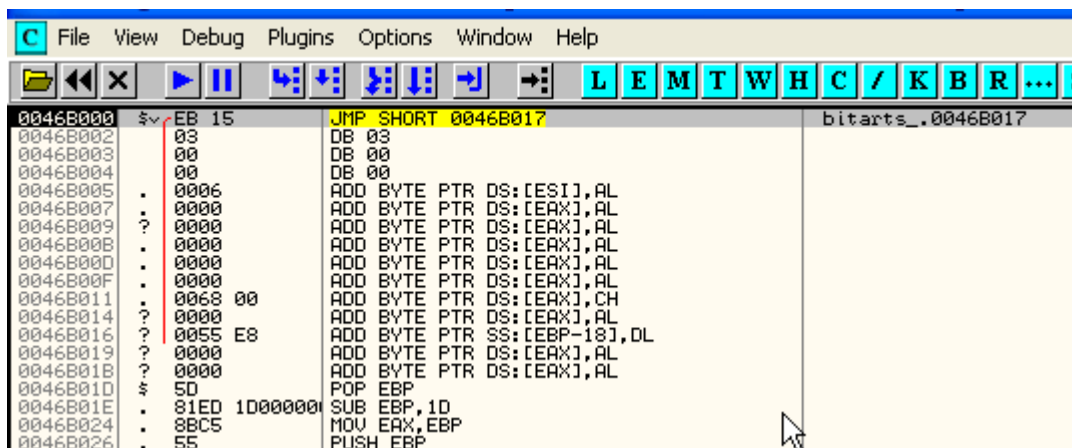


INTRODUCCION AL CRACKING EN OLLYDBG PARTE 36

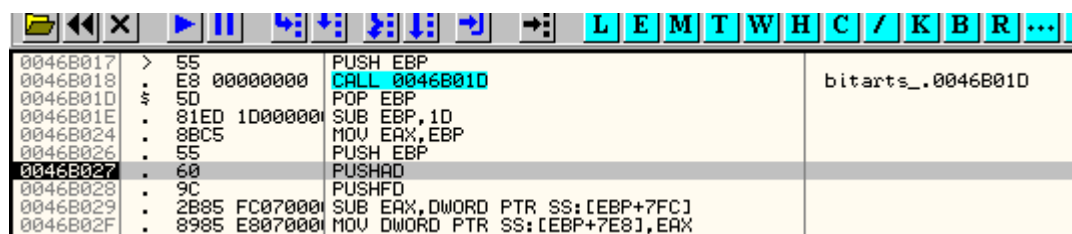
Seguimos aumentando muy suavemente la dificultad en los packers que vamos viendo, en esta entrega veremos dos desempacados, el crunch o bitarts 5.0 y el telock 0.98, que nos servira para introducirlos en el tema de las entradas de la IAT redireccionadas, ambos unpackmes estan adjuntos a este tutorial, por lo que no hay problemas para obtenerlos.

Empezaremos con el mas sencillo con el BITARTS lo arrancamos en OLLYDBG.



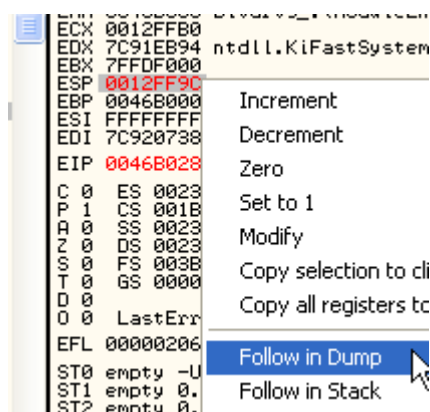
```
File View Debug Plugins Options Window Help
[Icons] [L] [E] [M] [T] [W] [H] [C] [/] [K] [B] [R] [...] [S]
0046B000  $v EB 15      JMP SHORT 0046B017      bitarts_.0046B017
0046B002  03           DB 03
0046B003  00           DB 00
0046B004  00           DB 00
0046B005  . 0006       ADD BYTE PTR DS:[ESI],AL
0046B007  . 0000       ADD BYTE PTR DS:[EAX],AL
0046B009  ? 0000       ADD BYTE PTR DS:[EAX],AL
0046B00B  . 0000       ADD BYTE PTR DS:[EAX],AL
0046B00D  . 0000       ADD BYTE PTR DS:[EAX],AL
0046B00F  . 0000       ADD BYTE PTR DS:[EAX],AL
0046B011  . 0068 00    ADD BYTE PTR DS:[EAX],CH
0046B014  ? 0000       ADD BYTE PTR DS:[EAX],AL
0046B016  . 0055 E8    ADD BYTE PTR SS:[EBP-18],DL
0046B019  ? 0000       ADD BYTE PTR DS:[EAX],AL
0046B01B  ? 0000       ADD BYTE PTR DS:[EAX],AL
0046B01D  $ 5D         POP EBP
0046B01E  . 81ED 1D0000 SUB EBP,1D
0046B024  . 8BC5       MOV EAX,EBP
0046B026  . 55         PUSH EBP
```

Vemos que no hay un PUSHAD inicial, asi que traceemos un poco con f7.



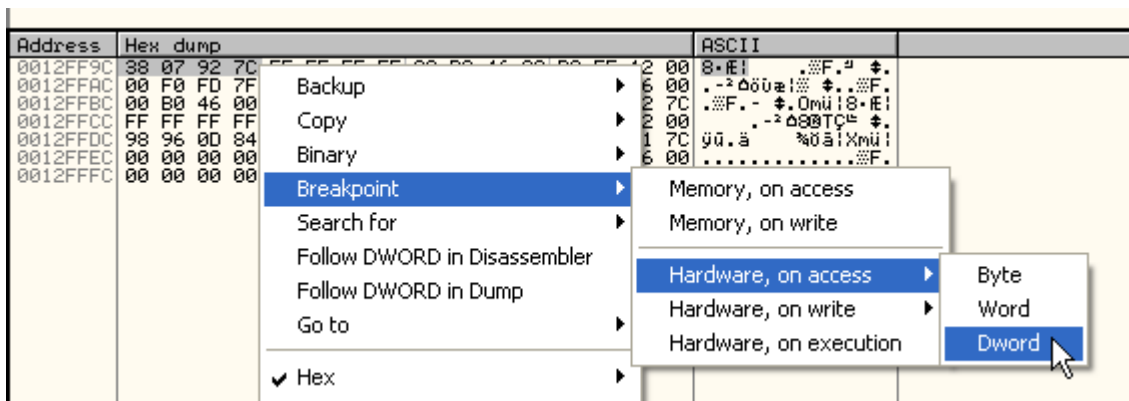
```
[Icons] [L] [E] [M] [T] [W] [H] [C] [/] [K] [B] [R] [...] [S]
0046B017  > 55         PUSH EBP
0046B018  . E8 00000000 CAL 0046B01D
0046B01D  $ 5D         POP EBP
0046B01E  . 81ED 1D0000 SUB EBP,1D
0046B024  . 8BC5       MOV EAX,EBP
0046B026  . 55         PUSH EBP
0046B027  . 60         PUSHAD
0046B028  . 9C         PUSHFD
0046B029  . 2B85 FC0700 SUB EAX,DWORD PTR SS:[EBP+7FC]
0046B02F  . 8985 E80700 MOV DWORD PTR SS:[EBP+7E8],EAX
0046B035  . 55         PUSH EBP
```

Vemos que ahi llega a un PUSHAD asi que pasemoslo con f7 y hagamos ESP- FOLLOW IN DUMP.

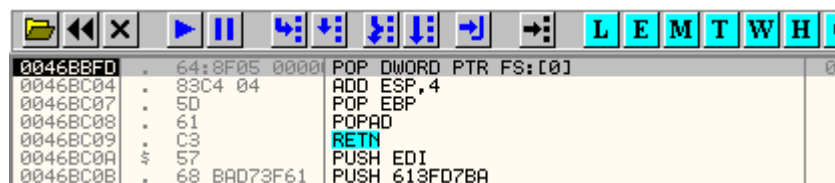


```
ECX 0012FFB0
EDX 7C91EB94 ntdll.KiFastSystem
EBX 7FFDF000
ESP 0012FF9C
EBP 0046B000 Increment
ESI FFFFFFFF Decrement
EDI 7C920738
EIP 0046B028 Zero
C 0 ES 0023 Set to 1
P 1 CS 001B
A 0 SS 0023 Modify
Z 0 DS 0023
S 0 FS 003B Copy selection to cli
T 0 GS 0000 Copy all registers to
D 0
O 0 LastErr
EFL 00000206
ST0 empty -U
ST1 empty 0.
ST2 empty 0.
```

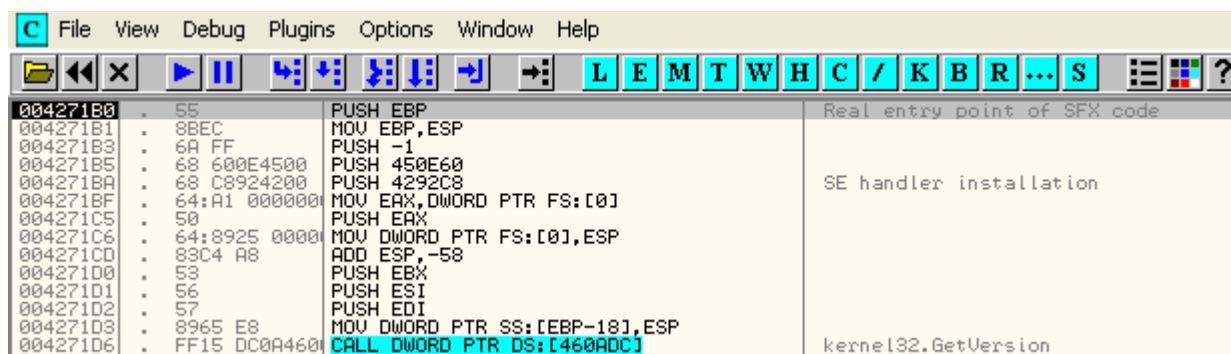
Y pongo un Hardware Breakpoint on access en los primeros bytes.



Ahora apreto f9 y despues que pasa alguna que otra excepcion, para por el Hardware Breakpoint.



Traceo unas lineas y llego al OEP.



Al OEP se puede llegar tanto usando el buscador de OEPs de OLLYDBG, el OLLYDBG parcheado para OEPs o casi cualquiera de los metodos que vimos en las partes anteriores de la introduccion.

Como vemos dicha direccion esta en la primera seccion despues del header

003E0000	00004000				Priv	RW	RW	
003F0000	00002000				Map	R	R	
00400000	00001000	bitarts_		PE header	Imag	R	RWE	
00401000	0000A000	bitarts_	.text	code	Imag	R	RWE	
0044B000	0000C000	bitarts_	.rdata		Imag	R	RWE	
00457000	00009000	bitarts_	.data	data	Imag	R	RWE	
00460000	00003000	bitarts_	.idata		Imag	R	RWE	
00463000	00008000	bitarts_	.rsrc	resources	Imag	R	RWE	
0046B000	00008000	bitarts_	.edata	SFX, imports	Imag	R	RWE	
004C0000	00009000				Map	R E	R E	

Por lo cual la cosa parece bien ordenada, incluso esta marcada como seccion CODE asi que no habra problemas

Address	Disassembly	Comment
004271B0	PUSH EBP	Real entry point of SFX cc
004271B1	MOV EBP, ESP	
004271B3	PUSH -1	
004271B5	PUSH 450E60	
004271B8	PUSH 4292C8	SE handler installation
004271BF	MOV EAX, DWORD PTR FS:[0]	
004271C5	PUSH EAX	
004271C6	MOV DWORD PTR FS:[0], ESP	
004271CD	ADD ESP, -58	
004271D0	PUSH EBX	
004271D1	PUSH ESI	
004271D2	PUSH EDI	
004271D3	MOV DWORD PTR SS:[EBP-18], ESP	
004271D6	CALL DWORD PTR DS:[460ADC]	kernel32.GetVersion
004271DC	XOR EDX, EDX	
004271DE	MOV DL, AH	
004271E0	MOV DWORD PTR DS:[45E634], EDX	
004271F6	MOV ECX, ECX	

Vemos que en este caso la primera API que llama es GetVersion por lo cual tambien poner un BP en dicha api y cuando para en ella, llamada desde primera seccion, al retornar de la misma, nos hubiera dejado, en la zona del OEP.

Bueno el OEP no tiene misterios, miremos un poco la IAT, por supuesto alli a la vista tenemos una llamada directa a una api, por lo cual es facil ver que el CALL INDIRECTO, toma el valor correcto en mi maquina de la api GetVersion, para dirigirse a la misma.

Por lo tanto 460ADC es una entrada de la IAT, la correspondiente a GetVersion, vayamos en el dump a verificar la IAT.

00427102		. 57		PUSH ESI		MOV ESI, EBP		JMP 00427103		00427103		. 58		JMP 00427104		00427104		. 59		JMP 00427105		00427105		. 60		JMP 00427106		00427106		. 61		JMP 00427107		00427107		. 62		JMP 00427108		00427108		. 63		JMP 00427109		00427109		. 64		JMP 0042710A		0042710A		. 65		JMP 0042710B		0042710B		. 66		JMP 0042710C		0042710C		. 67		JMP 0042710D		0042710D		. 68		JMP 0042710E		0042710E		. 69		JMP 0042710F		0042710F		. 70		JMP 00427110		00427110		. 71		JMP 00427111		00427111		. 72		JMP 00427112		00427112		. 73		JMP 00427113		00427113		. 74		JMP 00427114		00427114		. 75		JMP 00427115		00427115		. 76		JMP 00427116		00427116		. 77		JMP 00427117		00427117		. 78		JMP 00427118		00427118		. 79		JMP 00427119		00427119		. 80		JMP 0042711A		0042711A		. 81		JMP 0042711B		0042711B		. 82		JMP 0042711C		0042711C		. 83		JMP 0042711D		0042711D		. 84		JMP 0042711E		0042711E		. 85		JMP 0042711F		0042711F		. 86		JMP 00427120		00427120		. 87		JMP 00427121		00427121		. 88		JMP 00427122		00427122		. 89		JMP 00427123		00427123		. 90		JMP 00427124		00427124		. 91		JMP 00427125		00427125		. 92		JMP 00427126		00427126		. 93		JMP 00427127		00427127		. 94		JMP 00427128		00427128		. 95		JMP 00427129		00427129		. 96		JMP 0042712A		0042712A		. 97		JMP 0042712B		0042712B		. 98		JMP 0042712C		0042712C		. 99		JMP 0042712D		0042712D		. 100		JMP 0042712E		0042712E		. 101		JMP 0042712F		0042712F		. 102		JMP 00427130		00427130		. 103		JMP 00427131		00427131		. 104		JMP 00427132		00427132		. 105		JMP 00427133		00427133		. 106		JMP 00427134		00427134		. 107		JMP 00427135		00427135		. 108		JMP 00427136		00427136		. 109		JMP 00427137		00427137		. 110		JMP 00427138		00427138		. 111		JMP 00427139		00427139		. 112		JMP 0042713A		0042713A		. 113		JMP 0042713B		0042713B		. 114		JMP 0042713C		0042713C		. 115		JMP 0042713D		0042713D		. 116		JMP 0042713E		0042713E		. 117		JMP 0042713F		0042713F		. 118		JMP 00427140		00427140		. 119		JMP 00427141		00427141		. 120		JMP 00427142		00427142		. 121		JMP 00427143		00427143		. 122		JMP 00427144		00427144		. 123		JMP 00427145		00427145		. 124		JMP 00427146		00427146		. 125		JMP 00427147		00427147		. 126		JMP 00427148		00427148		. 127		JMP 00427149		00427149		. 128		JMP 0042714A		0042714A		. 129		JMP 0042714B		0042714B		. 130		JMP 0042714C		0042714C		. 131		JMP 0042714D		0042714D		. 132		JMP 0042714E		0042714E		. 133		JMP 0042714F		0042714F		. 134		JMP 00427150		00427150		. 135		JMP 00427151		00427151		. 136		JMP 00427152		00427152		. 137		JMP 00427153		00427153		. 138		JMP 00427154		00427154		. 139		JMP 00427155		00427155		. 140		JMP 00427156		00427156		. 141		JMP 00427157		00427157		. 142		JMP 00427158		00427158		. 143		JMP 00427159		00427159		. 144		JMP 0042715A		0042715A		. 145		JMP 0042715B		0042715B		. 146		JMP 0042715C		0042715C		. 147		JMP 0042715D		0042715D		. 148		JMP 0042715E		0042715E		. 149		JMP 0042715F		0042715F		. 150		JMP 00427160		00427160		. 151		JMP 00427161		00427161		. 152		JMP 00427162		00427162		. 153		JMP 00427163		00427163		. 154		JMP 00427164		00427164		. 155		JMP 00427165		00427165		. 156		JMP 00427166		00427166		. 157		JMP 00427167		00427167		. 158		JMP 00427168		00427168		. 159		JMP 00427169		00427169		. 160		JMP 0042716A		0042716A		. 161		JMP 0042716B		0042716B		. 162		JMP 0042716C		0042716C		. 163		JMP 0042716D		0042716D		. 164		JMP 0042716E		0042716E		. 165		JMP 0042716F		0042716F		. 166		JMP 00427170		00427170		. 167		JMP 00427171		00427171		. 168		JMP 00427172		00427172		. 169		JMP 00427173		00427173		. 170		JMP 00427174		00427174		. 171		JMP 00427175		00427175		. 172		JMP 00427176		00427176		. 173		JMP 00427177		00427177		. 174		JMP 00427178		00427178		. 175		JMP 00427179		00427179		. 176		JMP 0042717A		0042717A		. 177		JMP 0042717B		0042717B		. 178		JMP 0042717C		0042717C		. 179		JMP 0042717D		0042717D		. 180		JMP 0042717E		0042717E		. 181		JMP 0042717F		0042717F		. 182		JMP 00427180		00427180		. 183		JMP 00427181		00427181		. 184		JMP 00427182		00427182		. 185		JMP 00427183		00427183		. 186		JMP 00427184		00427184		. 187		JMP 00427185		00427185		. 188		JMP 00427186		00427186		. 189		JMP 00427187		00427187		. 190		JMP 00427188		00427188		. 191		JMP 00427189		00427189		. 192		JMP 0042718A		0042718A		. 193		JMP 0042718B		0042718B		. 194		JMP 0042718C		0042718C		. 195		JMP 0042718D		0042718D		. 196		JMP 0042718E		0042718E		. 197		JMP 0042718F		0042718F		. 198		JMP 00427190		00427190		. 199		JMP 00427191		00427191		. 200		JMP 00427192		00427192		. 201		JMP 00427193		00427193		. 202		JMP 00427194		00427194		. 203		JMP 00427195		00427195		. 204		JMP 00427196		00427196		. 205		JMP 00427197		00427197		. 206		JMP 00427198		00427198		. 207		JMP 00427199		00427199		. 208		JMP 0042719A		0042719A		. 209		JMP 0042719B		0042719B		. 210		JMP 0042719C		0042719C		. 211		JMP 0042719D		0042719D		. 212		JMP 0042719E		0042719E		. 213		JMP 0042719F		0042719F		. 214		JMP 004271A0		004271A0		. 215		JMP 004271A1		004271A1		. 216		JMP 004271A2		004271A2		. 217		JMP 004271A3		004271A3		. 218		JMP 004271A4		004271A4		. 219		JMP 004271A5		004271A5		. 220		JMP 004271A6		004271A6		. 221		JMP 004271A7		004271A7		. 222		JMP 004271A8		004271A8		. 223		JMP 004271A9		004271A9		. 224		JMP 004271AA		004271AA		. 225		JMP 004271AB		004271AB		. 226		JMP 004271AC		004271AC		. 227		JMP 004271AD		004271AD		. 228		JMP 004271AE		004271AE		. 229		JMP 004271AF		004271AF		. 230		JMP 004271B0		004271B0		. 231		JMP 004271B1		004271B1		. 232		JMP 004271B2		004271B2		. 233		JMP 004271B3		004271B3		. 234		JMP 004271B4		004271B4		. 235		JMP 004271B5		004271B5		. 236		JMP 004271B6		004271B6		. 237		JMP 004271B7		004271B7		. 238		JMP 004271B8		004271B8		. 239		JMP 004271B9		004271B9		. 240		JMP 004271BA		004271BA		. 241		JMP 004271BB		004271BB		. 242		JMP 004271BC		004271BC		. 243		JMP 004271BD		004271BD		. 244		JMP 004271BE		004271BE		. 245		JMP 004271BF		004271BF		. 246		JMP 004271C0		004271C0		. 247		JMP 004271C1		004271C1		. 248		JMP 004271C2		004271C2		. 249		JMP 004271C3		004271C3		. 250		JMP 004271C4		004271C4		. 251		JMP 004271C5		004271C5		. 252		JMP 004271C6		004271C6		. 253		JMP 004271C7		004271C7		. 254		JMP 004271C8		004271C8		. 255		JMP 004271C9		004271C9		. 256		JMP 004271CA		004271CA		. 257		JMP 004271CB		004271CB		. 258		JMP 004271CC		004271CC		. 259		JMP 004271CD		004271CD		. 260		JMP 004271CE		004271CE		. 261		JMP 004271CF		004271CF		. 262		JMP 004271D0		004271D0		. 263		JMP 004271D1		004271D1		. 264		JMP 004271D2		004271D2		. 265		JMP 004271D3		004271D3		. 266		JMP 004271D4		004271D4		. 267		JMP 004271D5		004271D5		. 268		JMP 004271D6		004271D6		. 269		JMP 004271D7		004271D7		. 270		JMP 004271D8		004271D8		. 271		JMP 004271D9		004271D9		. 272		JMP 004271DA		004271DA		. 273		JMP 004271DB		004271DB		. 274		JMP 004271DC		004271DC		. 275		JMP 004271DD		004271DD		. 276		JMP 004271DE		004271DE		. 277		JMP 004271DF		004271DF		. 278		JMP 004271E0		004271E0		. 279		JMP 004271E1		004271E1		. 280		JMP 004271E2		004271E2		. 281		JMP 004271E3		004271E3		. 282		JMP 004271E4		004271E4		. 283		JMP 004271E5		004271E5		. 284		JMP 004271E6		004271E6		. 285		JMP 004271E7		004271E7		. 286		JMP 004271E8		004271E8		. 287		JMP 004271E9		004271E9		. 288		JMP 004271EA		004271EA		. 289		JMP 004271EB		004271EB		. 290		JMP 004271EC		004271EC		. 291		JMP 004271ED		004271ED		. 292		JMP 004271EE		004271EE		. 293		JMP 004271EF		004271EF		. 294		JMP 004271F0		004271F0		. 295		JMP 004271F1		004271F1		. 296		JMP 004271F2		004271F2		. 297		JMP 004271F3		004271F3		. 298		JMP 004271F4		004271F4		. 299		JMP 004271F5		004271F5		. 300		JMP 004271F6		004271F6		. 301		JMP 004271F7		004271F7		. 302		JMP 004271F8		004271F8		. 303		JMP 004271F9		004271F9		. 304		JMP 004271FA		004271FA		. 305		JMP 004271FB		004271FB		. 306		JMP 004271FC		004271FC		. 307		JMP 004271FD		004271FD		. 308		JMP 004271FE		004271FE		. 309		JMP 004271FF		004271FF		. 310		JMP 00427200		00427200		. 311		JMP 00427201		00427201		. 312		JMP 00427202		00427202		. 313		JMP 00427203		00427203		. 314		JMP 00427204		00427204		. 315		JMP 00427205		00427205		. 316		JMP	
----------	--	------	--	----------	--	--------------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	--------------	--	----------	--	-------	--	-----	--

mezcladas un par correspondientes a la ntdll, pero ya verificamos que el sistema hace una excepcion para esas apis, que originalmente eran de kernel32.dll y como fueron reemplazadas por apis similares de ntdll.dll, pues por compatibilidad, las acepta como si fueran de la kernel32.dll sin hacer problemas, en cualquier otro caso que el sistema no tenga este tipo de excepciones que son muy pocas realmente (creo que las unicas), las apis deben ir ordenadas por dll, y separadas por ceros con los de otra dll, en una IAT correcta.

Bueno despues de la separacion vienen hacia abajo un grupo de apis de direcciones del tipo 77xxxxxx.

Si uno tiene alguna duda de alguna entrada a que dll pertenece, ademas de ir al mapa de memoria y fijarse, puede marcarla en la IAT y hacer click derecho-FIND REFERENCES.

Address	Hex dump
00460A0C	AB 14 81 7C F4 97 80 7C 01 B0 85 7C 19 62 82 7C
00460A0E	5C E8 81 7C 53 00 83 7C 19 3C 87 7C CB D8 81 7C
00460A0F	C1 0F 87 7C 5B B2 81 7C E9 06 87 7C 4E 99 80 7C
00460A10	AC 92 80 7C 11 07 87 7C 42 24 80 7C F3 B8 81 7C
00460A11	A9 2C 87 7C F4 2C 87 7C BA 38 87 7C 0C 6E 82 7C
00460A12	F1 BA 80 7C 30 31 87 7C 83 31 87 7C CC 37 87 7C
00460A13	77 1D 80 7C 28 AC 80 7C 66 AA 80 7C A9 2C 81 7C
00460A14	ED CB 81 7C 3D 0D 87 7C 19 90 83 7C 59 35 81 7C
00460A15	31 03 92 7C 40 03 92 7C D7 EF 80 7C 2D FF 80 7C
00460A16	2F FE 80 7C 51 28 81 7C 11 03 81 7C B1 C7 80 7C
00460A17	65 A0 80 7C CF C6 80 7C 21 2E 82 7C B0 99 80 7C
00460A18	88 2D 82 7C 5D 99 80 7C 94 97 80 7C 7B 97 80 7C
00460A19	29 B5 80 7C CF C6 80 7C 00 00 00 00 C0 48 0F 77
00460A1A	38 04 0F 77 94 A5 11 77 59 4B 0F 77 82 4E 0F 77
00460A1B	9B D4 11 77 9B 50 0F 77 4F 50 0F 77 10 50 0F 77
00460A1C	0F 03 0F 77 00 00 0F 77 00 00 0F 77 00 00 0F 77

Con lo cual buscara en el listado, todas las instrucciones que utilizan dicha entrada, en este caso.(esto funcionara siempre y cuando el listado este mostrando la seccion del programa donde trabaja el mismo, en este caso la primera seccion, si busco referencias, y el listado lo tengo mostrando otra seccion diferente, buscara en ella, y probablemente no hallara nada, asi que debo verificar antes de usar este metodo, estar en la seccion correcta donde el programa corre ya desempacado, o la zona del oep, que es lo mismo)

Address	Disassembly	Comment
00405435	CALL DWORD PTR DS:[460BA8]	oleaut32.VariantClear
0041CC2D	CALL DWORD PTR DS:[460BA8]	oleaut32.VariantClear
0041F68C	CALL DWORD PTR DS:[460BA8]	oleaut32.VariantClear
00421197	CALL DWORD PTR DS:[460BA8]	oleaut32.VariantClear
0042133C	CALL DWORD PTR DS:[460BA8]	oleaut32.VariantClear
004213E5	CALL DWORD PTR DS:[460BA8]	oleaut32.VariantClear
0042198F	CALL DWORD PTR DS:[460BA8]	oleaut32.VariantClear
00421D90	CALL DWORD PTR DS:[460BA8]	oleaut32.VariantClear
004477B1	CALL DWORD PTR DS:[460BA8]	oleaut32.VariantClear
004477D8	CALL DWORD PTR DS:[460BA8]	oleaut32.VariantClear
004478B1	CALL DWORD PTR DS:[460BA8]	oleaut32.VariantClear
00447E7D	CALL DWORD PTR DS:[460BA8]	oleaut32.VariantClear
00447F0E	CALL DWORD PTR DS:[460BA8]	oleaut32.VariantClear
00447F67	CALL DWORD PTR DS:[460BA8]	oleaut32.VariantClear
00495284	DD bitarts_.00460BA8	

Alli vemos los CALLs que existen en la priumera seccion correspondientes a dicha entrada. La cual pertenece a la api VariantClear de la OleAut32.dll.

76B22000	00002000	winmm	.data	data	Image	RWE	
76B2C000	0000A000	winmm	.rsrc	resources	Image	RWE	
76B2C000	00002000	winmm	.reloc	relocations	Image	RWE	
770F0000	00001000	oleaut32		PE header	Image	RWE	
770F1000	0007F000	oleaut32	.text	code, import	Image	RWE	
77170000	00002000	oleaut32	.orpc	code	Image	RWE	
77172000	00003000	oleaut32	.data	data	Image	RWE	
77175000	00001000	oleaut32	.rsrc	resources	Image	RWE	
77176000	00006000	oleaut32	.reloc	relocations	Image	RWE	
773A0000	00001000	comctl_1		PE header	Image	RWE	

Mirando en el mapa de memoria, vemos que por supuesto el grupo siguiente cae en la seccion code de la OleAut32.dll, como corresponde.

Address	Hex dump	ASCII
00460D9C	42 8C 01 77 2E 8C 01 77 8B 14 03 77 FE EC 03 77	B i 0 w . i 0 w i 0 w i 0 w
00460DAC	83 F7 04 77 DE F2 02 77 DF 1A 03 77 F6 F0 04 77	ä · é w i = é w i → é w i - é w
00460DBC	9C F3 04 77 33 F2 02 77 6C C9 01 77 F6 8B 01 77	ä · é w 3 = é w i f 0 w + i 0 w
00460DDC	B8 96 01 77 0C 94 01 77 61 C6 03 77 81 E5 02 77	0 0 0 w . 0 0 w ä é w i i 0 é w
00460DDC	80 03 03 77 55 E6 01 77 A0 A8 01 77 EA 04 05 77	Ç · é w U p 0 w i ç 0 w 0 · ' w
00460DEC	24 13 02 77 58 BF 01 77 33 B9 01 77 65 F6 04 77	\$! ! é w X 7 0 w 3 i 0 w e + é w
00460DFC	2F BF 01 77 B4 F6 04 77 6C BF 01 77 F5 B5 01 77	/ ä 0 w + é w i 7 0 w ä 0 w
00460E0C	24 15 03 77 E2 C2 01 77 29 69 05 77 DF BA 05 77	\$ \$ é w 0 + 0 w) i ' w ' w
00460E1C	8C 14 02 77 4C 1F 03 77 F9 07 01 77 F7 06 01 77	i 0 w L · é w - i 0 w - i 0 w
00460E2C	65 C4 01 77 04 B6 01 77 C8 BD 01 77 AE B6 01 77	e - 0 w é ä 0 w = c 0 w · ä 0 w
00460E3C	CD 48 02 77 3E 0B 02 77 C7 86 01 77 9D 86 01 77	= H é w > ä é w ä ä 0 w 0 ä 0 w
00460E4C	26 BF 01 77 3F B5 01 77 69 08 01 77 85 CB 01 77	% 7 0 w ? ä 0 w i i 0 w ä i 0 w
00460E5C	71 BE 01 77 6E C6 01 77 9D 8F 01 77 FD BE 01 77	q ä 0 w n ä 0 w 0 ä 0 w 2 · ä 0 w
00460E6C	31 B6 01 77 17 E9 03 77 00 EE 04 77 9A F3 02 77	1 ä 0 w · ä 0 w . · é w 0 % é w
00460E7C	B5 37 02 77 78 8E 01 77 8B EE 04 77 00 00 00 00	ä 7 é w x ä ä 0 w i · é w
00460E8C	F7 A8 B1 76 00 00 00 00 C8 74 F8 72 73 66 F9 72	- ç · é w t · r s f · r
00460E9C	87 72 F8 72 43 80 F8 72 67 37 F9 72 FB 41 F9 72	ç r · r C ç · r g 7 · r ' A · r
00460EAC	67 83 F8 72 90 53 F8 72 00 00 00 00 CE 00 37 76	g ä · r é S · r f · 7 v
00460EBC	7C 86 37 76 80 86 37 76 33 25 36 76 1E 31 36 76	i ä 7 v · ä 7 v 3 2 6 v · 1 6 v
00460ECC	08 7C 37 76 89 C2 37 76 CD 46 38 76 CE EE 36 76	i i 7 v é t 7 v = F 8 v f · 6 v
00460ECC	00 00 00 00 48 D0 4C 77 9C C8 4D 77 CC 42 4F 77 H ä L w ä i 0 w i f B 0 w
00460EEC	2C D0 4C 77 DA F6 4C 77 73 33 50 77 10 64 4D 77	· ä L w r + L w s 3 P w · d i w
00460EEC	03 0E 52 77 33 0F 52 77 40 A6 54 77 F1 A7 54 77	· ä R w 3 · R w 0 ä T w : ä T w
00460EFC	92 9C 4F 77 6F 57 52 77 99 33 4E 77 B2 5D 4E 77	i é ä 0 w o l R w 0 3 N w · j N w
00460F0C	90 C0 5A 77 00 00 00 00 F3 F0 CC 74 00 00 00 00	e ' ä w % - f é
00460F2C	0B 00 50 6C 61 79 53 6F 75 6E 64 41 00 00 57 49	ä · P l a y S o u n d A . . W I
00460F3C	4E 4D 4D 2E 64 6C 6C 00 FE 00 47 65 74 4D 6F 64	N M M . d l l . · G e t M o d
00460F4C	75 6C 65 48 61 6E 64 6C 65 41 00 00 7E 01 49 6E	u l e H a n d l e A . . · 0 I n
00460F5C	74 65 72 6C 6F 63 6B 65 64 49 6E 63 72 65 6D 65	t e r l o c k e d I n c r e m e
00460F6C	6E 74 00 00 7B 01 49 6E 74 65 72 6C 6F 63 6B 65	n t . . (0 I n t e r l o c k e
00460F7C	64 44 65 63 72 65 6D 65 6E 74 00 00 9A 01 4C 6F	d D e c r e m e n t . . ü 0 L o
00460F8C	63 61 6C 46 72 65 65 00 9C 01 4C 6F 63 61 6C 4C	c a l F r e e . é 0 L o c a l l
00460F9C	6F 63 6B 00 96 01 4C 6F 63 61 6C 41 6C 6C 6F 63	o c k . ü 0 L o c a l A l l o c
00460FAC	00 00 A0 01 4C 6F 63 61 6C 55 6E 6C 6F 63 6B 00	. . ä 0 L o c a l U n l o c k .

Si seguimos bajando sin especificar cada dll, vemos grupos de entradas contiguas que van a otras dll, separacion con ceros, otro grupo, otra separacion y asi, seguimos bajando hasta ver donde acaba este esquema para encontrar el final de la iat ya que no se ven ceros donde termina.

00460EBC	7C 86 37 76 80 86 37 76 33 25 36 76 1E 31 36 76	i ä 7 v · ä 7 v 3 2 6 v · 1 6 v
00460ECC	08 7C 37 76 89 C2 37 76 CD 46 38 76 CE EE 36 76	i i 7 v é t 7 v = F 8 v f · 6 v
00460EDC	00 00 00 00 48 D0 4C 77 9C C8 4D 77 CC 42 4F 77 H ä L w ä i 0 w i f B 0 w
00460EEC	2C D0 4C 77 DA F6 4C 77 73 33 50 77 10 64 4D 77	· ä L w r + L w s 3 P w · d i w
00460EEC	03 0E 52 77 33 0F 52 77 40 A6 54 77 F1 A7 54 77	· ä R w 3 · R w 0 ä T w : ä T w
00460F0C	92 9C 4F 77 6F 57 52 77 99 33 4E 77 B2 5D 4E 77	i é ä 0 w o l R w 0 3 N w · j N w
00460F1C	90 C0 5A 77 00 00 00 00 F3 F0 CC 74 00 00 00 00	e ' ä w % - f é
00460F2C	0B 00 50 6C 61 79 53 6F 75 6E 64 41 00 00 57 49	ä · P l a y S o u n d A . . W I
00460F3C	4E 4D 4D 2E 64 6C 6C 00 FE 00 47 65 74 4D 6F 64	N M M . d l l . · G e t M o d
00460F4C	75 6C 65 48 61 6E 64 6C 65 41 00 00 7E 01 49 6E	u l e H a n d l e A . . · 0 I n
00460F5C	74 65 72 6C 6F 63 6B 65 64 49 6E 63 72 65 6D 65	t e r l o c k e d I n c r e m e
00460F6C	6E 74 00 00 7B 01 49 6E 74 65 72 6C 6F 63 6B 65	n t . . (0 I n t e r l o c k e
00460F7C	64 44 65 63 72 65 6D 65 6E 74 00 00 9A 01 4C 6F	d D e c r e m e n t . . ü 0 L o
00460F8C	63 61 6C 46 72 65 65 00 9C 01 4C 6F 63 61 6C 4C	c a l F r e e . é 0 L o c a l l
00460F9C	6F 63 6B 00 96 01 4C 6F 63 61 6C 41 6C 6C 6F 63	o c k . ü 0 L o c a l A l l o c
00460FAC	00 00 A0 01 4C 6F 63 61 6C 55 6E 6C 6F 63 6B 00	. . ä 0 L o c a l U n l o c k .
00460FBC	A3 01 4C 6F 63 6B 52 65 73 6F 75 72 63 65 00 00	ü 0 L o c k R e s o u r c e . .

Alli vemos la parte final donde se termina este esquema, vemos en celeste un grupo de apis, correspondientes a una dll, luego la separacion, luego en rosado otro grupo de apis, la separacion y vemos una entrada marcada con una flecha y otra separacion, luego de la cual ya no se mantiene el mismo esquema, podemos verificar si esta api de la flecha pertenece a la IAT, pues fijemosnos si va a alguna dll con los dos metodos.

Address	Hex dump	Follow DWORD in Dump	Find references	View executable file	Copy to executable file	Go to
00460D9C	42 8C D1 77 2E 8C D1 77 8B 14 D3 77					
00460DAC	83 F7 D4 77 DE F2 D2 77 DF 1A D3 77					
00460DBC	9C F3 D4 77 33 F2 D2 77 6C C9 D1 77					
00460DCC	B8 96 D1 77 0C 94 D1 77 61 C6 D3 77					
00460DDC	80 03 D3 77 55 E6 D1 77 AD A8 D1 77					
00460DEC	24 13 D2 77 58 BF D1 77 33 B9 D1 77					
00460DFC	2F B7 D1 77 B4 F6 D4 77 6C BF D1 77					
00460E0C	24 15 D3 77 E2 C2 D1 77 29 69 D5 77					
00460E1C	8C 14 D2 77 4C 1F D3 77 F9 D7 D1 77					
00460E2C	65 C4 D1 77 D4 B6 D1 77 C8 BD D1 77					
00460E3C	CD 48 D2 77 3E 0B D2 77 C7 86 D1 77					
00460E4C	26 BF D1 77 3F B5 D1 77 69 D8 D1 77					
00460E5C	71 BE D1 77 6E C6 D1 77 9D 8F D1 77					
00460E6C	31 B6 D1 77 17 E9 D3 77 00 EE D4 77					
00460E7C	B5 37 D2 77 78 8E D1 77 88 EE D4 77					
00460E8C	F7 A8 B1 76 00 00 00 00 C8 74 F8 72					
00460E9C	87 72 F8 72 43 80 F8 72 67 37 F9 72					
00460EAC	67 83 F8 72 90 53 F8 72 00 00 00 00					
00460EBC	7C 86 37 76 B0 86 37 76 33 25 36 76					
00460ECC	D8 7C 37 76 89 C2 37 76 CD 46 38 76					
00460EDC	00 00 00 00 48 D0 4C 77 9C CB 4D 77					
00460EEC	2C D0 4C 77 DA F6 4C 77 73 33 50 77					
00460EFC	03 0E 52 77 33 0F 52 77 40 A6 54 77					
00460F0C	92 9C 4F 77 6F 57 52 77 99 33 4E 77					
00460F1C	90 C0 5A 77 00 00 00 00 F3 F0 CC 74					
00460F2C	0B 00 50 6C 61 79 53 6F 75 6E 64 41					
00460F3C	4E 4D 4D 2E 64 6C 6C 00 FE 00 47 65					

Address	Disassembly	Comment
00435D38	JMP DWORD PTR DS:[460F24]	oledlg.OleUIBusyA

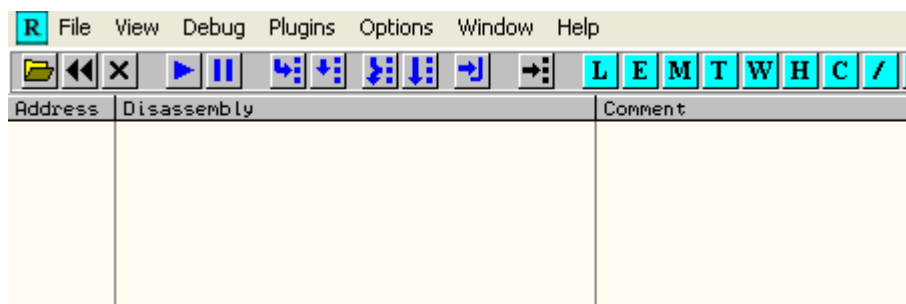
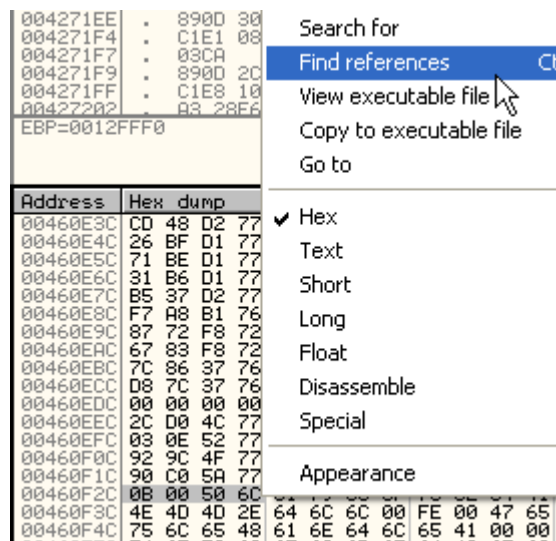
Vemos que es una entrada de la IAT pues nos lleva a una api en este caso OleUiBusyA de la oledlg.dll, si verifico mirando el mapa de memoria.

72FA4000	00002000	winspool	.reloc	relocations	Image R	RWE
74CC0000	00001000	oledlg		PE header	Image R	RWE
74CC1000	00011000	oledlg	.text	code,import	Image R	RWE
74CD2000	00002000	oledlg	.data	data	Image R	RWE
74CD3000	0000B000	oledlg	.rsrc	resources	Image R	RWE
74CD4000	00001000	oledlg	.reloc	relocations	Image R	RWE
76360000	00001000	comdlg32		PE header	Image R	RWE

Vemos que logicamente cae en la seccion CODE de dicha dll, por lo cual es una entrada de la IAT, la unica de esta dll, luego de eso, viene la separacion, y luego vienen grupos de numeros sueltos que no estan agrupados para ir a direcciones contiguas, ni sus direcciones apuntan a ninguna dll.

00460EBC	00 00 00 00 48 D0 4C 77 9C CB 4D 77 CC 42 4F 77	...NoLwdrInW
00460EEC	2C D0 4C 77 DA F6 4C 77 73 33 50 77 10 64 4D 77	,\$Lwr+Lws3Pw
00460EFC	03 0E 52 77 33 0F 52 77 40 A6 54 77 F1 A7 54 77	00Rw3*Rw@ATw
00460F0C	92 9C 4F 77 6F 57 52 77 99 33 4E 77 B2 5D 4E 77	E\$Qw0wRw03Nw\$
00460F1C	90 C0 5A 77 00 00 00 00 F3 F0 CC 74 00 00 00 00	E'Zw...%lft.
00460F2C	0B 00 50 6C 61 79 53 6F 75 6E 64 41 00 00 57 49	\$.PlaySoundA.
00460F3C	4E 4D 4D 2E 64 6C 6C 00 FE 00 47 65 74 4D 6F 64	NMM.dll. Get
00460F4C	75 6C 48 61 6E 6C 65 41 00 00 7E 01 63 6E	uleHandleA..'
00460F5C	74 65 63 6F 63 63 65 64 49 6E 63 72 65 6D 65	terlockedInco
00460F6C	6E 74 00 00 7B 01 49 6E 74 65 73 6C 6F 63 6B 65	nt..(0Interlo
00460F7C	64 44 65 63 72 65 6D 65 6E 74 00 00 9A 01 4C 6F	dDecrement..i
00460F8C	63 61 6C 46 72 65 65 00 9C 01 4C 6F 63 61 6C 4C	calFree.\$0Loc
00460F9C	2E 23 2B 0A 62 81 4D 2E 23 21 2D 41 2C 2C 2E 23	ok \$0Loc

Si marcamos cualquiera de ellos, y hacemos FIND REFERENCES.



No hay resultados, que nos lleven a apis, por lo cual podemos deducir ya que mas abajo no encontramos mas entradas que nos lleven a apis, que aquí se termino la IAT por lo tanto el final de la misma es

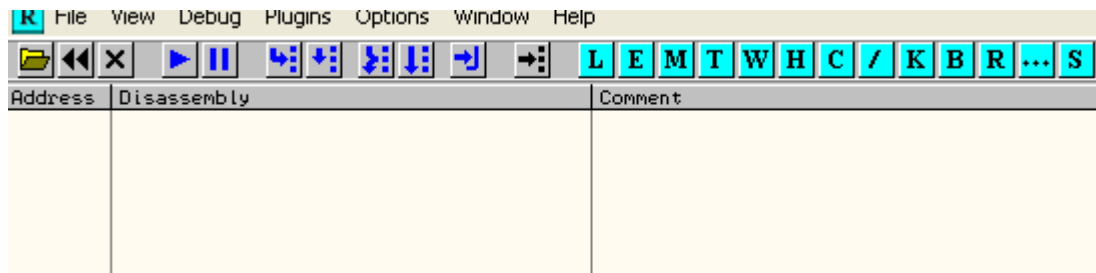
FINAL: 460F28

Address	Hex dump	ASCII
00460EF8	10 64 40 77 03 0E 52 77 33 0F 52 77 40 A6 54 77	!dMwRw3Rw@Tw
00460F08	F1 A7 54 77 92 9C 4F 77 6F 57 52 77 99 33 4E 77	!TwE0w0Rw03Nw
00460F18	B2 5D 4E 77 90 C0 5A 77 00 00 00 00 F3 F0 CC 74	!Nwe'Zw...-lft
00460F28	00 00 00 00 0B 00 50 6C 61 79 53 6F 75 6E 64 41	...PlaySoundA
00460F38	00 00 57 49 4E 4D 4D 2E 64 6C 6C 00 FE 00 47 65	..WINMM.dll..Ge
00460F48	74 4D 6F 64 75 6C 65 48 61 6E 64 6C 65 41 00 00	tModuleHandleA..
00460F58	7E 01 49 6E 74 65 72 6C 6F 63 6B 65 64 49 6E 63	"@InterlockedInc

Ahora hagamos lo mismo hacia arriba verificando donde comienza el esquema de la IAT.

Address	Hex dump	ASCII
004607C0	80 28 06 00 70 28 06 00 00 00 00 00 0C 2B 06 00	C(.p(.+...+.
004607D0	FA 2A 06 00 E8 20 06 00 1E 2B 06 00 08 2A 06 00	.*.p*.A+.i*.
004607E0	C6 0A 06 00 AE 20 06 00 92 2B 06 00 72 2A 06 00	g*..<*.E*.x*.
004607F0	BC 00 06 00 A8 2B 06 00 92 2B 06 00 78 2B 06 00	+*.L+.+...+.
00460800	60 00 06 00 4C 2B 06 00 2E 2B 06 00 00 00 00 00	'+.L+.+...+.
00460810	08 00 00 80 00 00 00 00 F0 6B DA 77 1B 76 DA 77	...C...-k rwv rw
00460820	F4 EA DA 77 E7 EB DA 77 83 78 DA 77 00 00 00 00	!U rwB rwAX rw...
00460830	DD 15 C5 58 2E BD C3 58 00 00 00 00 04 6A EF 77	!S+X.c!X...Ej'w
00460840	66 95 EF 77 89 6A EF 77 F3 AD EF 77 ED 09 EF 77	f0'wEj'w%j'wY'w
00460850	99 8B EF 77 0B 85 EF 77 2A 7D EF 77 B2 7C EF 77	0i'w'A'w*)'w
00460860	77 53 F2 77 1E C9 F1 77 0C BC EF 77 52 04 EF 77	wS=wAf'w.'wRE'w
00460870	FA 8D EF 77 F1 0D EF 77 51 B2 EF 77 26 05 EF 77	.i'w!:'wC'w'w
00460880	2A E3 EF 77 5F 39 F2 77 71 B4 EF 77 2E AD EF 77	*0'w_9=wq'w.i'w
00460890	E1 61 EF 77 B8 85 EF 77 CC 02 EF 77 43 70 EF 77	Pa'w03'wFE'wCp'w
004608A0	FB EA F0 77 12 83 EF 77 01 72 F0 77 A9 34 F0 77	'U-w03'w0r-w04-w
004608B0	D5 93 EF 77 68 EF EF 77 AA 02 EF 77 B2 6F EF 77	'0'wh'wE'w0o'w
004608C0	3F 38 F2 77 D6 E8 EF 77 68 E0 EF 77 00 60 EF 77	?8=wIb'wh0'w.'w
004608D0	90 5B EF 77 6D AC EF 77 94 6C F0 77 22 8D EF 77	E['wm%w0l-w'i'w
004608E0	3D C8 F1 77 30 6D F0 77 6F C0 EF 77 85 7B EF 77	=!w-m-w0'w0'w
004608F0	26 D9 EF 77 FB 5E EF 77 36 8A EF 77 FC 8A EF 77	&'w'^'w6E'wE'w
00460900	0F 62 EF 77 49 5E EF 77 97 5D EF 77 1A 9A EF 77	*b'wI'^'wU]'wU'w

Vemos que el esquema se va repitiendo hacia arriba hasta que llegamos aquí, en celeste estan marcadas las separaciones, el grupo marcado en amarillo es el que visualmente reconozco como un grupo ordenado que va a una dll, luego hay una separacion y una entrada que va a 8000008, como no se si no corresponde a alguna entrada suelta de alguna dll, verifico haciendo FIND REFERENCES.

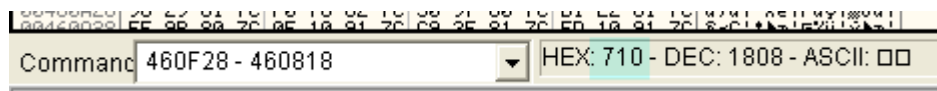


No hay resultados que vayan a apis, ni en esa entrada, ni en ninguna superior por lo cual, podemos asegurar que la primera entrada es 460818.

Address	Hex dump	ASCII
004607D8	1E 2B 06 00 08 2A 06 00 C6 2A 06 00 AE 2A 06 00	Δ+Δ. I**Δ. Δ**Δ. <Δ**Δ.
004607E8	92 2A 06 00 72 2A 06 00 BC 2B 06 00 A8 2B 06 00	Δ**Δ. Δ**Δ. Δ+Δ. Δ+Δ.
004607F8	92 2B 06 00 78 2B 06 00 60 2B 06 00 4C 2B 06 00	Δ+Δ. Δ+Δ. Δ+Δ. L+Δ.
00460808	2E 2B 06 00 00 00 00 00 08 00 00 80 00 00 00 00	.+Δ.....Δ...Δ....
00460818	F0 6B DA 77 1B 76 DA 77 F4 EA DA 77 E7 EB DA 77	Δk r w+ v r w Δ U r w Δ U r w
00460828	83 78 DA 77 00 00 00 00 DD 15 C5 58 2E BD C3 58	Δk r w.....f s+X.c t+X
00460838	00 00 00 00 04 6A EF 77 66 95 EF 77 89 6A EF 77Ej' w f Δ' w Δ j' w
00460848	F3 AD EF 77 ED 09 EF 77 99 8B EF 77 C0 85 EF 77	%Δ' w Δ' w Δ i' w Δ' w
00460858	2A 7D EF 77 B2 7C EF 77 77 53 F2 77 1E C9 F1 77	*)' w Δ' w Δ S= w Δ f Δ' w
00460868	0C BC EF 77 52 04 EF 77 FA 8D EF 77 F1 DD EF 77	.Δ' w Δ E' w Δ' w Δ i' w Δ' w
00460878	51 B2 EF 77 26 05 EF 77 2A E3 EF 77 5F 39 F2 77	CΔΔ' w Δ' w Δ Δ' w Δ 9= w
00460888	71 B4 EF 77 2E AD EF 77 E1 61 EF 77 B8 85 EF 77	q t' w Δ' w Δ β Δ' w Δ β Δ' w
00460898	CC 02 EF 77 43 70 EF 77 FB EA F0 77 12 83 EF 77	l f E' w Δ C p' w Δ' U' w Δ β Δ' w
004608A8	01 72 F0 77 A9 34 F0 77 05 93 EF 77 68 EF EF 77	θ r- w Δ 4- w Δ' Δ' w h' w
004608B8	AA 02 EF 77 B2 6F EF 77 3F 38 F2 77 D6 E8 EF 77	Δ E' w Δ Δ Δ' w Δ 8= w Δ i β' w
004608C8	68 E0 EF 77 00 60 EF 77 90 5B EF 77 6D AC EF 77	h Δ' w Δ' w Δ E Δ' w Δ k Δ' w
004608D8	94 6C F0 77 22 8D EF 77 3D C8 F1 77 3D 6D F0 77	Δ l- w Δ' w Δ E Δ' w Δ n- w
004608E8	6F C0 EF 77 85 7B EF 77 26 09 EF 77 FB 5E EF 77	Δ t' w Δ Δ' w Δ Δ' w Δ' Δ' w
004608F8	36 8A EF 77 FC 8A EF 77 0F 62 EF 77 49 5E EF 77	Δ Δ' w Δ' w Δ Δ' w Δ i Δ' w
00460908	97 5D EF 77 1A 9A EF 77 6B FA EF 77 7B C9 F0 77	Δ j' w Δ' U' w Δ k Δ' w Δ f Δ' w
00460918	DA 98 F2 77 1A 40 F2 77 55 EA EF 77 C5 61 EF 77	r Δ' w Δ θ= w Δ U Δ' w Δ a' w
00460928	70 E6 EF 77 F0 81 EF 77 2D 6C EF 77 98 6E EF 77	Δ Δ' w Δ' U' w Δ' w Δ n' w

por lo tanto ya tenemos el INICIO y FINAL de la IAT, hallamos el largo.

FINAL-INICIO= 460F28 – 460818



o sea que el largo es 710

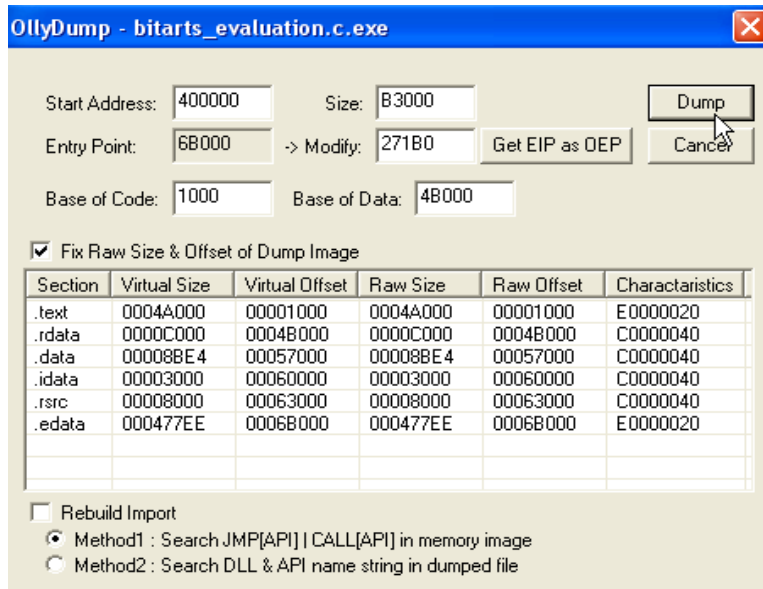
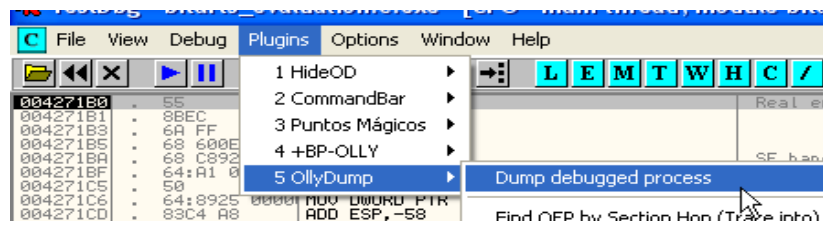
recopilando para el IMP REC (le restamos ya la imagebase 400000 al OEP y INICIO)

OEP= **271B0**

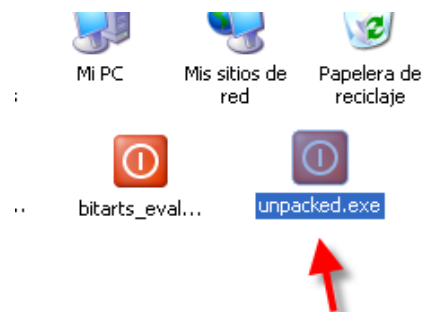
INICIO o RVA= **60818**

LARGO= **710**

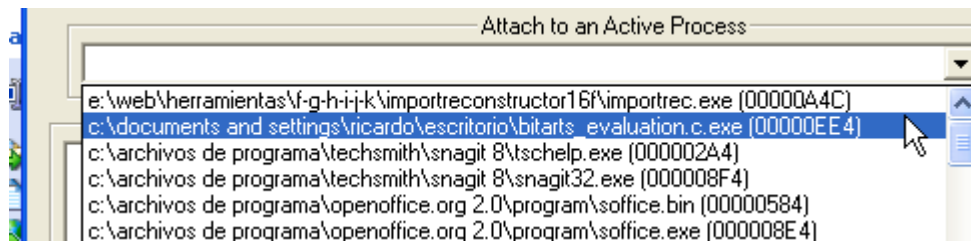
Bueno hacemos el dumpeado con el OLLYDMP.



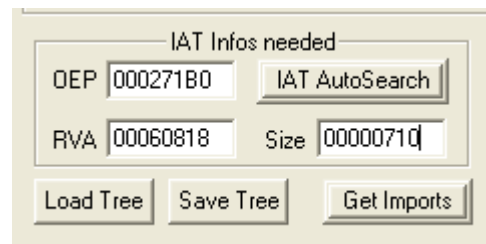
Le quitamos la tilde a Rebuild Import y dumpeamos.



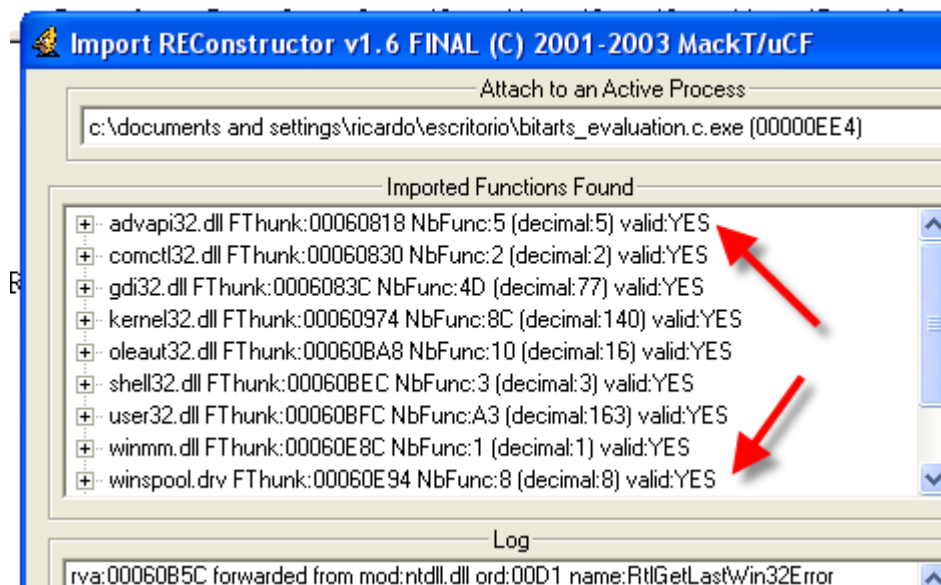
Luego abramos el IMP REC sin cerrar el OLLYDBG, con el original detenido en el OEP.



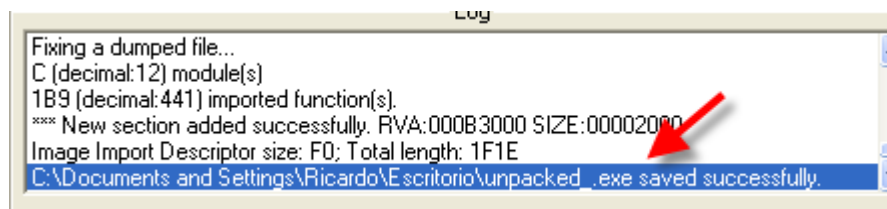
Buscamos el proceso en el menu desplegable y le colocamos los valores hallados.



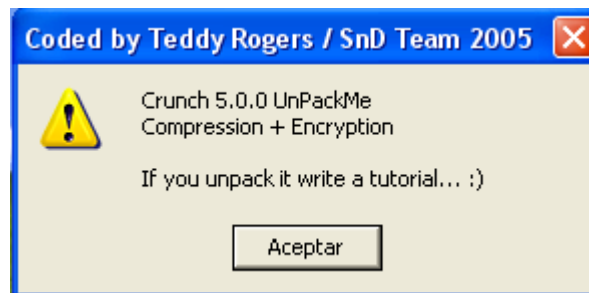
Y apretamos GET IMPORTS



Vemos que el packer no hace mucho por molestarnos todas las dlls estan correctas asi que apretamos FIX DUMP y buscamos el dumpeado para que lo repare.

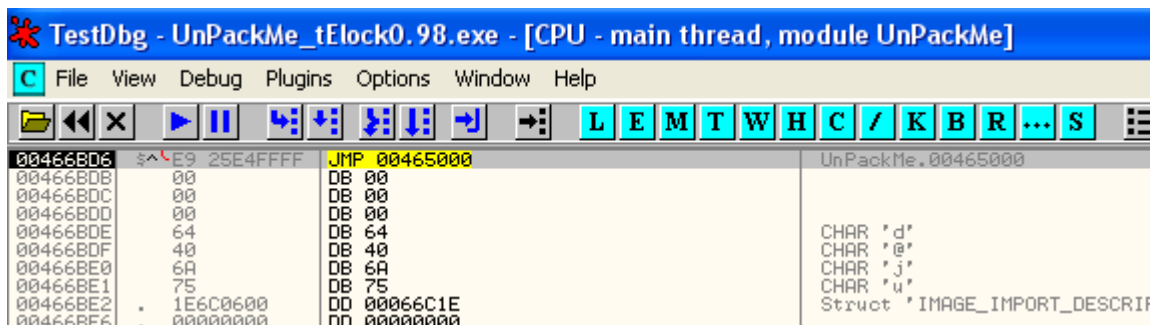


Lo guardo al reparado con el nombre unpacked_.exe, probemoslo a ver si funciona.

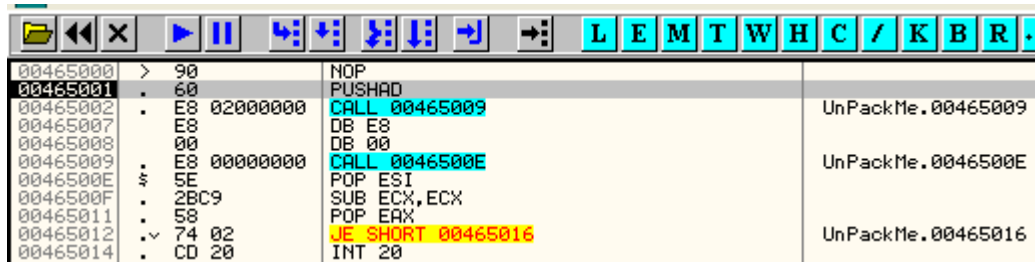


Funciona perfectamente, por lo cual no tiene antidumps, los cuales seguramente encontraremos mas adelante en packers mas complejos.

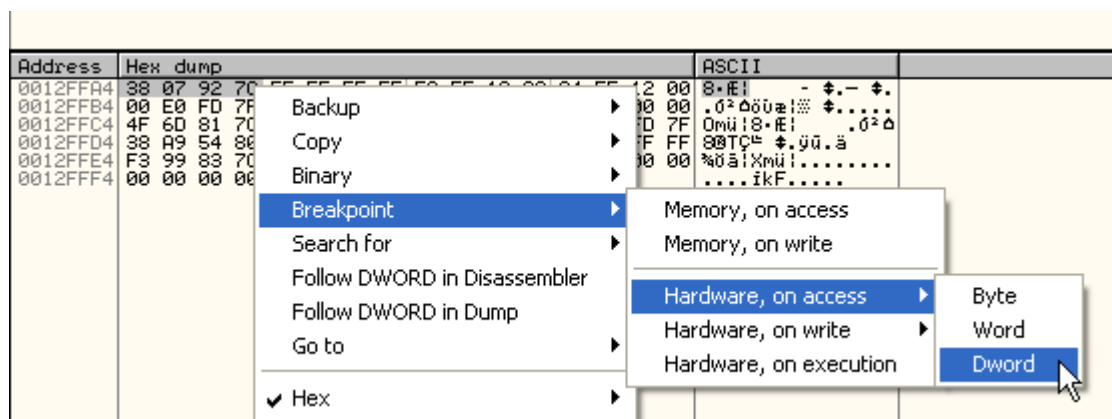
La proxima victima es el telock 0.98 que nos servira para empezar el tema de las entradas de la IAT redireccionadas.

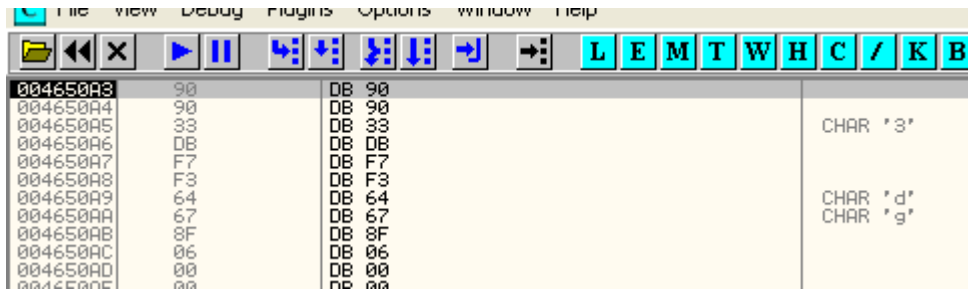


Pribemos el metodo del pushad, traceemos un poco a ver si hay algun PUSHAD.



Pasemoslo con f7, luego marcamos ESP-FOLLOW IN DUMP.

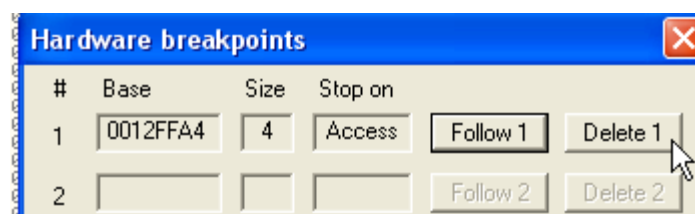
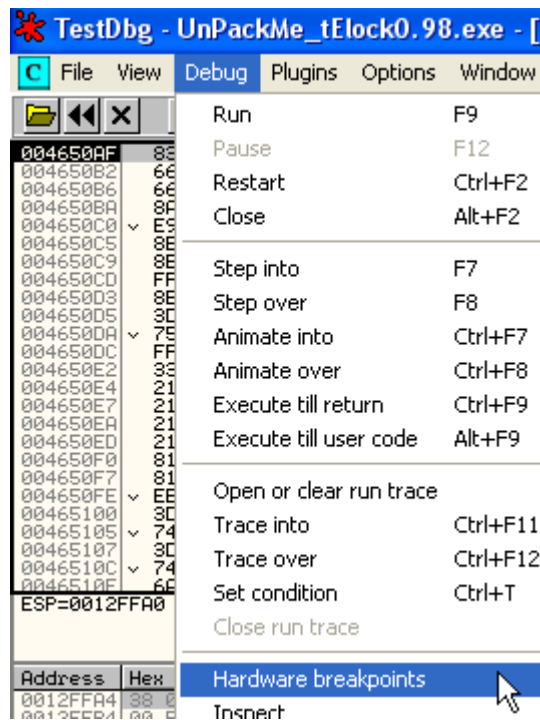




Quitamos el analisis a si se ve el codigo.



Vemos que el metodo del pushad no funciona y que ademas esta protegido contra HARDWARE BREAKPOINTS porque si lo sigo corriendo da error, asi que quito el hardware breakpoint que coloque.



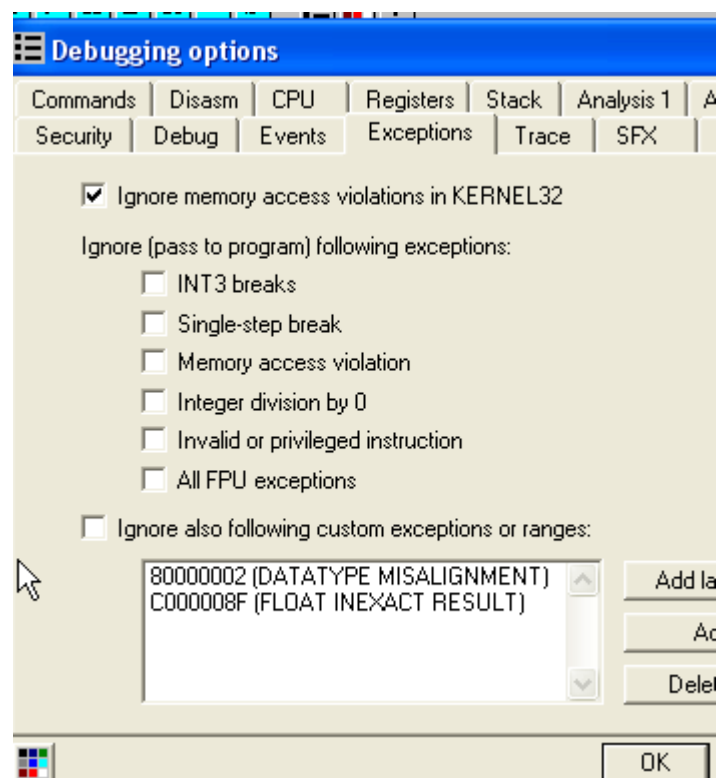
Y reinicio el OLLYDBG.

Pues ese metodo no va, probemos con el de las excepciones, limpio el LOG y lo hago correr al programa para que me liste las excepciones que pasa.

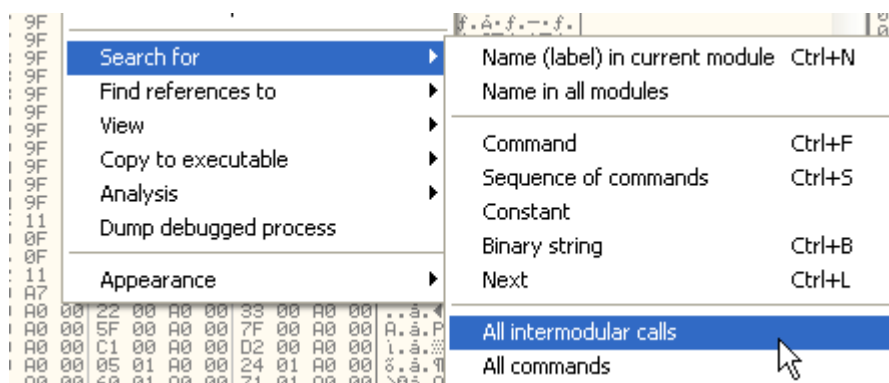
```
0046508C INT3 command at UnPackMe.0046508C
004650A7 Integer division by zero
004656A8 Illegal instruction
00465AA1 Integer division by zero
00465B27 INT3 command at UnPackMe.00465B27
00465B67 Integer division by zero
00465BA6 Access violation when reading [FFFFFFFF]
00465BF0 INT3 command at UnPackMe.00465BF0
76B00000 Module C:\WINDOWS\system32\WINMM.dll
77DA0000 Module C:\WINDOWS\system32\ADVAPI32.dll
77E50000 Module C:\WINDOWS\system32\RPCRT4.dll
5D160000 Module C:\WINDOWS\system32\serwvdrv.dll
5B480000 Module C:\WINDOWS\system32\umdmpfm.dll
76360000 Module C:\WINDOWS\system32\comdlg32.dll
77F40000 Module C:\WINDOWS\system32\SHLWAPI.dll
77BE0000 Module C:\WINDOWS\system32\msvcrt.dll
50C30000 Module C:\WINDOWS\system32\COMCTL32.dll
70900000 Module C:\WINDOWS\system32\SHELL32.dll
72900000 Module C:\WINDOWS\WinSxS\x86_Microsoft.Windows
74000000 Module C:\WINDOWS\system32\WINSPOOL.DRV
74000000 Module C:\WINDOWS\system32\oledlg.dll
774B0000 Module C:\WINDOWS\system32\ole32.dll
004666F1 Illegal instruction
770F0000 Module C:\WINDOWS\system32\OLEAUT32.dll
5B150000 Module C:\WINDOWS\system32\uxtheme.dll
746B0000 Module C:\WINDOWS\system32\MSCTF.dll
73260000 Module C:\WINDOWS\system32\RICHED32.DLL
```

Command

Pues quitemos las tildes de las excepciones y reiniciemos tratando de pescar la de 4666f1 que es la ultima del desempacador.



Pues luego de pasar varias excepciones con SHIFT mas f9 llegamos a 4666f1.



para ver los CALLs que se dirigen a otras secciones, intentando ver si hay calls a alguna api.

Address	Disassembly	Destination
004231C8	CALL FC4273FE	
00423310	CALL DWORD PTR DS:[460AB0]	DS:[00460AB0]=009F061C
00423A4F	CALL DWORD PTR DS:[460A00]	DS:[00460A00]=009F02AF
00423B1C	CALL DWORD PTR DS:[4609FC]	DS:[004609FC]=009F029E
00423C43	CALL DWORD PTR DS:[4609FC]	DS:[004609FC]=009F029E
00423CA8	CALL DWORD PTR DS:[4609F8]	DS:[004609F8]=009F028D
00423E5C	CALL DWORD PTR DS:[460A58]	DS:[00460A58]=009F046A
00423E96	CALL DWORD PTR DS:[460B5C]	DS:[00460B5C]=009F0973
004249B6	CALL DWORD PTR DS:[4609F4]	DS:[004609F4]=009F027C
00425003	CALL DWORD PTR DS:[460B98]	DS:[00460B98]=009F0AA0
004251B3	CALL DWORD PTR DS:[460B98]	DS:[00460B98]=009F0AA0
004251C7	CALL DWORD PTR DS:[460B94]	DS:[00460B94]=009F0A7D
0042520B	CALL DWORD PTR DS:[460B94]	DS:[00460B94]=009F0A7D
004252D4	CALL DWORD PTR DS:[460B94]	DS:[00460B94]=009F0A7D
004252FB	CALL DWORD PTR DS:[460978]	DS:[00460978]=009F0011
00425306	CALL DWORD PTR DS:[460974]	DS:[00460974]=009F0000
0042535E	CALL DWORD PTR DS:[4609B0]	DS:[004609B0]=009F0124
004259D1	CALL DWORD PTR DS:[460A54]	DS:[00460A54]=009F0447
004259E8	CALL DWORD PTR DS:[460A3C]	DS:[00460A3C]=009F03DC
004259F5	CALL DWORD PTR DS:[460A44]	DS:[00460A44]=009F03FB
00425B2B	CALL DWORD PTR DS:[460AF8]	DS:[00460AF8]=009F0785
00425B3F	CALL DWORD PTR DS:[460AF8]	DS:[00460AF8]=009F0785
00425B53	CALL DWORD PTR DS:[460AF8]	DS:[00460AF8]=009F0785
00425BF1	CALL DWORD PTR DS:[460AF8]	DS:[00460AF8]=009F0785
00425C9C	CALL DWORD PTR DS:[46097C]	DS:[0046097C]=009F0022
00425CA6	CALL DWORD PTR DS:[460B5C]	DS:[00460B5C]=009F0973
00425D71	CALL DWORD PTR DS:[460A54]	DS:[00460A54]=009F0447
00425D8B	CALL DWORD PTR DS:[460A3C]	DS:[00460A3C]=009F03DC
00425DBB	CALL DWORD PTR DS:[460A44]	DS:[00460A44]=009F03FB
00425E13	CALL DWORD PTR DS:[460B98]	DS:[00460B98]=009F0AA0
00425E27	CALL DWORD PTR DS:[460B94]	DS:[00460B94]=009F0A7D
00425E6B	CALL DWORD PTR DS:[460B94]	DS:[00460B94]=009F0A7D
00425F34	CALL DWORD PTR DS:[460B94]	DS:[00460B94]=009F0A7D
004271B0	PUSH EBP	(Initial CPU selection)
004271D6	CALL DWORD PTR DS:[460ADC]	DS:[00460ADC]=009F06F7
0042723E	CALL DWORD PTR DS:[460984]	DS:[00460984]=009F0041
004272D5	CALL DWORD PTR DS:[460980]	DS:[00460980]=009F0033
004272F6	CALL DWORD PTR DS:[460B9C]	DS:[00460B9C]=009F0AB1
004274F7	CALL 00435CC0	UnPackMe.00435CC0
004277F3	CALL 00435CC0	UnPackMe.00435CC0
00427929	CALL DWORD PTR DS:[46098C]	DS:[0046098C]=009F005F
00427E07	CALL DWORD PTR DS:[460A84]	DS:[00460A84]=009F0539
00427E0E	CALL DWORD PTR DS:[460994]	DS:[00460994]=009F008D
00427E98	CALL DWORD PTR DS:[460990]	DS:[00460990]=009F007F
00428029	CALL DWORD PTR DS:[4609F8]	DS:[004609F8]=009F028D
004280BB	CALL DWORD PTR DS:[460A08]	DS:[00460A08]=009F02CC
00428823	CALL DWORD PTR DS:[460A54]	DS:[00460A54]=009F0447
0042884D	CALL DWORD PTR DS:[460A3C]	DS:[00460A3C]=009F03DC
0042886E	CALL DWORD PTR DS:[460A44]	DS:[00460A44]=009F03FB
004288AE	CALL DWORD PTR DS:[460A3C]	DS:[00460A3C]=009F03DC
004288E0	CALL DWORD PTR DS:[460A3C]	DS:[00460A3C]=009F03DC
0042891E	CALL DWORD PTR DS:[460A44]	DS:[00460A44]=009F03FB
00428950	CALL DWORD PTR DS:[460A44]	DS:[00460A44]=009F03FB
00428983	CALL DWORD PTR DS:[460B98]	DS:[00460B98]=009F0AA0
00428C15	CALL DWORD PTR DS:[460A9C]	DS:[00460A9C]=009F0AB1

Veo que hay muchísimos Calls indirectos que en vez de dirigirse a una api de una dll, van en este caso una sección 9fxxxx en mi máquina, pero que en su máquina puede variar y ser otra dirección parecida..

Si miro más abajo en esta lista

00435D95	CALL	DWORD PTR DS:[460B70]	DS:[00460B70]=009F062C
00435D96	CALL	DWORD PTR DS:[460B74]	DS:[00460B74]=009F06F0
00435D9F	CALL	DWORD PTR DS:[460AB0]	DS:[00460AB0]=009F061C
00435E06	CALL	DWORD PTR DS:[460AB4]	DS:[00460AB4]=009F0636
00435FA5	CALL	00435CDE	comdlg32.PrintDlgA
00436004	CALL	DWORD PTR DS:[460AD0]	DS:[00460AD0]=009F06B5
0043602E	CALL	DWORD PTR DS:[460AD0]	DS:[00460AD0]=009F06B5
0043606C	CALL	DWORD PTR DS:[460AD0]	DS:[00460AD0]=009F06B5
004360A8	CALL	DWORD PTR DS:[460AD0]	DS:[00460AD0]=009F06B5
004360D1	CALL	00435CDE	comdlg32.PrintDlgA
00436118	CALL	DWORD PTR DS:[460878]	DS:[00460878]=00A10124
00436230	CALL	DWORD PTR DS:[460B74]	DS:[00460B74]=009F06F0
004362EC	CALL	DWORD PTR DS:[460CAC]	DS:[00460CAC]=00A00360
00436315	CALL	DWORD PTR DS:[460CD4]	DS:[00460CD4]=00A00419
00436351	CALL	00435CEA	comdlg32.GetOpenFileNameA
00436358	CALL	00435CE4	comdlg32.GetSaveFileNameA
00436374	CALL	DWORD PTR DS:[460DF4]	DS:[00460DF4]=00A009B7
00436382	CALL	DWORD PTR DS:[460CB0]	DS:[00460CB0]=00A00370
00436407	CALL	DWORD PTR DS:[460E78]	DS:[00460E78]=00A00C52
0043644E	CALL	DWORD PTR DS:[460E78]	DS:[00460E78]=00A00C52
004366CA	CALL	DWORD PTR DS:[460DFC]	DS:[00460DFC]=00A009F0
004368C5	CALL	DWORD PTR DS:[460D48]	DS:[00460D48]=00A00669
004368D8	CALL	DWORD PTR DS:[460D48]	DS:[00460D48]=00A00669
004368F1	CALL	DWORD PTR DS:[460D48]	DS:[00460D48]=00A00669
00436907	CALL	DWORD PTR DS:[460D48]	DS:[00460D48]=00A00669
00436910	CALL	DWORD PTR DS:[460D48]	DS:[00460D48]=00A00669
00436933	CALL	DWORD PTR DS:[460D48]	DS:[00460D48]=00A00669
00436A6A	CALL	DWORD PTR DS:[460E78]	DS:[00460E78]=00A00C52
00436BD0	CALL	00435CFC	comdlg32.FindTextA
00436BD7	CALL	00435CF6	comdlg32.ReplaceTextA
00436CB1	CALL	DWORD PTR DS:[460AA8]	DS:[00460AA8]=009F05EC
00436CC3	CALL	DWORD PTR DS:[460AAC]	DS:[00460AAC]=009F05FD
0043799F	CALL	DWORD PTR DS:[460AD8]	DS:[00460AD8]=00A00669

Vemos que hay algunos CALLS directos que nos marca que van a apis, seguramente por medio de un JMP INDIRECTO, vayamos a ver alguno de estos CALLS.

Address	Disassembly	Comment
00435FA5	E8 34FDFFFF CALL 00435CDE	JMP to comdlg32.PrintDlgA
00435FAA	8BCE MOV ECX,ESI	
00435FAC	8BF8 MOV EDI,EAX	
00435FAE	E8 6E4E0000 CALL 0043AE21	UnPackMe.0043AE21
00435FB3	85FF TEST EDI,EDI	
00435FB5	74 04 JE SHORT 00435FBB	UnPackMe.00435FBB
00435FB7	8BC7 MOV EAX,EDI	
00435FB9	EB 03 JMP SHORT 00435FBE	UnPackMe.00435FBE
00435FBB	6A 02 PUSH 2	
00435FBD	58 POP EAX	
00435FBE	5F POP EDI	
00435FBF	5E POP ESI	
00435FC0	C3 RETN	

Alli tenemos uno, es un CALL 435CDE que ira seguro a los JMPS INDIRECTOS a las apis, marquemoslo y hagamos click derecho- FOLLOW.

00435CBF	90	NOP	
00435CC0	- FF25 88094600	JMP DWORD PTR DS:[460988]	
00435CC6	- FF25 000C4600	JMP DWORD PTR DS:[460C00]	
00435CCC	- FF25 040C4600	JMP DWORD PTR DS:[460C04]	
00435CD2	- FF25 D00E4600	JMP DWORD PTR DS:[460ED0]	comdlg32.ChooseFontA
00435CD8	- FF25 D80E4600	JMP DWORD PTR DS:[460ED8]	comdlg32.ChooseColorA
00435CDE	- FF25 D40E4600	JMP DWORD PTR DS:[460ED4]	comdlg32.PrintDlgA
00435CE4	- FF25 CC0E4600	JMP DWORD PTR DS:[460ECC]	comdlg32.GetSaveFileNameA
00435CEA	- FF25 C80E4600	JMP DWORD PTR DS:[460EC8]	comdlg32.GetOpenFileNameA
00435CF0	- FF25 C40E4600	JMP DWORD PTR DS:[460EC4]	comdlg32.GetFileTitleA
00435CF6	- FF25 C00E4600	JMP DWORD PTR DS:[460EC0]	comdlg32.ReplaceTextA
00435CFC	- FF25 BC0E4600	JMP DWORD PTR DS:[460EBC]	comdlg32.FindTextA
00435D02	- FF25 B80E4600	JMP DWORD PTR DS:[460EB8]	comdlg32.CommDlgExtendedError
00435D08	- FF25 B00E4600	JMP DWORD PTR DS:[460EB0]	WINSPOOL.ClosePrinter
00435D0E	- FF25 940E4600	JMP DWORD PTR DS:[460E94]	WINSPOOL.EndDocPrinter
00435D14	- FF25 AC0E4600	JMP DWORD PTR DS:[460EAC]	WINSPOOL.StartPagePrinter
00435D1A	- FF25 A80E4600	JMP DWORD PTR DS:[460EA8]	WINSPOOL.StartDocPrinter
00435D20	- FF25 A40E4600	JMP DWORD PTR DS:[460EA4]	WINSPOOL.OpenPrinterA
00435D26	- FF25 A00E4600	JMP DWORD PTR DS:[460EA0]	WINSPOOL.EndPagePrinter
00435D2C	- FF25 9C0E4600	JMP DWORD PTR DS:[460E9C]	WINSPOOL.WritePrinter
00435D32	- FF25 980E4600	JMP DWORD PTR DS:[460E98]	WINSPOOL.DocumentPropertiesA
00435D38	- FF25 240F4600	JMP DWORD PTR DS:[460F24]	oledlg.OleUIBusyA
00435D3E	CC	INT3	
00435D3F	CB	INT3	

Alli vemos los saltos indirectos a las apis, por lo tanto sabemos que toman valores de la IAT, o sea que 460ED4 es una entrada de la IAT, vayamos en el DUMP a verla.

Address	Hex dump	ASCII
00460EB4	00 00 00 00 CE 00 37 76 7C 86 37 76 B0 86 37 76f.7v!37v37v
00460EC4	33 25 36 76 1E 31 36 76 D8 7C 37 76 89 C2 37 76	3%6v16vii7v37v
00460ED4	CD 46 38 76 CE EE 36 76 00 00 00 00 48 D0 4C 77	=F8v!6v...H3Lw
00460EE4	9C CB 4D 77 CC 42 4F 77 2C D0 4C 77 DA F6 4C 77	5fMw!FBOw,3Lwr+Lw
00460EF4	73 33 50 77 10 64 4D 77 03 0E 52 77 33 0F 52 77	s3Pw!dMw#Rw3*Ww
00460F04	40 A6 54 77 F1 A7 54 77 92 9C 4F 77 6F 57 52 77	@3Tw!3Tw!E6Dw0!Rw
00460F14	99 33 4E 77 B2 5D 4E 77 90 C0 5A 77 00 00 00 00	03Nw!JNw!2w....
00460F24	F3 F0 CC 74 00 00 00 00 00 00 00 00 00 00 00	%-!ft.....
00460F34	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00460F44	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00460F54	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00460F64	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00460F74	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00460F84	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00460F94	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00460FA4	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

Vemos que la parte final de la IAT es correcta y coincide con la del ejemplo de CRUNCH, el final de la IAT aquí también es 460F28, es fácil detectarlo pues aquí si termina la IAT y empiezan todos ceros, ahora subamos.

EBP=0012FFF0		
Address	Hex dump	ASCII
00460D54	95 06 A0 00 B5 06 A0 00 C3 06 A0 00 E6 06 A0 00	00
00460D64	F7 06 A0 00 08 07 A0 00 1A 07 A0 00 2A 07 A0 00	00
00460D74	3B 07 A0 00 5A 07 A0 00 74 07 A0 00 85 07 A0 00	00
00460D84	96 07 A0 00 A7 07 A0 00 B5 07 A0 00 C4 07 A0 00	00
00460D94	D3 07 A0 00 F3 07 A0 00 01 08 A0 00 24 08 A0 00	00
00460DA4	35 08 A0 00 46 08 A0 00 58 08 A0 00 68 08 A0 00	00
00460DB4	79 08 A0 00 98 08 A0 00 B2 08 A0 00 C3 08 A0 00	00
00460DC4	D4 08 A0 00 E5 08 A0 00 F3 08 A0 00 02 09 A0 00	00
00460DD4	11 09 A0 00 31 09 A0 00 3F 09 A0 00 62 09 A0 00	00
00460DE4	73 09 A0 00 84 09 A0 00 96 09 A0 00 A6 09 A0 00	00
00460DF4	B7 09 A0 00 D6 09 A0 00 F0 09 A0 00 01 0A 00 00	00
00460E04	12 0A 00 00 23 0A 00 00 31 0A 00 00 40 0A 00 00	00
00460E14	4F 0A 00 00 6F 0A 00 00 7D 0A 00 00 0A 0A 00 00	00
00460E24	B1 0A 00 00 C2 0A 00 00 D4 0A 00 00 E4 0A 00 00	00
00460E34	F5 0A 00 00 14 0B 00 00 2E 0B 00 00 3F 0B 00 00	00
00460E44	5D 0B 00 00 61 0B 00 00 6F 0B 00 00 7E 0B 00 00	00
00460E54	8D 0B 00 00 AD 0B 00 00 BB 0B 00 00 DE 0B 00 00	00
00460E64	EF 0B 00 00 00 0C 00 00 12 0C 00 00 22 0C 00 00	00
00460E74	33 0C 00 00 52 0C 00 00 6C 0C 00 00 7D 0C 00 00	00
00460E84	8E 0C 00 00 9F 0C 00 00 F7 0C 00 00 B1 76 00 00 00	00
00460E94	C8 74 F8 72 73 66 F9 72 87 72 F8 72 43 80 F8 72	00
00460EA4	67 37 F9 72 FB 41 F9 72 67 83 F8 72 90 53 F8 72	00
00460EB4	00 00 00 00 CE 00 37 76 7C 86 37 76 B0 86 37 76f.7v!37v37v
00460EC4	33 25 36 76 1E 31 36 76 D8 7C 37 76 89 C2 37 76	3%6v16vii7v37v
00460ED4	CD 46 38 76 CE EE 36 76 00 00 00 00 48 D0 4C 77	=F8v!6v...H3Lw
00460EE4	9C CB 4D 77 CC 42 4F 77 2C D0 4C 77 DA F6 4C 77	5fMw!FBOw,3Lwr+Lw
00460EF4	73 33 50 77 10 64 4D 77 03 0E 52 77 33 0F 52 77	s3Pw!dMw#Rw3*Ww
00460F04	40 A6 54 77 F1 A7 54 77 92 9C 4F 77 6F 57 52 77	@3Tw!3Tw!E6Dw0!Rw
00460F14	99 33 4E 77 B2 5D 4E 77 90 C0 5A 77 00 00 00 00	03Nw!JNw!2w....
00460F24	F3 F0 CC 74 00 00 00 00 00 00 00 00 00 00 00	%-!ft.....
00460F34	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00460F44	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00460F54	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00460F64	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

Vemos ya, que el proximo grupo esta conflictuado, la primera entrada arriba de la separacion corresponde a 76B1A8F7 si la marco y hago click derecho-FIND REFERENCES.

Address	Disassembly	Comment
0040E203	CALL DWORD PTR DS:[460E8C]	WINMM.PlaySoundA
00460E95	JE SHORT 00460E8F	UnPackMe.00460E8F

Vemos que pertenece a la api PlaySoundA de la WINMM.dll de allí para arriba, no encuentro mas valores que vayan a dlls, pero si busco referencias.

Address	Disassembly	Comment
004038A6	CALL DWORD PTR DS:[460E48]	DS:[00460E48]=00A00B61
004047D0	CALL DWORD PTR DS:[460E48]	DS:[00460E48]=00A00B61
00404923	CALL DWORD PTR DS:[460E48]	DS:[00460E48]=00A00B61
00404AD9	CALL DWORD PTR DS:[460E48]	DS:[00460E48]=00A00B61
00413318	CALL DWORD PTR DS:[460E48]	DS:[00460E48]=00A00B61
004171DE	CALL DWORD PTR DS:[460E48]	DS:[00460E48]=00A00B61
0041A61E	CALL DWORD PTR DS:[460E48]	DS:[00460E48]=00A00B61
00421045	CALL DWORD PTR DS:[460E48]	DS:[00460E48]=00A00B61
0043D912	CALL DWORD PTR DS:[460E48]	DS:[00460E48]=00A00B61
0043DCC0	CALL DWORD PTR DS:[460E48]	DS:[00460E48]=00A00B61
0043E9D6	CALL DWORD PTR DS:[460E48]	DS:[00460E48]=00A00B61
0043EA8A	CALL DWORD PTR DS:[460E48]	DS:[00460E48]=00A00B61
00441B0A	CALL DWORD PTR DS:[460E48]	DS:[00460E48]=00A00B61

Veo que si existen referencias, por lo cual aquí hay un punto, cuando llegabamos subiendo y bajando al inicio de la IAT si buscabamos arriba del inicio, por ejemplo, no encontrabamos ninguna referencia, pues el programa salta a las apis de las dlls, por medio de la IAT, fuera de la misma ya no encontramos ninguna referencia, ahora aquí vemos que estas son posibles entradas de la IAT, ya que hallamos referencias que toman valores de ellas en el codigo, pero, en vez de saltar a las dlls, salta a una seccion que en mi caso, esta en Axxxxx o puede variar según cada maquina, como es esto?

Pues esto es lo que se llaman entradas redireccionadas, el desempacador al ejecutarse, sobrescribe algunas de las entradas a la IAT con valores que apuntan a rutinas propias, en el caso de la imagen anterior

004038A6 CALL DWORD PTR DS:[460E48]

Comment=DS:[00460E48]=00A00B61

En vez de guardar la direccion correcta de la api en mi maquina, el desempacador reemplaza dicha direccion por una direccion de una seccion propia creada por el, en tiempo de ejecucion, y alli pone una rutina que al final termina yendo a la api correcta.

Para aclarar un poco veamos la entrada a GetVersion que esta en el inicio del programa debajo del OEP.

Address	Disassembly	Comment
004271B0	55	PUSH EBP
004271B1	8BEC	MOV EBP,ESP
004271B3	6A FF	PUSH -1
004271B5	68 600E4500	PUSH 450E60
004271B9	68 C8924200	PUSH 4292C8
004271BF	64:A1 00000000	MOV EAX,DWORD PTR FS:[0]
004271C5	50	PUSH EAX
004271C6	64:8925 000000	MOV DWORD PTR FS:[0],ESP
004271CD	83C4 A8	ADD ESP,-58
004271D0	53	PUSH EBX
004271D1	56	PUSH ESI
004271D2	57	PUSH EDI
004271D3	8965 E8	MOV DWORD PTR SS:[EBP-18],ESP
004271D6	FF15 DC0A4600	CALL DWORD PTR DS:[460ADC]
004271DC	33D2	XOR EDX,EDX
004271DE	8AD4	MOV DL,AH
004271E0	8915 34E64500	MOV DWORD PTR DS:[45E634],EDX

Realmente no sabemos que va a GetVersion, solo porque el ejemplo anterior era un programa similar pero empacado con CRUNCH lo sabemos, pero realmente hasta aquí, para nosotros es un CALL INDIRECTO, llegemos hasata el con f7 y entremos en el a ver donde va.

Address	Hex dump	ASCII
00460A9C	B9 05 9F 00 CA 05 9F 00 DC 05 9F 00 EC 05 9F 00	11 05 9F 00 CA 05 9F 00 DC 05 9F 00 EC 05 9F 00
00460AAC	FD 05 9F 00 1C 06 9F 00 36 06 9F 00 47 06 9F 00	21 05 9F 00 1C 06 9F 00 36 06 9F 00 47 06 9F 00
00460ABC	58 06 9F 00 69 06 9F 00 77 06 9F 00 86 06 9F 00	31 06 9F 00 69 06 9F 00 77 06 9F 00 86 06 9F 00
00460ACC	95 06 9F 00 B5 06 9F 00 C3 06 9F 00 E6 06 9F 00	41 06 9F 00 B5 06 9F 00 C3 06 9F 00 E6 06 9F 00
00460ADC	F7 06 9F 00 08 07 9F 00 1A 07 9F 00 2A 07 9F 00	51 07 9F 00 08 07 9F 00 1A 07 9F 00 2A 07 9F 00
00460AEC	3B 07 9F 00 5A 07 9F 00 74 07 9F 00 85 07 9F 00	61 07 9F 00 5A 07 9F 00 74 07 9F 00 85 07 9F 00
00460AFC	96 07 9F 00 A7 07 9F 00 B5 07 9F 00 C4 07 9F 00	71 07 9F 00 A7 07 9F 00 B5 07 9F 00 C4 07 9F 00
00460B0C	D3 07 9F 00 F3 07 9F 00 01 08 9F 00 24 08 9F 00	81 08 9F 00 F3 07 9F 00 01 08 9F 00 24 08 9F 00
00460B1C	35 08 9F 00 46 08 9F 00 58 08 9F 00 68 08 9F 00	91 08 9F 00 46 08 9F 00 58 08 9F 00 68 08 9F 00
00460B2C	79 08 9F 00 98 08 9F 00 B2 08 9F 00 C3 08 9F 00	01 09 9F 00 98 08 9F 00 B2 08 9F 00 C3 08 9F 00
00460B3C	D4 08 9F 00 E5 08 9F 00 F3 08 9F 00 02 09 9F 00	11 09 9F 00 E5 08 9F 00 F3 08 9F 00 02 09 9F 00
00460B4C	11 09 9F 00 31 09 9F 00 3F 09 9F 00 62 09 9F 00	21 09 9F 00 31 09 9F 00 3F 09 9F 00 62 09 9F 00
00460B5C	73 09 9F 00 84 09 9F 00 96 09 9F 00 A6 09 9F 00	31 09 9F 00 84 09 9F 00 96 09 9F 00 A6 09 9F 00
00460B6C	B7 09 9F 00 D6 09 9F 00 F0 09 9F 00 01 0A 9F 00	41 09 9F 00 D6 09 9F 00 F0 09 9F 00 01 0A 9F 00
00460B7C	12 0A 9F 00 23 0A 9F 00 31 0A 9F 00 40 0A 9F 00	51 0A 9F 00 23 0A 9F 00 31 0A 9F 00 40 0A 9F 00
00460B8C	4F 0A 9F 00 6F 0A 9F 00 7D 0A 9F 00 A0 0A 9F 00	61 0A 9F 00 6F 0A 9F 00 7D 0A 9F 00 A0 0A 9F 00
00460B9C	B1 0A 9F 00 C2 0A 9F 00 D4 0A 9F 00 C0 4B 0F 77	71 0A 9F 00 C2 0A 9F 00 D4 0A 9F 00 C0 4B 0F 77
00460BAC	3B 4C 0F 77 94 A5 11 77 59 4B 0F 77 82 4E 0F 77	81 0A 9F 00 3B 4C 0F 77 94 A5 11 77 59 4B 0F 77
00460BBC	98 04 11 77 9B 50 0F 77 4F 50 0F 77 10 50 0F 77	91 0A 9F 00 98 04 11 77 9B 50 0F 77 4F 50 0F 77
00460BCC	3F 50 0F 77 D9 66 0F 77 50 48 0F 77 55 4C 0F 77	01 0B 9F 00 3F 50 0F 77 D9 66 0F 77 50 48 0F 77
00460BDC	C2 4B 0F 77 95 D2 11 77 80 5D 15 77 00 00 00 00	11 0B 9F 00 C2 4B 0F 77 95 D2 11 77 80 5D 15 77
00460BEC	00 00 A7 00 11 00 A7 00 22 00 A7 00 33 00 A7 00	21 0B 9F 00 00 00 A7 00 11 00 A7 00 22 00 A7 00
00460BFC	00 00 A0 00 11 00 A0 00 22 00 A0 00 33 00 A0 00	31 0B 9F 00 00 00 A0 00 11 00 A0 00 22 00 A0 00
00460C0C	41 00 A0 00 50 00 A0 00 5F 00 A0 00 7F 00 A0 00	41 0B 9F 00 41 00 A0 00 50 00 A0 00 5F 00 A0 00
00460C1C	8D 00 A0 00 B0 00 A0 00 C1 00 A0 00 D2 00 A0 00	51 0B 9F 00 8D 00 A0 00 B0 00 A0 00 C1 00 A0 00
00460C2C	E4 00 A0 00 F4 00 A0 00 05 01 A0 00 24 01 A0 00	61 0B 9F 00 E4 00 A0 00 F4 00 A0 00 05 01 A0 00
00460C3C	3E 01 A0 00 4F 01 A0 00 60 01 A0 00 71 01 A0 00	71 0B 9F 00 3E 01 A0 00 4F 01 A0 00 60 01 A0 00
00460C4C	7F 01 A0 00 8E 01 A0 00 9D 01 A0 00 BD 01 A0 00	81 0B 9F 00 7F 01 A0 00 8E 01 A0 00 9D 01 A0 00
00460C5C	CB 01 A0 00 EE 01 A0 00 FF 01 A0 00 10 02 A0 00	91 0B 9F 00 CB 01 A0 00 EE 01 A0 00 FF 01 A0 00
00460C6C	22 02 A0 00 32 02 A0 00 43 02 A0 00 62 02 A0 00	01 0C 9F 00 22 02 A0 00 32 02 A0 00 43 02 A0 00
00460C7C	7C 02 A0 00 8D 02 A0 00 9E 02 A0 00 AF 02 A0 00	11 0C 9F 00 7C 02 A0 00 8D 02 A0 00 9E 02 A0 00
00460C8C	BD 02 A0 00 CC 02 A0 00 DB 02 A0 00 FE 02 A0 00	21 0C 9F 00 BD 02 A0 00 CC 02 A0 00 DB 02 A0 00
00460C9C	09 03 A0 00 2C 03 A0 00 3D 03 A0 00 4E 03 A0 00	31 0C 9F 00 09 03 A0 00 2C 03 A0 00 3D 03 A0 00
00460CAC	60 03 A0 00 70 03 A0 00 81 03 A0 00 93 03 A0 00	41 0C 9F 00 60 03 A0 00 70 03 A0 00 81 03 A0 00
00460CBC	BA 03 A0 00 CB 03 A0 00 DC 03 A0 00 ED 03 A0 00	51 0C 9F 00 BA 03 A0 00 CB 03 A0 00 DC 03 A0 00
00460CCC	FB 03 A0 00 0A 04 A0 00 19 04 A0 00 39 04 A0 00	61 0C 9F 00 FB 03 A0 00 0A 04 A0 00 19 04 A0 00
00460CDC	47 04 A0 00 6A 04 A0 00 7B 04 A0 00 8C 04 A0 00	71 0C 9F 00 47 04 A0 00 6A 04 A0 00 7B 04 A0 00
00460CEC	9E 04 A0 00 AE 04 A0 00 BF 04 A0 00 DE 04 A0 00	81 0C 9F 00 9E 04 A0 00 AE 04 A0 00 BF 04 A0 00
00460CFE	F8 04 00 00 09 05 00 00 10 05 00 00 2B 05 00 00	91 0C 9F 00 F8 04 00 00 09 05 00 00 10 05 00 00

Vemos que para ver donde va a tomar el valor de lo que aun no sabemos con certeza, pero son posibles entradas de la IAT ya que mas abajo vemos entradas correctas a apis.

Address	Disassembly	Comment
009F06F7	85E4	TEST ESP,ESP
009F06F9	79 03	JNS SHORT 009F06FE
009F06FB	0F9142 40	SETNO BYTE PTR DS:[EDX+40]
009F06FF	B8 D3179F00	MOV EAX,9F17D3
009F0704	40	INC EAX
009F0705	FF30	PUSH DWORD PTR DS:[EAX]
009F0707	C3	RETN
009F0708	EB 02	JMP SHORT 009F070C
009F070A	CD 20	INT 20
009F070C	05 1DAEE17C	ADD EAX,7CE1AE1D
009F0711	B8 D7179F00	MOV EAX,9F17D7
009F0712	40	INC EAX

Por lo tanto el programa al ejecutar ese CALL va aquí que en mi maquina a la direccion 9F06F7, en las suyas puede cambiar.

Esta direccion no pertenece a las secciones del programa en si.

00260000	00003000				Map	RW	RW	
00270000	00016000				Map	R	R	\Device\HarddiskVolu
00290000	0003D000				Map	R	R	\Device\HarddiskVolu
002D0000	00041000				Map	R	R	\Device\HarddiskVolu
00320000	00006000				Map	R	R	\Device\HarddiskVolu
00330000	00041000				Map	R	R	
00380000	00001000				Priv	RWE	RWE	
00390000	00001000				Priv	RWE	RWE	
003A0000	00001000				Priv	RW	RW	
003B0000	00001000				Priv	RW	RW	
00400000	00001000	UnPackMe		PE header	Imag	RW	RWE	
00401000	0004A000	UnPackMe	.teddy	code	Imag	RW	RWE	
0044B000	0000C000	UnPackMe	.teddy	data	Imag	RW	RWE	
00457000	00009000	UnPackMe	.teddy		Imag	RW	RWE	
00460000	00003000	UnPackMe	.teddy		Imag	RW	RWE	
00463000	00002000	UnPackMe	.rsrc	resources	Imag	RW	RWE	
00465000	00004000	UnPackMe	.teddy	SFX, imports	Imag	RW	RWE	
00470000	00009000				Map	R E	R E	
00530000	00002000				Map	R E	R E	
00540000	00103000				Map	R	R	
00650000	00106000				Map	R E	R E	
00970000	00001000				Priv	RW	RW	
009F0000	00002000				Priv	RW	RW	
00A00000	00002000				Priv	RW	RW	
00A10000	00001000				Priv	RW	RW	
00A20000	00004000				Priv	RW	RW	
00A30000	00003000				Map	R	R	\Device\HarddiskVolu
00A40000	00004000				Priv	RW	RW	
00A50000	00003000				Priv	RW	RW	
00A60000	00002000				Map	R	R	
00A70000	00001000				Priv	RW	RW	
A1270000	00002000				Map	R	R	

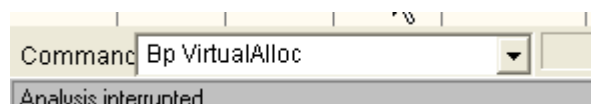
Alli vemos las secciones del programa en celeste y mas abajo una seccion sin nombre donde esta ubicada la rutina donde salta el programa, fuera de las secciones del mismo.

Si reiniciamos el programa vemos que esa seccion no existe al inicio.

00330000	00041000				Map	R	R	
00380000	00001000				Priv	RWE	RWE	
00390000	00001000				Priv	RWE	RWE	
003A0000	00001000				Priv	RW	RW	
003B0000	00001000				Priv	RW	RW	
00400000	00001000	UnPackMe		PE header	Imag	R	RWE	
00401000	0004A000	UnPackMe	.teddy	code	Imag	R	RWE	
0044B000	0000C000	UnPackMe	.teddy	data	Imag	R	RWE	
00457000	00009000	UnPackMe	.teddy		Imag	R	RWE	
00460000	00003000	UnPackMe	.teddy		Imag	R	RWE	
00463000	00002000	UnPackMe	.rsrc	resources	Imag	R	RWE	
00465000	00004000	UnPackMe	.teddy	SFX, imports	Imag	R	RWE	
00470000	00009000				Map	R E	R E	
00530000	00002000				Map	R E	R E	
00540000	00103000				Map	R	R	
00650000	00106000				Map	R E	R E	
77D10000	00001000	user32		PE header	Imag	R	RWE	
77D11000	0005F000	user32	.text	code, import	Imag	R	RWE	
77D70000	00002000	user32	.data	data	Imag	R	RWE	
77D72000	00002000	user32	.rsrc	resources	Imag	R	RWE	

Por lo tanto vemos que fue creada por el programa mientras se va desempacando, ahora, podemos verificar el momento en que dicha seccion es creada?

Pues podemos poner un BP VirtualAlloc que es la api encaragada de crear secciones virtuales y ubicarlas.



Ahora demos RUN y si pongo todas las tildes en exceptions vemos que el programa no corre y se termina, por lo cual es obvio que detecta el BP que acabo de poner, probemos ponerlo en el RET de la api.

7C809A86	FF75 14	PUSH DWORD PTR SS:[EBP+14]	
7C809A89	FF75 10	PUSH DWORD PTR SS:[EBP+10]	
7C809A8C	FF75 0C	PUSH DWORD PTR SS:[EBP+C]	
7C809A8F	FF75 08	PUSH DWORD PTR SS:[EBP+8]	
7C809A92	6A FF	PUSH -1	
7C809A94	E8 09000000	CALL 7C809AA2	kernel32.VirtualAllocEx
7C809A99	5D	POP EBP	
7C809A9A	C2 1000	RETN 10	
7C809A9D	90	NOP	
7C809A9E	90	NOP	
7C809A9F	90	NOP	

Ahora doy RUN

7C809A86	FF75 14	PUSH DWORD PTR SS:[EBP+14]	
7C809A89	FF75 10	PUSH DWORD PTR SS:[EBP+10]	
7C809A8C	FF75 0C	PUSH DWORD PTR SS:[EBP+C]	
7C809A8F	FF75 08	PUSH DWORD PTR SS:[EBP+8]	
7C809A92	6A FF	PUSH -1	
7C809A94	E8 09000000	CALL 7C809AA2	kernel32.VirtualAllocEx
7C809A99	5D	POP EBP	
7C809A9A	C2 1000	RETN 10	
7C809A9D	90	NOP	
7C809A9E	90	NOP	
7C809A9F	90	NOP	

Registers (FPU)
 EAX: 003C0000
 ECX: 7C809AE9 kern
 EDX: 7C91EB94 ntdl
 EBX: 00028000
 ESP: 0012FF64
 EBP: 0005995E
 ESI: 00000004
 EDI: 0005995E
 EIP: 7C809A9A ke
 C 0 ES 0023 32t

Como para en el retorno de la api, la misma devuelve en EAX la direccion base de la seccion creada en este caso par, y creo una seccion en 3c0000, sigamos.

Registers (FPU)
 EAX: 00950000
 ECX: 7C809AE9 ke
 EDX: 7C91EB94 ntdl
 EBX: 00001610
 ESP: 0012FF64
 EBP: 0005995E
 ESI: 00000002
 EDI: 00059976
 EIP: 7C809A9A ke
 C 0 ES 0023 32t

Vemos que va creando secciones las cuales podemos ver en el mapa de memoria

003B0000	00001000	UnPackMe		PE header	Priv	Rw	Rw
00400000	00001000	UnPackMe		code	Imag	R	RwE
00401000	0004A000	UnPackMe	.teddy	data	Imag	R	RwE
0044B000	0000C000	UnPackMe	.teddy		Imag	R	RwE
00457000	00009000	UnPackMe	.teddy		Imag	R	RwE
00460000	00003000	UnPackMe	.teddy		Imag	R	RwE
00463000	00002000	UnPackMe	.rsrc	resources	Imag	R	RwE
00465000	00004000	UnPackMe	.teddy	SFX,imports	Imag	R	RwE
00470000	00009000				Map	R E	R E
00530000	00002000				Map	R E	R E
00540000	00103000				Map	R	R
00650000	00106000				Map	R E	R E
00950000	00002000				Priv	Rw	Rw

Vemos como ayuda que cuando el programa crea secciones las que usara, son marcadas como PRIV o PRIVADAS, mas la ayuda que significa ver que esas secciones no estaban en el inicio, antes de desempacar, pues, sabemos con certeza que son secciones creadas al desempacar.

Por lo tanto quitamos el BP y llegamos al OEP, como anteriormente, quitando las tildes de las excepciones y llegando a la ultima, poniendo un BPM ON ACCESS en la primera seccion, saltando la ultima con SHIFT mas F9.

004271B0	55	PUSH EBP	Real entry point of SFX code
004271B1	8BEC	MOV EBP,ESP	
004271B3	6A FF	PUSH -1	
004271B5	68 600E4500	PUSH 450E60	
004271B8	68 C8924200	PUSH 4292C8	
004271BF	64:A1 00000000	MOV EAX,DWORD PTR FS:[0]	
004271C5	50	PUSH EAX	
004271C6	64:8925 00000000	MOV DWORD PTR FS:[0],ESP	
004271CD	83C4 A8	ADD ESP,-58	
004271D0	53	PUSH EBX	
004271D1	56	PUSH ESI	
004271D2	57	PUSH EDI	
004271D3	8965 E8	MOV DWORD PTR SS:[EBP-18],ESP	
004271D6	FF15 DC0A4600	CALL DWORD PTR DS:[460ADC]	
004271DC	33D2	XOR EDX,EDX	
004271DE	80D4	MOVB AL,BH	

Alli llegamos nuevamente al OEP y si vemos el el mapa de memoria

00400000	00001000	UnPackMe		PE header	Image	RW	RWE	
00401000	0004A000	UnPackMe	.teddy	code	Image	RW	RWE	
0044B000	0000C000	UnPackMe	.teddy	data	Image	RW	RWE	
00457000	00009000	UnPackMe	.teddy		Image	RW	RWE	
00460000	00003000	UnPackMe	.teddy		Image	RW	RWE	
00463000	00002000	UnPackMe	.rsrc	resources	Image	RW	RWE	
00465000	00004000	UnPackMe	.teddy	SFX, imports	Image	RW	RWE	
00470000	00009000				Map	R E	R E	
00530000	00002000				Map	R E	R E	
00540000	00103000				Map	R	R	
00650000	00106000				Map	R E	R E	
00970000	00001000				Priv	RW	RW	
009F0000	00002000				Priv	RW	RW	
00A00000	00002000				Priv	RW	RW	
00A10000	00001000				Priv	RW	RW	
00A20000	00004000				Priv	RW	RW	
00A30000	00003000				Map	R	R	\Devi
00A40000	00004000				Priv	RW	RW	
00A50000	00003000				Priv	RW	RW	
00A60000	00002000				Map	R	R	
00A70000	00001000				Priv	RW	RW	
01270000	00002000				Map	R	R	
58C30000	00001000	COMCTL32		PE header	Image	R	RWE	
58C31000	00070000	COMCTL32	.text	code, import	Image	R	RWE	
58CA1000	00003000	COMCTL32	.data	data	Image	R	RWE	

Vemos secciones creadas que van a ser utilizadas, estan marcadas como PRIV y alli se dirige el call ese indirecto al cual volvemos a entrar traceando con f7.

009F06F7	85E4	TEST ESP,ESP	
009F06F9	79 03	JNS SHORT 009F06FE	
009F06FB	0F9142 40	SETNO BYTE PTR DS:[EDX+40]	
009F06FF	B8 D3179F00	MOV EAX,9F17D3	
009F0704	40	INC EAX	
009F0705	FF30	PUSH DWORD PTR DS:[EAX]	
009F0707	C3	RET	
009F0708	EB 02	JMP SHORT 009F070C	
009F070A	CD 20	INT 20	
009F070C	05 1DAEE17C	ADD EAX,7CE1AE1D	
009F0711	B8 D7179F00	MOV EAX,9F17D7	

Pues traceemos a ver donde llega esta rutina

009F06FE	40	INC EAX	
009F06FF	B8 D3179F00	MOV EAX,9F17D3	
009F0704	40	INC EAX	
009F0705	FF30	PUSH DWORD PTR DS:[EAX]	kernel32.GetVersion
009F0707	C3	RET	
009F0708	EB 02	JMP SHORT 009F070C	
009F070A	CD 20	INT 20	
009F070C	05 1DAEE17C	ADD EAX,7CE1AE1D	
009F0711	B8 D7179F00	MOV EAX,9F17D7	

Vemos que llega aquí a un PUSH que pone la direccion de GetVersion en el stack, y luego salta a la api al llegar al RET, por lo cual esta rutina es como un intermediario para llegar a la api GetVersion.

O sea que el desempacador reemplazo la entrada de la api GetVersion, con una direccion que apunta a una seccion propia o creada por el y no a una dll, y que si traceamos al final del cuentas nos lleva a la api correcta.

Pues esta es la definicion de una entrada redireccionada, exactamente.

Por lo tanto cuando verificamos el inicio y final de la IAT no solo debemos verificar que una entrada vaya a una dll, si no tambien debemos aceptar como entradas de la IAT, aquellas que tienen referencias y que nos llevan a un codigo propio del packer, que resulta ser un intermediario para llegar a la api finalmente.

Por lo cual volvamos a la IAT como estabamos mirando para hallar el INICIO y FINAL.

Address	Hex dump	ASCII
00460C5C	CB 01 A0 00 EE 01 A0 00 FF 01 A0 00 10 02 A0 00	if0á.70á.0á.00á.
00460C6C	22 02 A0 00 32 02 A0 00 43 02 A0 00 62 02 A0 00	"0á.20á.C0á.b0á.
00460C7C	7C 02 A0 00 8D 02 A0 00 9E 02 A0 00 AF 02 A0 00	!0á.10á.x0á.x0á.
00460C8C	BD 02 A0 00 CC 02 A0 00 DB 02 A0 00 FB 02 A0 00	c0á.1f0á.00á.00á.
00460C9C	09 03 A0 00 2C 03 A0 00 3D 03 A0 00 4E 03 A0 00	.0á.00á.00á.00á.
00460CAC	60 03 A0 00 70 03 A0 00 81 03 A0 00 A0 03 A0 00	.0á.p0á.u0á.00á.
00460CBC	BA 03 A0 00 CB 03 A0 00 DC 03 A0 00 ED 03 A0 00	0á.1f0á.00á.00á.
00460CCC	FB 03 A0 00 0A 04 A0 00 19 04 A0 00 39 04 A0 00	'0á..0á.00á.00á.
00460CDC	47 04 A0 00 6A 04 A0 00 7B 04 A0 00 8C 04 A0 00	G0á.J0á.C0á.00á.
00460CEC	9E 04 A0 00 AE 04 A0 00 BF 04 A0 00 DE 04 A0 00	x0á.<0á.00á.00á.
00460CFC	F8 04 A0 00 09 05 A0 00 1A 05 A0 00 2B 05 A0 00	o0á..0á.00á.00á.
00460D0C	39 05 A0 00 48 05 A0 00 57 05 A0 00 77 05 A0 00	90á.H0á.U0á.W0á.
00460D1C	85 05 A0 00 48 05 A0 00 B9 05 A0 00 CA 05 A0 00	á0á.C0á.00á.00á.
00460D2C	DC 05 A0 00 EC 05 A0 00 FD 05 A0 00 1C 06 A0 00	00á.Y0á.00á.00á.
00460D3C	36 06 A0 00 47 06 A0 00 58 06 A0 00 69 06 A0 00	60á.G0á.X0á.00á.
00460D4C	77 06 A0 00 86 06 A0 00 95 06 A0 00 B5 06 A0 00	w0á.00á.00á.00á.
00460D5C	C3 06 A0 00 E6 06 A0 00 F7 06 A0 00 08 07 A0 00	t0á.p0á.00á.00á.
00460D6C	1A 07 A0 00 2A 07 A0 00 3B 07 A0 00 5A 07 A0 00	+0á.*0á.00á.00á.
00460D7C	74 07 A0 00 85 07 A0 00 96 07 A0 00 A7 07 A0 00	t0á.00á.00á.00á.
00460D8C	B5 07 A0 00 C4 07 A0 00 D3 07 A0 00 F3 07 A0 00	A0á.-0á.E0á.00á.
00460D9C	01 08 A0 00 24 08 A0 00 35 08 A0 00 46 08 A0 00	G0á.00á.00á.00á.
00460DAC	58 08 A0 00 68 08 A0 00 79 08 A0 00 98 08 A0 00	X0á.h0á.y0á.00á.
00460DBC	B2 08 A0 00 C3 08 A0 00 D4 08 A0 00 E5 08 A0 00	00á.00á.00á.00á.
00460DCC	F3 08 A0 00 02 09 A0 00 11 09 A0 00 31 09 A0 00	%0á.00á.00á.00á.
00460DDC	3F 09 A0 00 62 09 A0 00 73 09 A0 00 84 09 A0 00	?0á.b0á.00á.00á.
00460DEC	96 09 A0 00 A6 09 A0 00 B7 09 A0 00 D6 09 A0 00	ú0á.00á.00á.00á.
00460DFC	F0 09 A0 00 01 0A 00 12 0A 00 23 0A 00 00	-0á.00á.00á.00á.
00460E0C	31 0A 00 40 0A 00 4F 0A 00 6F 0A 00 00	10á.00á.00á.00á.
00460E1C	7D 0A 00 A0 0A 00 B1 0A 00 C2 0A 00 00	J0á.00á.00á.00á.
00460E2C	D4 0A 00 E4 0A 00 F5 0A 00 14 0B A0 00 00	E0á.00á.00á.00á.
00460E3C	2E 0B A0 00 3F 0B A0 00 50 0B A0 00 61 0B A0 00	.0á.00á.00á.00á.
00460E4C	6F 0B A0 00 7E 0B A0 00 8D 0B A0 00 AD 0B A0 00	o0á.00á.00á.00á.
00460E5C	BB 0B A0 00 DE 0B A0 00 EF 0B A0 00 00 0C A0 00	ñ0á.i0á.00á.00á.
00460E6C	12 0C A0 00 22 0C A0 00 33 0C A0 00 52 0C A0 00	00á."0á.00á.00á.
00460E7C	6C 0C A0 00 7D 0C A0 00 8E 0C A0 00 9F 0C A0 00	l0á.J0á.00á.00á.
00460E8C	F7 A8 B1 76 00 00 00 00 C8 74 F8 72 73 66 F9 72	~0á...00á.00á.00á.
00460E9C	87 72 F8 72 43 80 F8 72 67 37 F9 72 FB 41 F9 72	Gr°rC°r97°r°rA°r
00460EAC	67 83 F8 72 90 53 F8 72 00 00 00 00 CE 00 37 76	gá°rES°r...00á.7v
00460EBC	7C 86 37 76 B0 86 37 76 33 25 36 76 1E 31 36 76	!87v0087v326v16v
00460ECC	D8 7C 37 76 89 C2 37 76 CD 46 38 76 CE EE 36 76	i!7v007v°F8v°°6v
00460EDC	00 00 00 00 48 D0 4C 77 9C CB 4D 77 CC 42 4F 77	...H\$Lw6FmW°°B0w
00460EEC	2C D0 4C 77 DA F6 4C 77 73 33 50 77 10 64 4D 77	.\$Lw°°Lw\$3Pw°°dMw

Todas esas entradas que van a la seccion 0Axxxxxx en mi maquina son entradas redireccionadas, van a codigo creado por el despachador, que si el programa arrancara desde el OEP no existiria, por lo cual sigamos subiendo para ver si hallamos el INICIO de la IAT.

Address	Hex dump	ASCII
0046099C	C1 00 9F 00 D2 00 9F 00 E4 00 9F 00 F4 00 9F 00	±.f.
004609AC	05 01 9F 00 24 01 9F 00 3E 01 9F 00 4F 01 9F 00	±0f.
004609BC	60 01 9F 00 71 01 9F 00 7F 01 9F 00 8E 01 9F 00	'0f.
004609CC	90 01 9F 00 BD 01 9F 00 C8 01 9F 00 EE 01 9F 00	00f.
004609DC	FF 01 9F 00 10 02 9F 00 22 02 9F 00 32 02 9F 00	0f.
004609EC	43 02 9F 00 62 02 9F 00 7C 02 9F 00 8D 02 9F 00	C0f.
004609FC	9E 02 9F 00 AF 02 9F 00 BD 02 9F 00 CC 02 9F 00	×0f.
00460A0C	0B 02 9F 00 FB 02 9F 00 09 03 9F 00 2C 03 9F 00	00f.
00460A1C	3D 03 9F 00 4E 03 9F 00 60 03 9F 00 70 03 9F 00	±0f.
00460A2C	81 03 9F 00 A0 03 9F 00 BA 03 9F 00 CB 03 9F 00	U0f.
00460A3C	DC 03 9F 00 ED 03 9F 00 FB 03 9F 00 0A 04 9F 00	±0f.
00460A4C	19 04 9F 00 39 04 9F 00 47 04 9F 00 6A 04 9F 00	↓0f.
00460A5C	7B 04 9F 00 8C 04 9F 00 9E 04 9F 00 AE 04 9F 00	C0f.
00460A6C	BF 04 9F 00 DE 04 9F 00 F8 04 9F 00 09 05 9F 00	70f.
00460A7C	1A 05 9F 00 2B 05 9F 00 39 05 9F 00 48 05 9F 00	7±0f.
00460A8C	57 05 9F 00 77 05 9F 00 85 05 9F 00 A8 05 9F 00	W0f.
00460A9C	B9 05 9F 00 CA 05 9F 00 DC 05 9F 00 EC 05 9F 00	W±0f.
00460AAC	FD 05 9F 00 1C 06 9F 00 36 06 9F 00 47 06 9F 00	±0f.
00460ABC	58 06 9F 00 69 06 9F 00 77 06 9F 00 86 06 9F 00	×0f.
00460ACC	95 06 9F 00 B5 06 9F 00 C3 06 9F 00 E6 06 9F 00	0±0f.
00460ADC	F7 06 9F 00 08 07 9F 00 1A 07 9F 00 2A 07 9F 00	±0f.
00460AEC	3B 07 9F 00 5A 07 9F 00 74 07 9F 00 85 07 9F 00	±.f.
00460AFC	96 07 9F 00 A7 07 9F 00 B5 07 9F 00 C4 07 9F 00	±.f.
00460B0C	D3 07 9F 00 F3 07 9F 00 01 08 9F 00 24 08 9F 00	E±.f.
00460B1C	35 08 9F 00 46 08 9F 00 58 08 9F 00 68 08 9F 00	50f.
00460B2C	79 08 9F 00 98 08 9F 00 B2 08 9F 00 C3 08 9F 00	U0f.
00460B3C	04 08 9F 00 E5 08 9F 00 F3 08 9F 00 02 09 9F 00	E0f.
00460B4C	11 09 9F 00 31 09 9F 00 3F 09 9F 00 62 09 9F 00	±.f.
00460B5C	73 09 9F 00 84 09 9F 00 96 09 9F 00 A6 09 9F 00	s.f.
00460B6C	B7 09 9F 00 D6 09 9F 00 F0 09 9F 00 01 0A 9F 00	±.f.
00460B7C	12 0A 9F 00 23 0A 9F 00 31 0A 9F 00 40 0A 9F 00	±.f.
00460B8C	4F 0A 9F 00 6F 0A 9F 00 7D 0A 9F 00 A0 0A 9F 00	0.f.
00460B9C	B1 0A 9F 00 C2 0A 9F 00 D4 0A 9F 00 C0 4B 0F 77	±.f.
00460BAC	3B 4C 0F 77 94 A5 11 77 59 4B 0F 77 82 4E 0F 77	±.f.
00460BBC	98 D4 11 77 9B 50 0F 77 4F 50 0F 77 10 50 0F 77	±.f.
00460BCC	3F 50 0F 77 D9 66 0F 77 50 48 0F 77 55 4C 0F 77	±.f.
00460BDC	C2 4B 0F 77 95 D2 11 77 80 5D 15 77 00 00 00 00	±.f.
00460BEC	00 00 A7 00 11 00 A7 00 22 00 A7 00 33 00 A7 00	±.f.
00460BFC	00 00 A0 00 11 00 A0 00 22 00 A0 00 33 00 A0 00	±.f.

Vemos que luego hay algunas entradas que van a dlls, y luego mas arriba entradas que van a la seccion 9Fxxxx otra seccion creada por el desempacador.

Address	Disassembly	Comment
0042DB9B	CALL DWORD PTR DS:[460AC0]	DS:[00460AC0]=009F0669
0042DBBF	CALL DWORD PTR DS:[460AC0]	DS:[00460AC0]=009F0669
0043070A	CALL DWORD PTR DS:[460AC0]	DS:[00460AC0]=009F0669
00430731	CALL DWORD PTR DS:[460AC0]	DS:[00460AC0]=009F0669
0043170F	CALL DWORD PTR DS:[460AC0]	DS:[00460AC0]=009F0669
00431752	CALL DWORD PTR DS:[460AC0]	DS:[00460AC0]=009F0669
00433301	CALL DWORD PTR DS:[460AC0]	DS:[00460AC0]=009F0669
004342EE	CALL DWORD PTR DS:[460AC0]	DS:[00460AC0]=009F0669
00434336	CALL DWORD PTR DS:[460AC0]	DS:[00460AC0]=009F0669
00434391	CALL DWORD PTR DS:[460AC0]	DS:[00460AC0]=009F0669
0043440D	CALL DWORD PTR DS:[460AC0]	DS:[00460AC0]=009F0669
00434B04	CALL DWORD PTR DS:[460AC0]	DS:[00460AC0]=009F0669
00434B33	CALL DWORD PTR DS:[460AC0]	DS:[00460AC0]=009F0669
004355E0	MOV EBP,DWORD PTR DS:[460AC0]	DS:[00460AC0]=009F0669
0043C0A4	CALL DWORD PTR DS:[460AC0]	DS:[00460AC0]=009F0669
0043DB25	CALL DWORD PTR DS:[460AC0]	DS:[00460AC0]=009F0669
0044668D	MOV EDI,DWORD PTR DS:[460AC0]	DS:[00460AC0]=009F0669
004466D3	MOV EDI,DWORD PTR DS:[460AC0]	DS:[00460AC0]=009F0669

Ademas comprobando, vemos que dichas entradas tienen referencias, por lo cual, son entradas de la IAT, sigamos subiendo.

Address	Hex dump	ASCII
004607BC	62 28 06 00 80 28 06 00 70 28 06 00 00 00 00 00	b(.,.(.p(.,....
004607CC	0C 28 06 00 FA 2A 06 00 E8 2A 06 00 1E 28 06 00	.+.,.+.b+.+.+
004607DC	08 2A 06 00 C6 2A 06 00 AE 2A 06 00 92 2A 06 00	i+.+.+.+.+.+.+.+
004607EC	72 2A 06 00 BC 28 06 00 A8 28 06 00 92 28 06 00	r+.+.+.+.+.+.+.+
004607FC	78 28 06 00 60 28 06 00 4C 28 06 00 2E 28 06 00	x+.+.+.+.L+.+.+
0046080C	00 00 00 00 00 00 00 80 00 00 00 00 F0 6B DA 77C.....-k rw
0046081C	1B 76 DA 77 F4 EA DA 77 E7 EB DA 77 83 78 DA 77	+v rw 0 rw 0 rw 0 rw
0046082C	00 00 00 00 DD 15 C5 58 2E BD C3 58 00 00 00 00!\$+X.c!X....
0046083C	00 00 A1 00 11 00 A1 00 22 00 A1 00 33 00 A1 00	...i.4.i."i.3.i.
0046084C	41 00 A1 00 50 00 A1 00 5F 00 A1 00 7F 00 A1 00	A.i.P.i..i.Δ.i.
0046085C	80 00 A1 00 B0 00 A1 00 C1 00 A1 00 D2 00 A1 00	i.i.::i.i.i.È.i.
0046086C	E4 00 A1 00 F4 00 A1 00 05 01 A1 00 24 01 A1 00	\$.i.¶.i.+\$i.+\$i.
0046087C	3E 01 A1 00 4F 01 A1 00 60 01 A1 00 71 01 A1 00	>0i.00i.'0i.q0i.
0046088C	7F 01 A1 00 8E 01 A1 00 9D 01 A1 00 BD 01 A1 00	Δ0i.Δ0i.00i.c0i.
0046089C	CB 01 A1 00 EE 01 A1 00 FF 01 A1 00 10 02 A1 00	π0i.'0i.'0i.00i.
004608AC	28 02 A1 00 32 02 A1 00 43 02 A1 00 62 02 A1 00	"0i.20i.C0i.b0i.
004608BC	7C 02 A1 00 8D 02 A1 00 9E 02 A1 00 AF 02 A1 00	!0i.i0i.x0i.>0i.
004608CC	BD 02 A1 00 CC 02 A1 00 DB 02 A1 00 FB 02 A1 00	c0i.f0i.00i.'0i.
004608DC	09 03 A1 00 2C 03 A1 00 3D 03 A1 00 4E 03 A1 00	.0i..0i.=0i.N0i.
004608EC	60 03 A1 00 70 03 A1 00 81 03 A1 00 A0 03 A1 00	'0i.p0i.ü0i.ä0i.
004608FC	BA 03 A1 00 CB 03 A1 00 DC 03 A1 00 ED 03 A1 00	0i.π0i.00i.ÿ0i.
0046090C	FB 03 A1 00 0A 04 A1 00 19 04 A1 00 39 04 A1 00	'0i..0i.40i.90i.
0046091C	47 04 A1 00 6A 04 A1 00 7B 04 A1 00 8C 04 A1 00	G0i.j0i.c0i.i0i.
0046092C	9E 04 A1 00 AE 04 A1 00 BF 04 A1 00 DE 04 A1 00	x0i.<0i.70i.i0i.
0046093C	F8 04 A1 00 09 05 A1 00 1A 05 A1 00 2B 05 A1 00	00i..0i.+0i.+0i.
0046094C	39 05 A1 00 48 05 A1 00 57 05 A1 00 77 05 A1 00	90i.H0i.W0i.w0i.
0046095C	85 05 A1 00 A8 05 A1 00 B9 05 A1 00 CA 05 A1 00	ä0i.ä0i.j0i.ä0i.
0046096C	DC 05 A1 00 EC 05 A1 00 00 00 9F 00 11 00 9F 00	00i.00i...f.4.f.
0046097C	22 00 9F 00 33 00 9F 00 41 00 9F 00 50 00 9F 00	".f.3.f.A.f.P.f.
0046098C	5F 00 9F 00 7F 00 9F 00 8D 00 9F 00 B0 00 9F 00	..f.Δ.f.i.f.::f.
0046099C	C1 00 9F 00 D2 00 9F 00 E4 00 9F 00 F4 00 9F 00	..f.È.f.\$.¶.f.
004609AC	05 01 9F 00 24 01 9F 00 3E 01 9F 00 4F 01 9F 00	+0f.\$0f.>0f.00f.
004609BC	60 01 9F 00 71 01 9F 00 7F 01 9F 00 8E 01 9F 00	'0f.q0f.Δ0f.Δ0f.
004609CC	9D 01 9F 00 BD 01 9F 00 CB 01 9F 00 EE 01 9F 00	00f.c0f.π0f.'0f.
004609DC	FF 01 9F 00 10 02 9F 00 22 02 9F 00 32 02 9F 00	0f.00f."0f.20f.
004609EC	43 02 9F 00 62 02 9F 00 7C 02 9F 00 8D 02 9F 00	C0f.b0f.'0f.i0f.
004609FC	9E 02 9F 00 AF 02 9F 00 BD 02 9F 00 CC 02 9F 00	x0f.>0f.c0f.f0f.
00460A0C	DB 02 9F 00 FB 02 9F 00 09 03 9F 00 2C 03 9F 00	00f.'0f..0f..0f.
00460A1C	3D 03 9F 00 4E 03 9F 00 60 03 9F 00 70 03 9F 00	=0f.N0f.'0f..0f.
00460A2C	81 03 9F 00 A0 03 9F 00 BA 03 9F 00 CB 03 9F 00	ü0f.ä0f. 0f.π0f.
00460A3C	DC 03 9F 00 ED 03 9F 00 FB 03 9F 00 0A 04 9F 00	00f.ÿ0f.'0f..0f.
00460A4C	19 04 9F 00 39 04 9F 00 47 04 9F 00 6A 04 9F 00	+0f.90f.G0f.j0f.
00460A5C	7B 04 9F 00 8C 04 9F 00 9F 04 9F 00 0F 04 9F 00	c0f.i0f.x0f.00f.

Luego vemos entradas que en mi maquina van a A1xxxx que es otra seccion creada por el packer

00540000	00103000				Map	R	-	R	-	
00650000	00106000				Map	R	E	R	E	
00970000	00001000				Priv	RW		RW		
009F0000	00002000				Priv	RW		RW		
00A00000	00002000				Priv	RW		RW		
00A10000	00001000				Priv	RW		RW		
00A20000	00004000				Priv	RW		RW		
00A30000	00003000				Map	R		R		\Dev
00A40000	00004000				Priv	RW		RW		
00A50000	00003000				Priv	RW		RW		
00A60000	00002000			N	Map	R		R		

y que ademas si busco tienen referencias.

Address	Disassembly	Comment
00404F8E	CALL DWORD PTR DS:[460894]	DS:[00460894]=00A1019D

Por lo cual son entradas de la IAT y si sigo subiendo veo que ya llego a

004607FC	78 2B 06 00 B0 2B 06 00 4C 2B 06 00 2E 2B 06 00	x+.+.+.L.
0046080C	00 00 00 00 00 00 00 80 00 00 00 F0 6B DA 77C.
0046081C	1B 76 DA 77 F4 EA DA 77 E7 EB DA 77 83 78 DA 77	+v rw 0 rw 0 rw 0 rw
0046082C	00 00 00 00 DD 15 C5 58 2E BD C3 58 00 00 00 00!\$+X.c!X....
0046083C	00 00 A1 00 11 00 A1 00 22 00 A1 00 33 00 A1 00	...i.4.i."i.3.i.
0046084C	41 00 A1 00 50 00 A1 00 5F 00 A1 00 7F 00 A1 00	A.i.P.i..i.Δ.i.
0046085C	80 00 A1 00 B0 00 A1 00 C1 00 A1 00 D2 00 A1 00	i.i.::i.i.i.È.i.

Donde hay entradas que van a una dll, luego la separacion y luego mas arriba ninguna entrada tiene referencias.

Address	hex dump	ASCII
004607F8	92 2B 06 00 78 2B 06 00 60 03 06 00 4C 2B 06 00	Æ+..x+..'+..L+..
00460808	2E 2B 06 00 00 00 00 00 00 00 00 00 80 00 00 00	..+..+..+..+..+..
00460818	F0 6B 0A 77 18 76 0A 77 F4 EA 0A 77 E7 EB 0A 77	-k rw+vrw10 rw0 rw
00460828	83 78 0A 77 00 00 00 00 DD 15 C5 58 2E 80 C3 58	âx rw...!\$+X.c!X
00460838	00 00 00 00 00 00 00 00 11 00 01 00 22 00 01 00l..l..l..
00460848	33 00 01 00 41 00 01 00 50 00 01 00 5F 00 01 00	3..l..A..l..P..l..
00460858	7F 00 01 00 80 00 01 00 B0 00 01 00 C1 00 01 00	â..l..l..l..l..l..
00460868	D2 00 01 00 E4 00 01 00 F4 00 01 00 05 01 01 00	Ê..l..s..l..l..#0i.
00460878	24 01 01 00 3E 01 01 00 4F 01 01 00 60 01 01 00	\$0i.>0i.00i.'0i.
00460888	71 01 01 00 7F 01 01 00 8E 01 01 00 90 01 01 00	q0i.00i.00i.00i.
00460898	BD 01 01 00 CB 01 01 00 EE 01 01 00 FF 01 01 00	c0i.π0i.'0i. 0i.
004608A8	10 02 01 00 22 02 01 00 32 02 01 00 43 02 01 00	00i."0i.20i.C0i.
004608B8	62 02 01 00 7C 02 01 00 8D 02 01 00 9E 02 01 00	b0i.!0i.!0i.x0i.
004608C8	AF 02 01 00 BD 02 01 00 CC 02 01 00 DB 02 01 00	>0i.c0i.l0i.00i.
004608D8	FB 02 01 00 09 03 01 00 2C 03 01 00 3D 03 01 00	'0i.'0i.'0i.'0i.
004608E8	4E 03 01 00 60 03 01 00 70 03 01 00 81 03 01 00	N0i.'0i.'0i.'0i.
004608F8	A0 03 01 00 BA 03 01 00 CB 03 01 00 DC 03 01 00	â0i. 0i.π0i.00i.
00460908	ED 03 01 00 FB 03 01 00 0A 04 01 00 19 04 01 00	ÿ0i.'0i.'0i.'0i.
00460918	39 04 01 00 47 04 01 00 6A 04 01 00 7B 04 01 00	90i.G0i.j0i.C0i.
00460928	8C 04 01 00 9F 04 01 00 AF 04 01 00 BF 04 01 00	i0i.x0i.<0i.'0i.

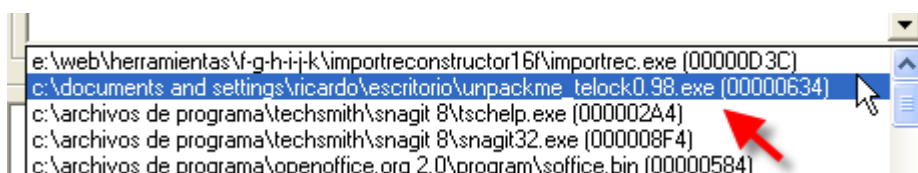
Como en el CRUNCH el inicio de la IAT es 460818 el largo 710 y el OEP es 4271B0, restandole la image base.

OEP= 271B0

INICIO o RVA= 60818

LARGO= 710

Por lo tanto abramos el IMP REC sin cerrar este OLLYDBG.



Y coloquemosle los valores que hallamos.

IAT Infos needed

OEP 000271B0

IAT AutoSearch

RVA 00060818

Size 0000710

Load Tree

Save Tree

Get Imports

Ahora apreto GET IMPORTS

+	advapi32.dll	FTThunk:00060818	NbFunc:5 (decimal:5)	valid:YES
+	comctl32.dll	FTThunk:00060830	NbFunc:2 (decimal:2)	valid:YES
+	?	FTThunk:0006083C	NbFunc:EB (decimal:235)	valid:NO
+	?	FTThunk:00060BEC	NbFunc:A9 (decimal:169)	valid:NO
+	winspool.drv	FTThunk:00060E94	NbFunc:8 (decimal:8)	valid:YES
+	comdlg32.dll	FTThunk:00060EB8	NbFunc:9 (decimal:9)	valid:YES
+	ole32.dll	FTThunk:00060EE0	NbFunc:10 (decimal:16)	valid:YES
+	oledlg.dll	FTThunk:00060F24	NbFunc:1 (decimal:1)	valid:YES

Como vemos el IMP REC detecta que hay entradas redireccionadas y nos pone NO en algunas, miremoslas apretando SHOW INVALIDS que nos mostrara las invalidas.

Imported Fur	
rva:0006083C	ptr:00A10000
rva:00060840	ptr:00A10011
rva:00060844	ptr:00A10022
rva:00060848	ptr:00A10033
rva:0006084C	ptr:00A10041
rva:00060850	ptr:00A10050
rva:00060854	ptr:00A1005F
rva:00060858	ptr:00A1007F
rva:0006085C	ptr:00A1008D

Vemos que el IMP REC nos muestra lo mismo que vimos en la IAT entradas que no van a ninguna api, y que van alli en la imagen, a direcciones de secciones creadas por el packer o codigo propio del mismo.

Por supuesto no vamos a tracear todas esas entradas incorrectas a mano, hay varios metodos para repararlas, el IMP REC trae algunas posibilidades, otros metodos pueden hacerse a mano pero no traceando jeje

Todos estos metodos los veremos en la parte 37, para que tengan bien claro el tema de las apis redireccionadas repasen bien esta parte, asi en la parte siguiente vemos metodos para repararlas. Vemos que asi como esta, no podemos reparar un dumpeado, pues debe tener el IMP REC todo YES o sea todas las entradas deben apuntar a apis de dlls, y no a entradas redireccionadas ni codigo choto y eso debemos arreglarlo nosotros.

Hasta la parte 37
Ricardo Narvaja
17/03/06