

ESTUDIO DE PROTECCIONES BASICO PARA PRINCIPIANTES

(También llamado cracking)

Impartido por Ratón (Nivel principiante)

Nota

Cada capitulo ira acompañado de su crackme correspondiente.No facilitare paginas de donde bajarse herramientas ni enlaces a paginas de crackers, la intención es que busquéis en Internet todo lo necesario. Seguro que encontráis mas paginas de herramientas, tutoriales y utilidades relacionadas con este tema que las que yo pueda deciros.

Con esto solo quiero fomentar vuestro interés, además se que la búsqueda os proporcionara gratas sorpresas.

A todos un saludo.

Capitulo 6

Victima

"Crackmes" AcitiviWonder (3) de Ratón

Herramientas

Olly Debugger.

Hackman 7.03

Visual Basic reformer

Exescope. PE Explorer o Resource Hacker también servirán para el mismo propósito.

Instinto

Objetivo

Conocer otro tipo de protección.

Ver conversión hexadecimal - ASCII .

Pasar un rato entretenidos.

Al ataque

Este tipo de protección lo encontramos algunas veces con programas share que tienen alguna característica deshabilitada hasta que paguemos el precio del registro (parece una película de Brian de Palma: Scarcracker el precio del registro), como por ejemplo el botón Save o guardar. En definitiva este capitulo va de activar o habilitar botones.

Utilizaremos editores de recursos [Exescope](#), un editor hexa (y algo mas) [Hackman 7.03](#) y [Olly](#)

He creado 3 mini programas para ver este tema, cada uno en un lenguaje de programación diferente (C++ Delphi y Visual Basic)

Empezaremos por el crackme escrito utilizando Borland c+ builder.

Pasamos por esta vez de analizarlos con Peid pues no les puse protección.

ActiviWonder1 CrackmeC.exe

Al abrirlo vemos una ventana con 2 botones desactivados y en la barra del menú 2 opciones desactivadas, pulsamos Ayuda para ver que va la cosa y aparecen las instrucciones, ya sabemos de qué va esto.



Para hacer esto vamos a utilizar un editor de recursos: Exescope
Empecemos por los botones del formulario y dejemos los del menú superior para después (por llevar la contraria al autor nada mas).

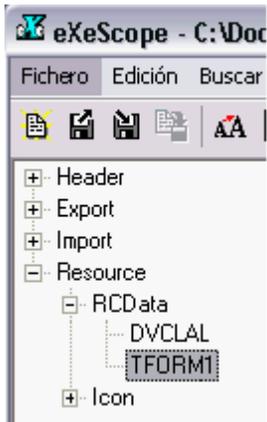
Vemos que el Caption del primer botón es Activame (explicado en el capítulo de DeDe)

Abrimos Exescope y cargamos el programa, nos fijamos en la parte de la izquierda y desplegamos [RCData](#).

También podríamos buscar en el apartado [Dialogs](#) si lo hubiera, quedaros con esto.

TFORM1 hace referencia a un formulario, en este caso el programa solo tiene un formulario que es la ventana principal del mismo, la que aparece al ejecutarlo.

Si hacemos click encima de TFORM1 vemos que en la ventana derecha se rellena con texto



Vamos buscando por el texto has encontrar la palabra Actívame (el caption del botón que queremos habilitar)

Cuando llegamos vemos que esta dentro de object Button vemos que nos indica que es un botón (TButton) y que su propiedad Caption es Actívame

También vemos el ancho - Width, largo- Height y posición- Left y Top en el form y una propiedad que en este caso es la que nos interesa: **Enabled**

Enabled = False

Traducción al sioux = Enabled – activado, habilitado False = falso Valor 00

Enabled tiene su antónimo que es Disabled = desactivado, deshabilitado

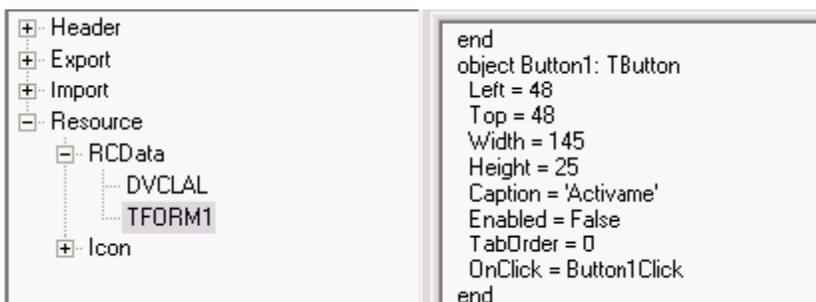
Así mismo la palabra contraria a False es True = verdadero Valor 01

La propiedad Enable vemos que esta False, o sea que Activado = falso (00), como estamos en la propiedad Enable que pertenece al Button1 el cual tiene el Caption Actívame deducimos:

1-Que no podemos pulsar el botón Actívame porque al hacer este programa el capullo que lo hizo dejo sin activar este botón dándole a la propiedad Enable el valor False.

2-Que a lo mejor si cambiamos False por True (01) habilitamos el botón.

(Los que sepan programar se estarán desc#j#nand# con la “explicación” anterior de los valores booleanos, pero ni esto es un curso de programación ni el que lo imparte va mucho mas allá de los tres programas que acompañan a este capítulo).



Vamos a hacerlo, es tan fácil como escribir True en el lugar que ocupa false

```
object Button1: TButton
  Left = 48
  Top = 48
  Width = 145
  Height = 25
  Caption = 'Activame'
  Enabled = False
  TabOrder = 0
  OnClick = Button1Click
end
```

```
object Button1: TButton
  Left = 48
  Top = 48
  Width = 145
  Height = 25
  Caption = 'Activame'
  Enabled = True
  TabOrder = 0
  OnClick = Button1Click
end
```

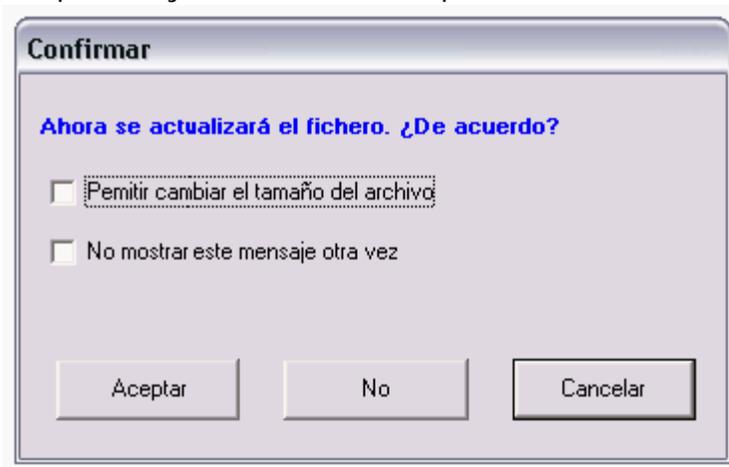
Una vez que escribimos True guardamos el cambio en ejecutable con guardar actualización



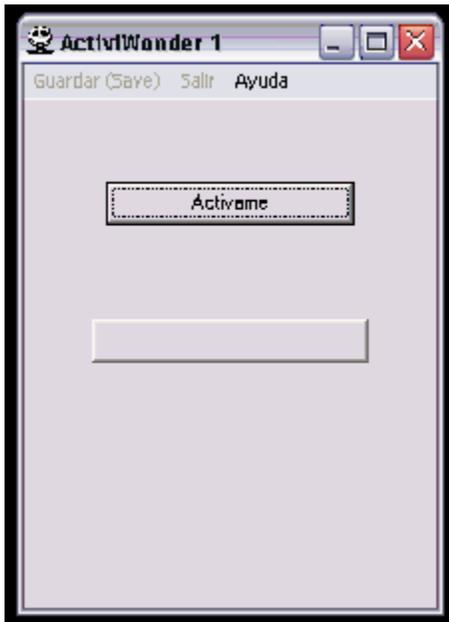
Nos aparece esta ventana y [desmarcamos Permitir cambiar el tamaño del ejecutable](#).

Siempre intentaremos alterar lo menos posible los ejecutables pues algunos programas al detectar el cambio no funcionarían.

Aceptamos y Cerramos Exescope



Ejecutamos el crackme y vemos el botón activado

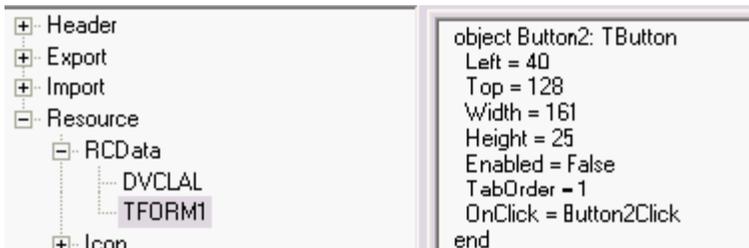


Lo pulsamos



Vamos a por el segundo botón, este no tiene Caption, no pone nada, pero como somos astutos sabremos encontrarlo

Cargamos otra vez el crackme en el Exescope y buscamos un botón (TButton) sin Caption
Encontramos el Button2 justo debajo del anterior.



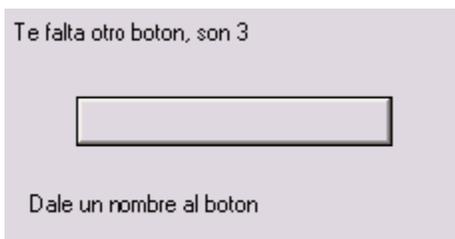
Vemos la diferencia con el botón 1, le falta la propiedad Caption este también tiene la propiedad Enable = False pues la cambiamos y guardamos el cambio en el ejecutable

```

object Button1: TButton
  Left = 48
  Top = 48
  Width = 145
  Height = 25
  Caption = 'Activame'
  Enabled = True
  TabOrder = 0
  OnClick = Button1Click
end
object Button2: TButton
  Left = 40
  Top = 128
  Width = 161
  Height = 25
  Enabled = True
  TabOrder = 1
  OnClick = Button2Click
end

```

Cerramos Exescope y abrimos el crackme vemos el botón 2 activado, lo pulsamos y nos salen dos avisos, hagámosles caso



Vamos a darle un nombre al botón, vamos a escribir el Caption del botón

Cerramos el crackme y abrimos Exescope

Buscamos el botón dos y le añadimos la propiedad Caption y el valor que queramos

```

object Button2: TButton
  Left = 40
  Top = 128
  Width = 161
  Height = 25
  Caption = 'PEPE'
  Enabled = True
  TabOrder = 1
  OnClick = Button2Click
end

```

Justo debajo está el boton3 con el caption Soy invisible (así se las ponían a Felipe 2°)

Vemos una nueva propiedad: Visible = False o sea que no se ve ;)

No creo que haga falta explicar mucho: lo cambiamos a True

```

object Button3: TButton
  Left = 40
  Top = 216
  Width = 105
  Height = 33
  Caption = 'Soy invisible'
  TabOrder = 2
  Visible = False
  OnClick = Button3Click
end

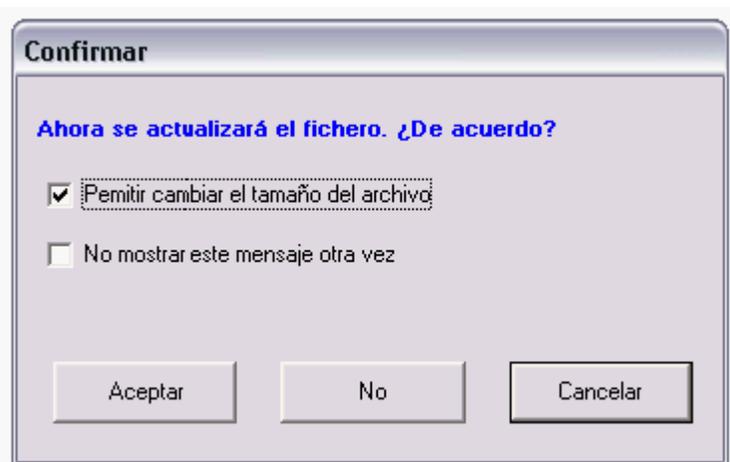
```

Guardamos los cambios sin permitir cambiar el tamaño del archivo y...

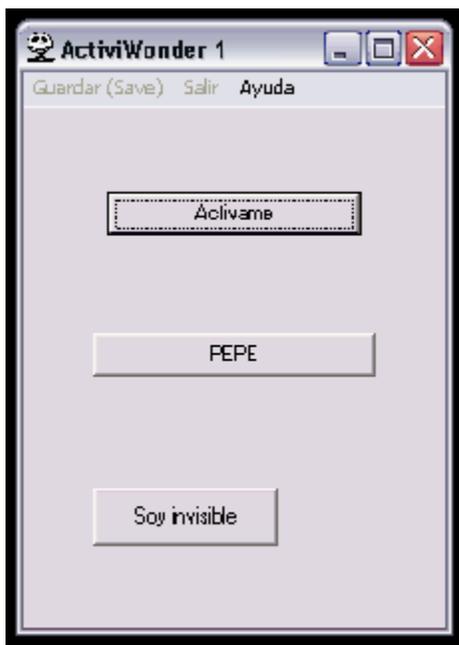


No podemos (claro no lo hemos permitido) y además al añadir una propiedad mas al botón 2 el ejecutable ha engordado unos bits tened esto en cuenta en el futuro al trabajar con editores de recursos.

Vamos a volver intentarlo a ver si este crackme nos funciona cambiándole el tamaño
Marcamos esta vez la casilla



Guardamos cambios y cerramos Exescope
Abrimos el crackme y...



Pulsadlo si queréis

Ahora vamos a por el menú.

Lo buscamos en el Exescope, ya sabéis como no ¿?

Una vez que lo encontremos observamos las propiedades y los Captions y cambiamos lo que necesitamos igual que hiciéramos con los otros botones y ya tenemos todo habilitado

```
object MainMenu1: TMainMenu
  Left = 200
  Top = 8
  object Guardarsave1: TMenuItem
    Caption = 'Guardar (Save)'
    Enabled = False
    OnClick = Guardarsave1Click
  end
  object Salir2: TMenuItem
    Caption = 'Salir'
    Enabled = False
    OnClick = Salir2Click
  end
  object Ayuda1: TMenuItem
    Caption = 'Ayuda'
    OnClick = Ayuda1Click
  end
end
```

Cuando tengáis habilitados todos los botones guardad los cambios y [quedaros con una copia de este crackme habilitado](#), aparte claro de conservar el original

Otra manera de hacerlo, un poco más liosa pero nos remitirá a tomar contacto con un editor hexadecimal y aprender cosas nuevas

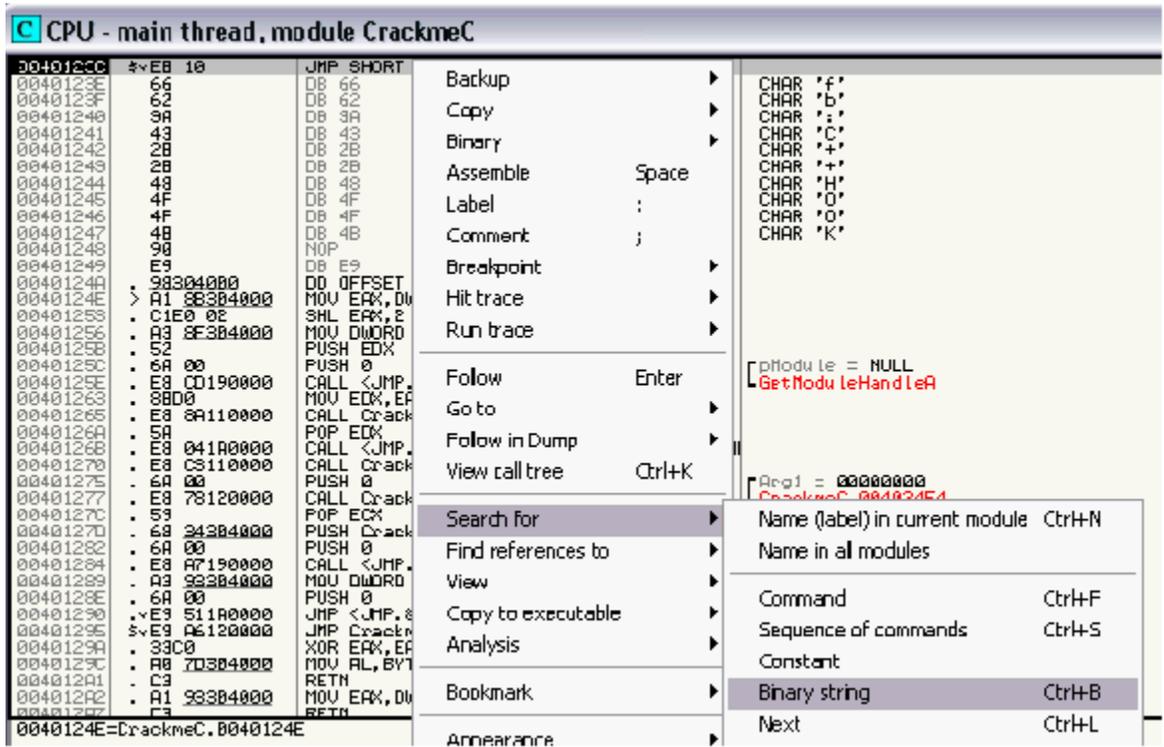
Se pueden hacer estos cambios en el código con Ollly pero os voy a enseñar como y por que no diciendo cambia esto por esto y ya está

Lo voy a hacer así para que veáis como a veces hay que buscarse la vida y para que experimentéis con más opciones de Olly

Esta parte es un poco complicada así que si no lo entendéis a la primera tranquilos.

Abrimos el Olly con el crackme e intentamos buscar primero el botón `Actívame` para hacerlo en el mismo orden que antes

Con el botón derecho Control +B o Search for - Binary String



Escribimos la palabra o cadena de caracteres a buscar

Importante: fijaros en la parte donde pone HEX en esta ventana vemos una serie de números y letras de dos en dos

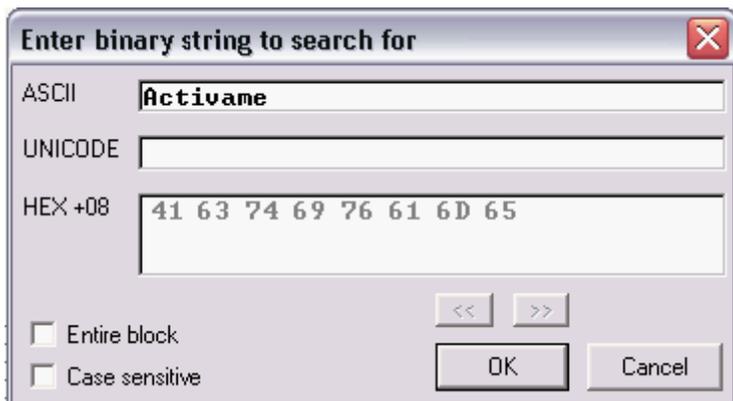
Es la traducción a hexadecimal de los caracteres ASCII

Si miráis la tabla del capítulo 0 podréis comprobarlo

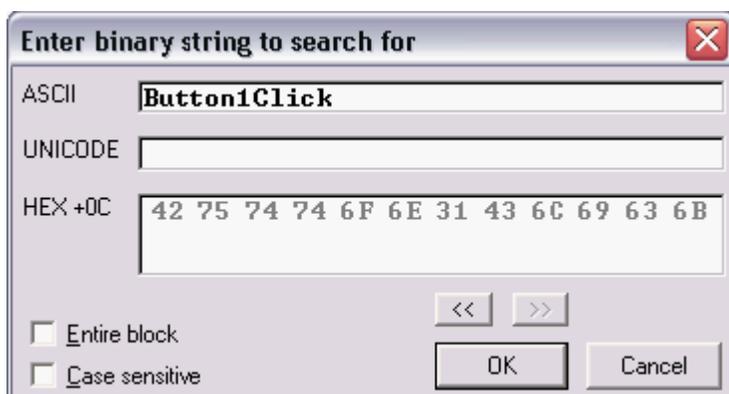
41-A 63-c 74-t 69-i 76-v

61-a 6D-m 65-e

Pulsamos ok pero no encontramos la cadena `actívame` por ningún lado probaremos con otra

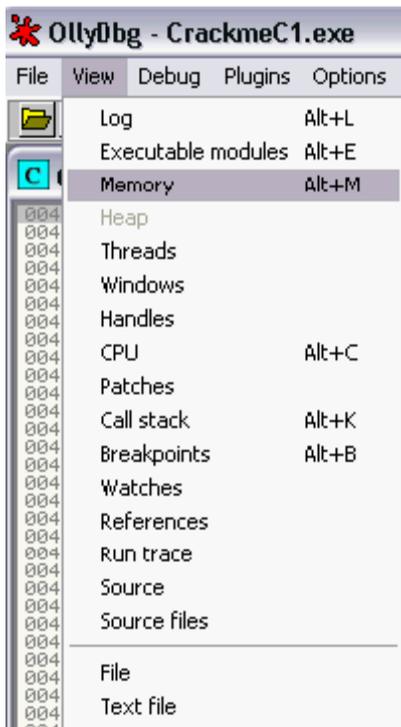


Fijaros que siempre hace la conversión ASCII - Hexa
Buscamos por ejemplo el evento click del botón 1 (Activame)



Tampoco la encontramos, pero debe andar por algún lado, pues tanto el botón como sus propiedades y eventos pertenecen al programa, el botón es uno de los recursos incluidos en el crackme

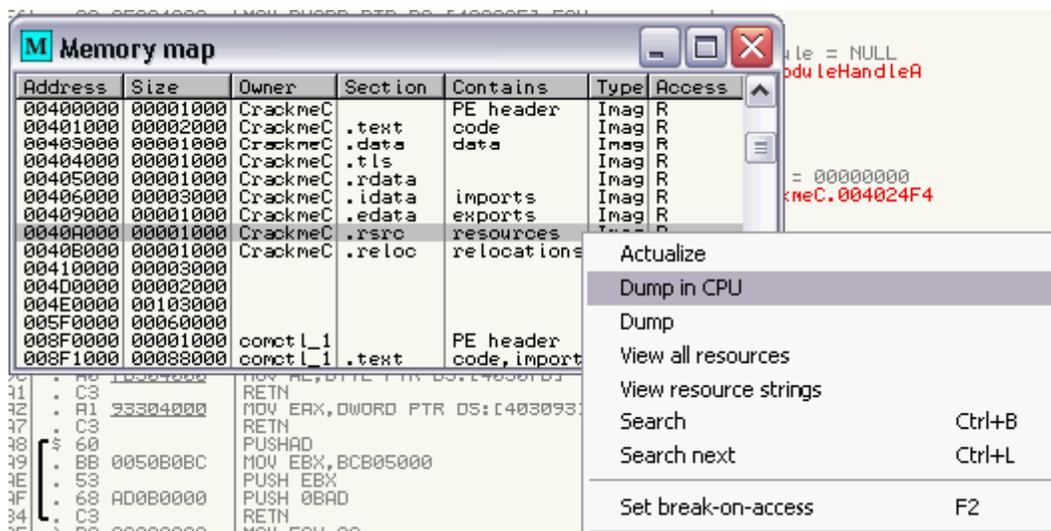
Vamos a buscarlo en la memoria
Pulsamos sobre View - Memory



Aparece esta ventana

Paso uno- estamos buscando los recursos del crackmeC.exe, pues buscamos el ejecutable en la ventana de la memoria lo vemos y vemos la palabra resources (recursos)

Paso dos- Sobre la palabra resources click derecho y escogemos Dump in cpu (volcar en la cpu, vomita lo que lles dentro!!) vamos a ver si haciendo un volcado de la memoria conseguimos encontrar alguna de las cadenas del crackme que nos den pistas para poder activar los botones



Busco Button1Click (podría haber buscado Actívame o la cadena que realmente se necesita para habilitar el botón pero daré un rodeo, de momento se que si encuentro Button1Click es que el botón esta ahí que es lo que interesa ahora)

Address	Hex dump	UNICODE
0040A000	00 00 00 00 D4 74 30 30 00 00 00 00 00 00 00 00
0040A010	03 00 00 00 28 00 00 00 0A 00 00 00 00 40 00 00 00
0040A020	0E 00 00 00 60 00 00 00 00 00 00 00 00 D4 74 30 30
0040A030	00 00 00 00 00 00 01 00 01 00 00 00 00 78 00 00 00
0040A040	00 00 00 00 D4 74 30 30 00 00 00 00 00 02 00 00 00
0040A050	18 01 00 00 90 00 00 00 26 01 00 00 0A 00 00 00 00
0040A060	00 00 00 00 D4 74 30 30 00 00 00 00 00 01 00 00 00
0040A070	34 01 00 00 C0 00 00 00 00 00 00 00 00 D4 74 30 30
0040A080	00 00 00 00 00 00 01 00 0A 0C 00 00 00 D8 00 00 00
0040A090	00 00 00 00 D4 74 30 30 00 00 00 00 00 00 00 01 00
0040A0A0	00 00 00 00 E8 00 00 00 00 00 00 00 00 D4 74 30 30
0040A0B0	00 00 00 00 00 00 01 00 00 00 00 00 00 F8 00 00 00
0040A0C0	00 00 00 00 D4 74 30 30 00 00 00 00 00 00 00 01 00
0040A0D0	0A 0C 00 00 08 01 00 00 4B A1 00 00 E8 02 00 00 00
0040A0E0	00 00 00 00 00 00 00 00 30 A4 00 00 10 00 00 00 00
0040A0F0	00 00 00 00 00 00 00 00 4B A4 00 00 79 09 00 00 00
0040A100	00 00 00 00 00 00 00 00 B4 AD 00 00 14 00 00 00 00
0040A110	00 00 00 00 00 00 00 00 06 00 00 00 44 00 56 00 48 00
0040A120	4C 00 41 00 4C 00 06 00 54 00 46 00 4F 00 52 00 LAL↑TFOR
0040A130	4D 00 31 00 08 00 4D 00 41 00 49 00 4E 00 49 00 M↑MAINI
0040A140	43 00 4F 00 4E 00 00 00 2B 00 00 00 20 00 00 00 CON↓.

Enter binary string to search for

ASCII:

UNICODE:

HEX +0C:

Entire block

Case sensitive

<< >>

OK Cancel

Ahora si la encontró fijaros solo la vemos en hexa a la izquierda (sombreada)

Address	Hex dump	UNICODE
0040ABE6	42 75 74 74 6F 6E 31 43 6C 69 63 6B 00 00 07 54
0040ABF6	42 75 74 74 6F 6E 07 42 75 74 74 6F 6E 32 04 4C
0040AC06	65 66 74 02 28 03 54 6F 70 03 90 00 05 57 69 64
0040AC16	74 68 03 A1 00 06 48 65 69 67 68 74 02 19 07 45
0040AC26	6E 61 62 6C 65 64 08 08 54 61 62 4F 72 64 65 72
0040AC36	02 01 07 4F 6E 43 6C 69 69 6B 07 0C 42 76 74 74
0040AC46	6F 6E 32 43 6C 69 63 6B 00 00 07 54 42 75 74 74

Buscamos Button2Click y también esta

Address	Hex dump	UNICODE
0040ABC2	74 69 76 61 6D 65 07 45 6E 61 62 6C 65 64 08 08
0040ABD2	54 61 62 4F 72 64 65 72 02 00 07 4F 6E 42 6C 69
0040ABE2	63 68 07 0C 42 75 74 74 6F 6E 31 43 6C 69 63 6B
0040ABF2	00 00 07 54 42 75 74 74 6F 6E 07 42 75 74 74 6F
0040AC02	6E 32 04 4C 65 66 74 02 28 03 54 6F 70 03 90 00
0040AC12	05 57 69 64 74 68 03 A1 00 06 48 65 69 67 68 74
0040AC22	02 19 07 45 6E 61 62 6C 65 64 08 08 54 61 62 4F
0040AC32	72 64 65 72 02 01 07 4F 6E 43 6C 69 63 6B 07 0C
0040AC42	42 75 74 74 6F 6E 07 42 75 74 74 6F 6E 33 04 4C
0040AC52	42 75 74 74 6F 6E 07 42 75 74 74 6F 6E 33 04 4C
0040AC62	65 66 74 02 28 03 54 6F 70 03 D8 00 05 57 69 64
0040AC72	74 68 02 69 06 48 65 69 67 68 74 02 21 07 43 61

Lo mismo para el botón 3

Address	Hex dump	UNICODE
0040ACB4	42 75 74 74 6F 6E 33 43 6C 69 63 6B 00 00 09 54
0040ACC4	4D 61 69 6E 4D 65 6E 75 09 4D 61 69 6E 4D 65 6E
0040ACD4	75 31 04 4C 65 66 74 03 C8 00 03 54 6F 70 02 08
0040ACE4	00 09 54 4D 65 6E 75 49 74 65 6D 0C 47 75 61 72
0040ACF4	64 61 72 73 61 76 65 31 07 43 61 70 74 69 6F 6E
0040AD04	06 0E 47 75 61 72 64 61 72 20 28 53 61 76 65 29
0040AD14	07 45 6E 61 62 6C 65 64 08 07 4F 6E 43 6C 69 63
0040AD24	6B 07 11 47 75 61 72 64 61 72 73 61 76 65 31 43
0040AD34	6C 69 63 6B 00 00 09 54 4D 65 6E 75 49 74 65 6D
0040AD44	06 53 61 6C 69 72 32 07 43 61 70 74 69 6F 6E 06
0040AD54	05 53 61 6C 69 72 07 45 6E 61 62 6C 65 64 08 07
0040AD64	4F 6E 43 6C 69 63 68 07 0B 53 61 6C 69 72 32 43
0040AD74	6C 69 63 6B 00 00 09 54 4D 65 6E 75 49 74 65 6D
0040AD84	06 41 79 75 64 61 31 07 43 61 70 74 69 6F 6E 06

Llegados a este punto imaginamos que entre esa maraña de números y letras tiene que estar la clave para cambiar la propiedad Enabled del botón
 Como saber que valor cambiar, donde esta y cual es el valor que debemos darle ¿?
 Hackeamos y la lógica más aplastante entran en juego.

Empecé el capítulo por la parte más fácil para poder tener un crackme totalmente operativo en poco tiempo sin problemas, pues el Exescope nos hace todos los cambios sin apenas esfuerzo.

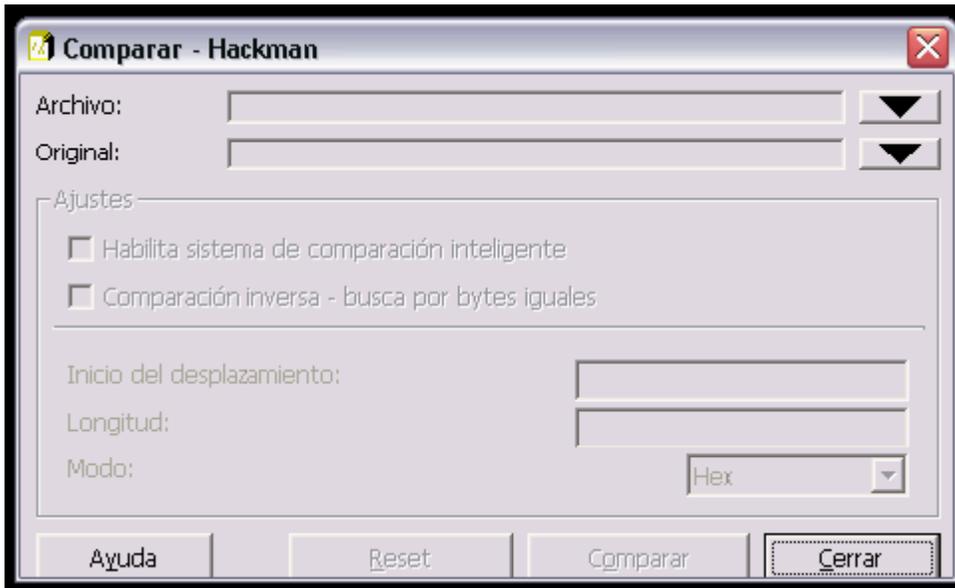
Si miramos en la carpeta donde se instalo Hackman vemos muchas herramientas
Nos interesa Compare.exe, porque para saber lo que tenemos que cambiar vamos a comparar el crackme original con el parcheado por nosotros con el Exescope y ver que es lo que cambia



Ejecutamos compare.exe y lo vemos claro (si os acordáis es parecido a lo que nos hacia el Patch FX para crear el nuestro propio parche) debemos introducir el original y el parcheado

Si no os funciona clickeando en las flechas negras haced dobleclick encima de las casillas vacías de archivo y original

Una vez se hayan cargado, sin marcar ninguna opción mas pulsad comparar

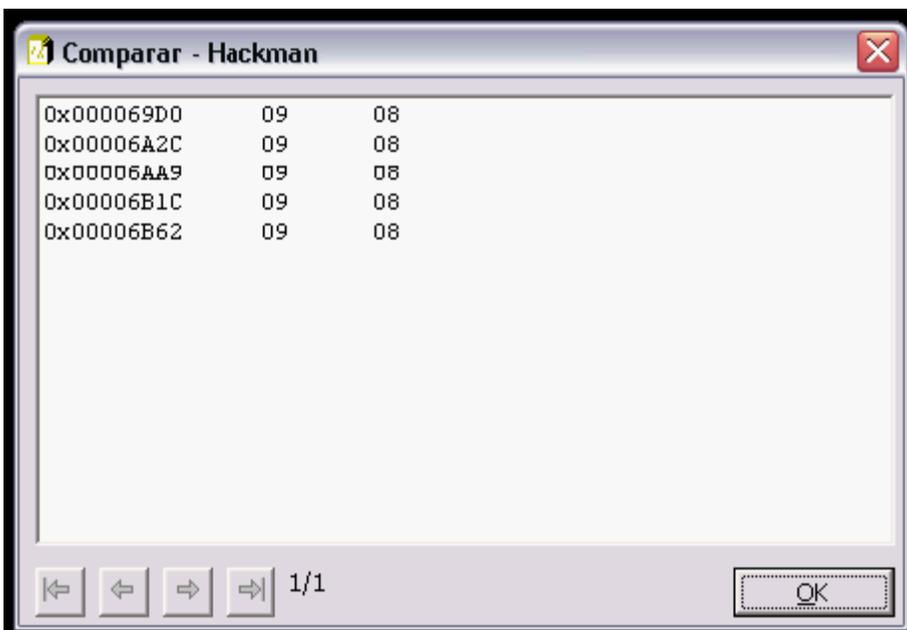


hay 5 diferencias entre el original y el parcheado, en uno pone 09 y en el otro el valor es 08
Tenemos 5 botones desactivados en el crackme

Esos botones están Enabled = false

Tenemos 5 botones activados en el crackme que parcheamos con Exescope Enabled = True

08 y 09 son la diferencia entre estar activado y desactivado, entre false y true



Tenemos datos de sobra para poder cambiar los saltos, abramos Ollly

Sigamos los pasos anteriores y hagamos el volcado de memoria en el Dump
Busquemos la palabra mágica Enable

Address	Hex dump	UNICODE
00400000	00 00 00 00 04 74 30 30 00 00 00 00 00 00 03 00	
00400010	03 00 00 00 20 00 00 00 0A 00 00 00 40 00 00 00B
00400020	0E 00 00 00 60 00 00 00 00 00 00 00 04 74 30 30K
00400030	00 00 00 00 00 00 01 00 01 00 00 00 78 00 00 00
00400040	00 00 00 00 04 74 30 30 00 00 00 00 02 00 00 00B
00400050	18 01 00 00 90 00 00 00 26 01 00 00 A8 00 00 00C
00400060	00 00 00 00 04 74 30 30 00 00 00 00 01 00 00 00B
00400070	34 01 00 00 00 00 00 00 00 00 00 00 04 74 30 30
00400080	00 00 00 00 00 00 01 00 0A 0C 00 00 08 00 00 00i
00400090	00 00 00 00 04 74 30 30 00 00 00 00 00 00 01 00
004000A0	00 00 00 00 E8 00 00 00 00 00 00 00 04 74 30 30b
004000B0	00 00 00 00 00 00 01 00 00 00 00 00 F8 00 00 00b
004000C0	00 00 00 00 04 74 30 30 00 00 00 00 00 00 01 00
004000D0	0A 0C 00 00 08 01 00 00 48 A1 00 00 E8 02 00 00
004000E0	00 00 00 00 00 00 00 00 30 04 00 00 00 00 00 00
004000F0	00 00 00 00 00 00 00 00 40 04 00 00 73 09 00 00
00400100	00 00 00 00 00 00 00 00 B4 00 00 00 14 00 00 00l
00400110	00 00 00 00 00 00 00 00 06 00 44 00 56 00 43 00DUC
00400120	4C 00 41 00 4C 00 06 00 54 00 46 00 4F 00 52 00LALTFDR
00400130	4D 00 31 00 08 00 4D 00 41 00 49 00 4E 00 49 00MILMAINI
00400140	43 00 4F 00 4E 00 00 00 28 00 00 00 20 00 00 00CON...

Enter binary string to search for

ASCII Enable

UNICODE

HEX +06 45 6E 61 62 6C 65

Entire block

Case sensitive

OK Cancel

El primer sitio donde para mirad después de Enable esta el numero 64 (d) con lo cual la palabra completa seria Enabled después 08 y 08, parece que estamos en el lugar correcto, la primera parada correspondería al uno de los 5 botones desactivados

Address	Hex dump	UNICODE
0040AB69	0F F0 08 88 00 08 88 88 88 88 88 88 88 88 88 88
0040AB79	80 00 88 07 56 69 73 69 62 6C 65 08 00 00 07 54
0040AB89	42 75 74 74 6F 6E 07 42 75 74 74 6F 6E 31 04 4C
0040AB99	65 66 74 02 30 03 54 6F 70 02 30 05 57 69 64 74
0040ABA9	68 03 91 00 06 48 65 69 67 68 74 02 19 07 43 61
0040ABB9	70 74 69 6F 6E 06 08 41 63 74 69 76 61 60 65 07
0040ABC9	45 6E 61 62 6C 65 64 08 08 54 61 62 4F 72 64 65
0040ABD9	72 02 00 07 4F 6E 43 6C 69 63 68 07 0C 42 75 74
0040ABE9	74 6F 6E 31 43 6C 69 63 6B 00 00 07 54 42 75 74k
0040ABF9	74 6F 6E 07 42 75 74 74 6F 6E 32 04 4C 65 66 74
0040AC09	02 28 03 54 6F 70 03 80 00 05 57 69 64 74 68 03
0040AC19	A1 00 06 48 65 69 67 68 74 02 19 07 45 6E 61 62i
0040AC29	6C 65 64 08 08 54 61 62 4F 72 64 65 72 02 01 07
0040AC39	4F 6E 43 6C 69 63 68 07 0C 42 75 74 74 6F 6E 32
0040AC49	43 6C 69 63 6B 00 00 07 54 42 75 74 74 6F 6E 07k
0040AC59	42 75 74 74 6F 6E 33 04 4C 65 66 74 02 28 03 54

Si queréis asegurarnos (y para ver de que botón se trata) pues traducid desde Enabled hacia atrás Lo podemos hacer directamente desde la ventana search for binary String escribiendo los valores sombreados
 Creo que en la imagen queda claro

Address	Hex dump	UNICODE
0040AB69	0F F0 08 88 00 08 88 88 88 88 88 88 88 88 88 88
0040AB79	80 00 88 07 56 69 73 69 62 6C 65 08 00 00 07 54
0040AB89	42 75 74 74 6F 6E 07 42 75 74 74 6F 6E 31 04 4C
0040AB99	65 66 74 02 30 03 54 6F 70 02 30 05 57 69 64 74
0040ABA9	68 03 91 00 06 48 65 69 67 68 74 02 19 07 43 61
0040ABB9	70 74 69 6F 6E 06 08 41 63 74 69 76 61 60 65 07
0040ABC9	45 6E 61 62 6C 65 64 08 08 54 61 62 4F 72 64 65
0040ABD9	72 02 00 07 4F 6E 43 6C 69 63 68 07 0C 42 75 74
0040ABE9	74 6F 6E 31 43 6C 69 63 6B 00 00 07 54 42 75 74k
0040ABF9	74 6F 6E 07 42 75 74 74 6F 6E 32 04 4C 65 66 74
0040AC09	02 28 03 54 6F 70 03 80 00 05 57 69 64 74 68 03
0040AC19	A1 00 06 48 65 69 67 68 74 02 19 07 45 6E 61 62i
0040AC29	6C 65 64 08 08 54 61 62 4F 72 64 65 72 02 01 07
0040AC39	4F 6E 43 6C 69 63 68 07 0C 42 75 74 74 6F 6E 32
0040AC49	43 6C 69 63 6B 00 00 07 54 42 75 74 74 6F 6E 07k
0040AC59	42 75 74 74 6F 6E 33 04 4C 65 66 74 02 28 03 54

Enter binary string to search for

ASCII Height Caption Activame Enabled

UNICODE ?

HEX +47 06 48 65 69 67 68 74 02 19 07 43 61
70 74 69 6F 6E 06 08 41 63 74 69 76
61 6D 65 07 45 6E 61 62 6C 65 64

Entire block

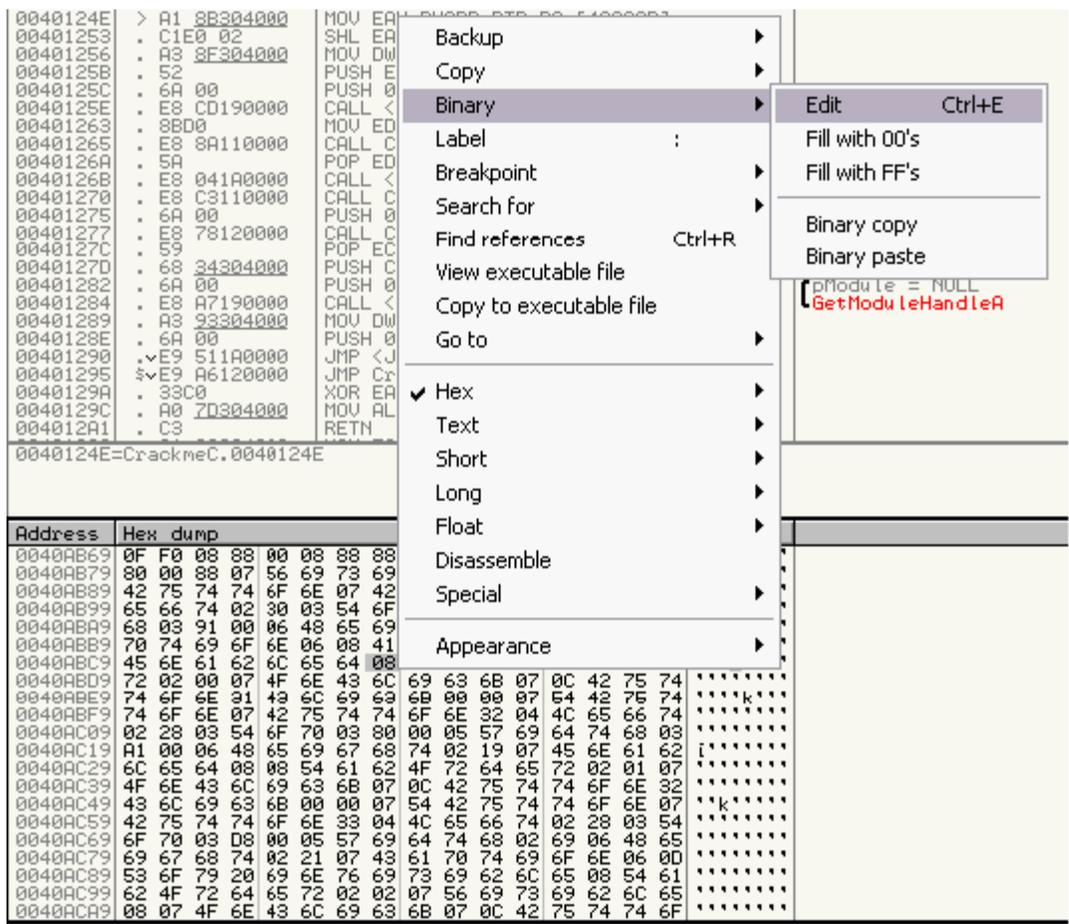
Case sensitive

OK Cancel

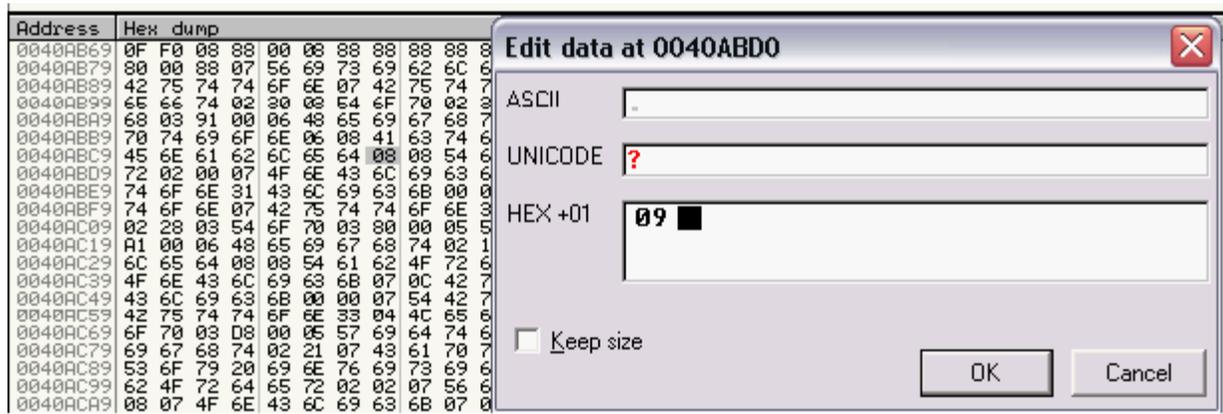
Hackman nos dijo que la diferencia entre Enabled y Disabled era cuestión de que el valor fuera 08 o 09
 Vamos a cambiarlo, de los dos 08 que vemos a continuación cambiaremos solo el 08 que esta justo a

continuación de la d (64) final de la palabra Enabled

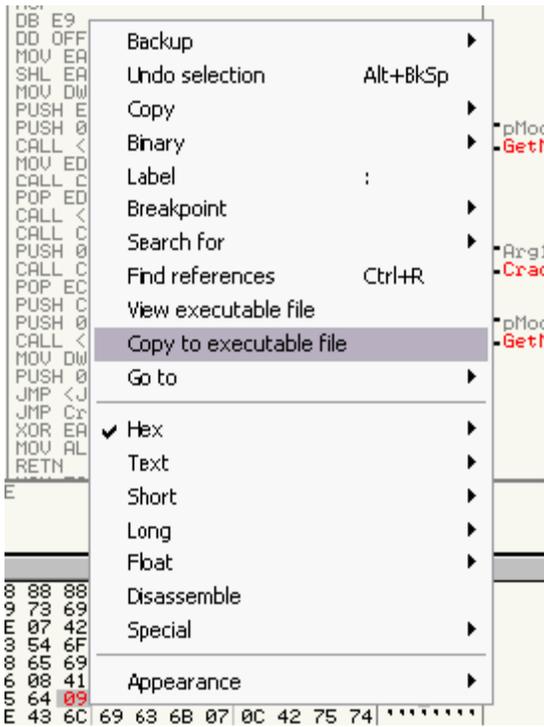
Lo sombreamos y click derecho Binary - Edit o Control + E



En la ventana que aparece cambiamos el valor para alterar la propiedad del botón



Una vez cambiado guardamos los cambios en el ejecutable con el botón derecho del ratón Copy to executable file



Tenemos el primer salto cambiado, cerramos Olly y lo probamos
 Los demás botones los buscaremos y cambiaremos de la misma forma buscando Enabled (a estas alturas supongo que habréis descubierto Control + L) y cambiando el valor
 Esto será válido para 4 de los 5 botones, en el 5º la propiedad que hay que cambiar es "Visible"

Hacedlo de uno en uno y comprobando los cambios en el ejecutable

ActiviWonder2 CrackmeD.exe

Trastead con el lo que queráis para aclararos sobre lo visto anteriormente, solo haré un pequeño apunte

Una vez que encontréis los valores a cambiar en Olly (acordaros de volcar la memoria) los podemos cambiar todos a la vez

Address	Hex dump	UNICODE
00479F92	06 0E 53 6F 79 20 65 6C 20 70 72 69 6D 65 72 6F
00479FA2	07 45 6E 61 62 6C 65 64 09 08 54 61 62 4F 72 64
00479FB2	65 72 02 00 07 4F 6E 43 6C 69 63 6B 07 0C 42 75
00479FC2	74 74 6F 6E 31 43 6C 69 63 6B 00 00 07 54 42 75
00479FD2	74 74 6F 6E 07 42 75 74 74 6F 6E 32 04 4C 65 66
00479FE2	74 02 10 03 54 6F 70 02 68 05 57 69 64 74 68 03
00479FF2	99 01 06 48 65 69 67 68 74 02 21 07 43 61 70 74
0047A002	69 6F 6E 06 45 59 6F 20 76 6F 79 20 65 6E 20 73
0047A012	65 67 75 6E 64 6F 20 6C 75 67 61 72 20 70 75 65
0047A022	73 20 61 75 6E 71 75 65 20 71 75 69 73 69 65 72
0047A032	61 20 6E 6F 20 70 6F 64 72 69 61 20 73 65 72 20
0047A042	65 6C 20 70 72 69 6D 65 72 6F 07 45 6E 61 62 6C
0047A052	65 64 09 08 54 61 62 4F 72 64 65 72 02 01 07 4F
0047A062	6E 43 6C 69 63 6B 07 0C 42 75 74 74 6F 6E 32 43
0047A072	6C 69 63 6B 00 00 07 54 42 75 74 74 6F 6E 07 42
0047A082	75 74 74 6F 6E 33 04 4C 65 66 74 02 18 03 54 6F
0047A092	70 09 49 01 06 67 69 64 74 69 09 09 00 06 49 65
0047A0A2	69 67 68 74 02 19 07 43 61 70 74 69 6F 6E 06 1F
0047A0B2	53 69 20 71 75 69 65 72 65 73 20 73 61 6C 69 72
0047A0C2	20 61 63 74 69 76 61 6D 65 20 61 6E 74 65 73 07
0047A0D2	45 6E 61 62 6C 65 64 09 08 54 61 62 4F 72 64 65
0047A0E2	72 02 02 07 4F 6E 43 6C 69 63 6B 07 0C 42 75 74
0047A0F2	74 6F 6E 32 43 65 66 74 02 18 03 54 6F 70 02 68

y guardar los cambios de una vez también en el ejecutable seleccionando hasta que pillemos todos

los cambios sombreados y Copy to executable

004648E4 . 0000 ADD BYTE PTR DS:CE
004648E6 . 0000 ADD BYTE PTR DS:CE
004648E8 . 0000 ADD BYTE PTR DS:CE
004648EA . 0000 ADD BYTE PTR DS:CE
004648EC . 0000 ADD BYTE PTR DS:CE
004648EE . 0000 ADD BYTE PTR DS:CE
004648F0 . 0000 ADD BYTE PTR DS:CE
004648F2 . 0000 ADD BYTE PTR DS:CE
004648F4 . 0000 ADD BYTE PTR DS:CE
004648F6 . 0000 ADD BYTE PTR DS:CE
004648F8 . 0000 ADD BYTE PTR DS:CE
EBP=0012FFF0

Address	Hex dump
00479F92	06 0E 53 6F 79 20 65 6C 20 70 72 69
00479FA2	07 45 6E 61 62 6C 65 64 09 08 54 61
00479FB2	65 72 02 00 07 4F 6E 43 6C 69 63 6B
00479FC2	74 74 6F 6E 31 43 6C 69 63 6B 00 00
00479FD2	74 74 6F 6E 07 42 75 74 74 6F 6E 32
00479FE2	74 02 10 03 54 6F 70 02 68 05 57 69
00479FF2	99 01 06 48 65 69 67 68 74 02 21 07
0047A002	69 6F 6E 06 45 59 6F 20 76 6F 79 20
0047A012	65 67 75 6E 64 6F 20 6C 75 67 61 72
0047A022	73 20 61 75 6E 71 75 65 20 71 75 69
0047A032	61 20 6E 6F 20 70 6F 64 72 69 61 20
0047A042	65 6C 20 70 72 69 6D 65 72 6F 07 45
0047A052	65 64 09 00 54 61 62 4F 72 64 65 72
0047A062	6E 43 6C 69 63 6B 07 0C 42 75 74 74
0047A072	6C 69 63 6B 00 00 07 54 42 75 74 74
0047A082	75 74 74 6F 6E 33 04 4C 65 66 74 02
0047A092	70 03 48 01 05 57 69 64 74 68 03 09
0047A0A2	69 67 68 74 02 19 07 43 61 70 74 69
0047A0B2	53 69 20 71 75 69 65 72 65 73 20 73
0047A0C2	20 61 63 74 69 76 61 6D 65 20 61 6E
0047A0D2	45 6E 61 62 6C 65 64 08 08 54 61 62
0047A0E2	72 02 02 07 4F 6E 43 6C 69 63 6B 07

Hacedlo siempre y cuando estéis seguros que esos son los valores a cambiar, sino hacedlo despacio, de uno en uno traduciendo Hexa a ASCII y probando los cambios en memoria.

ActiviWonder3 CracmeVB.exe

Esta hecho en Visual Basic

Ejecutamos el crackme para ver de qué va

Lo abrimos directamente con Exescope y nos damos de narices con esto



No se ven ni RCDATA ni Dialogs

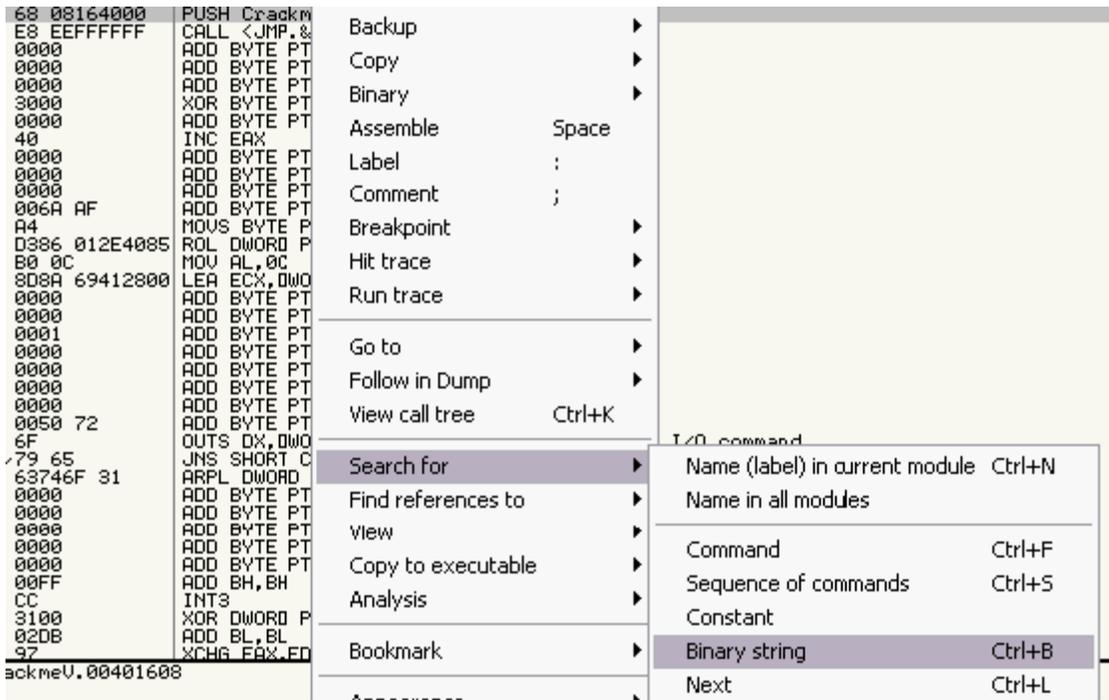
Bueno siempre nos quedara Paris...

Quería decir que tenemos a Oilly esperándonos

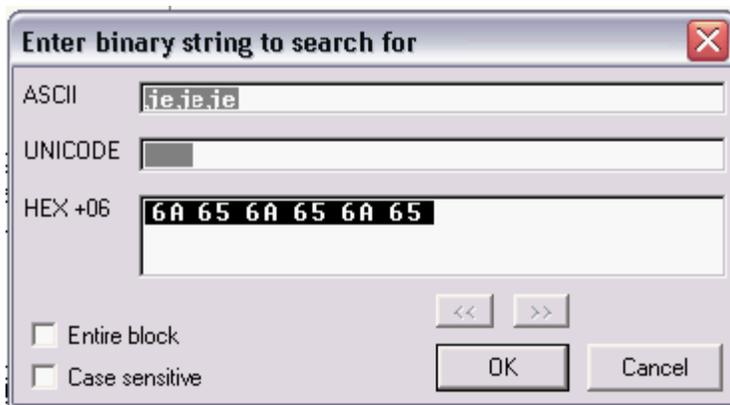
Abrimos Oilly y cargamos el crackme, hacemos lo que habíamos visto anteriormente: el volcado de la memoria y buscamos Enabled y no lo vemos

Estamos perdidos ¿? pues no mirad vamos a hacer una cosa que aprendí en un manual de Coco (en la web de crackslatinos nos encontramos con un tutorial de Coco titulado habilitando botones entre otros)

Sin volcados de memoria, directamente Search for binary String



Introducimos el caption del botón o parte de el



Aparecemos aquí y vemos jejejeje y arriba command 1 (en delphi a los botones los llaman Button en visual basic command)

En este crackme no nos podemos equivocar jamás de botón pues solo tiene 1

```

00401540 00 DB 00
00401540 . 43 6F 6D 6D 6 ASCII "Command1",0
00401556 04 DB 04
00401557 01 DB 01
00401558 08 DB 08
00401559 00 DB 00
0040155A . 6A 65 6A 65 6 ASCII "jejejeje",0
00401563 04 DB 04
00401564 90 NOP
00401565 06 DB 06
00401566 D0 DB D0
00401567 02 DB 02
00401568 BF DB BF
00401569 04 DB 04
0040156A EF DB EF
0040156B 01 DB 01
0040156C 08 DB 08
0040156D 00 DB 00
0040156E 11 DB 11
0040156F 00 DB 00
00401570 00 DB 00
00401571 FF DB FF
00401572 03 DB 03
00401573 43 DB 43
00401574 00 DB 00

```

Atención al truco de magia:

Contamos 11 líneas desde jejejeje y en esa dirección cambiamos 00 por 01

```

00401548 08 DB 08
0040154C 00 DB 00
0040154D . 43 6F 6D 6D 6 ASCII "Command1",0
00401556 04 DB 04
00401557 01 DB 01
00401558 08 DB 08
00401559 00 DB 00
0040155A . 6A 65 6A 65 6 ASCII "jejejeje",0
00401563 04 DB 04
00401564 90 NOP
00401565 06 DB 06
00401566 D0 DB D0
00401567 02 DB 02
00401568 BF DB BF
00401569 04 DB 04
0040156A EF DB EF
0040156B 01 DB 01
0040156C 08 DB 08
0040156D 00 DB 00
0040156E 11 DB 11
0040156F 00 DB 00
00401570 00 DB 00
00401571 FF DB FF
00401572 03 DB 03
00401573 43 DB 43
00401574 00 DB 00
00401575 00 DB 00

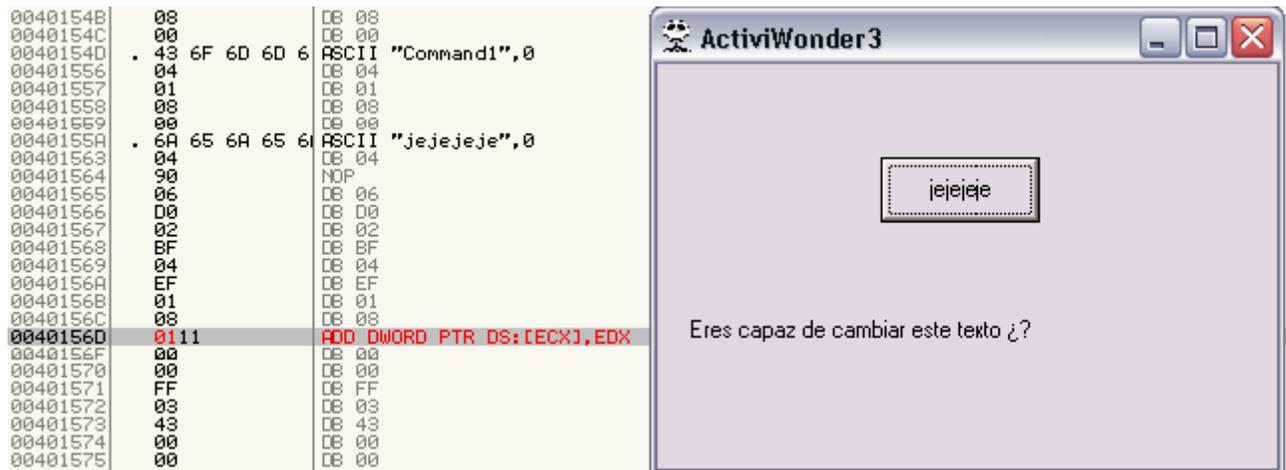
```



Corremos el crackme con F9 y botón habilitado
 Porque la línea 11 y no otra ¿?

Pues por que las otras cambiarían el valor de otra propiedad, como la altura la posición, anchura, etc. (Width, Height, Top, etc)

Esto no lo comprobé por mi mismo pues la fuente de donde proviene es más que fiable

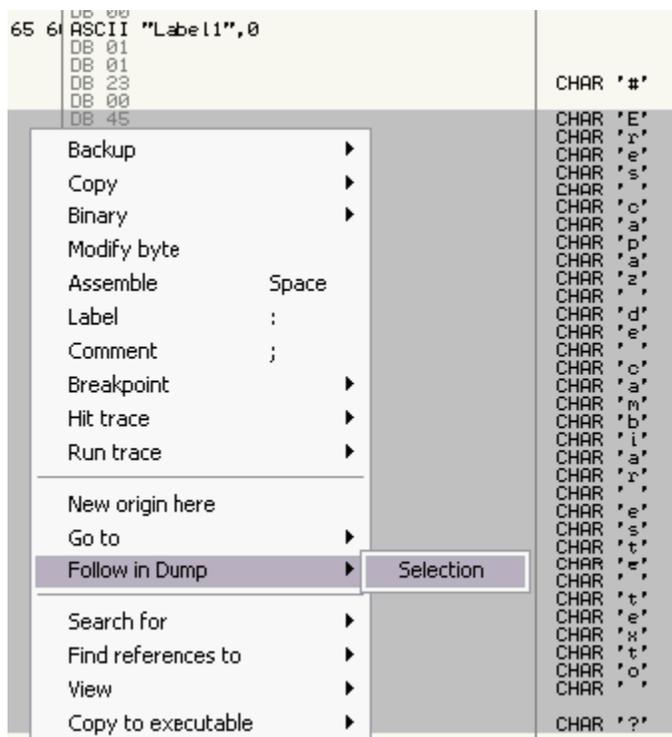


Segunda parte vamos a cambiar el texto

Hacemos lo mismo buscamos la binary String pero esta vez el caption del label Eres capaz ...

Lo vemos en posición vertical y lo sombreamos

Con click derecho Follow in Dump para seguirlo abajo en la ventana del Dump



En la ventana del Dump Control + E para editarlo

Address	Hex dump	UNICODE
00401565	06 D0 02 BF 04 EF 01 08 00 11 00 00 FF 03 43 00C
00401575	00 00 02 06 00 4C 61 62 65 6C 31 00 01 01 23 0013
00401585	45 72 65 73 20 63 61 70 61 7A 20 64 65 20 63 61?
00401595	60 62 69 61 72 20 65 73 74 65 20 74 65 78 74 6F?
004015A5	20 BF 3F 00 05 F0 00 00 07 77 10 EF 01 12 01 00?
004015B5	FF 02 04 50 00 00 00 0B 97 47 81 C2 8E 83 42 83?
004015C5	07 00 B9 54 99 3C F4 00 00 00 00 00 00 00 00 00?
004015D5	00 00 00 00 00 00 00 00 00 10 01 00 00 00 00?
004015E5	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00?
004015F5	00 00 00 1C 04 00 00 00 00 00 00 9C 11 40 00 4C?
00401605	00 00 00 56 42 35 21 F0 1F 56 42 36 45 53 2E 44?
00401615	4C 4C 00 00 00 00 00 2A 00 00 00 00 00 00 00 00?
00401625	00 00 00 00 00 0A 00 0A 0C 00 00 09 04 00 00 00?
00401635	00 00 00 B8 18 40 00 12 F0 30 00 00 FF FF FF 00?
00401645	00 00 00 01 00 00 00 01 00 00 00 E9 00 00 00 B8?
00401655	15 40 00 B8 15 40 00 58 11 40 00 78 00 00 00 82?

Edit data at 00401585

ASCII:

UNICODE:

HEX +00:
45 72 65 73 20 63 61 70 61 7A 20 64 65 20 63 61
65 20 63 61 6D 62 69 61 72 20 65 73
74 65 20 74 65 78 74 6F 20 BF 3F

Keep size

OK Cancel

Atención

Tenemos que sustituir carácter por carácter contando los espacios (20) como un carácter más
 Fijaros que después de listillo en la parte ascii parece que no hay nada, pero si miráis en la parte hexa esta lleno de "barra espaciadora" (20)
 Hasta completar el numero de caracteres de la cadena original.

Address	Hex dump	UNICODE
00401565	06 D0 02 BF 04 EF 01 08 00 11 00 00 FF 03 43 00C
00401575	00 00 02 06 00 4C 61 62 65 6C 31 00 01 01 23 0013
00401585	45 72 65 73 20 63 61 70 61 7A 20 64 65 20 63 61?
00401595	60 62 69 61 72 20 65 73 74 65 20 74 65 78 74 6F?
004015A5	20 BF 3F 00 05 F0 00 00 07 77 10 EF 01 12 01 00?
004015B5	FF 02 04 50 00 00 00 0B 97 47 81 C2 8E 83 42 83?
004015C5	07 00 B9 54 99 3C F4 00 00 00 00 00 00 00 00 00?
004015D5	00 00 00 00 00 00 00 00 00 10 01 00 00 00 00?
004015E5	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00?
004015F5	00 00 00 1C 04 00 00 00 00 00 00 9C 11 40 00 4C?
00401605	00 00 00 56 42 35 21 F0 1F 56 42 36 45 53 2E 44?
00401615	4C 4C 00 00 00 00 00 2A 00 00 00 00 00 00 00 00?
00401625	00 00 00 00 00 0A 00 0A 0C 00 00 09 04 00 00 00?
00401635	00 00 00 B8 18 40 00 12 F0 30 00 00 FF FF FF 00?
00401645	00 00 00 01 00 00 00 01 00 00 00 E9 00 00 00 B8?
00401655	15 40 00 B8 15 40 00 58 11 40 00 78 00 00 00 82?

Edit data at 00401585

ASCII:

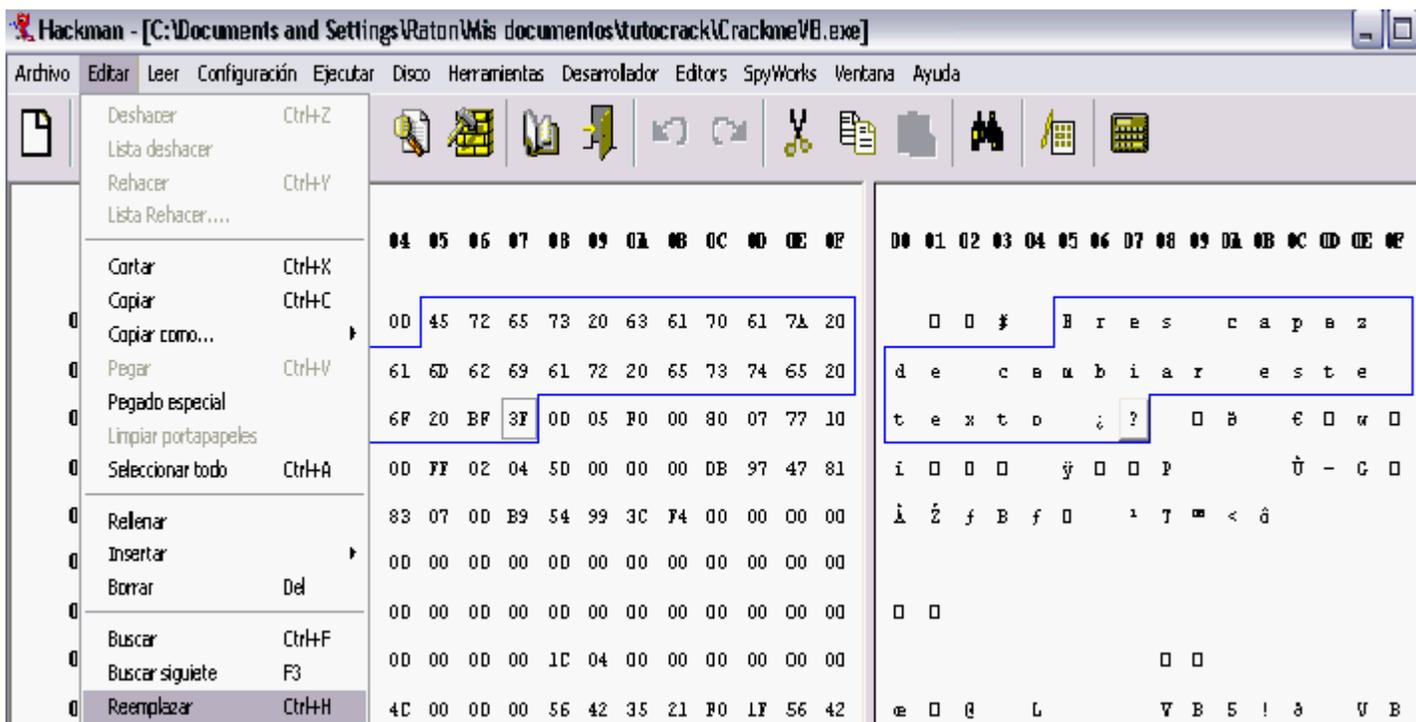
UNICODE:

HEX +23:
50 75 65 73 20 73 69 20 20 20 20 20 6C
69 73 74 69 6C 6C 6F 20 20 20 20 20
20 20 20 20 20 20 20 20 20 20 20

Keep size

OK Cancel

Sombreamos todo en el Dump y a guardar



Como dije en los primeros capítulos Oilly es capaz de resolvernros casi todo
Después de tanto rollo no me quedan ganas de dar consejos

Un saludo

Por cierto existe un programa que es un editor de recursos para Visual Basic que se llama
VBReFormer Jejeje

Gracias

A todas las personas que colaboran desde el foro de HackxCrack para llevar adelante el curso, tanto los que colaboran aportando sus conocimientos como complemento al curso como a los que postean sus dudas para que aprendamos todos y por supuesto a los moderadores del mismo

A todos los crackers y programadores de los cuales he aprendido y sigo aprendiendo.

A los creadores de crackmes

En especial y sin menospreciar a nadie a [Ricardo Narvaja](#) por su aportación y su trabajo sobre el estudio de las protecciones y sus tutoriales en castellano y a [Makkakko](#) por sus tutoriales con Olly Debugger (Recomendados 100%) y por supuesto a [Shoulck](#) por la ayuda desinteresada que me está prestando a costa de algo tan preciado como su tiempo.

A ti que me estás leyendo.