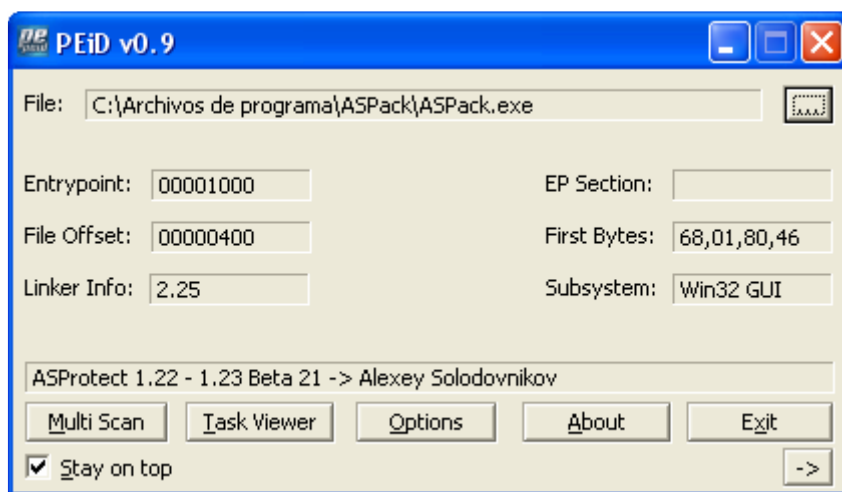


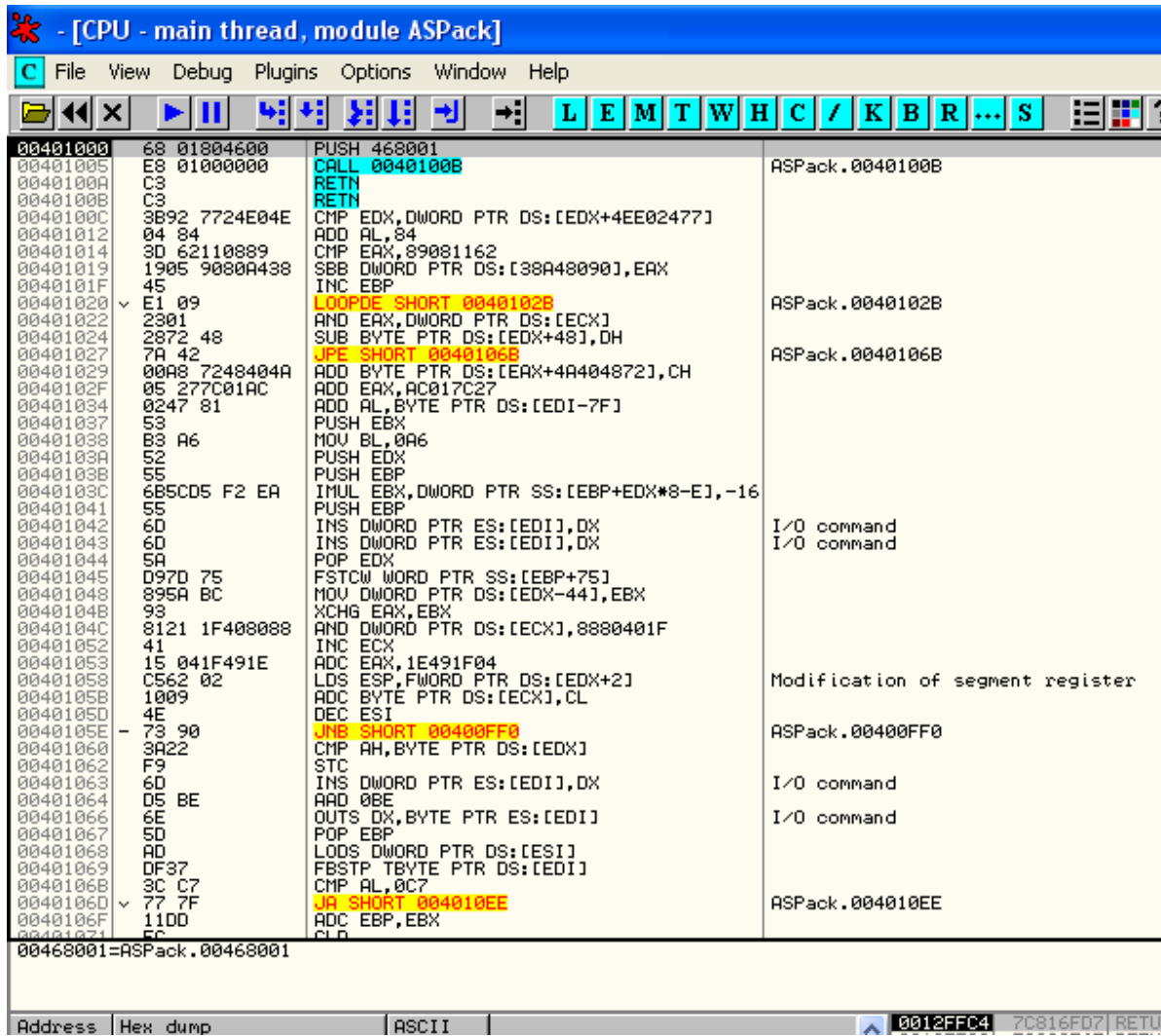


➤ Programa	ASPACK
➤ Versión	2.12
➤ Herramientas	OLLYDBG 1.10, IMPORT REC, PEID
➤ Compilador	Borland Delphi Heuristic Mode
➤ Objetivos	DESEMPACAR EL EJECUTABLE
➤ Cracker	AGUML

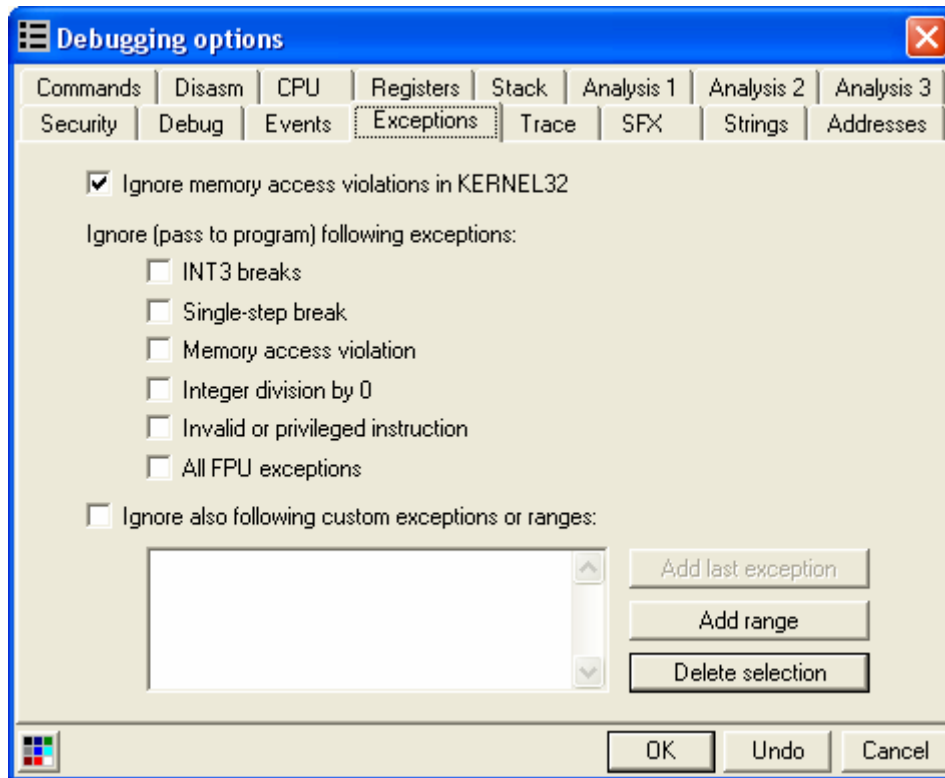
Antes de nada conozcamos a nuestro enemigo. Abrimos PEID y miramos a ver con que esta empackado y:



Ufff, con lo poco que me gusta ese packer, jejeje.
 Bueno pues carguémoslo en Olly a ver que pasa:



Ahora usaremos el método que mas he visto y que a mi me ha dado resultado para llegar al OEP. Simplemente nos vamos a Options->Debugging Options o si lo prefieren Alt + O y dan en la pestaña Exceptions y dejan marcada solo la primera opción que hay:



Le dan a OK y ahora simplemente le dan a F9 y empiezan a pasar las excepciones con Shift + F7 y F9 y teniendo mucho cuidado al contar las excepciones hasta que el programa arranque (en mi caso fueron 26 excepciones). Ahora reiniciamos Olly con Ctrl. + F2 y hacemos lo mismo pero esta vez nos pararemos en la ultima excepción y le damos a Shift + F7 solo para pasarla.

Ahora pulsamos Alt + M para sacar la ventana Memory Map y marcamos la sección .code del ejecutable y le ponemos un BP Memory on Access haciendo clic derecho sobre ella como se muestra en la imagen:

Memory map

Address	Size	Owner	Section	Contains	Type	Access	Initial	Mapped as
00010000	00001000				Priv	RW	RW	
00020000	00001000				Priv	RW	RW	
0012A000	00001000			stack of ma	Priv	RW	Gua: RW	
0012B000	00005000				Priv	RW	Gua: RW	
00130000	00001000				Priv	RWE	RWE	
00140000	00003000				Map	R	R	
00150000	0000F000				Priv	RW	RW	
00250000	00006000				Priv	RW	RW	
00260000	00004000				Map	RW	RW	
00270000	00016000				Map	R	R	
00290000	0003D000				Map	R	R	\Device\HarddiskVolume1\WINDC
002D0000	00041000				Map	R	R	\Device\HarddiskVolume1\WINDC
00320000	00006000				Map	R	R	\Device\HarddiskVolume1\WINDC
00330000	00041000				Map	R	R	
00380000	00001000				Priv	RWE	RWE	
00390000	00001000				Priv	RWE	RWE	
003A0000	00001000				Priv	RWE	RWE	
003B0000	00001000				Priv	RW	RW	
003C0000	00001000				Priv	RW	RW	
003D0000	00004000				Priv	RW	RW	
003E0000	00003000				Map	R	R	\Device\HarddiskVolume1\WINDC
003F0000	00002000				Map	R	R	
00400000	00001000	ASPack		PE header	Imag	R	RWE	
00401000	00042000	ASPack		code	I			
00443000	00002000	ASPack		data	I			
00445000	00001000	ASPack			I			
00446000	00002000	ASPack			I			
00448000	00002000	ASPack			I			
0044A000	00001000	ASPack			I			
0044B000	00001000	ASPack			I			
0044C000	00004000	ASPack			I			
00450000	00018000	ASPack	.rsrc	resources	I			
00468000	0000F000	ASPack	.ass	imports, rel	I			
00477000	00001000	ASPack	.adata		I			
00480000	0000B000				M			
00540000	00002000				M			
8B0424				MOV EAX, DWORD PTR SS:[ESP]				
8BE5				MOV ESP, EBP				
5D				POP EBP				
C3				RETN				
90				NOP				
8DA424	00000000			LEA ESP, DWORD PTR SS:[ESP]				
8D49	00			LEA ECX, DWORD PTR DS:[ECX]				
90				NOP				
90				NOP				
90				NOP				
90				NOP				

Y ahora demos a F9 y...

```

[CPU - main thread, module ASPack]
File View Debug Plugins Options Window Help
L E M T W H C / K B R ... S
0044289C 55 PUSH EBP
0044289D 8BEC MOV EBP,ESP
0044289F 83C4 F4 ADD ESP,-0C
004428A2 E8 400BF0FF CALL 0040033F4
004428A7 E8 0C21F0FF CALL 004049B8
004428AC E8 6354F0FF CALL 00407D14
004428B1 E8 06C6F0FF CALL 0040EF8C
004428B6 E8 BDC7F0FF CALL 0040F078
004428BB E8 08E7F0FF CALL 00410FC8
004428C0 E8 974DF0FF CALL 0041765C
004428C5 E8 B61AF0FF CALL 00424380
004428CA E8 F986F0FF CALL 0042AFC8
004428CF E8 E899F0FF CALL 0042C2BC
004428D4 E8 639AF0FF CALL 0042C33C
004428D9 E8 CE95F0FF CALL 0042CEAC
004428DE E8 95ABF0FF CALL 0042D478
004428E3 E8 308AF0FF CALL 0042E318
004428E8 E8 6F8AF0FF CALL 0042E35C
004428ED E8 FABAF0FF CALL 0042E3EC
004428F2 E8 69E1F0FF CALL 00430A60
004428F7 E8 A4AFF0FF CALL 004308A0
004428FC A1 30564400 MOV EAX,DWORD PTR DS:[445630]
00442901 E8 D210F0FF CALL 004239D8
00442906 BA 38294400 MOV EDX,442938
0044290B A1 30564400 MOV EAX,DWORD PTR DS:[445630]
00442910 E8 DF0DF0FF CALL 004236F4
00442915 FF15 0C494400 CALL DWORD PTR DS:[44490C]
0044291B A1 30564400 MOV EAX,DWORD PTR DS:[445630]
00442920 E8 5311F0FF CALL 00423A78
00442925 E8 D21BF0FF CALL 004044FC
0044292A 8BE5 MOV ESP,EBP
0044292C 5D POP EBP
0044292D C3 RETN
0044292E 0000 ADD BYTE PTR DS:[EAX],AL
00442930 FFFF
00442932 FFFF
00442934 06 PUSH ES
00442935 0000 ADD BYTE PTR DS:[EAX],AL
00442937 0041 53 ADD BYTE PTR DS:[ECX+53],AL
0044293A 50 PUSH EAX
0044293B 61 POPAD
0044293C 636B 00 ARPL WORD PTR DS:[EBX],BP
0044293F 0000 ADD BYTE PTR DS:[EAX],AL
00442941 0000 ADD BYTE PTR DS:[EAX],AL
00442943 0000 ADD BYTE PTR DS:[EAX],AL
00442945 0000 ADD BYTE PTR DS:[EAX],AL
00442947 0000 ADD BYTE PTR DS:[EAX],AL
EBP=0012FFF0
Address Hex dump ASCII 0012FFC4 7C816FD7 RETURN to kernel32.7
    
```

Ya estamos en el OEP y tiene buena pinta porque parece ser que no hay Stolen Bytes así que sigamos.

Para encontrar la IAT simplemente ponte encima de la primera CALL y das Intro a ver a donde nos lleva:

Address	Hex dump	Instruction	Module
004033F4	E8 B3FFFFFF	CALL 004033AC	ASPack.004033AC
004033F9	6A 00	PUSH 0	
004033FB	E8 D0DEFFFF	CALL 004012D0	ASPack.004012D0
00403400	8905 14504400	MOV DWORD PTR DS:[445014],EAX	
00403406	E8 ADDEFFFF	CALL 004012B8	ASPack.004012B8
0040340B	8905 1C504400	MOV DWORD PTR DS:[44501C],EAX	
00403411	C705 18504400	MOV DWORD PTR DS:[445018],0A	
00403418	B8 4C314000	MOV EAX,40314C	
00403420	C3	RETN	
00403421	8D40 00	LEA EAX,DWORD PTR DS:[EAX]	
00403424	53	PUSH EBX	
00403425	833D 04534400	CMP DWORD PTR DS:[4453D4],0	
0040342C	7D 0A	JGE SHORT 00403438	ASPack.00403438
0040342E	B8 E2000000	MOV EAX,0E2	
00403433	E8 D9100000	CALL 00404511	ASPack.00404511
00403438	68 08000000	PUSH 8	
0040343D	6A 40	PUSH 40	
0040343F	E8 DCDEFFFF	CALL 00401320	ASPack.00401320
00403444	8BD8	MOV EBX,EAX	
00403446	85DB	TEST EBX,EBX	
00403448	75 0C	JNZ SHORT 00403456	ASPack.00403456
0040344A	B8 E2000000	MOV EAX,0E2	
0040344F	E8 BD100000	CALL 00404511	ASPack.00404511
00403454	EB 0C	JMP SHORT 00403462	ASPack.00403462
00403456	53	PUSH EBX	
00403457	A1 D4534400	MOV EAX,DWORD PTR DS:[4453D4]	
0040345C	50	PUSH EAX	
0040345D	E8 86DEFFFF	CALL 004012E8	ASPack.004012E8
00403462	891D 90544400	MOV DWORD PTR DS:[445490],EBX	
00403468	5B	POP EBX	
00403469	C3	RETN	
0040346A	8BC0	MOV EAX,EAX	
0040346C	8A0D 34504400	MOV CL,BYTE PTR DS:[445034]	
00403472	8B05 D4534400	MOV EAX,DWORD PTR DS:[4453D4]	
00403478	84C9	TEST CL,CL	
0040347A	75 28	JNZ SHORT 004034A4	ASPack.004034A4
0040347C	64:8B15 2C000000	MOV EDX,DWORD PTR FS:[2C]	
00403483	8B0482	MOV EAX,DWORD PTR DS:[EDX+EAX*4]	
00403486	C3	RETN	
00403487	E8 98FFFFFF	CALL 00403424	ASPack.00403424
0040348C	8B05 D4534400	MOV EAX,DWORD PTR DS:[4453D4]	
00403492	50	PUSH EAX	
00403493	E8 48DEFFFF	CALL 004012E0	ASPack.004012E0
00403498	85C0	TEST EAX,EAX	
0040349A	74 01	JE SHORT 0040349D	ASPack.0040349D
0040349C	C3	RETN	
0040349D	8B05 90544400	MOV EBX,DWORD PTR DS:[445490]	

No se ve nada interesante pero como siempre se llama a alguna API muy cerca del OEP, pues probare a ver que hay dentro de esas CALLs.

En la primera no hay tampoco nada interesante pero en la segunda salimos aquí:

The screenshot shows a debugger window titled "[CPU - main thread, module ASPack]". The main area displays assembly code with instructions like `JMP DWORD PTR DS:[44615C]` and `oleaut32.SysAllocStringLen`. Several instructions are highlighted in yellow. The right side of the window shows a register list. At the bottom, a dump panel shows memory addresses and their corresponding hex and ASCII values.

Address	Hex dump	ASCII
00443000	02 00 8B C0 00 8D 40 00	0.i+.i0.
00443008	98 3D 40 00 A8 20 40 00	v=0.¿0.

Bueno pues ya dimos con la zona caliente para dar con la IAT. Ahora nos ponemos encima de cualquiera de esos saltos incondicionales que empiezan con FF25 y nos vamos al panel que es la parte de debajo de la ventana Disassembler y hacemos clic derecho sobre el valor que se ve y elegimos Follow Address in Dump:

The screenshot shows the OllyDbg interface with the following assembly code:

```

00401200 - FF25 5C614400 JMP DWORD PTR DS:[44615C]
00401206 - 8BC0 MOV EAX, EAX
00401208 - FF25 58614400 JMP DWORD PTR DS:[446158]
0040120E - 8BC0 MOV EAX, EAX
00401210 - FF25 54614400 JMP DWORD PTR DS:[446154]
00401216 - 8BC0 MOV EAX, EAX
00401218 - FF25 50614400 JMP DWORD PTR DS:[446150]
0040121E - 8BC0 MOV EAX, EAX
00401220 - FF25 4C614400 JMP DWORD PTR DS:[44614C]
00401226 - 8BC0 MOV EAX, EAX
00401228 - FF25 C4614400 JMP DWORD PTR DS:[4461C4]
0040122E - 8BC0 MOV EAX, EAX
00401230 - FF25 C0614400 JMP DWORD PTR DS:[4461C0]
00401236 - 8BC0 MOV EAX, EAX
00401238 - FF25 BC614400 JMP DWORD PTR DS:[4461BC]
0040123E - 8BC0 MOV EAX, EAX
00401240 - FF25 B8614400 JMP DWORD PTR DS:[4461B8]
00401246 - 8BC0 MOV EAX, EAX
00401248 - FF25 B4614400 JMP DWORD PTR DS:[4461B4]
0040124E - 8BC0 MOV EAX, EAX
00401250 - FF25 48614400 JMP DWORD PTR DS:[446148]
00401256 - 8BC0 MOV EAX, EAX
00401258 - FF25 44614400 JMP DWORD PTR DS:[446144]
0040125E - 8BC0 MOV EAX, EAX
00401260 - FF25 40614400 JMP DWORD PTR DS:[446140]
00401266 - 8BC0 MOV EAX, EAX
00401268 - FF25 3C614400 JMP DWORD PTR DS:[44613C]
0040126E - 8BC0 MOV EAX, EAX
00401270 - FF25 38614400 JMP DWORD PTR DS:[446138]
00401276 - 8BC0 MOV EAX, EAX
00401278 - FF25 34614400 JMP DWORD PTR DS:[446134]
0040127E - 8BC0 MOV EAX, EAX
00401280 - FF25 30614400 JMP DWORD PTR DS:[446130]
00401286 - 8BC0 MOV EAX, EAX
00401288 - FF25 2C614400 JMP DWORD PTR DS:[44612C]
0040128E - 8BC0 MOV EAX, EAX
00401290 - 53 PUSH EBX
00401292 - 56 PUSH ESI
00401294 - BE 3C544400 MOV ESI, 44543C
00401296 - 833E 00 CMP DWORD PTR DS:[ESI], 0
00401298 - 75 3A JNZ SHORT 004013A6
0040129A - 68 44060000 PUSH 644
0040129C - 6A 00 PUSH 0
0040129E - E8 A8FFFFFF CALL 00401320
004012A0 - 8BC8 MOV ECX, EAX
004012A2 - 85C9 TEST ECX, ECX
004012A4 - 75 0F JNZ SHORT 00401320
    
```

The context menu is open over the instruction at address 00443000, with the following options:

- Copy pane to clipboard
- Modify data
- Follow address in Dump
- Follow value in Dump
- Appearance

The dump view at the bottom shows a sequence of 00 bytes, indicating a null zone:

```

Address Hex dump
00443000 02 00
00443010 30 22
00443020 52 75
00443030 20 20
00443040 72 6F
00443050 20 20
00443060 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20
00443070 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20
00443080 20 20 20 20 00 0A 0B C0 00 00 00 00 00 88 30 44 00
00443090 00 00 00 00 00 00 00 00 00 00 00 00 00 80 00 00 40
004430A0 00 00 00 C0 00 00 00 00 00 00 00 00 00 01 00 00 00
004430B0 02 00 00 00 03 00 00 00 02 00 00 00 89 FF 00 00
004430C0 03 00 00 00 0A FF 00 00 04 00 00 00 0B FF 00 00
    
```

Y ya estamos en la IAT aunque tiene muy mala pinta y esta redireccionado casi todo hacia el Asprotect pero vamos a repararla. Lo primero es ver donde empieza y donde termina y para ello subamos buscando a ver si vemos una zona de 00000000 o vemos direcciones seguidas que no tengan referencias (para saber si un dword tiene referencias lo marcamos y hacemos Ctrl. + R y si sale la lista vacía es porque no es un dato valido de nuestra IAT y sobra así que si no se ve la zona de 00000000 tendríamos que buscar el inicio de nuestra IAT mirando si los dwords tienen referencias o no. En este caso tenemos suerte y Podemos ver el comienzo de la IAT en 446128 :

Address	Hex dump	ASCII
00446050	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00446060	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00446070	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00446080	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00446090	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
004460A0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
004460B0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
004460C0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
004460D0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
004460E0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
004460F0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00446100	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00446110	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00446120	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00446130	74 44 A3 00 80 44 A3 00 8C 44 A3 00 98 44 A3 00	tDu.CDu.iDu.yDu.
00446140	A4 44 A3 00 B0 44 A3 00 C4 44 A3 00 D8 44 A3 00	kDu.~Du.-Du.iDu.
00446150	E4 44 A3 00 F0 44 A3 00 FC 44 A3 00 B8 C9 9E 00	sDu.-Du.*Du.@F*. [EU.4u.†*×.¶Eu.
00446160	08 45 A3 00 34 03 A3 00 18 CA 9E 00 14 45 A3 00	[EU.4u.†*×.¶Eu.
00446170	28 45 A3 00 34 45 A3 00 48 45 A3 00 54 45 A3 00	[EU.4u.†*×.¶Eu.
00446180	60 45 A3 00 6C 45 A3 00 80 45 A3 00 8C 45 A3 00	[EU.4u.†*×.¶Eu.
00446190	98 45 A3 00 A4 45 A3 00 B0 45 A3 00 BC 45 A3 00	[EU.4u.†*×.¶Eu.
004461A0	C8 45 A3 00 D4 45 A3 00 E0 00 00 00 E0 45 A3 00	[EU.4u.†*×.¶Eu.
004461B0	00 00 00 00 EA 6A 0F 77 1E 4E 0F 77 20 49 0F 770j*waN*w I*w
004461C0	7E 4C 0F 77 A7 48 0F 77 00 00 00 00 E7 EB DA 77	*L*w@K*w.....bUrw
004461D0	49 6F DB 77 83 78 00 77 B5 C1 DC 77 1B 76 00 77	TcMwv en@t wtUkU

Y en 4466A0 podemos ver también claramente el final:

Address	Hex dump	ASCII
004465D0	DC 4E A3 00 E8 4E A3 00 F4 4E A3 00 08 4F A3 00	■Nú.βNú.¶Nú.□Ou.
004465E0	14 4F A3 00 20 4F A3 00 2C 4F A3 00 38 4F A3 00	¶Ou. Ou. .Ou.8Ou.
004465F0	44 4F A3 00 50 4F A3 00 5C 4F A3 00 68 4F A3 00	DOu.POu.\Ou.hOu.
00446600	74 4F A3 00 80 4F A3 00 8C 4F A3 00 98 4F A3 00	tOu.COu.iOu.yOu.
00446610	A4 4F A3 00 B0 4F A3 00 00 00 00 00 E0 0E 72 7E	kOu.~Ou.....0R~
00446620	00 00 00 00 BC 4F A3 00 C8 4F A3 00 D4 4F A3 00°Ou.°Ou.°Ou.
00446630	E0 4F A3 00 EC 4F A3 00 F8 4F A3 00 04 50 A3 00	°Ou.yOu.°Ou.°Pu.
00446640	10 50 A3 00 1C 50 A3 00 28 50 A3 00 34 50 A3 00	¶Pu.LPu.(Pu.4Pu.
00446650	40 50 A3 00 4C 50 A3 00 00 00 00 00 1E 31 36 76	@Pu.LPu.....416v
00446660	00 00 00 00 73 33 50 77 DA F6 4C 77 2C 00 4C 77s3Pwr+Lw.3Lw
00446670	D0 05 4D 77 00 00 00 00 80 48 0F 77 00 00 00 00	s'Mw.....CH*w.....
00446680	8C 82 19 77 4E 57 19 77 A3 30 19 77 4C 4D 19 77	Ie+wnN+tw@+wLM+tw
00446690	09 60 19 77 02 79 19 77 8D 36 19 77 00 00 00 00	J+woy+wi6+w.....
004466A0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
004466B0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
004466C0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
004466D0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
004466E0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
004466F0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00446700	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00446710	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00446720	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00446730	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00446740	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00446750	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

Entonces tenemos que:

El OEP es 44289C

El inicio de la IAT esta en 446128

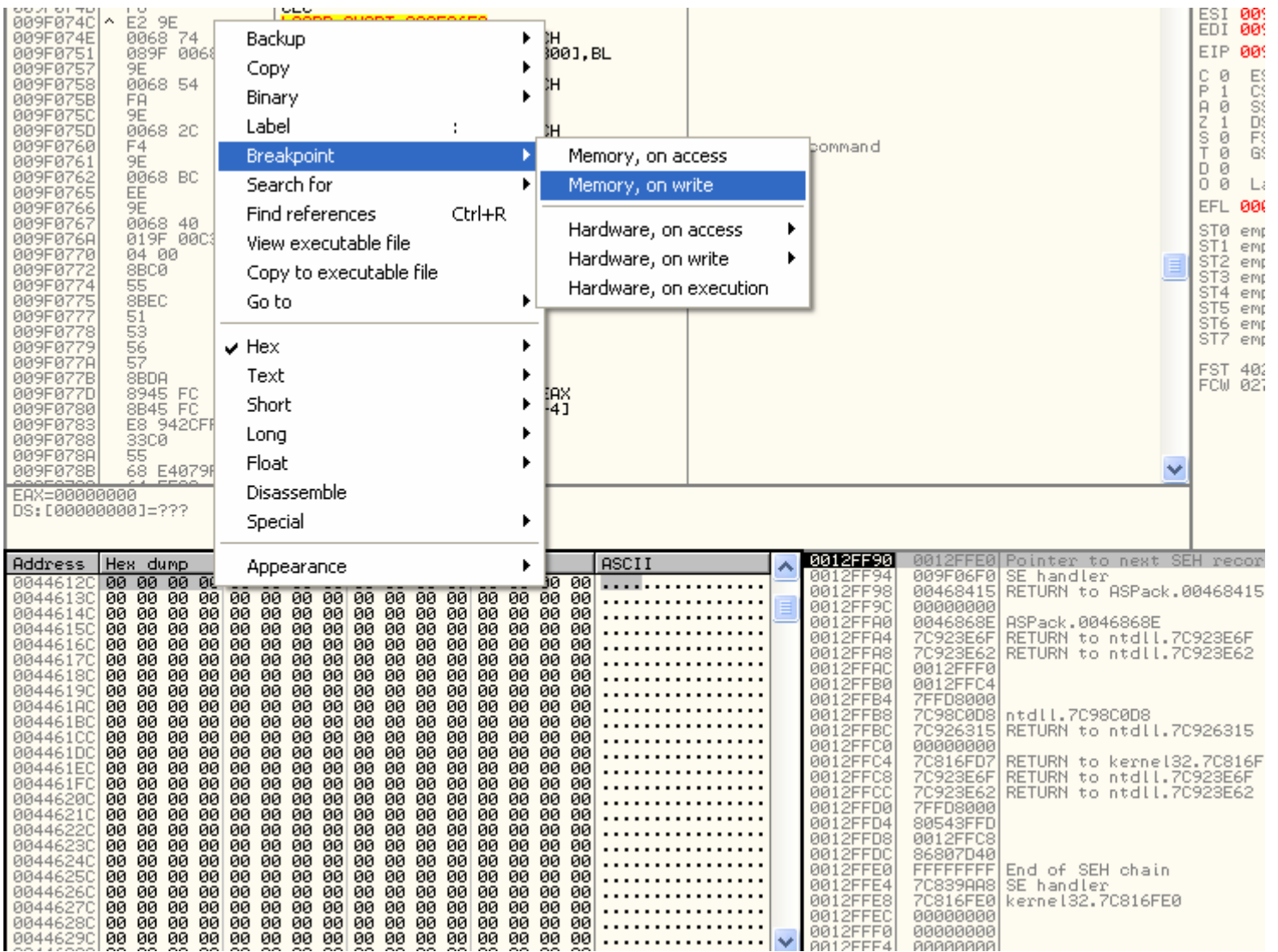
El fin de la IAT esta en 4466A0

Y su tamaño es el resultado de restarle al fin de la IAT el inicio de la IAT así que:

Tamaño de la IAT = 4466A0 - 446128= 578

A LA CAZA DEL CALL SEMI-MAGICO

Ahora reiniciemos y vayamos a 44612C que es el principio de las direcciones que debe escribir en nuestra IAT y pongamos un Memory Breakpoint on Write:

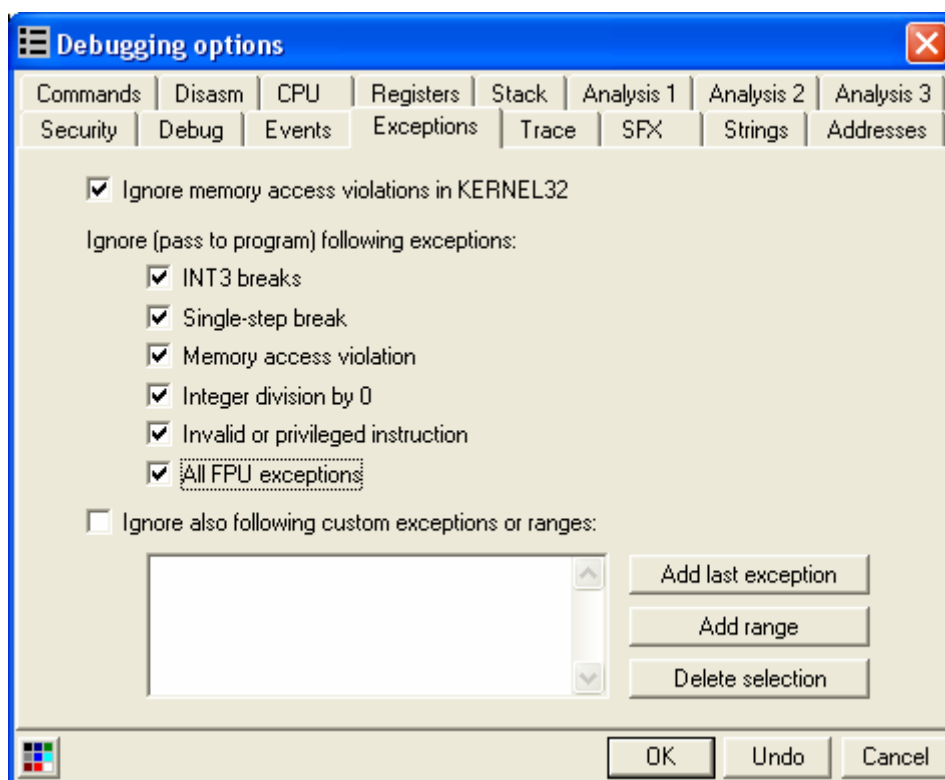


Y le damos a F9 y empezamos a pasar las excepciones otra vez igual que antes pero esta vez hay que tener mucho cuidado porque hay que contar muy bien las excepciones ya que te puedes equivocar y contar la parada en el BPM como si fuera una excepción.

La primera vez que para por nuestro BPM es en :

Esta si es importante ya que esta justo después del salto semi-mágico que nos arreglara casi toda la IAT.

El salto que tendremos que nopear será el que esta encima de donde nos detuvimos así que apuntemos la dirección donde esta ese CALL y volvamos a reiniciar con Ctrl. + F2. Le damos a F9, le damos a Ctrl. + G y metemos la dirección de donde estaba la CALL que en mi caso esta en 009EF9C1 y le damos a OK y nos llevara a donde esta la CALL. Le ponemos un BP y pulsamos Alt + O y marcamos todas opciones de la pestaña Exceptions excepto la última:



Damos a OK y ahora Shift + F7 y F9 y llegamos al BP.

The screenshot displays a debugger interface with three main panels:

- Assembly Window:** Shows assembly instructions with their addresses and hex values. The current instruction is `POPAD` at address `009EFC82`. Other instructions include `PUSH EBX`, `CMP AL, 2`, `JE SHORT 009EFC73`, `MOVZX ECX, BYTE PTR DS:[ESI]`, `INC ECX`, `JMP SHORT 009EFC78`, `MOV ECX, 4`, `ADD ESI, ECX`, `CALL 009EF940`, `POP EBX`, `JMP SHORT 009EFC50`, `POPAD`, `CALL 009EFC02`, `PUSH 9EFC91`, `INC DWORD PTR SS:[ESP]`, `RETN`, `MOV ESP, 0C24448B`, `JMP SHORT 009EFC99`, `XCHG BYTE PTR DS:[EBX+B880], AL`, `ADD BYTE PTR DS:[EDX], AL`, `JMP SHORT 009EFCBC`, `MOV ESP, EBEB17EB`, `ADC AL, 0E8`, `JMP SHORT 009EFCBC`, `CALL EC870B96`, `OR EBX, EAX`, `JMP SHORT 009EFCBC`, `INT 20`, `JMP SHORT 009EFCBC`, `CALL 3286FE98`, `ROL BL, 0EB`, `ADD EAX, EBX`, `XOR EAX, EAX`, `JMP SHORT 009EFC08`, `INT 20`, `PUSH DWORD PTR FS:[EAX]`, `JMP SHORT 009EFC0E`, and `JMP 31BF8636`.
- Registers (FPU) Window:** Shows the state of various registers. `EIP` is `009EFC7F`. `ES` is `0023`, `CS` is `001B`, `SS` is `0023`, `DS` is `0023`, `FS` is `003B`, and `GS` is `0000`. `LastErr` is `ERROR_SUCCESS (00000000)`. `EFL` is `00000216 (NO, NB, NE, A, NS, PE, GE, G)`. `ST0-ST7` are empty. `FST` is `4020` and `FCW` is `027F`.
- Stack Window:** Shows the stack at address `0012FF40` containing `00A32D0C (00A32D0C), ASCII "kernel32.dll"` and `EBX=0012FF80, (ASCII "0aD")`.

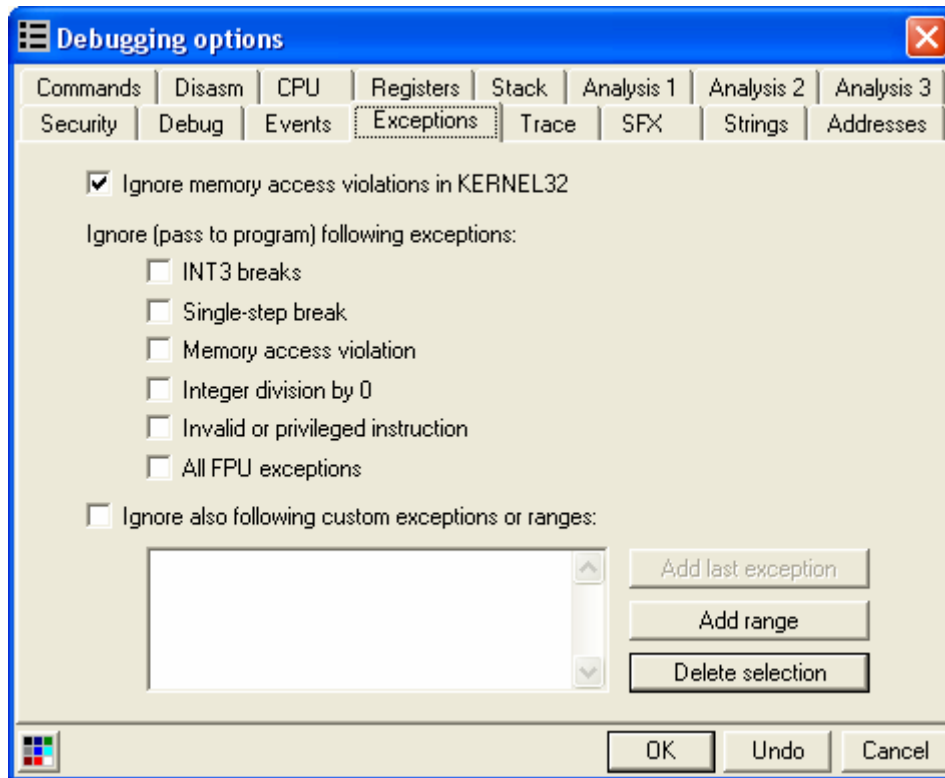
At the bottom, there is a hex dump window showing memory addresses from `0044612C` to `004461FC` with their corresponding hex values and ASCII representations.

Bueno, en la anterior imagen podemos ver el POPAD, pero no solo eso, también podemos ver que nuestro primer valor de la IAT ya si que parece una entrada de la IAT buena. Para que nos la repare toda sin tener que trazar para reparar toda la IAT, pongamos un BP en el POPAD y demos a F9.

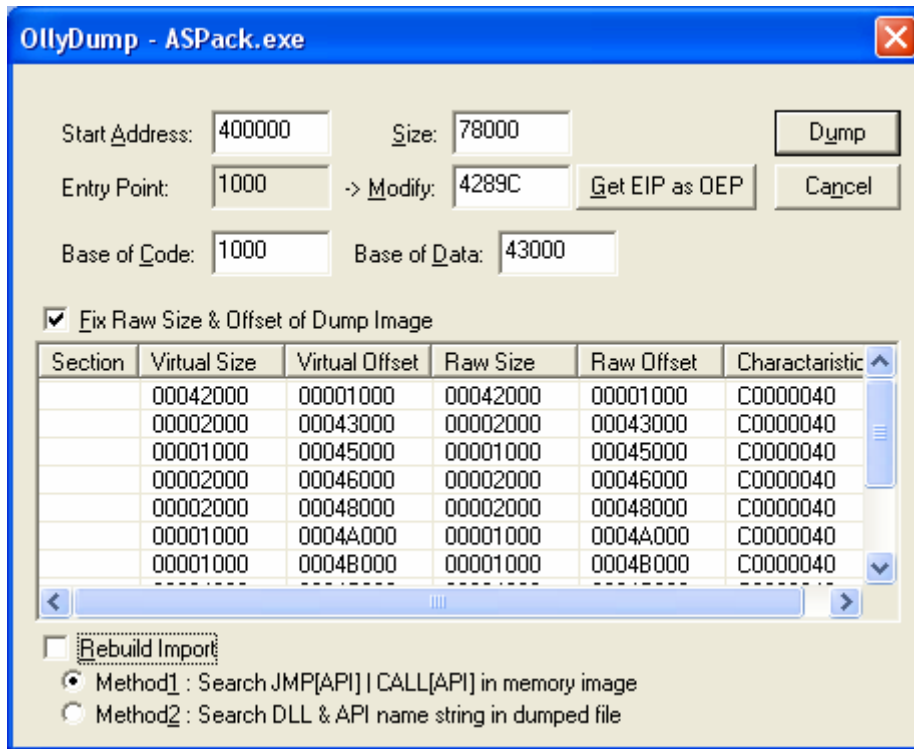
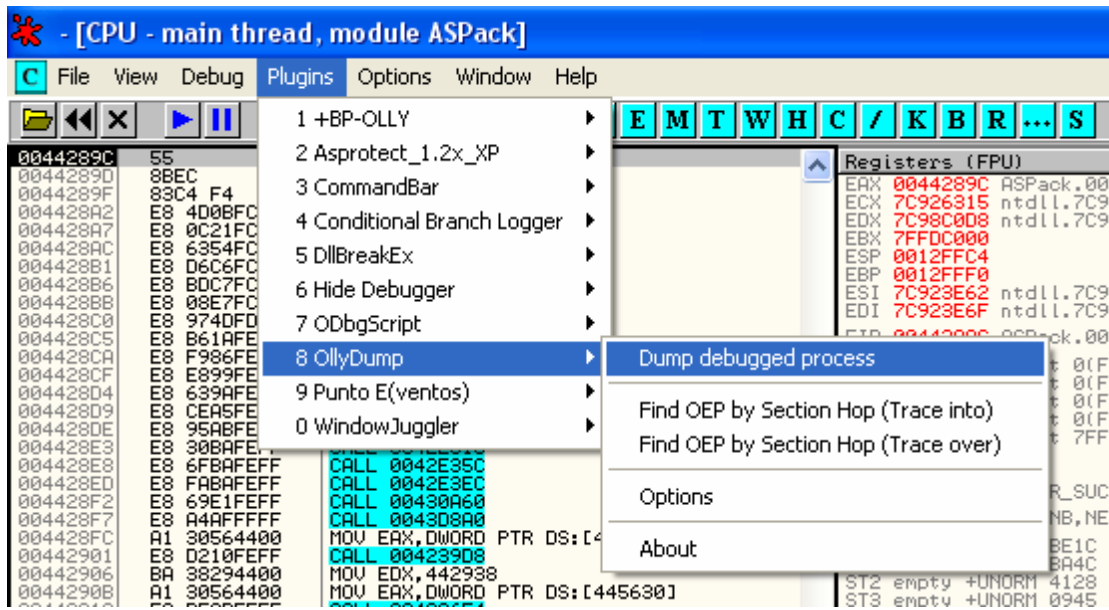
The screenshot shows a debugger interface with three main panes:

- Assembly Pane (Left):** Displays assembly instructions with addresses from 009EF985 to 009EF9E9. Instructions include MOV, INC, XOR, MOV BL, BYTE PTR SS:[EBP-1], MOV ECX, EBX, LEA EAX, DWORD PTR SS:[EBP-101], CALL, PUSH, and NOP. A context menu is overlaid on this pane, listing actions like Backup, Copy, Binary, Undo selection (Alt+BkSp), Assemble (Space), Label (:), Comment (;), Breakpoint, New origin here (Ctrl+Gray *), Go to, Follow in Dump, Search for, Find references to, Analysis, Conditional Branch Logger, Run Script, Dump debugged process, and Appearance.
- Registers (FPU) Pane (Top Right):** Shows the state of CPU registers. EAX is 00000000, ECX is 0012FE0C, EDX is 00446698 (ASPack.00446698), EBX is 00A3429E (ASCII "wininet.dll"), ESP is 0012FF44, EBP is 0012FF88, ESI is 00A34332, EDI is 00446680 (ASPack.00446680), and EIP is 009EFC82. It also shows control flags (C0, P1, A0, Z1, S0, T0, D0, O0) and error codes (LastErr: ERROR_SUCCESS).
- Hex Dump and ASCII Pane (Bottom):** The Hex dump shows memory addresses from 0044612C to 0044629C. The ASCII pane shows the corresponding characters, including "OIX" and "ASPack.0044669C".

Ahora volvemos a pulsar Ctrl. + O y volvemos a dejar desmarcadas todas las opciones menos la primera:



Aceptamos y le damos a F9 y empezamos a contar las excepciones incluida la primera que te da a al darle a F9. En mi caso, a la 10 ya arranca el programa así que vuelvo a reiniciar el Olly y vuelvo a hacer todo igual para reparar la IAT hasta volver a llegar a este punto donde ya se que tengo que parar en la excepción 9ª en mi caso. Una vez parado en la ultima excepción antes de que arranque el programa, voy al Memory Map con Alt + M y pongo un Breakpoint on Memory Access en la sección .code del ejecutable y le doy a F7 y F9 y nos parara en el OEP y esta vez con la IAT prácticamente reparada.

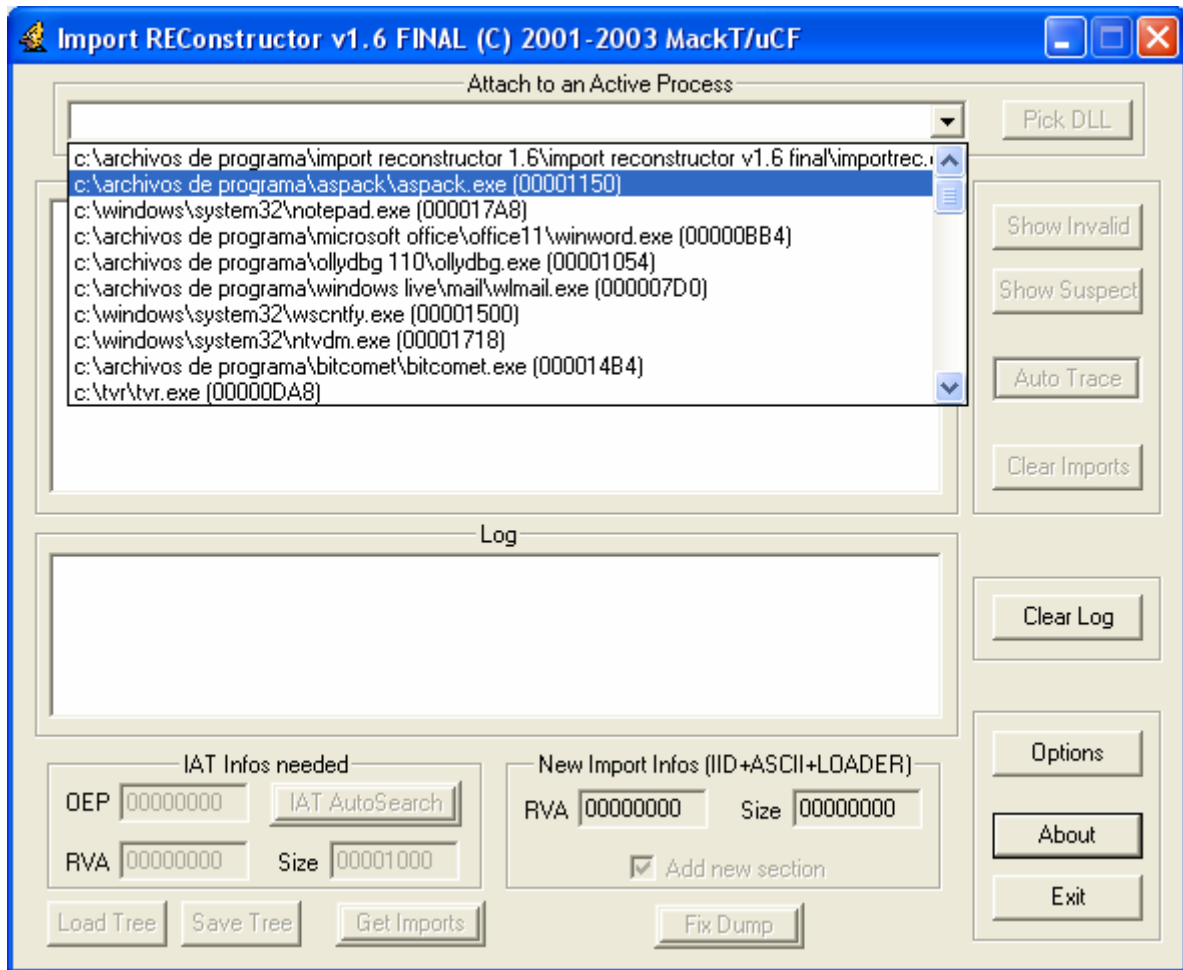


Nos aparece una ventana como la de arriba a la cual le desmarcamos la opción Rebuild Import y le damos a Dump y en la siguiente ventana elegimos un nombre y una ubicación y le damos a Guardar y ya tenemos nuestro dumpeado.



Ahora le vamos a reparar la IAT y para ello usaremos Import Rec.

Lo ejecutamos y como se muestra en la siguiente imagen, buscamos en la lista nuestro proceso para atacharlo y hacemos clic encima.



Ahora llega el momento de meter el OEP, RVA y Size y para eso lo apuntamos todo antes.

El Oep es el que tenemos apuntado menos la Image Base que es 400000.

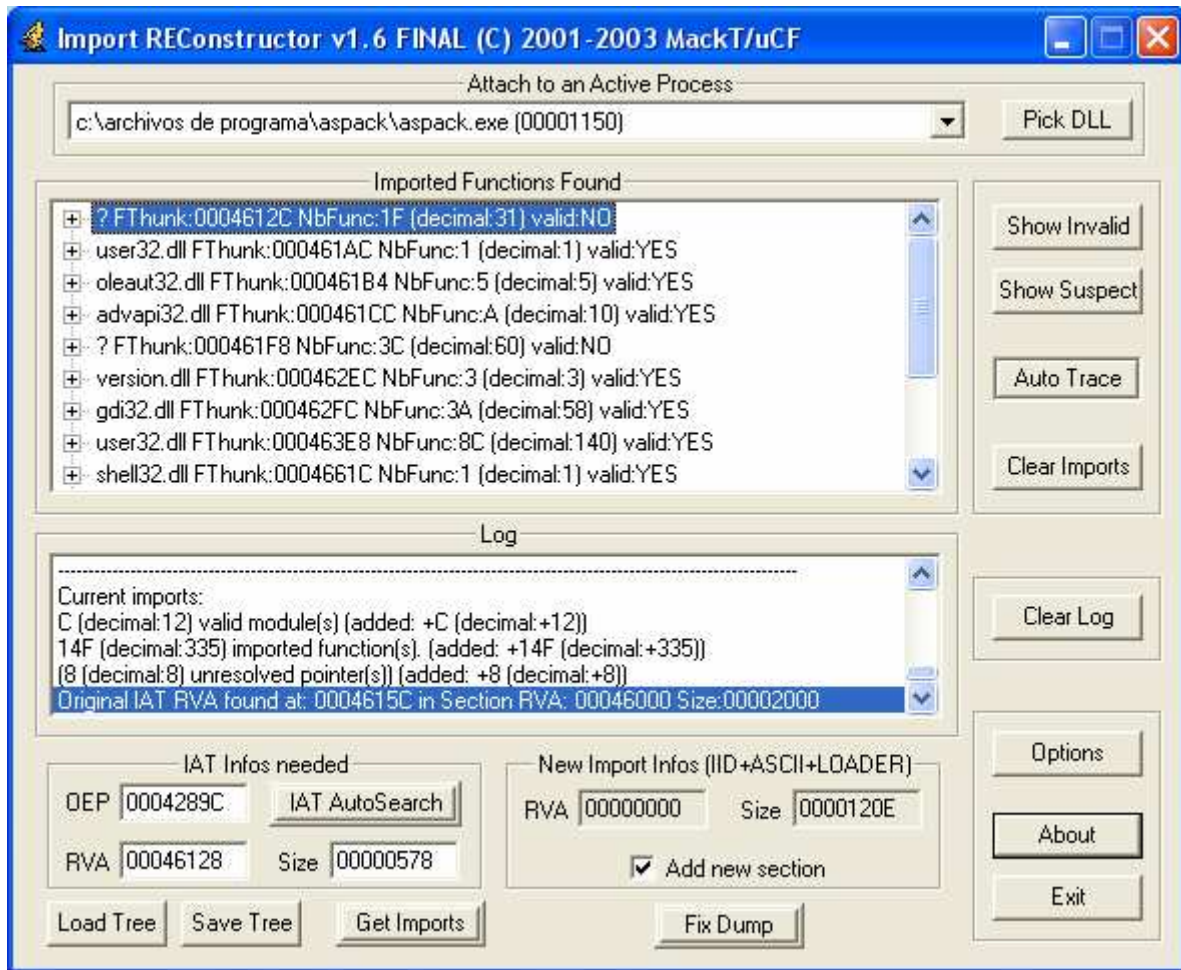
$44289C - 400000 = 4289C$ Axial que ya tenemos el dato para meter en el OEP.

El RVA es el inicio de nuestra IAT menos la Image Base así que:

$446128 - 400000 = 46128$

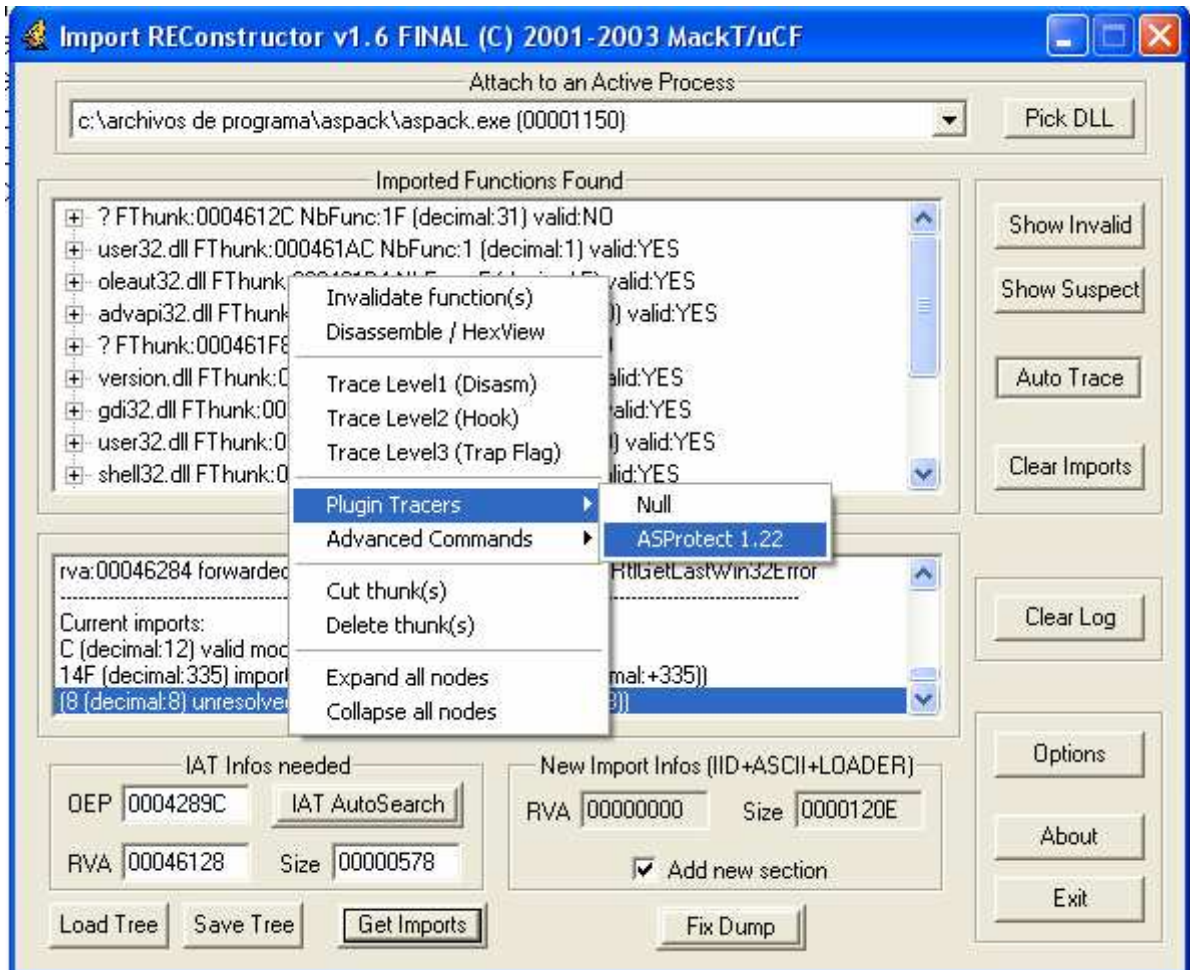
Y el Size es el tamaño de nuestra IAT.

Metamos solo el OEP y demos al botón IAT AutoSearch a ver que pasa.

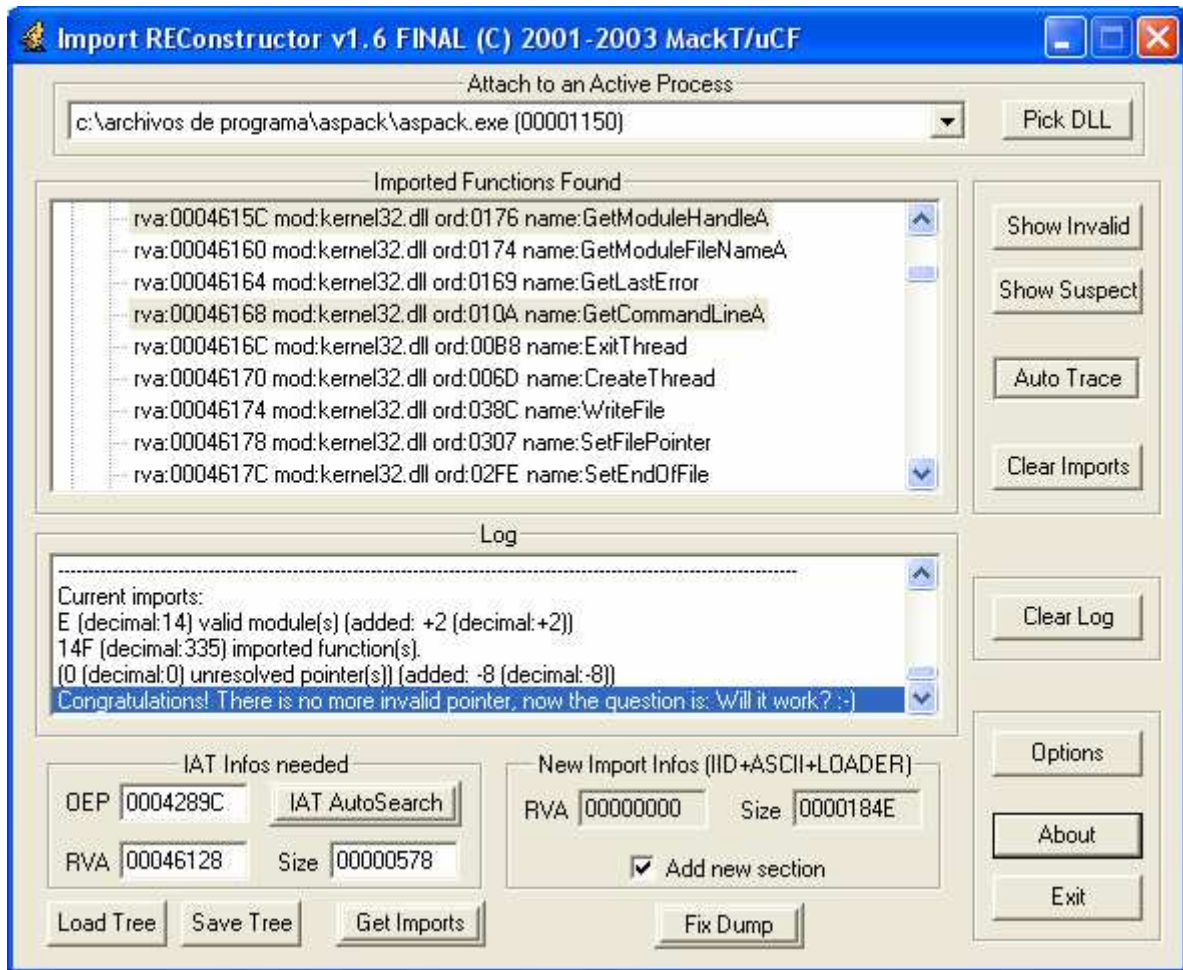


Como vemos, todos los datos coinciden así que vamos bien. Ahora demos al botón *Get Imports* y nos deben de aparecer todas las llamadas a las apis que realiza nuestra IAT.

Pero hay un problema, como pueden ver arriba, hay entradas malas que tendremos que resolver nosotros pero antes me queda un ultimo cartucho, usaremos el plugin para Asprotect 1.22 que se le puede meter al ImportRec y que en mi caso ya lo trae así que hago clic en el botón *Show Invalid* y acto seguido clic derecho encima de la lista que aparece y ejecuto el plugin.



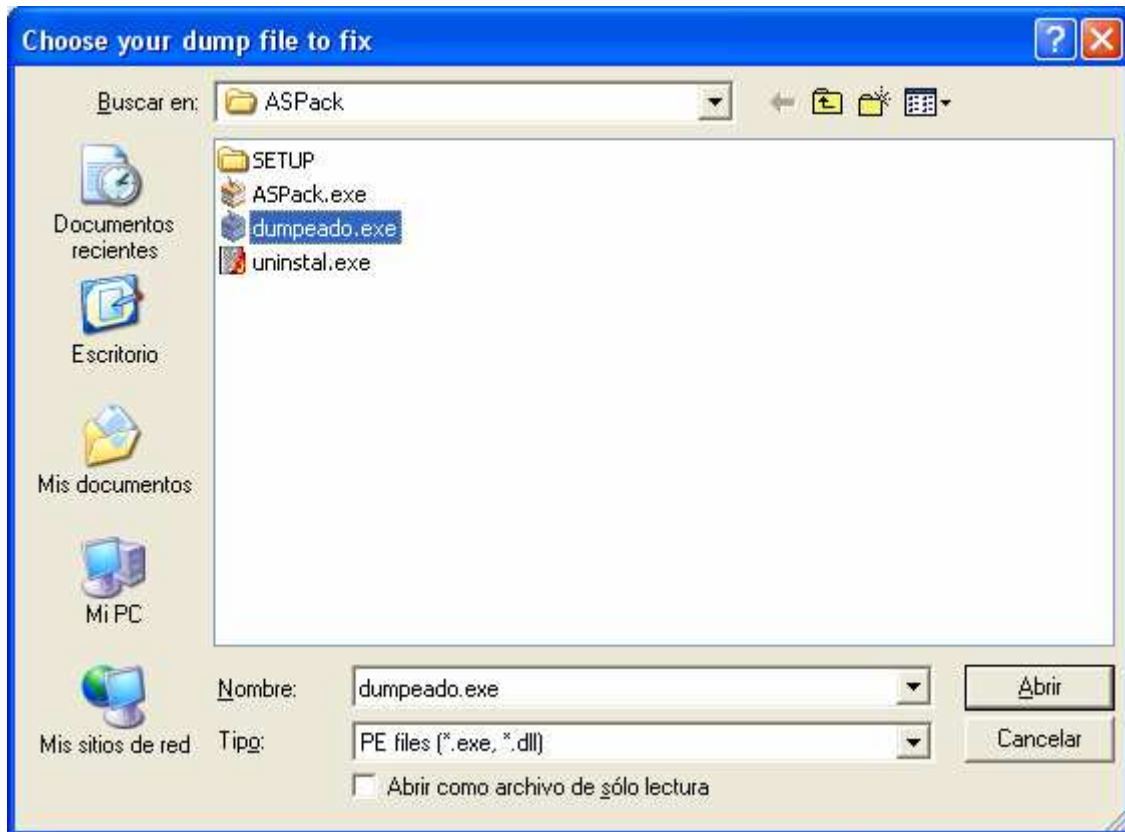
Y el resultado es...



Como se puede apreciar, en el Log nos dice que ya no quedan mas entradas malas por resolver y nos felicita.

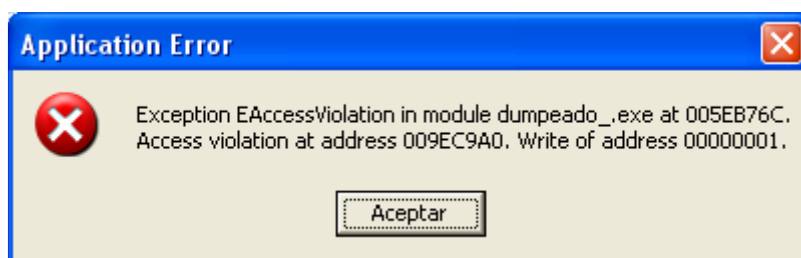
¡ Ya la tenemos reparada!

Ahora demos al botón Fix Dump y elijamos el dumpeado que hicimos antes:



Le damos a Abrir y...

En el Log nos dice que el archivo se guardo sin problemas así que intentemos ejecutar nuestro dumpeado a ver que pasa. Ojo, el dumpeado con la IAT correcta se guardara como dumpeado_.exe.



Bueno, este Asprotect tiene algunos ases en la manga pero intentaremos derrotarlo. Carguémoslo en otro Olly .

Bueno, aparecemos en el OEP bueno y el método que voy a usar para sacar el error es el siguiente.

En el Olly con el dumpeado, dejo pulsada F8 hasta que me da la siguiente excepción:

Access violation when writing to [00000001]

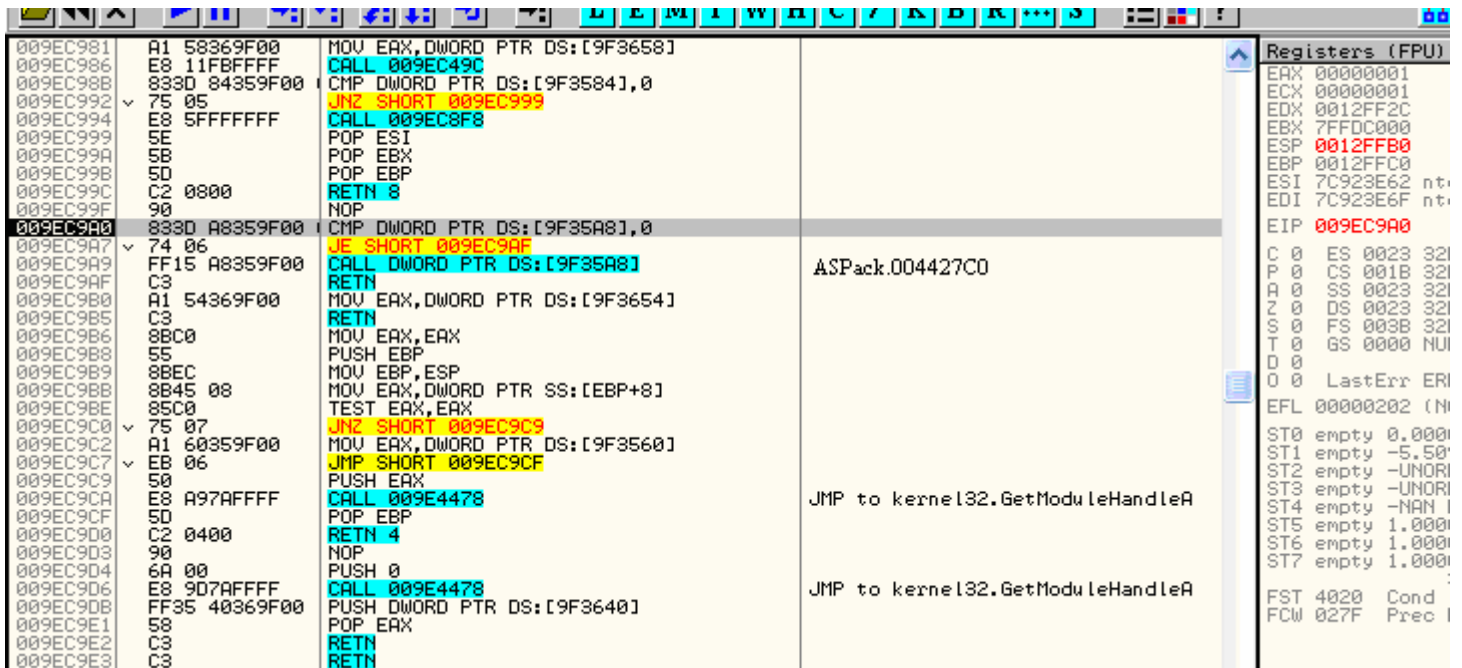
Si miramos la pila vemos justo arriba del todo:

0012FFB0 0044291B RETURN to dumpeado.0044291B from 009EC9A0

O sea, que donde estamos hemos llegado desde un CALL y si todo hubiera ido bien nos habría retornado a 0044291B así que vayamos allí a ver que vemos. Para ello hacemos clic derecho sobre esa entrada de la pila y elegimos Follow in Disassembler o pulsamos Intro y apareceremos en el retorno de esa CALL.

Es curioso pero estamos unas líneas mas abajo del OEP y el CALL que nos mando allí es el que esta justo encima y es el único de todos los que vemos ahí que usa una dirección indirecta.

Pues bien, vayamos al Olly que tiene el programa original parado en el OEP y vayamos con F8 hasta llegar a ese CALL sin ejecutarlo y cuando estemos encima de el pulsamos F7y vemos:



Pues vale, guardemos esta dirección por si nos volviera a hacer falta y ahora con F7 vamos traceando a ver que pasa y veremos que no se cumple la condición y el salto no se realiza y nos quedamos parados un momento en el CALL que esta justo después del

salto para ver a donde va ya que justo después tenemos un RETN y ese CALL es lo ultimo que se ejecutara en ese trozo de código y veremos esto:

```
009EC9A9 FF15 A8359F00 CALL DWORD PTR DS:[9F35A8] ;  
ASPack.004427A8
```

O sea, que eso es lo mismo que un CALL 004427A8, pues apuntemos eso también que va ha ser vital.

Ahora vamos a comprobar algo; estando parado en el CALL, pon un BP en el salto que tienes justo encima y que no se realizo antes porque la condición no se cumplía. Ahora quitemos el BPM y demos a F9 a ver si vuelve a parar ahí.

Bueno pues ya os digo que no para y como 004427C0 esta dentro del código del ejecutable pues vamos a hacer algo.

Sabemos que:

```
00442915 FF15 0C494400 CALL DWORD PTR DS:[44490C]
```

Nos manda de cabeza a ...

```
009EC9A9 FF15 A8359F00 CALL DWORD PTR DS:[9F35A8] ;  
ASPack.004427A8
```

Y que este ultimo CALL es igual que hacer CALL 004427C0

Pues me la juego, voy a hacer algo interesante porque no tendré ni que crear ningún tipo de injerto especial ni nada, simplemente cambiare al primer CALL para que vaya al lugar que va el otro CALL ya que el error que me da en el dumpeado es porque el salto indirecto del primer CALL no esta bien y con esto podríamos solucionarlo.

Quedaría así:

004428D9	E8 837FEFF	CALL 0042C538	dumpeado.0042C538	A 0
004428DE	E8 CEAF8FF	CALL 0042CEAC	dumpeado.0042CEAC	Z 1
004428E3	E8 95AB8FF	CALL 0042D478	dumpeado.0042D478	S 0
004428E8	E8 30BA8FF	CALL 0042E318	dumpeado.0042E318	T 0
004428ED	E8 6FBA8FF	CALL 0042E35C	dumpeado.0042E35C	D 0
004428F2	E8 FABAF8FF	CALL 0042E3EC	dumpeado.0042E3EC	O 0
004428F7	E8 69E18FF	CALL 00430A60	dumpeado.00430A60	O 0
004428FC	E8 A4AF8FF	CALL 0043D8A0	dumpeado.0043D8A0	EFL
00442901	A1 30564400	MOV EAX,DWORD PTR DS:[445630]		ST0
00442906	E8 D2108FF	CALL 004239D8	dumpeado.004239D8	ST1
0044290B	BA 38294400	MOV EDX,442938	ASCII "ASPack"	ST2
00442910	A1 30564400	MOV EAX,DWORD PTR DS:[445630]		ST3
00442915	E8 DF0D8FF	CALL 004236F4	dumpeado.004236F4	ST4
0044291A	E8 8E8E8FF	CALL 004427A8	dumpeado.004427A8	ST5
0044291B	90	NOP		ST6
00442920	A1 30564400	MOV EAX,DWORD PTR DS:[445630]		ST7
00442925	E8 53118FF	CALL 00423A78	dumpeado.00423A78	FST
0044292A	E8 D21B8FF	CALL 004044FC	dumpeado.004044FC	FCW
0044292E	8BE5	MOV ESP,EBP		
00442932	5D	POP EBP		
00442938	C3	RETN		
0044293D	0000	ADD BYTE PTR DS:[EAX],AL		
00442941	FFFF	?	Unknown command	
00442947	FFFF	?	Unknown command	
0044294C	06	PUSH ES		
00442950	0000	ADD BYTE PTR DS:[EAX],AL		
00442956	0041 53	ADD BYTE PTR DS:[ECX+53],AL		
0044295B	50	PUSH EAX		
00442960	61	POPAD		
00442966	636B 00	ARPL WORD PTR DS:[EBX],BP		
0044296C	0000	ADD BYTE PTR DS:[EAX],AL		

Ahora guardemos los cambios seleccionando las líneas que hemos modificado y haciendo clic derecho y dándole a Copy to Executable->Selección y acto seguido reiniciemos el Olly que tiene el dumpeado y ejecutémolo a ver que pasa ahora.

Y nos vuelve a dar otro error igual que el de antes. Axial que lo mismo, miremos en el Stack a ver si hay suerte y...

0012FE00 0043F1BE RETURN to dumpeado.0043F1BE from 009EC8F4

Pues vallamos a 0043F1BE colocándonos encima de esa entrada del Stack y dándole a Intro a ver que se cuece por allí.

0043F188	33D2	XOR EDX,EDX			
0043F18D	8B45 FC	MOV EAX,DWORD PTR SS:[EBP-4]			ST0 empty
0043F190	8B80 58020000	MOV EAX,DWORD PTR DS:[EAX+258]			ST1 empty
0043F196	E8 FD3EFDFF	CALL 00413098	dumpeado.00413098		ST2 empty
0043F198	8B45 FC	MOV EAX,DWORD PTR SS:[EBP-4]			ST3 empty
0043F19E	8B80 E4020000	MOV EAX,DWORD PTR DS:[EAX+2E4]			ST4 empty
0043F1A4	C640 20 00	MOV BYTE PTR DS:[EAX+20],0			ST5 empty
0043F1A8	33D2	XOR EDX,EDX			ST6 empty
0043F1AA	8B45 FC	MOV EAX,DWORD PTR SS:[EBP-4]			ST7 empty
0043F1AD	8B80 7C020000	MOV EAX,DWORD PTR DS:[EAX+27C]			
0043F1B3	E8 1C3FFDFF	CALL 004130D4	dumpeado.004130D4		FST 4020
0043F1B8	FF15 04494400	CALL DWORD PTR DS:[444904]			FCW 027F
0043F1BE	E9 83000000	JMP 0043F246	dumpeado.0043F246		
0043F1C3	26:F4	HLT	Privileged command		
0043F1C5	CC	INT3			
0043F1C6	46	INC ESI			
0043F1C7	F2:	PREFIX REPNE:	Superfluous prefix		
0043F1C8	B8 A1340EBD	MOV EAX,BD0E34A1			
0043F1CD	59	POP ECX			
0043F1CE	6B0F 05	IMUL ECX,DWORD PTR DS:[EDI],5			
0043F1D1	F8	CLC			
0043F1D2	58	POP EAX			
0043F1D3	DBD8	FCMOVNU ST,ST			
0043F1D5	4C	DEC ESP			
0043F1D6	EB B0	JMP SHORT 0043F188	dumpeado.0043F188		
0043F1D8	FC 2520DC52	MOV EBP,520C202E			

Pues miremos en el original a ver que hace el CALL que tenemos justo arriba que es el culpable del error. Simplemente vamos a la dirección donde esta la CALL y damos a Intro y, sorpresa sorpresa, jeje, lo único que hace es ir a esa zona que se crea en ejecución y donde solo se encuentra un RETN con lo cual ese CALL es simplemente una trampa para que no lo podamos dumpear limpiamente así que nopeemos ese CALL en el dumpeado y quedara así:

0043F18D	8B45 FC	MOV EAX,DWORD PTR SS:[EBP-4]			
0043F190	8B80 58020000	MOV EAX,DWORD PTR DS:[EAX+258]			
0043F196	E8 FD3EFDFF	CALL 00413098	dumpeado.00413098		
0043F198	8B45 FC	MOV EAX,DWORD PTR SS:[EBP-4]			
0043F19E	8B80 E4020000	MOV EAX,DWORD PTR DS:[EAX+2E4]			
0043F1A4	C640 20 00	MOV BYTE PTR DS:[EAX+20],0			
0043F1A8	33D2	XOR EDX,EDX			
0043F1AA	8B45 FC	MOV EAX,DWORD PTR SS:[EBP-4]			
0043F1AD	8B80 7C020000	MOV EAX,DWORD PTR DS:[EAX+27C]			
0043F1B3	E8 1C3FFDFF	CALL 004130D4	dumpeado.004130D4		
0043F1B8	90	NOP			
0043F1B9	90	NOP			
0043F1BA	90	NOP			
0043F1BB	90	NOP			
0043F1BC	90	NOP			
0043F1BD	90	NOP			
0043F1BE	E9 83000000	JMP 0043F246	dumpeado.0043F246		
0043F1C3	26:F4	HLT	Privileged command		
0043F1C5	CC	INT3			
0043F1C6	46	INC ESI			
0043F1C7	F2:	PREFIX REPNE:	Superfluous prefix		
0043F1C8	B8 A1340EBD	MOV EAX,BD0E34A1			
0043F1CD	59	POP ECX			
0043F1CE	6B0F 05	IMUL ECX,DWORD PTR DS:[EDI],5			
0043F1D1	F8	CLC			

Volvamos a guardar los cambios realizados y volvamos a reiniciar el Olly que tiene el dumpeado y volvamos a ejecutarlo a ver que pasa ahora y...

Otra vez igual, y esto empieza a desesperar. Ahora en el Stack tengo esto:

0012FE00 0043F24C RETURN to dumpeado.0043F24C from 009EC8F4

Axial que, como en las otras ocasiones, me planto encima de esa entrada y le doy a Intro y llego a:

0043F240	D4	DB D4		
0043F241	98	DB 98		
0043F242	BA	DB BA		
0043F243	9D	DB 9D		
0043F244	29	DB 29		
0043F245	8D	DB 8D		CHAR ')'
0043F246	> FF15 0849440	CALL DWORD PTR DS:[444908]		
0043F24C	. B2 01	MOV DL,1		
0043F24E	. B8 54B04200	MOV EAX,42B054		
0043F253	. E8 04BFFEFF	CALL 0042B15C		dumpeado.0042B15C
0043F258	. 8945 E0	MOV DWORD PTR SS:[EBP-20],EAX		
0043F25B	. 33C0	XOR EAX,EAX		
0043F25D	. 55	PUSH EBP		
0043F25E	. 68 B2F24300	PUSH 43F2B2		
0043F263	. 64:FF30	PUSH DWORD PTR FS:[EAX]		
0043F266	. 64:8920	MOV DWORD PTR FS:[EAX],ESP		
0043F269	. B1 01	MOV CL,1		
0043F26B	. BA 44F54300	MOV EDX,43F544		ASCII "Software\ASPack\Options"
0043F270	. 8B45 E0	MOV EAX,DWORD PTR SS:[EBP-20]		
0043F273	. E8 BCC0FEFF	CALL 0042B334		dumpeado.0042B334
0043F278	. 8D4D D4	LEA ECX,DWORD PTR SS:[EBP-2C]		
0043F27B	. BA 64F54300	MOV EDX,43F564		ASCII "Lang_LanguageFile"
0043F280	. 8B45 E0	MOV EAX,DWORD PTR SS:[EBP-20]		
0043F283	. E8 28C4FEFF	CALL 0042B6B0		dumpeado.0042B6B0
0043F288	. 8B55 D4	MOV EDX,DWORD PTR SS:[EBP-2C]		
0043F28B	. 8B45 FC	MOV EAX,DWORD PTR SS:[EBP-4]		
0043F28E	. 8B80 E401000	MOV EAX,DWORD PTR DS:[EAX+1E4]		
0043F294	. 83C0 20	ADD EAX,20		
0043F297	. E8 E842FCFF	CALL 00403584		dumpeado.00403584
0043F29C	. 33C0	XOR EAX,EAX		
0043F29E	. 5A	POP EDX		
0043F29F	. 59	POP ECX		
0043F2A0	. 59	POP ECX		
0043F2A1	. 64:8910	MOV DWORD PTR FS:[EAX],EDX		

Vuelvo a ir al Olly que tiene el original y voy a la dirección donde esta esa CALL que vemos encima de donde retornamos y vemos que otra vez es lo mismo, un CALL que nos manda a un lugar de la memoria creado en ejecución y que lo único que ejecuta allí es un RETN. Pero claro, en el dumpeado tampoco existe ese lugar y, por tanto, no existe ese RETN pero como ya dije, un CALL con un RETN detrás no sirve absolutamente para nada así que nopeamos esa CALL en el dumpeado y volvemos a guardar de nuevo los cambios.

Volvemos a reiniciar el Olly con el dumpeado y volvemos a ejecutarlo y...



Por fin, jeje. Pero eso de los 0 días es porque modifique la fecha haciendo pruebas pero te saldrá la cantidad de días que te quedan. Eso si, para que mi método funcione y no caduque, tienes que descomprimirlo antes de que caduque ya que la comprobación se encuentra en la primera CALL que modificamos y, no se como, pero, cuando caduca, en el original esa CALL coge otro valor que esta justo debajo del destino que tiene ahora y si vais veréis algo muy curioso allí jeje.

Bueno, yo suponía que se había acabado el trabajo pero me dio por probar a comprimir algún exe y cuando acepto la ventanita que me dice que va a comprimir el archivo y le digo que si, vuelve a saltar otro error como los anteriores así que veamos a ver donde esta el fallo.

Esta vez en el Stack tenemos esto:

```
0012F8DC 004410FE RETURN to dumpeado.004410FE from 009EC8F4
```

Usando el mismo método que en todas las otras fallas, llego a:

004410EC	. 8B45 FC	MOV EAX,DWORD PTR SS:[EBP-4]		
004410EF	. 8A80 10030000	MOV AL,BYTE PTR DS:[EAX+310]		
004410F5	. 8B43 53	MOV BYTE PTR DS:[EBX+53],AL		
004410F8	. FF15 04494400	CALL DWORD PTR DS:[444904]		
004410FE	. E9 44000000	JMP 00441147	dumpeado.00441147	
00441103	D6	DB D6		
00441104	0F	DB 0F		
00441105	D8	DB D8		
00441106	19	DB 19		
00441107	22	DB 22	CHAR '''	
00441108	00	DB 00		
00441109	90	DB 90		
0044110A	14	DB 14		
0044110B	A3	DB A3		
0044110C	6E	DB 6E	CHAR 'n'	
0044110D	3D	DB 3D	CHAR '='	
0044110E	41	DB 41	CHAR 'A'	
0044110F	93	DB 93		
00441110	36	DB 36	CHAR '6'	
00441111	86	DB 86		
00441112	45	DB 45	CHAR 'E'	
00441113	4C	DB 4C	CHAR 'L'	
00441114	8E	DB 8E		
00441115	FF	DB FF		
00441116	D3	DB D3		
00441117	8C	DB 8C		
00441118	73	DB 73	CHAR 's'	
00441119	19	DB 19		
0044111A	1C	DB 1C		
0044111B	C0	DB C0		
0044111C	CA	DB CA		
0044111D	8F	DB 8F		

Me vuelvo a ir a la dirección de este CALL en el original y una vez encima pulso Intro y otra vez la misma historia, otro RETN así que a nopear ese CALL en el dumpeado y a guardar los cambios.

Otra vez vuelvo a intentarlo y esta vez al volver a aceptar la ventanita que sale para comprimir me vuelve a dar otro error y esta vez en el Stack tenemos:

0012F8DC 0044114D RETURN to dumpeado.0044114D from 009EC8F4

Y en la dirección de retorno:

00441140	B6	DB B6		
00441141	FE	DB FE		
00441142	06	DB 06		
00441143	0C	DB 0C		
00441144	6E	DB 6E	CHAR 'n'	
00441145	CE	DB CE		
00441146	1A	DB 1A		
00441147	> FF15 04494400	CALL DWORD PTR DS:[444908]		
0044114D	. 8B45 FC	MOV EAX,DWORD PTR SS:[EBP-4]		
00441150	. 8B80 08030000	MOV EAX,DWORD PTR DS:[EAX+308]		
00441156	. 05 98000000	ADD EAX,98		
0044115B	. 8B55 FC	MOV EDX,DWORD PTR SS:[EBP-4]		
0044115E	. 8B92 04030000	MOV EDX,DWORD PTR DS:[EDX+304]		
00441164	. E8 1B24FCFF	CALL 00403584	dumpeado.00403584	
00441169	. BA 7CED4300	MOV EDX,43ED7C		
0044116E	. 8B45 FC	MOV EAX,DWORD PTR SS:[EBP-4]		
00441171	. 8B80 08030000	MOV EAX,DWORD PTR DS:[EAX+308]		
00441177	. 8990 7C020000	MOV DWORD PTR DS:[EAX+27C],EDX		
0044117D	. 8B10	MOV EDX,DWORD PTR DS:[EAX]		
0044117F	. FF92 C4000000	CALL DWORD PTR DS:[EDX+C4]		
00441185	. 33C0	XOR EAX,EAX		

Pues hago oootra vez lo mismo y oootra vez igual, otro RETN así que a nopear la CALL y a volver a guardar los cambios.

Vuelvo a reiniciar Olly un poco cansado ya de esto y vuelvo a intentarlo y esta vez casi termina pero al final de la compresión salta otro error y siguiendo otra vez el método pues tengo en el Stack:

0012F92C 00441E29 RETURN to dumpeado.00441E29 from 009EC8F4

Y en el retorno:

00441E01	8B93 2C020000	MOV EDX,DWORD PTR DS:[EBX+22C]		
00441E07	8B83 F0010000	MOV EAX,DWORD PTR DS:[EBX+1F0]		
00441E0D	E8 E6E3FEFF	CALL 004301F8	dumpeado.004301F8	
00441E12	8BC3	MOV EAX,EBX		
00441E14	E8 87F0FFFF	CALL 00440EA0	dumpeado.00440EA0	
00441E19	FF45 FC	INC DWORD PTR SS:[EBP-4]		
00441E1C	4E	DEC ESI		
00441E1D	^ 0F85 B9FEFFFF	JNZ 00441CDC	dumpeado.00441CDC	
00441E23	FF15 04494400	CALL DWORD PTR DS:[444904]		
00441E29	▼ E9 3E000000	JMP 00441E6C	dumpeado.00441E6C	
00441E2E	17	POP SS		
00441E2F	11E3	ADC EBX,ESP	Modification of segment register	
00441E31	C6	?	Unknown command	
00441E32	BE 56D5EECA	MOV ESI,CAEED556		
00441E37	BE 2CA6DDF1	MOV ESI,F1DDA62C		
00441E3C	4C	DEC ESP		
00441E3D	2366 AA	AND ESP,DWORD PTR DS:[ESI-56]		
00441E40	▼ 76 05	JBE SHORT 00441E47	dumpeado.00441E47	
00441E42	CD AB	INT 0AB		
00441E44	24 59	AND AL,59		
00441E46	BE 646117A6	MOV ESI,A6176164		
00441E4B	F0:25 D9A444D2	LOCK AND EAX,D244A4D9	LOCK prefix is not allowed	
00441E51	17	POP SS	Modification of segment register	
00441E52	4C	DEC ESP		
00441E53	13EB	ADC EBP,EBX		
00441E55	038C61 088CDD7	ADD ECX,DWORD PTR DS:[ECX+77DD0808]		
00441E5C	D7	XLAT BYTE PTR DS:[EBX+AL]		
00441E5D	36:FD	STD	Superfluous prefix	
00441E5F	19AC21 9256AEB	SBB DWORD PTR DS:[ECX+BEAE5692],EBP		
00441E66	2F	DAS		
00441E67	5E	POP ESI		

Bueno, pues miremos esa CALL en el original a ver que es.

Y otra vez la misma historia así que a nopear también esa CALL en el dumpeado y a guardar los cambios.

Lo volvemos a intentar y otra vez mas el mismo error así que miremos otra vez a ver si esto se acaba ya.

En el Stack tenemos:

0012F92C 00441E72 RETURN to dumpeado.00441E72 from 009EC8F4

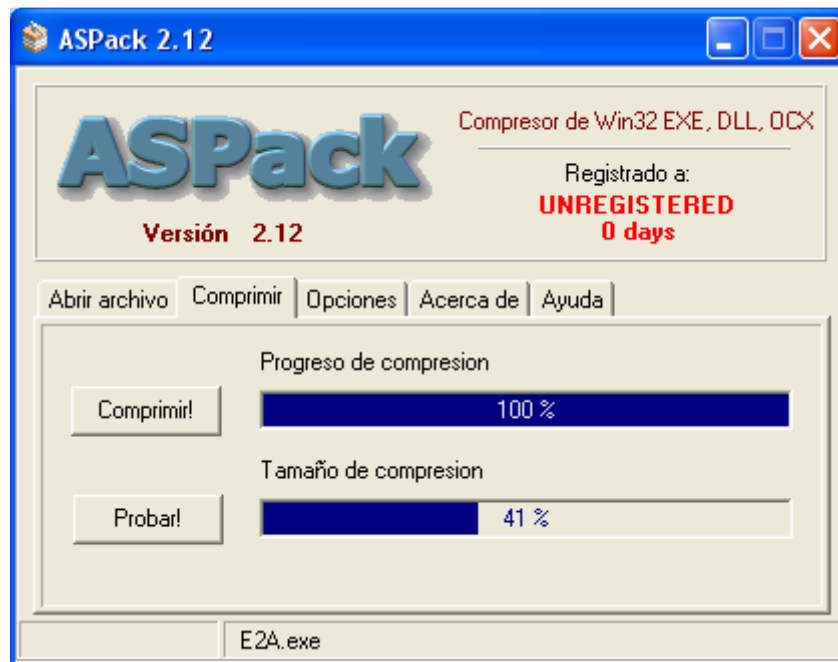
Y en el retorno:

```

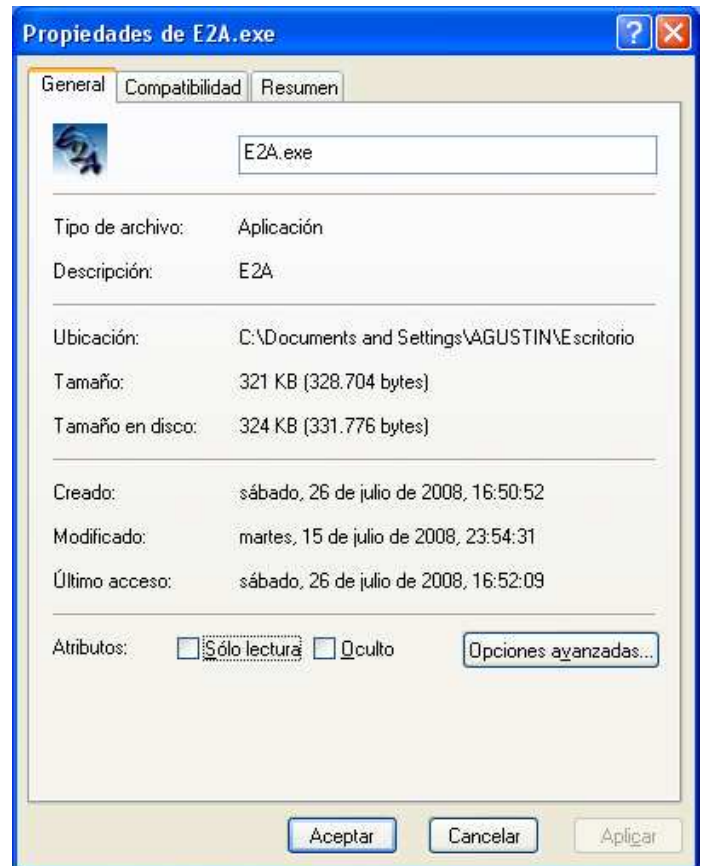
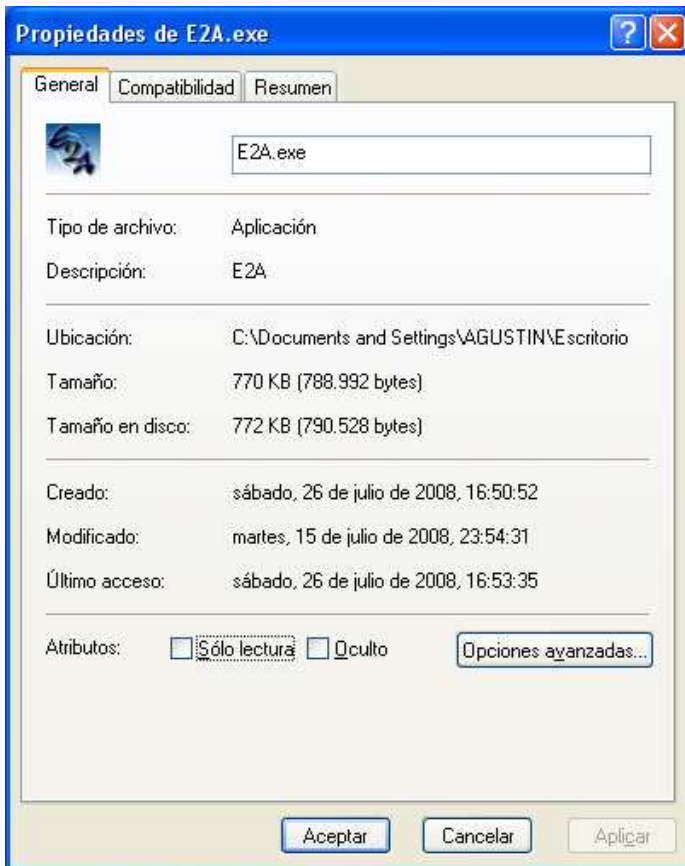
00441E65 | BE | DB BE
00441E66 | 2F | DB 2F | CHAR '/'
00441E67 | 5E | DB 5E | CHAR '^'
00441E68 | 95 | DB 95
00441E69 | 0D | DB 0D
00441E6A | 66 | DB 66 | CHAR 'f'
00441E6B | 11 | DB 11
00441E6C | > FF15 0849440 | CALL DWORD PTR DS:[444908]
00441E72 | > 33C0 | XOR EAX,EAX
00441E74 | . 5A | POP EDX
00441E75 | . 59 | POP ECX
00441E76 | . 59 | POP ECX
00441E77 | . 64:8910 | MOV DWORD PTR FS:[EAX],EDX
00441E7A | . 68 941E4400 | PUSH 441E94
00441E7F | > 8D45 F4 | LEA EAX,DWORD PTR SS:[EBP-C]
00441E82 | . BA 02000000 | MOV EDX,2
00441E87 | . E8 C816FCFF | CALL 00403554
00441E8C | . C3 | RETN
00441E8D | . ^ E9 8212FCFF | JMP 00403114
00441E92 | . ^ EB EB | JMP SHORT 00441E7F
00441E94 | . . 5F | POP EDI
00441E95 | . . 5E | POP ESI
00441E96 | . . 5B | POP EBX
00441E97 | . . 8BES | MOV ESP,EBP
00441E99 | . . 5D | POP EBP
00441E9A | . . C3 | RETN
00441E9B | . . 00 | DB 00
00441E9C | . . FFFFFFFF | DD FFFFFFFF
00441E9D | . . 00000000 | DD 00000000
    
```

Y al mirar esa CALL en el original veo que es más de lo mismo así que a nopearla y a guardar el dumpeado.

Y vuelvo a intentarlo y...



Y a continuación veréis que si lo comprimíó y funciona perfectamente.



Bueno, el objetivo de este tuto esta cumplido pero queda muy feo eso de UNREGISTERED así que lo cambiaremos usando un editor hexadecimal buscando las palabras esas ya que no aparecen en el Olly y con el editor hexa funciona.

Lo abrimos con cualquier editor hexa y buscamos y a mí con el Pspad me aparece así:

```

72 0709 6276 4C6F 7765 7265 6408  uter..bvLowered.
4F 7264 6572 0200 0006 5450 616E  TabOrder....TPan
50 616E 656C 3204 4C65 6674 0201  el.Panel2.Left..
70 0201 0557 6964 7468 039C 0106  .Top...Width.α..
67 6874 0328 0105 416C 6967 6E07  Height.(..Align.
54 6F70 0854 6162 4F72 6465 7202  .alTop.TabOrder.
54 4265 7665 6C06 4265 7665 6C31  ...TBevel.Bevel1
66 7402 0803 546F 7002 0805 5769  .Left...Top...Wi
03 8C01 0648 6569 6768 7402 5905  dth.Ⓔ..Height.Y.
70 6507 0762 7346 7261 6D65 0553  Shape..bsFrame.S
65 0708 6273 5261 6973 6564 0000  tyle..bsRaised..
61 6265 6C0A 4C61 6265 6C54 7269  .TLabel.LabelTri
4C 6566 7403 E000 0354 6F70 023E  al.Left.à..Top.>
64 7468 03AC 0006 4865 6967 6874  .Width.↵..Height
41 6C69 676E 6D65 6E74 0708 7461  . .Alignment..ta
74 6572 0841 7574 6F53 697A 6508  Center.AutoSize.
70 7469 6F6E 060C 554E 5245 4749  .Caption..UNREGI
52 4544 0A46 6F6E 742E 436F 6C6F  STERED.Font.Colo
63 6C52 6564 0B46 6F6E 742E 4865  r..clRed.Font.He
74 02F5 0946 6F6E 742E 4E61 6D65  ight.ö.Font.Name
53 2053 616E 7320 5365 7269 660A  ..MS Sans Serif.
74 2E53 7479 6C65 0B06 6673 426F  Font.Style..fsBo
0A 5061 7265 6E74 466F 6E74 0808  ld..ParentFont..
    
```

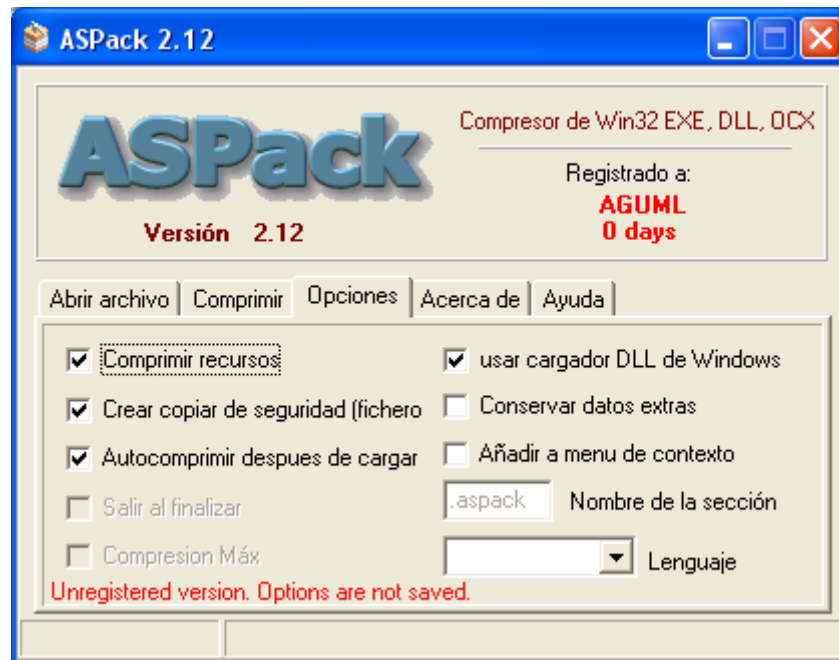
Y lo cambio por:

```

72 0709 6276 4C6F 7765 7265 6408  uter..bvLowered.
4F 7264 6572 0200 0006 5450 616E  TabOrder....TPan
50 616E 656C 3204 4C65 6674 0201  el.Panel2.Left..
70 0201 0557 6964 7468 039C 0106  .Top...Width.α..
67 6874 0328 0105 416C 6967 6E07  Height.(..Align.
54 6F70 0854 6162 4F72 6465 7202  .alTop.TabOrder.
54 4265 7665 6C06 4265 7665 6C31  ...TBevel.Bevel1
66 7402 0803 546F 7002 0805 5769  .Left...Top...Wi
03 8C01 0648 6569 6768 7402 5905  dth.Ⓔ..Height.Y.
70 6507 0762 7346 7261 6D65 0553  Shape..bsFrame.S
65 0708 6273 5261 6973 6564 0000  tyle..bsRaised..
61 6265 6C0A 4C61 6265 6C54 7269  .TLabel.LabelTri
4C 6566 7403 E000 0354 6F70 023E  al.Left.à..Top.>
64 7468 03AC 0006 4865 6967 6874  .Width.↵..Height
41 6C69 676E 6D65 6E74 0708 7461  . .Alignment..ta
74 6572 0841 7574 6F53 697A 6508  Center.AutoSize.
70 7469 6F6E 060C 2020 2020 4147  .Caption..  AG
20 2020 0A46 6F6E 742E 436F 6C6F  UML  .Font.Colo
63 6C52 6564 0B46 6F6E 742E 4865  r..clRed.Font.He
74 02F5 0946 6F6E 742E 4E61 6D65  ight.ö.Font.Name
53 2053 616E 7320 5365 7269 660A  ..MS Sans Serif.
74 2E53 7479 6C65 0B06 6673 426F  Font.Style..fsBo
    
```


Y guardo los cambios y voy a ver:

Fue bien pero hay mas cosillas que quitar de en medio como el rotulo que sale abajo cuando vas a la pestaña opciones.



Volvemos a abrirlo en el editor hexa y buscamos y llenamos esa frase con espacios y el resultado es:

```

442 6576 656C 0642 6576 656C 3404 ..TBevel.Bevel4.
674 0200 0354 6F70 0200 0557 6964 Left...Top...Wid
386 0106 4865 6967 6874 038D 0005 th.t...Height.D..
967 6E07 0861 6C43 6C69 656E 7400 Align..alClient.
44C 6162 656C 104C 6162 656C 5365 ..TLabel.LabelSe
96F 6E4E 616D 6504 4C65 6674 0307 ctionName.Left..
46F 7002 5205 5769 6474 6802 4806 ..Top.R.Width.H.
967 6874 020D 0743 6170 7469 6F6E Height...Caption
365 6374 696F 6E27 7320 6E61 6D65 ..Section's name
654 4C61 6265 6C09 4C61 6265 6C4C ...TLabel.LabelL
704 4C65 6674 032E 0103 546F 7002 ang.Left....Top.
769 6474 6802 3006 4865 6967 6874 p.Width.O.Height
743 6170 7469 6F6E 0608 4C61 6E67 ...Caption..Lang
765 0000 0654 4C61 6265 6C11 4C61 uage...TLabel.La
C55 6E72 6567 6973 7465 7265 6404 belUnregistered.
674 0204 0354 6F70 027F 0557 6964 Left...Top.D.Wid
3D8 0006 4865 6967 6874 020D 0743 th.Ø..Height...C
469 6F6E 062E 556E 7265 6769 7374 aption..Unregist
564 2076 6572 7369 6F6E 2E20 4F70 ered version. Op
F6E 7320 6172 6520 6E6F 7420 7361 tions are not sa
42E 2020 0A46 6F6E 742E 436F 6C6F ved;. .Font.Colo
563 6C52 6564 0B46 6F6E 742E 4865 r..clRed.Font.He
874 02F5 0946 6F6E 742E 4E61 6D65 ight.ö.Font.Name

```

```

44C 6162 656C 104C 6162 656C 5365 ..TLabel.LabelSe
96F 6E4E 616D 6504 4C65 6674 0307 ctionName.Left..
46F 7002 5205 5769 6474 6802 4806 ..Top.R.Width.H.
967 6874 020D 0743 6170 7469 6F6E Height...Caption
365 6374 696F 6E27 7320 6E61 6D65 ..Section's name
654 4C61 6265 6C09 4C61 6265 6C4C ...TLabel.LabelL
704 4C65 6674 032E 0103 546F 7002 ang.Left....Top.
769 6474 6802 3006 4865 6967 6874 p.Width.O.Height
743 6170 7469 6F6E 0608 4C61 6E67 ...Caption..Lang
765 0000 0654 4C61 6265 6C11 4C61 uage...TLabel.La
C55 6E72 6567 6973 7465 7265 6404 belUnregistered.
674 0204 0354 6F70 027F 0557 6964 Left...Top.D.Wid
3D8 0006 4865 6967 6874 020D 0743 th.Ø..Height...C
469 6F6E 062E 2020 2020 2020 2020 aption..
020 2020 2020 2020 2020 2020 2020
020 2020 2020 2020 2020 2020 2020
020 2020 0A46 6F6E 742E 436F 6C6F .Font.Colo
563 6C52 6564 0B46 6F6E 742E 4865 r..clRed.Font.He
874 02F5 0946 6F6E 742E 4E61 6D65 ight.ö.Font.Name
D53 2053 616E 7320 5365 7269 660A ..MS Sans Serif.
F74 2E53 7479 6C65 0B00 0A50 6172 Font.Style...Par

```

Otra cosilla que le voy a hacer es poner por defecto el lenguaje español ya que este exe no guarda la configuración, pues por lo menos que salga de entrada en español y lo

voy a hacer con el editor hexa también. Simplemente busco "english.ini" que es el que esta por defecto y solo aparece una entrada así que nos lo pone fácil.

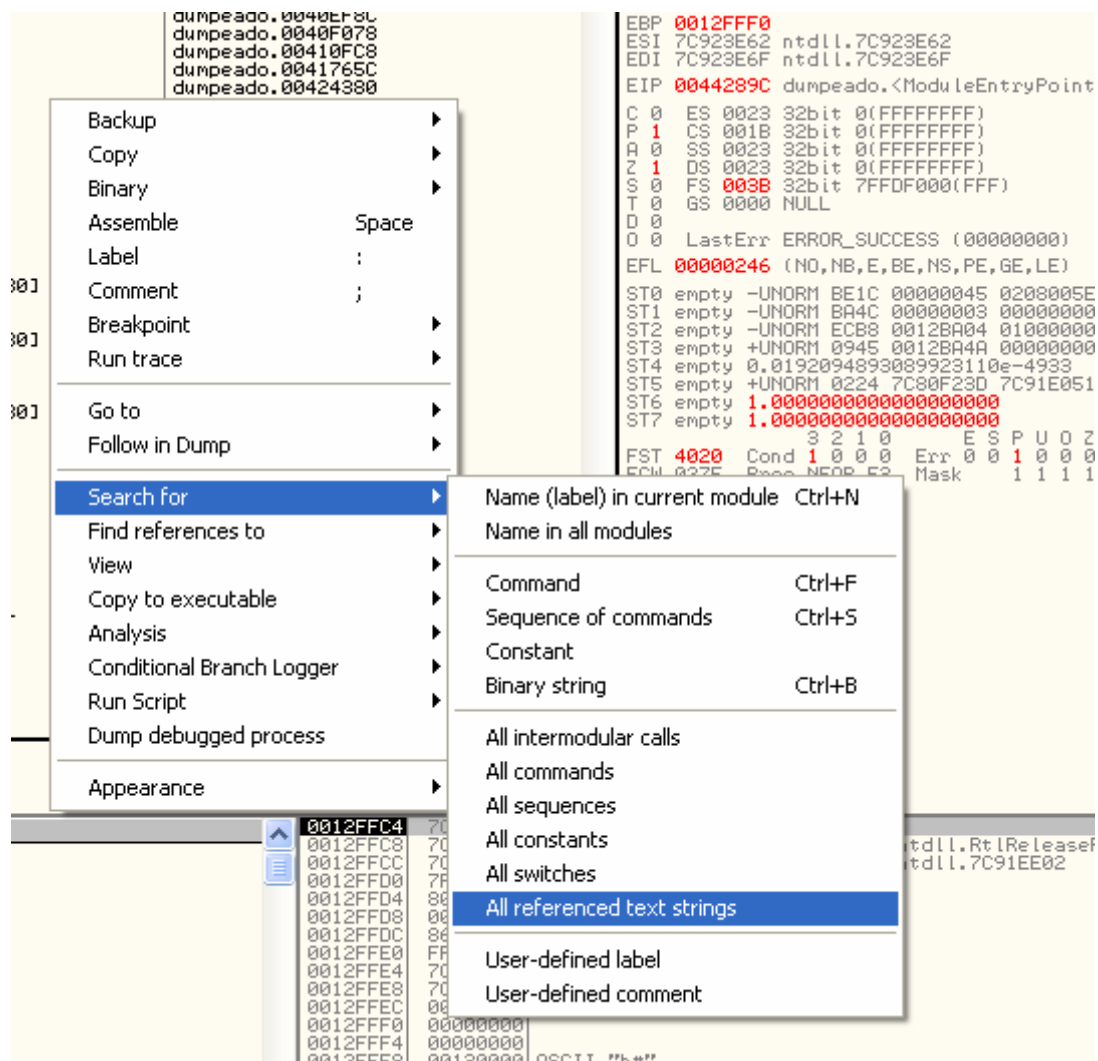
EFF	8B45	FC8B	80E4	0100	008B	4820	°°pÿ<Eü<€ä...<H
444	008B	45F0	E8E7	B3FE	FF33	C05A	°..D.<Eðèç°pÿ3ÀZ
489	1068	BA02	4400	8B45	FOE8	5629	YYd%.h°.D.<EðèV)
3E9	5C2E	FCFF	EBF0	33C0	5A59	5964	üÿÃé\ .üÿèð3ÀZYd
8D7	0244	008B	45F4	E839	29FC	FFC3	%.h×.D.<Eðè9)üÿÃ
EFC	FFEB	F080	7DFB	0075	246A	1068	é?.üÿèð€)û.u\$ÿ.h
400	6828	0444	008B	45FC	E807	65FD	.D.h(.D.<Eüè.eÿ
85D	4EFC	FFA1	3056	4400	E8E7	37FE	ÿPè]Nüÿ;OVD.èç7p
5FC	8B80	E401	0000	E84D	6BFF	FF8B	ÿ<Eü<€ä...èMkÿÿ<
B80	9002	0000	E8E7	74FE	FF8B	55FC	Eü<€Ð...èçtpÿ<Uü
401	0000	8942	0C33	C05A	5959	6489	<'ä...%B.3ÀZYd%
603	4400	8D45	E4E8	F631	FCFF	8D45	.hV.D.ÐEäèð1üÿÐE
E31	FCFF	8D45	ECE8	E631	FCFF	C3E9	èèí1üÿÐEièæ1üÿÃé
CFF	EBE0	5F5E	5B8B	E55D	C300	0000	À-üÿèà_^ [<ã]Ã...
FFF	0400	0000	2E69	6E69	0000	0000	ÿÿÿÿ.....ini....
FFF	0B00	0000	656E	676C	6973	682E	ÿÿÿÿ....english.
900	FFFF	FFFF	1400	0000	4C61	6E67	ini.ÿÿÿÿ....Lang
765	2066	696C	6573	7C2A	2E69	6E69	uage files *.ini
000	FFFF	FFFF	3700	0000	506C	6561ÿÿÿÿ7...Plea
C20	6368	6F6F	7365	2074	6865	2065	se, choose the e
C69	7368	2E69	6E69	206F	7220	616E	nglish.ini or an

EFF	8B45	FC8B	80E4	0100	008B	4820	°°pÿ<Eü<€ä...<H
444	008B	45F0	E8E7	B3FE	FF33	C05A	°..D.<Eðèç°pÿ3ÀZ
489	1068	BA02	4400	8B45	FOE8	5629	YYd%.h°.D.<EðèV)
3E9	5C2E	FCFF	EBF0	33C0	5A59	5964	üÿÃé\ .üÿèð3ÀZYd
8D7	0244	008B	45F4	E839	29FC	FFC3	%.h×.D.<Eðè9)üÿÃ
EFC	FFEB	F080	7DFB	0075	246A	1068	é?.üÿèð€)û.u\$ÿ.h
400	6828	0444	008B	45FC	E807	65FD	.D.h(.D.<Eüè.eÿ
85D	4EFC	FFA1	3056	4400	E8E7	37FE	ÿPè]Nüÿ;OVD.èç7p
5FC	8B80	E401	0000	E84D	6BFF	FF8B	ÿ<Eü<€ä...èMkÿÿ<
B80	9002	0000	E8E7	74FE	FF8B	55FC	Eü<€Ð...èçtpÿ<Uü
401	0000	8942	0C33	C05A	5959	6489	<'ä...%B.3ÀZYd%
603	4400	8D45	E4E8	F631	FCFF	8D45	.hV.D.ÐEäèð1üÿÐE
E31	FCFF	8D45	ECE8	E631	FCFF	C3E9	èèí1üÿÐEièæ1üÿÃé
CFF	EBE0	5F5E	5B8B	E55D	C300	0000	À-üÿèà_^ [<ã]Ã...
FFF	0400	0000	2E69	6E69	0000	0000	ÿÿÿÿ.....ini....
FFF	0B00	0000	7370	616E	6973	682E	ÿÿÿÿ....spanish.
900	FFFF	FFFF	1400	0000	4C61	6E67	ini.ÿÿÿÿ....Lang
765	2066	696C	6573	7C2A	2E69	6E69	uage files *.ini
000	FFFF	FFFF	3700	0000	506C	6561ÿÿÿÿ7...Plea
C20	6368	6F6F	7365	2074	6865	2065	se, choose the e
C69	7368	2E69	6E69	206F	7220	616E	nglish.ini or an

Y como tenemos la suerte de que las dos palabras tienen el mismo largo pues mejor aun.

Ya hay una cosa menos. Ahora a por lo de days, pero eso lo haré en el Olly para que me sirva de ayuda para encontrar la cifra que aparece también ya que days si aparece en las strings.

Hacemos una búsqueda de las strings en el exe:



Y buscamos la string "days" y le ponemos un BP con F2:

0043B741	MOV EAX, 43809C	ASCII 04,"TLZB"
0043B786	MOV EAX, 43CDB8	ASCII "12"
0043B8CF	MOV EAX, 43CDDC	ASCII "04"
0043B961	MOV EAX, 43CDB8	ASCII "12"
0043B99D	MOV EAX, 43CDD0	ASCII "05"
0043BA62	MOV EDX, 43CDDC	ASCII ".reloc"
0043BA80	CMP DWORD PTR DS:[EAX+34], 400000	ASCII "MZP"
0043C1FE	MOV EAX, 43CDEC	ASCII "06"
0043CB69	MOV EDX, 43CE00	ASCII ".bak"
0043CFF0	MOV EAX, 44390C	UNICODE "REGISTRYTYPELIB"
0043D23F	MOV EDX, 43D61C	ASCII "kernel32.dll"
0043D291	MOV EDX, 43D634	ASCII "\$"
0043DE2D	MOV EAX, 43DE80	ASCII "HTTP://"
0043DEC8	PUSH 43E090	ASCII "InternetReader"
0043DF01	PUSH 43E0A0	ASCII "HTTP/1.0"
0043DF0F	PUSH 43E0AC	ASCII "GET"
0043E233	PUSH 43E250	ASCII "\\VarFileInfo\Translation"
0043E338	MOV EDX, 43E3F4	ASCII "\\StringFileInfo"
0043E41F	MOV EDX, 43E458	ASCII "ProductVersion"
0043E4AE	MOV EAX, 43E4C8	ASCII "%d.%d.%d.%d"
0043E503	MOV EDX, 43E5D4	ASCII "ASPack - "
0043EFBA	MOV EDX, 43F4E8	ASCII "Jg"
0043EF51	MOV EDX, 43F4F4	ASCII " days"
0043F015	MOV ECX, 43F504	ASCII "Aspack.hlp"
0043F064	MOV EAX, 43E0E4	ASCII 0C,"TVersionInfo"
0043F0DE	MOV EDX, 43F518	ASCII "SOFTWARE\ASPack"
0043F0EE	MOV EDX, 43F530	ASCII "VersionNum"
0043F12F	MOV EDX, 43F530	ASCII "VersionNum"
0043F26B	MOV EDX, 43F544	ASCII "Software\ASPack\Options"
0043F27B	MOV EDX, 43F564	ASCII "Lang_LanguageFile"
0043F2FB	MOV EDX, 43F580	ASCII "*.ini"
0043F38E	MOV EAX, 43F590	ASCII ".ini"
0043F534	IMUL EBP, DWORD PTR DS:[EDI+6E], 6D754E	UNICODE "va"
0043F60A	MOV EDX, 43F6A0	ASCII "08"
0043F61E	MOV ECX, 43F6A4	ASCII "Error"
0043F6F0	MOV ECX, 43F7C8	ASCII "aspack.ini"
0043F722	MOV EDX, 43F7DC	ASCII "Item"
0043F747	MOV EDX, 43F7EC	ASCII "PopupMenuHistory"
0043F818	MOV ECX, 43F8F0	ASCII "aspack.ini"
0043F866	MOV EDX, 43F904	ASCII "Item"
0043F88B	MOV EDX, 43F914	ASCII "PopupMenuHistory"
0043F967	MOV EDX, 43FA60	ASCII "10"
0043F981	MOV ECX, 43FA64	ASCII "Error"
0043F998	MOV ECX, 43FA74	ASCII ".bak"
0043FB49	MOV EDX, 43FEAC	ASCII "exefile\shell\"
0043FB80	MOV EDX, 43FEC4	ASCII "\Command"
0043FBD4	MOV EDX, 43FED8	ASCII "%1"
0043FC16	MOV EDX, 43FEF4	ASCII "dllfile\shell\"

Damos a F9 y vemos que para aquí:

0043EFB0	8BC3	MOV EAX, EBX	
0043EFB2	E8 8541FDFD	CALL 0041313C	dumpeado.0041313C
0043EFB7	8D45 D8	LEA EAX, DWORD PTR SS:[EBP-28]	
0043EFBA	BA E8F44300	MOV EDX, 43F4E8	ASCII "Jg"
0043EFBF	E8 EC46FCFF	CALL 004036B0	dumpeado.004036B0
0043EFC4	8D45 D8	LEA EAX, DWORD PTR SS:[EBP-28]	
0043EFC7	50	PUSH EAX	
0043EFC8	8D55 D4	LEA EDX, DWORD PTR SS:[EBP-2C]	
0043EFCB	A1 00494400	MOV EAX, DWORD PTR DS:[444900]	
0043EFD0	E8 4F6DFCFF	CALL 00405D24	dumpeado.00405D24
0043EFD5	8B55 D4	MOV EDX, DWORD PTR SS:[EBP-2C]	
0043EFD8	58	POP EAX	
0043EFD9	E8 D246FCFF	CALL 004036B0	dumpeado.004036B0
0043EFDE	8D45 D8	LEA EAX, DWORD PTR SS:[EBP-28]	
0043EFE1	BA F4F44300	MOV EDX, 43F4F4	ASCII " days"
0043EFE6	E8 C546FCFF	CALL 004036B0	dumpeado.004036B0
0043EFEB	8B55 D8	MOV EDX, DWORD PTR SS:[EBP-28]	
0043EFEE	8BC3	MOV EAX, EBX	
0043EFF0	E8 7741FDFD	CALL 0041316C	dumpeado.0041316C
0043EFF5	8D55 D0	LEA EDX, DWORD PTR SS:[EBP-30]	
0043EFF8	33C0	XOR EAX, EAX	
0043EFFA	E8 AD38FCFF	CALL 004028AC	dumpeado.004028AC
0043EFFD	8B45 D0	MOV EAX, DWORD PTR SS:[EBP-30]	
0043F002	8D55 D4	LEA EDX, DWORD PTR SS:[EBP-2C]	
0043F005	E8 6E71FCFF	CALL 00406178	dumpeado.00406178
0043F00A	8B55 D4	MOV EDX, DWORD PTR SS:[EBP-2C]	
0043F00D	A1 30564400	MOV EAX, DWORD PTR DS:[445630]	
0043F012	83C0 30	ADD EAX, 30	
0043F015	B9 04F54300	MOV ECX, 43F504	ASCII "Aspack.hlp"
0043F01A	E8 D546FCFF	CALL 004036F4	dumpeado.004036F4
0043F01F	A1 30564400	MOV EAX, DWORD PTR DS:[445630]	
0043F024	C740 5C 881300	MOV DWORD PTR DS:[EAX+5C], 1388	
0043F02B	8B45 FC	MOV EAX, DWORD PTR SS:[EBP-4]	
0043F02E	C680 0C030000	MOV BYTE PTR DS:[EAX+30C], 0	
0043F035	8B45 FC	MOV EAX, DWORD PTR SS:[EBP-4]	
0043F038	8B80 E4020000	MOV EAX, DWORD PTR DS:[EAX+2E4]	
0043F03E	C640 20 00	MOV BYTE PTR DS:[EAX+20], 0	
0043F042	33D2	XOR EDX, EDX	
0043F044	8B45 FC	MOV EAX, DWORD PTR SS:[EBP-4]	
0043F047	8B80 7C020000	MOV EAX, DWORD PTR DS:[EAX+27C]	
0043F04D	E8 8240FDFD	CALL 004130D4	dumpeado.004130D4
0043F052	8D55 D4	LEA EDX, DWORD PTR SS:[EBP-2C]	
0043F055	A1 30564400	MOV EAX, DWORD PTR DS:[445630]	
0043F05A	F8 014CFDFD	CALL 00423030	dumpeado.00423030

Vamos traceando a ver si aparecen los días que muestra y:

0043FE05	8B55 D4	MOV EDX, DWORD PTR SS:[EBP-2C]			
0043FE08	58	POP EAX			
0043FE09	E8 0246FCFF	CALL 004036B0	dumpeado.004036B0		
0043FE0E	8D45 D8	LEA EAX, DWORD PTR SS:[EBP-28]			
0043FE11	BA F4F44300	MOV EDX, 43F4F4	ASCII " days"		
0043FE16	E8 C546FCFF	CALL 004036B0	dumpeado.004036B0		
0043FE1B	8B55 D8	MOV EDX, DWORD PTR SS:[EBP-28]			
0043FE1E	8BC3	MOV EAX, EBX			
0043FE20	E8 7741FDFF	CALL 0041316C	dumpeado.0041316C		
0043FE23	8D55 D0	LEA EDX, DWORD PTR SS:[EBP-30]			
0043FE26	33C0	XOR EAX, EAX			
0043FE29	E8 AD38FCFF	CALL 004028AC	dumpeado.004028AC		
0043FE2C	8B45 D0	MOV EAX, DWORD PTR SS:[EBP-30]			
0043FE2F	8D55 D4	LEA EDX, DWORD PTR SS:[EBP-2C]			
0043FE32	E8 6E71FCFF	CALL 00406178	dumpeado.00406178		
0043FE35	8B55 D4	MOV EDX, DWORD PTR SS:[EBP-2C]			
0043FE38	A1 30564400	MOV EAX, DWORD PTR DS:[445630]			
0043FE3B	83C0 30	ADD EAX, 30			
0043FE3E	B9 04F54300	MOV ECX, 43F504	ASCII "Aspack.hlp"		
0043FE41	E8 D546FCFF	CALL 004036F4	dumpeado.004036F4		
0043FE44	A1 30564400	MOV EAX, DWORD PTR DS:[445630]			
0043FE47	C740 5C 881300	MOV DWORD PTR DS:[EAX+5C], 1388			
0043FE4A	8B45 FC	MOV EAX, DWORD PTR SS:[EBP-4]			
0043FE4D	C680 0C030000	MOV BYTE PTR DS:[EAX+30C], 0			
0043FE50	8B45 FC	MOV EAX, DWORD PTR SS:[EBP-4]			
0043FE53	8B80 E4020000	MOV EAX, DWORD PTR DS:[EAX+2E41]			
0043FE56	C640 20 00	MOV BYTE PTR DS:[EAX+20], 0			
0043FE59	3BD2	XOR EDX, EDX			
0043FE5C	8B45 FC	MOV EAX, DWORD PTR SS:[EBP-4]			

Stack SS:[0012FE28]=00AA600C, (ASCII "UNREGISTERED\00 days")
 EDX=00AA601B, (ASCII " days")

Address	Hex dump	ASCII
00443000	02 00 3E C0 00 8D 40 00	0.1*.i@.
00443003	94 DE 42 00 A3 20 40 00	0iB.2 @.
00443006	30 22 40 00 A4 25 40 00	0m@.n%@.
00443009	32 1F 8B C0 32 13 8B C0	2%i+2!!!L
0044300C	52 75 6E 74 69 6D 65 20	Runtime
0044300F	65 72 72 6F 72 20 20 20	error
00443012	20 20 61 74 20 30 30 20	at 000
00443015	30 30 30 30 30 00 45 72	00000.Er
00443018	72 6F 72 00 20 20 20 20	ror.
0044301B	20 20 20 20 20 20 20 20	
0044301E	20 20 20 20 20 20 20 20	
00443021	20 20 20 20 20 20 20 20	
00443024	20 20 20 20 20 20 20 20	
00443027	20 20 20 20 20 20 20 20	
0044302A	20 20 20 20 20 20 20 20	
0044302D	20 20 20 20 20 20 20 20	
00443030	20 20 20 20 20 20 20 20	
00443033	20 20 20 20 20 20 20 20	
00443036	20 20 20 20 20 20 20 20	
00443039	20 20 20 20 20 20 20 20	
0044303C	20 20 20 20 20 20 20 20	
0044303F	20 20 20 20 20 20 20 20	
00443042	20 20 20 20 20 20 20 20	
00443045	20 20 20 20 20 20 20 20	
00443048	20 20 20 20 20 20 20 20	
0044304B	20 20 20 20 20 20 20 20	
0044304E	20 20 20 20 20 20 20 20	
00443051	20 20 20 20 20 20 20 20	
00443054	20 20 20 20 20 20 20 20	
00443057	20 20 20 20 20 20 20 20	
0044305A	20 20 20 20 20 20 20 20	
0044305D	20 20 20 20 20 20 20 20	
00443060	20 20 20 20 20 20 20 20	
00443063	20 20 20 20 20 20 20 20	
00443066	20 20 20 20 20 20 20 20	
00443069	20 20 20 20 20 20 20 20	
0044306C	20 20 20 20 20 20 20 20	
0044306F	20 20 20 20 20 20 20 20	
00443072	20 20 20 20 20 20 20 20	
00443075	20 20 20 20 20 20 20 20	
00443078	20 20 20 20 20 20 20 20	
0044307B	20 20 20 20 20 20 20 20	
0044307E	20 20 20 20 20 20 20 20	
00443081	20 20 20 20 20 20 20 20	
00443084	20 20 20 20 20 20 20 20	
00443087	20 20 20 20 20 20 20 20	
0044308A	20 20 20 20 20 20 20 20	
0044308D	20 20 20 20 20 20 20 20	
00443090	20 20 20 20 20 20 20 20	
00443093	20 20 20 20 20 20 20 20	
00443096	20 20 20 20 20 20 20 20	
00443099	20 20 20 20 20 20 20 20	
0044309C	20 20 20 20 20 20 20 20	
0044309F	20 20 20 20 20 20 20 20	
004430A2	20 20 20 20 20 20 20 20	
004430A5	20 20 20 20 20 20 20 20	
004430A8	20 20 20 20 20 20 20 20	
004430AB	20 20 20 20 20 20 20 20	
004430AE	20 20 20 20 20 20 20 20	
004430B1	20 20 20 20 20 20 20 20	
004430B4	20 20 20 20 20 20 20 20	
004430B7	20 20 20 20 20 20 20 20	
004430BA	20 20 20 20 20 20 20 20	
004430BD	20 20 20 20 20 20 20 20	
004430C0	20 20 20 20 20 20 20 20	
004430C3	20 20 20 20 20 20 20 20	
004430C6	20 20 20 20 20 20 20 20	
004430C9	20 20 20 20 20 20 20 20	
004430CC	20 20 20 20 20 20 20 20	
004430CF	20 20 20 20 20 20 20 20	
004430D2	20 20 20 20 20 20 20 20	
004430D5	20 20 20 20 20 20 20 20	
004430D8	20 20 20 20 20 20 20 20	
004430DB	20 20 20 20 20 20 20 20	
004430DE	20 20 20 20 20 20 20 20	
004430E1	20 20 20 20 20 20 20 20	
004430E4	20 20 20 20 20 20 20 20	
004430E7	20 20 20 20 20 20 20 20	
004430EA	20 20 20 20 20 20 20 20	
004430ED	20 20 20 20 20 20 20 20	
004430F0	20 20 20 20 20 20 20 20	
004430F3	20 20 20 20 20 20 20 20	
004430F6	20 20 20 20 20 20 20 20	
004430F9	20 20 20 20 20 20 20 20	
004430FC	20 20 20 20 20 20 20 20	
004430FF	20 20 20 20 20 20 20 20	

Bueno pues estuve haciendo pruebas y la forma que me funciono fue nopear los dos CALLs que están justo debajo del BP y con eso ya no salen los días.

0043EFC4	8045 D8	LEA EAX, DWORD PTR SS:[EBP-28]	dumpeado.00400000
0043EFC7	58	PUSH EAX	
0043EFC8	8055 D4	LEA EDX, DWORD PTR SS:[EBP-2C]	
0043EFCB	A1 00494400	MOV EAX, DWORD PTR DS:[444900]	
0043EFD0	E8 4F6DFCFF	CALL 00405D24	dumpeado.00405D24
0043EFD5	8B55 D4	MOV EDX, DWORD PTR SS:[EBP-2C]	
0043EFD8	58	POP EAX	
0043EFD9	E8 D246FCFF	CALL 004036B0	dumpeado.004036B0
0043EFD0E	8045 D8	LEA EAX, DWORD PTR SS:[EBP-28]	
0043EFE1	BA F4F44300	MOV EDX, 43F4F4	ASCII " days"
0043EFE6	90	NOP	
0043EFE7	90	NOP	
0043EFE8	90	NOP	
0043EFE9	90	NOP	
0043EFEA	90	NOP	
0043EFEB	8B55 D8	MOV EDX, DWORD PTR SS:[EBP-28]	
0043EFEE	8BC3	MOV EAX, EBX	
0043EFF0	90	NOP	
0043EFF1	90	NOP	
0043EFF2	90	NOP	
0043EFF3	90	NOP	
0043EFF4	90	NOP	
0043EFF5	8055 D0	LEA EDX, DWORD PTR SS:[EBP-30]	
0043EFF8	33C0	XOR EAX, EAX	
0043EFFF	E8 AD38FCFF	CALL 004028AC	dumpeado.004028AC
0043FFF0	8B45 D0	MOV EAX, DWORD PTR SS:[EBP-30]	
0043FFF2	8055 D4	LEA EDX, DWORD PTR SS:[EBP-2C]	
0043FFF5	E8 6E71FCFF	CALL 00406178	dumpeado.00406178
0043FFF8	8B55 D4	MOV EDX, DWORD PTR SS:[EBP-2C]	
0043FFFA	A1 30564400	MOV EAX, DWORD PTR DS:[445630]	
0043FFFB	83C0 30	ADD EAX, 30	
0043FFFD	B9 04F54300	MOV ECX, 43F504	ASCII "Aspack.hlp"
0043FFFE	E8 D546FCFF	CALL 004036F4	dumpeado.004036F4
0043FFF0	A1 30564400	MOV EAX, DWORD PTR DS:[445630]	
0043FFF2	C740 5C 8B1300	MOV DWORD PTR DS:[EAX+5C], 1388	
0043FFF4	8B45 FC	MOV EAX, DWORD PTR SS:[EBP-4]	
0043FFF6	C600 0C030000	MOV BYTE PTR DS:[EAX+30C], 0	
0043FFF8	8B45 FC	MOV EAX, DWORD PTR SS:[EBP-4]	
0043FFFA	8B80 E4020000	MOV EAX, DWORD PTR DS:[EAX+2E4]	
0043FFFB	C640 20 00	MOV BYTE PTR DS:[EAX+20], 0	
0043FFFD	33D2	XOR EDX, EDX	
0043FFF0	8B45 FC	MOV EAX, DWORD PTR SS:[EBP-4]	
0043FFF2	8B80 7C020000	MOV EAX, DWORD PTR DS:[EAX+27C]	
0043FFF4	E8 8240FDFF	CALL 004130D4	dumpeado.004130D4

Bueno, yo le hice algún que otro cambio más jejeje



Si se pulsa en donde pone **** Crackeado por Aguml **** te lleva a google

Bueno, lo hice para que lo entienda gente tan torpe como yo que empiecen desde 0 ya que la gran mayoría de la comunidad sabe manejarse en esto y no necesitan tantas imágenes.

Pos eso, espero que disfrutéis los Newbies tanto como yo cuando lo destripe.