

# Other day other malware in the way (died.exe)

[blog.cyttek.com/2019/08/28/other-day-other-malware-in-the-way-died-exe/](http://blog.cyttek.com/2019/08/28/other-day-other-malware-in-the-way-died-exe/)

Rafael Revert

August 28, 2019



ATM



## Rafael Revert

CTO and Co-founder of Cyttek Group and international consulting company specialized in providing Cyber Security , ATM, IA, Big Data and custom products for different sectors

[More posts](#) by Rafael Revert.



## Rafael Revert

28 Aug 2019 • 5 min read



Most of cyber security blogs will talk about the discovery of the malware in english to reach more people in the explanation and make a fancy claims that they have the solution to protect everything , pero no nosotros no parte de esta misión de este blog es centralizar conocimiento de varios tópicos de seguridad pero centrados en canales alternos y en especial ATM que hemos seguido este tópicó, en este caso vamos a hablar de un nuevo software descubierto en latino américa Died.exe o dd por los strings encontrados.

Primero que todo o existe un faltante al momento de la cuenta de efectivo por la transportadora de valores o hay un software nuevo que no debería estar en el ATM.

Died puede ser ejecutado desde cualquier carpeta del sistema operativo

equipo > Escritorio > raro

Nombre	Tipo	Tamaño
 died	Aplicación	1,082 KB

Generalmente lo primero es que el Software tiene referencias a funciones (service provider) SPI y importaciones de DLLs propias del CEN XFS, para poder ver si ya de primero tenemos suficiente motivo para que no esté en el ATM

```

.idata:00573444      extrn EndSystemLocalesEx:dword ; CODE XREF: sub_4FCCB0+11f
.idata:00573444      ; DATA XREF: sub_4FCCB0+11f ...
.idata:00573448 ; DWORD __stdcall SetFilePointer(HANDLE hFile, LONG lDistanceToMove, PLONG lpDistanceToMoveHigh, DWORD dwMoveMethod)
.idata:00573448      extrn SetFilePointer:dword ; CODE XREF: sub_4FFE40+631f
.idata:00573448      ; DATA XREF: sub_4FFE40+631f ...
.idata:0057344C ; DWORD __stdcall GetTimeZoneInformation(LPTIME_ZONE_INFORMATION lpTimeZoneInformation)
.idata:0057344C      extrn GetTimeZoneInformation:dword
.idata:0057344C      ; CODE XREF: sub_503420+1661f
.idata:0057344C      ; DATA XREF: sub_503420+1661f ...
.idata:00573450 ; BOOL __stdcall SetStdHandle(DWORD nStdHandle, HANDLE hHandle)
.idata:00573450      extrn SetStdHandle:dword ; CODE XREF: sub_506310+841f
.idata:00573450      ; sub_506310+901f ...
.idata:00573454 ; BOOL __stdcall ReadConsoleW(HANDLE hConsoleInput, LPVOID lpBuffer, DWORD nNumberOfCharsToRead, LPDWORD lpNumberOfCharsRead)
.idata:00573454      extrn ReadConsoleW:dword ; CODE XREF: sub_5072C0+7691f
.idata:00573454      ; sub_5072C0+7691f ...
.idata:00573458      ; DATA XREF: sub_5072C0+7691f ...
.idata:005734CC ; Imports from MSXFS.dll
.idata:005734CC ;
.idata:005734CC      extrn __imp_WFSExecute:dword ; DATA XREF: WFSExecute1r
.idata:005734D0      extrn __imp_WFSStartup:dword ; DATA XREF: WFSStartup1r
.idata:005734D4      extrn __imp_WFSOpen:dword ; DATA XREF: WFSOpen1r
.idata:005734D8      extrn __imp_WFSGetInfo:dword ; DATA XREF: WFSGetInfo1r
.idata:005734DC
.idata:005734DC
.idata:005734DC      end start

```

Luego de segundo es encontrar posibles rutinas o subrutinas que puedan interaccionar con algún perimetral siguiendo la lógica del WFS primero debe inicializar , luego abrir la comunicación

```

.text:0047B015      mov     [ebp+var_10], eax
.text:0047B018      push   eax
.text:0047B019      lea    eax, [ebp+var_C]
.text:0047B01C      mov     large fs:0, eax
.text:0047B022      mov     [ebp+var_18], ecx
.text:0047B025      lea    ecx, [ebp+var_66C]
.text:0047B02B      call   sub_46B406
.text:0047B030      mov     [ebp+var_4], 0
.text:0047B037      push   offset sub_46D8B4
.text:0047B03C      push   offset unk_5703E0 ; "[+] WFSStartup()"
.text:0047B041      push   eax
.text:0047B046      call   sub_46E2BE
.text:0047B04B      add    esp, 8
.text:0047B04E      mov     ecx, eax
.text:0047B050      call   sub_46E30E
.text:0047B055      lea    eax, [ebp+var_228]
.text:0047B05B      push   eax
.text:0047B05C      push   0B020003h
.text:0047B061      call   j_WFSStartup
.text:0047B066      mov     [ebp+var_678], eax
.text:0047B06C      cmp     [ebp+var_678], 0
.text:0047B073      jz     loc_47B138
.text:0047B079      push   offset asc_54A58C ; ")"
.text:0047B07E      mov     eax, [ebp+var_678]
.text:0047B084      push   eax
.text:0047B085      lea    ecx, [ebp+var_8A0]
.text:0047B08B      push   ecx
.text:0047B08C      mov     ecx, [ebp+var_18]
.text:0047B08F      call   sub_46C473
.text:0047B094      mov     [ebp+var_8A8], eax
.text:0047B09A      mov     edx, [ebp+var_8A8]
.text:0047B0A0      mov     [ebp+var_8AC], edx

```

Luego en otra subrutina vemos que tiene que ejecutar la instrucción de apertura del canal SPI

```

add     esp, 8
mov     ecx, eax
call   sub_46E30E
mov     byte ptr [ebp+var_4], 3
lea     ecx, [ebp+var_87C]
call   sub_46C293
mov     byte ptr [ebp+var_4], 2
lea     ecx, [ebp+var_858]
call   sub_46C293
mov     byte ptr [ebp+var_4], 0
lea     ecx, [ebp+var_834]
call   sub_46C293
mov     [ebp+var_684], 0
mov     [ebp+var_690], 0
mov     [ebp+var_69C], offset unk_54955D
push   offset sub_46D8B4
push   offset asc_54A58C ; ")"
mov     eax, [ebp+arg_0]
push   eax
push   offset aWfsopen ; "[+] WFSopen("
push   offset unk_5703E0
call   sub_46E2BE
add     esp, 8
push   eax
call   sub_46E2BE
add     esp, 8
push   eax
call   sub_46E2BE
add     esp, 8
mov     ecx, eax
call   sub_46E30E
mov     eax, [ebp+var_18]
add     eax, 1Ch
push   eax
lea     ecx, [ebp+var_648]
push   ecx
lea     edx, [ebp+var_438]
push   edx
push   0B020003h
mov     eax, [ebp+var_18]
mov     ecx, [eax+10h]
push   ecx
mov     edx, [ebp+var_690]

```

si toda el proceso de abertura no está ocupado por otro servicio esto quiere decir que entonces el usuario que lo ejecuta de acuerdo a la necesidad del CEN XFS no puede estar corriendo otro hserv para este servicio SPI por lo tanto el ATM debe estar sin otro servicio XFS para poder ejecutar el software

si el hserv se vuelve único y avanza el proceso del WFS open por la subrutina 46D8B4

```

mov [ebp+var_678], eax
cmp [ebp+var_678], 0
jz loc_47B3F8

push offset asc_54A58C ; ")"
mov eax, [ebp+var_678]
push eax
lea ecx, [ebp+var_800]
push ecx
mov ecx, [ebp+var_18]
call sub_46C473
mov [ebp+var_808], eax
mov edx, [ebp+var_808]
mov [ebp+var_80C], edx
mov byte ptr [ebp+var_4], 1
mov ecx, [ebp+var_80C]
call sub_46E2E9
push eax
push offset asc_54A590 ; "("
mov eax, [ebp+var_678]
push eax
push offset loc_46DDE1
push offset aWfsStartupRetu ; "[!] WfsStartup() return: 0x"
push offset unk_5703E0
call sub_46E29E
add esp, 8
mov ecx, eax
call sub_46B2E9
mov ecx, eax
call sub_46B938
push eax
call sub_46E2BE
add esp, 8
push eax
call sub_46E2BE
add esp, 8
mov byte ptr [ebp+var_4], 0
lea ecx, [ebp+var_800]
call sub_46C293
xor eax, eax
mov [ebp+var_77E], ax
mov [ebp+var_77E], 0FFFFFFFh
lea ecx, [ebp+var_80C]
call sub_46C293
mov ax, [ebp+var_77E]
jmp loc_47B5A2

offset sub_46D884
mov eax, [ebp+var_678]
push eax
lea ecx, [ebp+var_810]
push ecx
mov ecx, [ebp+var_18]
call sub_46C473
mov [ebp+var_808], eax
mov edx, [ebp+var_808]
mov [ebp+var_80C], edx
mov byte ptr [ebp+var_4], 5
mov ecx, [ebp+var_80C]
call sub_46E2E9
push eax
push offset aStrerror ; " strError: "
mov eax, [ebp+var_678]
push eax
push offset loc_46DDE1
push offset aWfsOpenRetu ; "[!] WfsOpen() return: 0x"
push offset unk_5703E0
call sub_46E29E
add esp, 8
mov ecx, eax
call sub_46B2E9
mov ecx, eax
call sub_46B938
push eax
call sub_46E2BE
add esp, 8
mov ecx, eax
call sub_46E2BE
add esp, 8
mov byte ptr [ebp+var_4], 0
lea ecx, [ebp+var_810]
call sub_46C293
xor eax, eax
mov [ebp+var_772], ax
mov [ebp+var_772], 0FFFFFFFh
lea ecx, [ebp+var_80C]
call sub_46C293
mov ax, [ebp+var_772]
jmp loc_47B5A2

loc_47B3F8:
push offset sub_46D884
movzx eax, [ebp+var_640]
push eax
lea ecx, [ebp+var_704]
push ecx
call sub_46BCF3
add esp, 8
mov [ebp+var_808], eax
mov edx, [ebp+var_808]
mov [ebp+var_80C], edx
mov byte ptr [ebp+var_4], 6
mov eax, [ebp+var_80C]
push eax
push offset a_ ; "
movzx ecx, [ebp+var_640]
push ecx
lea edx, [ebp+var_70C]
push edx
call sub_46BCF3
add esp, 8
mov [ebp+var_808], eax
mov ecx, [ebp+var_808]
mov [ebp+var_80C], ecx
mov byte ptr [ebp+var_4], 7
mov ecx, [ebp+var_80C]
push ecx
push offset a_ ; "
movzx ecx, [ebp+var_640]
push ecx
lea edx, [ebp+var_70C]
push edx
call sub_46BCF3
add esp, 8
mov [ebp+var_808], eax
mov ecx, [ebp+var_808]
mov [ebp+var_80C], ecx
mov byte ptr [ebp+var_4], 8
mov edx, [ebp+var_80C]
push edx
push offset aSpiVersion ; "\n[*] SPI Version: "
lea eax, [ebp+var_541]
push eax
push offset aSpiSystemStatus ; "\n[*] SPI System Status: "
lea ecx, [ebp+var_640]
push ecx
push offset aSpiDescription ; "\nSPI Description: "
mov edx, [ebp+var_18]
movzx eax, word ptr [edx+1Ch]
push eax
push offset loc_46DDE1
push offset aHandleDispense ; "[*] Handle Dispense 0x"
call sub_5703E0
add esp, 8
mov ecx, eax
call sub_46B2E9
mov ecx, eax
call sub_46B94C
push eax
call sub_46E2BE
add esp, 8
push eax

```

32.10) 00010418 0047B018: sub 47AFE0+38 (Svnsynchronized with Hex View-1)

revisa que en la subrutina se encuentre el Dispensador habilitado

```

jz loc_47B3F8
push offset sub_465308
push offset asc_54A58C ; ")"
mov eax, [ebp-54h]
push eax
lea ecx, [ebp-1A4h]
push ecx
mov ecx, [ebp-18h]
call sub_46C473
mov [ebp-1ACh], eax
mov edx, [ebp-1ACh]
mov [ebp-1B0h], edx
mov byte ptr [ebp-4], 1
mov eax, [ebp-1B0h]
push eax
push offset asc_54A590 ; "("
mov ecx, [ebp-54h]
push ecx
push offset loc_46DDE1
push offset aWfsGetInfoWfs_ ; "[!] WfsGetInfo(WFS_INF_CDM_STATUS) retu"
push offset unk_5703E0
call sub_46E2BE
add esp, 8
mov ecx, eax
call sub_46B2E9
mov ecx, eax
call sub_46B938
push eax
call sub_46E2BE
add esp, 8

```

en caso de que no se encuentra habilitado la propia funcion msxf6

WFS\_CDM\_DEVONLINE le entregará el status para poder procesar con la petición de operación con el dispensador

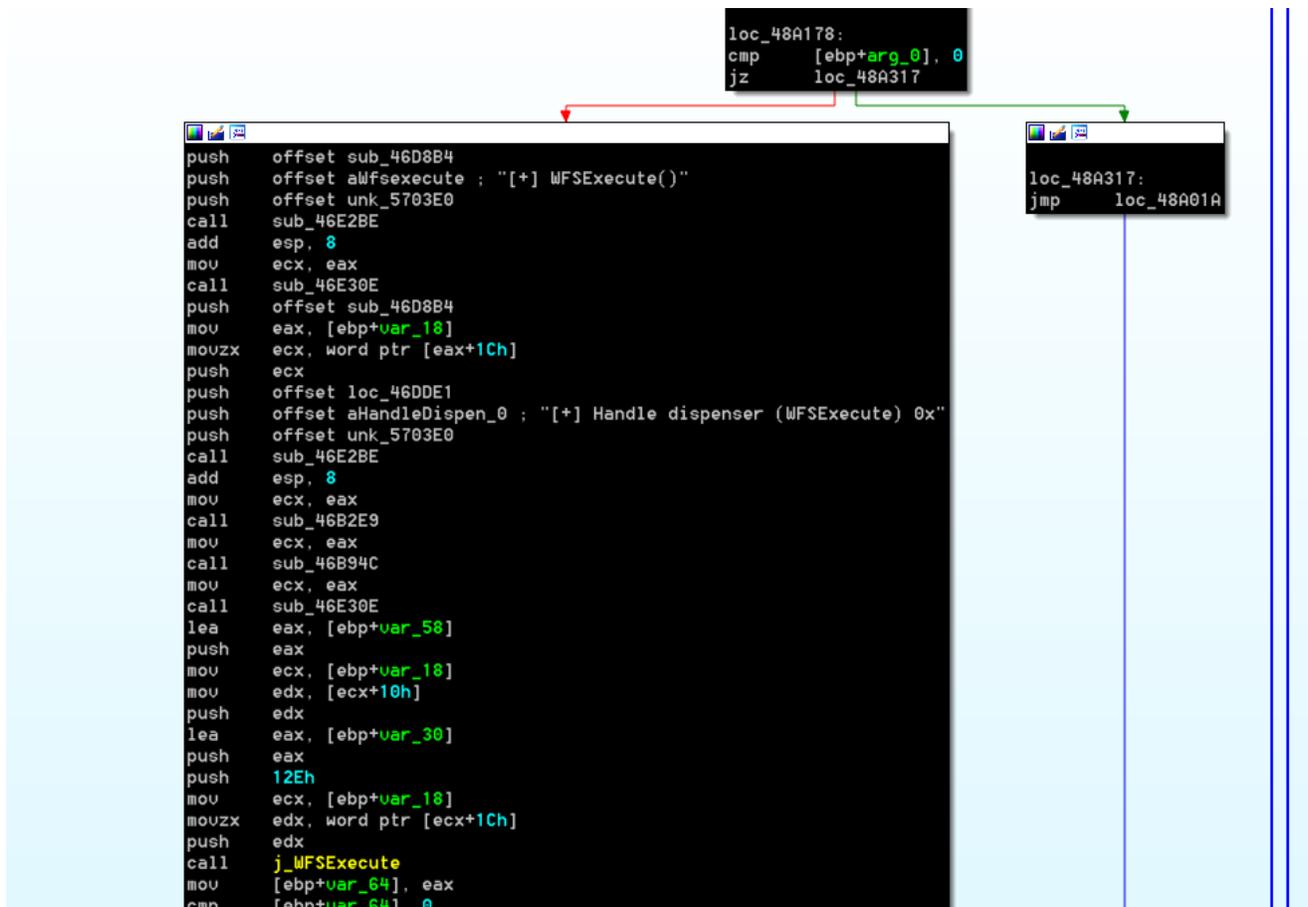
```

4 ;-----
4 loc_47ACA4: ; CODE XREF: .text:0047ABE1↑j
4     mov     eax, [ebp-48h]
7     mov     ecx, [eax+1Eh]
A     mov     [ebp-60h], ecx
D     mov     eax, [ebp-60h]
0     movzx   ecx, word ptr [eax]
3     test    ecx, ecx
5     jz     short loc_47ACF9
7     push   offset sub_463006
C     push   offset aAtmOfflineWfs_ ; "[!] ATM OFFLINE (WFS_CDM_DEVONLINE)"
1     push   offset unk_5703E0
6     call   sub_46E2BE
B     add     esp, 8
E     mov     ecx, eax
0     call   sub_46E30E
5     mov     dword ptr [ebp-150h], 0FFFFFFDh
F     mov     dword ptr [ebp-4], 0FFFFFFFh
6     lea    ecx, [ebp-3Ch]
9     call   sub_46C293
E     mov     eax, [ebp-150h]
4     jmp    loc_47AE62
9 ;-----
9 loc_47ACF9: ; CODE XREF: .text:0047ACB5↑j
9     push   offset sub_463006
E     mov     eax, [ebp-60h]
1     movzx   ecx, word ptr [eax]
4     push   ecx
5     push   offset aCdmStatus ; "CDM STATUS:."
A     push   offset unk_5703E0
F     call   sub_46E2BE
4     add     esp, 8
7     mov     ecx, eax

```

DE6: .text:0047ACE6 (Synchronized with Hex View-1)

luego entra a la función



y luego como exige el SDK del CEN llama a la función de WFS\_CMD\_CDM\_Disponse

```

ecx, [ebp+arg_4]
sub_46C293
eax, [ebp+var_1B8]
mov [ebp+var_214], eax
mov edx, [ebp+var_214]
mov [ebp+var_218], edx
mov byte ptr [ebp+var_4], 2
mov ecx, [ebp+var_218]
call sub_46BE60
push eax
push offset aStrerror_0 ; "\nstrError:: "
mov eax, [ebp+var_64]
push eax
lea ecx, [ebp+var_20C]
push ecx
call sub_46BCC1
add esp, 8
mov [ebp+var_21C], eax
mov edx, [ebp+var_21C]
mov [ebp+var_220], edx
mov byte ptr [ebp+var_4], 3
mov eax, [ebp+var_220]
push eax
push offset loc_46DDE1
push offset aWfsexecuteWfs_ ; "[!] Wfsexecute(WFS_CMD_CDM_DISPENSE) re"..].
push offset unk_5703E0
call sub_46E2BE
add esp, 8
mov ecx, eax
call sub_46B2E9
push eax
call sub_46C2BB
add esp, 8
push eax
call sub_46E2BE
add esp, 8
push eax
call sub_46E2BE
add esp, 8
mov ecx, eax
call sub_46E30E
mov byte ptr [ebp+var_4], 2
lea ecx, [ebp+var_20C]
call sub_46C293
mov byte ptr [ebp+var_4], 1
lea ecx, [ebp+var_1F8]

```

una peculiaridad de este software es que las funciones estan en ingles y las instrucciones de input de las subrutinas están en español y acepta múltiples denominaciones ISO estas denominaciones después de recorrer la subrutina se las trae de los estados WFS y junto con la cantidad de billetes del input esperado recorre los REGEDIT para poder traer el name del dispensador en la subrutina de posibles dispensadores

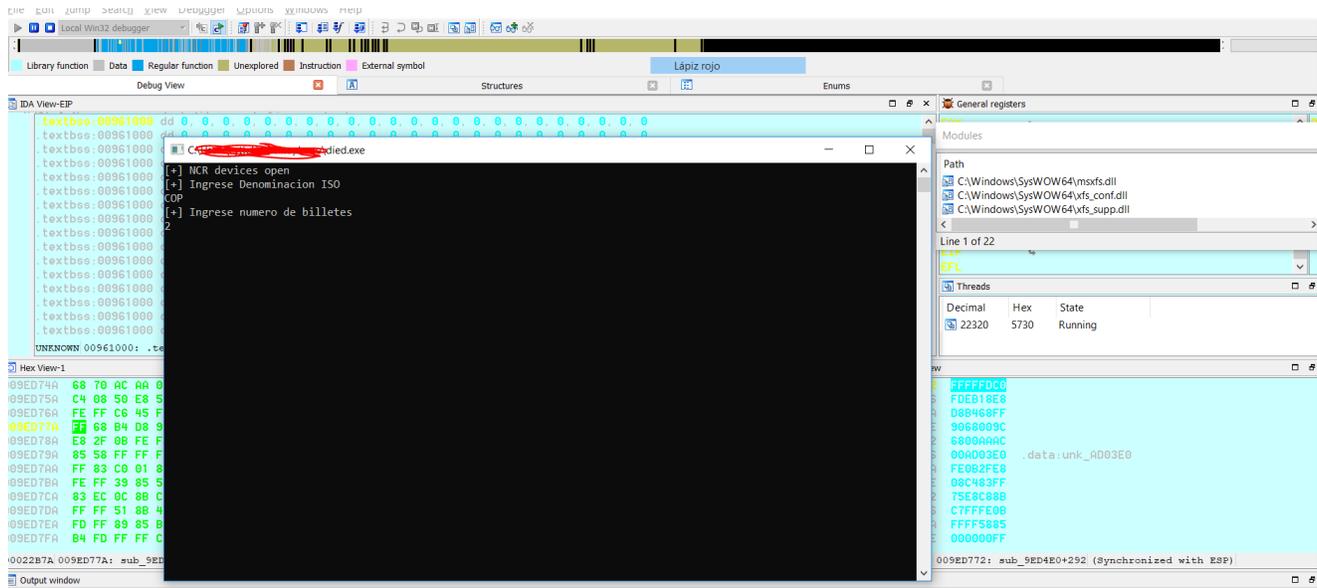
```

.rdata:0054ABE2 db 0
.rdata:0054ABE3 db 0
.rdata:0054ABE4 aIngresoDenomin db '[+] Ingreso Denominacion ISO',0
.rdata:0054AC01 ; DATA XREF: sub_48D4E0+72f0
.rdata:0054AC08 aIngresoNumeroD db '[+] Ingreso numero de billetes',0
.rdata:0054AC08 ; DATA XREF: sub_48D4E0+ADf0
.rdata:0054AC27 align 10h
.rdata:0054AC30 aNumero db '[+] Numero ',0 ; DATA XREF: sub_48D4E0+F4f0
.rdata:0054AC3C align 10h
.rdata:0054AC40 aFail__Dispensa db '[!] FAIL.. dispensadores no encontrados',0
.rdata:0054AC40 ; DATA XREF: sub_48D4E0+209f0
.rdata:0054AC68 align 10h
.rdata:0054AC70 aDispensadoresC db '[+] Dispensadores count ',0 ; DATA XREF: sub_48D4E0+26Af0
.rdata:0054AC89 align 10h
.rdata:0054AC90 aPosiblesDispen db '[+] Posibles dispensadores.. ',0
.rdata:0054AC90 ; DATA XREF: sub_48D4E0+2A0f0
.rdata:0054ACAE db 0
.rdata:0054ACAF db 0
.rdata:0054ACB0 db 0
.rdata:0054ACB1 db 0
.rdata:0054ACB2 db 0
.rdata:0054ACB3 db 0
.rdata:0054ACB4 asc_54ACB4 db ')',0 ; DATA XREF: sub_48D4E0+36Ff0
.rdata:0054ACB4 ; sub_51D960:loc_51D9FBf0
.rdata:0054ACB7 align 4
.rdata:0054ACB8 aDev db 'Dev(',0 ; DATA XREF: sub_48D4E0+3A7f0
.rdata:0054ACBD align 10h
.rdata:0054ACC0 aUnableContinue db '[!] Unable continue, IMPOSIBLE abrir dispenser',0
.rdata:0054ACC0 ; DATA XREF: sub_48D4E0+426f0
.rdata:0054ACEF db 0
.rdata:0054ACF0 db 0

```

Cualquier software que esté en el ATM que tenga funciones WFS y no tenga un firmado de la marca de ATM por lo tanto ya hay que sospechar cosas raras

Aquí el "softwarecito" corriendo del cual tiene una afectación principal para Diebold Agilis



y así es como lo ves en VT, ninguno de los típicos como symantec o mcafee lo identifica

DETECTION	DETAILS	BEHAVIOR	COMMUNITY
SecureAge APEX		Malicious	Undetected
Ad-Aware		Undetected	Undetected
AhnLab-V3		Undetected	Undetected
ALYac		Undetected	Undetected
Arcabit		Undetected	Undetected
Avast-Mobile		Undetected	Undetected

Puntos importantes:

- 1.-Cualquier software que esté en el ATM que tenga funciones WFS y no tenga un firmado de la marca de ATM por lo tanto ya hay que sospechar cosas raras
- 2.- Algunos strings en español por lo tanto de fabricación latino americana

- 3.- Utilización del CEN XFS estándar requiere de la importación de la librería MSXFS.dll por lo tanto es un software para operar ATMS
  - 4.- Soporta multi divisa pero requiere de input por lo tanto el atacante debe tener un teclado cerca habilitado por la ACL y seleccionar o ejecutar un Software que mediante un string input al proceso de consola del software pueda insertar los 2 input que requiere
  - 5.- Tiene una función muy parecida al peralta donde rebusca los dispensadores soportados por el XFS y trae el utilizado actualmente para poder pasarle por el canal del SPI las instrucciones
  - 6.- Al parecer esto tiene pinta de ser una fase de pruebas, puesto que no se encontro ningun packer ni cifrado y requiere de inputs que muy probablemente en ataques masivos se puedan automatizar y vender licencias por cantidad de billetes como ya se ha visto en malware pasados
  - 7.- con esto van 123 aproximadamente de malware y subfamilias reconocidas para atacar a ATMs
- 

#### Recomendaciones :

- 1.- Prohibir la ejecución de cualquier software no solo centrarse en el hash de este o de cualquier otro software /malware generalmente la ejecución por listas blancas n con APPLOCKER o algún software de ACL
  - 2.- Si por desgracia te lo esconden en algún update por la red de ATM, todos los últimos software de esta clase siempre buscan el mayor soporte de dispensadores por lo tanto borrar los regedit de forma remota de la denominación del dispensador y la ruta de la dll ayuda a que provoque errores el software al no encontrar el regedit y al menos pues se protege temporalmente y no se pierde dinero (ojo esta recomendación solo aplicarla en caso extremo ya que si no sabes operar bien lo regedit puedes terminar en que tengas que reinstalar el software del ATM)
  - 3.- Con ATX podemos remotamente bloquear el I/O controller del Dispensar para que cuando busque el Hardware device CDM no esté operativo y el ataque no pueda proceder por mucho que se intente.
  - 4.- Tener buenas políticas de seguridad para prevenir que te suban estos ejecutables al atm sea por un interno o por un externo al ATM
- 

Name: died.exe

MD 56a7732feaa62e8b6ae60f9203c742162

SHA-1 : 94dfa4d597090b34adf18576235df60e3da69b00

SHA-256 d6dff67a6b4423b5721908bdcc668951f33b3c214e318051c96e8c158e8931c0

Authentihash 4cdb89b93c770995763092ee6b5ad4c1a47ba9c3a5b6ef290fb7d11a4cebde29

File type Win32 EXE MagicPE 32 executable for MS Windows (console) Intel 80386 32-bit

File size 1.06 MB (1107968 bytes)

## **Subscribe to Cyttek Group**

---

Get the latest posts delivered right to your inbox

**Great!** Check your inbox and click the link to confirm your subscription.

Please enter a valid email address!