

## S3CC PROG por joehack

Acá está el primer tutorial de mi autoría, (la verdad no lo creo), Debo reconocer que en esto me dio toda información =InDuLgEo= **Gracias Maestro.**

### Descripción de la víctima

Este programa es una aplicación electrónica, hecho el Borland Delphi, 6, permite reprogramar el contenido las memorias de los chips de las nuevas impresoras, ya verán de que marca son.

Utilizaremos el OLLY DEBUG, para meternos en las entrañas de este proggie.

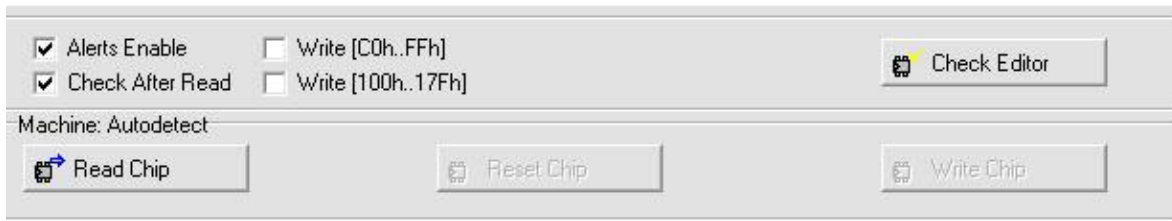
Enlace de descarga: <http://www.megaupload.com/?d=ZQZSL8HW>

### Proceso

Con la herramienta RDG Packer Detector v0.6.6, podemos fácilmente conocer cuál es el compilador que se usó para esta aplicación, en este caso es el Borland Delphi 6, se detecta que no tiene ningún otro empaquetador, por lo que podemos ir directamente a destripar al proggie.

Este programa no tiene ningún serial que ingresar, simplemente están bloqueadas varias funciones, en la descripción del programa miramos





De modo que ya sabemos que es lo que vamos a atacar.

Procedemos a abrir el programa con el Olly Dbg y se para en el OEP - Original Entry Point- (abierto sin el olly te sale en la form "Demo Mode")

Sobre el decompilado buscamos las “all referenced strings”

Buscamos en donde este escrito el “demo mode” y caemos en este lugar

```

00481F11 | . 8D45 BC      LEA EAX,[LOCAL.18]
00481F14 | . BA E8204800   MOV EDX,S3CCProg.004820F8
00481F19 | . E8 062EF8FF   CALL 00404D24
00481F1E | . 8B55 BC      MOV EDI,DWORD PTR SS:[LOCAL.18]
00481F21 | . 8BC3      MOV EBX,EBX

```

Subimos un poco más arriba y podemos ver esto...

```

00481E80 | . EB 07      JMP SHORT 00481E96
00481E8F | . C683 78040000 00 MOV BYTE PTR DS:[EBX+478],0
00481E96 | . 80BB 78040000 00 CMP BYTE PTR DS:[EBX+478],0
00481E9D | . 75 5C      JNE SHORT 00481EFB
00481E9F | . 8D55 C4    LEA EDI,[LOCAL.16]
00481EA2 | . 8BC3      MOV EBX,EBX

```

Ponemos un BP -Breake point- normal con el F2, y echamos a correr el programa con F9 y se para en el BP, debajo del olly encontramos esta descripción.

```

00481ED5 | . 8B55 C8    MOV E
00481ED8 | . 8BC3      MOV E
Imm=00
[00481F24]=01
Jumps from 481E45,481E8D

```

Una breve descripción de este punto: Cuando corremos el programa, el resultado de esta comparación deberá ser igual a 0, debido a la condición de la siguiente instrucción, que dice: JNE (salta si no es igual -0-), como vemos que el resultado es 1, coge el salto siguiente y se va a demo.

Esto es importante, nos dice que este byte en esa posición está a 01 luego en el BP lo compara con 00 y como son diferentes nos manda con el JNE al demo, lo entiendes?

Lo que haremos será editar esta línea para poner este byte a =0 editamos y ponemos donde esta el BP en esa línea esto

EL valor binario de [78040000 00](#) lo colocamos en [78040000 01](#) y(selecciona cmp byte, click derecho, edit, binary edit) el JNE llenamos con NOPS. (selecciona el JNE, le das Boton derecho, fill with nops)

00481E8B	EB 87050000	CALL 00482444
00481E8D	EB 07	JMP SHORT 00481E96
00481E8F	C683 78040000 00	MOV BYTE PTR DS:[EBX+478],0
00481E96	80BB 78040000 01	CMP BYTE PTR DS:[EBX+478],1
00481E9D	90	NOP
00481E9E	90	NOP
00481E9F	8D55 C4	LEA EDX,[LOCAL.16]
00481EA2	8BC3	MOV EAX,EBX

Luego seleccionas todo este espacio editado, click derecho copy to executable y le pones otro nombre o el mismo como quieras

Sal del Olly y dale al programa y que sale???



No se abre

Explicación:

El programa hace un CRC o sea comprueba que todos los bytes estén igual q el original y como hemos cambiado bytes, pues lo detecta y sale este bello mensaje.

Tomar en cuenta que el Byte clave es : 00481E96 80BB [78040000 00](#) CMP BYTE PTR DS:[EBX+478],0

Nos encontramos que el programa tiene muchas comprobaciones y sale por todos los sitios esto "D104 allowed in Registered version only. Exited.."}

<pre> ASCII "Machine: " ASCII " selected." MOV EDX,S3CCProg.00496014 PUSH S3CCProg.00496050 PUSH S3CCProg.00496064 MOV EDX,S3CCProg.00496050 ASCII "-D104 allowed in" ASCII " Registered users" </pre>	<pre> ASCII "-D104 allowed in Registered version only. Exited..." ASCII "Machine: " ASCII " selected." ASCII "Machine: " </pre>
--	---

No podemos ir cambiando todo pues es trabajoso pero bueno lo que haremos será ver donde pone este byte a 1.

Abrimos el Olly otra vez y cargamos otra vez el programa pero el original sin modificar.

Vamos otra vez a la zona de demo o sea donde estaba el CMP y un poco hacia arriba hasta ver esto

```
00481DE0 | . 8B08          | MOV ECX,DWORD PTR DS:[EAX]
00481DEF | . FF51 64        | CALL DWORD PTR DS:[ECX+64]
00481DF2 | . C683 78040000 01 | MOV BYTE PTR DS:[EBX+478],1
00481DF9 | . A1 C8184B00    | MOV EAX,DWORD PTR DS:[4B18C8]
00481DFF | . 8B08          | MOV ECX,DWORD PTR DS:[EAX]
```

Aqui es donde pone el byte a 1, si te fijas en el mismo EBX+478 del CMP !!!

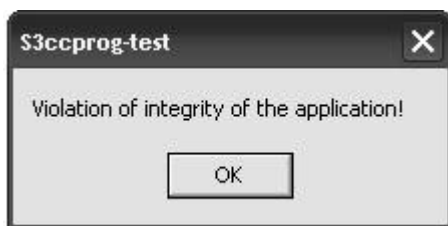
Editamos asi :

```
00481DE0 | . 8B08          | MOV ECX,DWORD PTR DS:[EAX]
00481DEF | . FF51 64        | CALL DWORD PTR DS:[ECX+64]
00481DF2 | . C683 78040000 00 | MOV BYTE PTR DS:[EBX+478],0
00481DF9 | . A1 C8184B00    | MOV EAX,DWORD PTR DS:[4B18C8]
00481DFF | . 8B08          | MOV ECX,DWORD PTR DS:[EAX]
```

MOV BYTE PTR DS:[EBX+478],0

Editamos y guardamos los cambios y le ponemos otro nombre

Salimos del olly e iniciamos a ver si arranca ???



Violation of integrity....

Recuerdas acerca del CRC ?

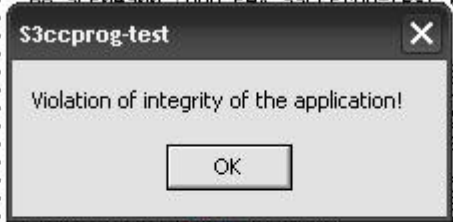
El programa hace un CRC o sea comprueba que todos los bytes esten igual q el original,,y como hemos cambiado bytes,,pues lo detecta y bye...

Cargamos al Olly este modificado y vamos por la parte del CRC ,tenemos q ver donde esta este CRC dale con F8 todo el rato hasta q te aparezca el mensaje

```

0049E28F . B8 3CF04900 IMUL EAX,S3CCProg-test.0049E03C
0049E294
0049E299
0049E29B
0049E29C
0049E2A1
0049E2A4
0049E2A7
0049E2AC
0049E2AE
0049E2B3
0049E2B8
0049E2BA
0049E2C1 . E8 C6FCFFFF CALL 0049DF8C
0049E2C6 . 84C0 TEST AL,AL
0049E2C8 . 75 17 JNE SHORT 0049E2E1
0049E2CA . 8D55 EC LEA EDI,[EBP-14]
0049E2CD . A1 E4194000 MOV EAX,DWORD PTR DS:[4A19E4]
0049E2D2 . E8 B989F6FF CALL 00406C90
0049E2D7 . 8B45 EC MOV EAX,DWORD PTR SS:[EBP-14]
0049E2DA . E8 89A4F9FF CALL 00438768
0049E2DF . EB 24 JMP SHORT 0049E305
0049E2F1 . 8B40 FC164000 MOV ECX,DWORD PTR DS:[4A16FC]

```



En este call, se para el Olly y sale el mensaje  
 0049E2DA |. E8 89A4F9FF CALL S3CCProg.00438768

Si te fijas arriba esta un salto  
 0049E2C8 |. /75 17 JNZ SHORT S3CCProg.0049E2E1

Y antes un 0049E2C6 |. 84C0 TEST AL,AL o sea tenemos q poner AL =1,

Reinicia el Olly y con F8 hasta aquí

```

0049E2B8 . 8B00 MOV EAX,DWORD PTR DS:[
0049E2BA . C740 74 4C1D MOV DWORD PTR DS:[EAX+
0049E2C1 . E8 C6FCFFFF CALL 0049DF8C
0049E2C6 . 84C0 TEST AL,AL
0049E2C8 . 75 17 JNE SHORT 0049E2E1

```

Ahora F7 para entrar en el CALL y vemos esto

```

0049DF8C . 55 PUSH EBP
0049DF8D . 8BEC MOV EBP,ESP
0049DF8F . 6A 00 PUSH 0
0049DF91 . 53 PUSH EBX
0049DF92 . 33C0 XOR EAX,EAX
0049DF94 . 55 PUSH EBP
0049DF95 . 68 CFDF4900 PUSH S3CCProg-test.0049DFCF
0049DF9A . 64:FF30 PUSH DWORD PTR FS:[EAX]
0049DF9D . 64:8920 MOV DWORD PTR FS:[EAX],ESP

```

Aquí está rutina se encarga del CRC traceamos con F8 hasta

```

0049DFD4 . EB F0 JMP SHORT 004
0049DFD6 . 8BC3 MOV EAX,EBX
0049DFD8 . 5B POP EBX
0049DFD9 . 59 POP ECX
0049DFDA . 5D POP EBP
0049DFDB . C3 RETN
0049DFDC . 55 PUSH EBP

```

Aquí si te fijas debajo esta el RET que vuelve bien, debemos poner a eax == 1  
 párate aquí

0049DFD6 . 8BC3 MOV EAX,EBX

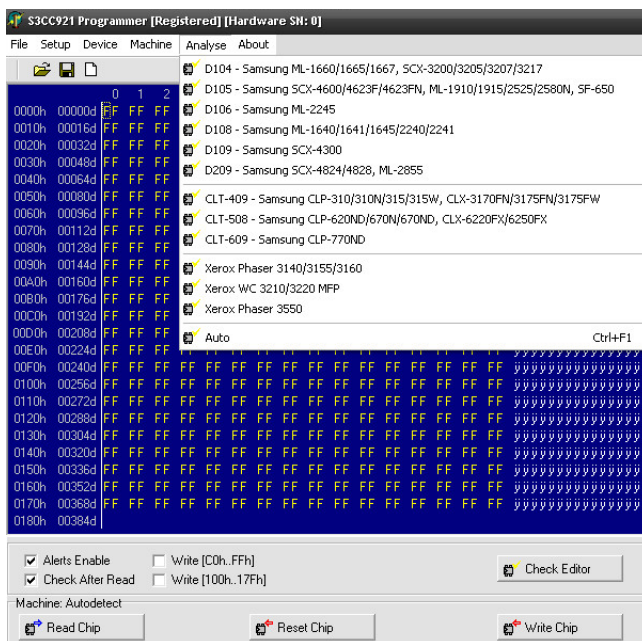
Pero atención, no podemos poner un MOV EAX,1, porque ocupa más bytes y nos descuadra el código, por lo tanto solo ponemos al registro de 8 bytes AL editamos así:



0049DFD6 B0 01 MOV AL,1

Al poner a AL=1 == EAX =1 al salir del RET y hacer el TEST AL, AL, 1, no cojerá el salto y no mostrara el mensaje y con eso saltamos el CRC.

Guardamos los cambios y salimos del Olly, ahora los botones están habilitados y no sale ningún error.



Hemos visto:

1. Habilitar los botones con el byte a 00481DF2 C683 [78040000 00](#) MOV BYTE PTR DS:[EBX+478],0
2. Saltar el CRC para poder cambiar Bytes

En realidad este programilla, al principio parecía que iba a volvernors locos, por eso de la comprobación (CRC), pero husmeando un poco en las entrañas del programa, a punta de F2, F7, F8, Y F9 en el Olly Debug, TRIUNFAMOS.... jejejejeje