         |
| Security Directory Size   | 0000011C | Dword | 00000000 |         |
| Relocation Directory RVA  | 00000120 | Dword | 00010000 | .reloc  |
| Relocation Directory Size | 00000124 | Dword | 0000000C |         |
| Debug Directory RVA       | 00000128 | Dword | 0000BCD0 | .text   |
| Debug Directory Size      | 0000012C | Dword | 0000001C |         |

最后算出Size为0x194C。

File Settings ?

File: \_05070000.exe

- Dos Header
- Nt Headers
  - File Header
  - Optional Header
  - Data Directories [x]
- Section Headers [x]
- Import Directory
- Resource Directory
- Relocation Directory
- Debug Directory
- .NET Directory
  - MetaData Header
  - MetaData Streams
- Address Converter
- Dependency Walker
- Hex Editor
- Identifier
- Import Adder
- Quick Disassembler
- Rebuilder
- Resource Editor

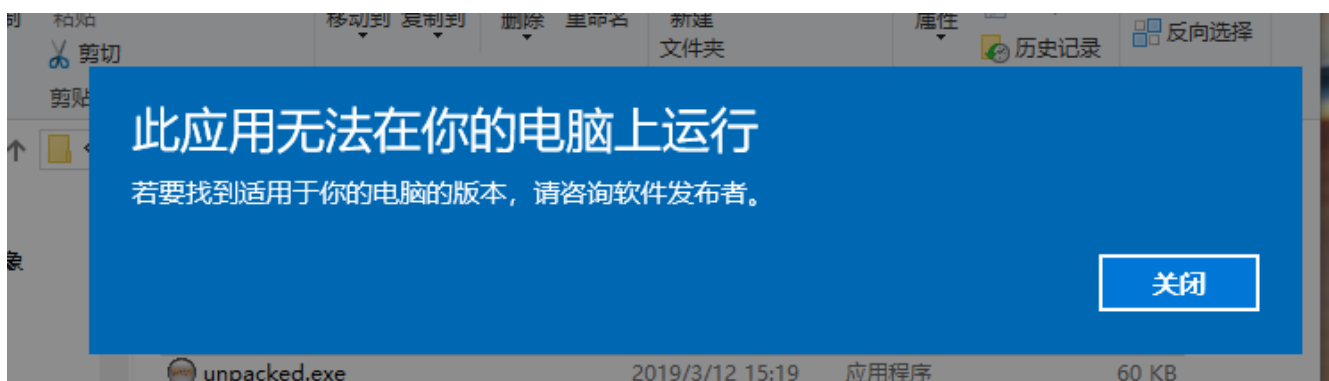
| Member                   | Offset   | Size  | Value    | Meaning    |
|--------------------------|----------|-------|----------|------------|
| cb                       | 00001008 | Dword | 00000048 |            |
| MajorRuntimeVersion      | 0000100C | Word  | 0002     |            |
| MinorRuntimeVersion      | 0000100E | Word  | 0005     |            |
| MetaData RVA             | 00001010 | Dword | 0000A400 |            |
| MetaData Size            | 00001014 | Dword | 0000194C |            |
| Flags                    | 00001018 | Dword | 00000001 | Click here |
| EntryPointToken          | 0000101C | Dword | 06000008 |            |
| Resources RVA            | 00001020 |       |          |            |
| Resources Size           | 00001024 |       |          |            |
| StrongNameSignature RVA  | 00001028 |       |          |            |
| StrongNameSignature Size | 0000102C |       |          |            |
| CodeManagerTable RVA     | 00001030 |       |          |            |
| CodeManagerTable Size    | 00001034 | Dword | 00000000 |            |
| VTableFixups RVA         | 00001038 | Dword | 00000000 |            |
| VTableFixups Size        | 0000103C | Dword | 00000000 |            |

CFF Explorer

Overwrite original file?

是(Y) 否(N) 取消

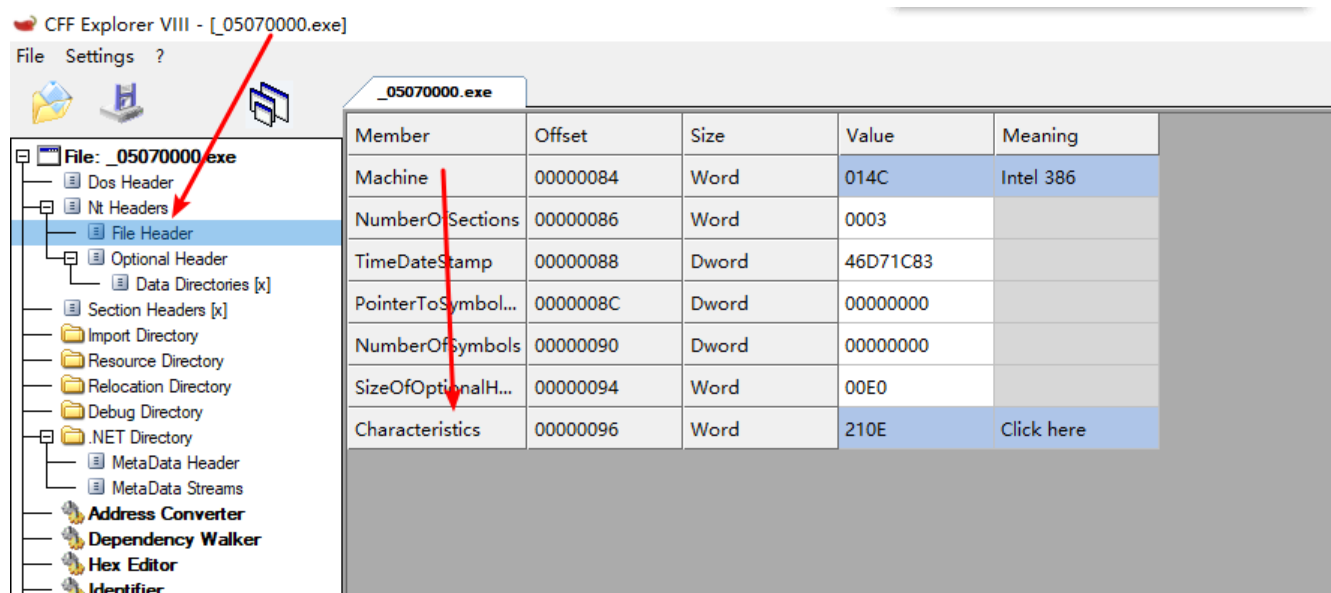
接着修改回正确的RVA，然后点击保存。



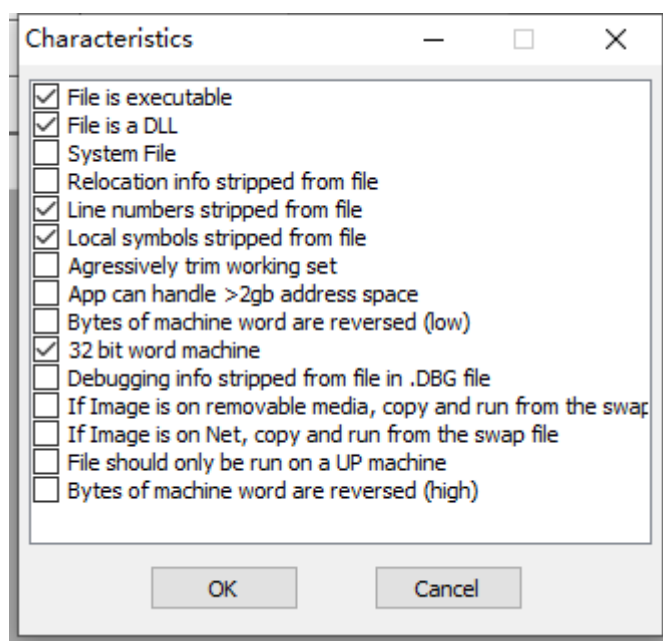


接着再次双击测试运行，还是无法运行。这里作者的原话是根据我之前的经验，我应该是忘记修改文件头属性了。

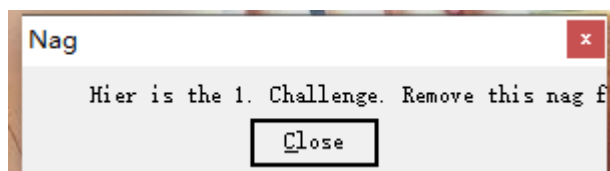
不得不感叹大神的经验就是强大。好吧 继续修复

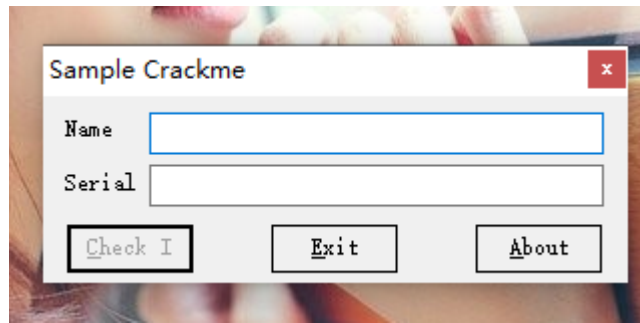


点击文件头 找到Characteristics，双击



属性显示这是一个DLL，难怪会报错。把勾去掉，再次保存。





OK 程序完美运行，这个壳也算是脱完了。

需要相关文件可以到我的Github下载:<https://github.com/TonyChen56/Unpack-Practice>