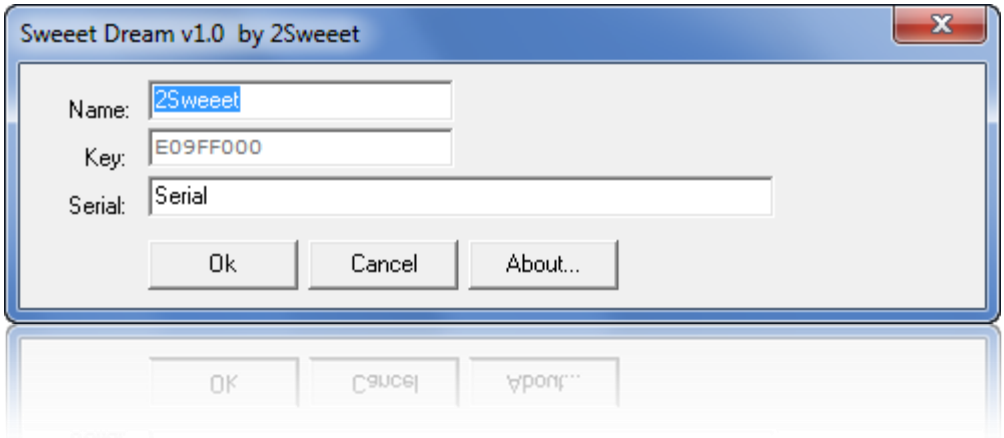
KeyGen para el CrackMe Sweeet Dream 1.0 de 2Sweeet

Desempacando tres capas



By deurus
21/09/2014

ÍNDICE

Equipo utilizado:

S.O: Windows 7 x32 /Windows 7 x64

Depurador: Ollydbg 2 (32bits) con plugins

Analizador: PEiD 0.95

1. Herramientas necesarias

- PEiD o similar.
- OllyDbg 2 con el plugin OllyDumpEX.
- ImportREC
- LordPE (opcional)

2. Introducción

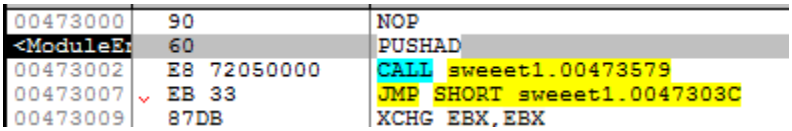
Hoy tenemos aquí un Crackme del año 2000 empacado y con un algoritmo aunque no muy complicado largo de tracear. Está empacado varias veces, algo poco habitual pero recordemos que es un Crackme antiguo. Tras el empacado se encuentra Delphi.

3. OllyDbg

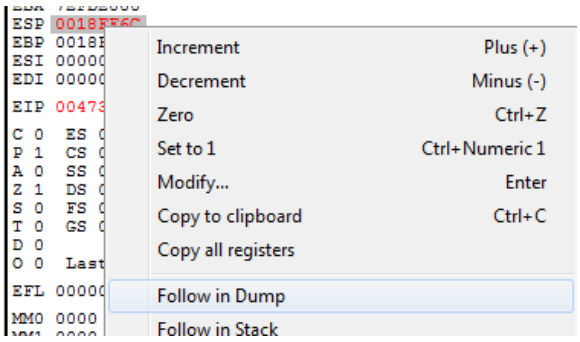
VideoTutorial del desempacado disponible

Si lo pasamos por **PEiD** nos dice que **Aspack 2.1**, **Exeinfo** no está muy seguro y **RDG packer detector** en el escaneo avanzado nos encuentra **Aspack**, **UPX** y **PE-Pack**.

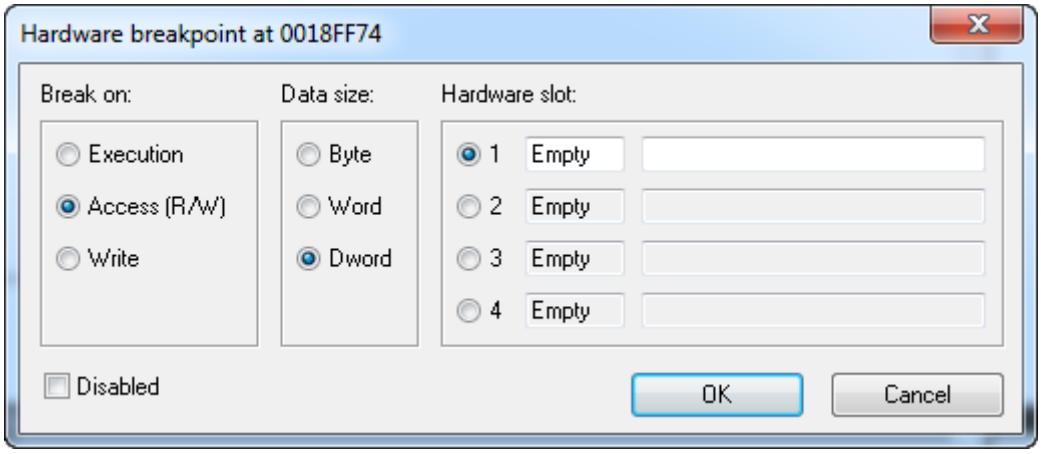
En principio nos enfrentamos a Aspack 2.1, abrimos el crackme con OllyDbg y vemos el típico PUSHAD.

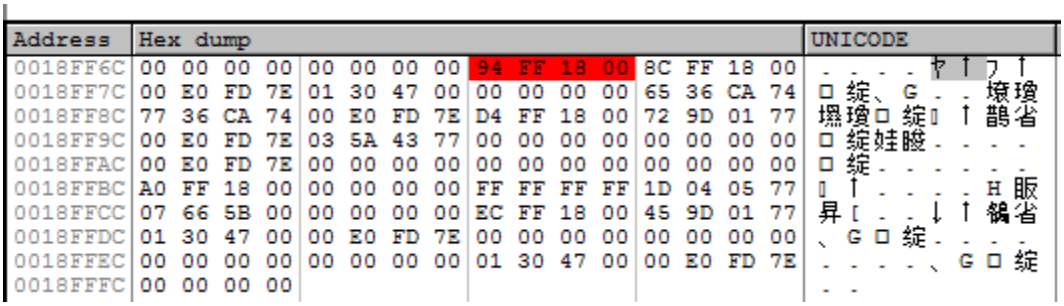


Pulsamos F8 (Step Over) y a continuación click derecho sobre el registro ESP y Follow in DUMP.

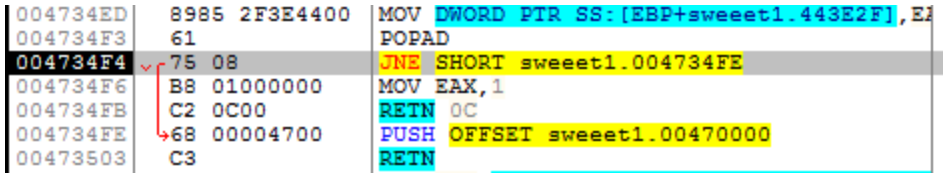


Seleccionamos los **primeros cuatro bytes** útiles del dump y les ponemos un **Breakpoint** de **Hardware**, **Access** y **Dword**.

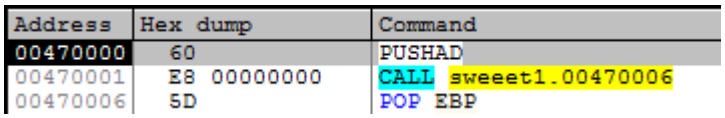




Pulsamos **F9** y nos para aquí:



Ya tenemos a Aspack contra las cuerdas, pulsamos **F8** hasta después del RETN para llegar al **OEP** (Original Entry Point).



Pero en el supuesto **OEP** vemos otro **PUSHAD** por lo que esto no ha terminado. Investigando un poco más vemos que la segunda capa se corresponde con **PE-PACK 1.0**. La estrategia a seguir es la misma, como ya tenemos el breakpoint puesto pulsamos **F9** y nos para aquí:

```
0047026B   894424 1C      MOV DWORD PTR SS:[ESP+1C],EAX
0047026F   61             POPAD
00470270   FFE0           JMP EAX
00470272   8D85 CE050000  LEA EAX,[EBP+5CE]
00470278   50             PUSH EAX
```

Pulsamos **F8** y nos llega a otro **PUSHAD**. Esta vez es UPX.

```
Address   Hex dump     Command
0046E820    60         PUSHAD
0046E821    BE 00404400 MOV ESI,sweeet1.00444000
0046E826    8DBE 00D0FBFF LEA EDI,[ESI+FFFBD000]
0046E82C    C787 D0940500 MOV DWORD PTR DS:[EDI+594D0],880E65E0
0046E836    57         PUSH EDI
```

Pulsamos de nuevo **F9** y paramos aquí:

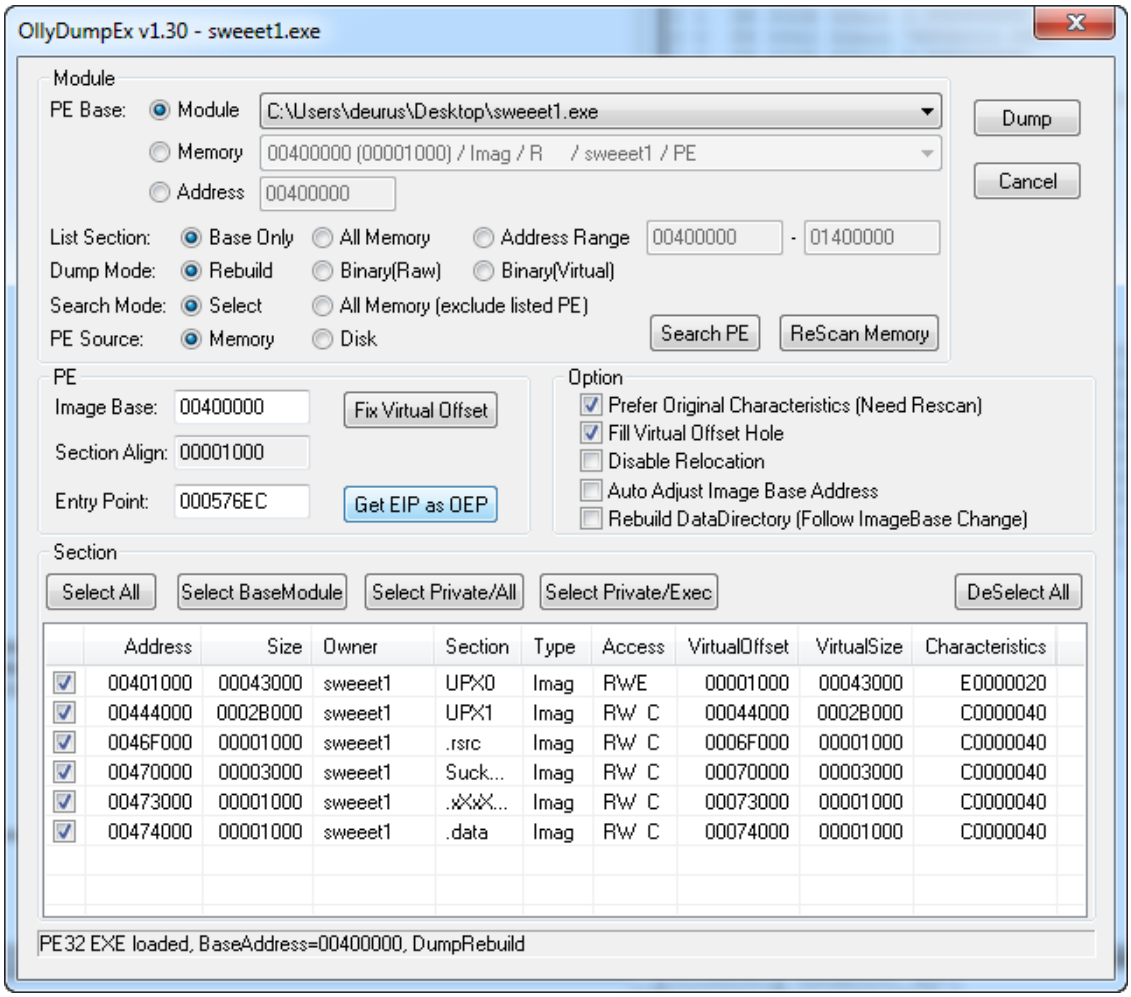
```
0046E970   FF96 4CEE0600  CALL DWORD PTR DS:[ESI+6EE4C]
0046E976   61             POPAD
0046E977   E9 708DFEFF    JMP sweeet1.004576EC
0046E97C   94             XCHG EAX,ESP
0046E97D   E9 4600A4E9    JMP E9EAE9C8
0046E982   46             INC ESI
```

Pulsamos **F8** y esta vez si llegamos al **OEP** (4576EC).

```
Address   Hex dump   Command
004576EC    55       PUSH EBP
004576ED    8BEC     MOV EBP,ESP
004576EF    83C4 F4  ADD ESP,-0C
004576F2    53       PUSH EBX
```

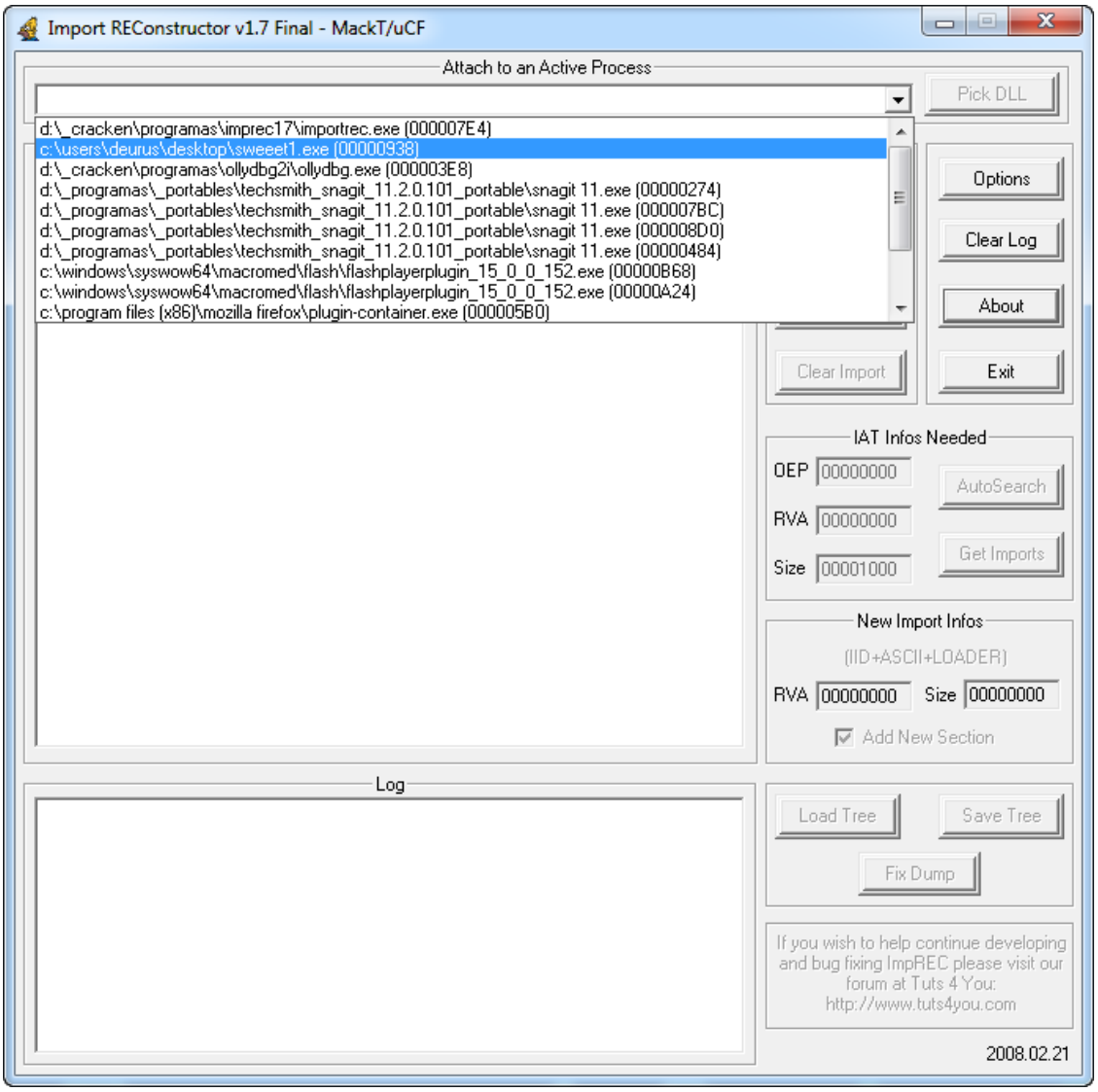
4. OllyDumpEX

A continuación vamos a **dumpear el archivo en memoria**. Vamos a **plugins > OllyDumpEX**, pulsamos sobre "**Get EIP as OEP**" y finalmente sobre "**Dump**".

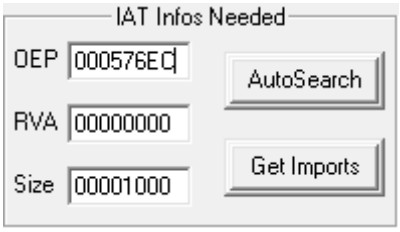


5. ImportREC

Minimizamos Olly (no cerrar), abrimos el programa **ImportREC** y seleccionamos el ejecutable “Sweeet1.exe”.



Pegamos el **OEP** original (576EC), le damos a **AutoSearch** y a continuación a **Get Imports**.

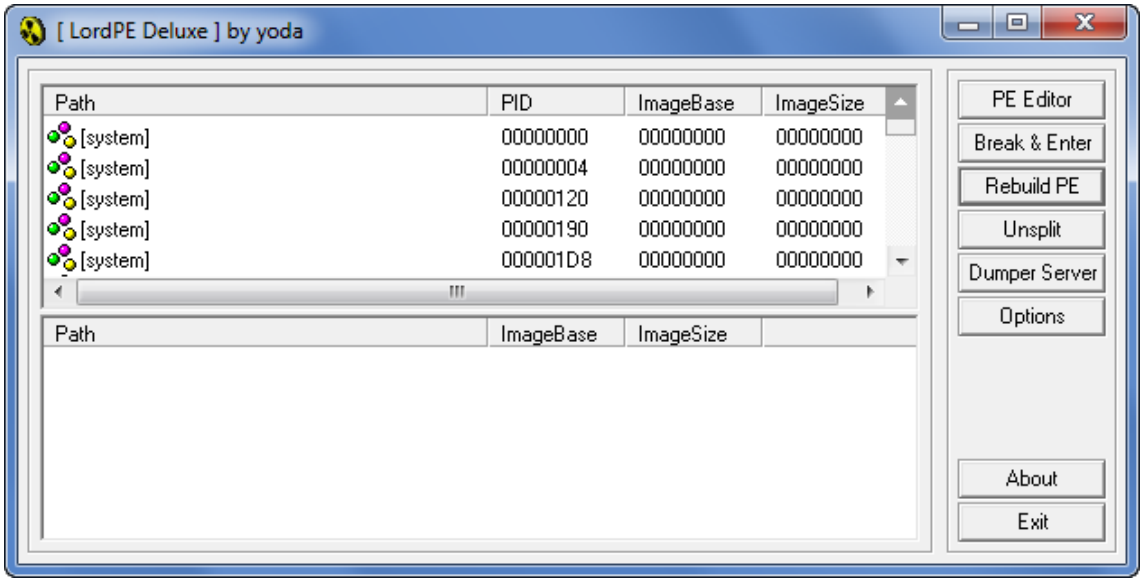


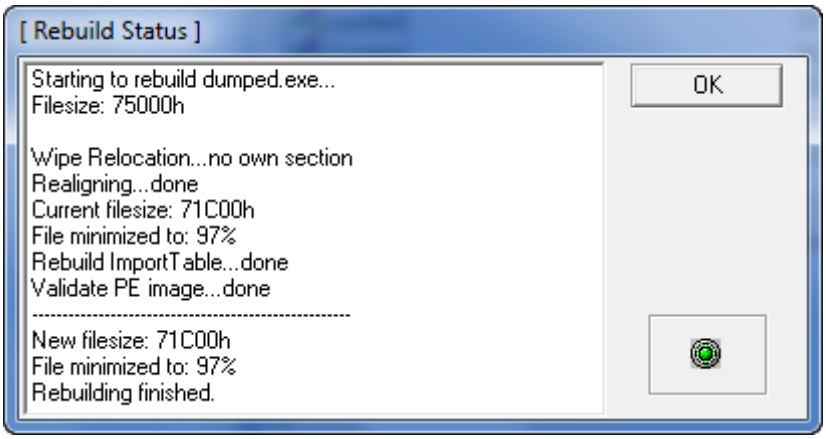
Finalmente pulsamos **Fix Dump** y elegimos el ejecutable dumpeado anteriormente. Esto nos genera un ejecutable dumpeado que es el ejecutable válido.

Ahora **PEiD** nos dice que estamos tratando con un crackme hecho en **Delphi**.

6. LordPE

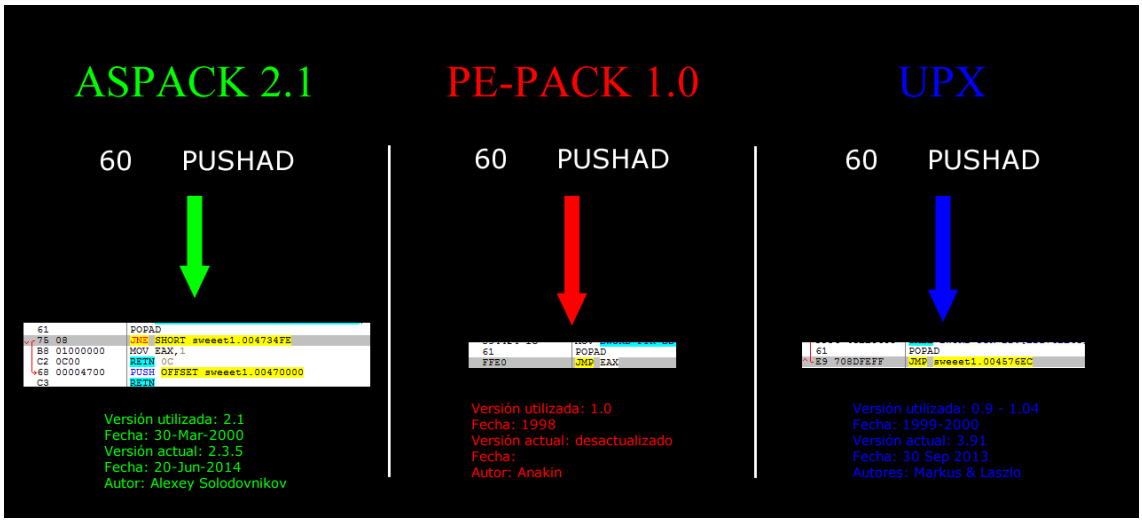
Abrimos **LordPE** y pulsamos sobre **Rebuid PE**, elegimos el ejecutable dumpeado y le dejamos trabajar.





7. Resumen del desempacado

Como he dicho al inicio, encontrarse un ejecutable comprimido varias veces es poco habitual por no decir que muy raro, pero tampoco imposible. Os dejo una imagen con información de los tres empacadores utilizados a modo de resumen.



8. El algoritmo

Cuando abrimos el Crackme nos fijamos en que genera una **key**. Esta key se genera en función del disco duro desde el que se ejecuta.

Como la secuencia de generación del serial válido es larga os pongo lo más importante muy resumido y con ejemplos como siempre.

El serial es del siguiente tipo:

Serial = 1ªParte-2ªParte-3ªParte

Serial = 0000XXXXX-SerialCalculado-xxxx000Z8

Como el seguimiento del serial es un poco extenso, os invito a verlo con más detalle en deurus.info. Os pongo un resumen.

```
Comprobación del tamaño del nombre
-----------------------------------
........
00456EAA    E8 01CCFAFF     CALL sweeet1_Fix_dump_rebuilded.00403AB0
00456EAF    83F8 04         CMP EAX,4 ------------------------------------------------; Nombre >=4
00456EB2    7D 13           JGE SHORT sweeet1_Fix_dump_rebuilded.00456EC7
00456EB4    A1 08954500     MOV EAX,DWORD PTR DS:[sweeet1_Fix_dump_rebuilded.459508]
00456EB9    8B00            MOV EAX,DWORD PTR DS:[EAX]
00456EBB    E8 0869FEFF     CALL sweeet1_Fix_dump_rebuilded.0043D7C8
00456EC0    BB 01000000     MOV EBX,1
00456EC5    EB 15           JMP SHORT sweeet1_Fix_dump_rebuilded.00456EDC
00456EC7    83FB 25         CMP EBX,25
00456ECA    7D 0E           JGE SHORT sweeet1_Fix_dump_rebuilded.00456EDA
00456ECC    83C3 32         ADD EBX,32
00456ECF    83C3 1E         ADD EBX,1E
00456ED2    83EB 4F         SUB EBX,4F
00456ED5    83FB 25         CMP EBX,25 ------------------------------------------------; Nombre <=25
00456ED8  ^ 7C F2           JL SHORT sweeet1_Fix_dump_rebuilded.00456ECC
00456EDA    33DB            XOR EBX,EBX
00456EDC    33C0            XOR EAX,EAX
........
```

1°Bucle - Nuestro nombre (A)

```
.........
00456F55    BE 1B000000     MOV ESI,1B ------------------------------; ESI = 1B
00456F5A    EB 21           JMP SHORT sweeet1_dump_.00456F7D
00456F5C    8D55 D4         LEA EDX,[EBP-2C]
00456F5F    A1 34A84500     MOV EAX,DWORD PTR DS:[sweeet1_dump_.45A8
00456F64    8B80 C4020000   MOV EAX,DWORD PTR DS:[EAX+2C4]
00456F6A    E8 B5DAFCFF     CALL sweeet1_dump_.00424A24
00456F6F    8B45 D4         MOV EAX,DWORD PTR SS:[EBP-2C]
00456F72    0FB64418 FF     MOVZX EAX,BYTE PTR DS:[EBX+EAX-1]---------; Coje digito
00456F77    03F0            ADD ESI,EAX -----------------------------; digito + ESI
00456F79    43              INC EBX
00456F7A    0FAFF3          IMUL ESI,EBX  ---------------------------; multiplica por i (bucle)
00456F7D    8D55 D4         LEA EDX,[EBP-2C]
.........
```

2°Bucle - La key (B)

```
.........
00456F9C        |.  BF 1A000000          MOV EDI,1A -------------------------;EDI = 1A
00456FA1        |.  BB 01000000          MOV EBX,1
00456FA6        |.  EB 1E                JMP SHORT sweeet1_.00456FC6
00456FA8        |>  8D55 D4              /LEA EDX,[LOCAL.11]
00456FAB        |.  A1 34A84500          |MOV EAX,DWORD PTR DS:[45A834]
00456FB0        |.  8B80 D0020000        |MOV EAX,DWORD PTR DS:[EAX+2D0]
00456FB6        |.  E8 69DAFCFF          |CALL sweeet1_.00424A24
00456FBB        |.  8B45 D4              |MOV EAX,[LOCAL.11]
00456FBE        |.  0FB64418 FF          |MOVZX EAX,BYTE PTR DS:[EAX+EBX-1]--;Coje dígito
00456FC3        |.  03F8                 |ADD EDI,EAX
00456FC5        |.  43                   |INC EBX
00456FC6        |>  8D55 D4               LEA EDX,[LOCAL.11]
00456FC9        |.  A1 34A84500          |MOV EAX,DWORD PTR DS:[45A834]
00456FCE        |.  8B80 D0020000        |MOV EAX,DWORD PTR DS:[EAX+2D0]
00456FD4        |.  E8 4BDAFCFF          |CALL sweeet1_.00424A24
00456FD9        |.  8B45 D4              |MOV EAX,[LOCAL.11]
00456FDC        |.  E8 CFCAFAFF          |CALL sweeet1_.00403AB0
00456FE1        |.  3BD8                 |CMP EBX,EAX
00456FE3        |.^ 7C C3                \JL SHORT sweeet1_.00456FA8
.........
```

Generación del serial central

```
.........
00456FE5        |.  B9 01000000          MOV ECX,1
00456FEA        |.  BB 01000000          MOV EBX,1
00456FEF        |.  8BC7                 MOV EAX,EDI
00456FF1        |.  F7EE                 IMUL ESI ----------; C = A * B
00456FF3        |.  99                   CDQ
.........
00456FFD        |.  2345 E8              AND EAX,[LOCAL.6]--; D = A and C
00457000        |.  2355 EC              AND EDX,[LOCAL.5]
00457003        |.  8945 E8              MOV [LOCAL.6],EAX
00457006        |.  8955 EC              MOV [LOCAL.5],EDX
.........
00457032        |.  8BC7                 MOV EAX,EDI
00457034        |.  99                   CDQ
00457035        |.  0345 E8              ADD EAX,[LOCAL.6]--; E = D + B
00457038        |.  1355 EC              ADC EDX,[LOCAL.5]
0045703B        |.  8945 E0              MOV [LOCAL.8],EAX
0045703E        |.  8955 E4              MOV [LOCAL.7],EDX
.........
```

```
00405732          8B4424 10             MOV EAX,DWORD PTR SS:[ESP+10]
00405736          F72424                MUL DWORD PTR SS:[ESP]
00405739          8BC8                  MOV ECX,EAX
0040573B          8B4424 04             MOV EAX,DWORD PTR SS:[ESP+4]
0040573F          F76424 0C             MUL DWORD PTR SS:[ESP+C]------; F = B * D
00405743          03C8                  ADD ECX,EAX
00405745          8B0424                MOV EAX,DWORD PTR SS:[ESP]
00405748          F76424 0C             MUL DWORD PTR SS:[ESP+C]------; G = A * F
........
0045705E        |.  0B0424              OR EAX,DWORD PTR SS:[ESP]-----; Serial central = G or A
........
00457077        |.  E8 FC07FBFF         CALL sweeet1_.00407878
0045707C        |.  8B45 F8             MOV EAX,[LOCAL.2]-------------; EAX = Serial central
........
004570D1        |.  E8 A207FBFF         CALL sweeet1_.00407878
004570D6        |.  8B45 D0             MOV EAX,[LOCAL.12]
004570D9        |.  E8 D2C9FAFF         CALL sweeet1_.00403AB0--------; Obtiene longitud del serial central en hexa
004570DE        |.  8BD8                MOV EBX,EAX
........
004570D1        |.  E8 A207FBFF         CALL sweeet1_.00407878--------;*Nota
```

*Nota:

A partir de aquí genera la primera y tercera parte del serial de la siguiente manera:

Serial = 1ªParte-2ªParte-3ªParte

Serial = 0000XXXXX-SerialCalculado-xxxx000Z8

1ºParte = 3ºdigSerial + 1ºdigSerial + 2ºdigSerial + 3ºdigSerial + 4ºdigNombreMayu + 2ºdigNombreMayu + 5ºdigNombreMayu + 1ºdigNombreMayu + 3ºdigNombreMayu

3ºParte = 3ºdigNombreMin + 1ºdigNombreMin + 4ºdigNombreMin + 2ºdigNombreMin + Tamaño Serial_2ªParte en Hexadecimal y de tres dígitos + Z8

2ºParte:

- Nombre: deurus
- Key: C0C0A000

1) A = 23A2A (Sum Nombre)

2) B = 1A1 (Sum Key)

3) C = B * A = 3A0BE6A

4) D = A and C = 3A2A

5) E = D + B = 3BCB

6) F = B * D = 5EBE6A

7) G = A * F = D303834164

8) Serial = G or A (Serial = D303834164 or 23A2A = D303837B6E (906297047918))

Finalmente el serial correcto quedaría: **6906REUDU-906297047918-udre00CZ8**

A tener en cuenta:

- 1ªParte del serial siempre mayúsculas.
- 2ªParte siempre numérico. Usa el registro de 64 bits (Qword) con signo.**Nota
- 3ªParte siempre minúsculas.

**Nota:

```
Nombre: deurus.info
Key:    E09FF000
Serial: 9169REUDU-16918236-udre008Z8

Fíjate que: -16918236 = FFFFFFFFFEFDD924

Nombre: deurus
Key:    C0C0A000
Serial: 6906REUDU-906297047918-udre00CZ8

906297047918 = 000000D303837B6E
```

9. Crackeando Crackmes by deurus

- https://mega.co.nz/#F!88BRwYoT!O0TzTSZYCdczKLOrfrOyGw
- Lolabits.es/blogcracking (Clave: **blogcrackhack**)
- Entrada en el Blog para el keygen.
- VideoTutorial del desempacado.