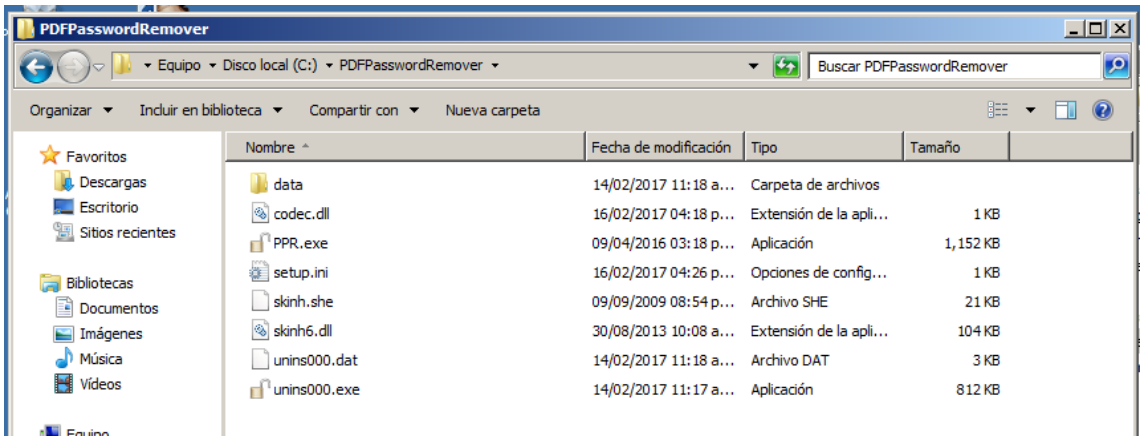
PDF Password Remover	
Version	1.5
URL:	http://www.pdfpasswordremover.com/
Proteccion :	Ninguna
Dificultad :	Newbie
Herramientas :	RDG Packer Detector v0.7.6 , Ollydbg y Mucho animos.
Objetivo :	Registrarlo + Parchar
Reverser :	L1oR
Grupo :	Peru Crackeando
Gracias a Ricardo Narvaja, por sus manuales de Ollydbg, donde fue la base de empezar en el Reverser, y a Davicorm por sus aportes constantes subidos al grupo, a Abel, Nox, K3n4n (Se encuentra secuestrado) y SoftDat.	

INFORMACION DE LA VICTIMA :

PDF Password Remover permite a los usuarios eliminar las restricciones de formato PDF, para que tenga la capacidad de imprimir y modificar, el costo del software es de unos $29.95 XD.

ATACANDO :

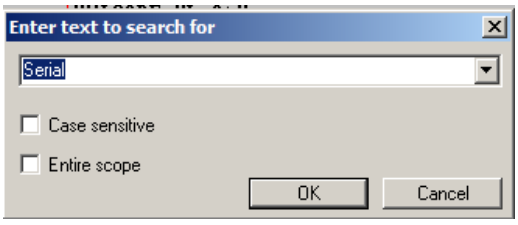
Lo primero que realizo es encontrar la ruta de instalacion del software, para luego crear un Backup, por medidas de seguridad.



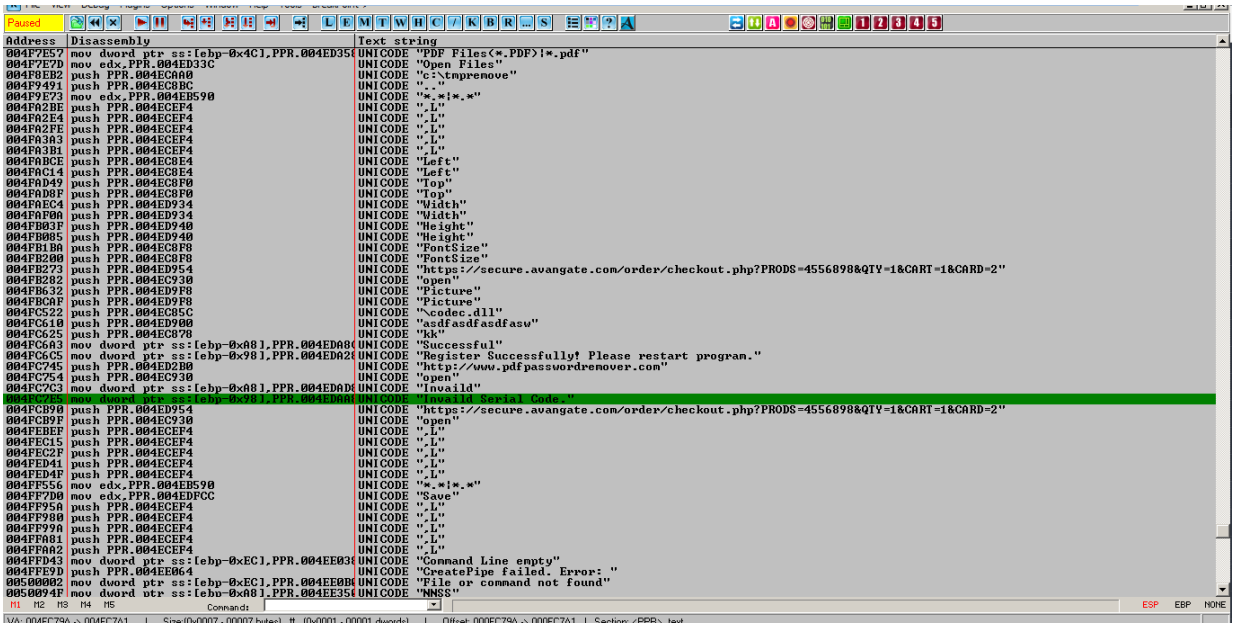
Luego de obtener un backup, procederemos a ver en que lenguaje se desarrollo dicha aplicación, como vemos esta desarrollado Visual Basic 6.0 y no tiene medida de seguridad, estamos en un buen camino.

Uno de los detalles es recolectar informacion, por ejemplo el BAD BOY, del software, en este caso nos muestra la venta “Invalid Serial Code”.

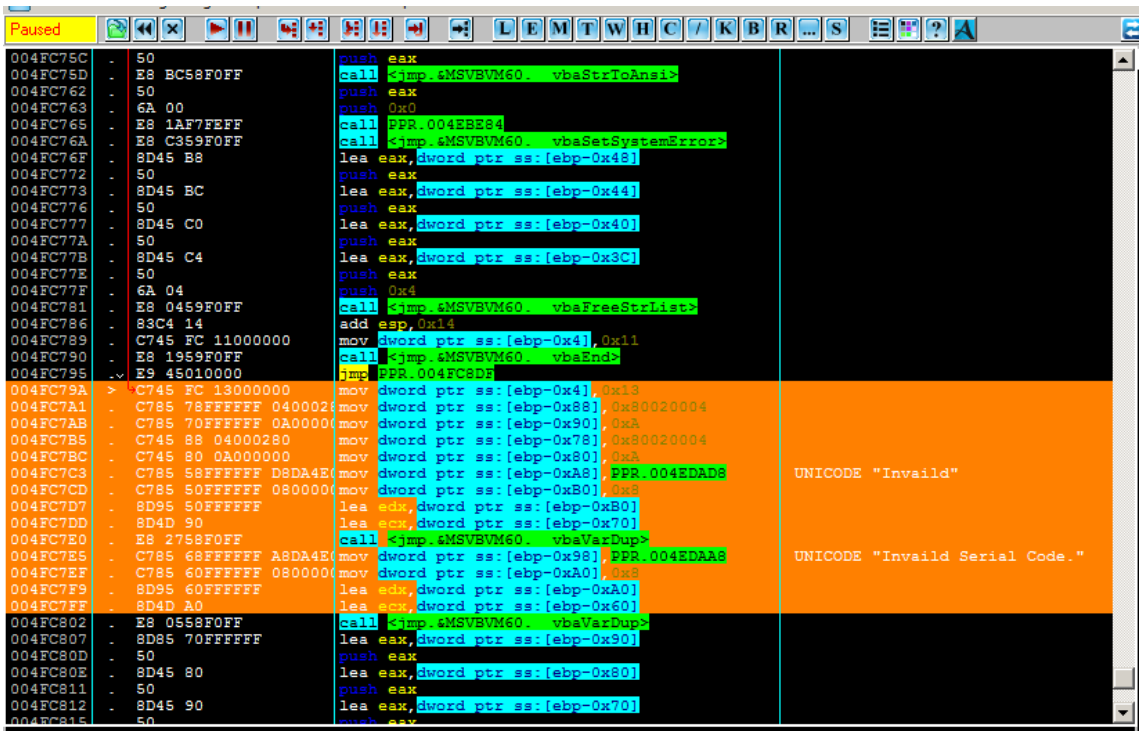
Con el dato obtenido del BAD BOY, buscaremos el STRING, buscaremos la palabra “Serial”.



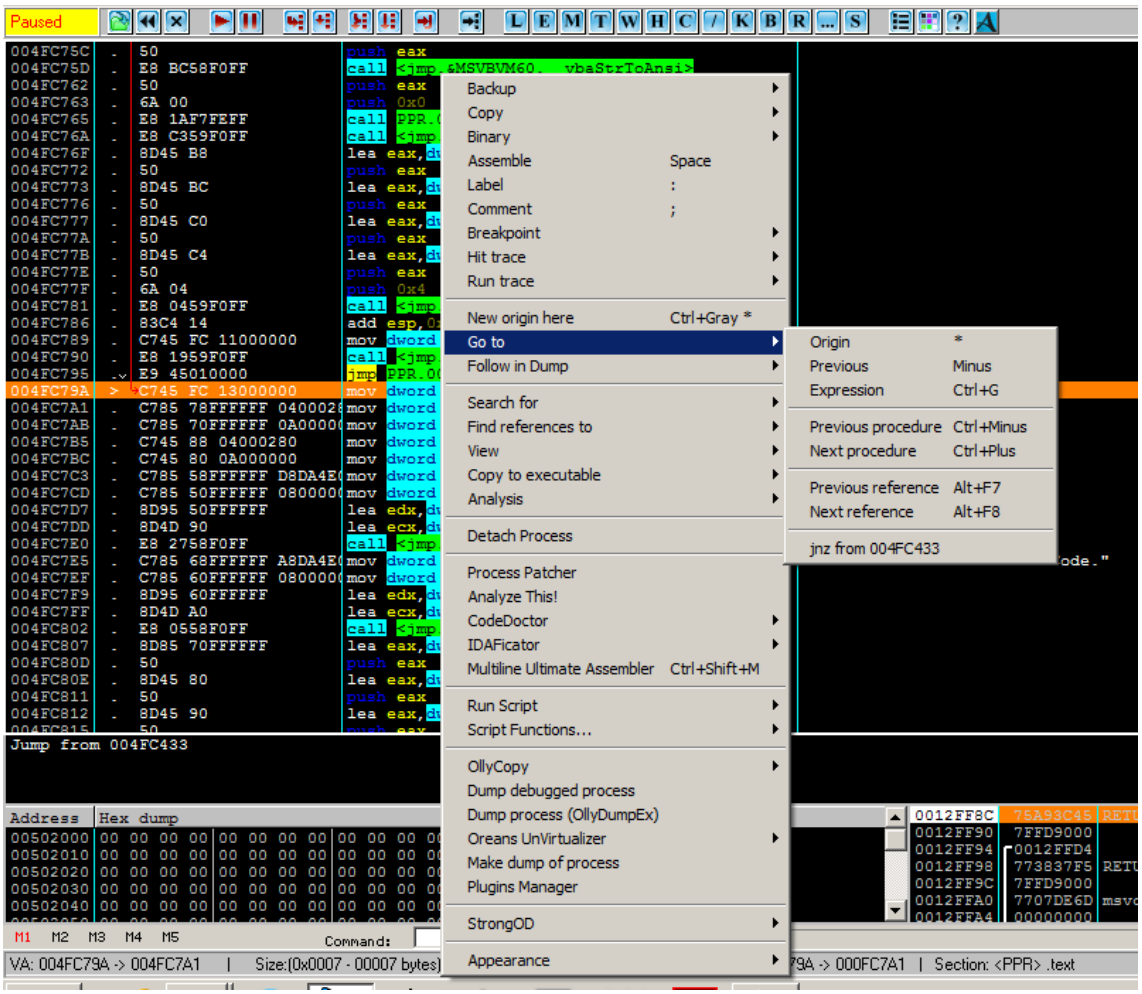
En este caso, encontramos el STRING “Invalid Serial Code”, en la direccion de memoria 004FC7E5.



Con este datos iremos a dicha direccion de memoria 004FC7C3, si observamos en la direccion 004FC79A nos muestra una linea de codigo que fue llamado por medio de un SALTO.

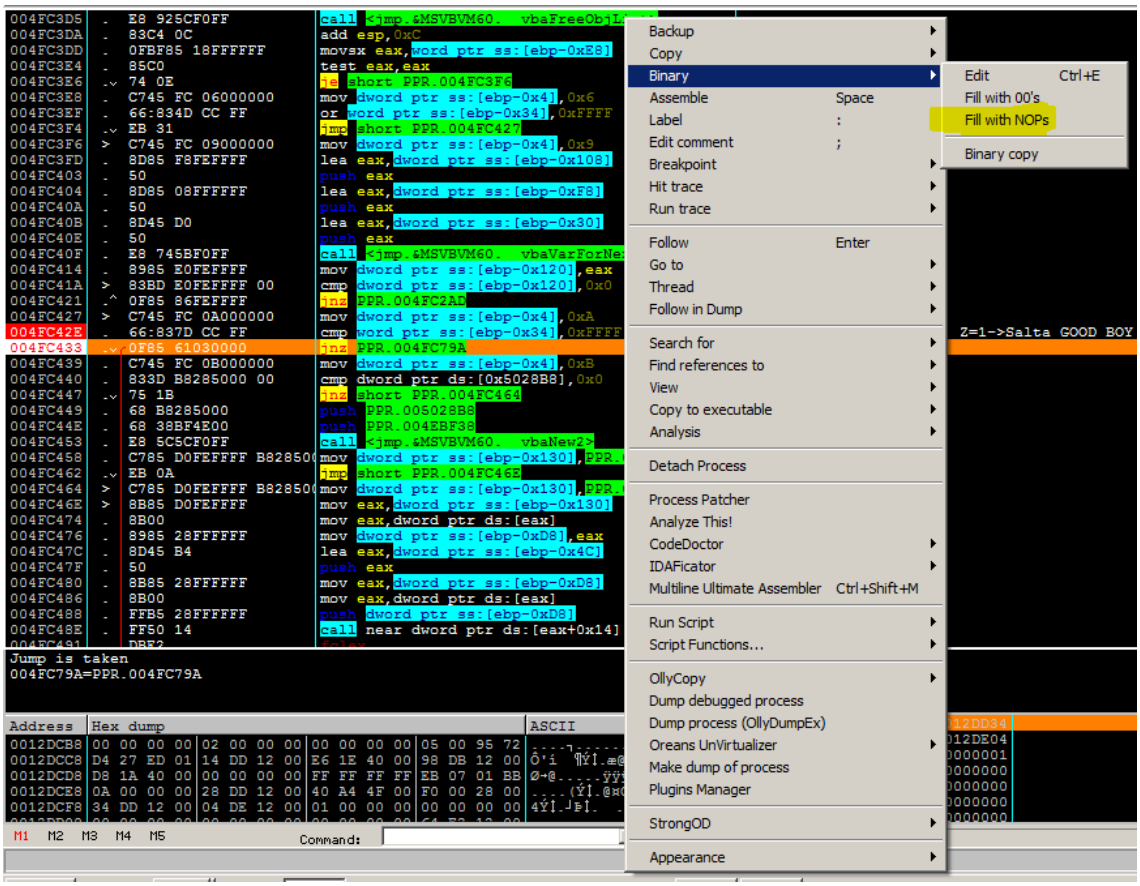


Para ello en la siguiente imagen nos muestra que linea de codigo hace el salto, clic derecho en la linea de comando con direccion 004FC79A -> Go to -> JNZ FROM 004FC433,



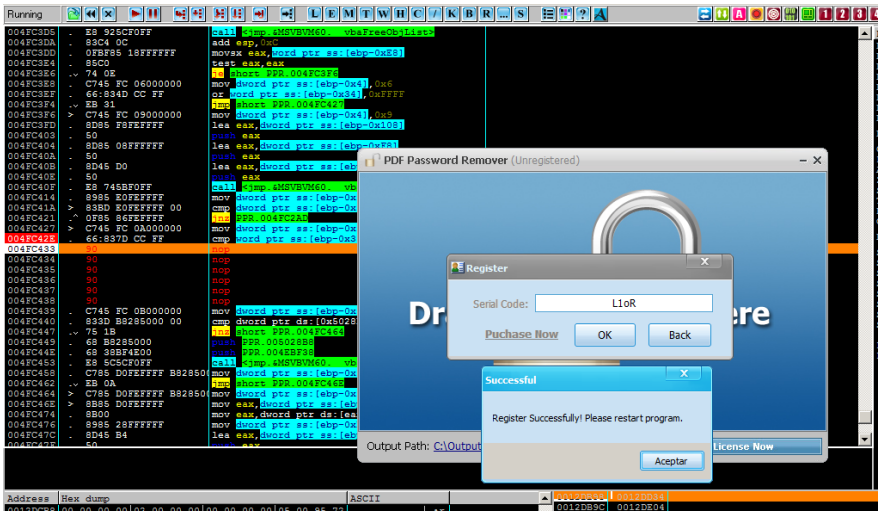
Como vemos la direccion de memoria 004FC433 se produce el salto al BAD BOY, pondremos un BP (BreakPoint), luego analizaremos la direccion de memoria con el codigo:
CMP word ptr ss:[ebp-0x34], 0xFFFF, quiere decir si Z=0 -> Salta al BAD BOY, pero si Z=1 -> GOOD BOY, en este punto podemos poder ser crativos.
1) Modificar el codigo y que sea : CMP 0Xffff, 0Xffff
2) Modificar agregando NOP en la direccion de memoria: 004FC433.

En mi caso utilize la segunda opcion, convertir el salto JNZ con direccion de memoria 004FC433 a NOPs, para que no salte y asi se vaya al GOOD BOY.



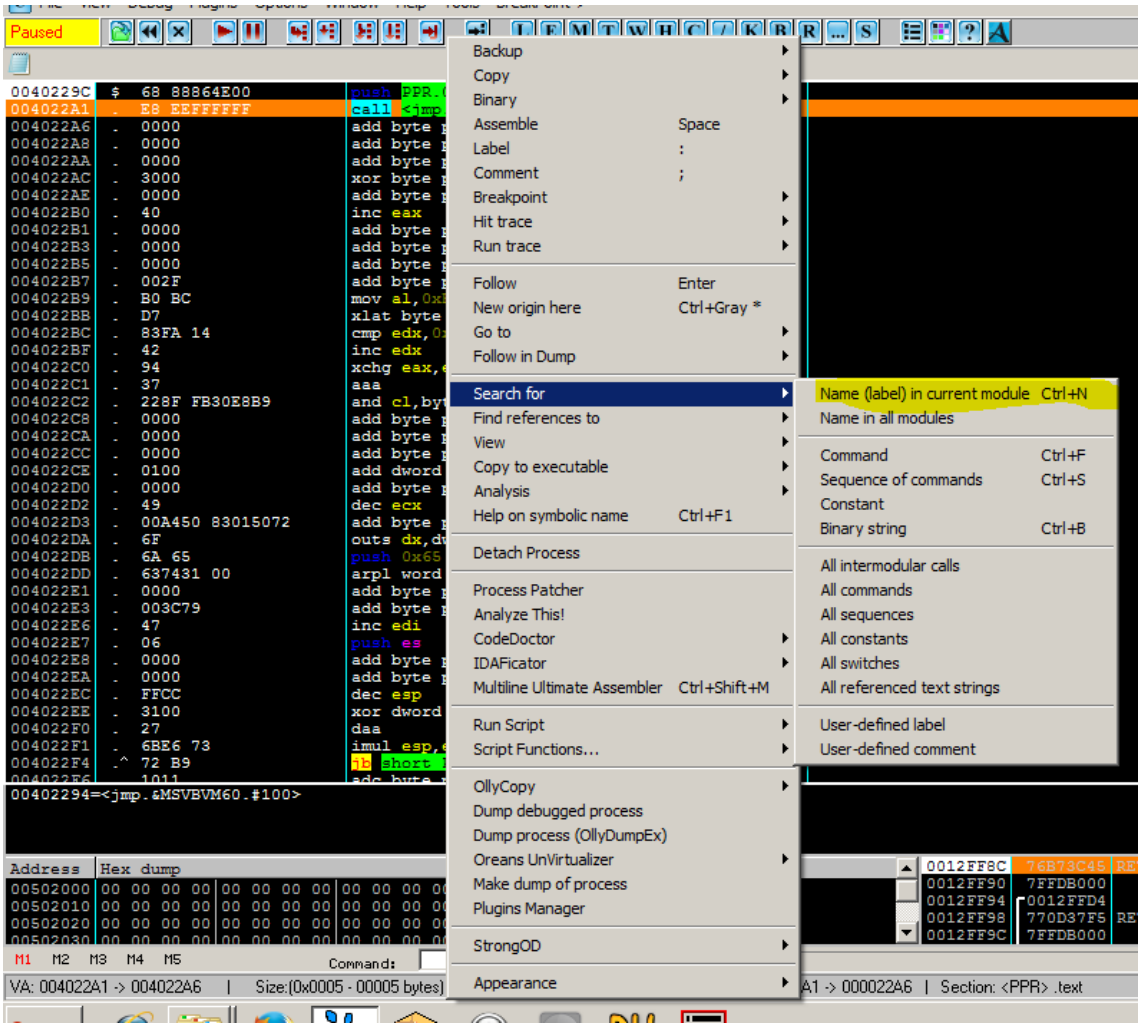
Como veras al final queda con 6 NOPs, como se comento esto realizar que no se haga ningun salto por eso le dice : NO OPERATION (No operación).

Para validar que nuestra modificacion es la correcta, ejecutamos el software con F9, y vemos que nos aparece que el GOOD BOY, reto vencido, veremos otro metodo, asi que no se aburran y procedamos.

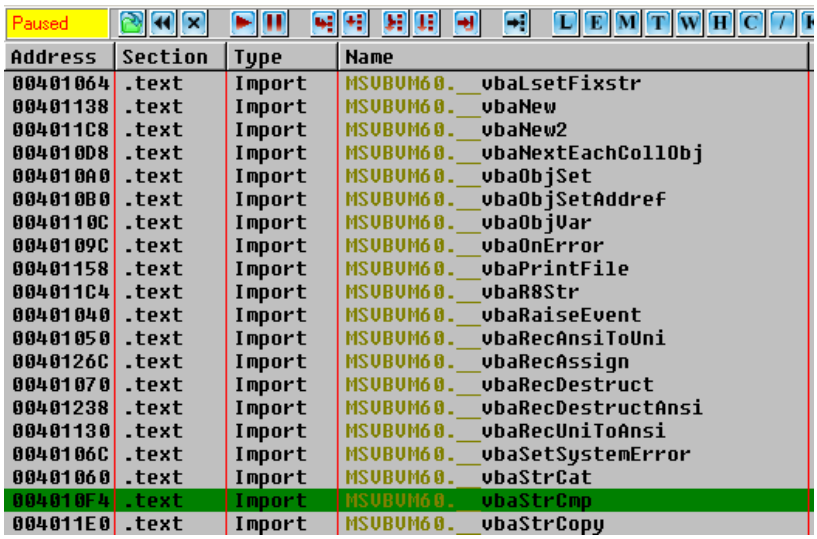


SEGUNDO METODO.

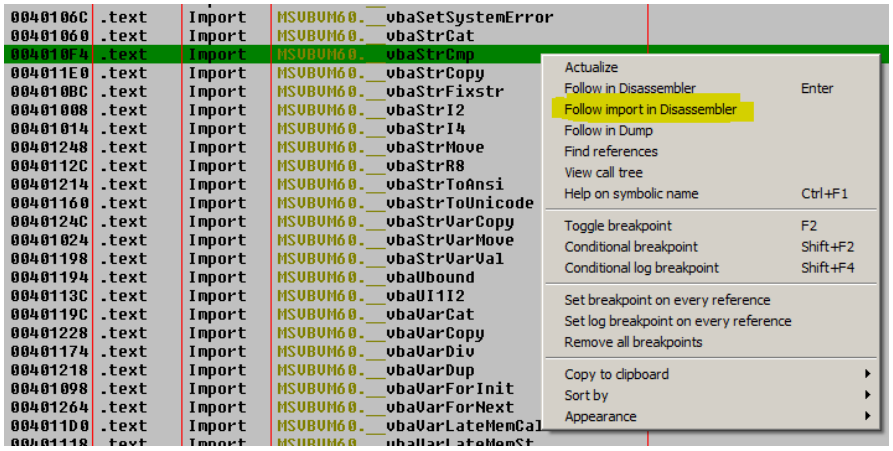
Primero comenzamos a depurar el software, y luego vamos a buscar un API.



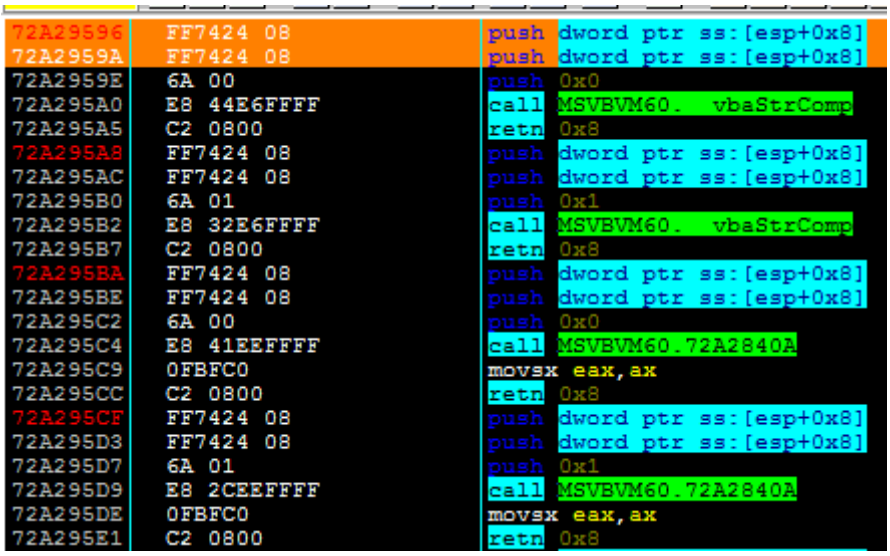
En este caso buscamos este API llamado VBASTRCMP que se encuentra en la direccion de memoria 004010F4 , lo que realiza es comparar dos string,



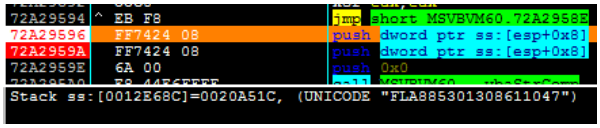
Hacemos clic derecho en 004010F4 -> Follow import in Disassembler, nos iremos a ver el codigo en el area de desemsanblado.

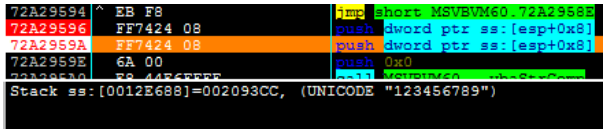


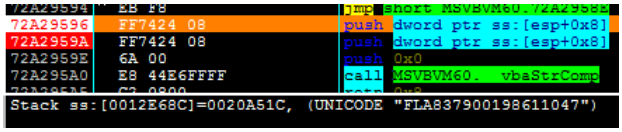
Veremos en la direccion de memoria 72A29596 y 72A2959A.

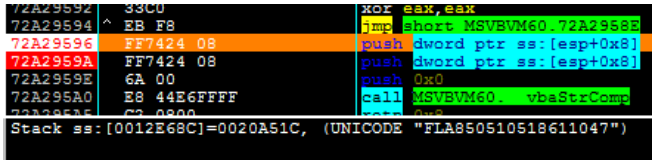


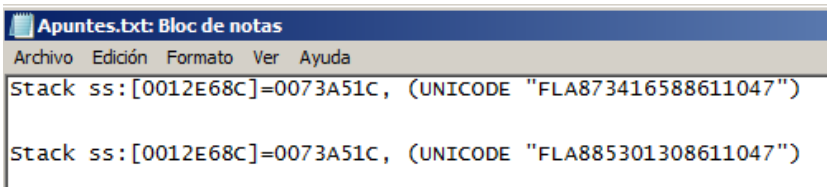
Hacemos clic en PLAY o F9, para poder ver la clave en este caso escribi “123456789”, en el primer PUSH con direccion de memoria 72A29596 y me muetra el dato FLA885301308611047, si seguimos dando PLAY (F9), en el primer PUSH, nos mostrara la clave.











Valimos copiando una de las claves para poder visualizar si nos aparece el GOOD BOY.

Graciass por su apoyo constante y un saludos. HAPPY CRACKING.