سوف نتطرق الأن إلي معرفة كيفية إضافة كود لبرنامجنا وذلك عن طريق إضافة إلي قسم الـCODE وبإستخدام برنامج LordPE سوف تري المعلومات المتوفرة عن هذا القسم كما يلي :

[ Section Table ]

| Name | VOffset | VSize | ROffset | RSize | Flags |
|---|---|---|---|---|---|
| CODE | 00001000 | 00057FCC | 00000400 | 00058000 | 60000020 |
| DATA | 00059000 | 0000111C | 00058400 | 00001200 | C0000040 |
| BSS | 0005B000 | 00000C51 | 00059600 | 00000000 | C0000000 |
| .idata | 0005C000 | 000021BC | 00059600 | 00002200 | C0000040 |
| .tls | 0005F000 | 00000010 | 0005B800 | 00000000 | C0000000 |
| .rdata | 00060000 | 00000018 | 0005B800 | 00000200 | 50000040 |
| .reloc | 00061000 | 00006108 | 0005BA00 | 00006200 | 50000040 |
| .rsrc | 00068000 | 00003A00 | 00061C00 | 00003A00 | 50000040 |

الـVirtualSize في الشكل السابق يوضح حجم البيانات المستخدمة في هذا القسم وليس حجم القسم ككل ولكن نحن نريد أن نجد مساحة لكي نُضيف كود ما في هذا البرنامج وإذا قمنا بجمع قيمة الـRSize مع الـROffset سوف يكون الناتج 00058400 وإذا لاحظت أن هذا الناتج هو بداية الـROffset لقسم الـDATA وإذا ذهبنا إلي هذا العنوان في برنامج Hex Workshop سوف تري هذا الشكل :

```
000583B0 8A 45 00 E8 B8 E5 FF FF A1 B8 9F 45 00 8B 00 E8
000583C0 2C E6 FF FF E8 FB B1 FA FF 8D 40 00 00 00 00 00
000583D0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000583E0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000583F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00058400 00 00 00 00 00 00 00 00 02 8D 40 00 32 13 8B C0
00058410 02 00 8B C0 00 8D 40 00 00 8D 40 00 00 8D 40 00
00058420 01 8D 40 00 00 00 00 00 00 00 00 00 54 21 40 00
```

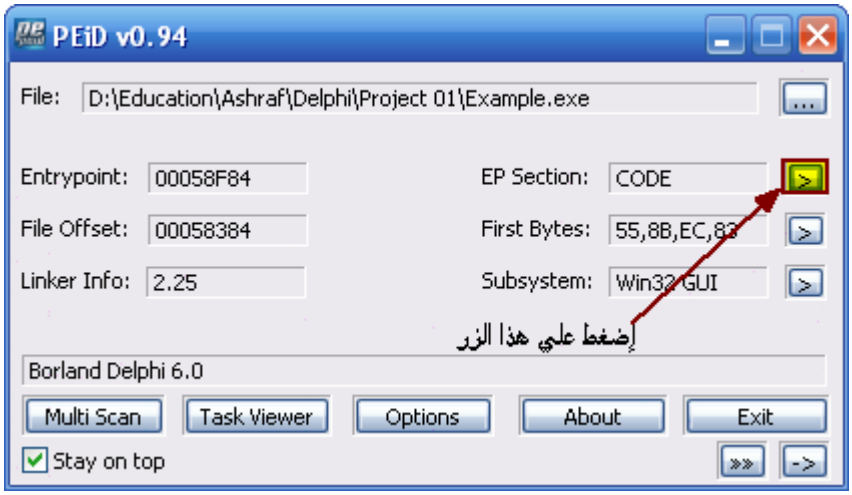
48 بايت لا يشغلون أي كود ومن الممكن إضافة الكود الذي نرغبه في اي بايت من هذة البايتات

بداية قسم الـDATA

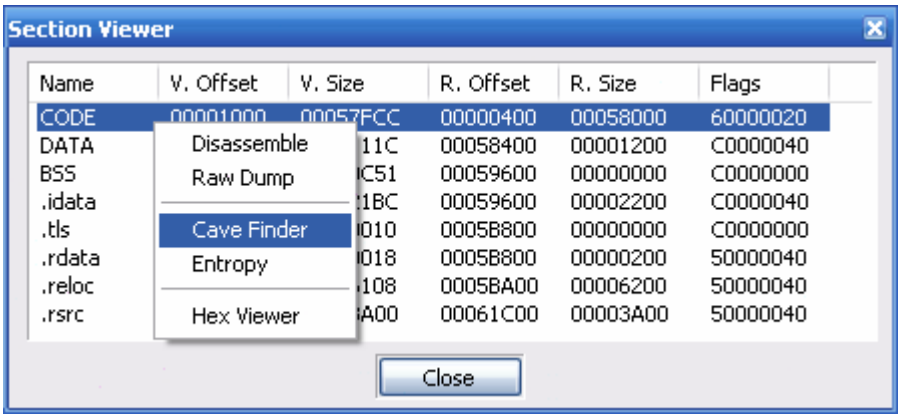
كما تري ففي الشكل السابق توجد مساحة إضافية حوالي ٤٨ بايت غير مستخدمة في الـMemory وهذه المساحة تُتيح لنا إضافة الكود الذي نرغب فيه ولكن نحن نريد أن نتأكد من أن الكود الذي سوف نضيفة سوف يكون له تأثير في البرنامج لذلك سوف نقوم بتزويد مساحة الـVSize لكي يشمل الكود الذي نُصيفة ويكون له تأثير في البرنامج فقيمة الـVsize الحالية هي 00057FCC وسوف نقوم بتزويدها لكي تصبح 00057FFF وهذه هي أكبر قيمة ولكن لماذا إهتدينا إلي هذه القيمة فإذا نظرنا إلي الـRSize سوف نجدها 00058000 وهذه حجم القسم ككل ولكن نحن نريد أكبر قيمة

0005800 لكـي يـصبح الـVSize 00057FFF

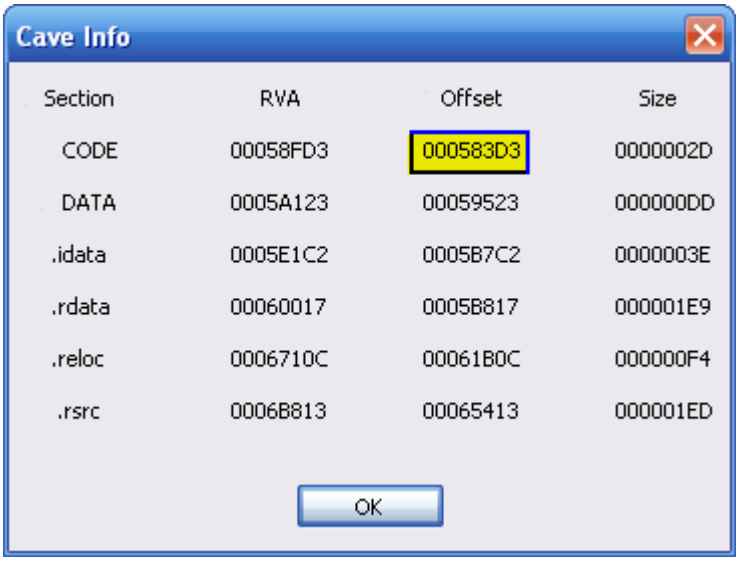
وبذلك نستفيد بأقصي مساحة ممكنة من قسم الـCODE وهناك طريقة أخري لحساب أكبر مساحة ممكنه لكي نضيف إليها الكود الذي نريدة وذلك عن طريق برنامج PEiD فقم بتشغيل هذا البرنامج ثم إخنر برنامجنا ثم إضغط علي الزر المشار إليه في الشكل التالي :



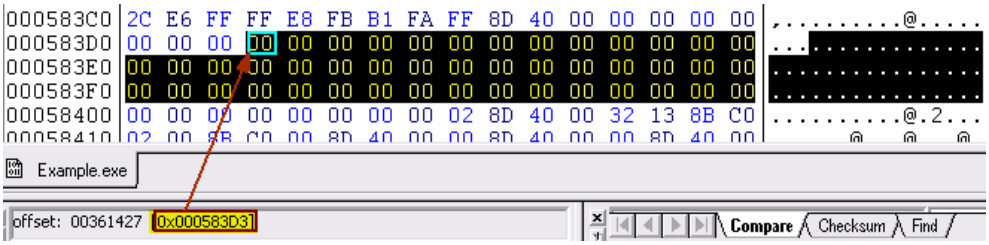

ثم إضغط علي قسم الـCODE ثم إختر هذا الإختيار :



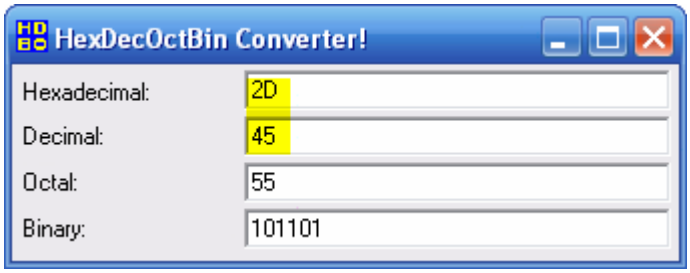

وبعد الضغط سوف يظهر لك ديالوج خاص بالمساحة الممكن أن تُضيف إليها الكود الذي ترغب فيها كما يلي :




كما تري فهذا الـOffset الخاص بالمساحة الخالية وإذا ذهبنا إليه في برنامج Hex Workshop سوف تري هذا الشكل :




كما تري فالشكل السابق يوضح الـOffset الخاص بأول بايت يمكن إضافة كود له وإذا أحصيت عدد البايتات سوف تجدها ٤٥ بايت وهذا ما كان يشير إليه الـSize الموجود في الشكل الخاص ببرنامج PEiD فلو رأيت الـSize سوف تجده 2D وإذا قمت بتحويل هذه القيمة إلي Decimal سوف تري القيمة ٤٥ كما في الشكل التالي :



وإذا إنتبهت إلي الـRVA الموجود أيضاً في الشكل الخاص ببرنامج PEiD سوف تجده 00058FD3 وإذا قمت بإضافة الـImageBase له سوف ينتج الـ Virtual Address وهو 00458FD3 وإذا ذهبت إلي برنامج Ollydbg عند هذا العنوان سوف تري هذا الشكل :

```
00458FD2  .  0000    ADD BYTE PTR DS:[EAX],AL
00458FD4  .  0000    ADD BYTE PTR DS:[EAX],AL
00458FD6  .  0000    ADD BYTE PTR DS:[EAX],AL
00458FD8  .  0000    ADD BYTE PTR DS:[EAX],AL
00458FDA  .  0000    ADD BYTE PTR DS:[EAX],AL
00458FDC  .  0000    ADD BYTE PTR DS:[EAX],AL
00458FDE  .  0000    ADD BYTE PTR DS:[EAX],AL
00458FE0  .  0000    ADD BYTE PTR DS:[EAX],AL
00458FE2  .  0000    ADD BYTE PTR DS:[EAX],AL
00458FE4  .  0000    ADD BYTE PTR DS:[EAX],AL
00458FE6  .  0000    ADD BYTE PTR DS:[EAX],AL
00458FE8  .  0000    ADD BYTE PTR DS:[EAX],AL
00458FEA  .  0000    ADD BYTE PTR DS:[EAX],AL
00458FEC  .  0000    ADD BYTE PTR DS:[EAX],AL
00458FEE  .  0000    ADD BYTE PTR DS:[EAX],AL
00458FF0  .  0000    ADD BYTE PTR DS:[EAX],AL
00458FF2  .  0000    ADD BYTE PTR DS:[EAX],AL
00458FF4  .  0000    ADD BYTE PTR DS:[EAX],AL
00458FF6  .  0000    ADD BYTE PTR DS:[EAX],AL
00458FF8  .  0000    ADD BYTE PTR DS:[EAX],AL
00458FFA  .  0000    ADD BYTE PTR DS:[EAX],AL
00458FFC  .  0000    ADD BYTE PTR DS:[EAX],AL
00458FFE  .  0000    ADD BYTE PTR DS:[EAX],AL
```

كما تري فالشكل السابق يوضح البايتات الممكن إضافة الكود لها وبعد ما عرفنا كيفية معرفة المساحة سوف نعرف كيف نُضيف الكود وإذا ذهبنا إلي المنطقة المتاح لنا إضافة الأكواد بها قي برنامج Hex Workshop سوف نري الأتي :

```
000583D0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000583E0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000583F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00058400 00 00 00 00 00 00 00 00 02 8D 40 00 32 13 8B C0
```

البايت المشار إليه في الشكل السابق هو الذي نريد أن نبدأ به الكود ونحن الأن نريد أن نحول هذا الـOffset إلي Virtual Address لكي نذهب إلي برنامج Ollydbg ونُضيف ما نريد والمعادلة التي سوف يتم التحويل بها هي :

Raw Offset + (VOffset of section – ROffset of section) + ImageBase = Virtual Address

وسوف يكون التحويل كالتالي :

000583E0 + (1000 – 400) + 400000 = **00458FE0**

وسوف نذهب إلي برنامج إلي برنامج Ollydbg لهذا العنوان كما يلي :

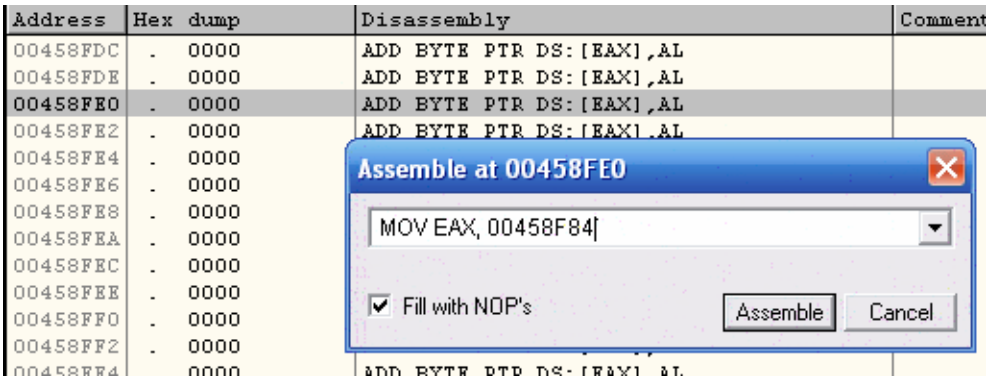
```
00458FDE  .  0000   ADD BYTE PTR DS:[EAX],AL
00458FE0  .  0000   ADD BYTE PTR DS:[EAX],AL
00458FE2  .  0000   ADD BYTE PTR DS:[EAX],AL
00458FE4  .  0000   ADD BYTE PTR DS:[EAX],AL
00458FE6  .  0000   ADD BYTE PTR DS:[EAX],AL
00458FE8  .  0000   ADD BYTE PTR DS:[EAX],AL
00458FEA  .  0000   ADD BYTE PTR DS:[EAX],AL
00458FEC  .  0000   ADD BYTE PTR DS:[EAX],AL
00458FEE  .  0000   ADD BYTE PTR DS:[EAX],AL
```

العنوان المشار إليه في الشكل السابق هو العنوان الذي سوف نُضيف له الكود وسوف نقوم بإضافة هذا الكود :

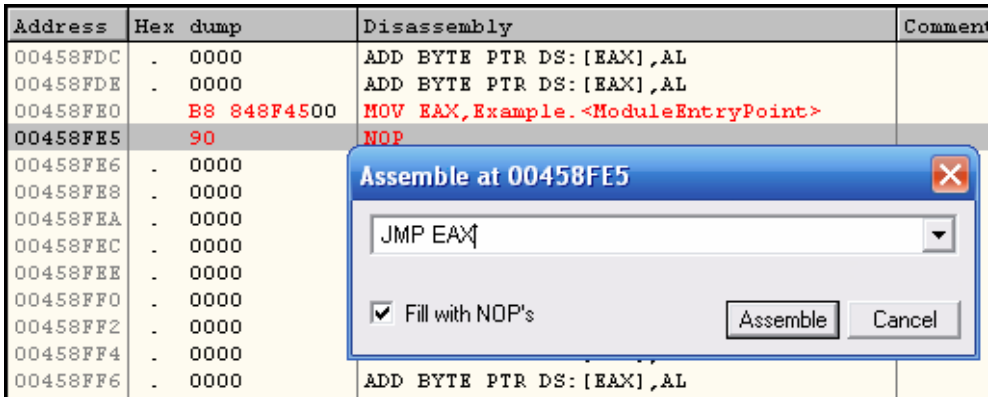
MOV EAX, 00458F84

JMP EAX

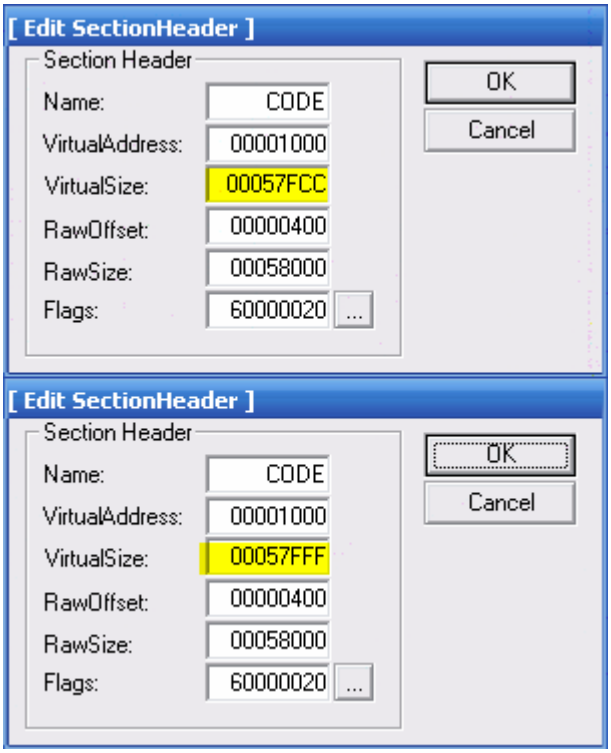
السطر الأول يفيد بأننا سوف ننقل الـOEP إلي المسجل EAX والسطر الثاني معناه بأننا سوف نقفز إلي الـOEP ومن ثم إكمال البرنامج بطريقة عادية جداً وسوف يتم ذلك عن طريق الوقوف علي العنوان المشار إليه في الشكل السابق ثم الضغط علي مفتاح space من لوحة المفاتيح ثم إضافة هذا الكود :



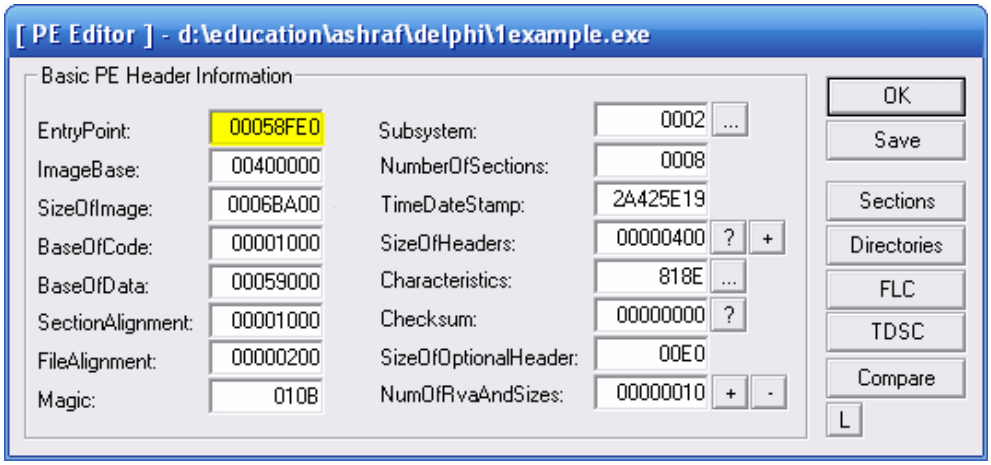
ثم الضغط علي زر Assemble ثم إضافة هذا الكود :



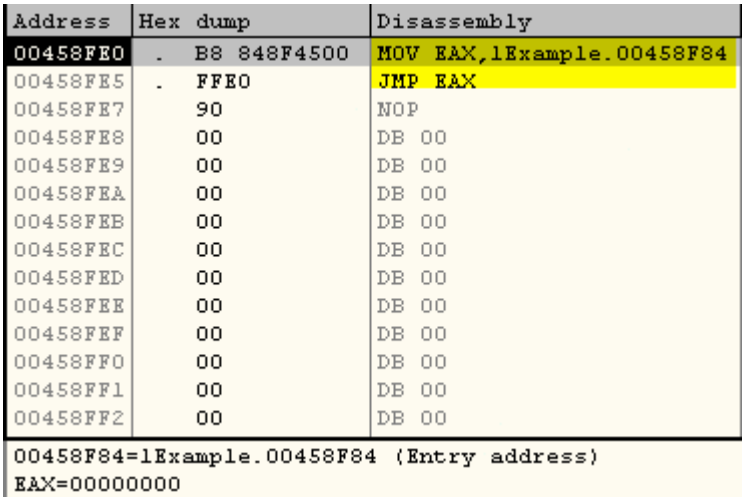

ثم الضغط علي زر Assemble ثم زر Cancel ثم نقوم بحفظ هذه الأكواد ثم نذهب إلي برنامج LordPE ثم نغير قيمة الـVSize إلي هذه :




ولقد قمنا بتزويد حجم بيانات القسم لكي يكون الكود الذي نضيفة له تأثير علي البرنامج وسوف نغير قيمة الـEntry Point إلي هذه القيمة :



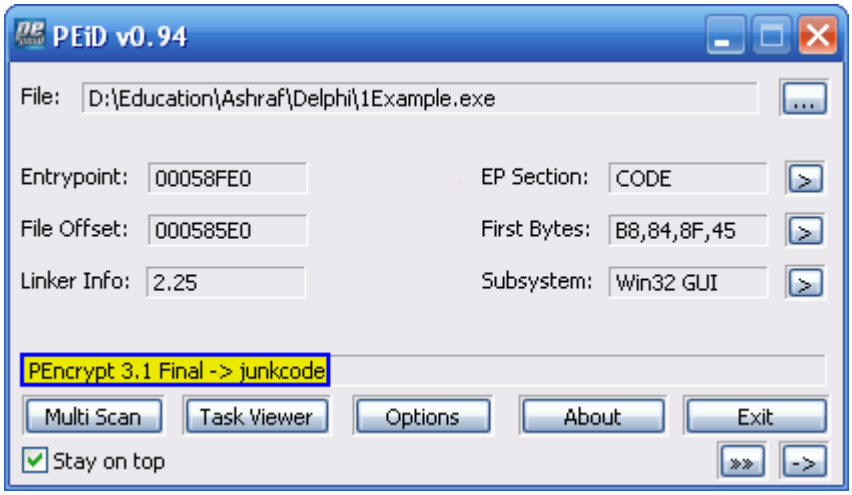

لأن هذه القيمة هي الـRVA الخاص بالـOEP الجديد ومن هذه النقطة سوق يبدأ البرنامج وإذا قمت بتحميل برنامجنا في برنامج Ollydbg سوف تري هذا الشكل :



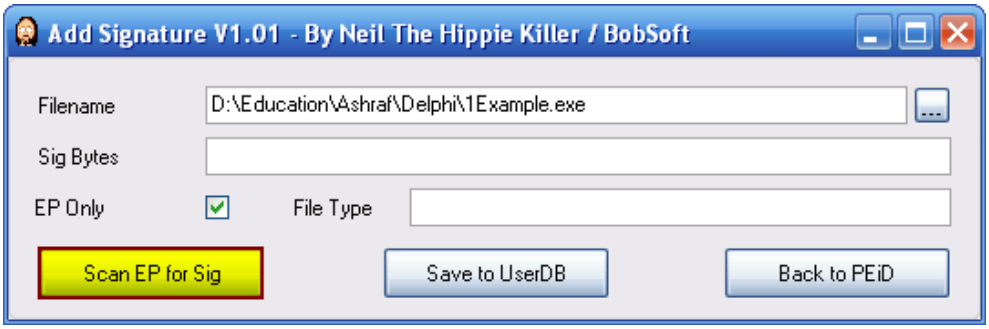
| Address | Hex dump | Disassembly |
|---|---|---|
| 00458FE0 | . B8 848F4500 | MOV EAX,1Example.00458F84 |
| 00458FE5 | . FFE0 | JMP EAX |
| 00458FE7 | 90 | NOP |
| 00458FE8 | 00 | DB 00 |
| 00458FE9 | 00 | DB 00 |
| 00458FEA | 00 | DB 00 |
| 00458FEB | 00 | DB 00 |
| 00458FEC | 00 | DB 00 |
| 00458FED | 00 | DB 00 |
| 00458FEE | 00 | DB 00 |
| 00458FEF | 00 | DB 00 |
| 00458FF0 | 00 | DB 00 |
| 00458FF1 | 00 | DB 00 |
| 00458FF2 | 00 | DB 00 |

00458F84=1Example.00458F84 (Entry address)
EAX=00000000

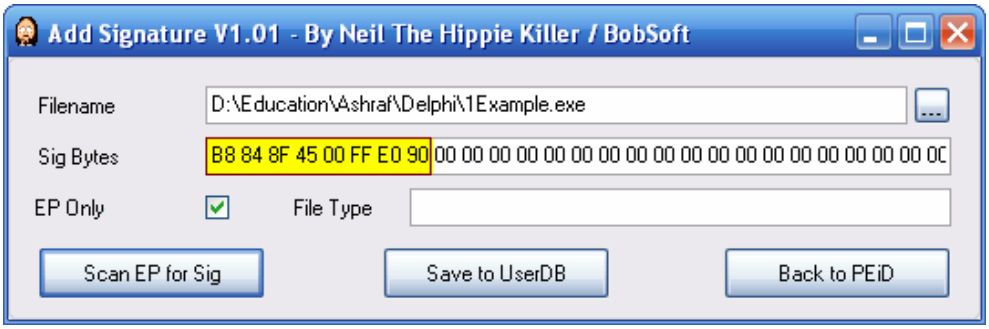
كما تري فعند تحميل البرنامج في برنامج Ollydbg تم تحميلة أتوماتيكياً علي العنوان الذي وضعناه في خانة الـEntry Point وإذا ضغطت F8 لتتبع البرنامج سوف تري أن البرنامج سوف ينتقل إلي السطر الثاني ثم إلي الـOEP ثم يُكمل البرنامج خطواته كما هي وبذلك نكون قد عرفنا كيفية إضافة كود وتزيد قيمة حجم بيانات القسم لكي يتم تأثير الكود الذي سوف نُضيفة علي البرنامج وتغيير الـEP إلي العنوان الجديد الذي سوف يبدأ منه البرنامج وإذا عملت مسح علي هذا البرنامج ببرنامج PEiD سوف تري هذا الشكل :




كما تري الشكل السابق يدل علي أن هذا البرنامج مشفر ولكن البرنامج غير مشفر ولكن هذا التغيير لأننا غيرنا في البرنامج من حيث الـEP وأيضاً الكود الذي يبدأ به البرنامج وإذا ذهبت إلي قائمة الـPlugins ثم إخترت Add Signature ثم ضغطت علي هذا الزر :




سوف تري هذه القيمة :

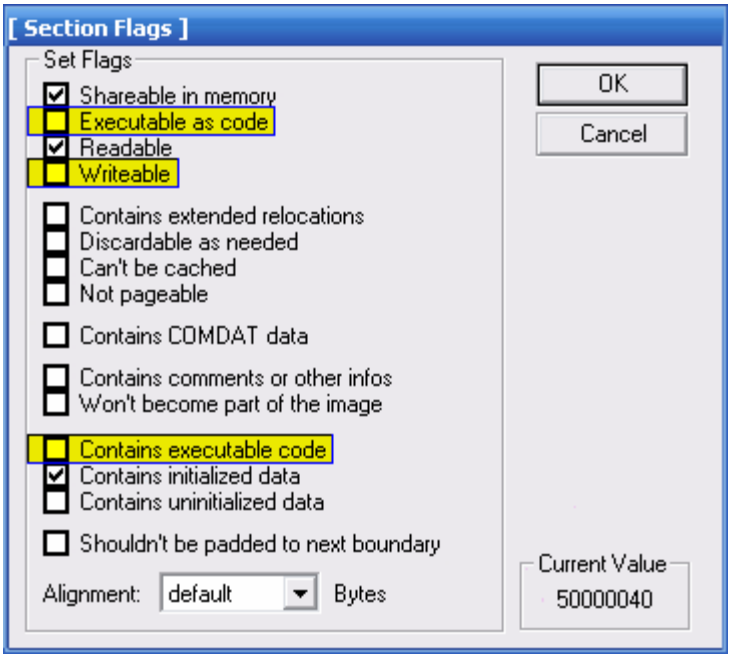


كما تري فهذه الأكواد التي قمنا بإضافتها وهذا البرنامج يري أن هذه الأكواد هي الخاصة بحماية PEncrypt 3.1 Final ولكن طبعاً البرنامج غير مشفر وهذه خدعة منّا ومن الممكن أن نستخدم هذه الخدعة لتضليل أي شخص وإشعارة بأن هذا البرنامج مشفر. ولكن إذا كان القسم ممتلئ بالبيانات ونحن نريد أن نُضيف بيانات فسوف نقوم بتوسيع القسم فمثلاً إذا أردنا توسيع قسم الـDATA فسوف تقابلنا عدة مشاكل :-

١- فقسم الـDATA توجد أقسام أخري تلية ولوقمنا بتوسيع هذا القسم فسوف نقوم بتحريك الأقسام التي تلية أيضاً.

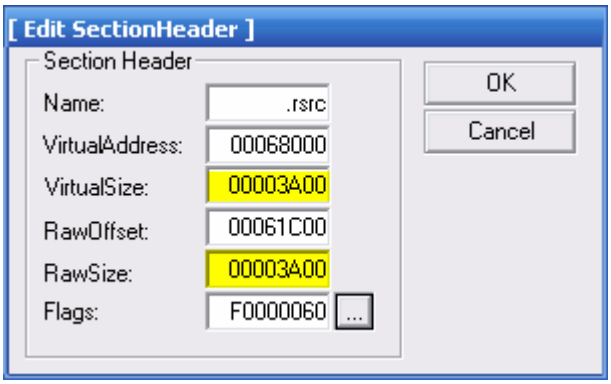
٢- توسيع هذا القسم سوف يعمل علي تغيير بعض خصائص الملف مثل حجم الملف.

ولكن هذه المشاكل من الممكن تفاديها عند التغيير في أخر قسم وهو rsrc وإذا إستعرضنا خصائص هذا القسم ببرنامج LordPE سوف نري هذا الشكل :

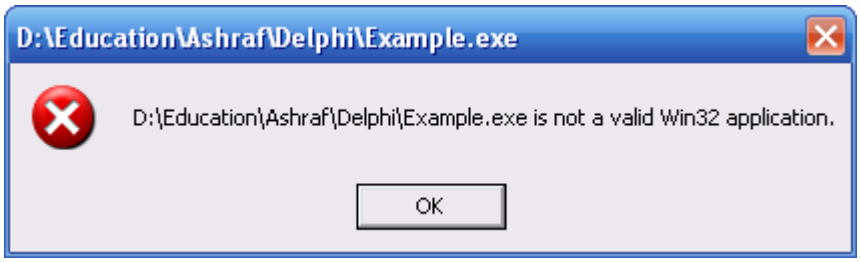



إذا قمت بإختيار الإختيارات المشار إليها في الصورة السابقة سوف نجعل هذا القسم قابل إلي executable , Writeable فمثلاً إذا كنا نريد إضافة 200h بايت إلي هذا

200h بايت إلي الـRawSize والـVirtualSize الخاص بهذا القسم وإذا لاحظت فسوف تري أن قيمة الـRawSize والـVirtualSize متساويين كما في الشكل التالي :




فـإذا أضـفنا 200h بايـت فـسوف يـصبح النـاتج = 00003A00h+200h **00003C00** وسوف نقوم بتعديل قيمة الدالتين إلي 00003C00h وإذا قمت بتشغيل البرنامج فسوف تري هذه الرسالة :




وهذه الرسالة تدل علي أن هناك أشياء أخري لابد من تغيير قيمتها فمثلاً حجم الملف لابد من تغييرة وإذا إستعرضنا خصائص الملف في برنامج LordPE سوف نري هذا الشكل :

**إلي هنا إنتهي هذا الدرس وسوف ترون الشكل في الدرس القادم إن شاء الله**