

Tyupkin'in Anatomisi

written by Mert SARICA | 2 February 2015

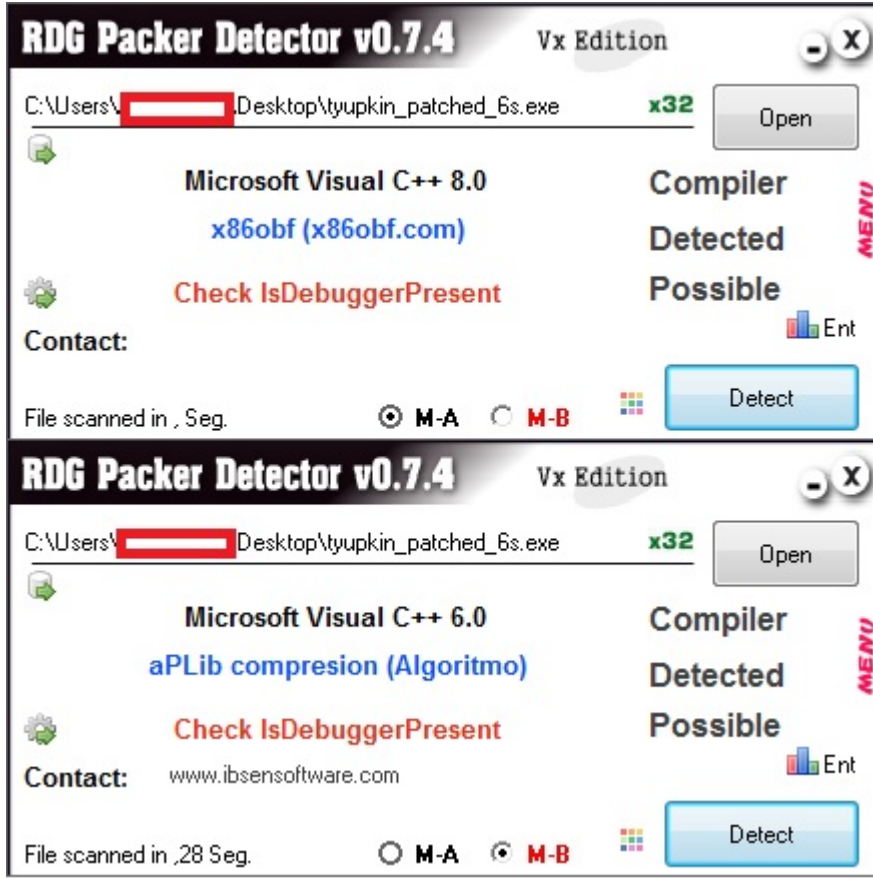
2 Ekim 2014 tarihinde Malezya'da, 18 tane NCR marka ATM'den bir zararlı yazılım yardımı ile yaklaşık 1 milyon dolar çalındığı haberlere yansdı. İşin ilginç yanı ise bankalar bu konuda 4 ay önce uyarılmış olmalarına rağmen gerekli önlemleri almamıştı. Ardından 7 Ekim 2014 tarihinde Kaspersky firması, Tyupkin adını verdikleri bu zararlı yazılım ile art niyetli kişilerin ATMlerden yüklü miktarda para çaldığını teknik detayları ile birlikte duyurdu. Bu duyuruda zararlı yazılımın Batı Avrupa'da yaklaşık 50 tane ATM'de tespit edildiği bilgisi de yer alıyordu. VirusTotal'a bu zararlı yazılımın hangi ülkelerden yüklendiği bilgisine bakıldığında ise Rusya başta olmak üzere, ABD, Hindistan, Çin, İsrail, Fransa ve Malezya'da da bu zararlı yazılımın tespit edildiği görölüyordu.

Merak edenleriniz için Tyupkin ATM zararlı yazılımının ATMlere nasıl yüklendiğine ve paranın nasıl çalındığını kısaca açıklamak gerekirse;

- Art niyetli kişi, ATM'ye fiziksel olarak eriştikten sonra önyüklenabilir (bootable) cd veya usb'yi ATM'ye takıyor, ATM'yi yeniden başlatıyor ve oradan uzaklaşıyor.
- İşletim sistemi yeniden başladıktan sonra Tyupkin zararlı yazılımı işletim sisteminde çalışmaya ve komut beklemeye başlıyor.
- Sadece Pazar ve Pazartesi günleri komut kabul eden bu zararlı yazılıma erişmek için gelen kurye, tuş takımına (pin pad) zararlı yazılımın beklediği rakamları (misal 22222) giriyor.
- Tuşlanan bu rakamlar sonrasında ekranda bir oturum kodu (session code) beliriyor. Bu kodu cep telefonu ile operatöre ileten kurye, doğru oturum anahtarını (session key) tuşladıktan sonra zararlı yazılımın özel menüsüne erişiyor.
- Bu menüde ATM'nin hangi bölmesinde ne kadar para olduğunu öğrenen kurye, parayı çektikten sonra kayıplara karışıyor. (Video)

Ocak 2015 itibariyle Tyupkin zararlı yazılımına Türkiye'de de rastlandığı söylentileri kulaktan kulağa yayılmaya başladı. Kimi vakada, art niyetli kişilerin Tyupkin zararlı yazılımını ATM'nin kasasını anahtarla açarak, ki vakada ise kart okuyucunun altını matkapla delerek bulaştırdıkları söyleniyordu. Bu söylentilerin tamamında ise önyüklenabilir CD yerine önyüklenabilir USB kullanıldığı söyleniyordu. Söylentilerde gerçeklik payı

olup olmadığını araştırmaya başladıktan kısa bir süre sonra, Türkiye’de tespit edilen Tyupkin zararlı yazılımına (Aralık 2014 tarihinde derlenmiş sürüm) VirusTotal üzerinden ulaşmayı başardım.



Hali hazırda Kaspersky tarafından detaylı bir şekilde analiz edilmiş bu zararlı yazılımı tekrar analiz etmek yerine, sisteme bulaştığının nasıl anlaşılacağına dair kilit noktalara hızlıca göz atmaya karar verdim. (Analizi ATM üzerinde gerçekleştirmediğim için analiz esnasında Pin Pad ile bağlantı kurulamamış ve bu nedenle Tyupkin’in tüm izleri sildiği, işlemleri geri aldığı senaryo üzerinden ilerlenmiştir.)

İlk olarak Tyupkin’in dinamik analizi atlatma adına çalıştıktan sonra 10 dakika uykuya geçmesine müdahale ederek (patch) bunu kısalttım.

```
call    ds:CreateWindowExW
cmp     eax, ebx
mov     dword_42EA34, eax
jnz     short loc_42024C
```

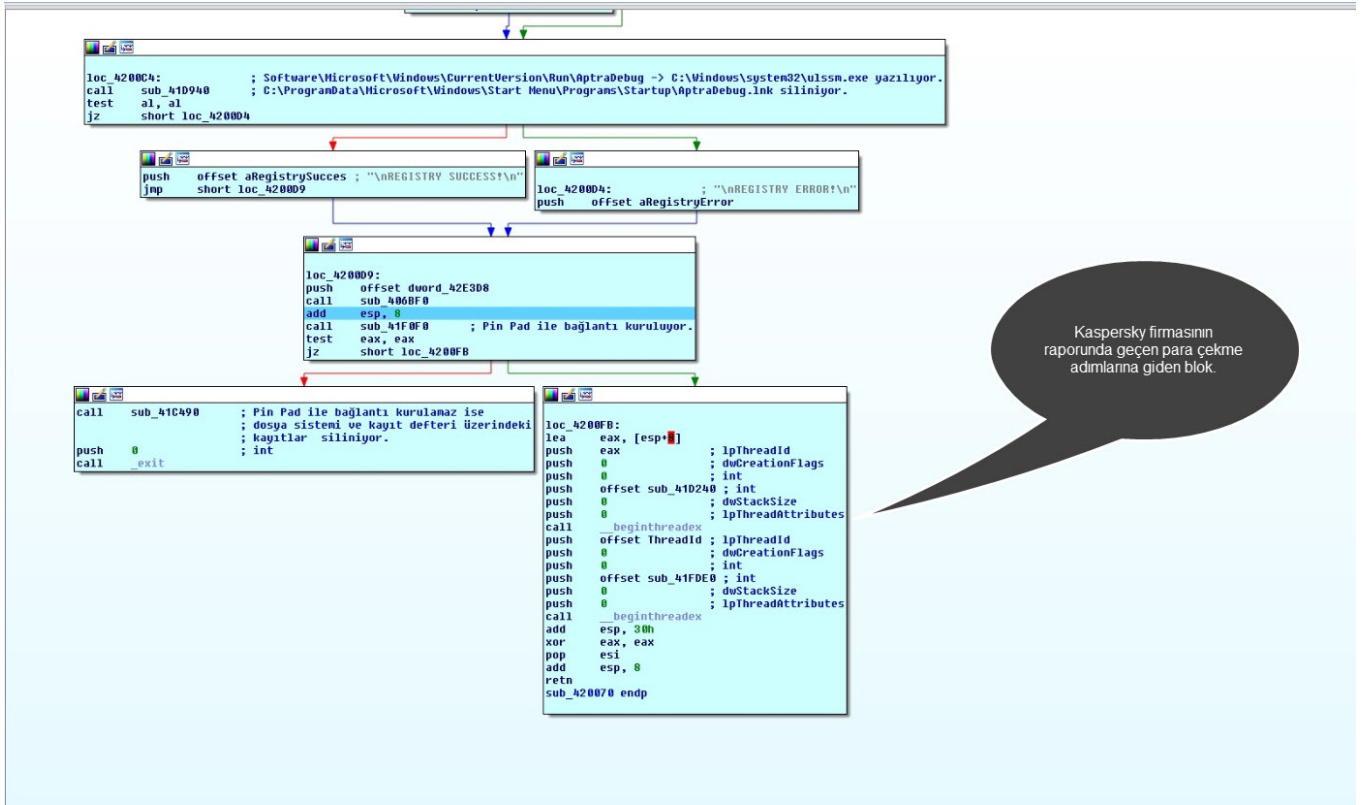
```
30h      ; uType
offset aError_0 ; "Error!"
offset aWindowCreation ; "Window Creation Failed!"
ebx      ; hWnd
ds:MessageBoxW
eax, eax
edi
esi
ebx
esp, ebp
ebp
10h
```

```
loc_42024C:      ; nCmdShow
push     ebx
push     eax      ; hWnd
call     ds:ShowWindow
mov     ecx, dword_42EA34
push     ecx      ; hWnd
call     ds:UpdateWindow
mov     edx, dword_42EA34
push     edx      ; hWnd
call     ds:GetDC
mov     hdc, eax
call     sub_41C110
push     600000    ; 10 dakika uyku vakti ;)
call     ds:Sleep
call     sub_41CA10
call     sub_420070
mov     esi, ds:GetMessageW
push     ebx      ; wParamFilterMax
push     ebx      ; wParamFilterMin
push     ebx      ; hWnd
lea     eax, [esp+74h+Msg]
push     eax      ; lParam
call     esi ; GetMessageW
test     eax, eax
jle     short loc_4202C7
```

```
mov     edi, ds:TranslateMessage
```

100.00% (218,1157) (878,239) 00020273 00420273: WinMain(x,x,x,x)+133

Analize başladıktan kısa bir süre sonra Tyupkin'in çalışmak ya da çalışmamak işte bütün mesela bu dediği ana kontrol adımına geldim. Burada ATM'nin Pin Pad'i ile bağlantı kurmaya çalışan Tyupkin, bağlantı kuramadığı taktirde hem önyüklenabilir aygıt (cd, usb) üzerinden hem de kendi üzerinden yaptığı değişiklikleri geri almaya başlıyordu. Bu adımlar sayesinde zararlı yazılımın hangi güvenlik yazılımlarını devre dışı bıraktığı da anlaşılabiliyordu. (Kaspersky'nin raporuna göre Tyupkin, McAfee'nin Application Control (SolidCore) yazılımını devre dışı bırakıyordu.)



Zararlı yazılımın çalıştıktan ancak Pin Pad'e bağlanamadıktan sonra hangi adımlardan geçtiğini kısaca açıklamam gerekirse;

Sistem başladıktan sonra çalışabilmesi adına kayıt defterindeki (registry) Software\Microsoft\Windows\CurrentVersion\Run\AptraDebug anahtarına C:\Windows\system32\ulssm.exe değerini yazıyor. (sub_41D940)

Önyüklenabilir aygıt üzerinden oluşturulduğunu tahmin ettiğim C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Startup\AptraDebug.lnk dosyasını (muhtemelen bu da system32\ulssm.exe dosyasını işaret ediyor.) siliyor. (sub_41D750)

Tyupkin, Pin Pad ile bağlantı kuramaz ise daha önce oluşturduğu Software\Microsoft\Windows\CurrentVersion\Run\AptraDebug anahtarını siliyor. Ardından yine önyüklenabilir aygıt üzerinden devre dışı bıraktığı McAfee Application Control'un (SolidCore) servislerinin, sistem yeniden başladığında tekrar çalışabilmesi adına aşağıdaki değişiklikleri gerçekleştiriyor. (sub_41C490)

HKLM\System\CurrentControlSet\services\scsrrvc\Start değerini 2 olarak değiştiriyor.

HKLM\System\CurrentControlSet002\services\scsrrvc\Start değerini 2 olarak değiştiriyor.

HKLM\System\CurrentControlSet003\services\scsrrvc\Start değerini 2 olarak değiştiriyor.

```

lea     edx, [esp+0078h+phkResult]
push    edx                ; phkResult
push    0F003Fh            ; samDesired
push    0                  ; ulOptions
push    offset SubKey      ; "Software\\Microsoft\\Windows\\CurrentVe"...
push    80000002h          ; hKey
mov     dword ptr [edi], const WCHAR SubKey
call    edi ; RegSubKey: ; DATA XREF: sub_41C490+F6f0
test    eax, eax           ; unicode 0, <Software\\Microsoft\\Windows\\CurrentVersion\\Run>,0
jnz     short loc_41C490

```

```

mov     eax, [esp+0078h+phkResult]
push    offset ValueName ; "AptaraDebug"
push    eax                ; hKey
call    ds:RegDeleteValueW

```

```

var_000=byte ptr -000h
var_4= dword ptr -4
push    ebp
mov     ebp, esp
and     esp, 0FFFFFFFh
sub     esp, 006Ch
mov     eax, ___security_cookie
xor     eax, esp
mov     [esp+006Ch+var_4], eax
mov     ecx, ds:dword_42805C
mov     eax, ds:dword_428058
mov     edx, ds:dword_428060
push    esi
push    edi
mov     [esp+0078h+var_04C], ecx
mov     ecx, 13h
mov     esi, offset aSystemControls ; "SYSTEM\\ControlSet001\\Services\\scsrvc"
lea     edi, [esp+0078h+SubKey]
push    18Ch                ; size_t
rep movsd
mov     dword ptr [esp+007Ch+var_7DC], eax
mov     ax, ds:word_428064
lea     ecx, [esp+007Ch+var_7DC]
push    0                  ; int
push    ecx                ; void *
mov     [esp+0084h+var_048], edx
mov     [esp+0084h+var_044], ax
call    _memset
push    18Ch                ; size_t
lea     edx, [esp+0088h+var_3CC]
push    0                  ; int
mov     ecx, 13h
mov     esi, offset aSystemContro_0 ; "SYSTEM\\ControlSet002\\Services\\scsrvc"
lea     edi, [esp+008Ch+var_418]
push    edx                ; void *
rep movsd
call    _memset
push    18Ch                ; size_t
lea     eax, [esp+0094h+var_504]
push    0                  ; int
mov     ecx, 13h
mov     esi, offset aSystemContro_1 ; "SYSTEM\\ControlSet003\\Services\\scsrvc"
lea     edi, [esp+0098h+var_620]
push    eax                ; void *
rep movsd
call    _memset
mov     ecx, 15h
mov     esi, offset aSystemCurrentc ; "SYSTEM\\CurrentControlSet\\Services\\sc..."
lea     edi, [esp+009Ch+var_210]
push    18Ch                ; size_t
rep movsd
lea     ecx, [esp+00A0h+var_18C]
push    0                  ; int
push    ecx                ; void *
call    _memset
mov     edi, ds:RegOpenKeyExW
add     esp, 30h

```

100.00% (-247,466) (452,268) 0001C490 0041C490: sub_41C490

Yine önyüklenebilir aygıt (bootable usb, cd) üzerinden gerçekleştirildiğini tahmin ettiğim bu işlemde, McAfee Application Control (SolidCore), McAfee Host IPS ve McAfee Antivirüs yazılımlarına ait olan aşağıdaki dosyaları C:\windows\system32\config klasöründen alıp, C:\windows\system32\drivers klasörüne kopyalıyor. (En başta config klasörüne kopyalamasının sebebi, u dosyaları bulamayan güvenlik yazılımlarının sistem başlangıcında çalışmasını engellemektir.) (sub_41BBF0)

HipShieldK.sys (McAfee Host IPS sürücüsü),
mfeapfk.sys (McAfee Antivirüs sürücüsü),
mfeavfk.sys (McAfee Antivirüs sürücüsü),
mfebopk.sys (McAfee Antivirüs sürücüsü),
mfehdk.sys (McAfee Antivirüs sürücüsü),
mfeclnk.sys (McAfee Antivirüs sürücüsü),
mferkdet.sys (McAfee Antivirüs sürücüsü),
mfewfpk.sys (McAfee Antivirüs sürücüsü),
mfenlfk.sys (McAfee Host IPS sürücüsü),

mfefirek.sys (Mcafee Host IPS sürücüsü)

```
push    ecx
mov     eax, __security_cookie
xor     eax, esp
mov     [esp+4+var_4], eax
push    offset aHipshieldk_sys ; "\\HipShieldK.sys"
call    sub_41B530
push    offset aMfeapfk_sys ; "\\mfeapfk.sys"
call    sub_41B530
push    offset aMfeavfk_sys ; "\\mfeavfk.sys"
call    sub_41B530
push    offset aMfebopk_sys ; "\\mfebopk.sys"
call    sub_41B530
push    offset aMfeclnk_sys ; "\\mfeclnk.sys"
call    sub_41B530
push    offset aMfehdk_sys ; "\\mfehdk.sys"
call    sub_41B530
push    offset aMferkdet_sys ; "\\mferkdet.sys"
call    sub_41B530
push    offset aMfewfpk_sys ; "\\mfewfpk.sys"
call    sub_41B530
push    offset aMfenlfk_sys ; "\\mfenlfk.sys"
call    sub_41B530
push    offset aMfefirek_sys ; "\\mfefirek.sys"
call    sub_41B530
mov     ecx, [esp+2Ch+var_4]
add     esp, 28h
xor     ecx, esp
xor     eax, eax
call    @_security_check_cookie@4 ; __security_check_cookie(x)
pop     ecx
```

0041BBF0: sub_41BBF0

NCR firmasının Aptra uygulaması ile birlikte dağıttı özelleştirilmiş Solidcore yazılımına ait kayıtları (solidcore.log ve s3diag.log) C:\program files\ncr aptra\Solidcore for APTRA\Logos klasöründen siliyor. (sub_41B450)

```
call    _memset
mov     esi, ds:SetFileAttributesW
add     esp, 0Ch
push    80h ; dwFileAttributes
push    offset aCProgramFilesN ; "C:\\program files\\ncr aptra\\Solidcore"...
call    esi ; SetFileAttributesW
push    104h ; const WCHAR aCProgramFilesN
lea     ecx, [esp+110h+aCProgramFilesN ; DATA XREF: sub_41B450+39fo
push    offset aCProgramFilesN ; "C:\\program files\\ncr aptra\\Solidcore for APTRA\\Logos\\solid"
push    ecx ; unicode 0, <core.log>,0
call    _wcstombs
lea     edx, [esp+118h+var_108]
push    edx ; char *
call    _remove
add     esp, 10h
push    80h ; dwFileAttributes
push    offset aCProgramFile_1 ; "C:\\program files\\ncr aptra\\Solidcore"...
call    esi ; SetFileAttributesW
push    104h ; size_t
mov     esi, eax
lea     eax, [esp+110h+var_108]
push    offset aCProgramFile_2 ; "C:\\program files\\ncr aptra\\Solidcore"...
push    eax ; char *
call    _wcstombs
lea     ecx, [esp+118h+var_108]
push    ecx ; char *
call    _remove
mov     ecx, [esp+11Ch+var_4]
add     esp, 10h
mov     eax, esi
pop     esi
xor     ecx, esp
```

0001B450 0041B450: sub_41B450

```

lea     eax, [esp+110h+var_107]
push    0 ; int
push    eax ; void *
mov     [esp+118h+var_108], 0
call    _memset
mov     esi, ds:SetFileAttributesW
add     esp, 0Ch
push    80h ; dwFileAttributes
push    offset aCProgramFilesN ; "C:\\program files\\ncr aptra\\Solidcore"...
call    esi ; SetFileAttributesW
push    104h ; size_t
lea     ecx, [esp+110h+var_108]
push    offset aCProgramFile_0 ; "C:\\program files\\ncr aptra\\Solidcore"...
push    ecx ; char *
call    _wcstombs
lea     edx, [esp+118h+var_108]
push    edx ; char *
call    _remove
add     esp, 10h
push    80h ; dwFileAttributes
push    offset aCProgramFile_1 ; "C:\\program files\\ncr aptra\\Solidcore"...
call    esi ; SetFileAttributesW
push    104h ; const WCHAR aCProgramFile_1
mov     esi, eax ; aCProgramFile_1 ; DATA XREF: sub_41B450+66fo
lea     eax, [esp+110h] ; unicode 0, <C:\\program files\\ncr aptra\\Solidcore for APTRA\\Logs\\s3dia>
push    offset aCProgramFile_1 ; <g.log>,0
push    eax ; char *
call    _wcstombs
lea     ecx, [esp+118h+var_108]
push    ecx ; char *
call    _remove
mov     ecx, [esp+11Ch+var_4]

```

,300) 0001B489 0041B489: sub_41B450+39

Yine önyüklenebilir aygıt üzerinden kopyalandığını tahmin ettiğim Windows\System32\kbd110.dll dosyasını siliyor. (sub_41C490)

Son olarak ise "C:\Windows\System32\cmd.exe" /C ping 127.0.0.1 -n 8 & del /F /S /Q C:\Windows\system32\ulssm.exe komutunu çalıştırarak ulssm.exe dosyasını siliyor. (sub_41C490)

Analizi tamamlamadan önce bu iz silme ve yapılan işlemleri geri alma fonksiyonunun (sub_41C490) başka hangi fonksiyonlardan çağrıldığına (xfref) baktığımda, karşıma çıkan iki fonksiyondan biri dikkatimi çekti. Bu fonksiyonda öncelikle yerel ağ bağlantısı durduruluyor ardından para çekme için izin veriliyor ve ardından 48 dakika sonra izleri silme, işlemleri geri alma (sub_41C490) fonksiyonu çağrılıyor.

```

; Attributes: bp-based frame

sub_41C490 proc near

Data      = byte ptr -0D6Ah
hKey      = dword ptr -0D64h
var_D60   = dword ptr -0D60h
var_D5C   = dword ptr -0D5Ch
phkResult = dword ptr -0D58h
var_D54   = dword ptr -0D54h
ValueName = word ptr -0D50h
var_D4C   = dword ptr -0D4Ch
var_D48   = dword ptr -0D48h
var_D44   = word ptr -0D44h
var_D40   = byte ptr -0D40h
var_D3F   = byte ptr -0D3Fh
Parameters= word ptr -0C38h
var_BE8   = dword ptr -0BE8h
FileName  = ...
var_A2E   = ...
SubKey    = ...
var_7DC   = ...
ControlSet003 = ...
var_5D4   = ...
ControlSet002 = ...
var_3CC   = ...
ControlSet = ...
var_1BC   = ...
var_4     = ...

cmd = esi

push esp
and ebp, esp
sub esp, 0FFFFFFFh
sub esp, 0D6Ch
mov eax, __security_cookie
xor eax, esp
[esp+0D6Ch+var_4], eax
mov ecx, ds:dword_42805C
mov eax, ds:dword_428058
mov edx, ds:dword_428060
push ebx
push cmd
push edi
mov [esp+0D78h+var_D4C], ecx
mov ecx, 13h
cmd, offset aSystemServiceControls ; "SYSTEM\ControlSet001\Services\scsrvc"
lea edi, [esp+0D78h+SubKey]
push 18Ch ; size_t
rep movsd
mov dword ptr [esp+0D7Ch+ValueName], eax
mov ax, ds:dword_428064
lea ecx, [esp+0D7Ch+var_7DC]

```

```

loc_41EC22:
mov eax, dword_42E568
esi, ds:SetWindowTextW
push offset aDisablingLocal ; "DISABLING LOCAL AREA NETWORK...\\PLEASE..."
mov ebx, 77777770h
push eax ; hWnd
mov edi, eax
mov color, ebx
call esi ; SetWindowTextW
push edi ; hWnd
mov edi, ds:UpdateWindow
call edi ; UpdateWindow
mov eax, dword_42E56C
push offset word_428490 ; lpString
push eax ; hWnd
mov ebp, eax
mov color, 0
call esi ; SetWindowTextW
push ebp ; hWnd
call edi ; UpdateWindow
push 0 ; pReserved
mov byte_42EA3B, 1
call ds:Coinitialize
push offset aLocalAreaCon_0 ; "Local Area Connection"
call sub_4108A0
add esp, 8
call ds:Coinitialize
push offset aDispensePermis ; "\nDISPENSE PERMISSION GRANTED\n"
push offset dword_42E3D8
call sub_406BF0
mov eax, dword_42E568
add esp, 8
push offset off_4284B4 ; lpString
push eax ; hWnd
mov ebp, eax
mov color, ebx
call esi ; SetWindowTextW
push ebp ; hWnd
call edi ; UpdateWindow
mov eax, dword_42E56C
push offset aToStartDispe_0 ; "TO START DISPENSE OPERATION - \nENTER C"...
push eax ; hWnd
mov ebx, eax
mov color, 0FFFFh
call esi ; SetWindowTextW
push ebp ; hWnd
call edi ; UpdateWindow
lea ecx, [esp+38h+ThreadId]
push ecx ; lpThreadId
push 0 ; dwCreationFlags
push 0 ; int
push offset loc_41C870 ; 48 dakika uyu ve temizlene fonksiyonuna git
push 0 ; dwStackSize
push 0 ; lpThreadAttributes
call beginthreadx
add esp, 10h

```



```

.text:0041C861 ;
.text:0041C862 align 10h
.text:0041C870 loc_41C870: ; DATA XREF: sub_41E9E0+39A10
.text:0041C870 push esi
.text:0041C871 mov esi, ds:Sleep
.text:0041C877 push edi
.text:0041C878 mov edi, 480
.text:0041C87D lea ecx, [ecx*0]
.text:0041C880 loc_41C880:
.text:0041C880 push 60000 ; CODE XREF: .text:0041C88A1j
.text:0041C881 call esi ; Sleep ; (480 x 6) / 60 = 48 dakika uyu
.text:0041C887 sub edi, 1 ; 480'den geriye say
.text:0041C88A jnz short loc_41C880 ; (480 x 6) / 60 = 48 dakika uyu
.text:0041C88C call sub_41C490 ; Dosya sistemi ve kayıt defteri üzerindeki
; kayıtlar siliniyor.
.text:0041C891 push 0
.text:0041C893 call _exit
.text:0041C893 dd 0CCCC5E5Fh, 0CCCCCCCCCh
.text:0041C8A0 ; ===== SUBROUTINE =====
.text:0041C8A0
.text:0041C8A0 int __cdecl sub_41C8A0(int, int, void *, int)
.text:0041C8A0 sub_41C8A0 proc near ; CODE XREF: sub_41FDE0+1CC1p
.text:0041C8A0
.text:0041C8A0 var_20 = dword ptr -20h
.text:0041C8A0 var_1C = dword ptr -1Ch
.text:0041C8A0 var_18 = byte ptr -18h
.text:0041C8A0 var_C = dword ptr -0Ch
.text:0041C8A0 var_4 = dword ptr -4
.text:0041C8A0 arg_0 = dword ptr 4
.text:0041C8A0 arg_8 = dword ptr 0Ch
.text:0041C8A0 arg_C = dword ptr 10h
.text:0041C8A0
.text:0041C8A0 push 37777777770
.text:0041C8A2 push offset sub_423AA1
.text:0041C8A7 mov eax, large fs:0
.text:0041C8A9 push eax
.text:0041C8AE sub esp, 10h
.text:0041C8B1 push ebx
.text:0041C8B2 push ebp
.text:0041C8B3 push esi
.text:0041C8B4 push edi
.text:0041C8B5 mov eax, __security_cookie
.text:0041C8B8 xor eax, esp
.text:0041C8BC push eax
.text:0041C8BD lea eax, [esp+30h+var_C]
.text:0041C8C1 mov large fs:0, eax
.text:0041C8C7 mov edi, [esp+30h+arg_0]
.text:0041C8CB xor esi, esi
.text:0041C8CD mov [esp+30h+var_20], esi
.text:0041C8D1 mov eax, 1
.text:0041C8D6 mov [esp+30h+var_4], eax
.text:0041C8DA mov [edi+4], esi
.text:0041C8DD mov [edi+8], esi

```

Gerçekleştirdiğim bu kısa analiz sonucunda, Türkiye’de görülen Tyupkin ile Kaspersky’nin raporunda yer alan sürüm arasında bazı farklar olduğu görülüyor.

Birincisi, Türkiye sürümünde sadece McAfee Application Control (Solidcore) değil bunun dışında sürücü dosyalarına bakılacak olursa McAfee Antivirüs ve McAfee Host IPS de devre dışı bırakılıyor gibi görünüyor.

İkincisi ise yine Kaspersky’nin raporunda para çekme işlemi gerçekleştirildikten sonra Tyupkin’in kendini sildiğine yer verilmemiş ancak mevcut sürümde böyle bir işlev bulunuyor. (Bu işlev nedeniyle, kurye tarafından ATM’den para çalınmadan önce Tyupkin bulaşmış bir ATM’yi tespit etmek isteyenlerin yapması gereken işlerden ikisi, sistem üzerinde ulssm.exe adlı bir yazılımın çalışıp çalışmadığını kontrol etmek ve bu dosyanın Windows\system32 klasörü altında olup olmadığını kontrol etmek yerinde olacaktır.)

Bu zararlı yazılıma karşı çözüm olarak ATM üzerinde disk şifreleme, fiziksel erişimde ve bağlantı noktalarında iş akışını dahi etkileyebilecek düzeyde radikal kısıtlamalar düşünülebilir.

Kısıtlı zaman ve teknik imkanlar dahilinde (eksikler veya hatalar olabilir) gerçekleştirdiğim bu analizin, Tyupkin’e karşı verilen mücadeleye katkısı olması dileğiyle bir sonraki yazıda görüşmek üzere herkese güvenli günler dilerim.