Enter Mordor

Pre-recorded Security Events from Simulated Adversarial Techniques

@Cyb3rWard0g

- Creator of
  - @HunterPlaybook
  - @THE_HELK
  - ATTACK-Python-Client
  - @OSSEM_Project
  - @Mordor_Project
  - Blacksmith & More
- Founder:
  - @HuntersForge

Agenda

- What is the goal of Purple Teaming?
- Purple Teaming Challenges?
  - Planning vs Execution
- Are we being effective at purple teaming?
- The Mordor Project
- Purple Teaming - Mordor Style
- Enabling the community!

What is the Goal of Purple Teaming?

Or at least one of the goals! :)

What can we do together?

Remember: It is NOT a Competition!

A Few Goals!

- Validate current detections
- Build new detections
- Brainstorm new tradecraft
- Map data to adversary actions
- Improve Telemetry
- **Learn from each other**
  - Red improves Blue

A Few Goals!

- Validate current detections
- Build new detections
- Brainstorm new tradecraft
- Map data to adversary actions
- Improve Telemetry
- **Learn from each other**
  - Red improves Blue
  - Blue improves Red too!

Believe me! You will always learn something new..

Rubeus

- Rubeus is a C# re-implementation of some of the functionality from Benjamin Delpy's Kekeo project
  - Kerberos structures built by hand...
  - Rubeus works nicely with execute-assembly
  - So why not use Kekeo? Because ASN.1!
    - Requires a commercial ASN.1 library to customize/rebuild the Kekeo codebase
- **Author: Will Schroeder @harmj0y**

Rubeus: over-pass-the-hash

Mimikatz pth

- "sacrificial" logon session that doesn't interact with the current logon session.
- Opens the LSASS process
- Patches associated logon session with hash/key.
- Normal Kerberos authentication process Kicks off (Hash->TGT)

Rubeus asktgt (createnetonly)

- CreateProcessWithLogonW Function to create a sacrificial process/logon session
- Rubeus builds/crafts AS-REQ (w/ preauth)
- TGT request successful
- Imports TGT to logon session (Hash->TGT)

https://www.youtube.com/watch?v=QCDBjFJ_C3g

Rubeus askTGT does NOT touch LSASS!

Mimikatz pth

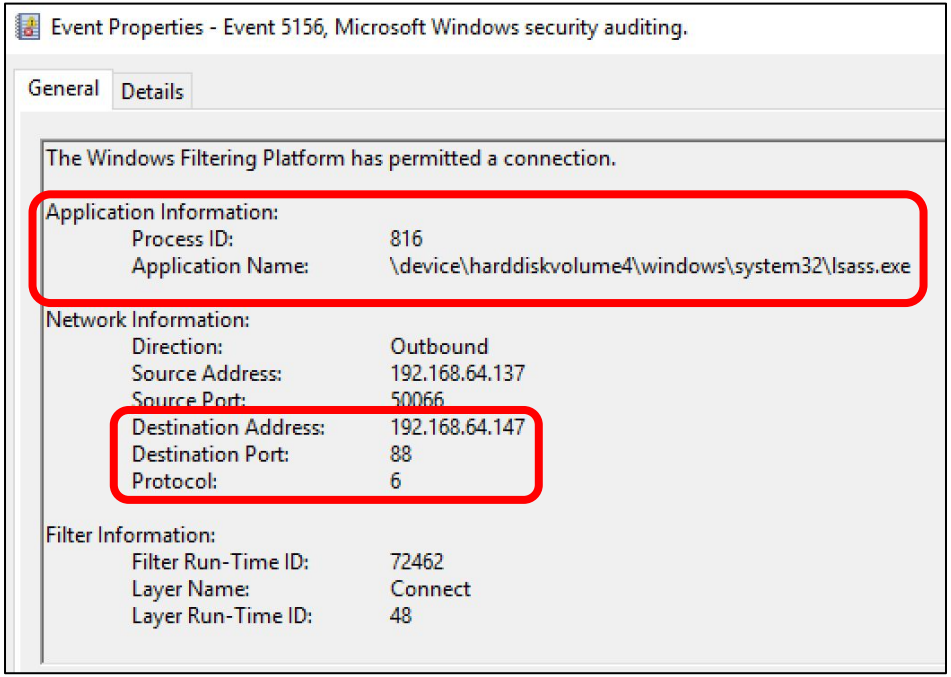
- "sacrificial" logon session that doesn't interact with the current logon session.
- **Opens the LSASS process**
- Patches associated logon session with hash/key.
- Normal Kerberos authentication process Kicks off (Hash->TGT)

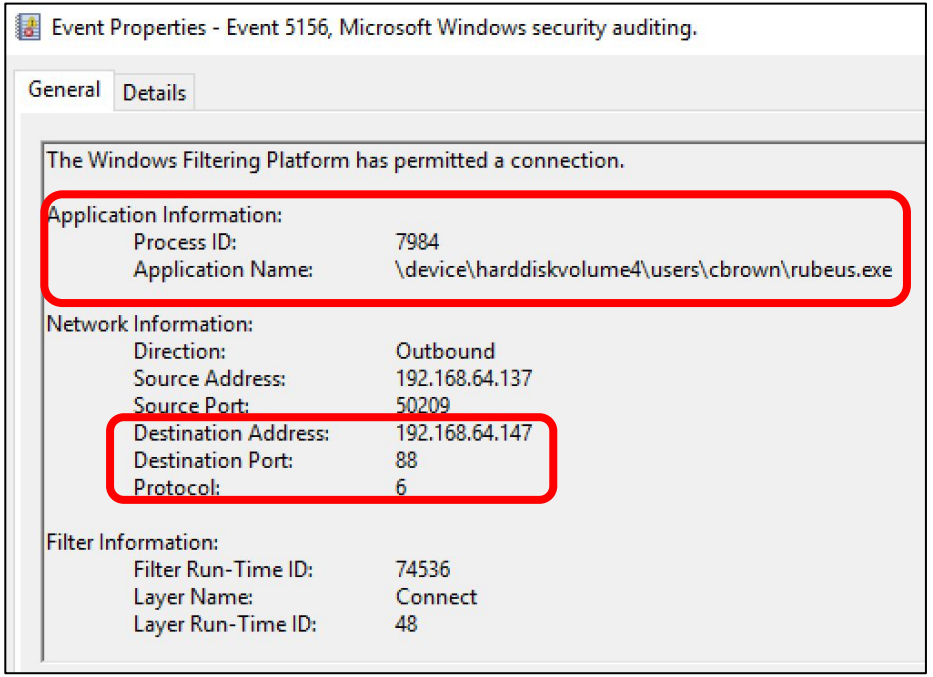
Rubeus asktgt (createnetonly)

- CreateProcessWithLogonW Function to create a sacrificial process/logon session
- Rubeus builds/crafts AS-REQ (w/ preauth)
- TGT request successful
- Imports TGT to logon session (Hash-TGT)

https://www.youtube.com/watch?v=QCDBjFJ_C3g

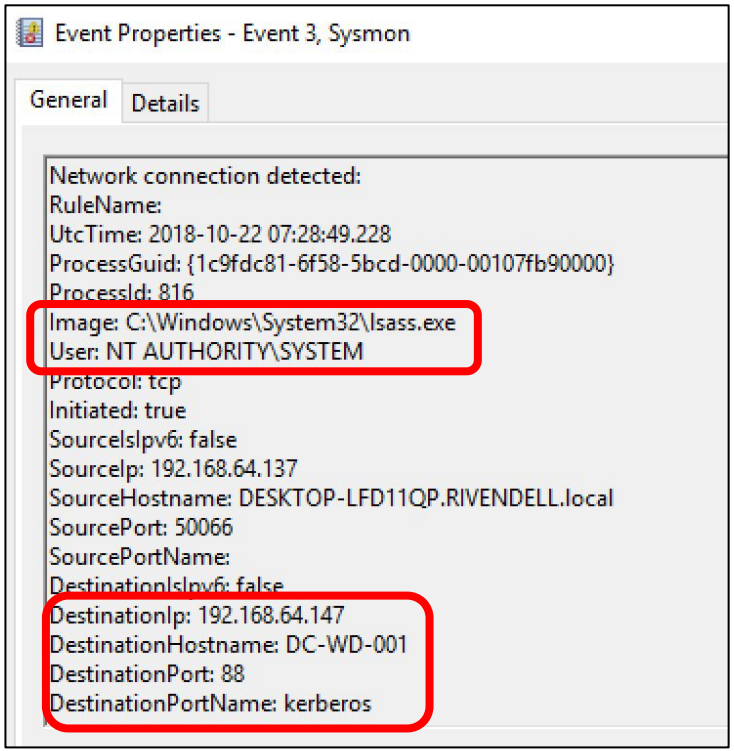
Mimikatz & Rubeus - Kerberos Authentication

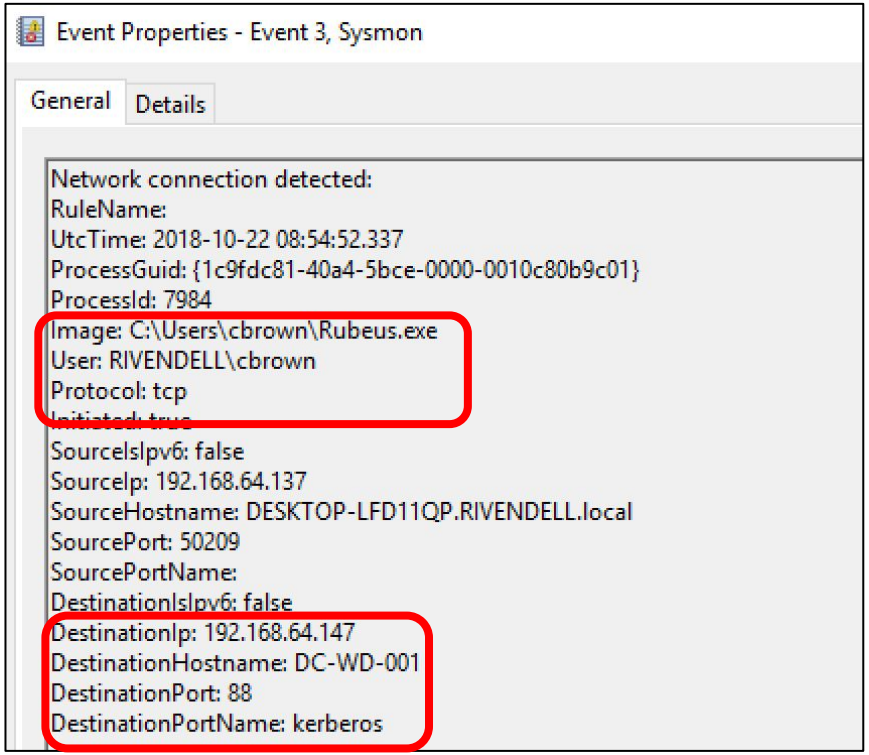




https://www.youtube.com/watch?v=QCDBjFJ_C3g

Mimikatz & Rubeus - Kerberos Authentication





https://www.youtube.com/watch?v=QCDBjFJ_C3g

Purple Teaming Challenges

Purple Team Challenges?

- What techniques do we prioritize?
- What is the scope? (Production?)
- What is the risk?
- How long is it going to take?
- Do we use our own Tradecraft?
- How do we handle communications?
- What are the deliverables?
- How do we track findings?
- Who is in charge?
- How do we split the team?
- Do we need to update our audit policies?
- Are we collecting data from every endpoint in scope?
- Do we even know what events we are collecting?
- Do we write a report together?
- Do we do do a tabletop exercise?
- What metrics can we use?
- Who else should be included?

Purple Team Challenges? (Strategy & Preparation)

- What techniques do we prioritize?
- What is the scope? (Production?)
- What is the risk?
- How long is it going to take?
- Do we use our own Tradecraft?
- How do we handle communications?
- What are the deliverables?
- How do we track findings?
- Who is in charge?
- How do we split the team?
- Do we need to update our audit policies?
- Are we collecting data from every endpoint in scope?
- Do we even know what events we are collecting?
- Do we write a report together?
- Do we do do a tabletop exercise?
- What metrics can we use?
- Who else should be included?

Purple Team Challenges? (Strategy & Preparation)

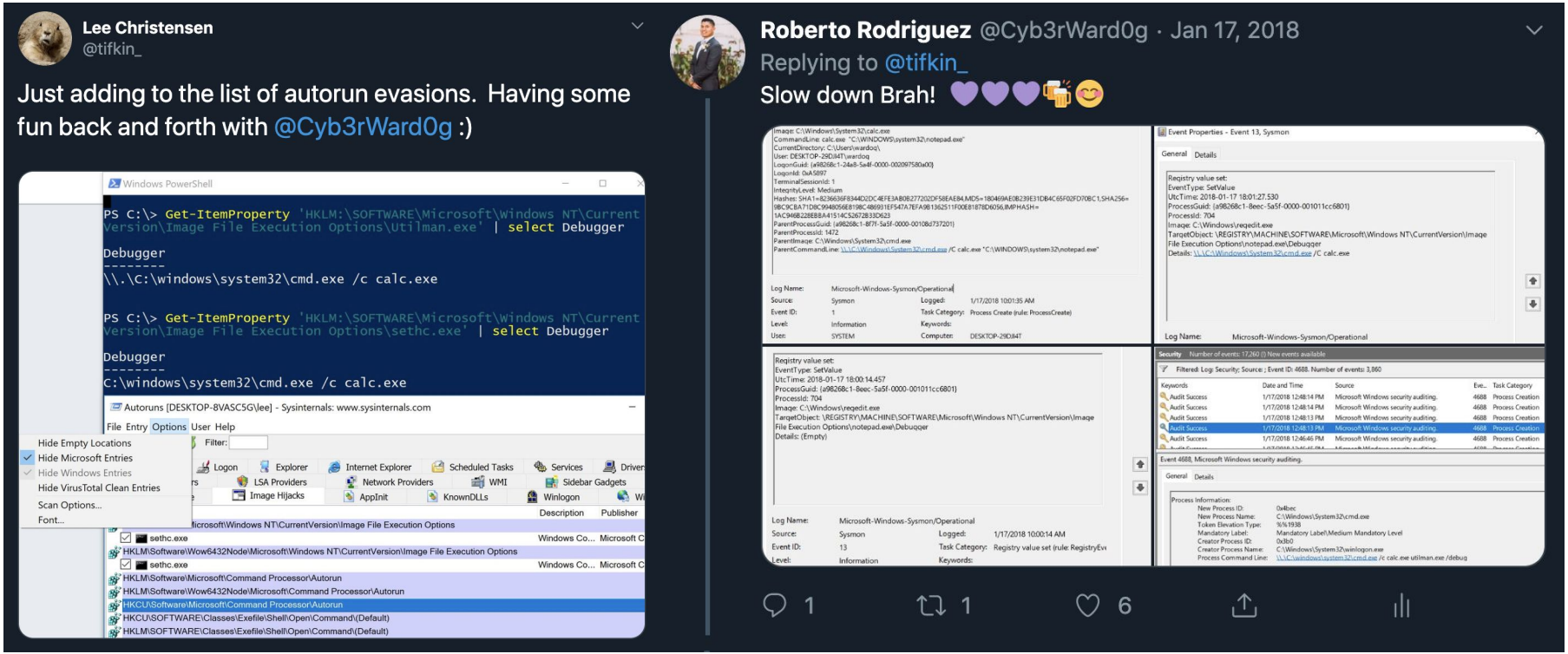
- **What techniques do we prioritize?**
- What is the scope? (Production?)
- What is the risk?
- How long is it going to take?
- **Do we use our own Tradecraft?**
- How do we handle communications?
- What are the deliverables?
- **How do we track findings?**
- Who is in charge?
- How do we split the team?

- **Do we need to update our audit policies?**
- Are we collecting data from every endpoint in scope?
- **Do we even know what events we are collecting?**
- Do we write a report together?
- Do we do do a tabletop exercise?
- What metrics can we use?
- Who else should be included?

Purple Teaming Challenges

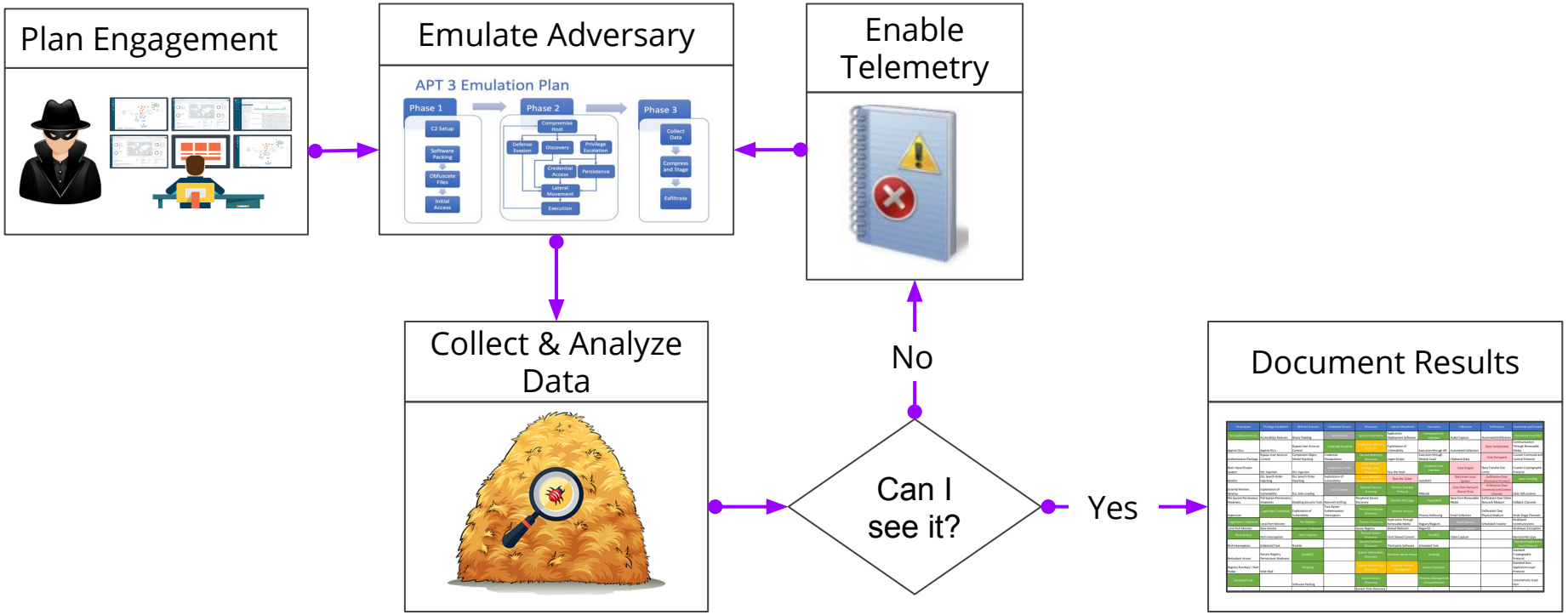
What about during the execution?

Purple Teaming: Ad-Hoc Execution! (I Love it!)

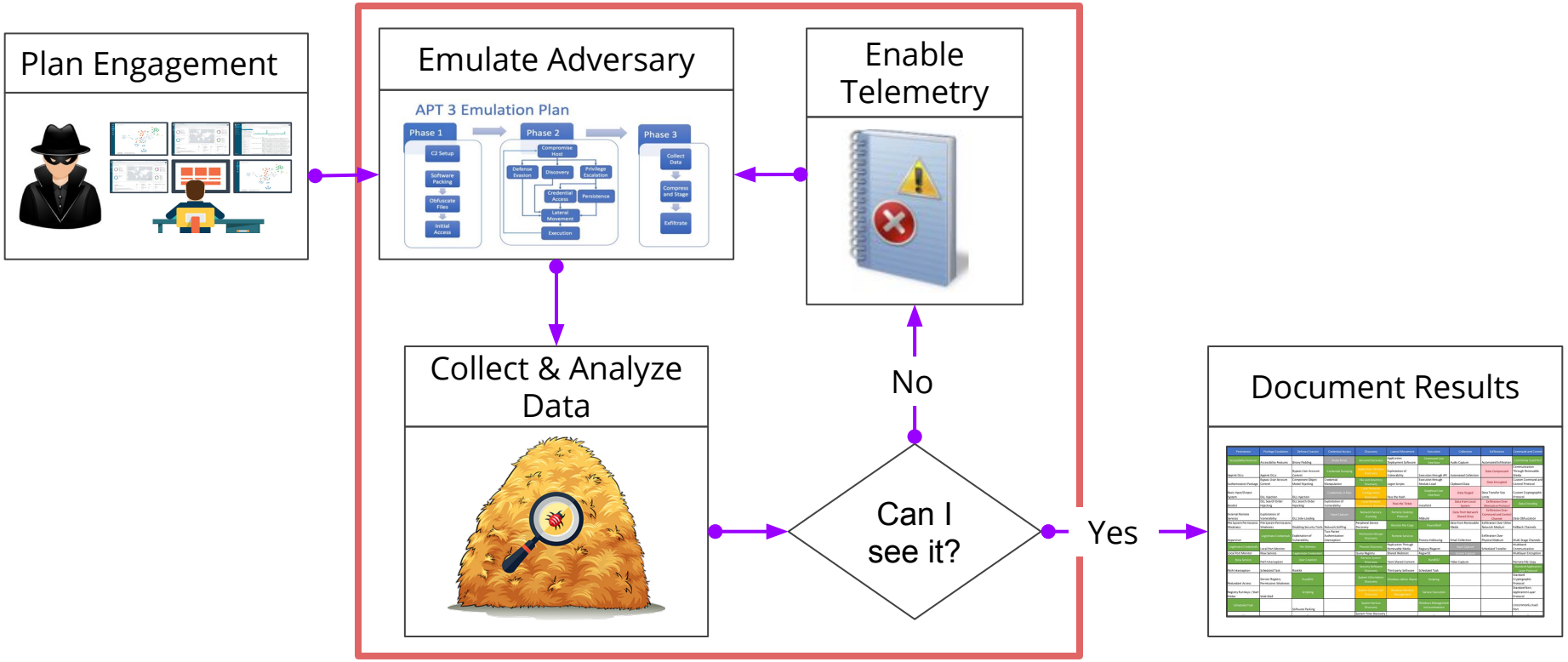


https://twitter.com/tifkin_/status/953688435482652674

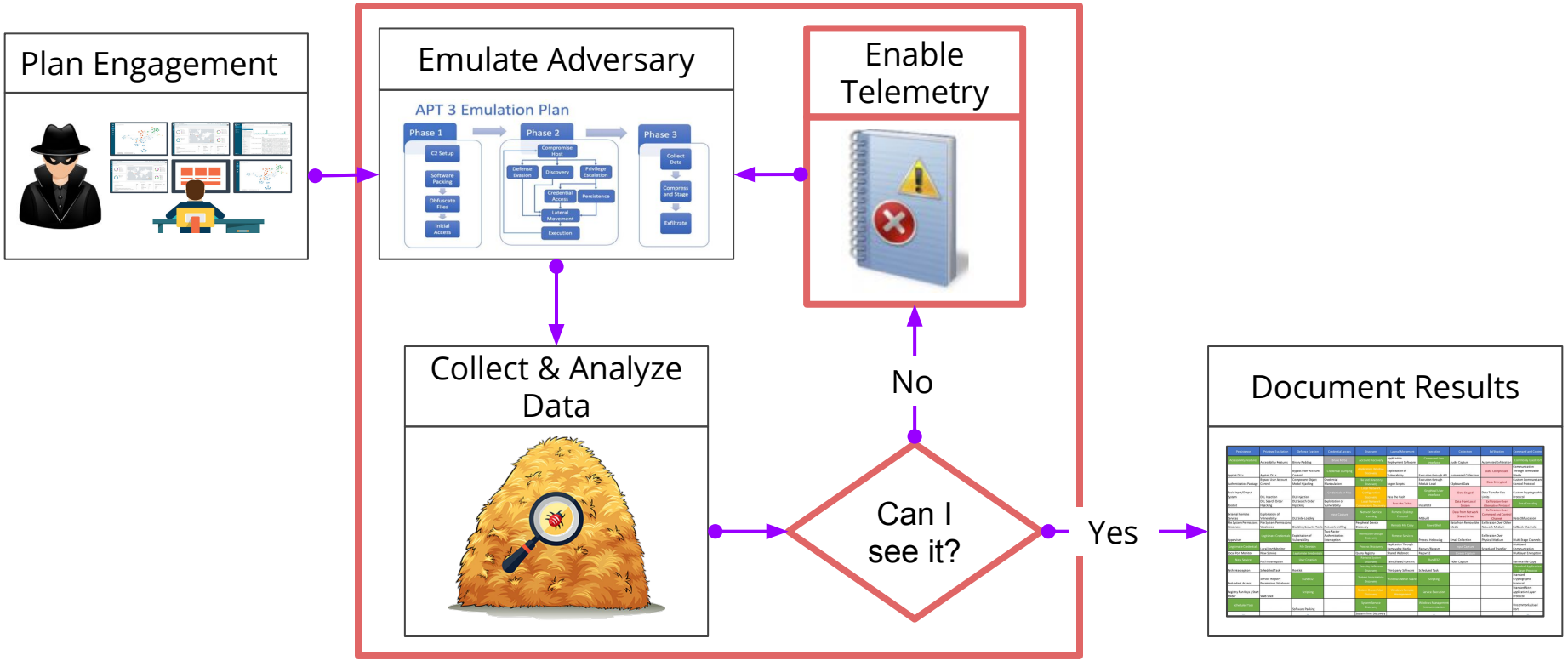
Purple Teaming: Basic Planned Execution



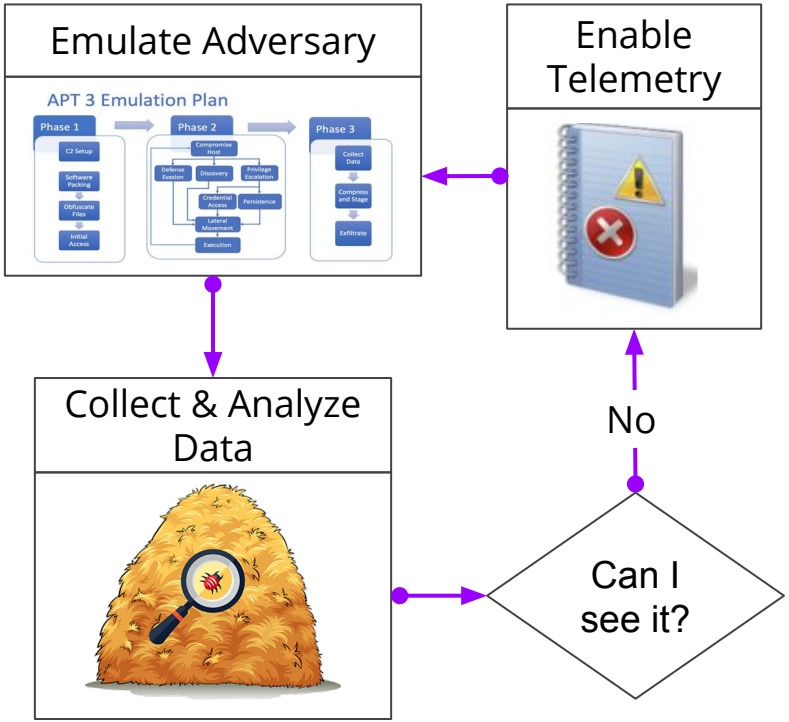
Purple Teaming: Basic Planned Execution



Purple Teaming: Basic Planned Execution

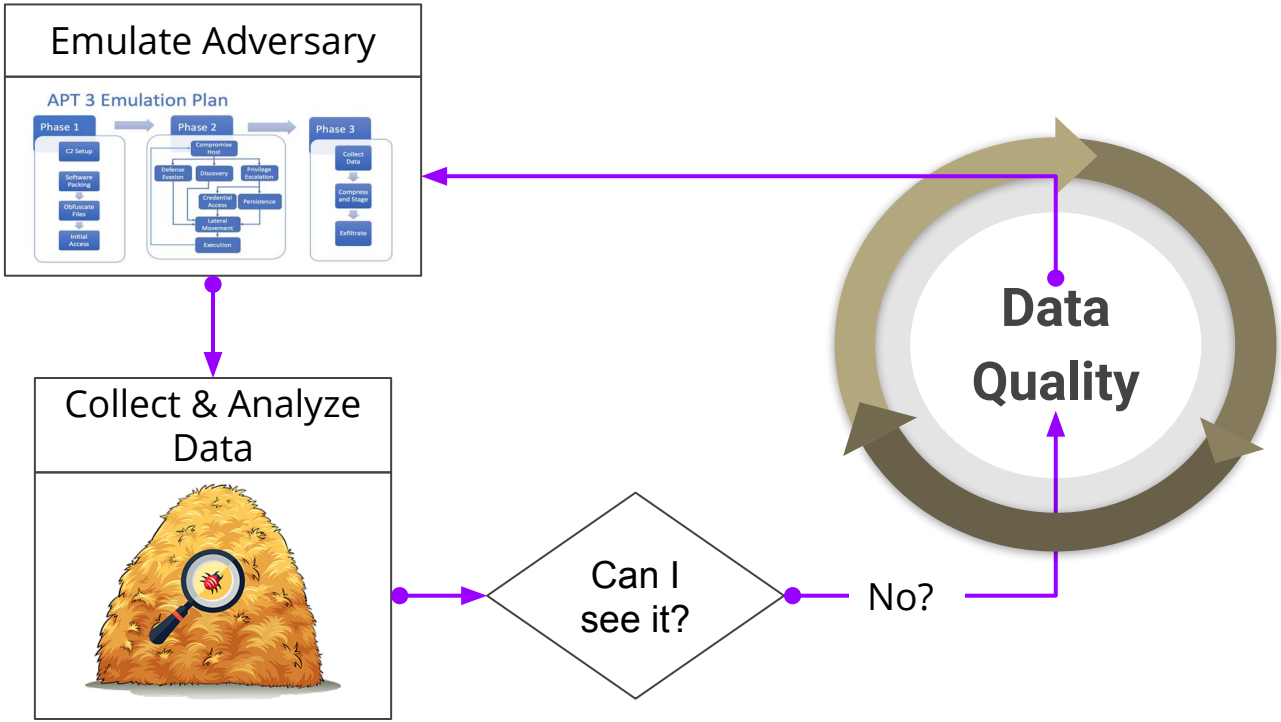


Planned Execution: Production Challenges

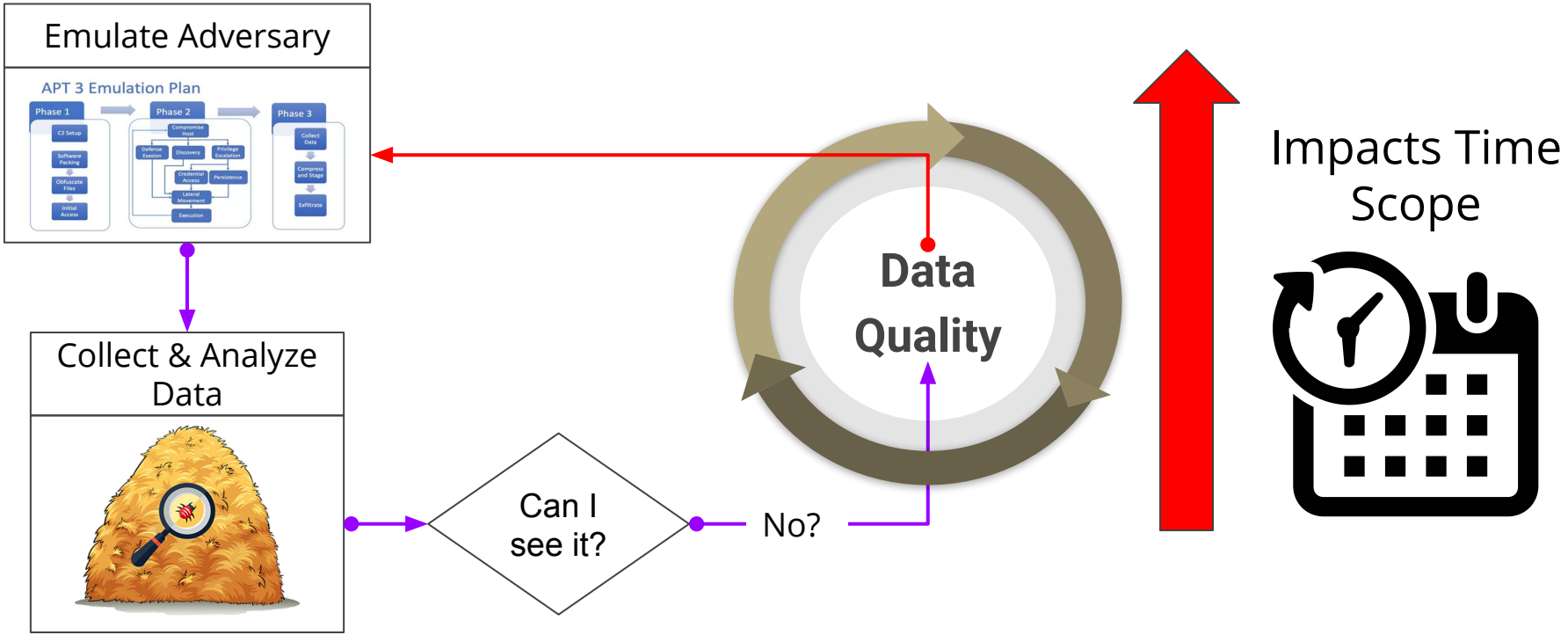


- How do you know what you need if you cant' even see it?
- How long would it take to approve an audit policy change?
- How do you know the log volume that you are about to enable?
- How do you know it is not your data pipeline failing?
- What if your event logs are not parsed or standardized?

Planned Execution: Production Challenges



Planned Execution: Production Challenges



Planned Execution: Production Challenges

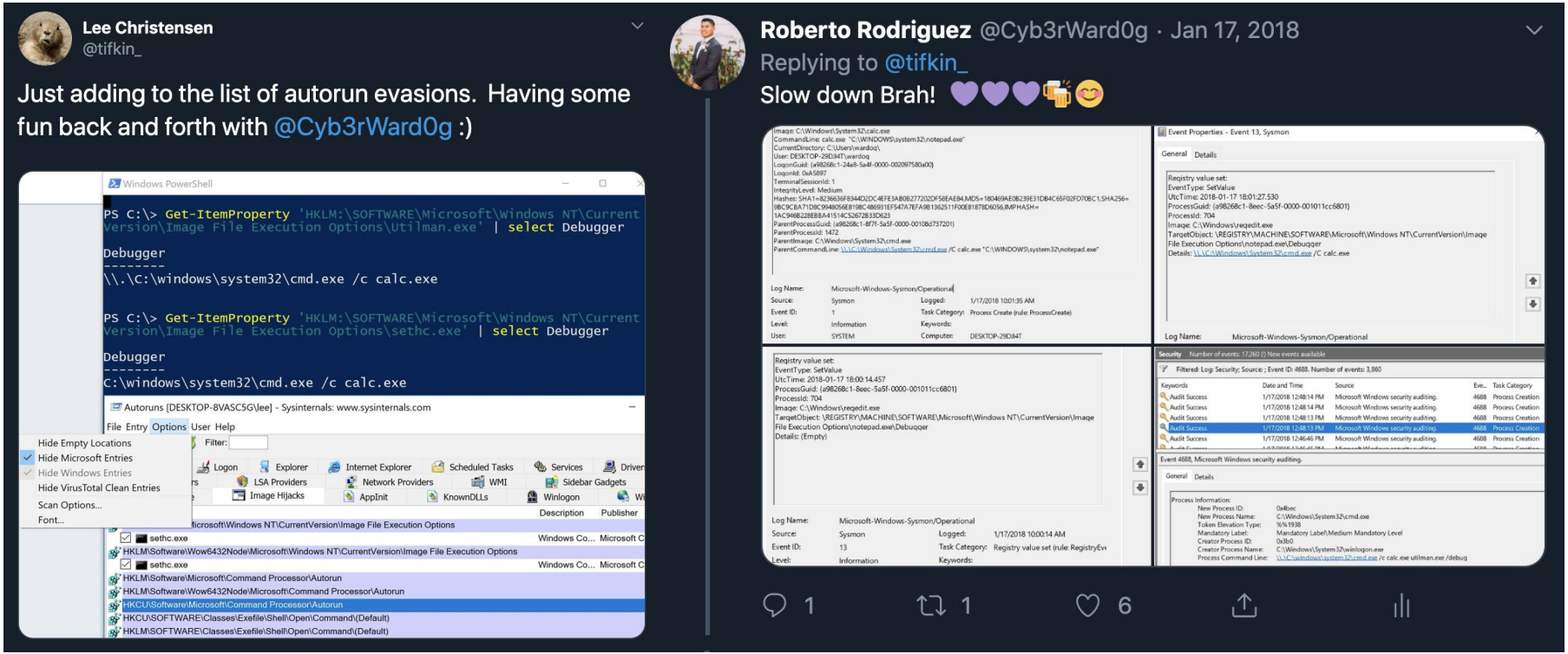
Time Spent

Time Spent

We all have busy schedules

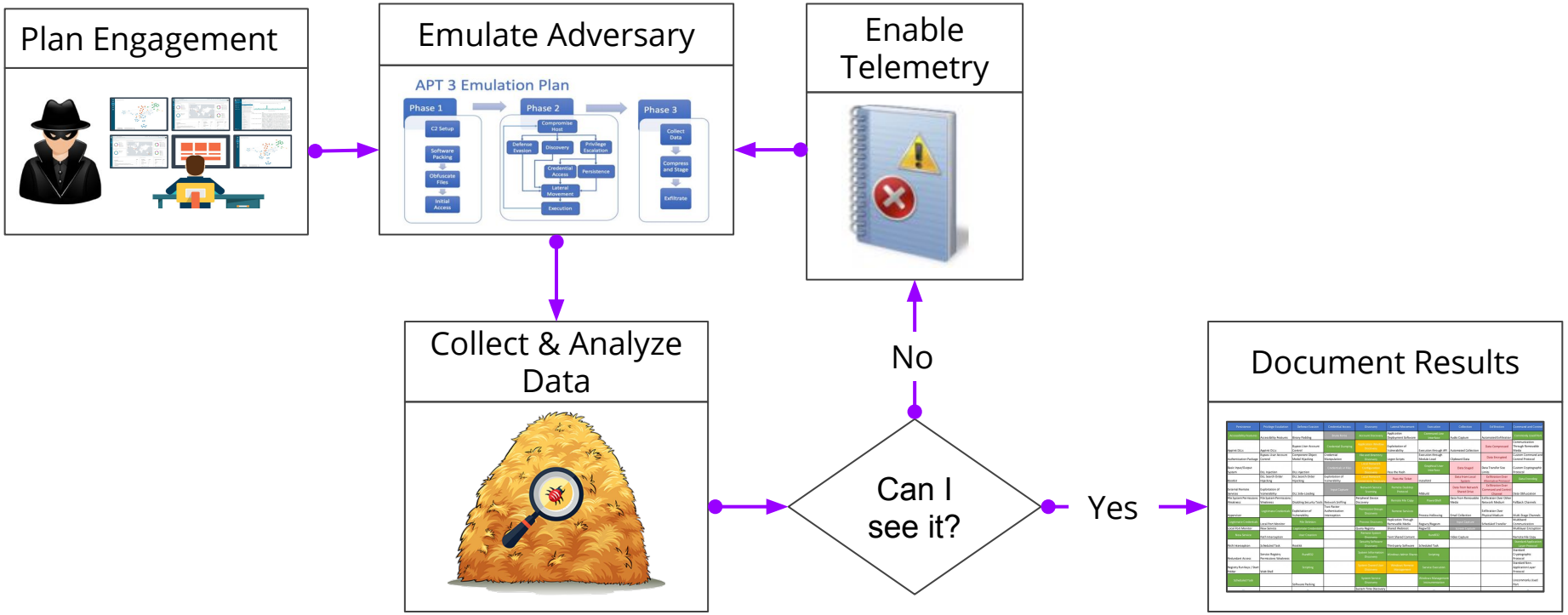
https://dribbble.com/GraphicChendol

Purple Teaming: Ad-Hoc Execution! (I Love it!)



https://twitter.com/tifkin_/status/953688435482652674

Planned Execution: A Lab Before Production



We can start improving this!

Time Spent

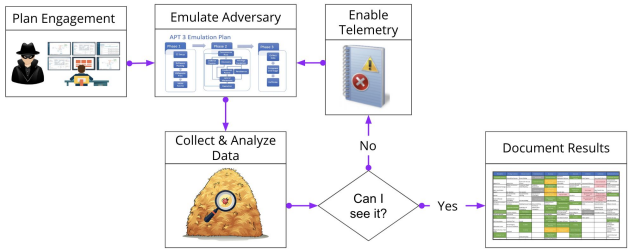
Time Spent

Purple Teaming Challenges

Can you run this technique variation one more time?

Execute -> Collect -> Analyze -> Repeat

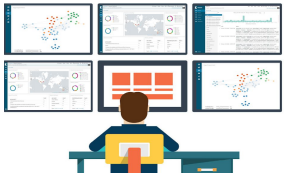
Simulating Adversarial Technique



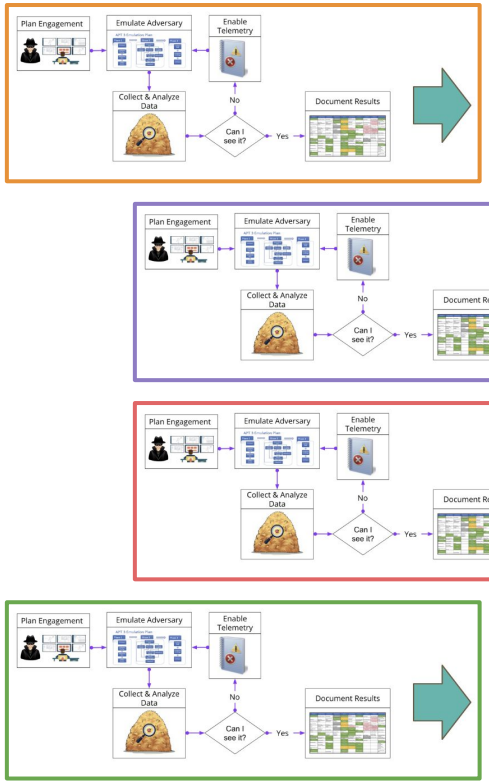
Data produced

Test Security Controls

Model Adversary Behavior



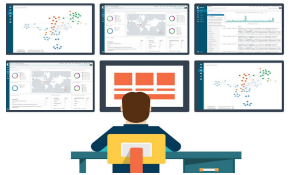
Same Technique + Some Variations



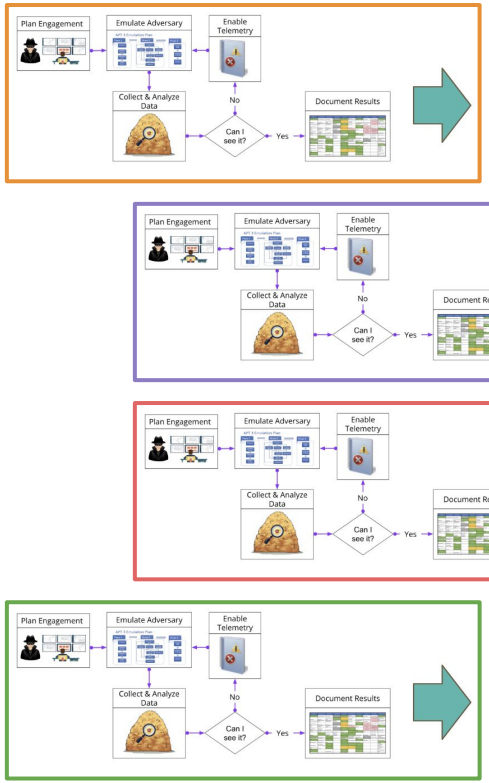
Data produced

Test Security Controls

Model Adversary Behavior



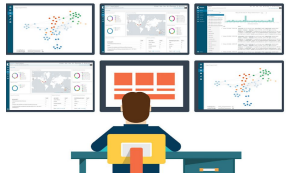
Same Technique + Some Variations



Data produced

Test Security Controls

Model Adversary Behavior

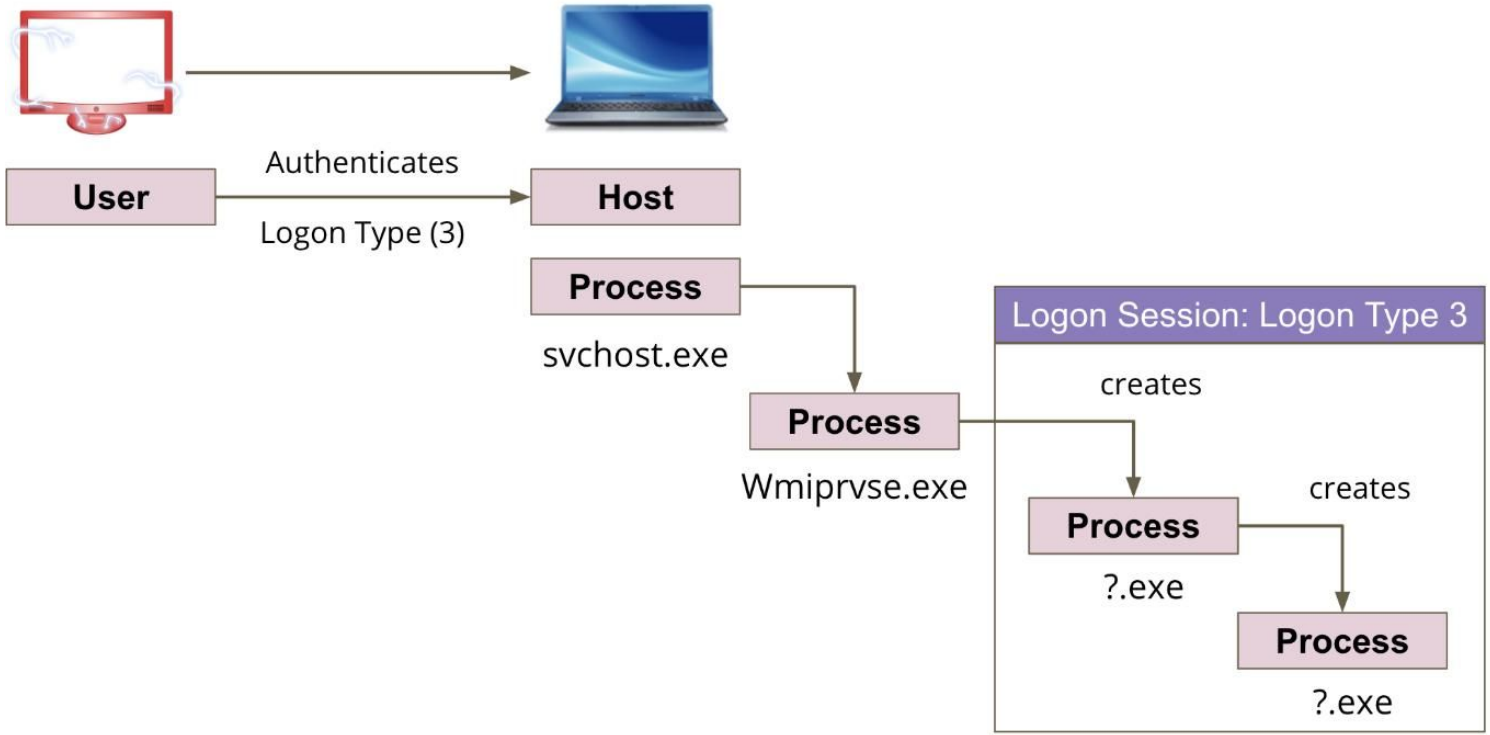


Same Data?
Similar Events?

Lateral Movement via WMI (Win32_Process Create)

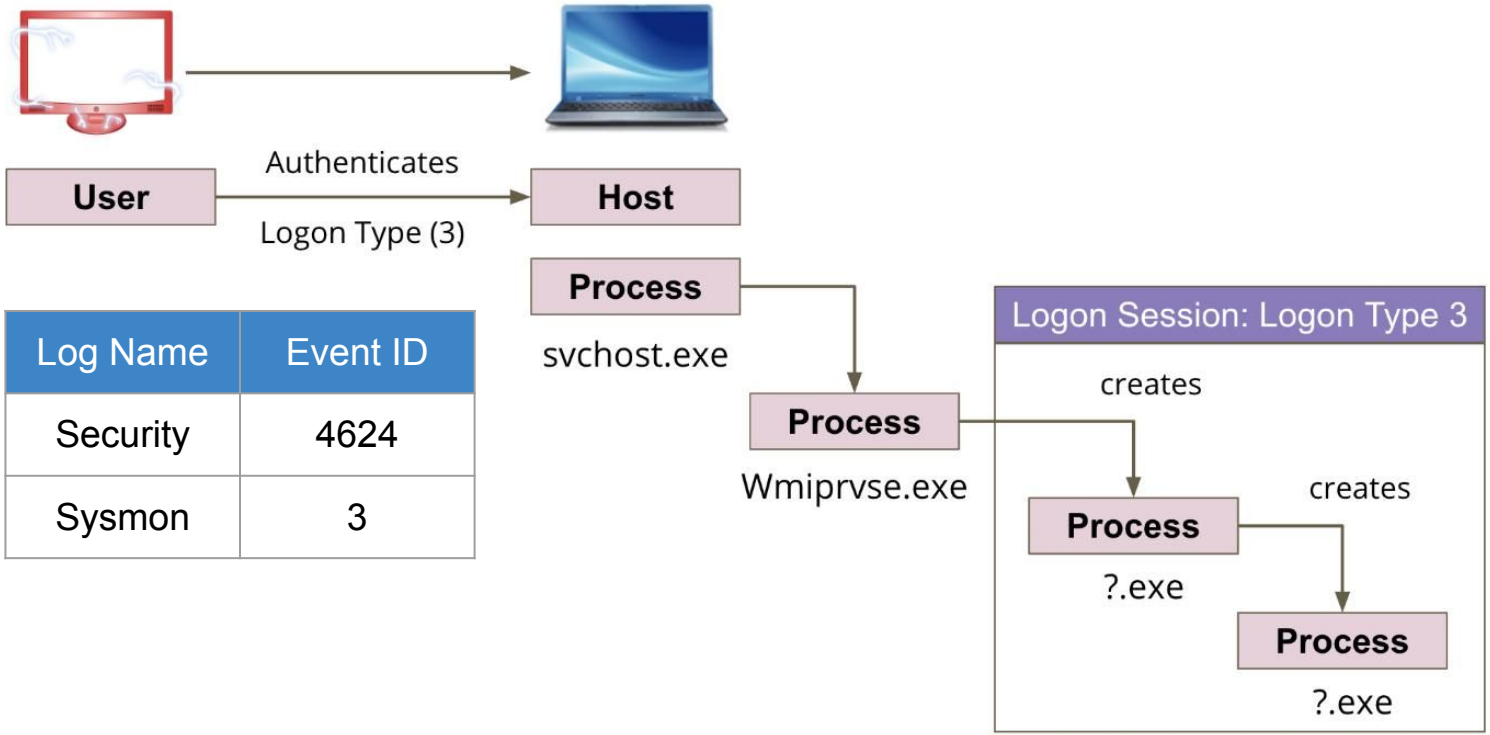
- **wmic** /node:172.18.39.106 /user:Administrator /password:P1ls3n! process call create cmd.exe
- **Invoke-WmiMethod** -ComputerName 172.18.39.106 -Credential Administrator -Class Win32_Process -Name Create -ArgumentList notepad.exe
- **SharpWMI.exe** action=create computername=HR01.shire.com command="powershell.exe -enc ZQBj..."
- **./wmiexec** shire.com/pgustavo@172.18.39.106

Lateral Movement via WMI - Behavior



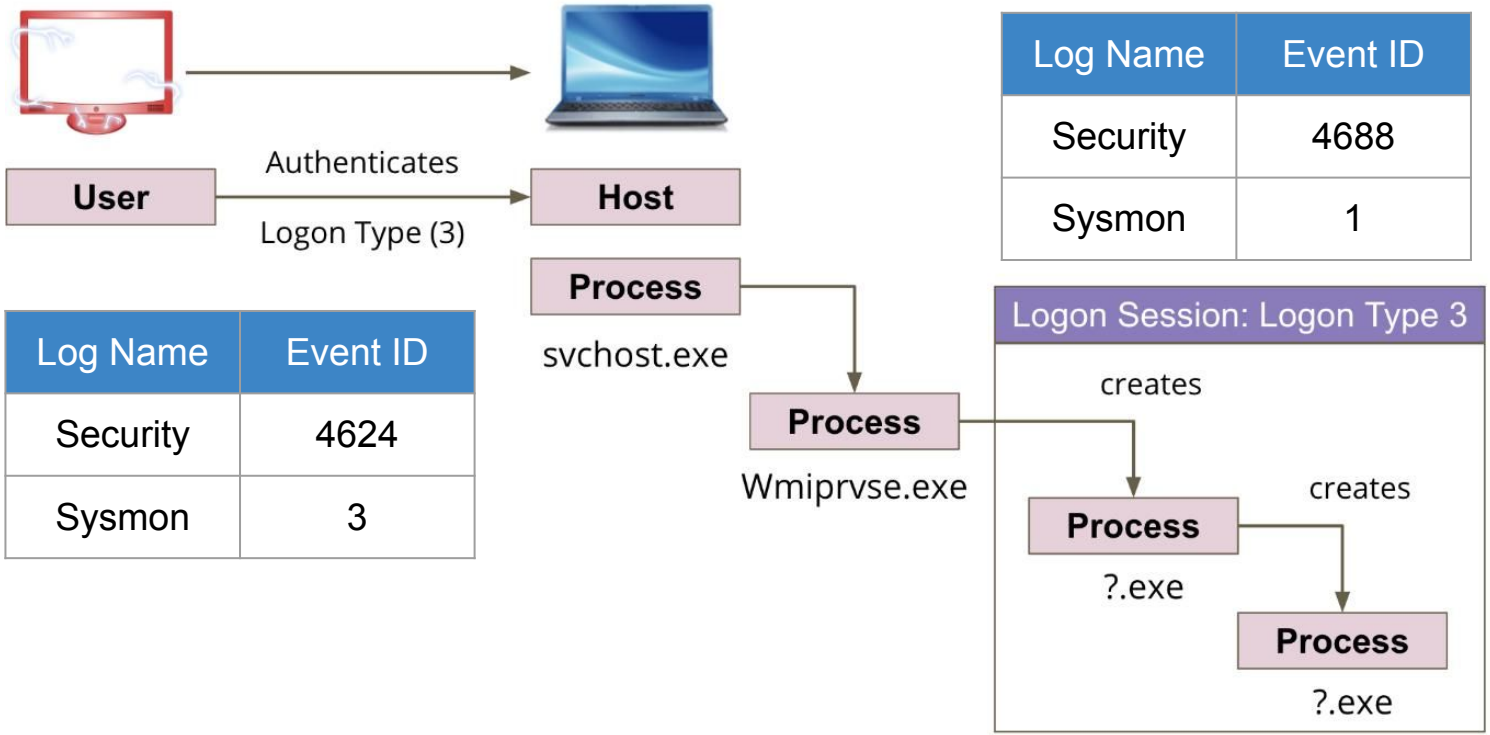
https://twitter.com/HunterPlaybook/status/1171103081246969856

Lateral Movement via WMI - Behavior



Log Name	Event ID
Security	4624
Sysmon	3

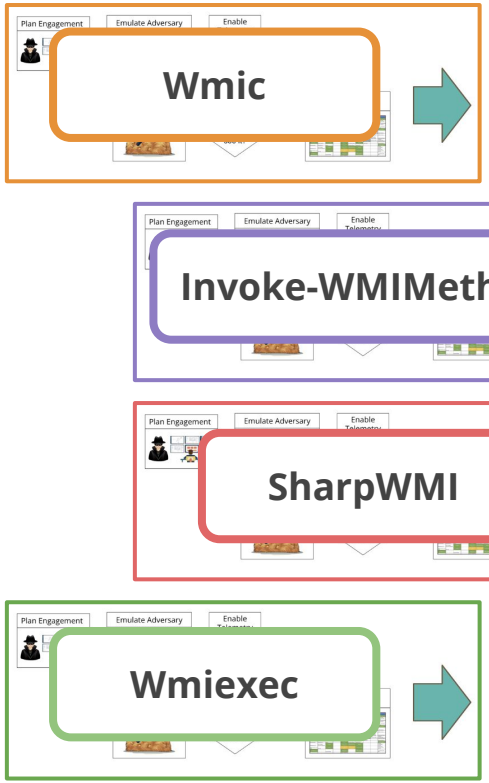
Lateral Movement via WMI - Behavior



Log Name	Event ID
Security	4624
Sysmon	3

Log Name	Event ID
Security	4688
Sysmon	1

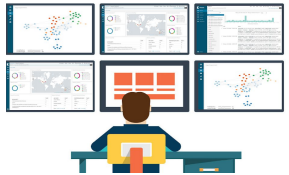
Same Technique + Some Variations



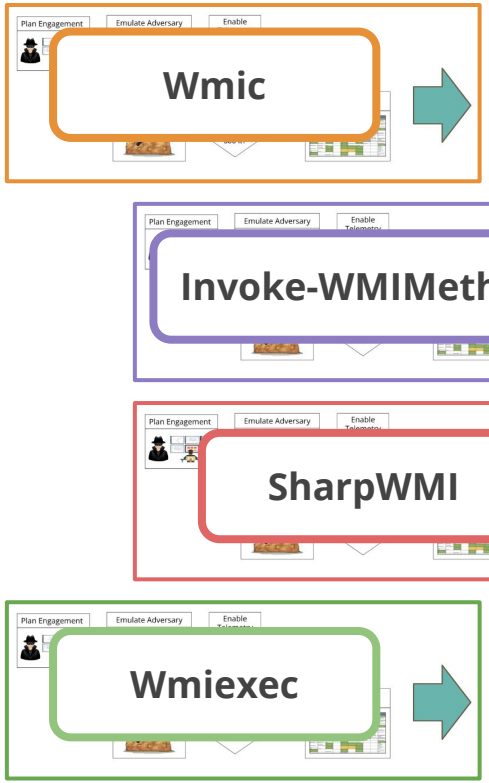
Data produced

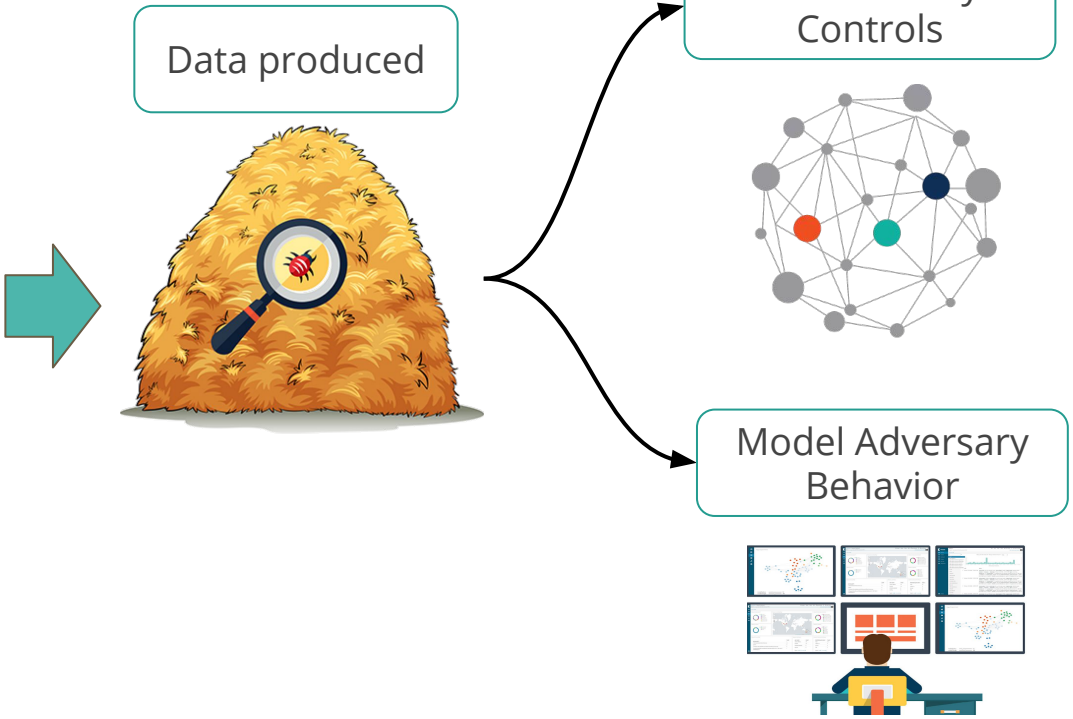
Test Security Controls

Model Adversary Behavior



Can you run it again? Please!





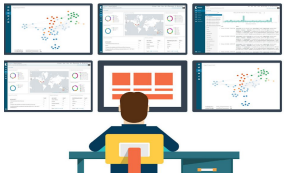
But… We are getting similar events…

Data produced

Log Name	Event ID
Security	4624
Sysmon	3
Security	4688
Sysmon	1

Test Security Controls

Model Adversary Behavior

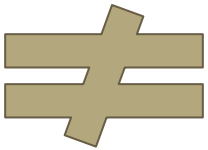


Are we being effective at purple teaming?

Effective or Efficient Purple Teaming?

Efficiency

Efficacy



Effectiveness

Efficiency

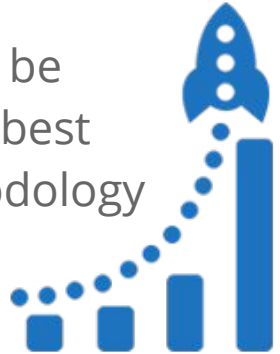
The way resources are used (or wasted), How can I make the most of the resources I have

Efficacy

It doesn't matter how we do it, but only on what we accomplish

Effectiveness

Accomplishes the goals (to be efficacious) employing the best and most economic methodology (to be efficient).



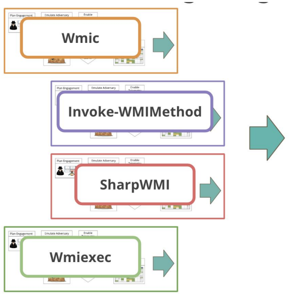
Efficiency

- Let's prioritize specific techniques
- Can we automate adversary emulation plans?
- How do we replicate results?
- How do we expedite unit testing?
- How do I reduce the number of times a technique is executed for research?

Efficacy

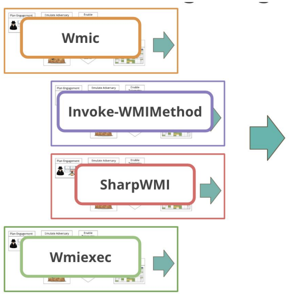
- Let's execute 5 techniques a week !
- At least 3 variations per technique!
- Let's emulate all APT groups!
- Let's test current security analytics!

We might be doing this over and over..



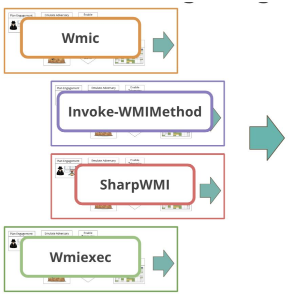
Data produced

Log Name	Event ID
Security	4624
Sysmon	3
Security	4688
Sysmon	1



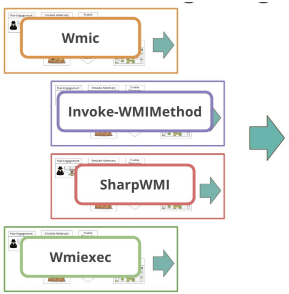
Data produced

Log Name	Event ID
Security	4624
Sysmon	3
Security	4688
Sysmon	1



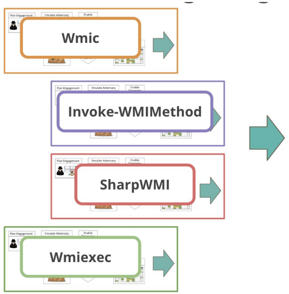
Data produced

Log Name	Event ID
Security	4624
Sysmon	3
Security	4688
Sysmon	1



Data produced

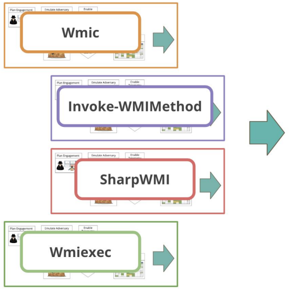
Log Name	Event ID
Security	4624
Sysmon	3
Security	4688
Sysmon	1



Data produced

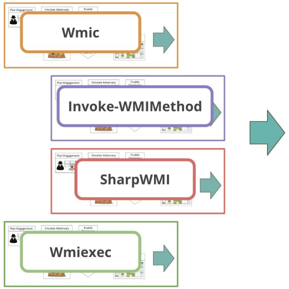
Log Name	Event ID
Security	4624
Sysmon	3
Security	4688
Sysmon	1

And we still get similar events..



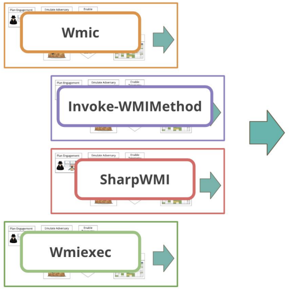
Data produced

Log Name	Event ID
Security	4624
Sysmon	3
Security	4688
Sysmon	1



Data produced

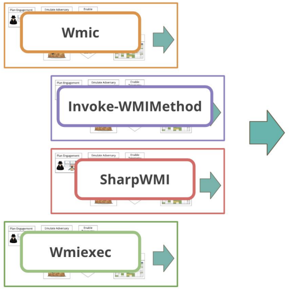
Log Name	Event ID
Security	4624
Sysmon	3
Security	4688
Sysmon	1



Data produced

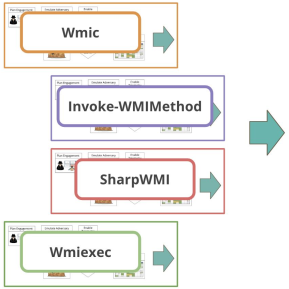
Log Name	Event ID
Security	4624
Sysmon	3
Security	4688
Sysmon	1

Log Name	Event ID
Security	4624
Sysmon	3
Security	4688
Sysmon	1



Data produced

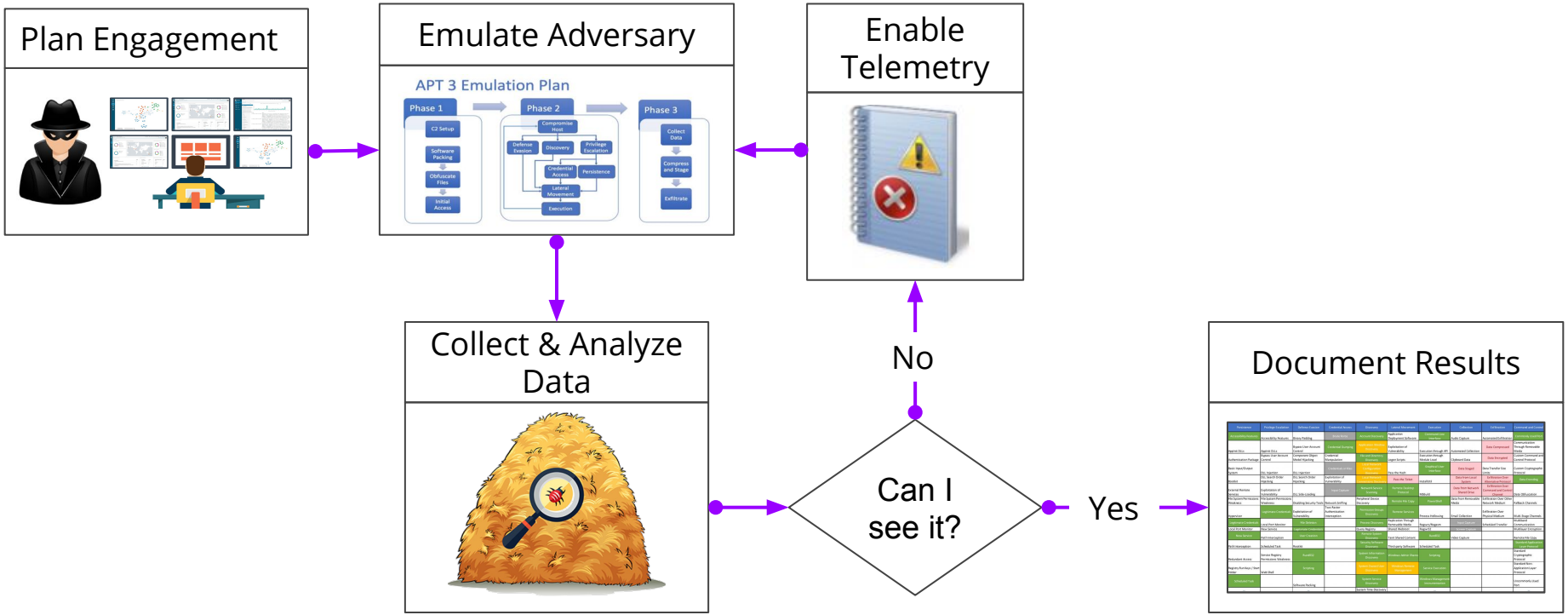
Log Name	Event ID
Security	4624
Sysmon	3
Security	4688
Sysmon	1



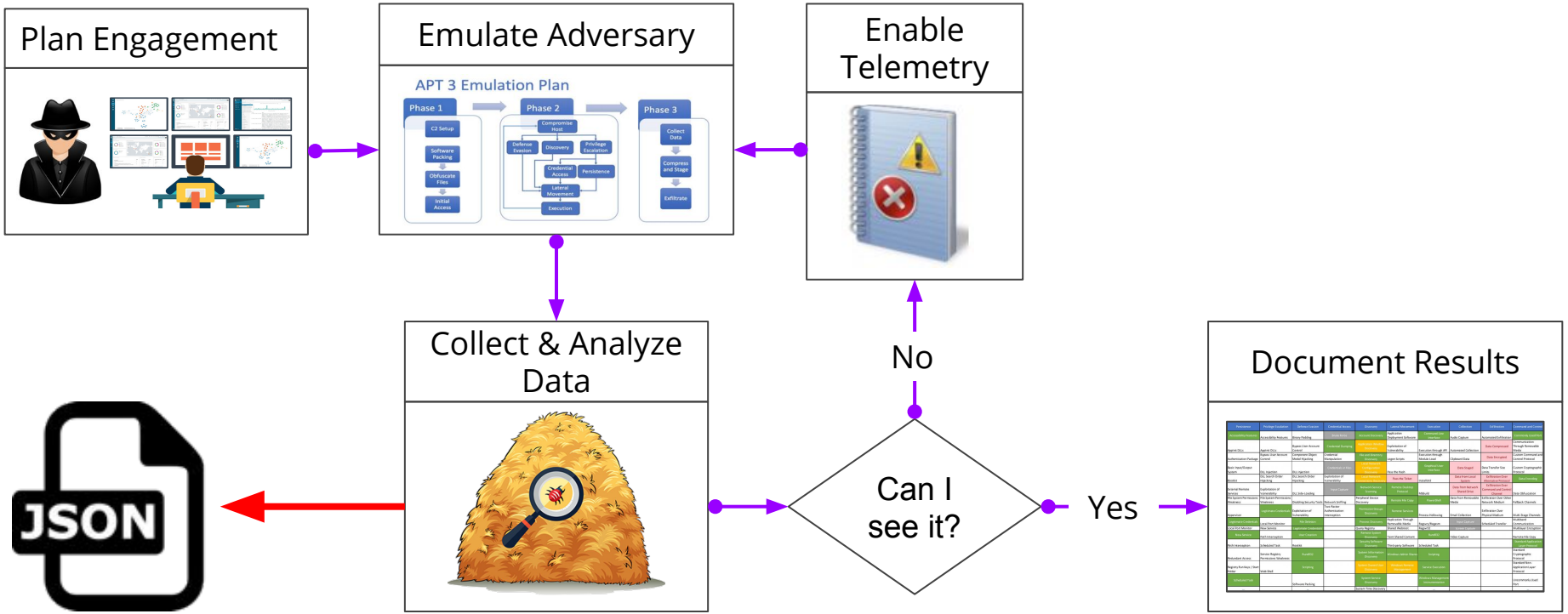
Data produced

Log Name	Event ID
Security	4624
Sysmon	3
Security	4688
Sysmon	1

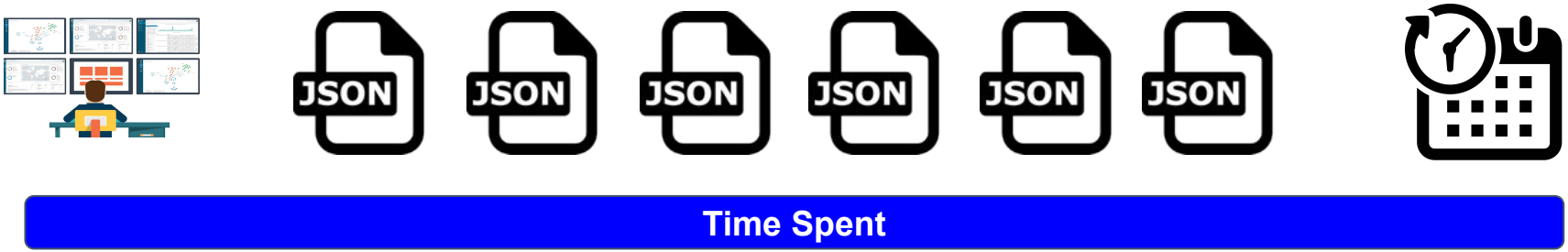
What if we could save our output?



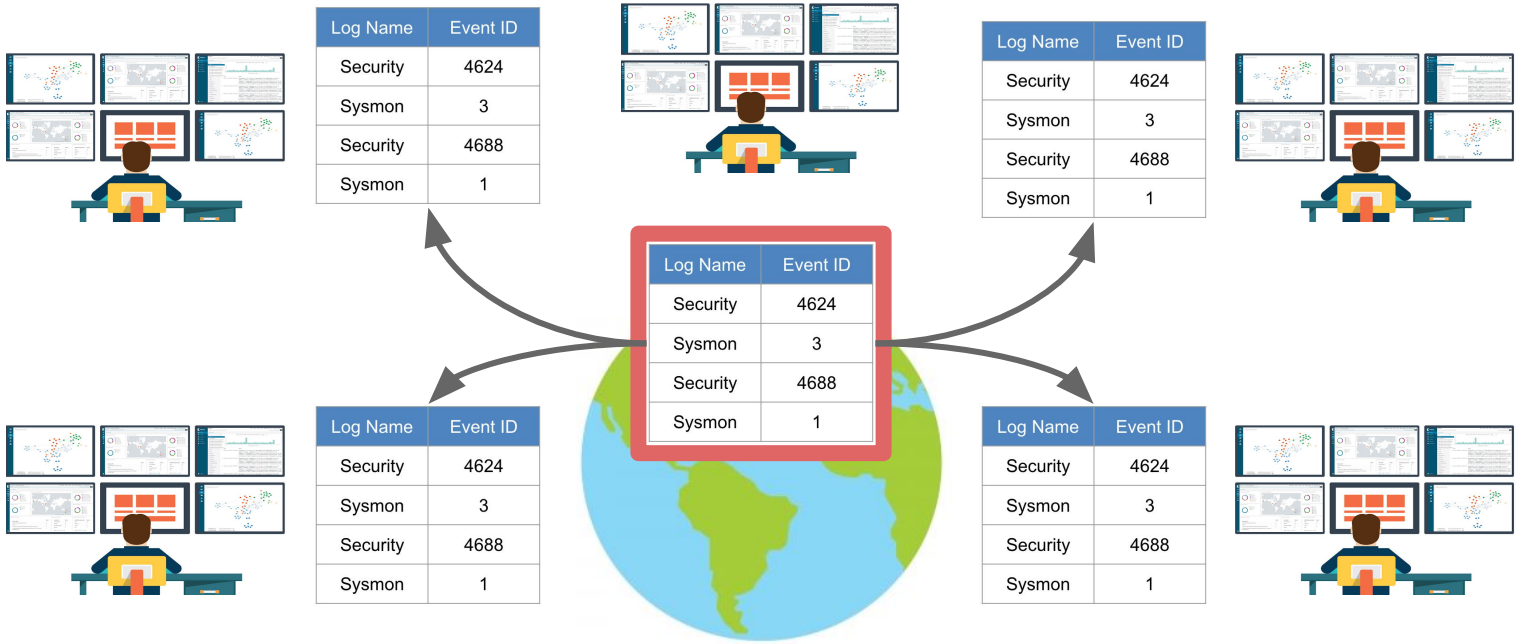
What if we could save our output?



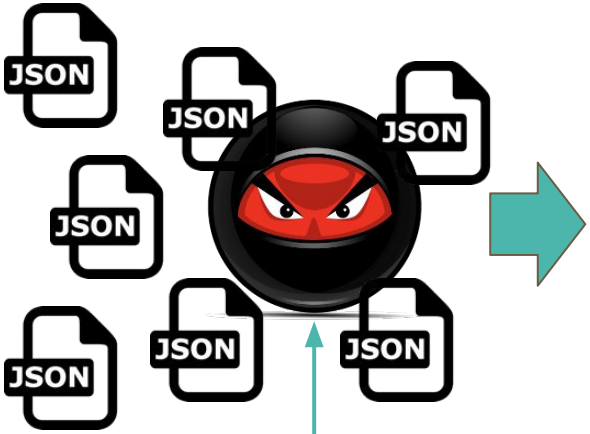
We could keep the momentum!



We could also train future purple teamers!

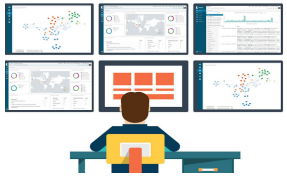


We could expedite analytic validation - Unit Testing



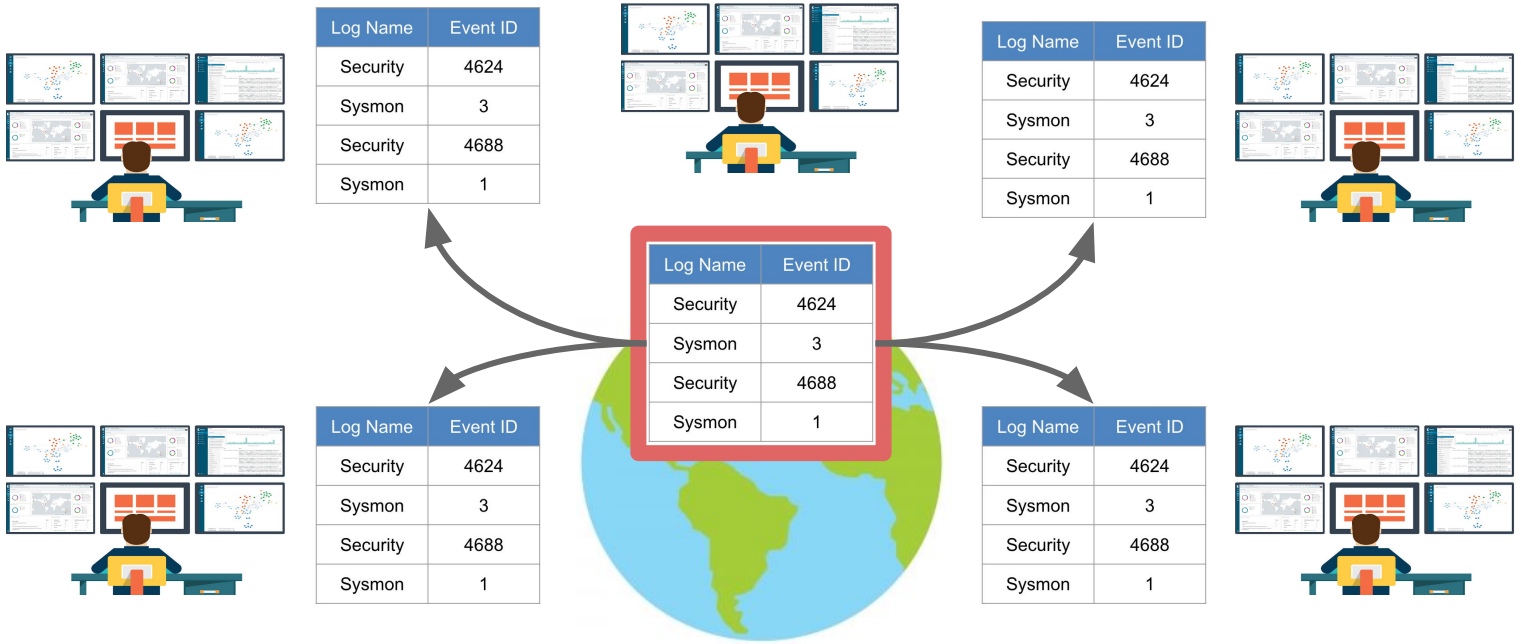
Mordor File

Validate Analytics



2 + 2 = 4

We could also help the community!



Enter Mordor

Mordor Project @Mordor_Project

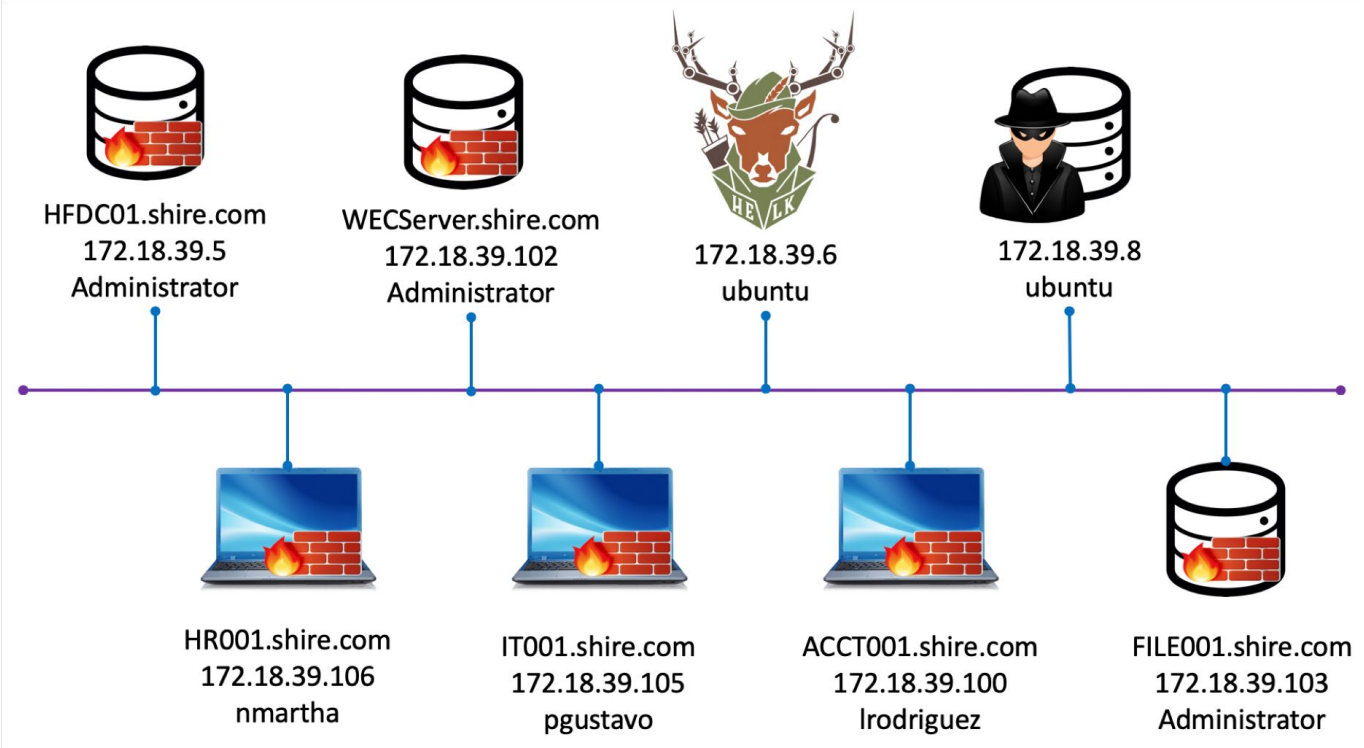
- Pre-recorded security events generated by simulated adversarial techniques in the form of JavaScript Object Notation (JSON)
- Pre-recorded data categorized by platforms, adversary groups, tactics and techniques defined by the Mitre ATT&CK Framework.
- Data represents not only specific known malicious events but additional context/events that occur around it.

Mordor Standard Environments

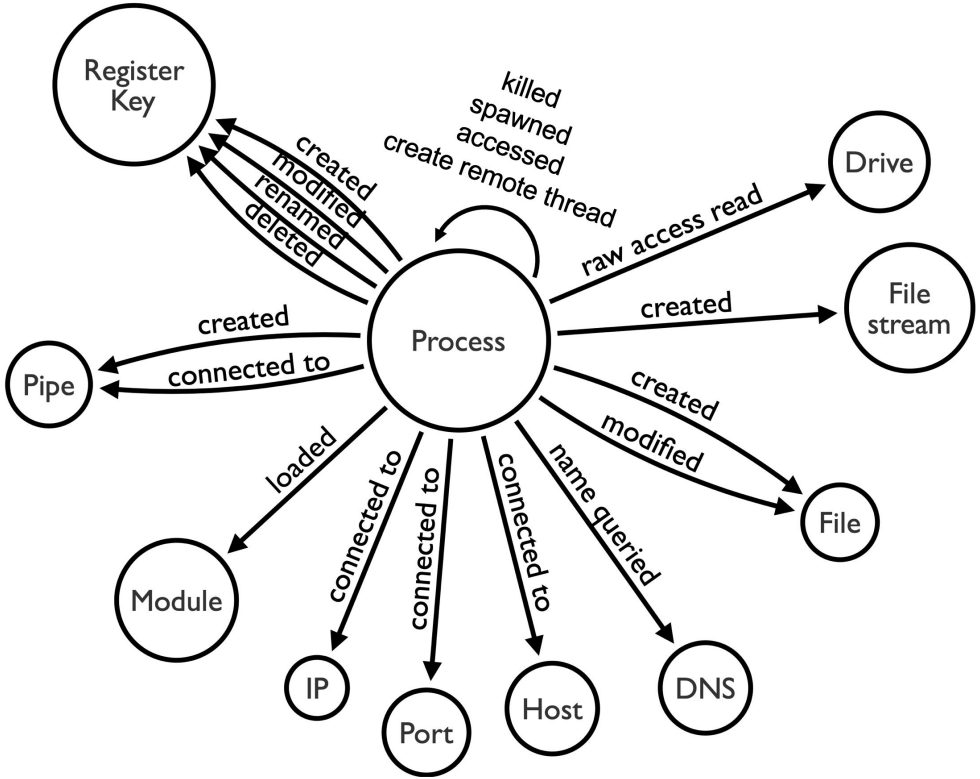
- Environment designed to replicate a small research network
- Standardized and documented setup
- Platforms
  - Windows
  - Linux
- Endpoints Telemetry
  - Windows Security Auditing
  - Event Tracing for Windows (ETW) (NEW!!)
- Network Telemetry
  - Network Logs
- Environments Available: Shire and Erebor deployed by **BlackSmith**!

Mordor Environments: The Shire

The Shire Design

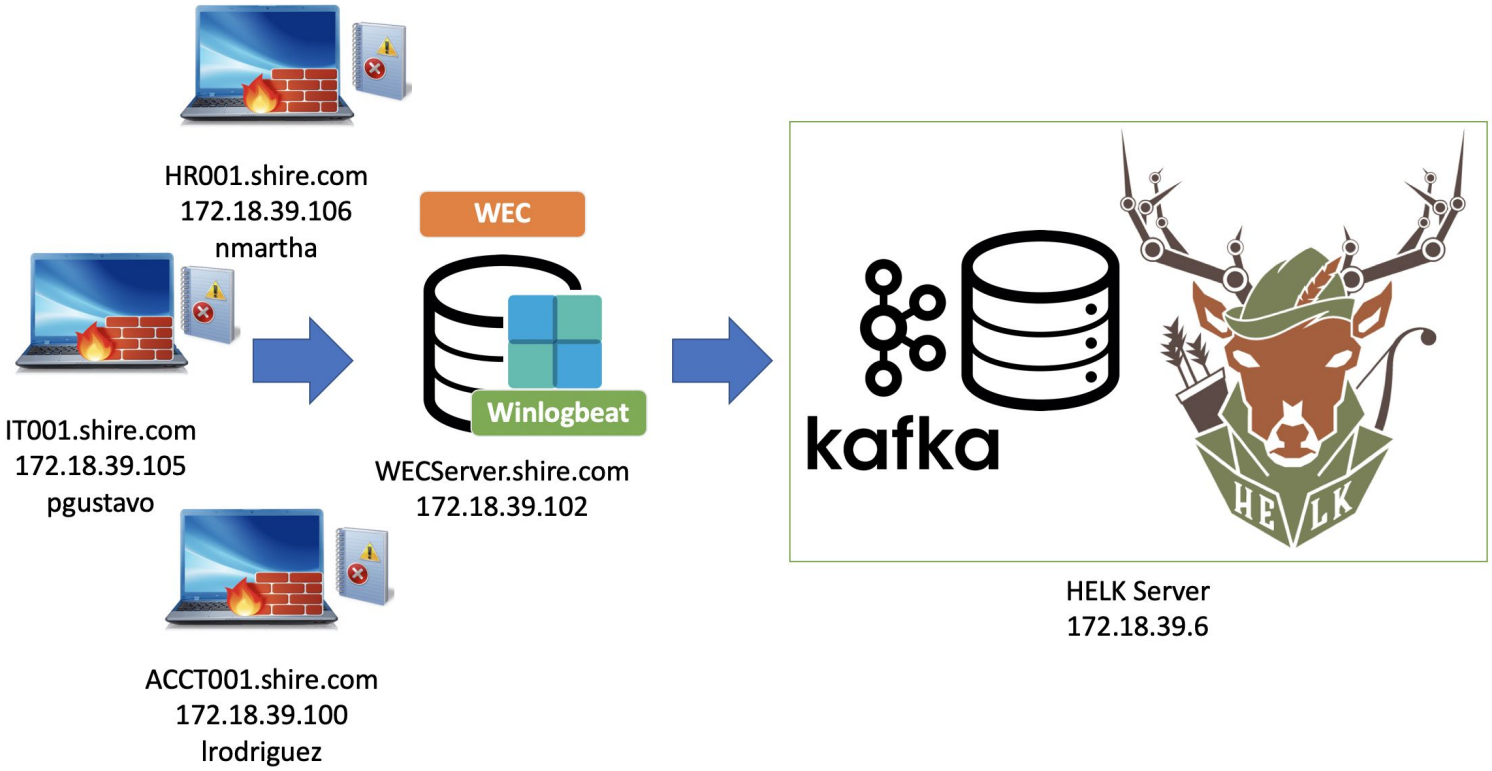


The Shire Telemetry: Win Logs & Sysmon



https://github.com/Cyb3rWard0g/OSSEM/tree/master/data_dictionaries/windows/sysmon

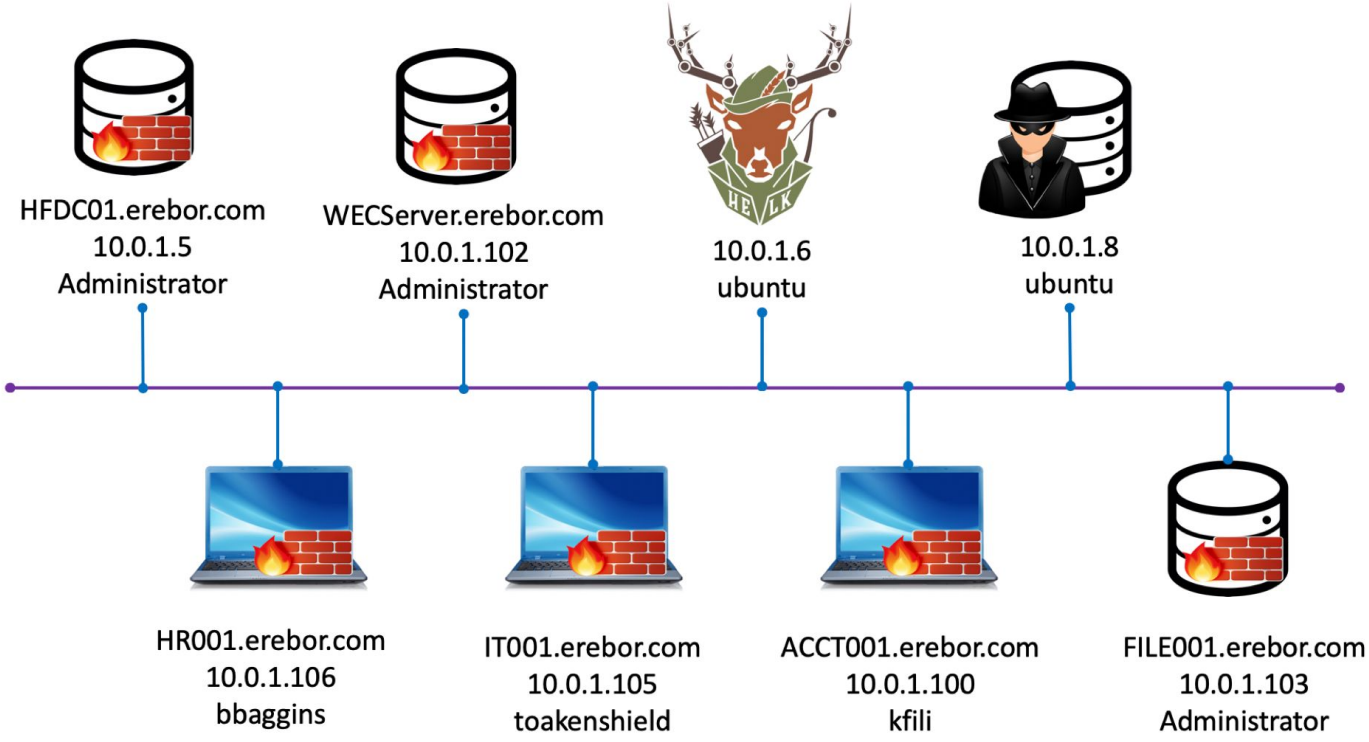
The Shire: Event Log -> WEC -> HELK



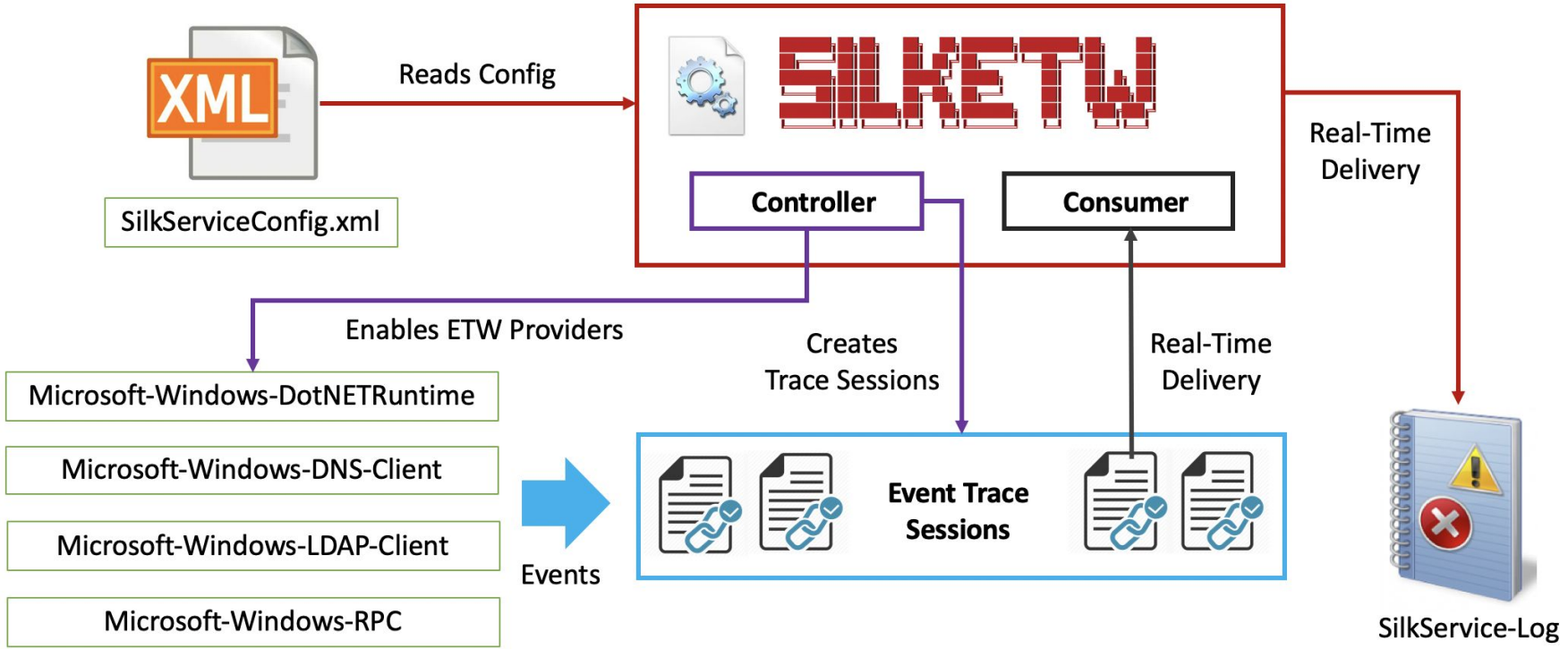
https://mordor.readthedocs.io/en/latest/mordor_shire.html#

Mordor Environments: Erebor (Lonely Mountain)

Erebor Design

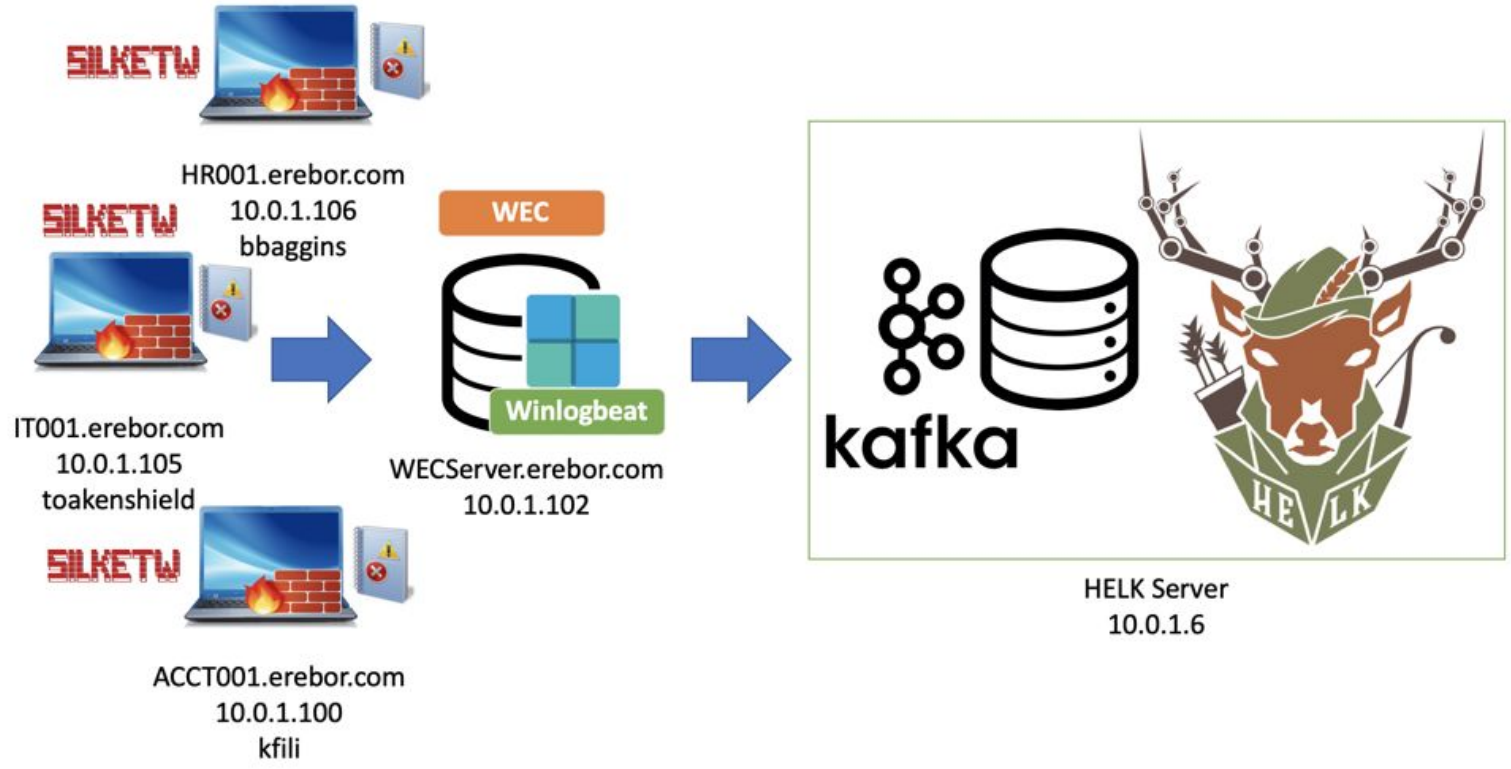


Erebor Telemetry: ETW Events via SilkETW



https://medium.com/threat-hunters-forge/threat-hunting-with-etw-events-and-helk-part-1-installing-silketw-6eb74815e4a0

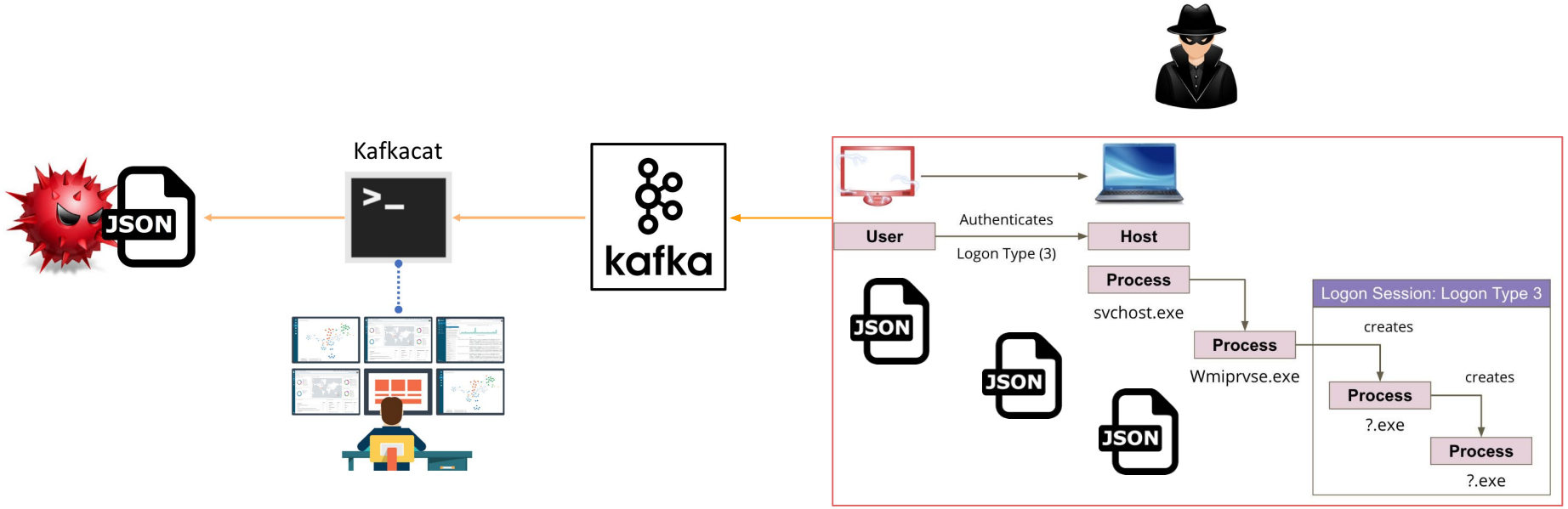
Erebor: ETW Events -> Event Log -> WEC -> HELK



How do you export (consume) data?

- I use **Kafkacat**!
- kafkacat is a generic non-JVM producer and consumer for Apache Kafka >=0.8, think of it as a netcat for Kafka.
- **In consumer mode**
  - Kafkacat reads messages from a topic and prints them to standard output (stdout). You can also redirect it to a file
- **In producer mode**
  - Kafkacat reads messages from standard input (stdin). You can also send data to kafkacat by adding data from a file.

Consuming Data -> Creating Mordor File

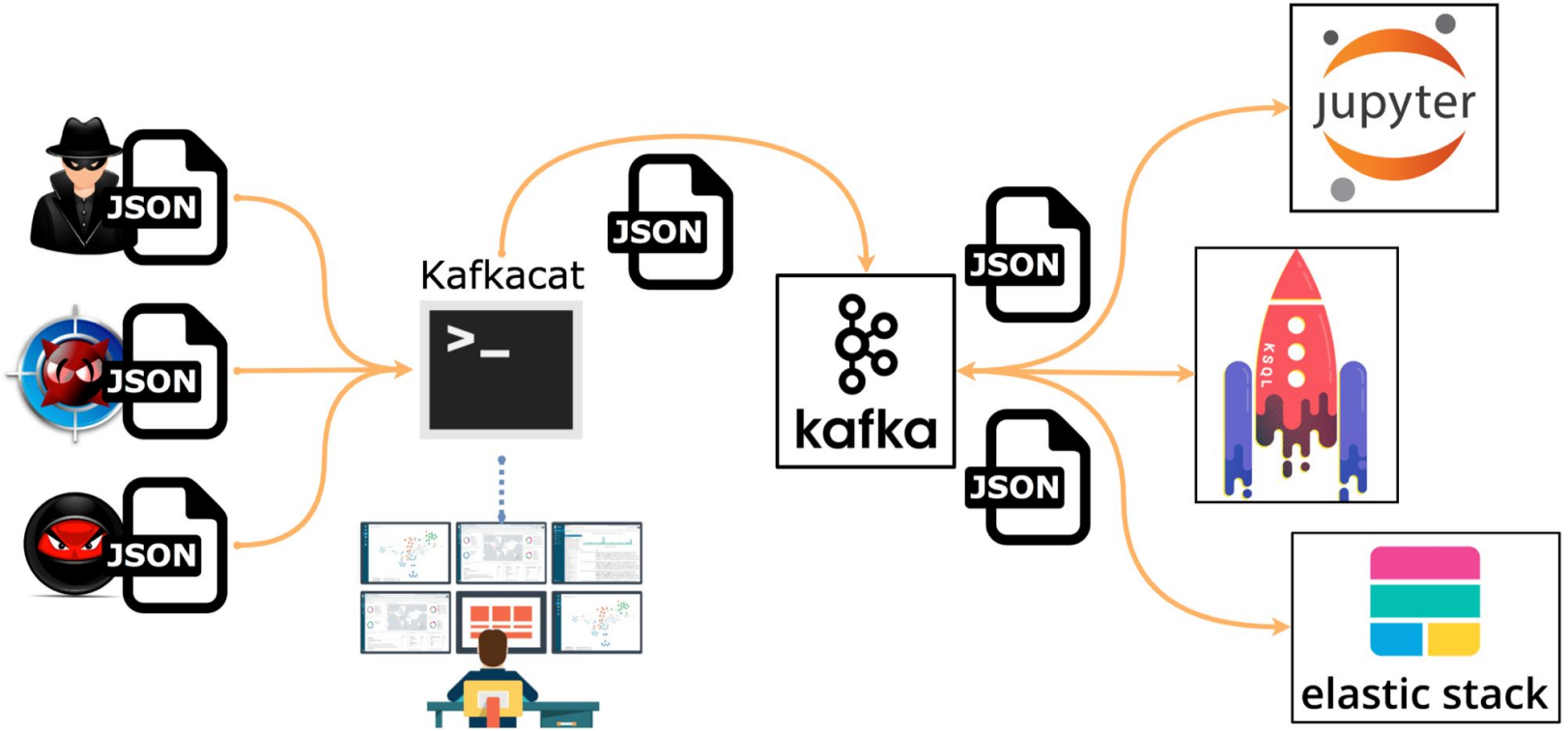


Consuming Data (Taking a snapshot of data)

```
$ kafkacat -b <Kafka-IP>:9092 -t
<kafka-Topic> -C -o end > file.json
```

- **-b** : Kafka broker
- **-t** : Topic to consume from
- **-C** : Consumer Mode
- **-o** : Offset to start consuming from

Producing Data (Injecting Adversary Dataset)



Producing Data (Injecting Adversary Dataset)

```
$ kafkacat -b <Kafka-IP>:9092 -t
<kafka-Topic> -P -l file.json
```

- **-b** : Kafka broker
- **-t** : Topic to produce to
- **-P** : Producer Mode
- **-l** : Send messages from a file

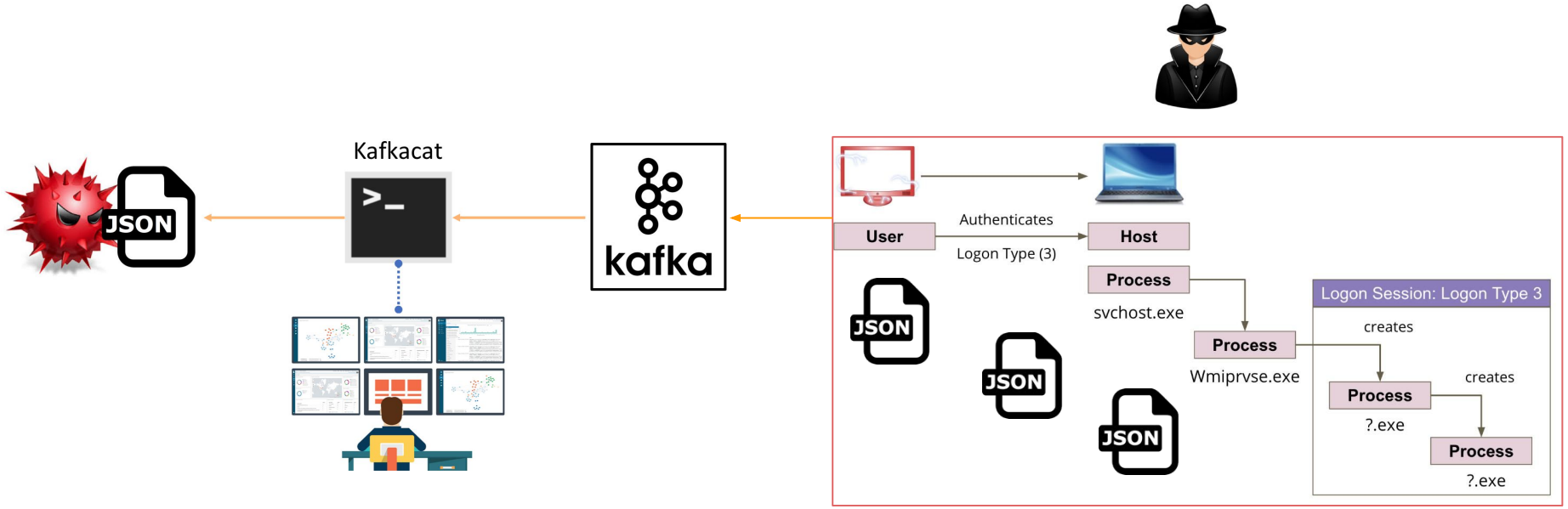
Can I download all the available Mordor datasets ?

```
$ git clone
https://github.com/hunters-forge/mordor.git

$ cd mordor/small_datasets/

$ find . -type f -name "*.tar.gz" -print0
| sudo xargs -0 -I{} tar xf {} -C .
```

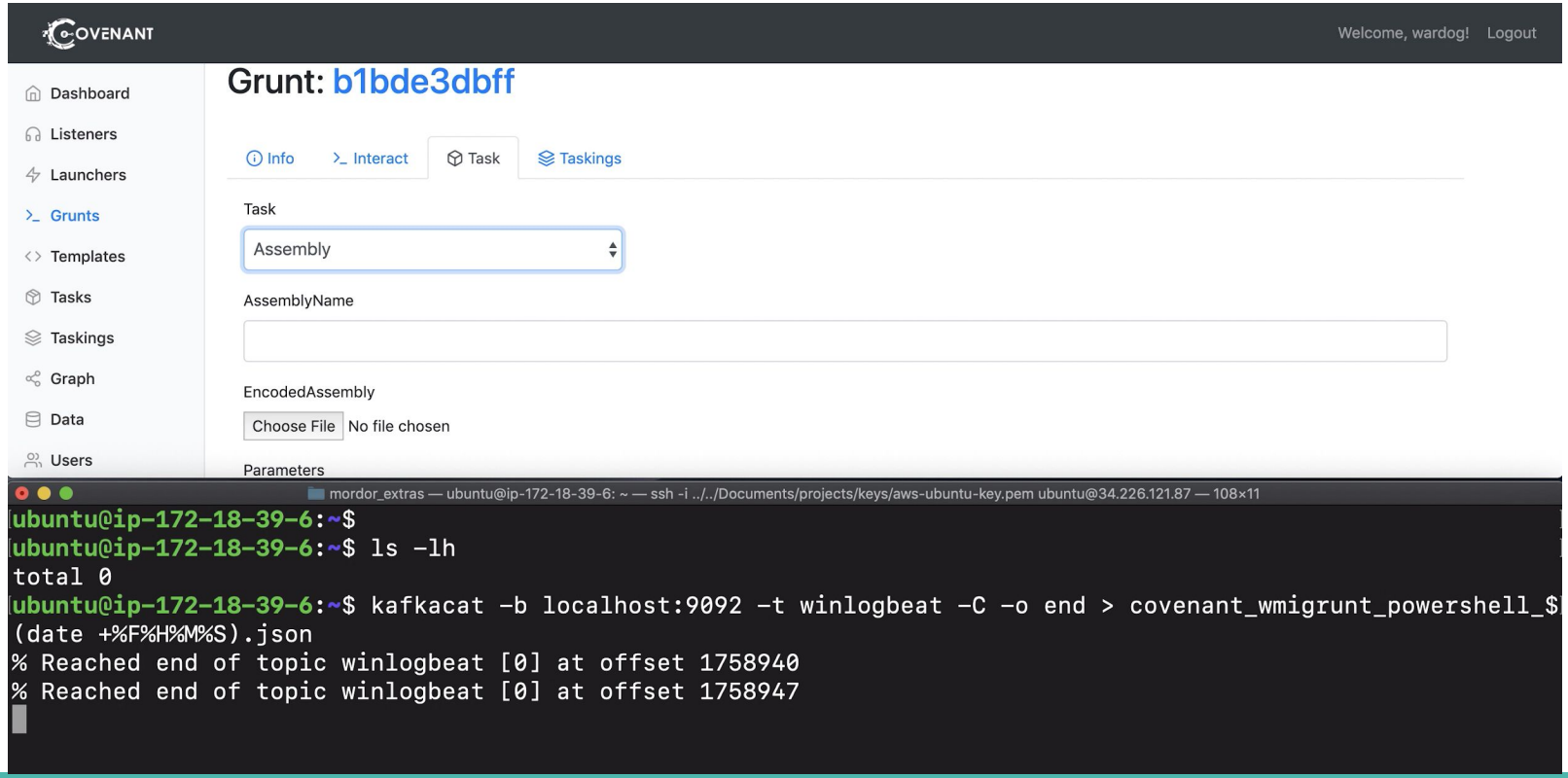
Purple Teaming - Mordor Style

One Adversarial Technique at the time!

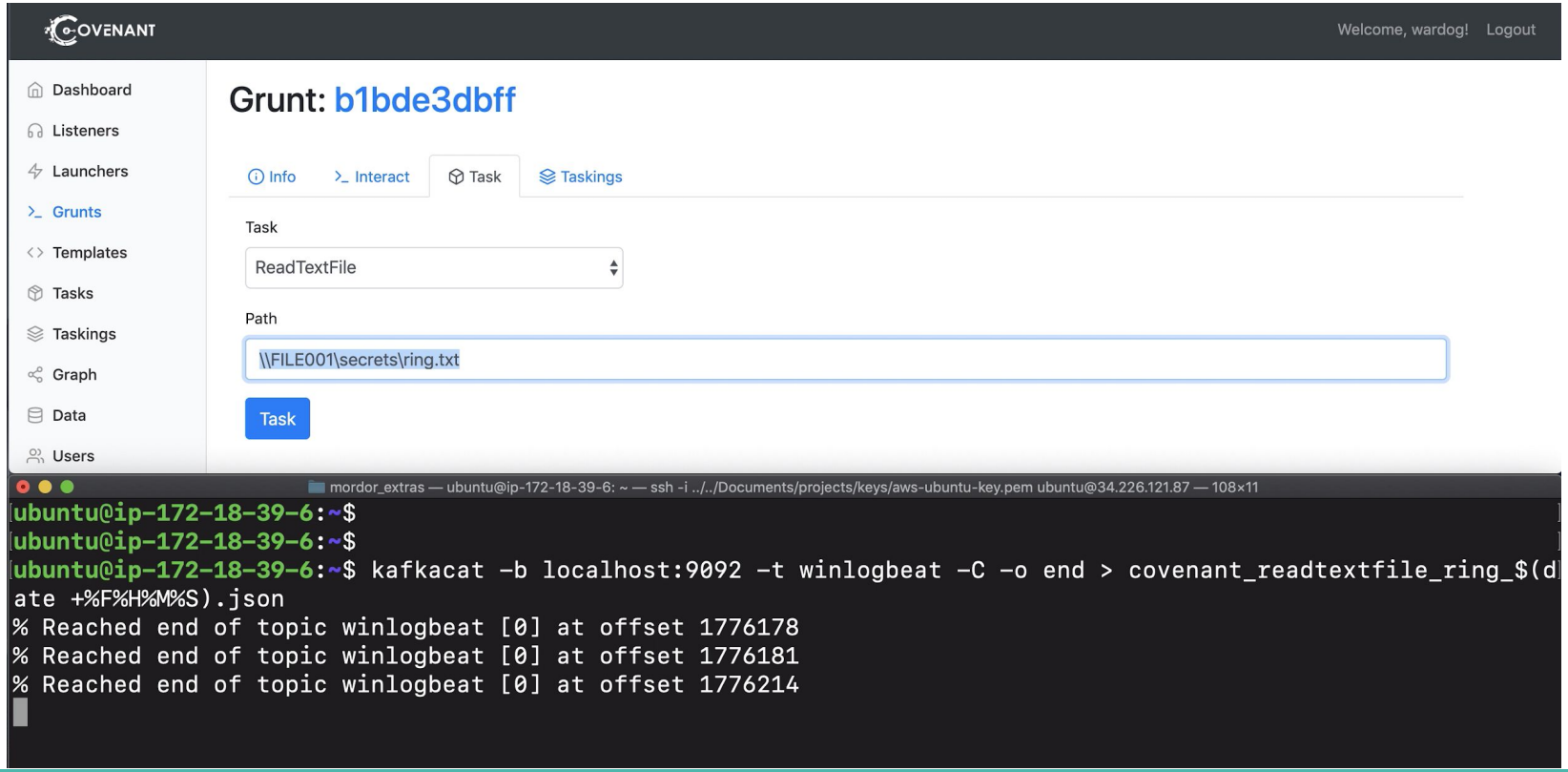
Let's do it!


JSON
Kafkacat
>_
kafka
User
Authenticates
Logon Type (3)
Host
Process
svchost.exe
Process
Wmiprvse.exe
Logon Session: Logon Type 3
creates
Process
?.exe
creates
Process
?.exe
JSON
JSON
JSON

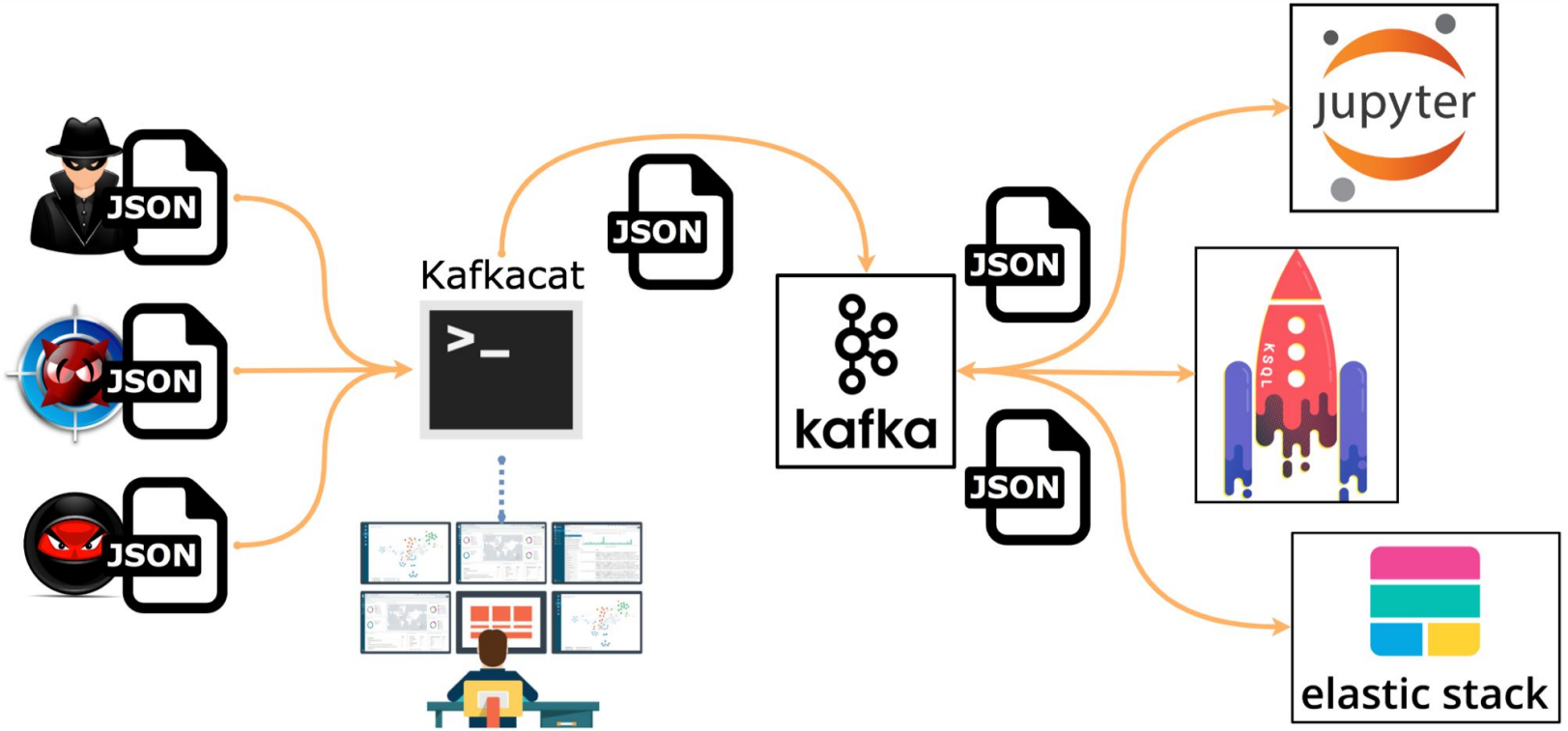
Let's do it! (Demo)



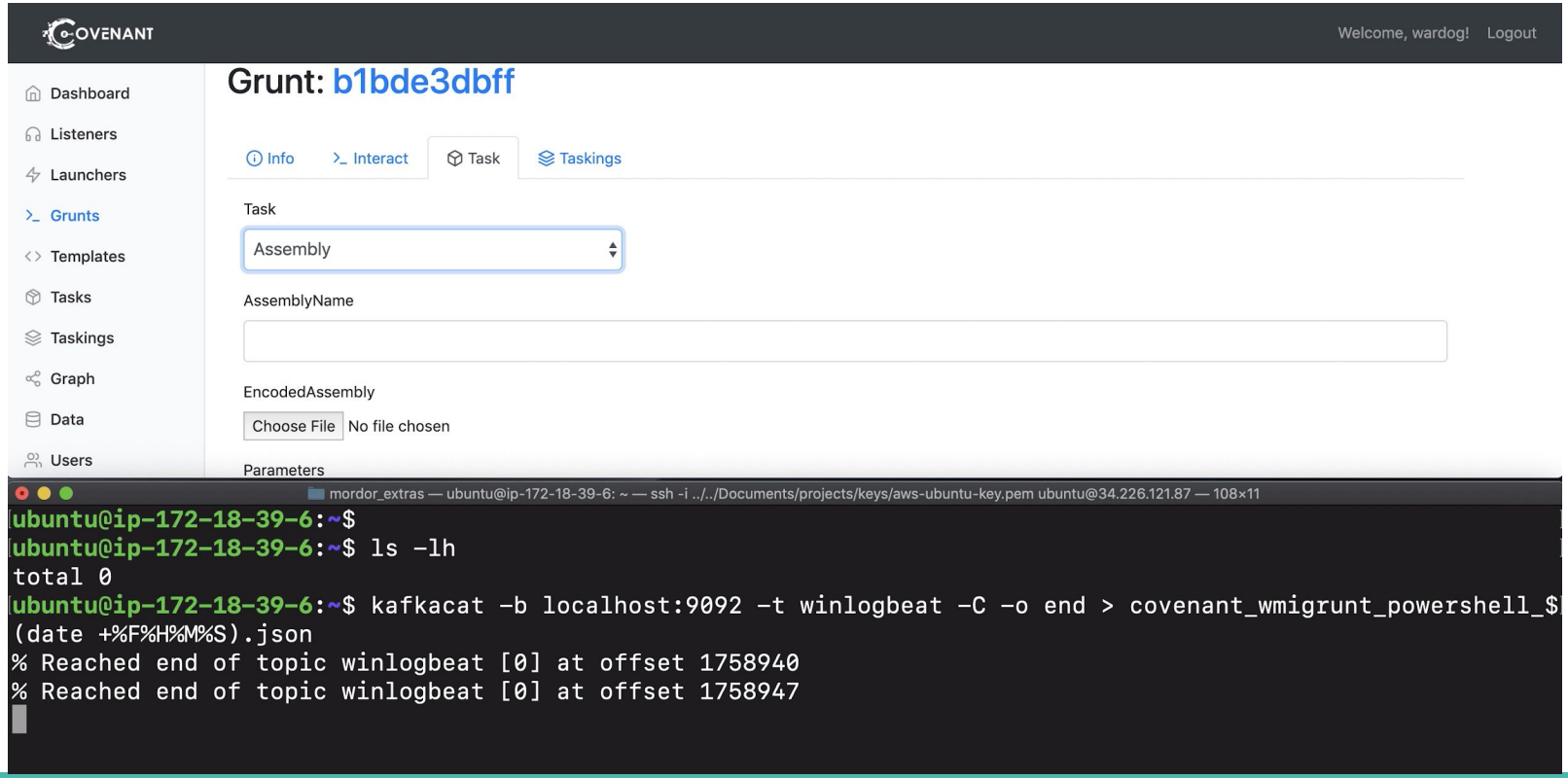
What about this other thing? (Demo)



Import and Analyze Mordor dataset (Demo)



Let's do it! (Demo)



Purple Teaming - Mordor Style

Adversary Emulation Plans (Full Campaign)!

Emulating Adversary APT3

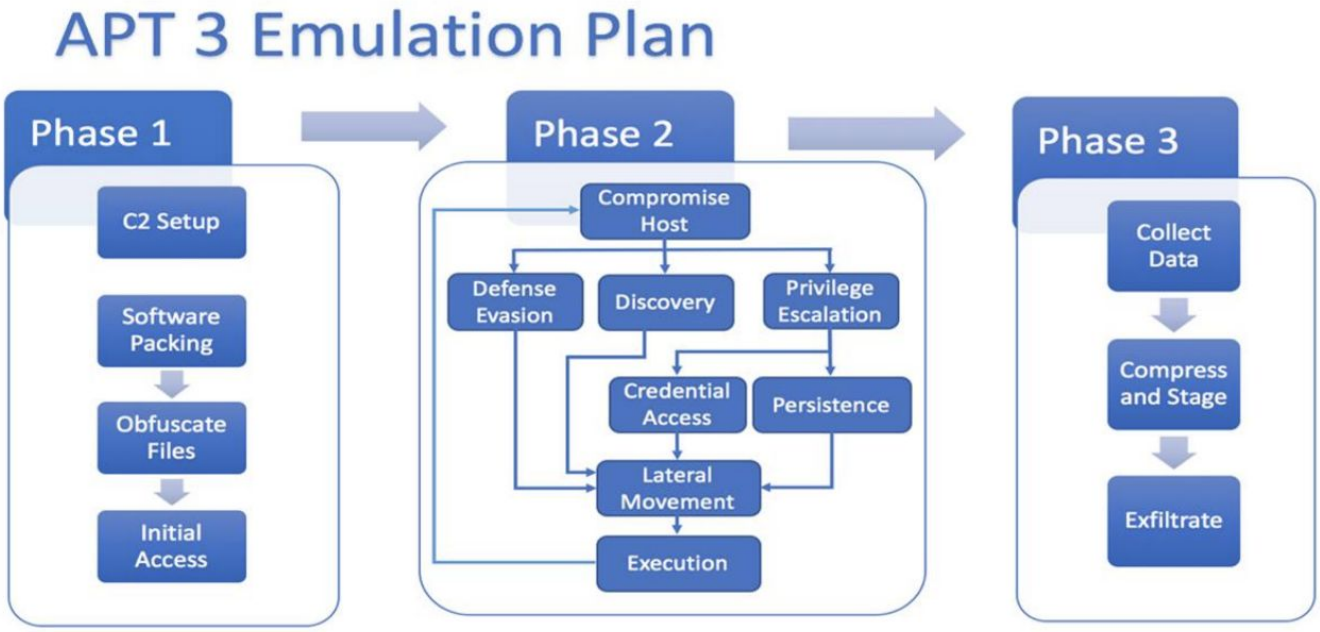
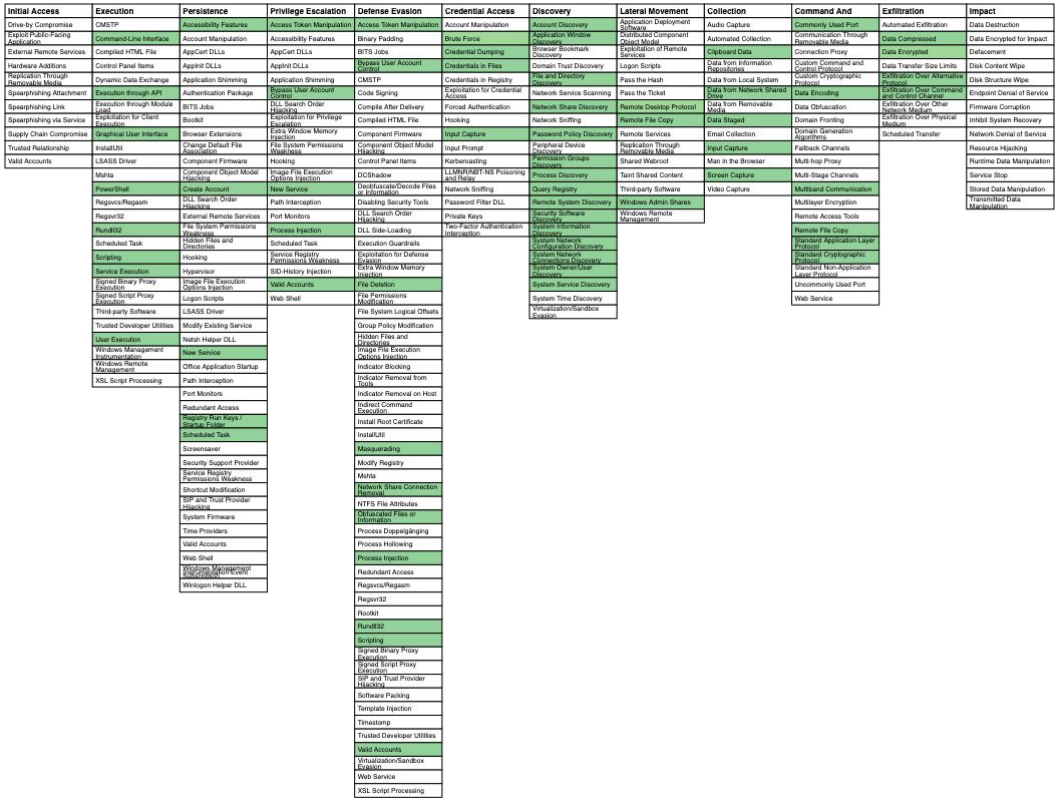


Figure 1 APT3's Three Phases of Action

https://attack.mitre.org/docs/APT3_Adversary_Emulation_Plan.pdf

ATT&CK Evaluations: APT 3 Round 1



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

https://attackevals.mitre.org/methodology/round1/scope.html

ATT&CK Evaluations: APT 3 Round 1 (Day 1 & 2)

Cobalt Strike

Step 1 - Initial Compromise

Step 2 - Initial Discovery

Step 3 - Privilege Escalation

Step 4 - Discovery for Lateral Movement

Step 5 - Credential Access

Step 6 - Lateral Movement

Step 7 - Persistence

Step 8 - Collection

Step 9 - Exfiltration

Step 10 - Execution of Persistence

Empire

Step 11 - Initial Compromise

Step 12 - Initial Discovery

Step 13 - Discovery for Lateral Movement

Step 14 - Privilege Escalation

Step 15 - Credential Access

Step 16 - Lateral Movement

Step 17 - Persistence

Step 18 - Collection

Step 19 - Exfiltration

Step 20 - Execution of Persistence

https://attackevals.mitre.org/methodology/round1/scope.html

ATT&CK Evaluations: APT 3 Round 1 Day 2

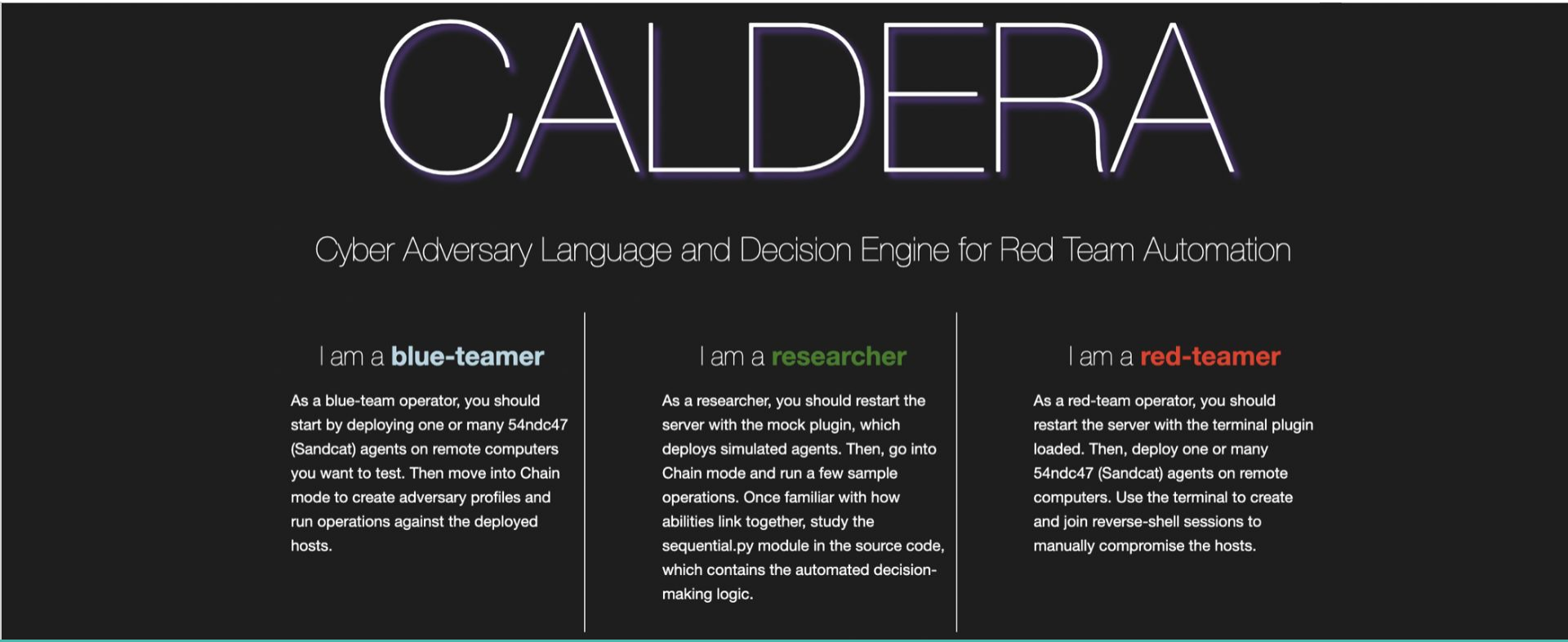
ATT&CK Eval Step	ATT&CK Eval Phase	Tactic	Technique Id	Technique Name	Empire Commands / Notes	Source Endpoint	Source Username	Target Endpoint	Target Username	ATT&CK Eval Scenario #2 Procedures
11.A.1	Initial Access	Defense Evasion Execution	T1064	Scripting	usestager windows/launcher_vbs set Listener https set OutFile /tmp/autoupdate.vbs execute user nmartha double clicks vbs script	HR001	nmartha	HR001	nmartha	Inintial compromise was emulated via a malicious VBScript. A legitimate user executed a VBScript stager, which launched PowerShell to download and execute an Empire payload
11.B.1	Initial Access	Command and Control	T1043	Commonly Used Port	listeners uselistener http set Name https set Host 10.0.10.106 set Port 443 set CertPath /opt/Empire/data execute	HR001	nmartha	HR001	nmartha	The executed Empire payload established an encrypted C2 channel over HTTPS on TCP port 443
11.B.1	Initial Access	Command and Control	T1071	Standard Application Layer Protocol	listeners uselistener http set Name https set Host 10.0.10.106 set Port 443 set CertPath /opt/Empire/data execute	HR001	nmartha	HR001	nmartha	The executed Empire payload established an encrypted C2 channel over HTTPS on TCP port 443
11.B.1	Initial Access	Command and Control	T1032	Standard Cryptographic Protocol	listeners uselistener http set Name https set Host 10.0.10.106 set Port 443 set CertPath /opt/Empire/data execute	HR001	nmartha	HR001	nmartha	The executed Empire payload established an encrypted C2 channel over HTTPS on TCP port 443
12.A.1	Initial Discovery	Discovery	T1016	System Network Configuration Discovery	shell route print	HR001	nmartha	HR001	nmartha	The route utility was executed via PowerShell to enumerate the local routing table.
12.A.2	Initial Discovery	Discovery	T1016	System Network Configuration Discovery	shell ipconfig /all	HR001	nmartha	HR001	nmartha	The ipconfig utility was executed via PowerShell to enumerate local TCP/IP network configuration information.
12.B.1	Initial Discovery	Discovery	T1033	System Owner/User Discovery	shell whoami /all /fo list	HR001	nmartha	HR001	nmartha	The whoami utility was executed via PowerShell to enumerate information about the current user
12.C.1	Initial Discovery	Discovery	T1057	Process Discovery	shell qprocess *	HR001	nmartha	HR001	nmartha	The qprocess utility was executed via PowerShell to enumerate local running processes.
12.D.1	Initial Discovery	Discovery	T1007	System Service Discovery	shell net start	HR001	nmartha	HR001	nmartha	The net utility was executed via PowerShell to enumerate local active services.
12.E.1	Initial Discovery	Defense Evasion Execution	T1064	Scripting	usemodule situational_awareness/host/winenum	HR001	nmartha	HR001	nmartha	Empire: Built-in WinEnum module executed to programmatically execute a series of enumeration techniques
12.E.1.1	Initial Discovery	Discovery	T1033	System Owner/User Discovery	usemodule situational_awareness/host/winenum	HR001	nmartha	HR001	nmartha	WinEnum: Get-UserInfo
12.E.1.2	Initial Discovery	Discovery	T1069	Permission Groups Discovery	usemodule situational_awareness/host/winenum	HR001	nmartha	HR001	nmartha	WinEnum: "AD Group Memberships"
12.E.1.3	Initial Discovery	Discovery	T1201	Password Policy Discovery	usemodule situational_awareness/host/winenum	HR001	nmartha	HR001	nmartha	WinEnum: "Password Last Changed"

https://github.com/hunters-forge/mordor/blob/master/large_datasets/apt3/scope/apt3_mordor_playbook.xlsx

Purple Teaming - Mordor Style

Automating Adversary Emulation Plans

Caldera: Automated Adversary emulation (ATT&CK)



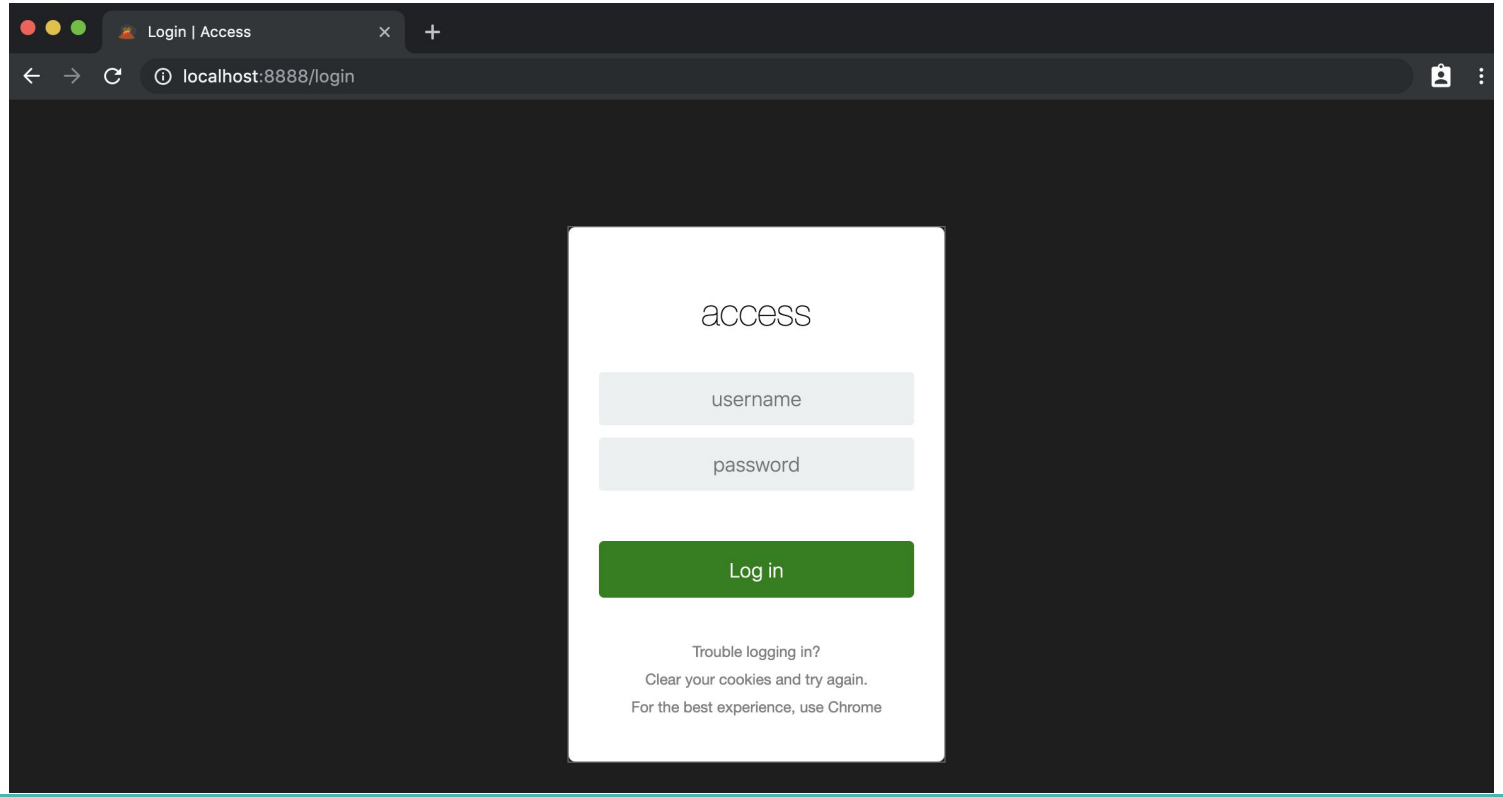
https://github.com/mitre/caldera

Caldera Installation

- git clone --recursive https://github.com/mitre/caldera.git
- cd caldera/
- pip3 install -r requirements.txt
- python -u server.py

- git clone https://github.com/Cyb3rWard0g/docker-caldera
- cd docker-caldera/
- docker-compose -f docker-compose-caldera.yml up --build -d

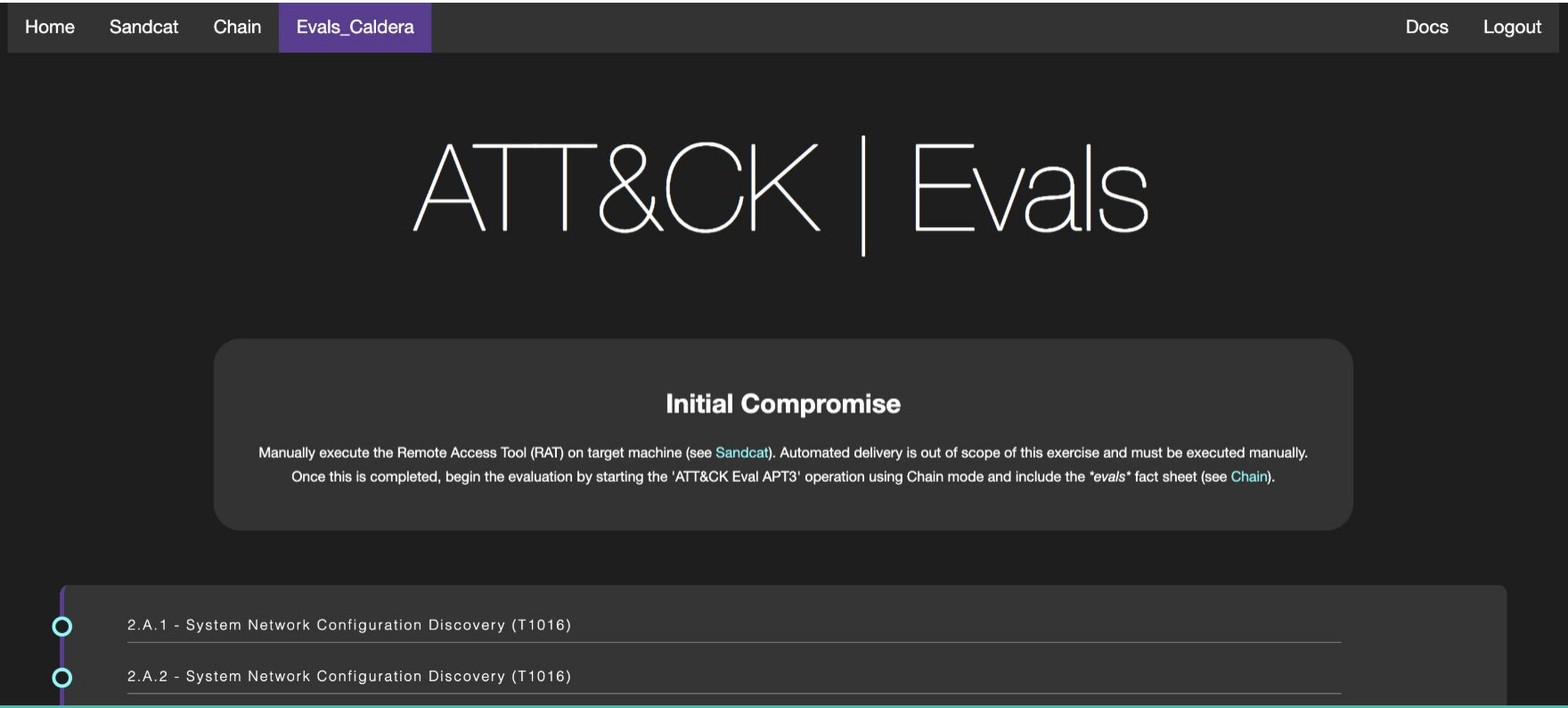
Caldera: Automated Adversary emulation (ATT&CK)



https://github.com/mitre/caldera

Is there a plugin for the ATT&CK evals?

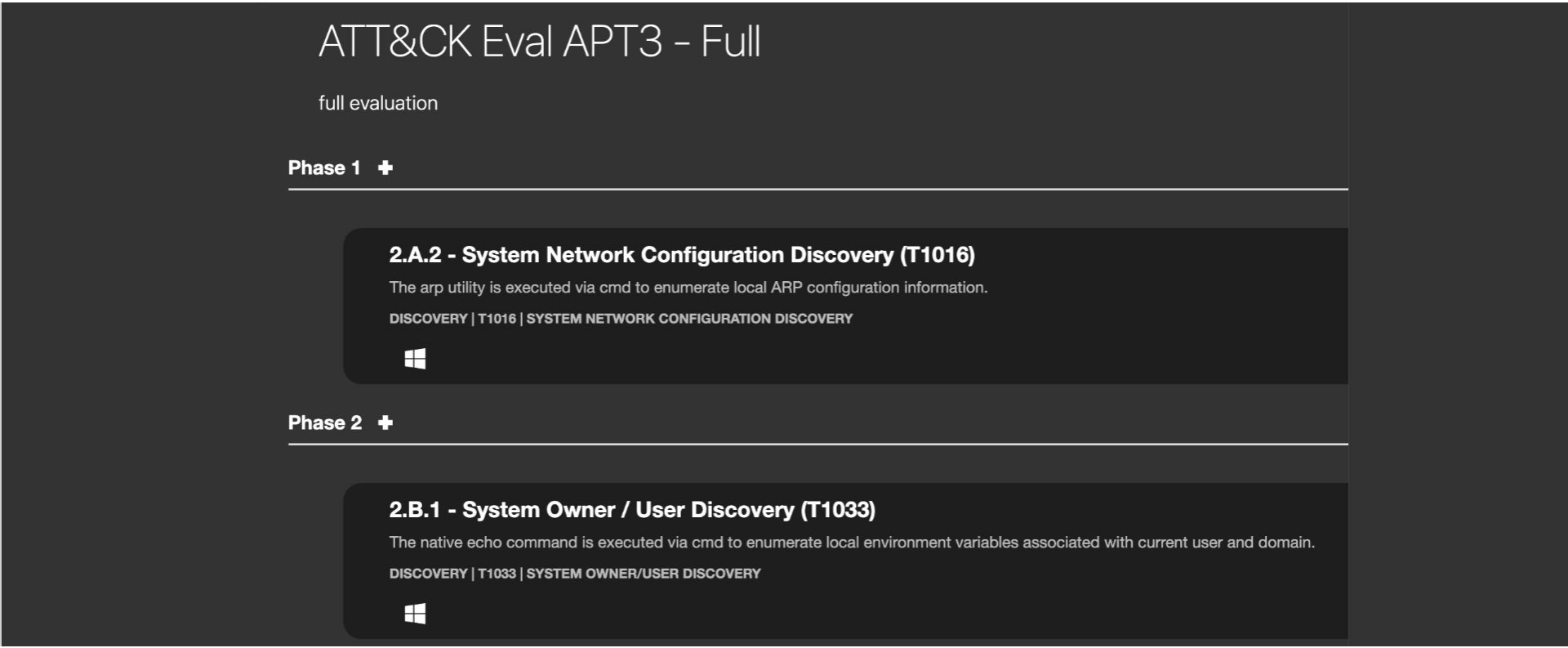
Caldera ATT&CK Evals Plugin (NEW!)



https://github.com/mitre-attack/evals_caldera

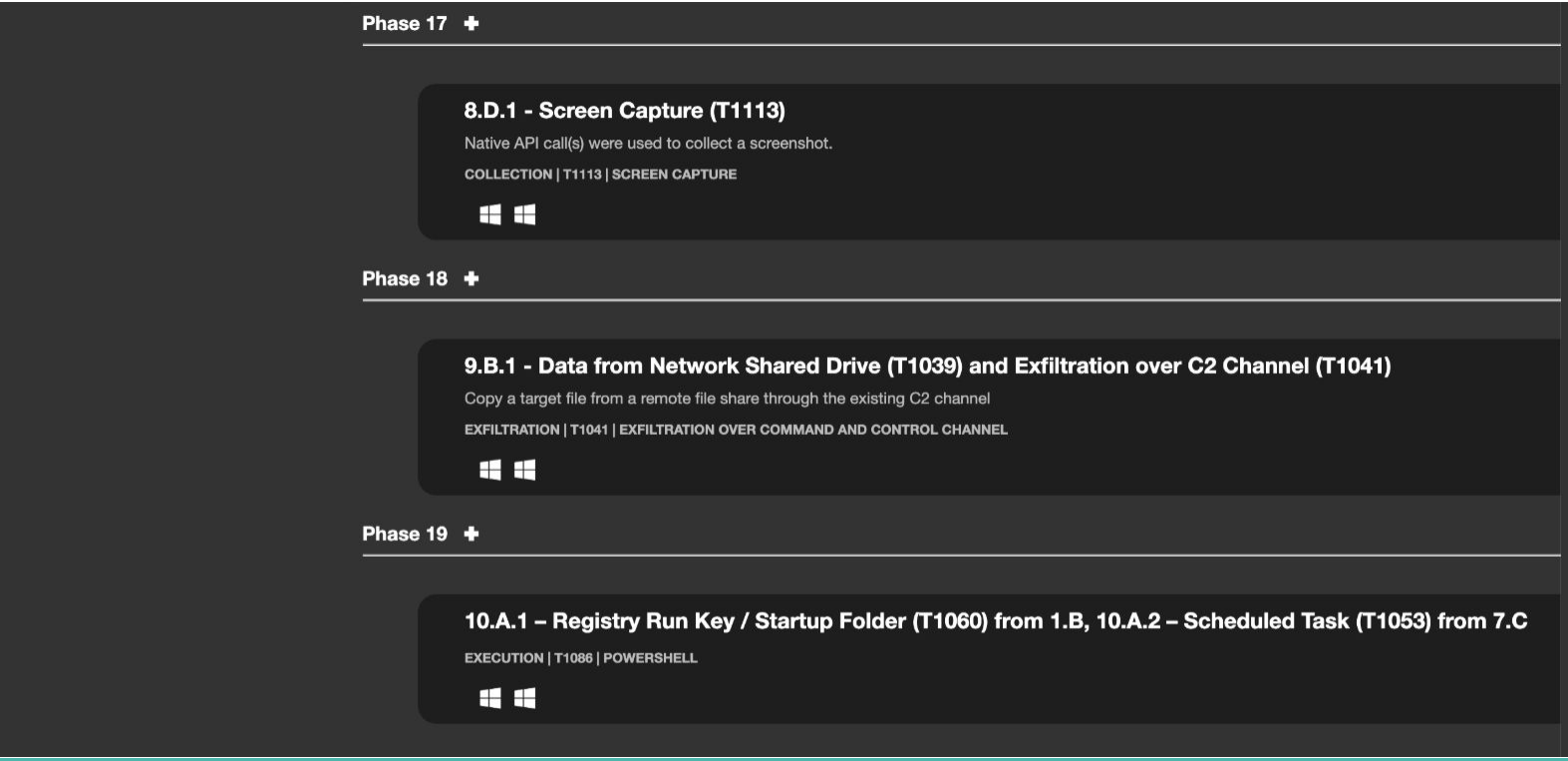
Whaaat? You can just let the plugin do it all?

Caldera: APT3 ATT&CK Eval Round 1 Day 1



https://attackevals.mitre.org/methodology/round1/scope.html

Caldera: APT3 ATT&CK Eval Round 1 Day 1



https://attackevals.mitre.org/methodology/round1/scope.html

Caldera: ATT&CK Evals Plugin Installation

- cd caldera/plugins
- git clone https://github.com/mitre-attack/evals_caldera.git
- Modify caldera/conf/local.yml file and add

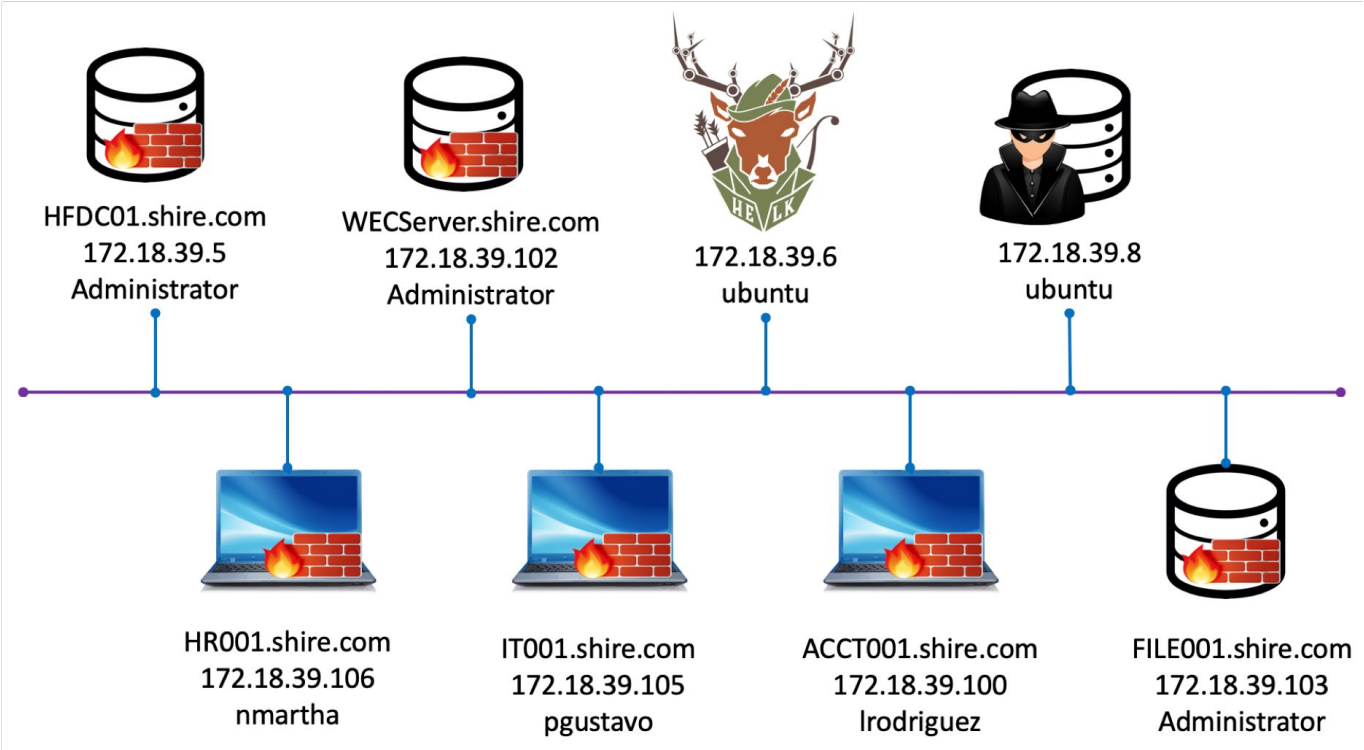
  “- evals_caldera” under plugins

Can I do it at home?

Blacksmith Project & Mordor!

- git clone https://github.com/hunters-forge/Blacksmith
- Follow pre-requirements before deploying **Shire Environment:** https://blacksmith.readthedocs.io/en/latest/mordor_shire.html
- Update parameter **RTODefaultC2** to **caldera** in cfn-parameters/shire/c2-server-parameters.json
- Update parameter **RestrictLocation** in cfn-parameters/shire/ec2-network-parameters.json
- Run **./deploy.sh -e ‘shire’**

Blacksmith Project & Mordor!

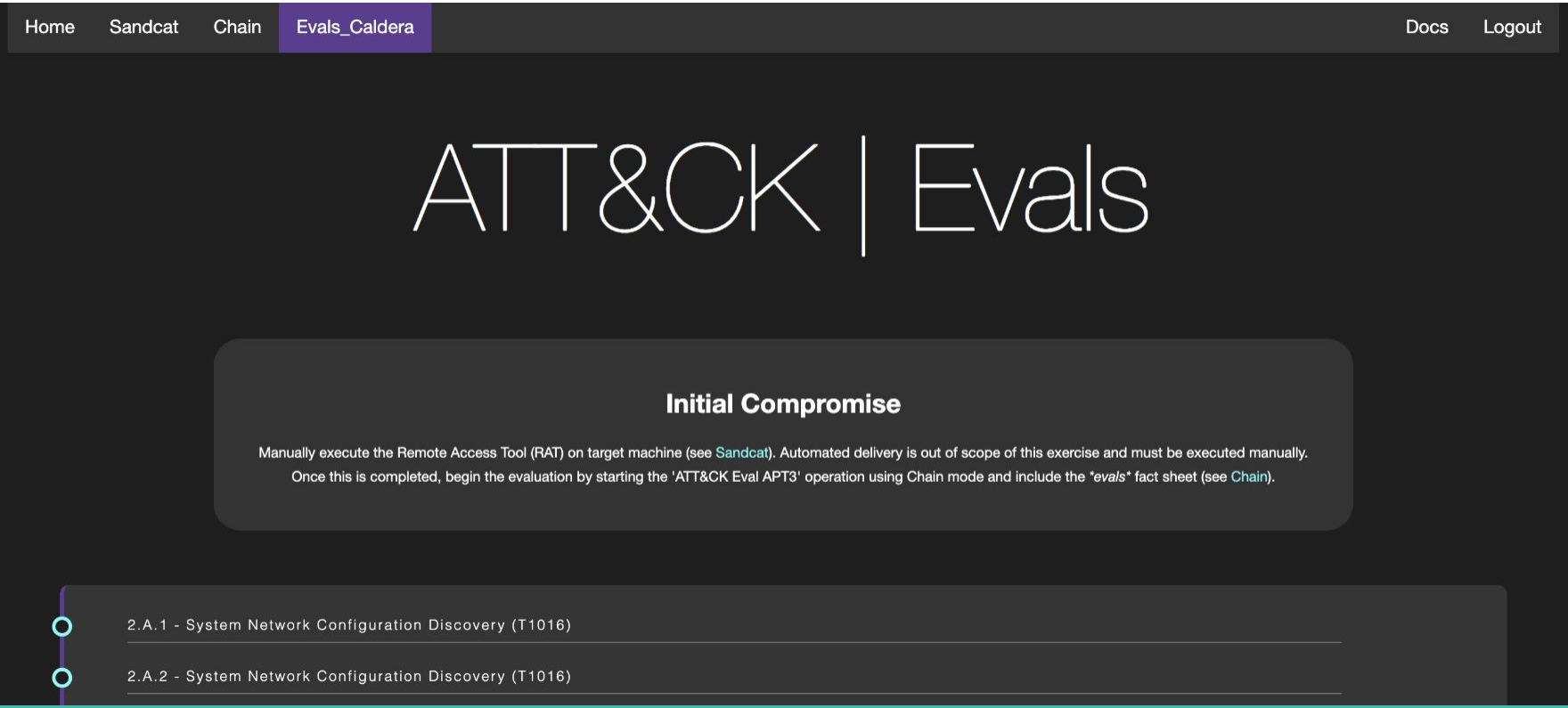


https://github.com/hunters-forge/Blacksmith/blob/master/aws/mordor

Thank you so much Daniel!

https://www.sans.org/cyber-security-summit/archives/file/summit-archive-1559672321.pdf

Caldera: APT3 ATT&CK Eval Round 1 Day 1 (Video)



What did we learn today?

Purple Teaming - Mordor Style

- Practice like you play before production!
- Be prepared and make sure you set rules of engagement!
- Keep the back and forth going and at the same time:
  - Take data snapshots of each technique or whole emulation plan (very very helpful and convenient!!)
  - Reuse pre-recorded data for additional deep research
  - Expedite analytic validations with the datasets
  - Train other purple teamers with re-recorded events
  - Learn from the data produced (Red & Blue)

Purple Teaming - Mordor Style

- Practice like you play before production!
- Be prepared and make sure you set rules
- Keep the back and forth going and at the same time:
  - Take data snapshots of each technique or whole emulation plan (very very helpful and convenient!!)
  - Reuse pre-recorded data for additional deep research
  - Expedite analytic validations with the datasets
  - Train other purple teamers with re-recorded events
  - **Learn from the data produced (Red & Blue)**

Purple Teaming - Mordor Style

- Practice like you play before production!
- Be prepared and make sure you set rules
- Keep the back and forth going and at the same time:
  - Take data snapshots of each technique or whole emulation plan (very very helpful and convenient!!)
  - Reuse pre-recorded data for additional deep research
  - Expedite analytic validations with the datasets
  - Train other purple teamers with re-recorded events
  - Learn from the data produced (Red & Blue)
- **Empower others and give back to the community!**

Threat Hunters Forge Community!

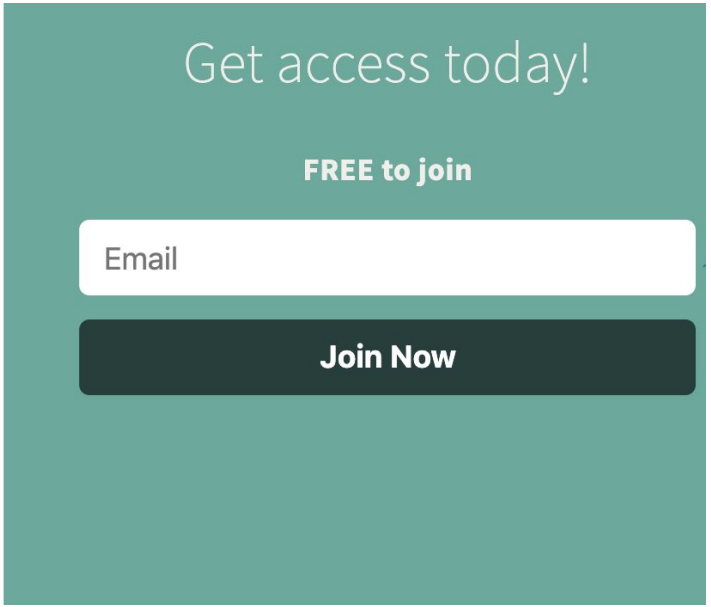
Threat Hunters Forge Slack Community!

ThreatHunting

Threat Hunters Forge

Join the Threat Hunters Forge Slack Community!

A community led effort to share detection strategies and to support open source projects to aid the development of security analytics and tooling for threat hunting!



https://launchpass.com/threathunting

Goal: Share and Empower the Community!

Let’s do it together!

Threat Hunters Forge References

- **GitHub**: https://github.com/hunters-forge
- **Python Library**: https://github.com/Cyb3rPanda/openhunt
- **Slack Invitation**: https://launchpass.com/threathunting
- **Official Blog**: https://medium.com/threat-hunters-forge
- **Founders**: @Cyb3rWard0g & @Cyb3rPandaH
- **Official Twitter:** @HuntersForge
- @HunterPlaybook
- @THE_HELK
- @OSSEM_Project, @Mordor_Project & More

Thank You! Muchas Gracias!