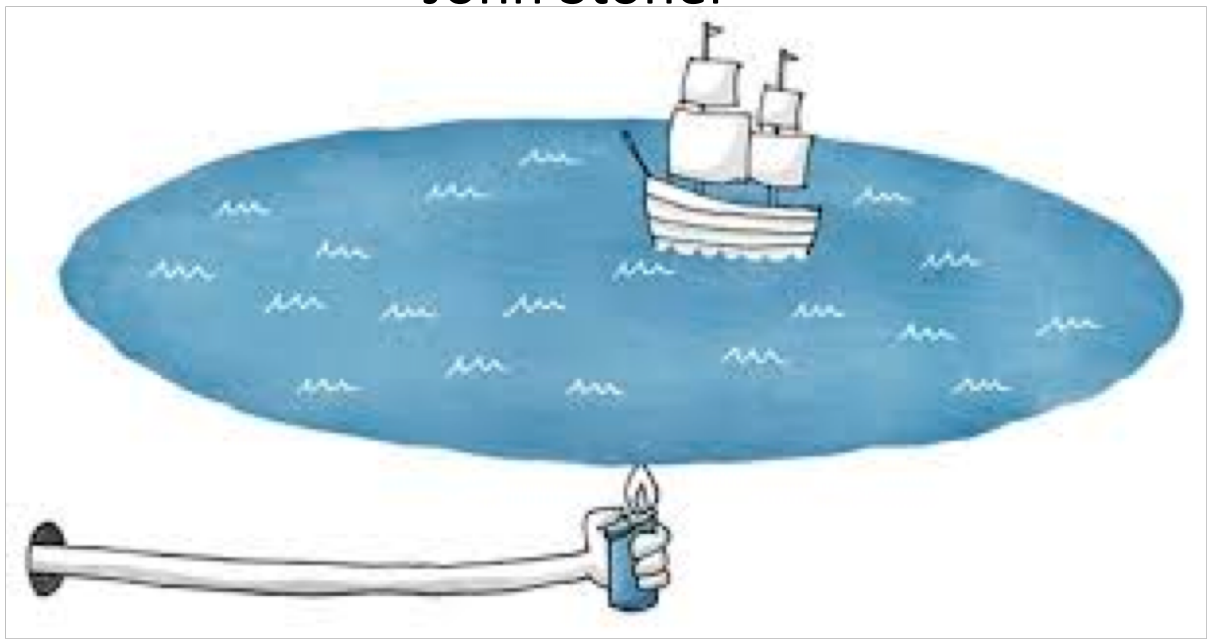
Don't Boil the Ocean: Using MITRE ATT&CK to Guide Threat Hunting Activities

2019 FIRST Technical Colloquium – April 3 2019

John Stoner



whoami > John Stoner

Principal Security Strategist

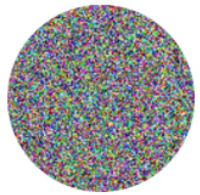
@stonerpsu

- 4 years @ Splunk
- Creator of SA-Investigator for Splunk
- Blogger on Hunting and SecOps
- Symantec→ArcSight→Splunk - I’ve Seen them all
- Loves The Smiths and all 80’s sadtimey music

Agenda

- Why Do We Hunt
- MITRE ATT&CK and Evolution
- Methods to Conduct Hunts
- What Have We Learned
- Operationalizing Our Hunts

Tweet

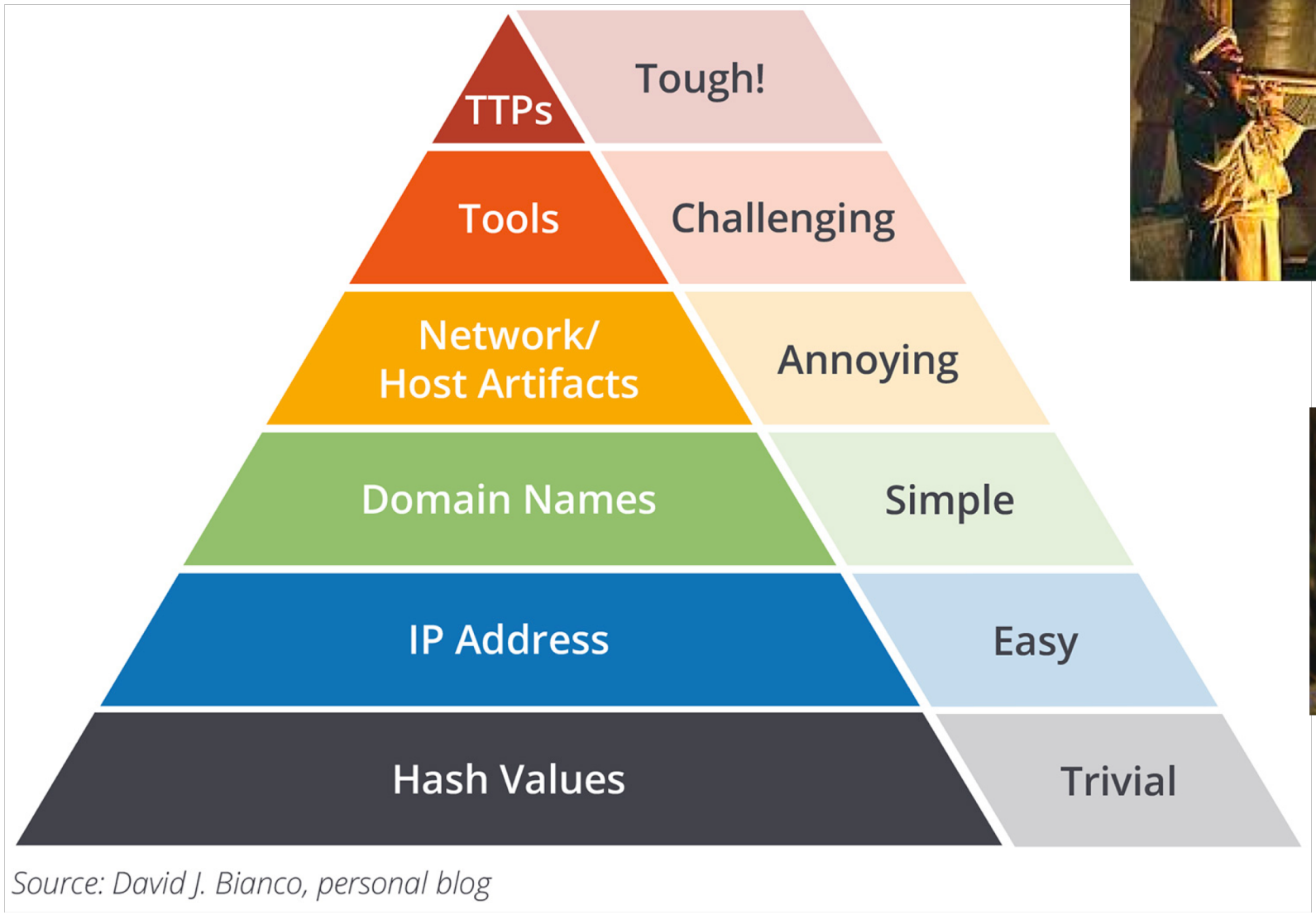


Matt Graeber
@mattifestation

Incident responder: "The machine was infected with crimeware. We just had IT rebuild the system. End of story."
Nation-state attacker: "We got our foothold and only lost a single host in the process."

2/18/18, 10:36 AM

What To Hunt For?

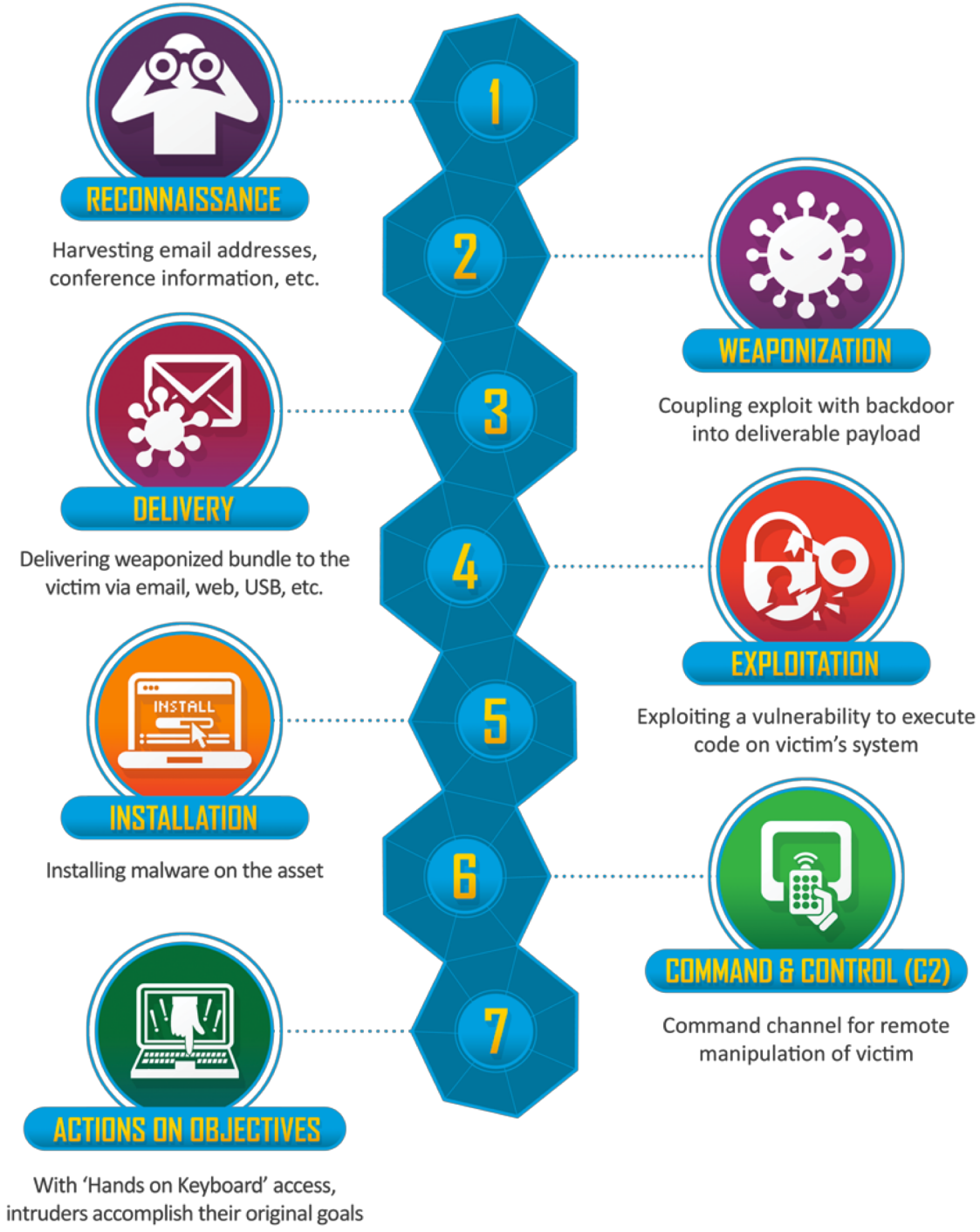


Source: David J. Bianco, personal blog


YOU MUST CHOOSE
...BUT CHOOSE WISELY

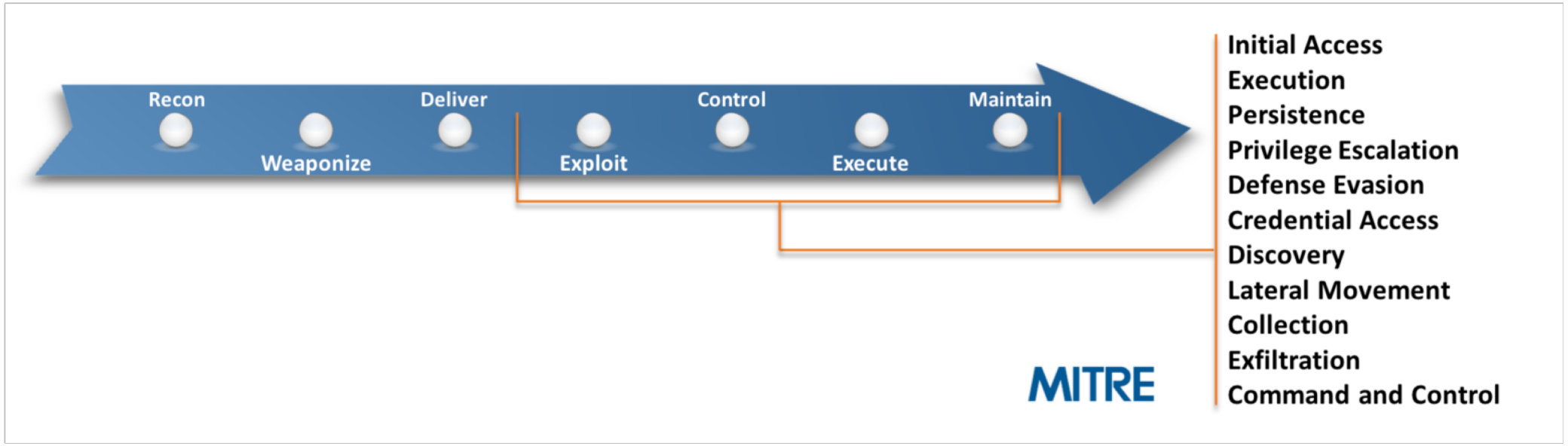
Lockheed Martin Kill Chain

- Sadly Over-Commercialized
- Still Great Conceptually
- Purpose Driven



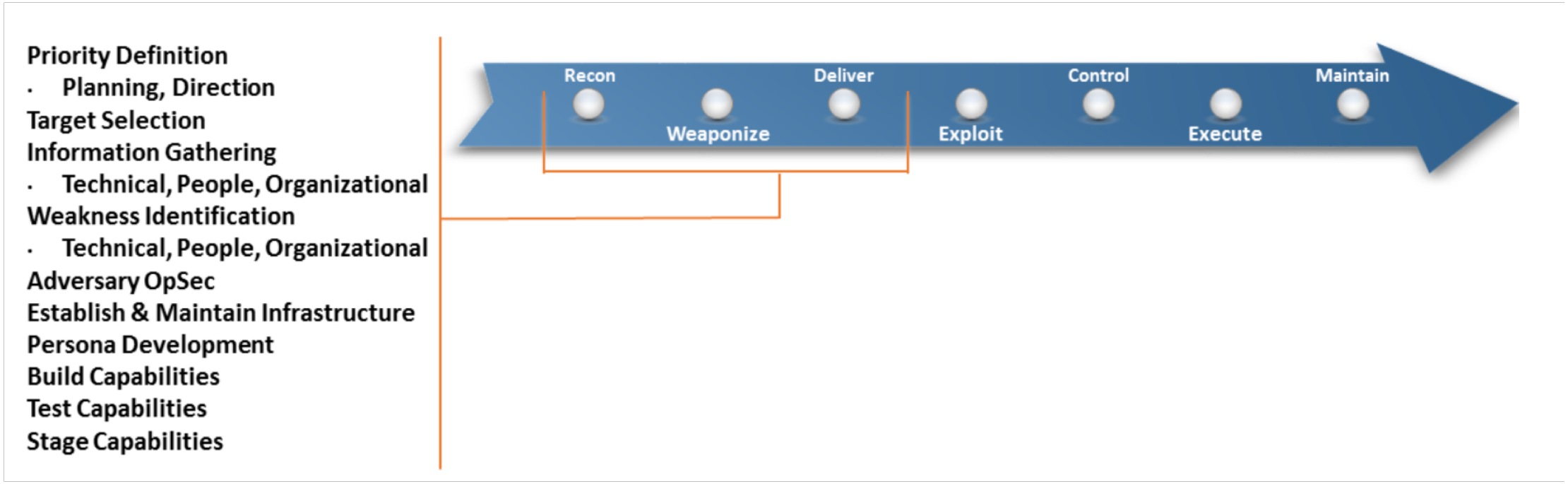
MITRE ATT&CK

- Adversarial Tactics, Techniques, and Common Knowledge
- Builds on Lockheed Martin's Kill Chain but focuses on tactics and techniques that occur during exploit and activity occurring post exploit

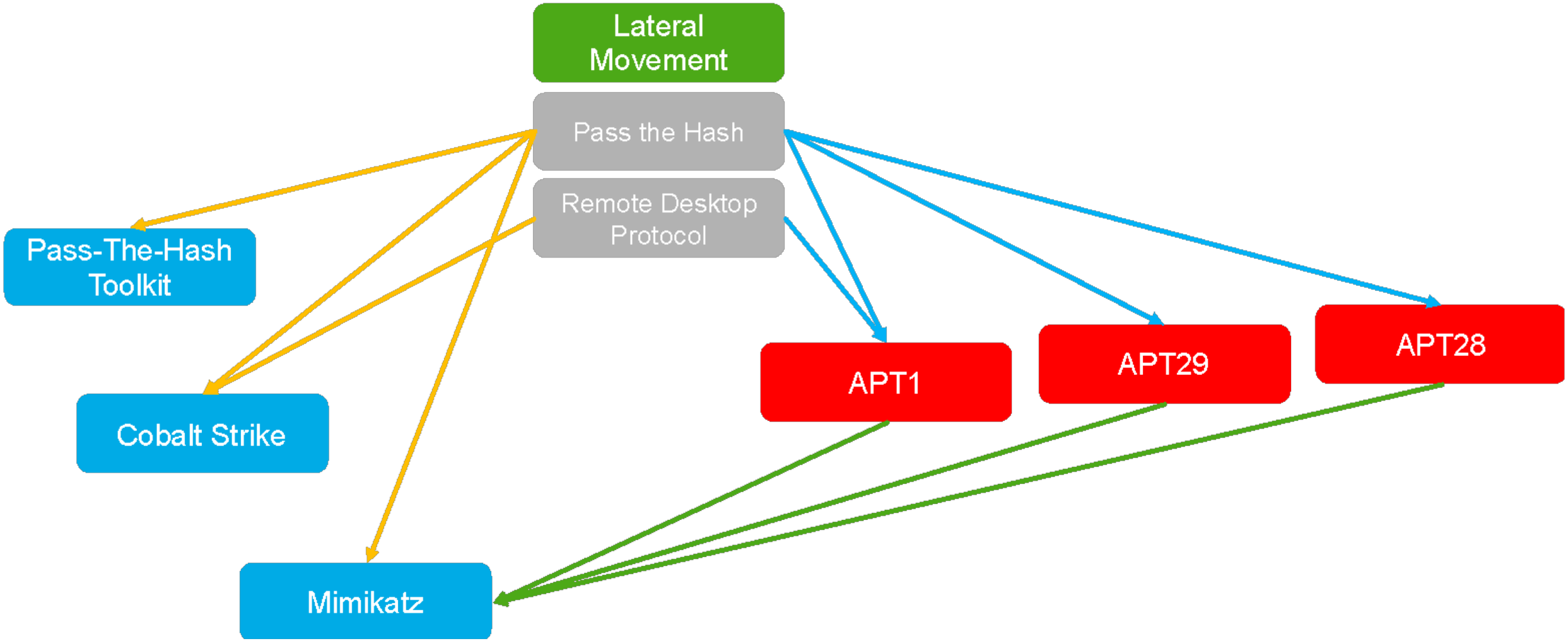


MITRE PRE-ATT&CK

- Builds on Lockheed Martin's Kill Chain but focuses on tactics and techniques that occur PRIOR to exploit



Tactic, Techniques, Adversaries and Software



Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control
Hardware Additions	Scheduled Task			Binary Padding	Credentials in Registry	Browser Bookmark Discovery	Exploitation of Remote Services	Data from Information Repositories	Exfiltration Over Physical Medium	Remote Access Tools
Trusted Relationship	LSASS Driver		Extra Window Memory Injection		Exploitation for Credential Access	Network Share Discovery	Distributed Component Object Model	Video Capture	Exfiltration Over Command and Control Channel	Port Knocking
Supply Chain Compromise	Local Job Scheduling		Access Token Manipulation		Forced Authentication	Peripheral Device Discovery	Remote File Copy	Audio Capture		Multi-hop Proxy
Spearphishing Attachment	Trap		Bypass User Account Control		Hooking	File and Directory Discovery	Pass the Ticket	Automated Collection		Domain Fronting
Exploit Public-Facing Application	Launchctl		Process Injection		Password Filter DLL	Permission Groups Discovery	Replication Through Removable Media	Clipboard Data		Data Encoding
Replication Through Removable Media	Signed Binary Proxy Execution	Image File Execution Options Injection			LLMNR/NBT-NS Poisoning	Process Discovery	Windows Admin Shares	Email Collection		Remote File Copy
Spearphishing via Service	User Execution	Plist Modification			Private Keys	System Network Connections Discovery	Pass the Hash	Screen Capture		Multi-Stage Channels
Spearphishing Link	Exploitation for Client Execution	Valid Accounts			Keychain	System Owner/User Discovery	Third-party Software	Data Staged		Web Service
Drive-by Compromise	CMSTP	DLL Search Order Hijacking		Signed Script Proxy Execution	Input Prompt	System Network Configuration Discovery	Shared Webroot	Input Capture		Standard Non-Application Layer Protocol
Valid Accounts	Dynamic Data Exchange	AppCert DLLs		DCShadow	Bash History	Application Window Discovery	Logon Scripts	Data from Network Shared Drive		Connection Proxy
	Mshta	Hooking		Port Knocking	Two-Factor Authentication Interception	Password Policy Discovery	Windows Remote Management	Data from Local System		Multilayer Encryption
	AppleScript	Startup Items		Indirect Command Execution		System Time Discovery	Application Deployment Software	Man in the Browser		Standard Application Layer Protocol
	Source	Launch Daemon		BITS Jobs	Replication Through Removable Media	Account Discovery	SSH Hijacking	Data from Removable Media		Commonly Used Port
	Space after Filename	Dylib Hijacking		Control Panel Items	Input Capture	System Information Discovery	AppleScript			Standard Cryptographic Protocol
	Execution through Module Load	Application Shimming		CMSTP	Network Sniffing	Security Software Discovery	Taint Shared Content			Custom Cryptographic Protocol
	Regsvcs/Regasm	AppInit DLLs		Process Doppelgänging	Credential Dumping	Network Service Scanning	Remote Desktop Protocol			Data Obfuscation
	InstallUtil	Web Shell		Mshta	Kerberoasting	Remote System Discovery	Remote Services			Custom Command and Control Protocol
	Regsvr32	Service Registry Permissions Weakness		Hidden Files and Directories	Securityd Memory	Query Registry				Communication Through Removable Media
	Execution through API	New Service		Space after Filename	Brute Force	System Service Discovery				Multiband Communication
	PowerShell	File System Permissions Weakness		LC_MAIN Hijacking	Account Manipulation					Fallback Channels
	Rundll32	Path Interception	Sudo Caching	HISTCONTROL	Credentials in Files					Uncommonly Used Port
	Third-party Software	Accessibility Features	SID-History Injection	Hidden Users						
	Scripting	Port Monitors	Sudo	Clear Command History						
	Graphical User Interface	Kernel Modules and Extensions	Setuid and Setgid	Gatekeeper Bypass						
	Command-Line Interface	Port Knocking	Exploitation for Privilege Escalation	Hidden Window						
	Service Execution	SIP and Trust Provider Hijacking		Deobfuscate/Decode Files or Information						
	Windows Remote Management	Screensaver		Trusted Developer Utilities						
	Signed Script Proxy Execution	Browser Extensions		Component Object Model Hijacking						
	Control Panel Items	Re-opened Applications		InstallUtil						
	Trusted Developer Utilities	Rc.common		Regsvr32						
	Windows Management Instrumentation	Login Item		Code Signing						
		LC_LOAD_DYLIB Addition		Modify Registry						
		Hidden Files and Directories		Component Firmware						
		Office Application Startup		Redundant Access						
		External Remote Services		File Deletion						
		Netsh Helper DLL		Web Service						
		Component Object Model Hijacking		Timestomp						
		Redundant Access		NTFS File Attributes						
		Security Support Provider		Process Hollowing						
		Bootkit		Disabling Security Tools						
		Hypervisor		Rundll32						
		Registry Run Keys / Start Folder		DLL Side-Loading						
		Logon Scripts		Indicator Removal on Host						
		Modify Existing Service		Scripting						
		Shortcut Modification		Indicator Blocking						
		System Firmware		Software Packing						
		Winlogon Helper DLL		Masquerading						
		Time Providers		Obfuscated Files or Information						
		BITS Jobs		Signed Binary Proxy Execution						
		Launch Agent		Exploitation for Defense Evasion						
		.bash_profile and .bashrc		SIP and Trust Provider Hijacking						
		Create Account		Launchctl						
		Authentication Package		Install Root Certificate						
		Component Firmware		Network Share Connection Removal						
		Windows Management Instrumentation Event Subscription		Regsvcs/Regasm						
		Change Default File Association		Indicator Removal from Tools						
				Rootkit						

THE MITRE ATT&CK™ ENTERPRISE FRAMEWORK

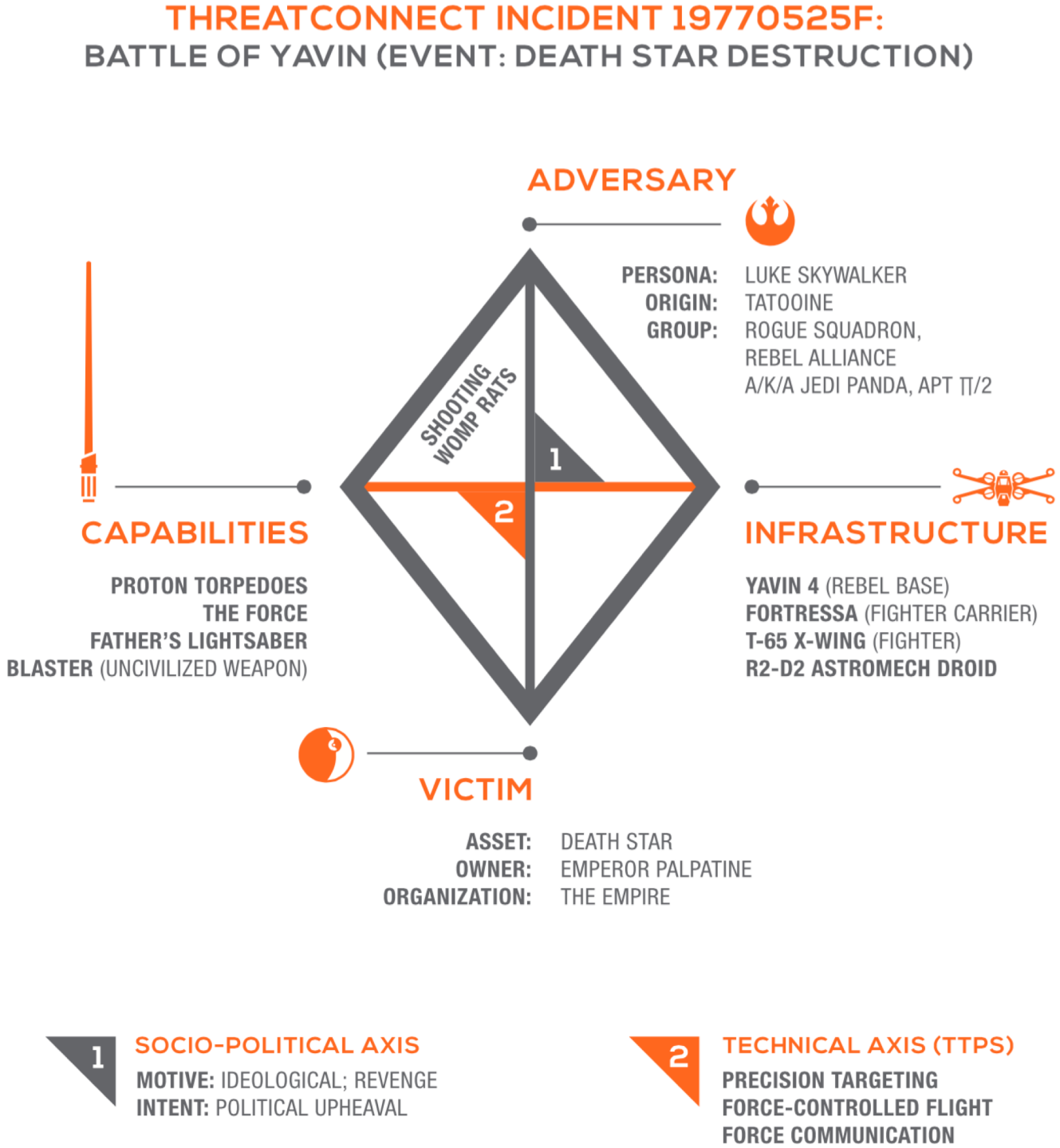
ATTACK.MITRE.ORG

ATT&CK™

MITRE

Diamond Model

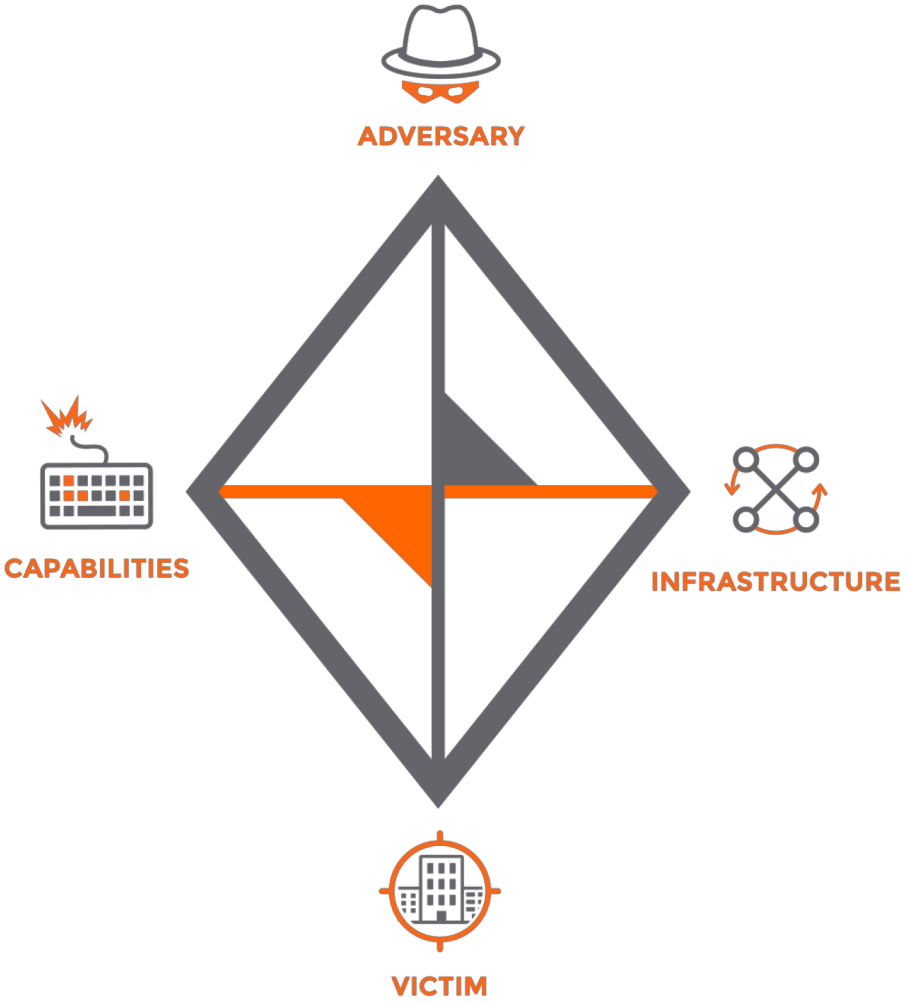
- More often used within Threat Intelligence, but has a place as part of Threat Hunting
- Used for contextualizing threat intelligence that is found during hunting
- Sergio Caltagirone, Andrew Pendergast, Christopher Betz
  - http://www.dtic.mil/dtic/tr/fulltext/u2/a586960.pdf
  - https://threatconnect.com/blog/diamond-model-threat-intelligence-star-wars/



How Are You Going To Hunt?

- Four Vertices to the Diamond Model
- Focus your hunt on any one of them to start

- Victim and Capability are generally best places to start



http://www.activeresponse.org/building-threat-hunting-strategy-with-the-diamond-model/

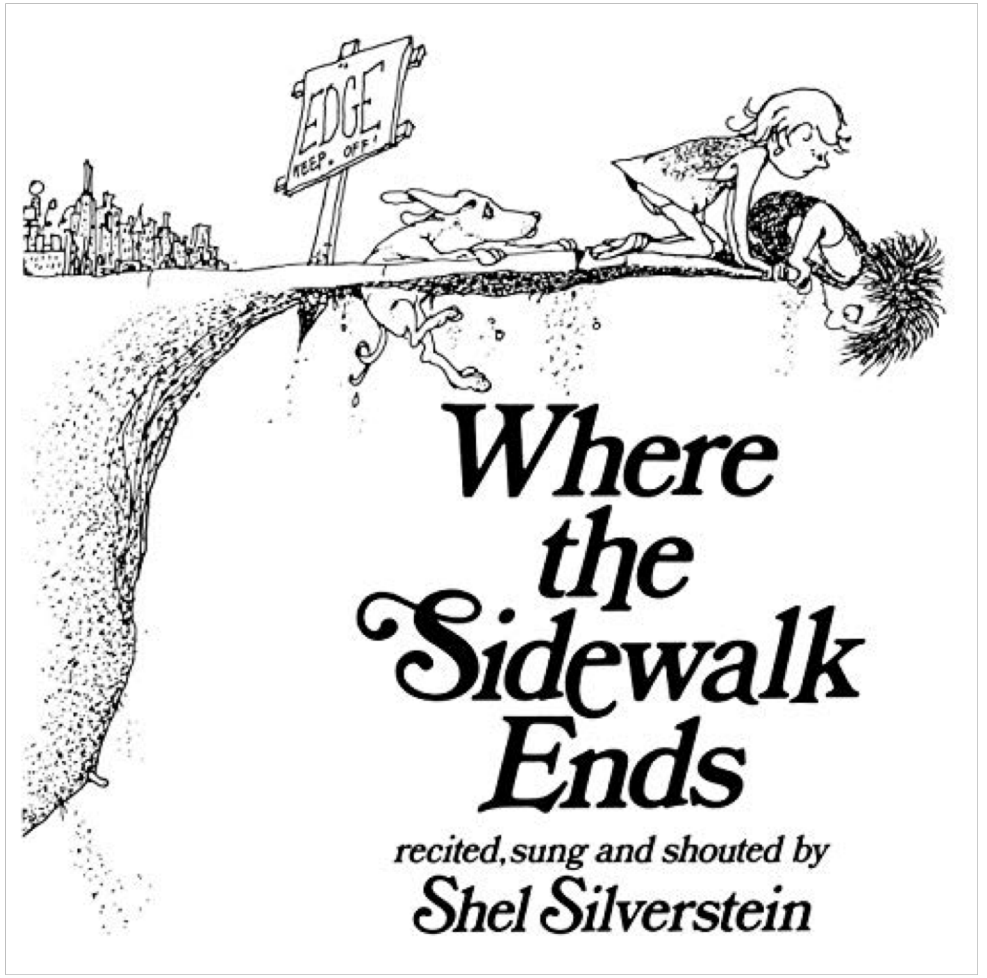

Threat Hunting
ATT&CK™
Threat Intelligence
Security Operations

Time Is A Crucial Factor

- Don’t Get Myopic on Your Hunt
- Start broadly and narrow so you don’t miss events
- Much of your data is time series data

Uncovering Unexpected Things

- Hunting against a hypothesis
  - Can take you in many directions
  - Note those turns so you can retrace your steps
  - Start new hunts when you reach a dead end



Hunts Do Not Exist in a Silo

- Techniques will cross paths with other techniques
- Use the techniques as guardrails

- Example: Hunting for PowerShell as the technique could yield the data encoding technique
  - Could we hunt just for data encoding?

Using ATT&CK Techniques To Build Our Hypothesis - PowerShell

ID: T1086

Tactic: Execution

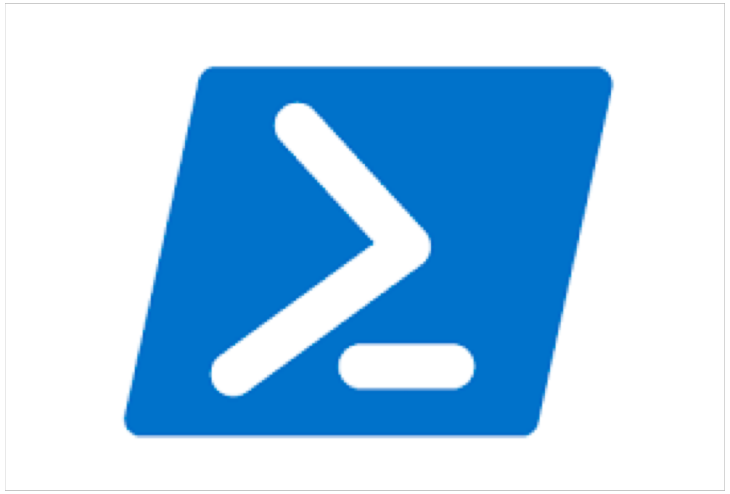
Platform: Windows

Permissions Required: User, Administrator

Data Sources: Windows Registry, File monitoring, Process monitoring, Process command-line parameters

Supports Remote: Yes

Version: 1.0



https://attack.mitre.org/wiki/Technique/T1086

Adversaries will use PowerShell Empire to establish a foothold and carry out attacks

How Might We Confirm or Refute Our Hypothesis?

- What is PowerShell?
- Where can I learn more about PowerShell Empire?
- Does PowerShell Empire have default settings that I could hunt for?
- What do data flows look like between sources and destinations?
- What user accounts are being used?
- What ports are being used?
- When did events occur?
- Are we able to see the contents of the scripts PowerShell is running to gain greater understanding?

Notional Flow of PSE Hunt

Indicators in SSL

Pivot to Find Internal/External IPs

- Firewall/IDS/ Wire
- User Agent Strings
- Domains

- Network Indicators
- Host attributes (User, Event Descriptions)

- Unique combinations of user/systems
- Pivot into specific users/systems of interest

- Map interesting processes
- Additional attributes: Parent Processes, Encoding, accounts used, sequencing of commands

- Research commands uncovered
- Compare commands executed on systems

Chaining Events Together

```
index=botsv2 sourcetype=XmlWinEventLog:Microsoft-Windows-Sysmon/Operational (CommandLine=*powershell*-enc* OR  ParentCommandLine=*powershell*-enc*) (host=wrk-btun OR host=mercury)
| eval shortCL=substr(CommandLine,1,90) | eval shortPCL=substr(ParentCommandLine,1,80)
| table _time host user shortPCL ParentProcessId ProcessId shortCL
| sort + _time
```

from Aug 22 through Aug 26, 2017

✓ 17 events (8/22/17 12:00:00.000 AM to 8/27/17 12:00:00.000 AM) No Event Sampling Job Smart Mode

Events Patterns **Statistics (17)** Visualization

20 Per Page Format Preview

_time	host	user	shortPCL	ParentProcessId	ProcessId	shortCL
2017-08-23 20:29:08	wrk-btun	FROTHLY\billy.tun	C:\Windows\system32\wbem\wmiprvse.exe -secured -Embedding	2240	4976	powershell -noP -sta -w 1 -enc WwBSAEUARgBdAC4AQQBTAFMARQBtAGIAbABZAC4ARwBlAFQAVABZAFAAZQ
2017-08-23 20:31:59	wrk-btun	FROTHLY\billy.tun	powershell -noP -sta -w 1 -enc WwBSAEUARgBdAC4AQQBTAFMARQBtAGIAbABZAC4ARwBlAFQA	4976	1512	"C:\Windows\system32\whoami.exe" /groups
2017-08-23 20:32:00	wrk-btun	FROTHLY\billy.tun	"C:\Windows\system32\eventvwr.exe"	3800	4468	"C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe" -NoP -NonI -c $x=$((gp HKCU:So
2017-08-23 20:32:00	wrk-btun	FROTHLY\billy.tun	powershell -noP -sta -w 1 -enc WwBSAEUARgBdAC4AQQBTAFMARQBtAGIAbABZAC4ARwBlAFQA	4976	3800	"C:\Windows\system32\eventvwr.exe"
2017-08-23 20:32:00	wrk-btun	FROTHLY\billy.tun	powershell -noP -sta -w 1 -enc WwBSAEUARgBdAC4AQQBTAFMARQBtAGIAbABZAC4ARwBlAFQA	4976	3816	"C:\Windows\system32\eventvwr.exe"
2017-08-23 20:32:00	wrk-btun	FROTHLY\billy.tun	powershell -noP -sta -w 1 -enc WwBSAEUARgBdAC4AQQBTAFMARQBtAGIAbABZAC4ARwBlAFQA	4976	4396	"C:\Windows\system32\whoami.exe" /groups
2017-08-23 20:32:01	wrk-btun	FROTHLY\billy.tun	"C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe" -NoP -NonI -c $x=$((	4468	3712	"C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe" -NoP -NonI -W Hidden -enc WwB
2017-08-23 20:33:29	wrk-btun	FROTHLY\billy.tun	"C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe" -NoP -NonI -W Hidde	3712	4456	"C:\Windows\system32\netsh.exe" advfirewall set allprofiles state off

Visualizing the Chaining of Events

Parent Process IDs and Process IDs

```
index=botsv2 sourcetype=XmlWinEventLog:Microsoft-Windows-Sysmon/Operational (CommandLine=*powershell*-enc* OR  ParentCommandLine=*powershell*-enc*) (host=wrk-btun OR host=mercury)
| stats count by ParentProcessId ProcessId
```

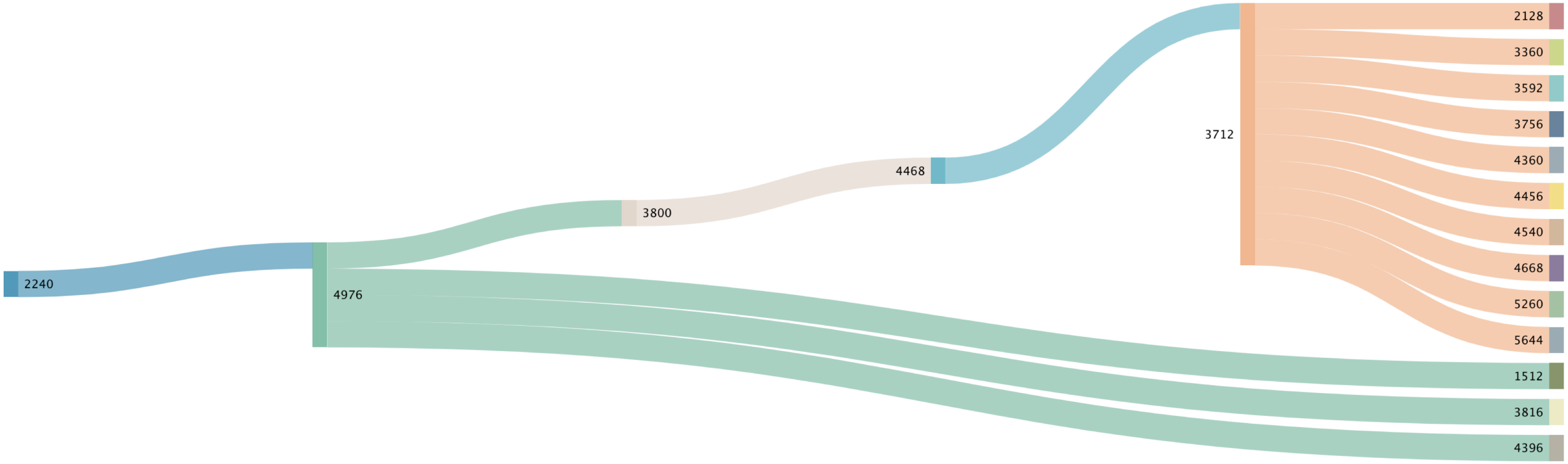
from Aug 22 through Aug 26, 2017

✓ 17 events (8/22/17 12:00:00.000 AM to 8/27/17 12:00:00.000 AM) No Event Sampling

Job Smart Mode

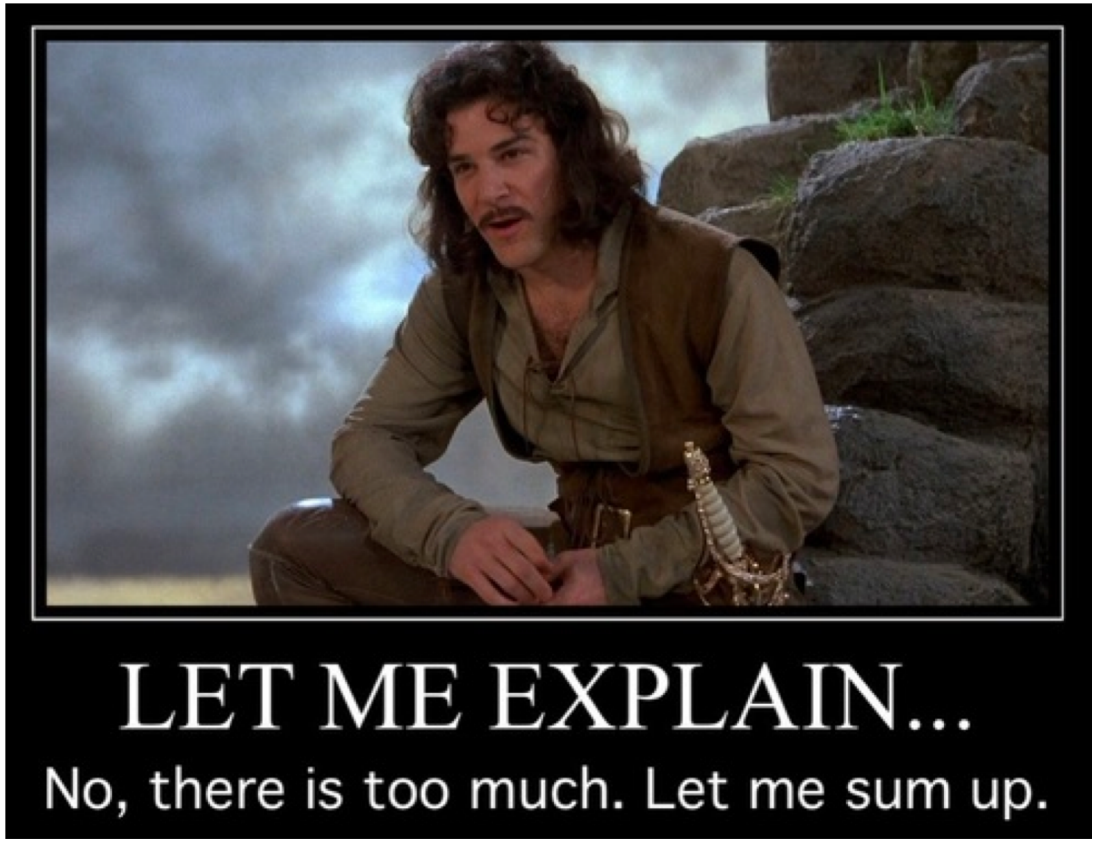
Events Patterns Statistics (17) Visualization

Sankey Diagram Format Trellis



As We Conclude A Hunt…

- Were we able to confirm or refute our hypothesis?
- What have we learned?
- What does our attack picture look like?
- How do our findings map to the diamond model?
- What other techniques were referenced?
- What should we operationalize?
- Where are our gaps?



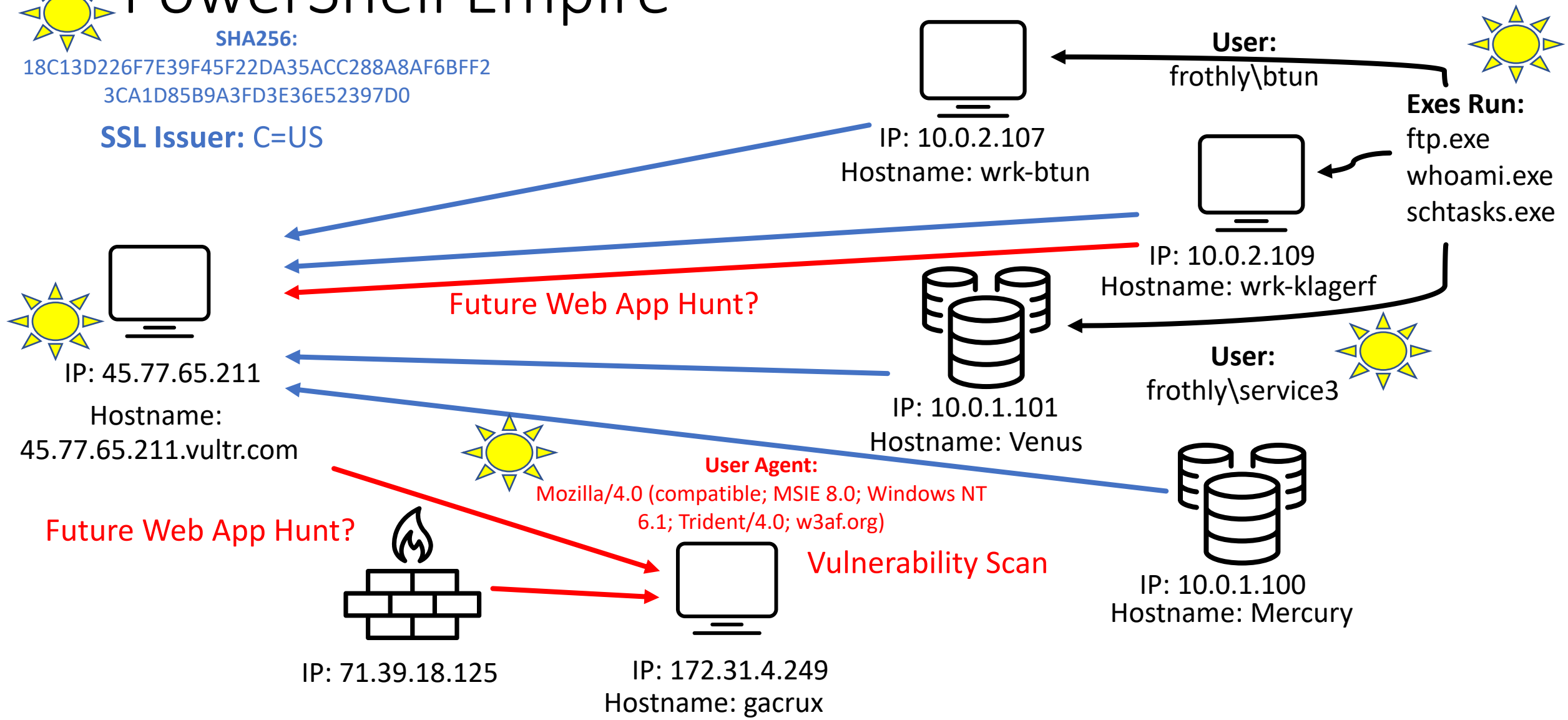
What Have We Learned?

- The default SSL Issuer value?
- Communication using this SSL Certificate exists between which systems?
- Is there outbound communication?
  - Between what systems?
  - Large or small percentage of overall traffic
  - What accounts are they associated with?
- Are specific processes running on systems?
  - Are they running under specific accounts?
  - Are they running in a specific order?
  - Are they all running encoded PowerShell?
  - Does anyone else see similar behavior by some variance?
- What other commands are being spawned?
- Can any of these nuggets found be found more broadly on the internet?

PowerShell Empire

SHA256:
18C13D226F7E39F45F22DA35ACC288A8AF6BFF2
3CA1D85B9A3FD3E36E52397D0

SSL Issuer: C=US



1 SOCIO-POLITICAL AXIS

ADVERSARY

CAPABILITIES

- PowerShell Empire

INFRASTRUCTURE

2 TECHNICAL AXIS

- Self signed SSL/TLS certificates

VICTIMS

Western innovative Brewers and Home Brewing companies

THREATCONNECT

MITRE ATT&CK

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command And Control
10 items	27 items	42 items	21 items	53 items	15 items	20 items	15 items	13 items	9 items	20 items
Drive-by Compromise	CMSTP	Accessibility Features	Access Token Manipulation	Access Token Manipulation	Account Manipulation	Account Discovery	Application Deployment Software	Audio Capture	Automated Exfiltration	Commonly Used Port
Exploit Public-Facing Application	Command-Line Interface	Account Manipulation	Accessibility Features	Binary Padding	Brute Force	Application Window Discovery	Distributed Component Object Model	Automated Collection	Data Compressed	Communication Through Removable Media
Hardware Additions	Compiled HTML File	AppCert DLLs	AppCert DLLs	BITS Jobs	Credential Dumping	Browser Bookmark Discovery	Exploitation of Remote Services	Clipboard Data	Data Encrypted	Connection Proxy
Replication Through Removable Media	Control Panel Items	AppInit DLLs	AppInit DLLs	Bypass User Account Control	Credentials in Files	File and Directory Discovery	Logon Scripts	Data from Information Repositories	Data Transfer Size Limits	Custom Command and Control Protocol
Spearphishing Attachment	Dynamic Data Exchange	Application Shimming	Application Shimming	CMSTP	Credentials in Registry	Network Service Scanning	Pass the Hash	Data from Local System	Exfiltration Over Alternative Protocol	Custom Cryptographic Protocol
Spearphishing Link	Execution through API	Authentication Package	Bypass User Account Control	Code Signing	Exploitation for Credential Access	Network Share Discovery	Pass the Ticket	Data from Network Shared Drive	Exfiltration Over Command and Control Channel	Data Encoding
Spearphishing via Service	Execution through Module Load	BITS Jobs	DLL Search Order Hijacking	Compiled HTML File	Forced Authentication	Network Sniffing	Remote Desktop Protocol	Data from Removable Media	Exfiltration Over Other Network Medium	Data Obfuscation
Supply Chain Compromise	Exploitation for Client Execution	Bootkit	Exploitation for Privilege Escalation	Component Firmware	Hooking	Password Policy Discovery	Remote File Copy	Data Staged	Exfiltration Over Physical Medium	Domain Fronting
Trusted Relationship	Graphical User Interface	Browser Extensions	Extra Window Memory Injection	Component Object Model Hijacking	Input Capture	Peripheral Device Discovery	Remote Services	Email Collection	Scheduled Transfer	Fallback Channels
Valid Accounts	InstallUtil	Change Default File Association	File System Permissions Weakness	Control Panel Items	Kerberoasting	Permission Groups Discovery	Replication Through Removable Media	Input Capture		Multi-hop Proxy
	LSASS Driver	Component Firmware	Hooking	DCShadow	LLMNR/NBT-NS Poisoning	Process Discovery	Shared Webroot	Man in the Browser		Multi-Stage Channels
	Mshta	Component Object Model Hijacking	Image File Execution Options Injection	Deobfuscate/Decode Files or Information	Network Sniffing	Query Registry	Taint Shared Content	Screen Capture		Multiband Communication
	PowerShell	Create Account	New Service	Disabling Security Tools	Password Filter DLL	Remote System Discovery	Third-party Software	Video Capture		Multilayer Encryption
	Regsvcs/Regasm	DLL Search Order Hijacking	Path Interception	DLL Search Order Hijacking	Private Keys	Security Software Discovery	Windows Admin Shares			Remote Access Tools
	Regsvr32	External Remote Services	Port Monitors	DLL Side-Loading	Two-Factor Authentication Interception	System Information Discovery	Windows Remote Management			Remote File Copy
	Rundll32	File System Permissions Weakness	Process Injection	Exploitation for Defense Evasion		System Network Configuration Discovery				Standard Application Layer Protocol
	Scheduled Task	Hidden Files and Directories	Scheduled Task	Extra Window Memory Injection		System Network Connections Discovery				Standard Cryptographic Protocol
	Scripting	Hooking	Service Registry Permissions Weakness	File Deletion		System Owner/User Discovery				Standard Non-Application Layer Protocol
	Service Execution	Hypervisor	SID-History Injection	File Permissions Modification		System Service Discovery				Uncommonly Used Port
	Signed Binary Proxy Execution	Image File Execution Options Injection	Valid Accounts	File System Logical Offsets		System Time Discovery				Web Service
	Signed Script Proxy Execution	Logon Scripts	Web Shell	Hidden Files and Directories						
	Third-party Software	LSASS Driver		Image File Execution Options Injection						
	Trusted Developer Utilities	Modify Existing Service		Indicator Blocking						
	User Execution	Netsh Helper DLL		Indicator Removal from Tools						
	Windows Management Instrumentation	New Service		Indicator Removal on Host						
	Windows Remote Management	Office Application Startup		Indirect Command Execution						
				Install Root Certificate						

https://mitre.github.io/attack-navigator

MITRE PRE-ATT&CK

Priority Definition Planning	Priority Definition Direction	Target Selection	Technical Information Gathering	People Information Gathering	Organizational Information Gathering	Technical Weakness Identification	People Weakness Identification	Organizational Weakness Identification	Adversary Opsec	Establish & Maintain Infrastructure	Persona Development	Build Capabilities	Test Capabilities	Stage Capabilities
13 items	4 items	5 items	20 items	11 items	11 items	9 items	3 items	6 items	23 items	16 items	6 items	11 items	7 items	6 items
Assess current holdings, needs, and wants	Assign KITs, KIQs, and/or intelligence requirements	Determine approach/attack vector	Acquire OSINT data sets and information	Acquire OSINT data sets and information	Acquire OSINT data sets and information	Analyze application security posture	Analyze organizational skillsets and deficiencies	Analyze business processes	Acquire and/or use 3rd party infrastructure services	Acquire and/or use 3rd party infrastructure services	Build social network persona	Build and configure delivery systems	Review logs and residual traces	Disseminate removable media
Assess KITs/KIQs benefits	Receive KITs/KIQs and determine requirements	Determine highest level tactical element	Conduct active scanning	Aggregate individual's digital footprint	Conduct social engineering	Analyze architecture and configuration posture	Analyze social and business relationships, interests, and affiliations	Analyze organizational skillsets and deficiencies	Acquire and/or use 3rd party software services	Acquire and/or use 3rd party software services	Choose pre-compromised mobile app developer account credentials or signing keys	Build or acquire exploits	Test ability to evade automated mobile application security analysis performed by app stores	Distribute malicious software development tools
Assess leadership areas of interest	Submit KITs, KIQs, and intelligence requirements	Determine operational element	Conduct passive scanning	Conduct social engineering	Determine 3rd party infrastructure services	Analyze data collected	Assess targeting options	Analyze presence of outsourced capabilities	Acquire or compromise 3rd party signing certificates	Acquire or compromise 3rd party signing certificates	Choose pre-compromised persona and affiliated accounts	C2 protocol development	Test callback functionality	Friend/Follow/Connect to targets of interest
Assign KITs/KIQs into categories	Task requirements	Determine secondary level tactical element	Conduct social engineering	Identify business relationships	Determine centralization of IT management	Analyze hardware/software security defensive capabilities		Assess opportunities created by business deals	Anonymity services	Buy domain name	Develop social network persona digital footprint	Compromise 3rd party or closed-source vulnerability/exploit information	Test malware in various execution environments	Hardware or software supply chain implant
Conduct cost/benefit analysis		Determine strategic target	Determine 3rd party infrastructure services	Identify groups/roles	Determine physical locations	Analyze organizational skillsets and deficiencies		Assess security posture of physical locations	Common, high volume protocols and software	Compromise 3rd party infrastructure to support delivery	Friend/Follow/Connect to targets of interest	Create custom payloads	Test malware to evade detection	Port redirector
Create implementation plan			Determine domain and IP address space	Identify job postings and needs/gaps	Dumpster dive	Identify vulnerabilities in third-party software libraries		Assess vulnerability of 3rd party vendors	Compromise 3rd party infrastructure to support delivery	Create backup infrastructure	Obtain Apple iOS enterprise distribution key pair and certificate	Create infected removable media	Test physical access	Upload, install, and configure software/tools
Create strategic plan			Determine external network trust dependencies	Identify people of interest	Identify business processes/tempo	Research relevant vulnerabilities/CVEs			Data Hiding	Domain registration hijacking		Discover new exploits and monitor exploit-provider forums	Test signature detection for file upload/email filters	
Derive intelligence requirements			Determine firmware version	Identify personnel with an authority/privilege	Identify business relationships	Research visibility gap of security vendors			DNSCalc	Dynamic DNS		Identify resources required to build capabilities		
Develop KITs/KIQs			Discover target logon/email address format	Identify sensitive personnel information	Identify job postings and needs/gaps	Test signature detection			Domain Generation Algorithms (DGA)	Install and configure hardware, network, and systems		Obtain/re-use payloads		
Generate analyst intelligence requirements			Enumerate client configurations	Identify supply chains	Identify supply chains				Dynamic DNS	Obfuscate		Post compromise tool development		
Identify analyst level gaps				Mine social media	Obtain templates/branding materials							Remote access tool development		

https://mitre.github.io/attack-navigator

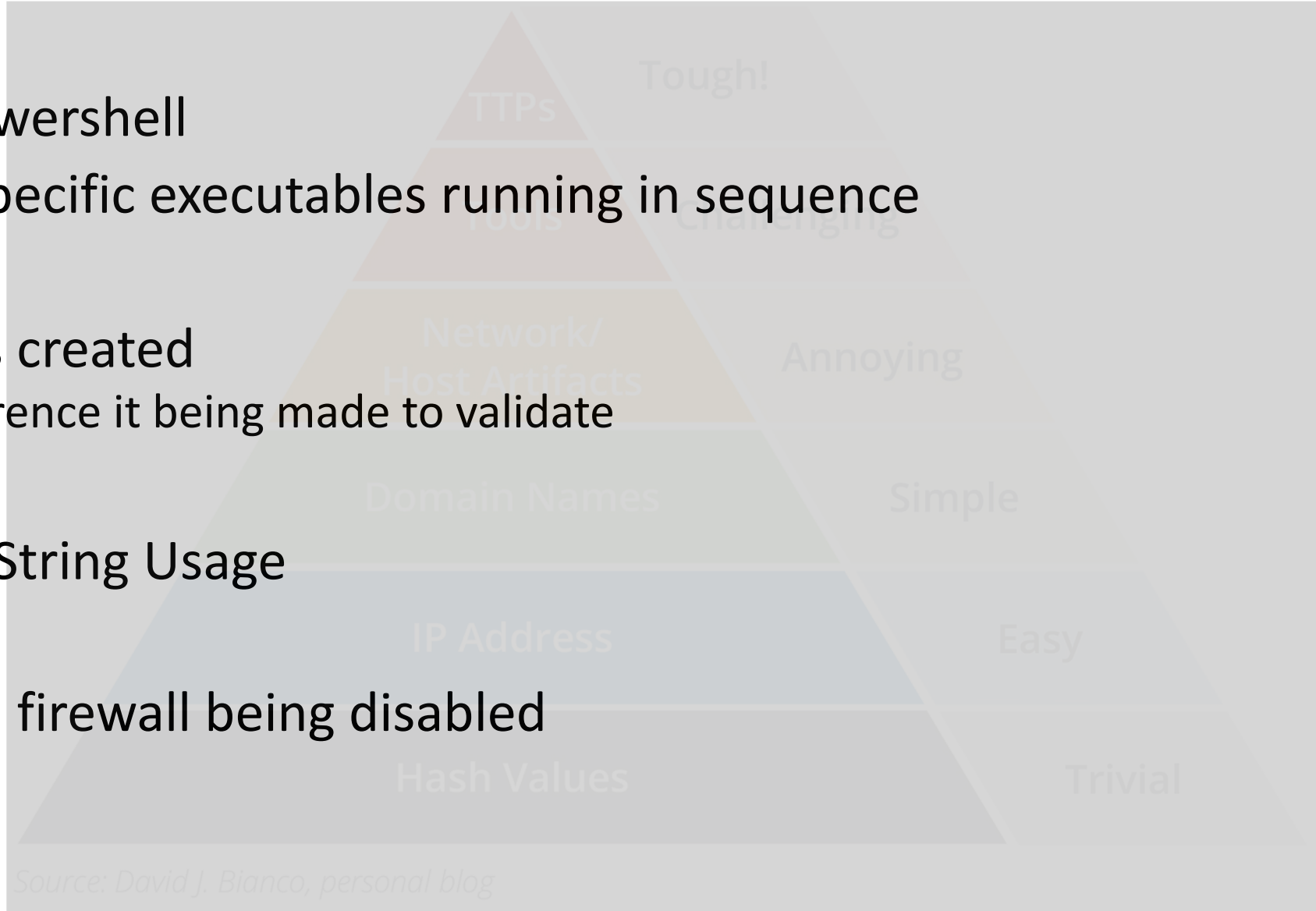
Operationalize Your Findings

- Create Feedback Loop from Hunting to Incident Response

- “End goal of hunting should be a change in policy or procedure - operationalization, don’t do the same thing over and over again”
  - Threat Hunting Webshells with Splunk, James Bower

What Could We Operationalize?

- Alert on encoded Powershell
- Alert when we see specific executables running in sequence
- Alert on SSL Issuer
- Detect new accounts created
  - Have a ticket to reference it being made to validate
- Blacklist IP Address
- Monitor User Agent String Usage
- Monitor for URIs
- Monitor and alert on firewall being disabled



Source: David J. Bianco, personal blog

Considerations when operationalizing ATT&CK

Example: Scheduled Task (T1053)

Tactic	TechniqueName	Tech ni	Data Source 1	Data Source 2	Data Source 3	Data Source 4
Execution,Persistence,Privilege Escalation	Scheduled Task	T1053	4688 Process CMD Line	4688 Process Execution	4663 File monitoring	Windows event logs

“Monitor scheduled task creation from common utilities using command-line invocation. Legitimate scheduled tasks may be created during installation of new software or through system administration functions. Monitor process execution from the svchost.exe in Windows 10 and the Windows Task Scheduler taskeng.exe for older versions of Windows.”

https://attack.mitre.org/techniques/T1053/

https://www.malwarearchaeology.com/cheat-sheets

```yaml
title: Scheduled Task Creation
status: experimental
description: Detects the creation of scheduled tasks in user session
author: Florian Roth
logsource:
    category: process_creation
    product: windows
detection:
    selection:
        Image: '*\schtasks.exe'
        CommandLine: '* /create *'
    filter:
        User: NT AUTHORITY\SYSTEM
    condition: selection and not filter
fields:
    - CommandLine
    - ParentCommandLine
tags:
    - attack.execution
    - attack.persistence
    - attack.privilege_escalation
    - attack.t1053
    - attack.s0111
falsepositives:
    - Administrative activity
    - Software installation
level: low
```

https://github.com/Neo23x0/sigma/blob/master/rules/windows/process_creation/win_susp_schtask_creation.yml

Operationalizing Technique (Scheduled Tasks)

- Monitor for
  - Schtasks.exe that deviate from an IT baseline
    - Need excellent coordination with IT to build lookup of standard tasks to look for outliers
    - Could be noisy depending on the frequency
  - Scheduled task names that don’t match with the IT standard
    - Compromised system could be using an IT standard and this would not be seen
  - Scheduled tasks running under unexpected users
    - Should tasks run as system or as a named user?
  - Scheduled tasks that have command strings out of the normal
    - Should PowerShell scripts be running as scheduled tasks, for some organizations yes, for others no

<Image condition="begin with" name="technique_id=T1036,technique_name=Masquerading">C:\Windows\security\</Image>
<Image condition="image">odbcconf.exe</Image>
<Image condition="image" name="technique_id=T1033,technique_name=System Owner/User Discovery">PsGetSID.exe</Image>
<Image condition="image" name="technique_id=T1033,technique_name=System Owner/User Discovery">whoami.exe</Image>
<Image condition="image" name="technique_id=T1070,technique_name=Indicator Removal on Host">wevtutil.exe</Image>
<Image condition="image" name="technique_id=T1057,technique_name=Process Discovery">PipeList.exe</Image>
<Image condition="image">hh.exe</Image>
<Image condition="image" name="technique_id=T1028,technique_name=Windows Remote Management">wsmprovhost.exe</Image>
<Image condition="image" name="technique_id=T1049,technique_name=System Network Connections Discovery">netstat.exe</Image>
<Image condition="contains" name="technique_id=T1036,technique_name=Masquerading">\wwwroot\</Image>
<CommandLine condition="contains" name="technique_id=T1196,technique_name=Control Panel Items">control.exe /name</CommandLine>
<CommandLine condition="contains" name="technique_id=T1054,technique_name=Indicator Blocking">fltmc unload</CommandLine>
<CommandLine condition="contains" name="technique_id=T1003,technique_name=Credential Dumping">-ma lsass.exe</CommandLine>
<CommandLine condition="contains" name="technique_id=T1196,technique_name=Control Panel Items">rundll32.exe shell32.dll,Control_RunDLL
<CommandLine condition="contains" name="technique_id=T1063,technique_name=Security Software Discovery">misc::mflt</CommandLine>
<CommandLine condition="contains" name="technique_id=T1027,technique_name=Obfuscated Files or Information">^</CommandLine>
<CommandLine condition="contains" name="technique_id=T1089,technique_name=Disabling Security Tools">DisableIOAVProtection</CommandLine
<CommandLine condition="contains" name="technique_id=T1089,technique_name=Disabling Security Tools">RemoveDefinitions</CommandLine>
<CommandLine condition="contains" name="technique_id=T1118,technique_name=InstallUtil">/logfile= /LogToConsole=false /U</CommandLine>
<CommandLine condition="contains" name="technique_id=T1089,technique_name=Disabling Security Tools">Add-MpPreference</CommandLine>
<ParentImage condition="image" name="technique_id=T1059,technique_name=Command-Line Interface">cmd.exe</ParentImage>
<ParentImage condition="image" name="technique_id=T1015,technique_name=Accessibility Features">utilman.exe</ParentImage>
<ParentImage condition="image" name="technique_id=T1015,technique_name=Accessibility Features">DisplaySwitch.exe</ParentImage>
<ParentImage condition="image" name="technique_id=T1015,technique_name=Accessibility Features">sethc.exe</ParentImage>
<ParentImage condition="image" name="technique_id=T1202,technique_name=Indirect Command Execution">wscript.exe</ParentImage>
<ParentImage condition="image" name="technique_id=T1202,technique_name=Indirect Command Execution">control.exe</ParentImage>
<ParentImage condition="image" name="technique_id=T1202,technique_name=Indirect Command Execution">cscript.exe</ParentImage>
<ParentImage condition="image" name="technique_id=T1088,technique_name=Bypass User Account Control">fodhelper.exe</ParentImage>
<ParentImage condition="image" name="technique_id=T1088,technique_name=Bypass User Account Control">eventvwr.exe</ParentImage>
<ParentImage condition="image" name="technique_id=T1015,technique_name=Accessibility Features">osk.exe</ParentImage>
<ParentImage condition="image" name="technique_id=T1086,technique_name=PowerShell">powershell.exe</ParentImage>
<ParentImage condition="image" name="technique_id=T1086,technique_name=PowerShell">powershell_ise.exe</ParentImage>

```
<Image condition="image" name="technique_id=T1070,technique_name=Indicator Removal on Host">wevtutil.exe</Image>
```

```
<CommandLine condition="contains" name="technique_id=T1089,technique_name=Disabling Security Tools">RemoveDefinitions</CommandLine>
```

```
<ParentImage condition="image" name="technique_id=T1086,technique_name=PowerShell">powershell.exe</ParentImage>
```

https://github.com/olafhartong/sysmon-modular

```
sourcetype="xmlwineventlog:microsoft-windows-sysmon/operational" RuleName=* powershell.exe ParentImage=* | table Image ParentImage RuleName
```

Last 24 hours

✓ 29 events (2/14/19 10:00:00.000 PM to 2/15/19 10:14:43.000 PM) No Event Sampling Job Smart Mode

Events Patterns **Statistics (29)** Visualization

20 Per Page Format Preview

< Prev 1 2 Next >

Image	ParentImage	RuleName
C:\Windows\Microsoft.NET\Framework64\v2.0.50727\csc.exe	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	technique_id=T1086,technique_name=PowerShell
C:\Windows\System32\whoami.exe	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	technique_id=T1033,technique_name=System Owner/User Discovery
C:\Windows\System32\ftp.exe	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	technique_id=T1086,technique_name=PowerShell
C:\Windows\System32\netsh.exe	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	technique_id=T1063,technique_name=Security Software Discovery
C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	technique_id=T1086,technique_name=PowerShell
C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	C:\Windows\System32\eventvwr.exe	technique_id=T1086,technique_name=PowerShell
C:\Windows\System32\eventvwr.exe	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	technique_id=T1086,technique_name=PowerShell
C:\Windows\System32\eventvwr.exe	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	technique_id=T1086,technique_name=PowerShell
C:\Windows\System32\whoami.exe	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	technique_id=T1033,technique_name=System Owner/User Discovery
C:\Windows\System32\whoami.exe	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	technique_id=T1033,technique_name=System Owner/User Discovery
C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	C:\Windows\System32\wbem\WmiPrvSE.exe	technique_id=T1086,technique_name=PowerShell

MITRE ATT&CK

Edit | Export | ...

Sysmon Events associated with MITRE ATT&CK Techniques

src | dest | user | tactic | technique | Time Range

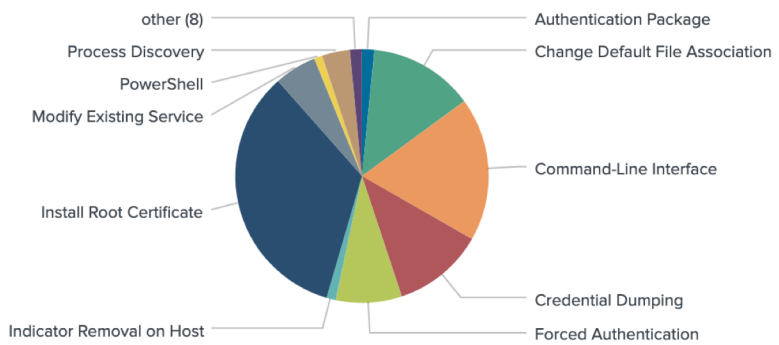
Persistence × Discovery × Credential Access × Defense Evasion × Execution × Lateral Movement ×

All ×

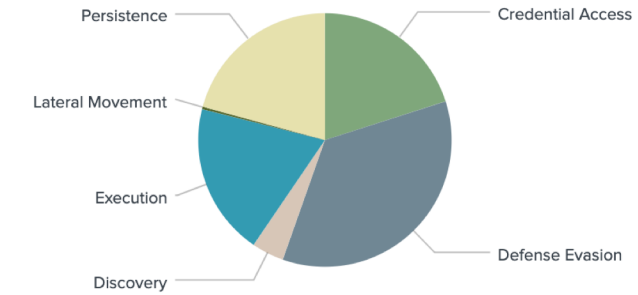
Feb 15, 2019

Submit | Hide Filters

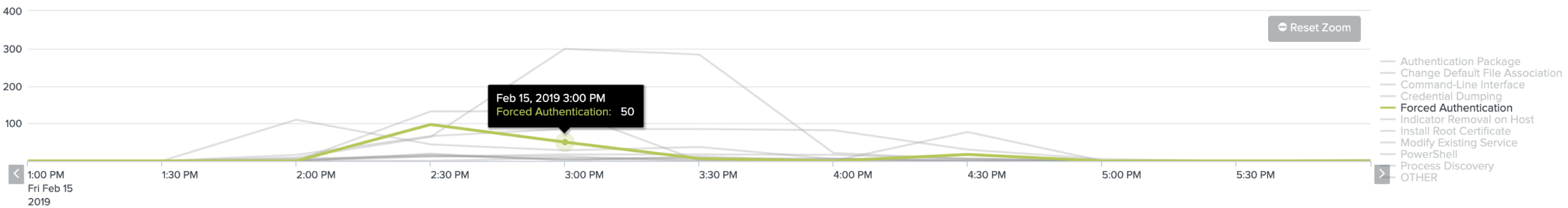
Technique



Tactic



Time Chart



ATT&CK - Windows Events

Edit | Export | ...

Windows Events Associated with MITRE ATT&CK Techniques

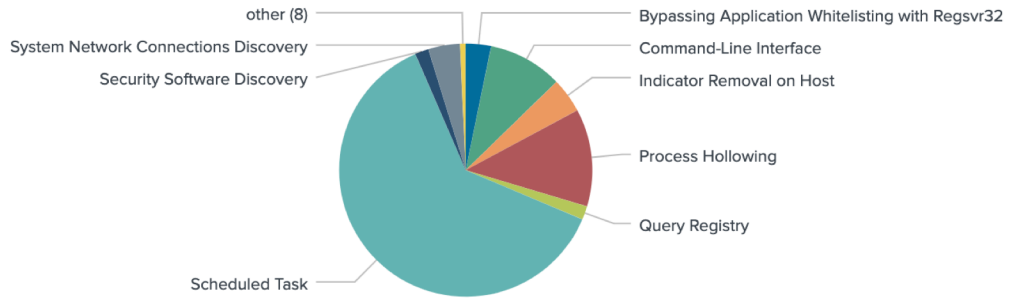
Techniques with Processes 3 Standard Deviations above the Mean

mitre_technique	count
PowerShell	14
Windows Management Instrumentation	6

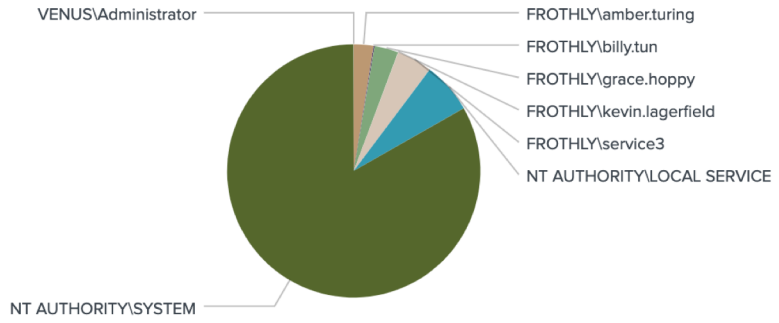
Techniques with Parent Processes 3 Standard Deviations above the Mean

mitre_technique	count
Indicator Removal on Host	970
Network Share Discovery	2
Scheduled Task	6
Security Software Discovery	2
System Network Configuration Discovery	2

Count by ATT&CK Technique



Count by User



_time	mitre_technique	event_description	process_command_line	user_name
2017-08-23 20:05:50	Command-Line Interface	Process Create	C:\Windows\system32\cmd.exe /c netstat -nao | findstr /r "LISTENING"	NT AUTHORITY\SYSTEM
2017-08-23 20:05:50	System Network Connections Discovery	Process Create	netstat -nao	NT AUTHORITY\SYSTEM
2017-08-23 20:06:23	Scheduled Task	Process Create	taskeng.exe {BFADB586-8B28-48D4-B32F-A9861BBE77C5} S-1-5-18:NT AUTHORITY\System:Service:	NT AUTHORITY\SYSTEM
2017-08-23 20:06:23	Scheduled Task	Process Create	taskeng.exe {BFADB586-8B28-48D4-B32F-A9861BBE77C5} S-1-5-18:NT AUTHORITY\System:Service:	NT AUTHORITY\SYSTEM
2017-08-23 20:06:50	Scheduled Task	Process Create	taskeng.exe {E1CE6623-6DEB-4878-A517-35CE0474C1EB} S-1-5-18:NT AUTHORITY\System:Service:	NT AUTHORITY\SYSTEM
2017-08-23 20:06:50	Scheduled Task	Process Create	taskeng.exe {E1CE6623-6DEB-4878-A517-35CE0474C1EB} S-1-5-18:NT AUTHORITY\System:Service:	NT AUTHORITY\SYSTEM

_time	mitre_technique	event_description	process_command_line
2017-08-23 20:22:07	Scheduled Task	Process Create	schtasks.exe /change /tn "Microsoft\Office\Office Automatic Updates" /enable
2017-08-23 20:22:07	Scheduled Task	Process Create	schtasks.exe /change /tn "Microsoft\Office\Office Automatic Updates" /enable
2017-08-23 20:29:08	Windows Management Instrumentation	Process Create	powershell -noP -sta -w 1 -enc WwBSAEUARgBdAC4AQQBTAFMARQBtAGIAbABZAC4ARwBlAFQAVABZAFAAZQAoACcAUwB5AHMAdABlAG0ALgBNAGEAbgBhAGcAZQBtAGUAbgB0AC4AQQB1AHQAbwBtAGEAdABpAG8AbgAuAEEAbQBzAGkAVQB0AGkAbABzACcAK(
2017-08-23 20:29:08	PowerShell	Process Create	powershell -noP -sta -w 1 -enc WwBSAEUARgBdAC4AQQBTAFMARQBtAGIAbABZAC4ARwBlAFQAVABZAFAAZQAoACcAUwB5AHMAdABlAG0ALgBNAGEAbgBhAGcAZQBtAGUAbgB0AC4AQQB1AHQAbwBtAGEAdABpAG8AbgAuAEEAbQBzAGkAVQB0AGkAbABzACcAK(
2017-08-23 20:29:08	PowerShell	Process Create	powershell -noP -sta -w 1 -enc WwBSAEUARgBdAC4AQQBTAFMARQBtAGIAbABZAC4ARwBlAFQAVABZAFAAZQAoACcAUwB5AHMAdABlAG0ALgBNAGEAbgBhAGcAZQBtAGUAbgB0AC4AQQB1AHQAbwBtAGEAdABpAG8AbgAuAEEAbQBzAGkAVQB0AGkAbABzACcAK(
2017-08-23 20:29:08	Windows Management Instrumentation	Process Create	powershell -noP -sta -w 1 -enc WwBSAEUARgBdAC4AQQBTAFMARQBtAGIAbABZAC4ARwBlAFQAVABZAFAAZQAoACcAUwB5AHMAdABlAG0ALgBNAGEAbgBhAGcAZQBtAGUAbgB0AC4AQQB1AHQAbwBtAGEAdABpAG8AbgAuAEEAbQBzAGkAVQB0AGkAbABzACcAK(
2017-08-23 20:29:55	Scheduled Task	Process Create	taskeng.exe {B9BCD9D8-1751-49D2-82DC-E34CC9778221} S-1-5-18:NT AUTHORITY\System:Service:
2017-08-23 20:29:55	Scheduled Task	Process Create	taskeng.exe {B9BCD9D8-1751-49D2-82DC-E34CC9778221} S-1-5-18:NT AUTHORITY\System:Service:
2017-08-23 20:31:27	Process Hollowing	Process Create	taskhost.exe $(Arg0)
2017-08-23 20:31:27	Process Hollowing	Process Create	taskhost.exe $(Arg0)

_time	mitre_technique	event_description	process_command_line	user_name
2017-08-23 20:43:29	Process Hollowing	Process Create	taskhost.exe $(Arg0)	NT AUTHORITY\LOCAL SERVICE
2017-08-23 20:43:29	Process Hollowing	Process Create	taskhost.exe $(Arg0)	NT AUTHORITY\LOCAL SERVICE
2017-08-23 20:44:35	Process Hollowing	Process Create	taskhost.exe $(Arg0)	NT AUTHORITY\LOCAL SERVICE
2017-08-23 20:44:35	Process Hollowing	Process Create	taskhost.exe $(Arg0)	NT AUTHORITY\LOCAL SERVICE
2017-08-23 20:45:03	Scheduled Task	Process Create	C:\Windows\system32\schtasks.exe" /Create /F /RU system /SC DAILY /ST 10:26 /TN Updater /TR "C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe -NonI -W hidden -c \"IEX ([Text.Encoding]::UNICODE.GetString([Convert]::FromBase64String((gp HKLM:\Software\Microsoft\Network debug).debug)))\"	FROTHLY\billy.tun
2017-08-23 20:45:03	Scheduled Task	Process Create	C:\Windows\system32\schtasks.exe" /Create /F /RU system /SC DAILY /ST 10:26 /TN Updater /TR "C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe -NonI -W hidden -c \"IEX ([Text.Encoding]::UNICODE.GetString([Convert]::FromBase64String((gp HKLM:\Software\Microsoft\Network debug).debug)))\"	FROTHLY\billy.tun
2017-08-23 20:48:54	Process Hollowing	Process Create	taskhost.exe	FROTHLY\billy.tun
2017-08-23 20:48:54	Process Hollowing	Process Create	taskhost.exe	FROTHLY\billy.tun
2017-08-23 20:52:08	Scheduled Task	Process Create	schtasks.exe /change /tn "Microsoft\Office\Office ClickToRun Service Monitor" /enable	NT AUTHORITY\SYSTEM
2017-08-23 20:52:08	Scheduled Task	Process Create	schtasks.exe /change /tn "Microsoft\Office\Office ClickToRun Service Monitor" /enable	NT AUTHORITY\SYSTEM

MITRE ATT&CK

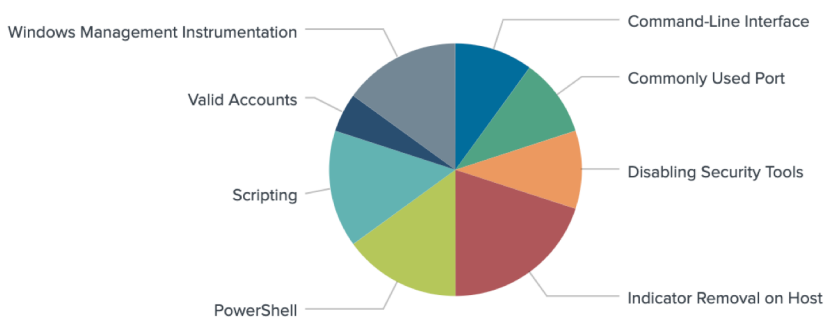
Edit | Export | ...

Notables associated with MITRE ATT&CK Techniques - Can be one notable to many techniques

src	dest	user	tactic	technique	notable status	Time Range
			All ×	All ×	All ×	Feb 15, 2019

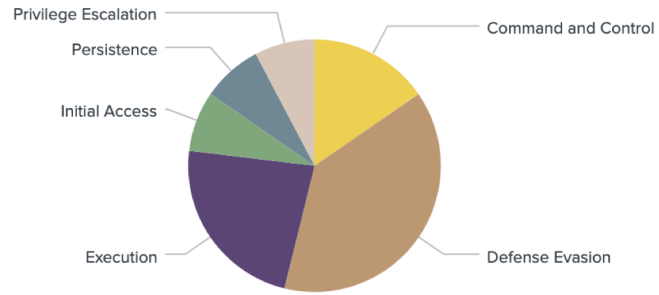
Submit | Hide Filters

Technique

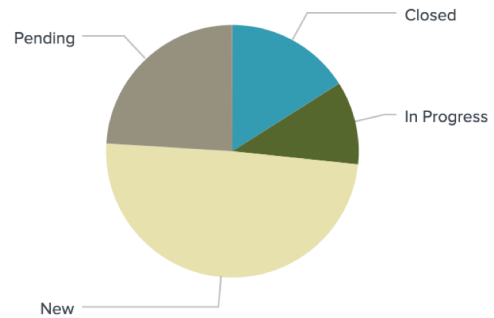


<1m ago

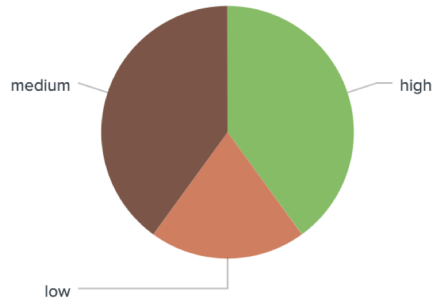
Tactic



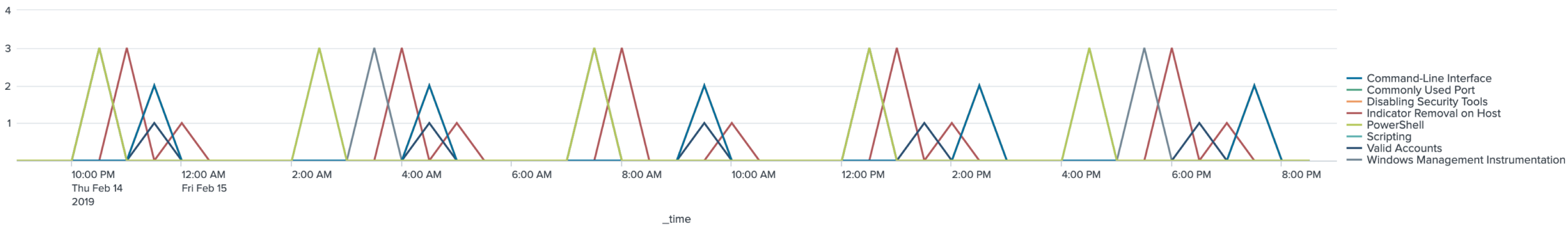
Status



Urgency



Time Chart



Detail

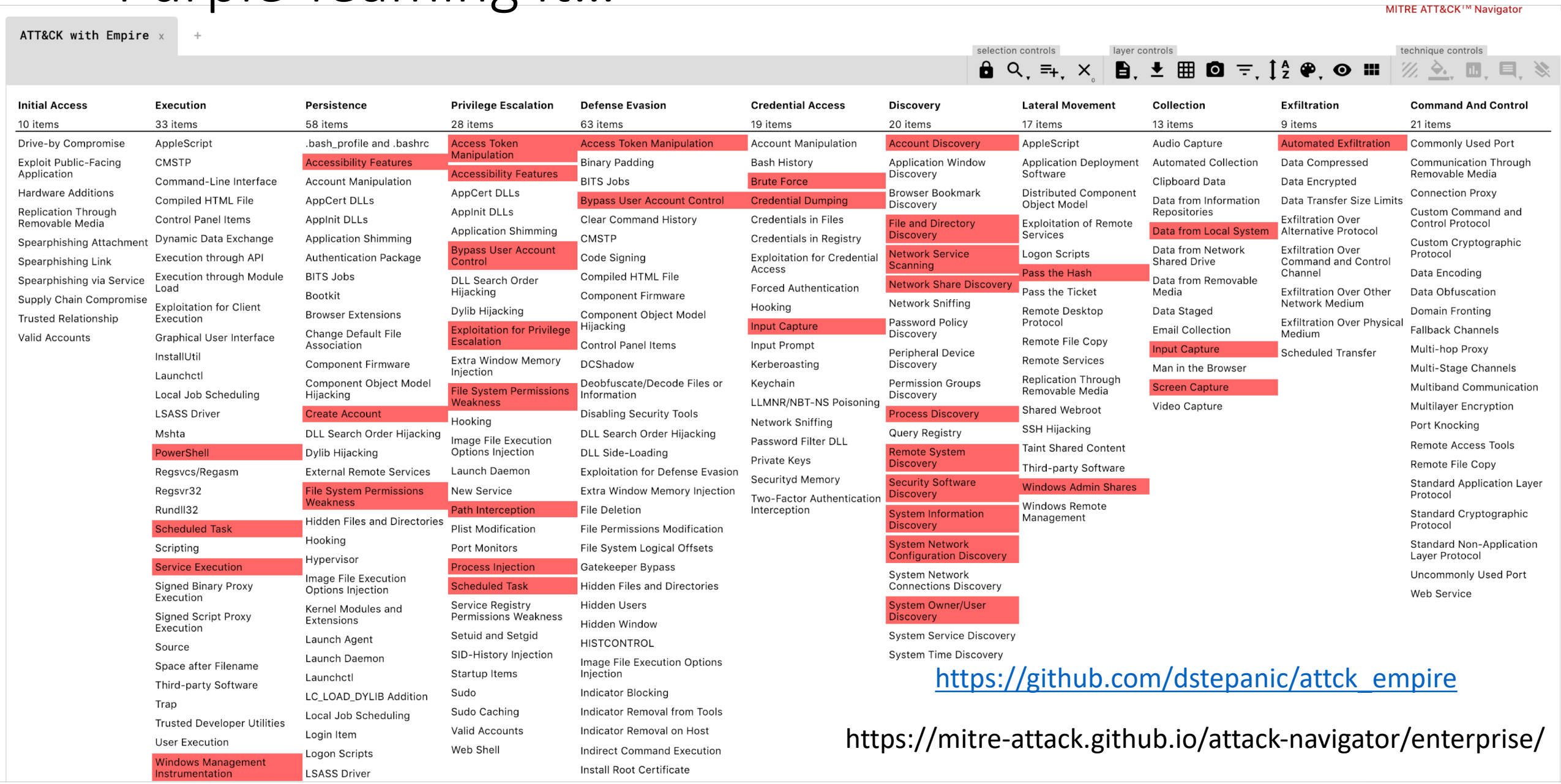
_time	src	dest	user	tactic	technique	rule_name	status_label	urgency
2019-02-14 22:33:29		wrk-btun.frothly.local	FROTHLY\billy.tun	Execution	Windows Management Instrumentation	Process Execution via WMI	New	low
2019-02-14 22:33:29		venus.frothly.local	FROTHLY\service3	Execution	Windows Management Instrumentation	Process Execution via WMI	New	high
2019-02-14 22:33:34		wrk-klagerf.frothly.local	FROTHLY\service3	Execution	Windows Management Instrumentation	Process Execution via WMI	New	high
2019-02-14 22:42:13		wrk-btun.frothly.local	FROTHLY\billy.tun	Execution Execution, Defense Evasion	PowerShell Scripting	Malicious PowerShell Process - Encoded Command	New	low
2019-02-14 22:42:13		venus.frothly.local	FROTHLY\service3	Execution Execution, Defense Evasion	PowerShell Scripting	Malicious PowerShell Process - Encoded Command	New	high

« prev 1 2 3 4 5 6 7 8 9 10 next »

Where Are Our Gaps?

- Credential Access is most glaring
  - Do we have logging to provide insight into this?
- Privilege Escalation is light
- Not a lot of Discovery seen to date
- Do we have data to address these gaps?
- These could be additional hunts

Purple Teaming It…



https://github.com/dstepanic/attck_empire

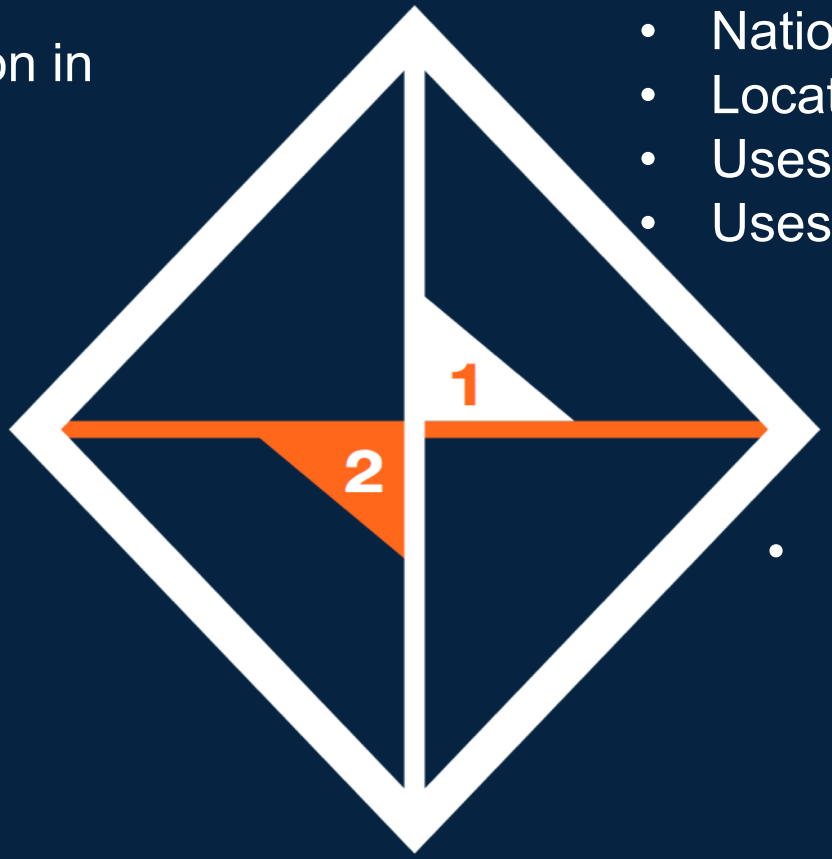
https://mitre-attack.github.io/attack-navigator/enterprise/

1 SOCIO-POLITICAL AXIS

- Seeking to obtain high end Western Beers for production in their breweries

ADVERSARY

- Nationstate sponsored adversary
- Located (+8.5 timezone)
- Uses Korean encoded language
- Uses Hancom Thinkfree Office



CAPABILITIES

- PowerShell Empire
- Spearphishing

INFRASTRUCTURE

- European VPS servers

2 TECHNICAL AXIS

- WMI lateral movement
- Self signed SSL/TLS certificates
- FTP/DNS Exfiltration
- Documents with .hwp suffix
- Korean fonts for English
- User svcvnc for Persistence
- Schtasks.exe for reboot persistence
- Naenara useragent string
- YMLP
- +8.5 hour time zone
- Korean text google translated to English

VICTIMS

Western innovative Brewers and Home Brewing companies

THREATCONNECT

Thanks ThreatConnect!

Additional Resources

- Hunting with Splunk Blog Series
  - https://www.splunk.com/blog/2017/07/06/hunting-with-splunk-the-basics.html
- Looking for Data Sets to Practice Against
  - Curated
    - https://www.splunk.com/blog/2018/05/03/introducing-the-security-datasets-project.html
    - http://live.splunk.com/splunk-security-dataset-project
  - DIY
    - https://www.splunk.com/blog/2018/05/10/boss-of-the-soc-scoring-server-questions-and-answers-and-dataset-open-sourced-and-ready-for-download.html
    - http://explore.splunk.com/BOTS_1_0_datasets
    - https://splunkbase.splunk.com/app/3985/
  - Version 2 of Our Dataset Will Be Available in April (Hopefully in the next week!)

More on MITRE ATT&CK

- https://attack.mitre.org/
  - https://medium.com/mitre-attack
- https://www.splunk.com/blog/2019/01/15/att-ck-ing-the-adversary-episode-1-a-new-hope.html
- https://www.splunk.com/blog/2019/02/04/att-ck-ing-the-adversary-episode-2-hunting-with-att-ck-in-splunk.html
- https://www.splunk.com/blog/2019/02/08/att-ck-ing-the-adversary-episode-3-operationalizing-att-ck-with-splunk.html

Thank You!

John Stoner

@stonerpsu