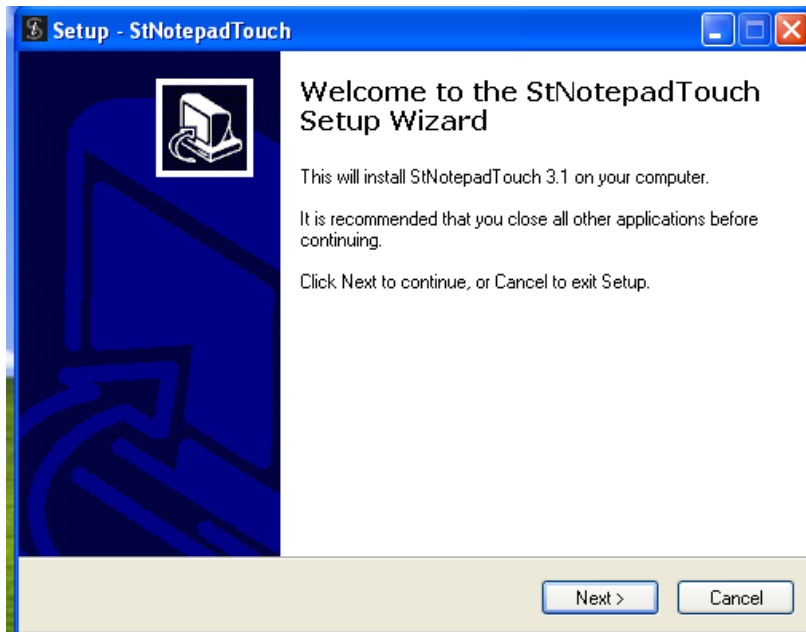


Hola.

El programa de hoy si se para que sirve..... :-)

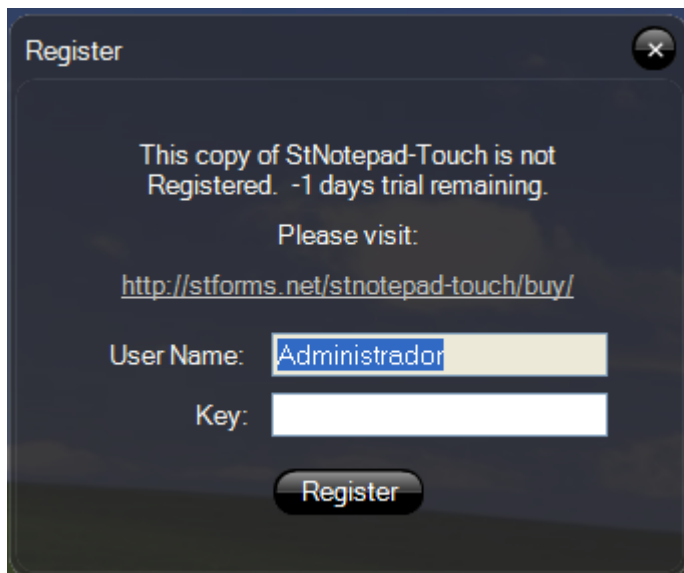
se trata de StNotepadTouch que es un programa que encontré buscando un editor al cual le pudiera alterar su transparencia para poder leer y escribir sin necesidad de estarlo moviendo de un lado para otro.



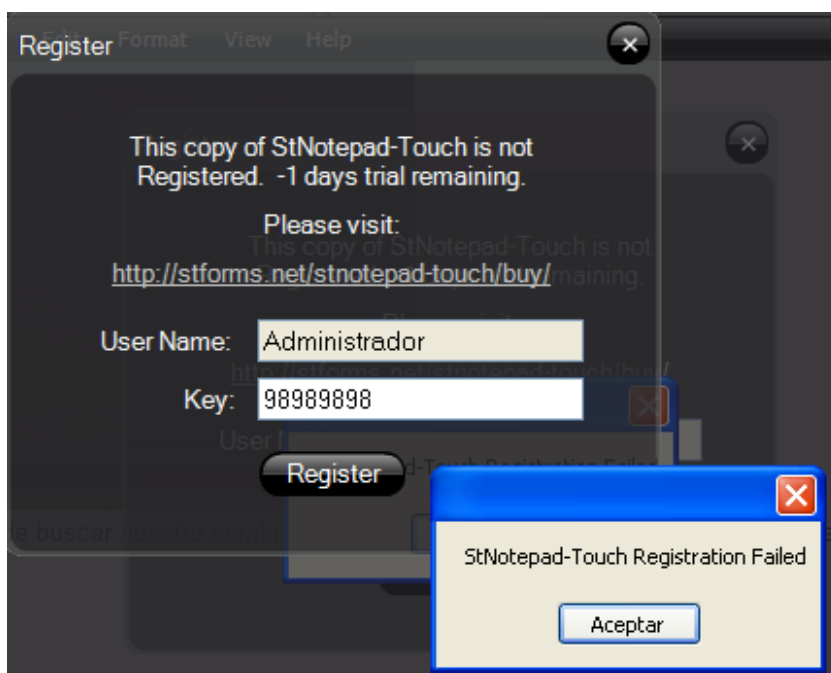
Después de su instalación, y arrancar el programa, tendremos nuestra interface recordándonos que no lo hemos pagado.



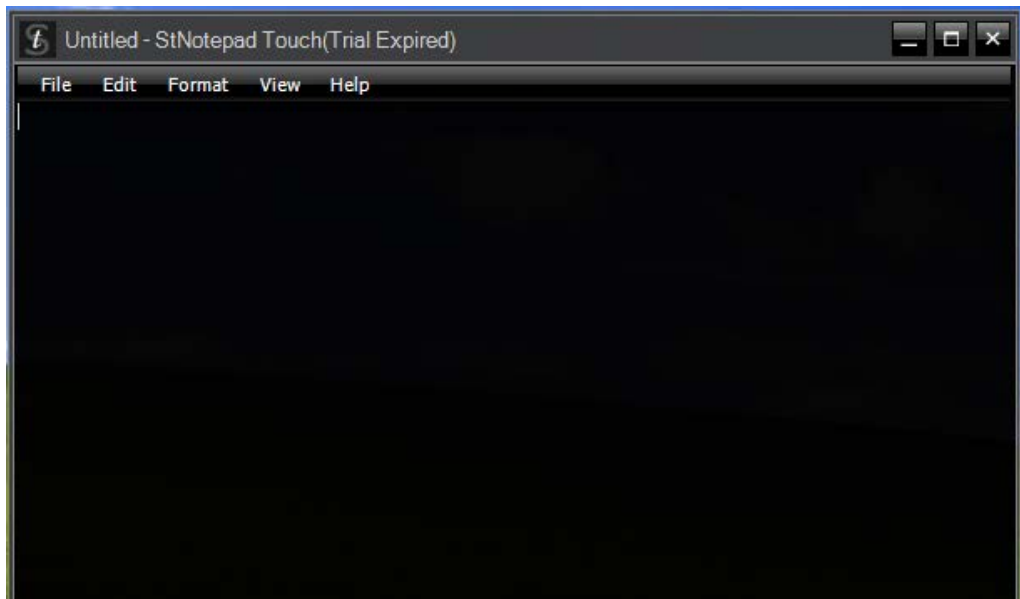
Y al buscar la opción de registrarlo, el mismo programa se encarga de buscar nuestro nombre en el equipo y colocarlo en la caja correspondiente.



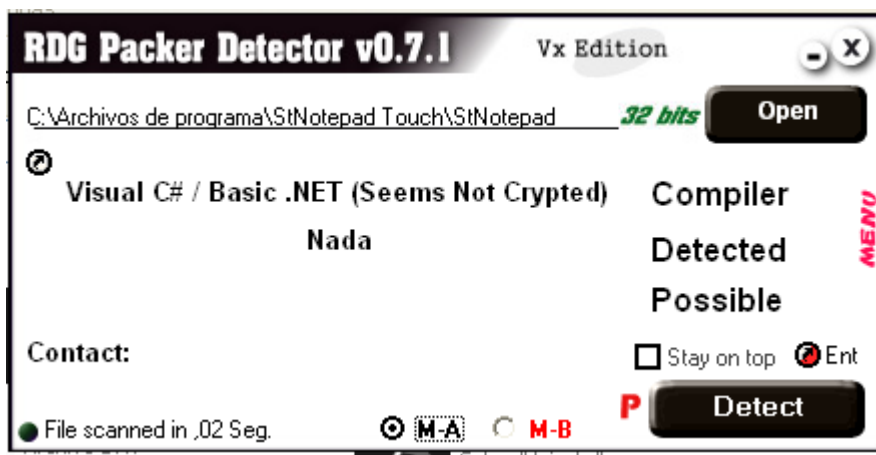
Al tratar de registrarlo:



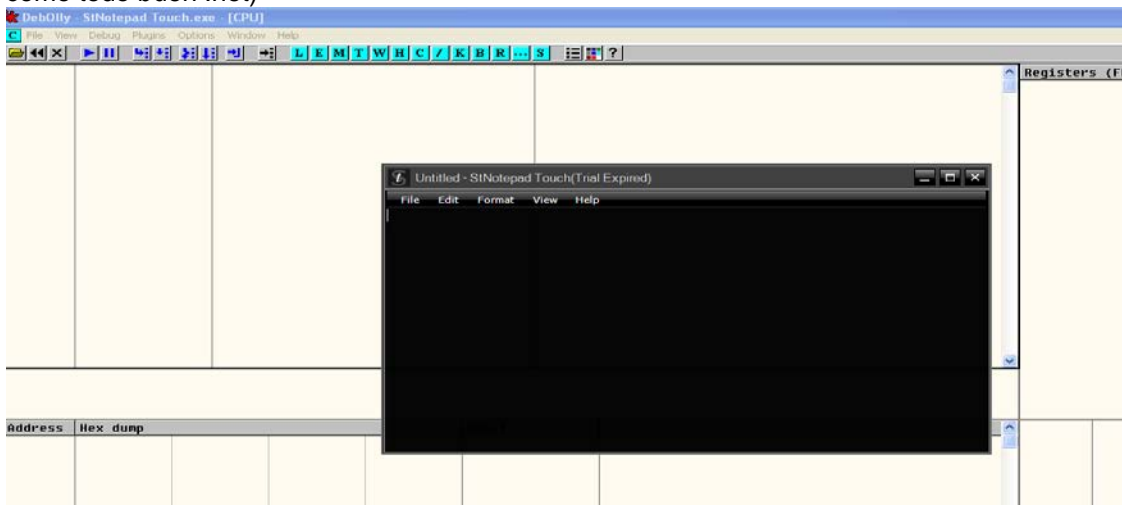
Y al adelantar la fecha para que se acabe el tiempo de prueba.



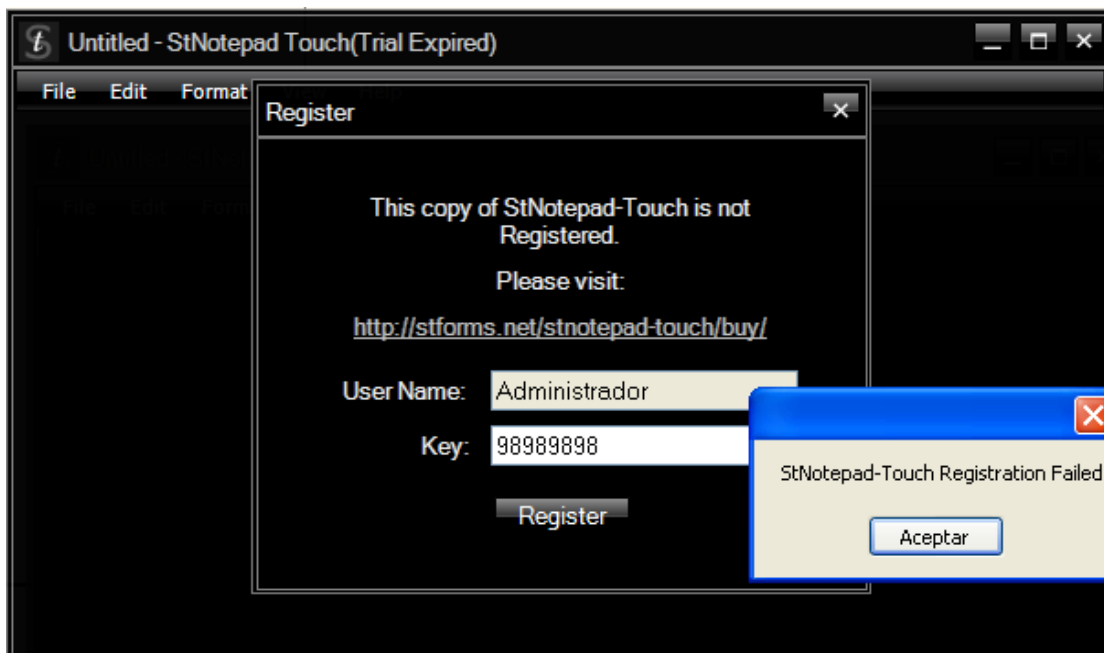
Ahora lo pasaremos por RDG.



Lo que haremos es abrirlo con olly y dejar que arranque normalmente. (Sin mostrar código ni nada como todo buen .net)



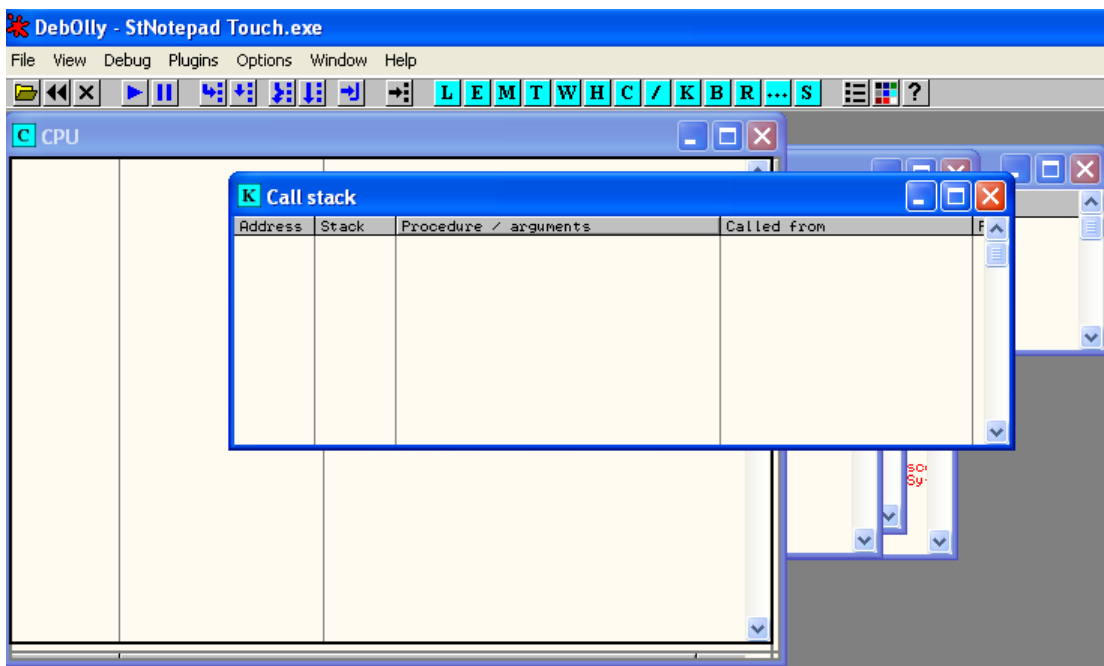
Nos iremos a la opción de registro e introduciremos los una clave pichi...  
y daremos al boton de register.



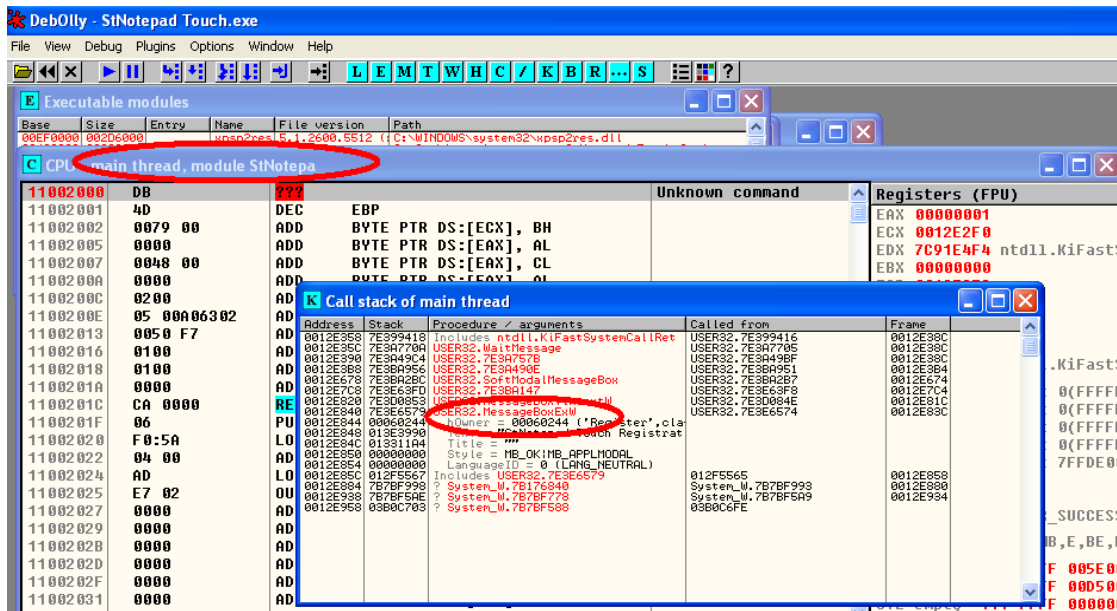
Ahora trataremos de saber qué tipo de mensaje es el que nos ha enviado el programa para poder colocar un bp.

Al colocar en pause el programa y oprimiremos "K" olly nos dirá que tipo de mensaje es.

Pero si lo hacemos en este momento en que no vemos el código del programa obtendremos esto:

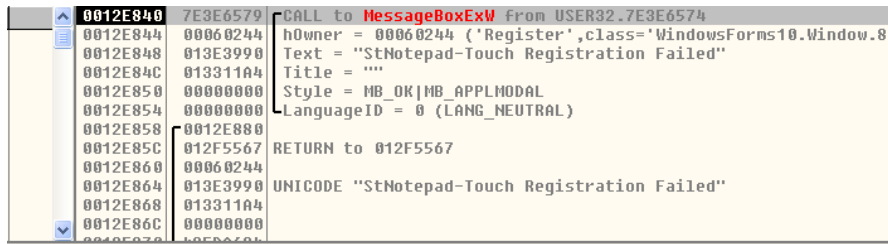
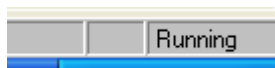


Si ahora nos vamos a "E" y seleccionamos el programa que estamos trabajando obtendremos esto:

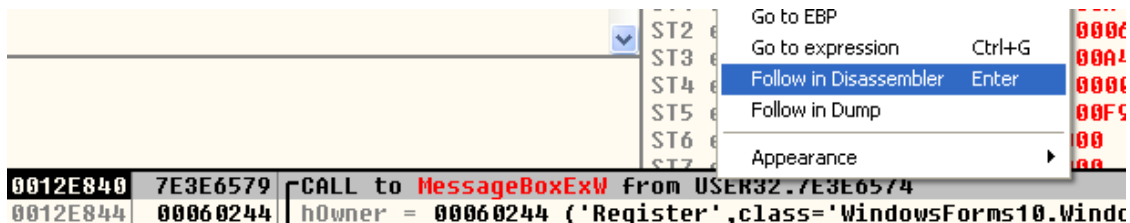


ya tenemos donde colocar nuestro bp.... :-)

Sola mente tenemos que reanudar el programa, oprimir el botón aceptar y colocar un "BP MessageBoxExW" y oprimir nuevamente register para que pique...



Tomaremos la dirección de retorno de user32 dándole click derecho y le prediremos que nos lo muestre en el desensamblado.



Donde colocaremos un bp con f2 para que pique cuando oprimamos el botón de aceptar, le damos f9 para que arranque de nuevo, oprimimos el botón aceptar y caemos en el bp que acabamos de colocar.  
y trocaremos hasta pasar 3 "RET" donde encontraremos USER32.GetActiveWindow (buscar

arriba en el código al pasar el tercer "RET")

7B7BF8F4	85C0	TEST	EAX, EAX	
7B7BF8F6	75 08	JNZ	SHORT 7B7BF900	System_W.7B7BF900
7B7BF8F8	F7C7 00002200	TEST	EDI, 220000	
7B7BF8FE	75 4E	JNZ	SHORT 7B7BF94E	System_W.7B7BF94E
7B7BF900	837D AC 00	CMP	DWORD PTR SS:[EBP-54], 0	
7B7BF904	75 3D	JNZ	SHORT 7B7BF943	System_W.7B7BF943
7B7BF906	C745 8C E4A808	MOV	DWORD PTR SS:[EBP-74], 7B08A8E4	
7B7BF90D	8965 90	MOV	DWORD PTR SS:[EBP-70], ESP	
7B7BF910	8B45 B8	MOV	EAX, DWORD PTR SS:[EBP-48]	
7B7BF913	68 25F97B7B	PUSH	7B7BF925	
7B7BF918	8F45 94	POP	DWORD PTR SS:[EBP-6C]	
7B7BF91B	C640 08 00	MOV	BYTE PTR DS:[EAX+8], 0	
7B7BF91F	FF15 10C00D7B	CALL	NEAR DWORD PTR DS:[7B0DC010]	USER32.GetActiveWindow
7B7BF925	8B4D B8	MOV	ECX, DWORD PTR SS:[EBP-48]	
7B7BF928	C641 08 01	MOV	BYTE PTR DS:[ECX+8], 1	
7B7BF92C	8B15 5413FD7A	MOV	EDX, DWORD PTR DS:[7AFD1354]	mscorlib.7A3B339C
7B7BF932	833A 00	CMP	DWORD PTR DS:[EDX], 0	
7B7BF935	74 07	JE	SHORT 7B7BF93E	System_W.7B7BF93E
7B7BF937	50	PUSH	EAX	
7B7BF938	E8 9B8698FF	CALL	7B147FD8	System_W.7B147FD8
7B7BF93D	58	POP	EAX	
7B7BF93E	8945 D4	MOV	DWORD PTR SS:[EBP-2C], EAX	
7B7BF941	EB 0B	JMP	SHORT 7B7BF94E	System_W.7B7BF94E

DS:[7B0DC010]=7E3AC2E8 (USER32.GetActiveWindow)

Por encima de estos tres saltos condicionales encontramos más llamadas a system\_w. Seleccionamos cualquiera por encima de estos, colocamos un bp con f2, oprimos f9, oprimos aceptar y registrar.

DebOilly - StNotepad Touch.exe - [CPU - main thread, module System\_W]

File View Debug Plugins Options Window Help

LEMTW H C / K B R ... S

7B7BF876	74 0C	JE	SHORT 7B7BF884	System_W.7B7BF884
7B7BF878	F7C7 00002200	TEST	EDI, 220000	
7B7BF87E	0F85 B79C2800	JNZ	7BA4953B	System_W.7BA4953B
7B7BF884	0FB645 08	MOVZX	EAX, BYTE PTR SS:[EBP+8]	
7B7BF888	85C0	TEST	EAX, EAX	
7B7BF88A	74 0C	JE	SHORT 7B7BF898	System_W.7B7BF898
7B7BF88C	F7C7 00002200	TEST	EDI, 220000	
7B7BF892	0F85 E89C2800	JNZ	7BA49580	System_W.7BA49580
7B7BF898	F7C7 FFFFE7FF	TEST	EDI, FFFFE7FF	
7B7BF89E	74 0E	JE	SHORT 7B7BF8AE	System_W.7B7BF8AE
7B7BF8A0	E8 AB9DA0FF	CALL	7B1C9650	System_W.7B1C9650
7B7BF8A5	80C8	MOV	ECX, EAX	
7B7BF8A7	3909	CMP	DWORD PTR DS:[ECX], ECX	
7B7BF8A9	E8 2A4A98FF	CALL	7B1442D8	System_W.7B1442D8
7B7BF8AE	E8 8D7AA0FF	CALL	7B1C7340	System_W.7B1C7340
7B7BF8B3	80C8	MOV	ECX, EAX	
7B7BF8B5	3909	CMP	DWORD PTR DS:[ECX], ECX	
7B7BF8B7	E8 1C4A98FF	CALL	7B1442D8	System_W.7B1442D8
7B7BF8BC	0FB645 08	MOVZX	EAX, BYTE PTR SS:[EBP+8]	
7B7BF8C0	85C0	TEST	EAX, EAX	
7B7BF8C2	75 04	JNZ	SHORT 7B7BF8C8	System_W.7B7BF8C8
7B7BF8C4	33C0	XOR	EAX, EAX	
7B7BF8C6	EB 05	JMP	SHORT 7B7BF8CD	System_W.7B7BF8CD

Jump is taken  
7B7BF884=System\_W.7B7BF884

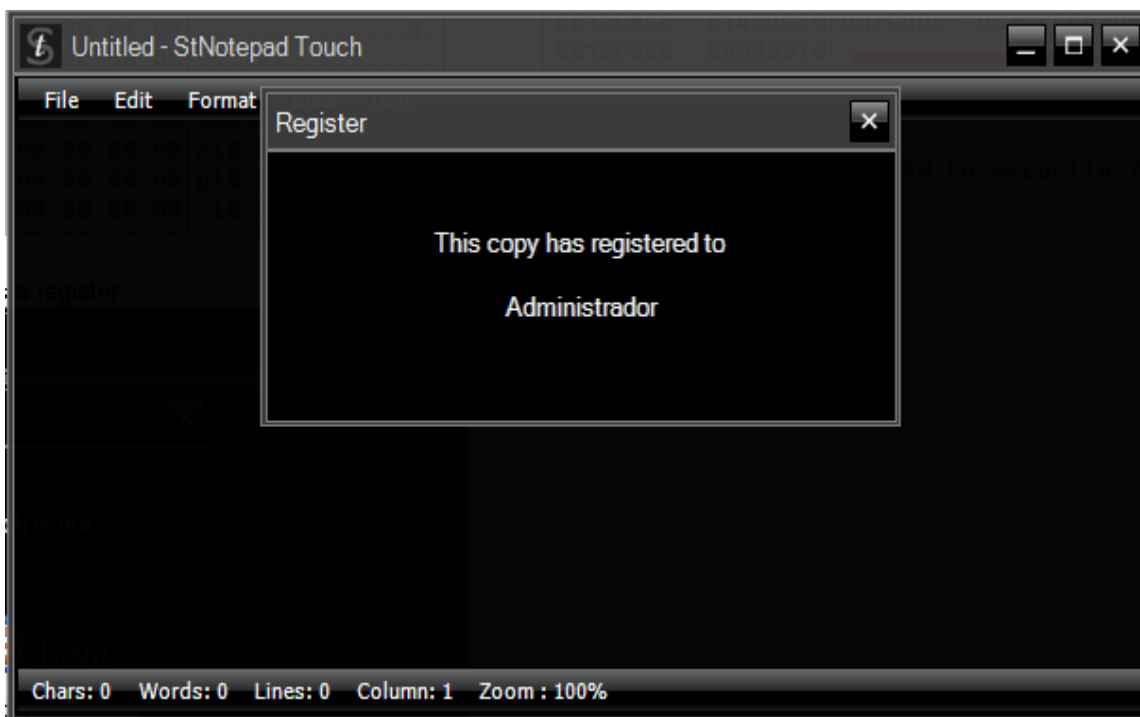
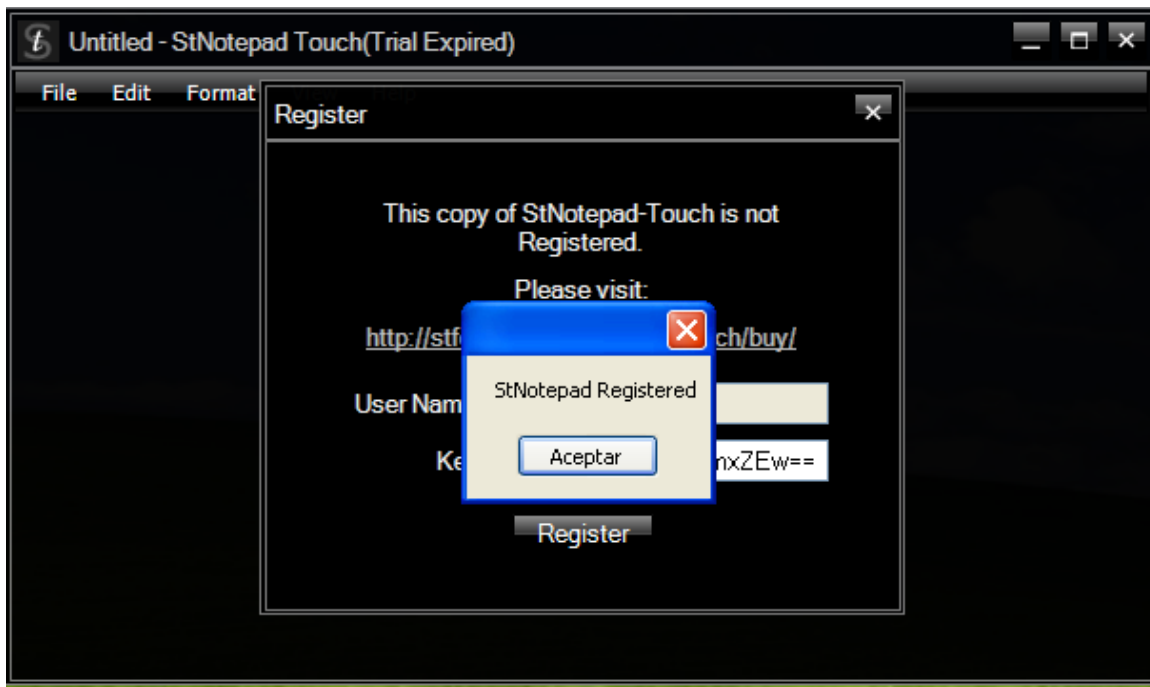
Registers (FPU)

EAX 00000001  
ECX 009C515C  
EDX 00000000  
EBX 0140CFC0  
ESP 0012E8B4  
EBP 0012E934  
ESI 01400C7C  
EDI 00000000  
EIP 7B7BF876 System\_W.7B7BF876  
C 0 ES 0023 32bit 0(FFFFFFFF)  
P 1 CS 001B 32bit 0(FFFFFFFF)  
A 0 SS 0023 32bit 0(FFFFFFFF)  
Z 1 DS 0023 32bit 0(FFFFFFFF)  
S 0 FS 003B 32bit 7FFDE000(FFF)  
T 0 GS 0000 NULL  
D 0  
0 0 LastErr ERROR\_SUCCESS (00000000)  
EFL 00000246 (NO,NO,E,OE,NS,PE,GE,LE)  
ST0 empty -NaN FFFF FF000000 FF000000  
ST1 empty -??? FFFF 065600A4 00A400A4  
ST2 empty -??? FFFF 00F90006 00060006  
ST3 empty -??? FFFF 065600A4 00A400A4  
ST4 empty -??? FFFF 00000000 00000000  
ST5 empty -??? FFFF 00F900F9 00F900F9

Address	Hex dump	ASCII
11076080	00 00 00 00 00 00 01 00 01 00 00 00 D8 00 00 00	.....0.....
11076090	00 00 00 00 00 00 00 00 00 00 00 00 00 01 00	.....0.....
110760A0	00 00 00 00 F0 00 00 00 00 00 00 00 00 00 00	.....0.....
110760B0	00 00 00 00 00 00 01 00 00 00 00 00 00 01 00	.....0.....
110760C0	00 00 00 00 00 00 00 00 00 00 00 00 00 01 00	.....0.....
110760D0	00 00 00 00 00 01 00 00 00 00 00 00 00 00 00	.....0.....
110760E0	00 00 00 00 00 00 01 00 00 00 00 00 20 01 00 00	.....0.....
110760F0	30 61 07 00 28 08 01 00 00 00 00 00 00 00 00	0a1.(00.....
11076100	58 69 08 00 14 00 00 00 00 00 00 00 00 00 00	Xi.....
11076110	70 69 08 00 48 03 00 00 00 00 00 00 00 00 00	pi..H.....
11076120	B8 6C 08 00 EA 01 00 00 00 00 00 00 00 00 00	.1..a.....

0012E8B4 425DA6C4  
0012E8B8 79E7A6B8 mscorwks.79E7A6B8  
0012E8BC 0012EA3C  
0012E8C0 00000001  
0012E8C4 00000000  
0012E8C8 0140DC20  
0012E8CC 00000010  
0012E8D0 0140DC14  
0012E8D4 0140DC00  
0012E8D8 0012E934  
0012E8DC 7929CC8E RETURN to mscorlib.7929CC8E from mscorlib.7929CD10  
0012E8E0 00000000  
0012E8E4 013F30A4

Cuando nuestro bp pica tendremos el serial en el la pila.....  
Lo copiamos y pegamos en la caja correspondiente y le damos a register



Nota:

Al realizarlo en window 7 cada vez que coloquemos un bp nos saldrá un cartel al cual le daremos que sí para que el bp se marque.

Para ver la dirección de retorno del mensaje posiblemente sea necesario seleccionar el `_CorExemain(mscore.dll)` (**Evadiendo StrongNames de .NET con OllyDbg - por marciano**)

y no stnotepad ,pero sin la necesidad de parar en ella al arrancar. Solamente seleccionarla para que muestre la ventana después de tratar de registrarnos.

En xp no tenemos mucho problema para los bp, (al introducir la clave mala, introducir números y letras ,pj:"989898hola", en una prueba que hice con solo números me aceptó la clave y se registró, pero no se desbloqueó el programa, ni dejó de mostrar el mensaje trial, tampoco daba la opción de registrarlo nuevamente ya que la clave había sido almacenada como buena )

