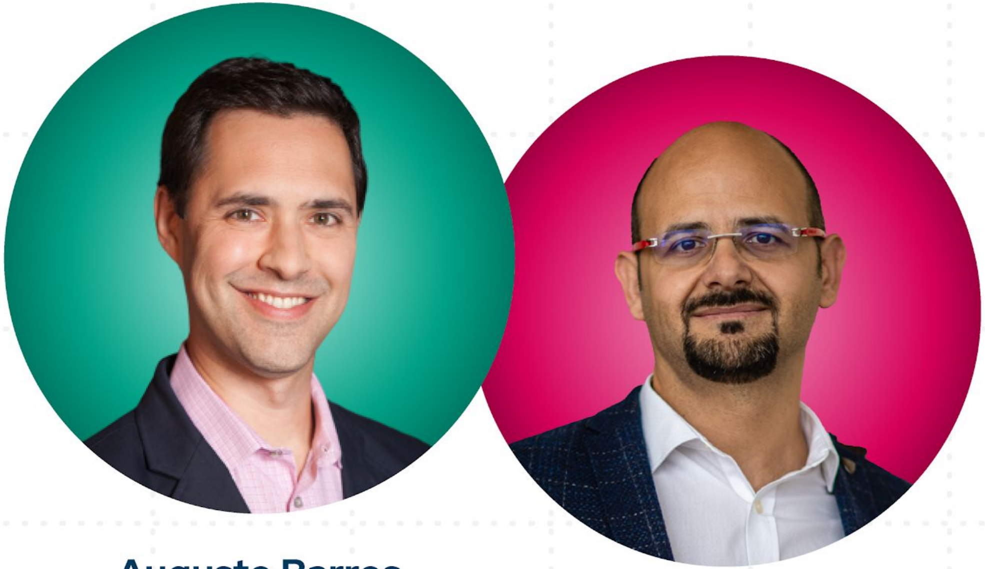
SOC ReLoad
#SOCReLoad21

TRACK 2 - OPERATIONS

Enabling the Detection Development Lifecycle with Attack Simulation



Augusto Barros
VP Security Evangelist

Volkan Ertürk
CTO & Co-Founder

SECURONIX

PICUS

Agenda

- The Detection Development Life-Cycle
- How can Attack Simulation Support the DDLC?

Detection Development?

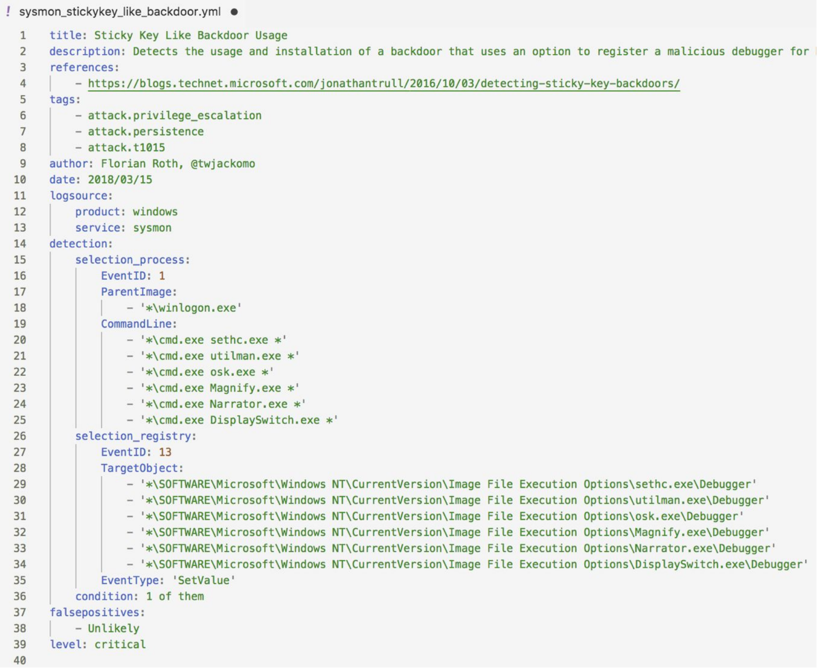
- Many threat detection tools have separate **“content”** that defines what they should be looking for
- A.K.A “use cases” (SIEM)
- Can also include signatures, policies and rules for tools such as EDR, NDR and DLP
- Content can be provided by the vendor or developed by the user
    - Is the content provided by the vendor enough, or aligned to what you need?

SIEM Use Case

- As described by Gartner:

  "a specific set of conditions or events, usually related to a specific threat, to be detected or reported by the security tool"

- *Usually seem as (a) rule(s)*
  - *More complex scenarios require multiple rules, ML models, threat chains and other resources*



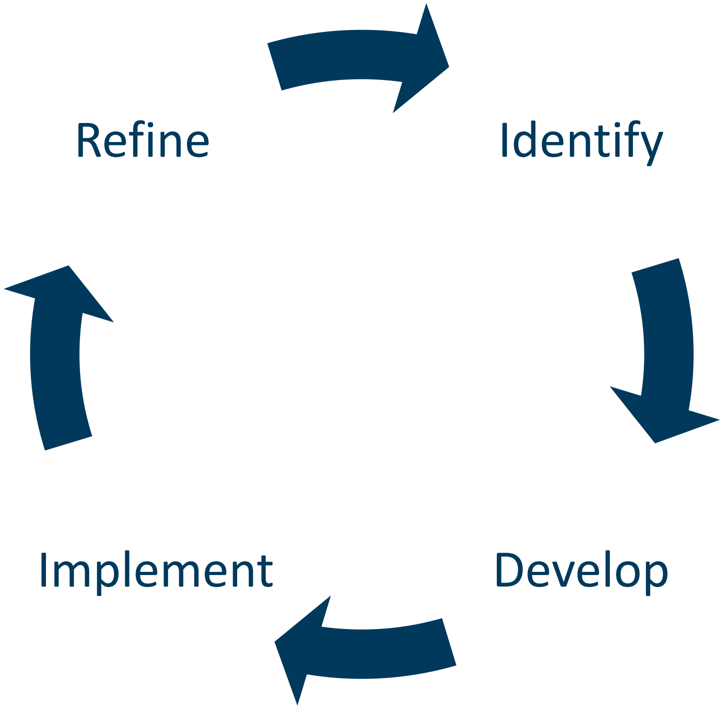
sysmon_stickykey_like_backdoor.yml

```yaml
title: Sticky Key Like Backdoor Usage
description: Detects the usage and installation of a backdoor that uses an option to register a malicious debugger for
references:
    - https://blogs.technet.microsoft.com/jonathantrull/2016/10/03/detecting-sticky-key-backdoors/
tags:
    - attack.privilege_escalation
    - attack.persistence
    - attack.t1015
author: Florian Roth, @twjackomo
date: 2018/03/15
logsource:
    product: windows
    service: sysmon
detection:
    selection_process:
        EventID: 1
        ParentImage:
            - '*\winlogon.exe'
        CommandLine:
            - '*\cmd.exe sethc.exe *'
            - '*\cmd.exe utilman.exe *'
            - '*\cmd.exe osk.exe *'
            - '*\cmd.exe Magnify.exe *'
            - '*\cmd.exe Narrator.exe *'
            - '*\cmd.exe DisplaySwitch.exe *'
    selection_registry:
        EventID: 13
        TargetObject:
            - '*\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options\sethc.exe\Debugger'
            - '*\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options\utilman.exe\Debugger'
            - '*\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options\osk.exe\Debugger'
            - '*\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options\Magnify.exe\Debugger'
            - '*\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options\Narrator.exe\Debugger'
            - '*\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options\DisplaySwitch.exe\Debugger'
        EventType: 'SetValue'
    condition: 1 of them
falsepositives:
    - Unlikely
level: critical

```

The Detection Development Life Cycle

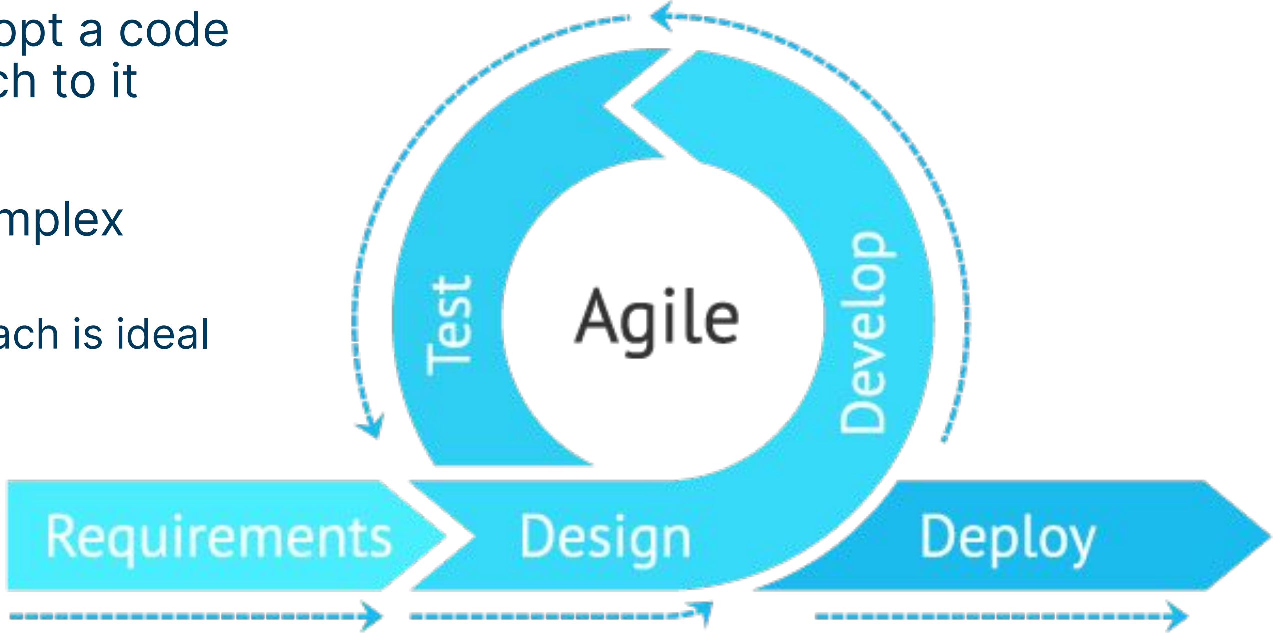
A mature threat detection practice must include processes to identify, develop, implement and refine detection content



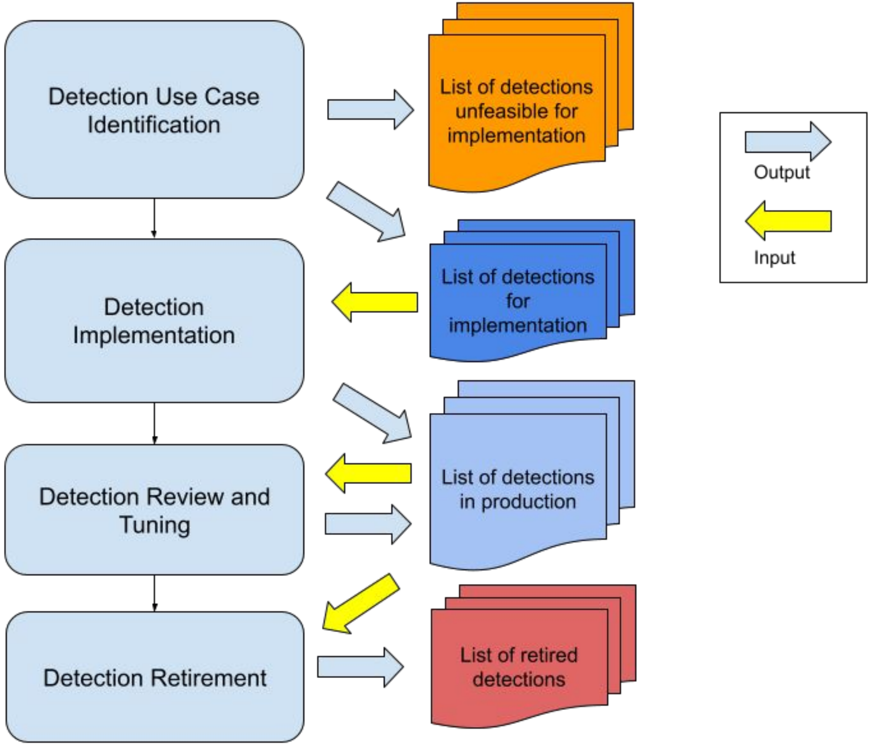
Why a "development life-cycle"?

- Organizations most successful with threat detection content adopt a code development approach to it

- But cannot be too complex
  - Agility is important
  - An Agile style approach is ideal



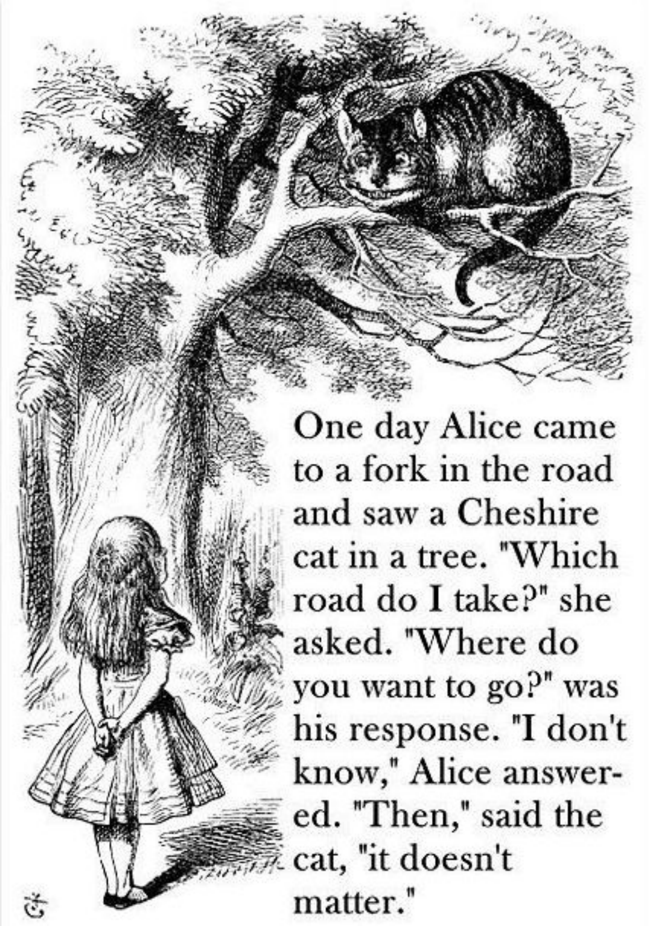
Example


Detection Use Case Identification
List of detections unfeasible for implementation
List of detections for implementation
Detection Implementation
List of detections in production
Detection Review and Tuning
List of retired detections
Detection Retirement
Output
Input

Where to start?

Major problem with use cases is answering the question:

Which ones do I want to deploy on my SIEM?

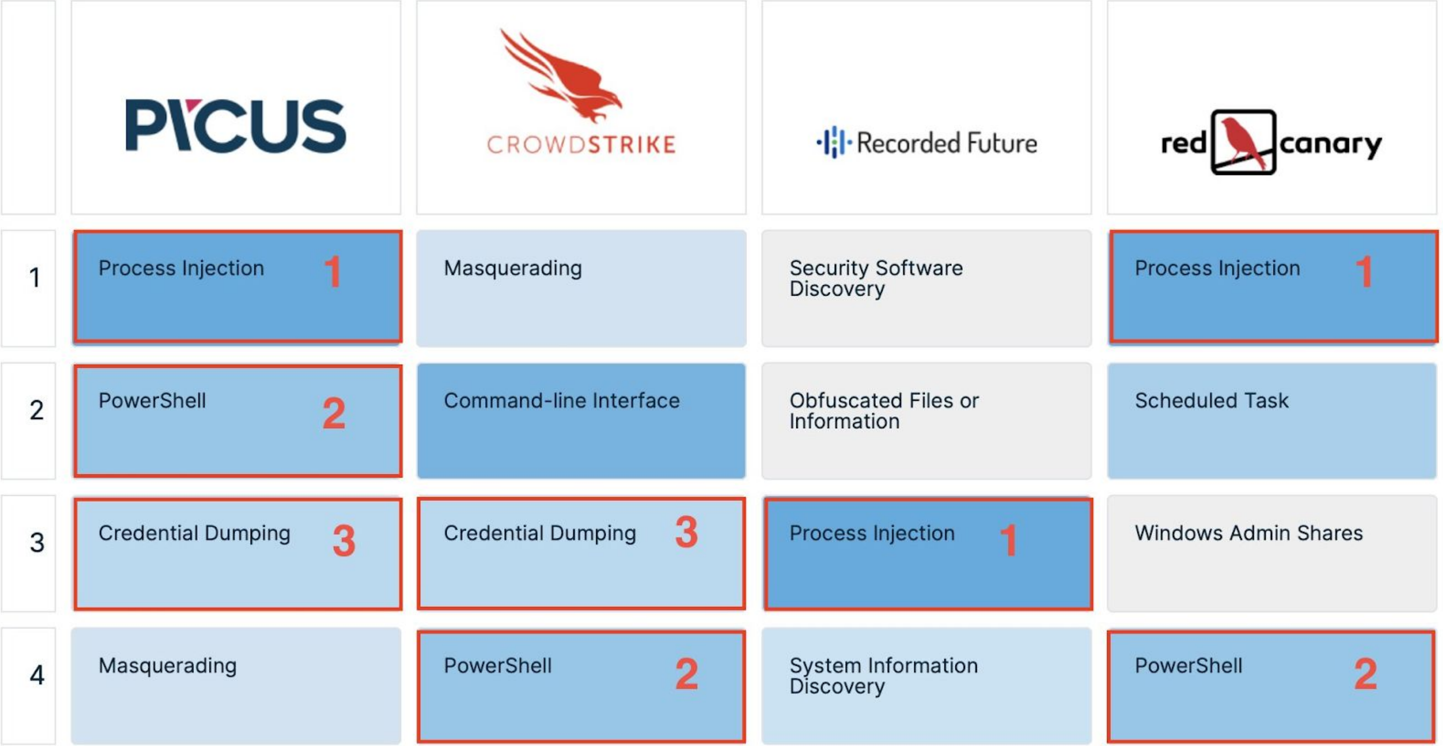


One day Alice came to a fork in the road and saw a Cheshire cat in a tree. "Which road do I take?" she asked. "Where do you want to go?" was his response. "I don't know," Alice answered. "Then," said the cat, "it doesn't matter."

Identifying Use Cases

You need to know first what you want to accomplish

Prioritizing importance



	PICUS	CROWDSTRIKE	Recorded Future	red canary
1	Process Injection 1	Masquerading	Security Software Discovery	Process Injection 1
2	PowerShell 2	Command-line Interface	Obfuscated Files or Information	Scheduled Task
3	Credential Dumping 3	Credential Dumping 3	Process Injection 1	Windows Admin Shares
4	Masquerading	PowerShell 2	System Information Discovery	PowerShell 2

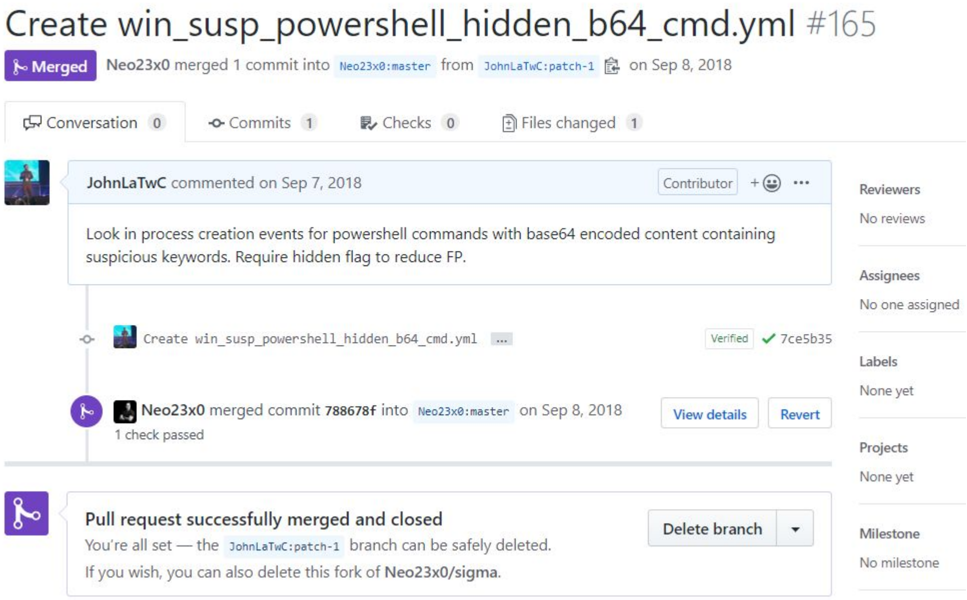
Source: Dr. Suleyman Ozarslan (@su13ym4n)

Prioritizing Feasibility

- Do you have the logs?
- Do you have context data?
- Do you have the tools?
- Can you handle the output?
  - People (# of analysts)
  - Technology (capacity)
  - Process (playbooks)

Implementation

- “Detection as Code” is becoming popular
- Strong analytics capabilities help increasing implementation options
  - What doesn’t work as a rule may work as ML model
- Out of the box content can speed up implementation of initial use cases
- Community efforts to share content as growing (e.g. Sigma)



Create win_susp_powershell_hidden_b64_cmd.yml #165

Merged Neo23x0 merged 1 commit into `Neo23x0:master` from `JohnLaTwC:patch-1` on Sep 8, 2018

JohnLaTwC commented on Sep 7, 2018 Contributor

Look in process creation events for powershell commands with base64 encoded content containing suspicious keywords. Require hidden flag to reduce FP.

`Create win_susp_powershell_hidden_b64_cmd.yml` ... Verified ✓ 7ce5b35

Neo23x0 merged commit **788678f** into `Neo23x0:master` on Sep 8, 2018

1 check passed

Pull request successfully merged and closed

You’re all set — the `JohnLaTwC:patch-1` branch can be safely deleted.

If you wish, you can also delete this fork of **Neo23x0/sigma**.

Reviewers: No reviews

Assignees: No one assigned

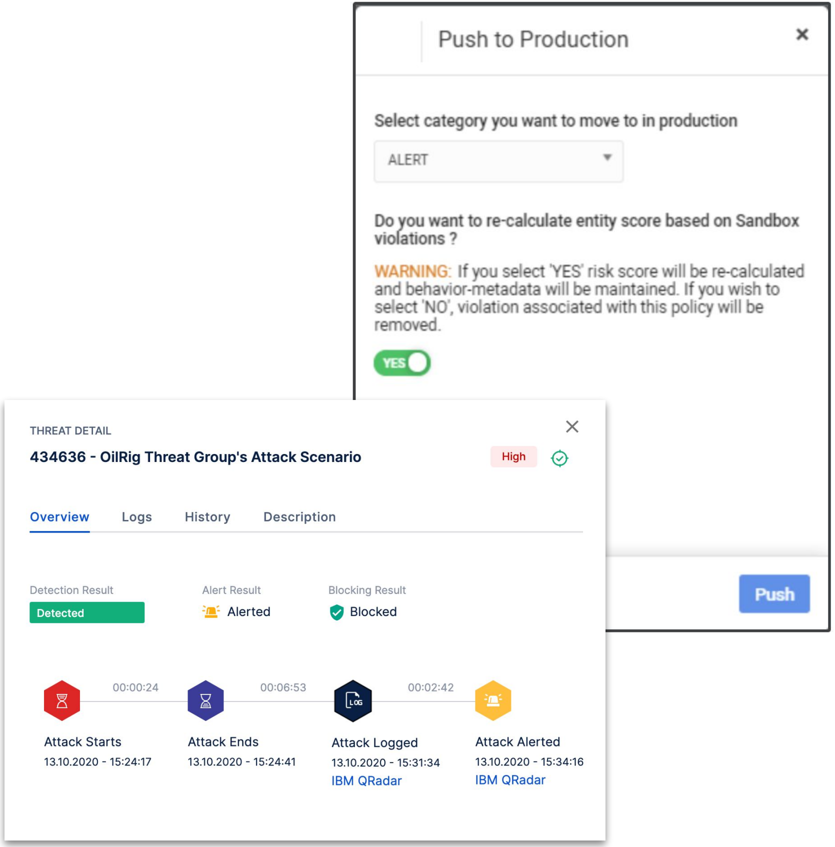
Labels: None yet

Projects: None yet

Milestone: No milestone

Testing

- Testing SIEM content is not always easy
  - Do you have a test environment?
  - Do you have the right data in the test environment?
- Testing environments are costly and it's hardly to replicate production context
  - User activity + Attacks
- Attack simulation allows testing in the production environment



Push to Production ×

Select category you want to move to in production

ALERT ▾

Do you want to re-calculate entity score based on Sandbox violations ?

WARNING: If you select 'YES' risk score will be re-calculated and behavior-metadata will be maintained. If you wish to select 'NO', violation associated with this policy will be removed.

YES

Push

THREAT DETAIL

434636 - OilRig Threat Group's Attack Scenario High

Overview Logs History Description

Detection Result: Detected

Alert Result: Alerted

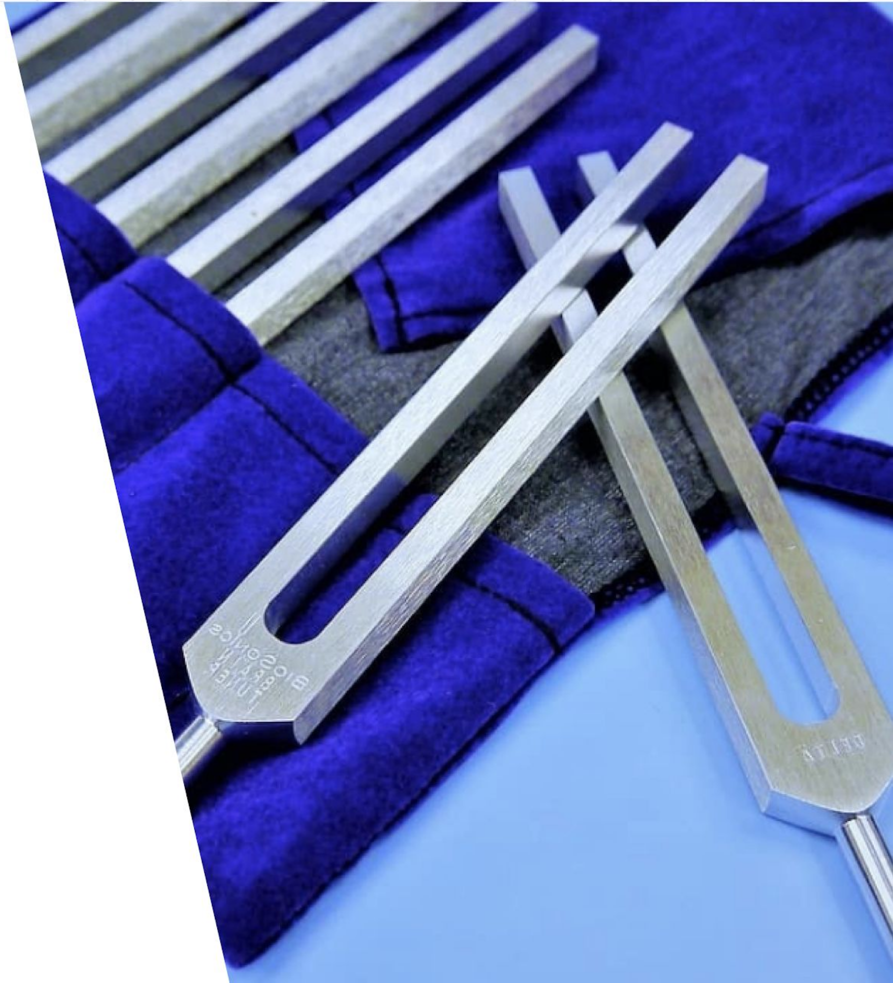
Blocking Result: Blocked

Attack Starts 13.10.2020 - 15:24:17 — 00:00:24 — Attack Ends 13.10.2020 - 15:24:41 — 00:06:53 — Attack Logged 13.10.2020 - 15:31:34 IBM QRadar — 00:02:42 — Attack Alerted 13.10.2020 - 15:34:16 IBM QRadar

Refining

The good and old “tuning”

- Do not believe in who tells you “there is no tuning”
- It’s not only about False Positives
- Prevalence of events matter!
- Can you answer "Is it still working?"



Measuring the Process

- A DDLC Process provides useful metrics
  - Metrics from unfeasible use cases
    - Data sources missing
    - Tools deficiencies
  - Metrics from tuning use cases
    - Implementation quality
    - Tools deficiencies
    - Data quality issues
  - Metrics from attack simulation
    - Gaps in coverage
    - Efficiency problems

How Can Attack Simulation Support the DDLC?

- Developing detections can be overwhelming
  - What to do?
  - What to do first?
  - Is it working?
- Attack Simulation can help in all these challenges

We have a detection development methodology!

Yet successful implementation is challenging.

- Manual and time-consuming
- Requires diverse skill-set
- Error-prone
- Ever-changing threats

How to Tackle

Empower Detection Development Lifecycle **with Attack Simulation**

1. Threat Selection
2. Adversary Emulation
3. Log Validation
4. Alert Validation
5. Continuous Improvement

1. Threat Selection

Identify the relevant set of threats to validate the use-case.

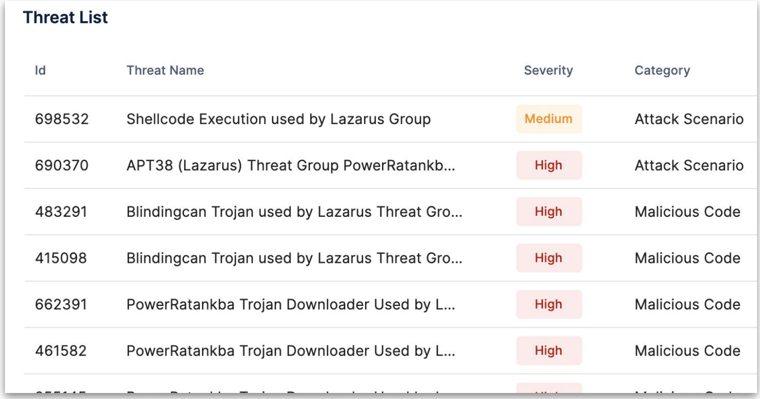
- Available threat content
- Based on your past incidents
- Threat Intelligence

1. Threat Selection

Identify the relevant set of threats to validate the use-case.

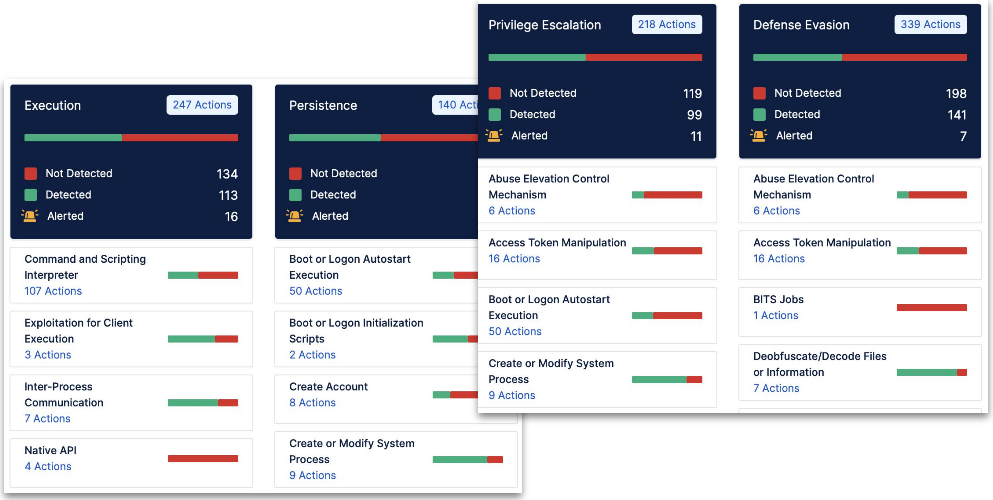
- Start with “Relevant Threats” that you do not have visibility of or based on your past incidents

- Start with “your weakest technique” or “most used”



Threat List

Id	Threat Name	Severity	Category
698532	Shellcode Execution used by Lazarus Group	Medium	Attack Scenario
690370	APT38 (Lazarus) Threat Group PowerRatankb...	High	Attack Scenario
483291	Blindingcan Trojan used by Lazarus Threat Gro...	High	Malicious Code
415098	Blindingcan Trojan used by Lazarus Threat Gro...	High	Malicious Code
662391	PowerRatankba Trojan Downloader Used by L...	High	Malicious Code
461582	PowerRatankba Trojan Downloader Used by L...	High	Malicious Code



Execution — 247 Actions

🟥 Not Detected	134
🟩 Detected	113
🚨 Alerted	16

- Command and Scripting Interpreter — 107 Actions
- Exploitation for Client Execution — 3 Actions
- Inter-Process Communication — 7 Actions
- Native API — 4 Actions

Persistence — 140 Acti

- 🟥 Not Detected
- 🟩 Detected
- 🚨 Alerted

- Boot or Logon Autostart Execution — 50 Actions
- Boot or Logon Initialization Scripts — 2 Actions
- Create Account — 8 Actions
- Create or Modify System Process — 9 Actions

Privilege Escalation — 218 Actions

🟥 Not Detected	119
🟩 Detected	99
🚨 Alerted	11

- Abuse Elevation Control Mechanism — 6 Actions
- Access Token Manipulation — 16 Actions
- Boot or Logon Autostart Execution — 50 Actions
- Create or Modify System Process — 9 Actions

Defense Evasion — 339 Actions

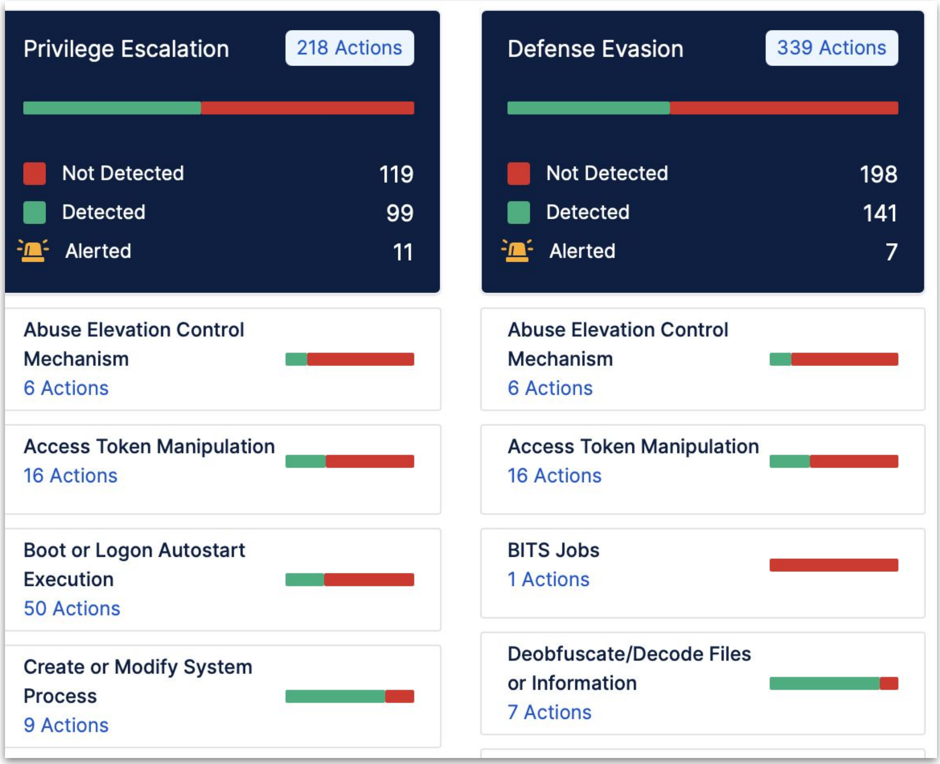
🟥 Not Detected	198
🟩 Detected	141
🚨 Alerted	7

- Abuse Elevation Control Mechanism — 6 Actions
- Access Token Manipulation — 16 Actions
- BITS Jobs — 1 Actions
- Deobfuscate/Decode Files or Information — 7 Actions

1. Threat Selection

Prioritization Idea:

Among the shortlisted threats, look for quick wins (telemetry data availability).



Privilege Escalation 218 Actions

🟥 Not Detected 119
🟩 Detected 99
🚨 Alerted 11

Abuse Elevation Control Mechanism
6 Actions

Access Token Manipulation
16 Actions

Boot or Logon Autostart Execution
50 Actions

Create or Modify System Process
9 Actions

Defense Evasion 339 Actions

🟥 Not Detected 198
🟩 Detected 141
🚨 Alerted 7

Abuse Elevation Control Mechanism
6 Actions

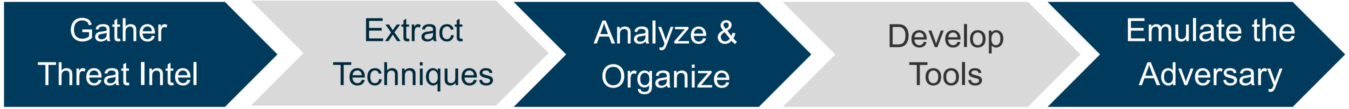
Access Token Manipulation
16 Actions

BITS Jobs
1 Actions

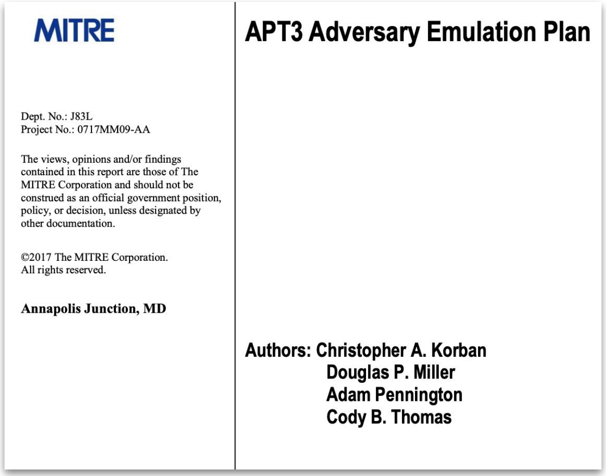
Deobfuscate/Decode Files or Information
7 Actions

2. Adversary Emulation

Create an adversary emulation plan and execute.



Adversary Emulation Process recommended by Mitre ATT&CK



MITRE

APT3 Adversary Emulation Plan

Dept. No.: J83L
Project No.: 0717MM09-AA

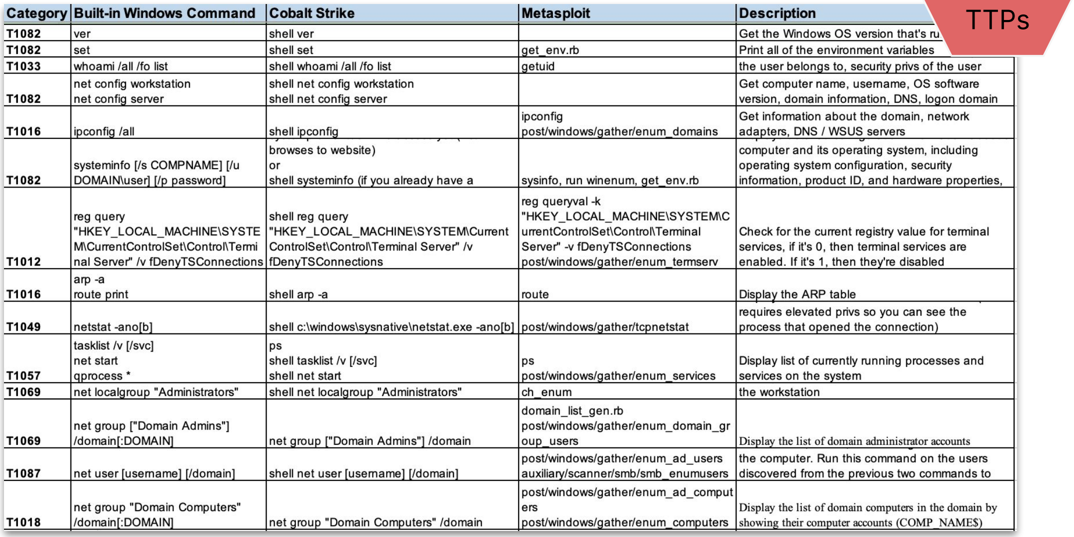
The views, opinions and/or findings contained in this report are those of The MITRE Corporation and should not be construed as an official government position, policy, or decision, unless designated by other documentation.

©2017 The MITRE Corporation.
All rights reserved.

Annapolis Junction, MD

Authors: Christopher A. Korban
Douglas P. Miller
Adam Pennington
Cody B. Thomas

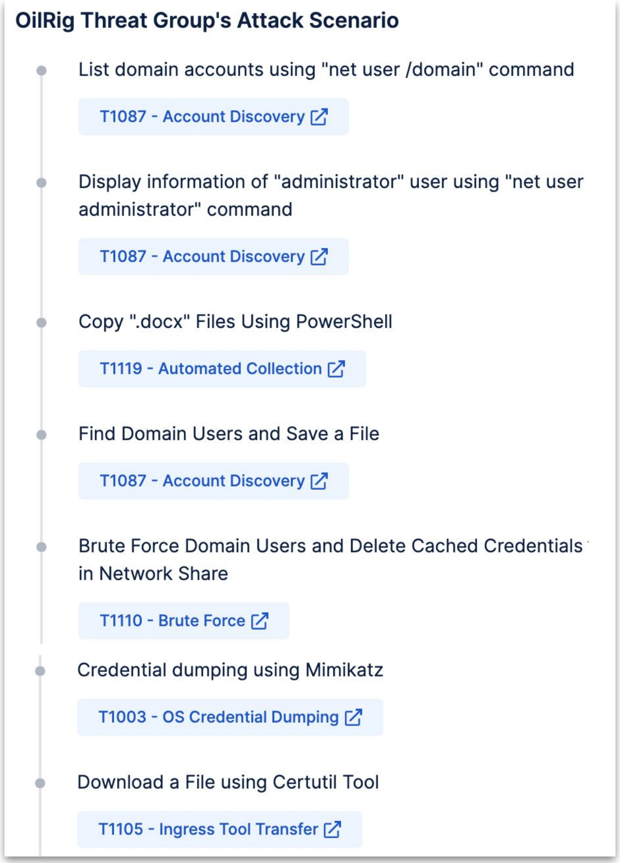
50
TTPs



Category	Built-in Windows Command	Cobalt Strike	Metasploit	Description
T1082	ver	shell ver		Get the Windows OS version that's ru
T1082	set	shell set	get_env.rb	Print all of the environment variables
T1033	whoami /all /fo list	shell whoami /all /fo list	getuid	the user belongs to, security privs of the user
T1082	net config workstation net config server	shell net config workstation shell net config server		Get computer name, username, OS software version, domain information, DNS, logon domain
T1016	ipconfig /all	shell ipconfig	ipconfig post/windows/gather/enum_domains	Get information about the domain, network adapters, DNS / WSUS servers
T1082	systeminfo [/s COMPNAME] [/u DOMAIN\user] [/p password]	browses to website) or shell systeminfo (if you already have a	sysinfo, run winenum, get_env.rb	computer and its operating system, including operating system configuration, security information, product ID, and hardware properties,
T1012	reg query "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server" /v fDenyTSConnections	shell reg query "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server" /v fDenyTSConnections	reg queryval -k "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server" -v fDenyTSConnections post/windows/gather/enum_termserv	Check for the current registry value for terminal services, if it's 0, then terminal services are enabled. If it's 1, then they're disabled
T1016	arp -a route print	shell arp -a	route	Display the ARP table
T1049	netstat -ano[b]	shell c:\windows\sysnative\netstat.exe -ano[b]	post/windows/gather/tcpnetstat	requires elevated privs so you can see the process that opened the connection)
T1057	tasklist /v [/svc] net start qprocess *	ps shell tasklist /v [/svc] shell net start	ps post/windows/gather/enum_services	Display list of currently running processes and services on the system
T1069	net localgroup "Administrators"	shell net localgroup "Administrators"	ch_enum	the workstation
T1069	net group ["Domain Admins"] /domain[:DOMAIN]	net group ["Domain Admins"] /domain	domain_list_gen.rb post/windows/gather/enum_domain_group_users	Display the list of domain administrator accounts
T1087	net user [username] [/domain]	shell net user [username] [/domain]	post/windows/gather/enum_ad_users auxiliary/scanner/smb/smb_enumusers	the computer. Run this command on the users discovered from the previous two commands to
T1018	net group "Domain Computers" /domain[:DOMAIN]	net group "Domain Computers" /domain	post/windows/gather/enum_ad_computers post/windows/gather/enum_computers	Display the list of domain computers in the domain by showing their computer accounts (COMP_NAME$)

2. Adversary Emulation

Leveraging **1000+** adversary emulation plans within **Picus Threat Library**



OilRig Threat Group's Attack Scenario

- List domain accounts using "net user /domain" command
  T1087 - Account Discovery
- Display information of "administrator" user using "net user administrator" command
  T1087 - Account Discovery
- Copy ".docx" Files Using PowerShell
  T1119 - Automated Collection
- Find Domain Users and Save a File
  T1087 - Account Discovery
- Brute Force Domain Users and Delete Cached Credentials in Network Share
  T1110 - Brute Force
- Credential dumping using Mimikatz
  T1003 - OS Credential Dumping
- Download a File using Certutil Tool
  T1105 - Ingress Tool Transfer

3. Log Validation

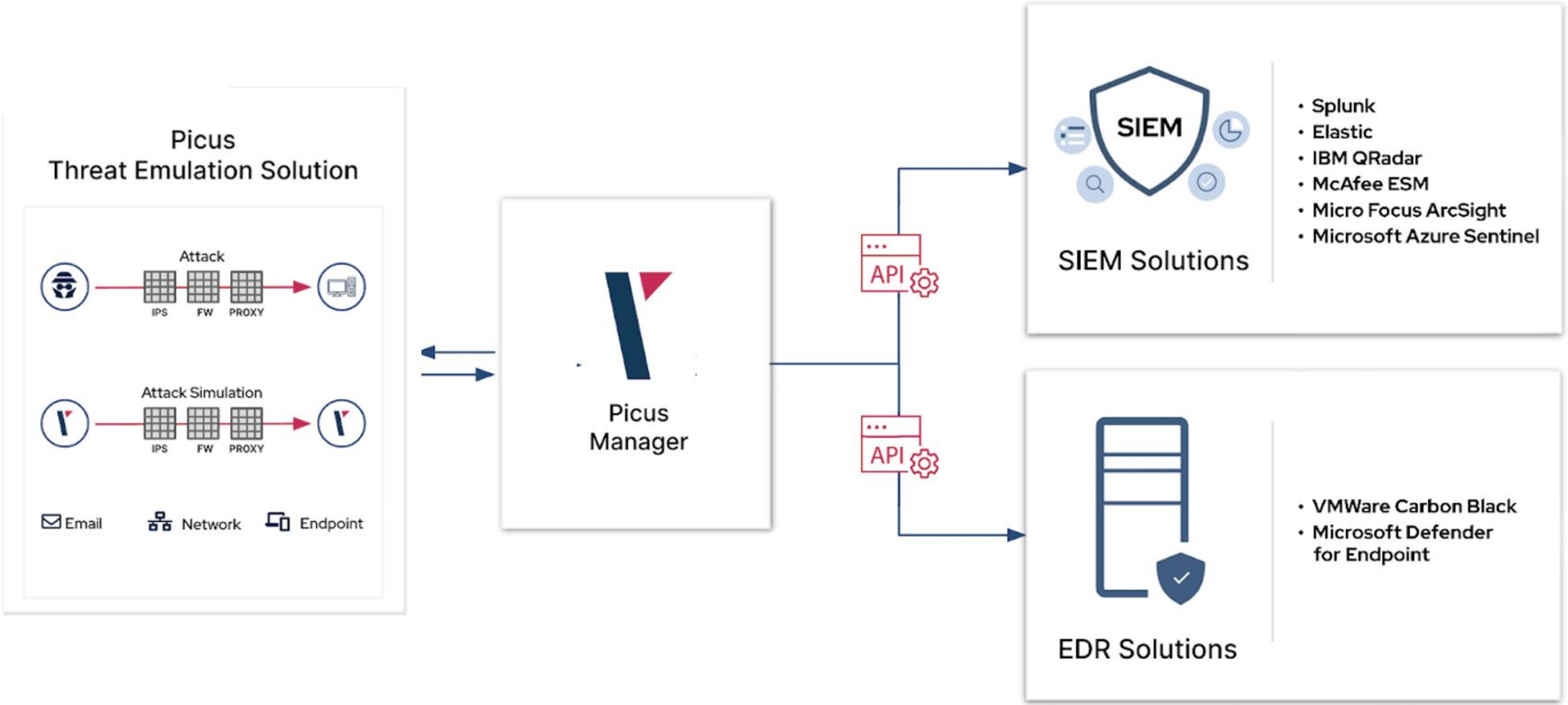
What We Need

- Right logs at the right verbose level
- Proper parsing and storage by the SIEM

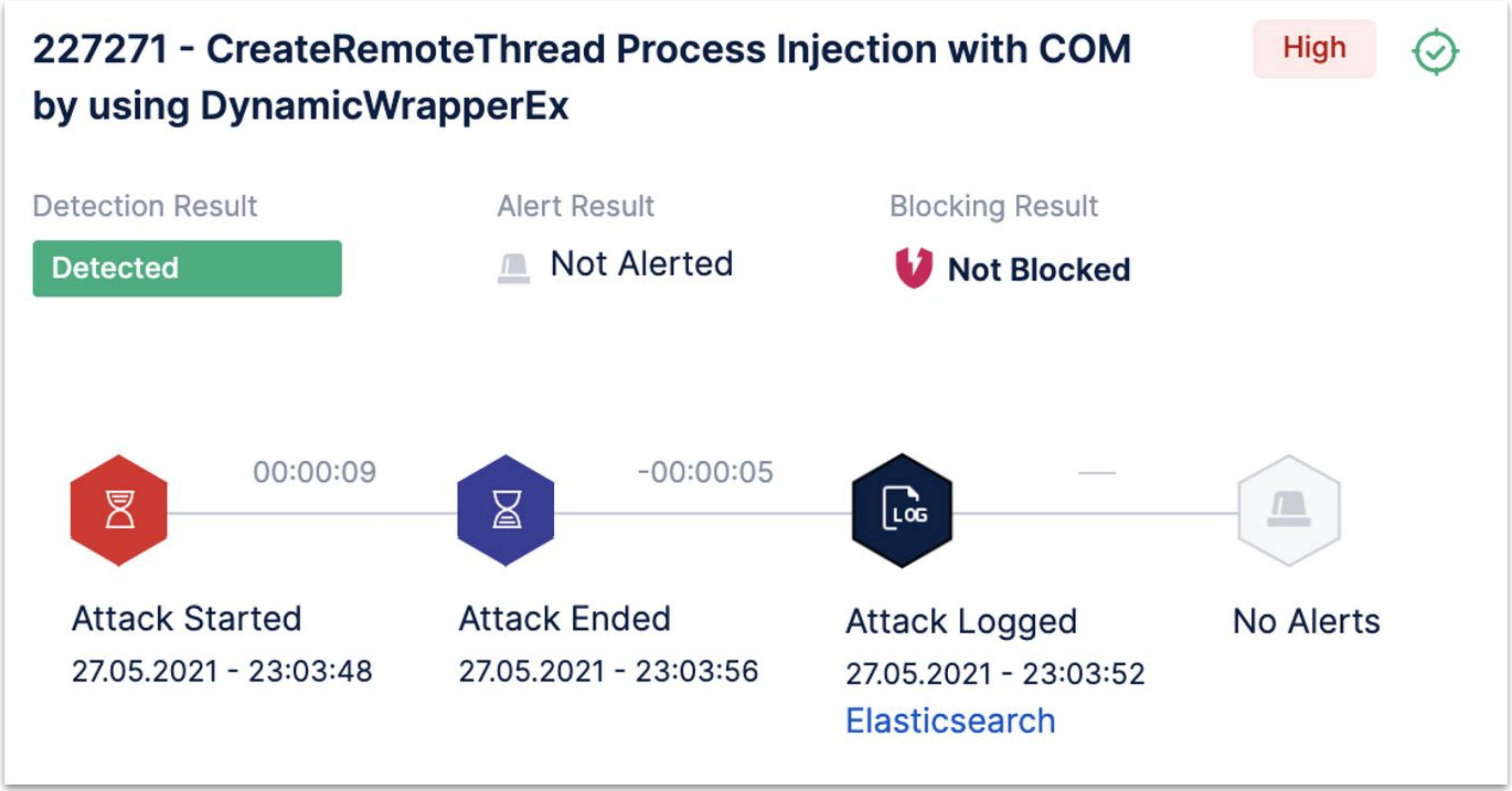
How to Validate

- Define expected data-sources
- Identify logs from expected data-sources
- Check required logs against simulated attacks

3. Log Validation



3. Log Validation



227271 - CreateRemoteThread Process Injection with COM by using DynamicWrapperEx

High

Detection Result: Detected

Alert Result: Not Alerted

Blocking Result: Not Blocked

00:00:09

-00:00:05

—

Attack Started
27.05.2021 - 23:03:48

Attack Ended
27.05.2021 - 23:03:56

Attack Logged
27.05.2021 - 23:03:52
Elasticsearch

No Alerts

4. Alert Validation

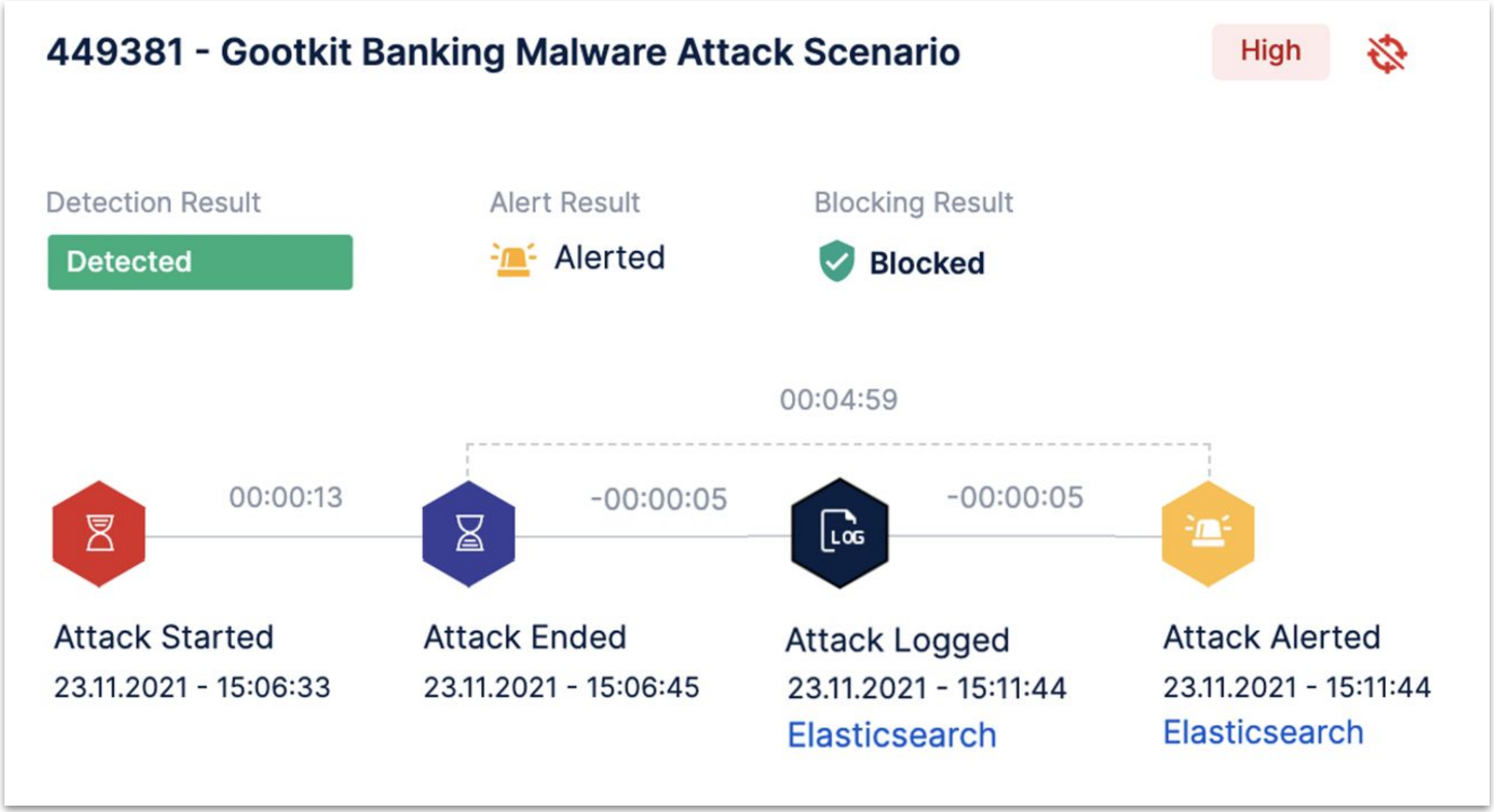
What We Need

- Right rules to manage the alert fatigue
- Being up to date for new TTPs of that threat

How to Validate

- Query alerts for the simulation agents
- Check required alerts against simulated attacks

4. Alert Validation



449381 - Gootkit Banking Malware Attack Scenario

High

Detection Result: Detected

Alert Result: Alerted

Blocking Result: Blocked

00:04:59

00:00:13

-00:00:05

-00:00:05

Attack Started
23.11.2021 - 15:06:33

Attack Ended
23.11.2021 - 15:06:45

Attack Logged
23.11.2021 - 15:11:44
Elasticsearch

Attack Alerted
23.11.2021 - 15:11:44
Elasticsearch

5. Continuous Improvement

Challenges

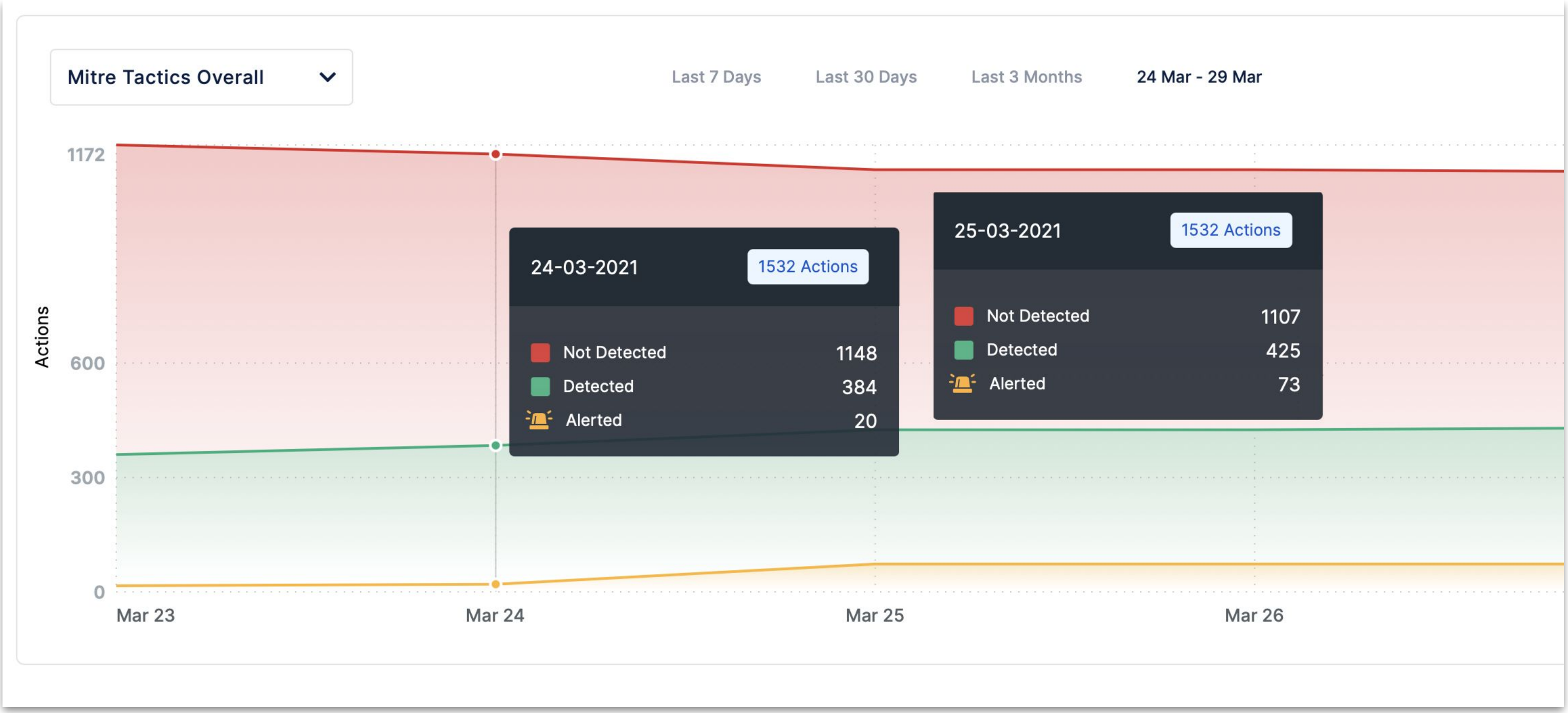
- Configuration drift
- Ever-changing threat landscape
- Managing the complexity of security tools
- Communication problems between the involved parties

How to improve

- Leverage automation opportunities for use-case development
- Streamline detection development process (CI/CD)
- Challenging ourselves against new threats

5. Continuous Improvement

Monitor your threat coverage against sudden drops and improvements.



Mitre Tactics Overall

Last 7 Days | Last 30 Days | Last 3 Months | 24 Mar - 29 Mar

24-03-2021 — 1532 Actions

🟥 Not Detected	1148
🟩 Detected	384
🚨 Alerted	20

25-03-2021 — 1532 Actions

🟥 Not Detected	1107
🟩 Detected	425
🚨 Alerted	73

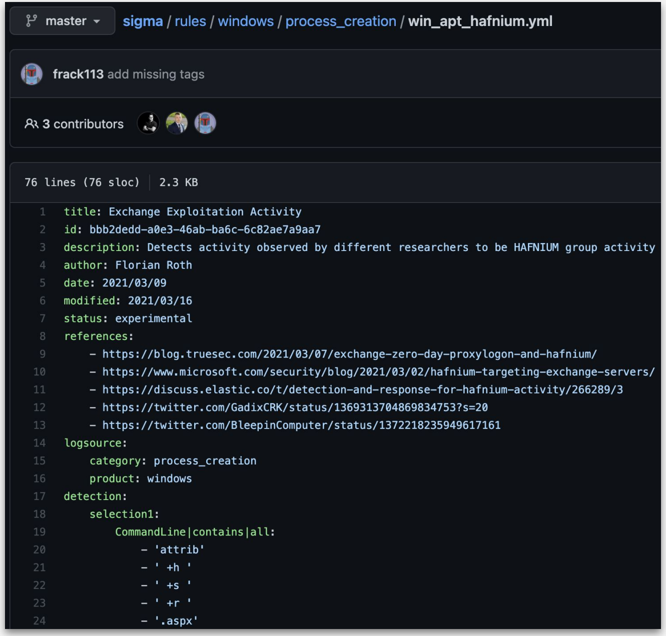
How to Content Development Under Limited-Resources?

SOC teams may leverage 3rd party contents to accelerate their detection content development process.

Lots of alert/correlation rule libraries available:

- Repository of your SIEM/EDR vendor
- Open-source libraries/projects
- Content libraries (ie Picus Mitigation Library)

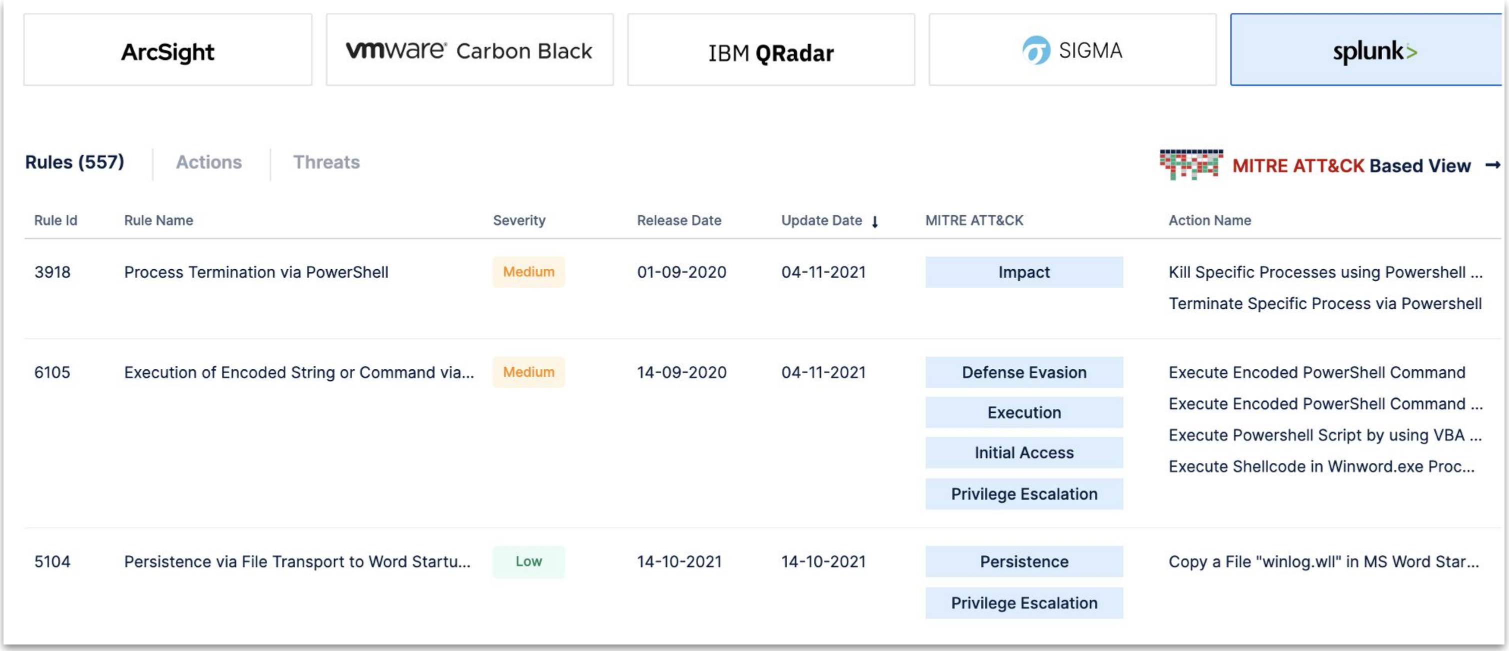
Either we implement them directly or use them as templates to craft your detection rule, these rules need to be validated too!


master
sigma / rules / windows / process_creation / win_apt_hafnium.yml
frack113 add missing tags
3 contributors
76 lines (76 sloc) | 2.3 KB

```
title: Exchange Exploitation Activity
id: bbb2dedd-a0e3-46ab-ba6c-6c82ae7a9aa7
description: Detects activity observed by different researchers to be HAFNIUM group activity
author: Florian Roth
date: 2021/03/09
modified: 2021/03/16
status: experimental
references:
    - https://blog.truesec.com/2021/03/07/exchange-zero-day-proxylogon-and-hafnium/
    - https://www.microsoft.com/security/blog/2021/03/02/hafnium-targeting-exchange-servers/
    - https://discuss.elastic.co/t/detection-and-response-for-hafnium-activity/266289/3
    - https://twitter.com/GadixCRK/status/1369313704869834753?s=20
    - https://twitter.com/BleepinComputer/status/1372218235949617161
logsource:
    category: process_creation
    product: windows
detection:
    selection1:
        CommandLine|contains|all:
            - 'attrib'
            - ' +h '
            - ' +s '
            - ' +r '
            - '.aspx'
```

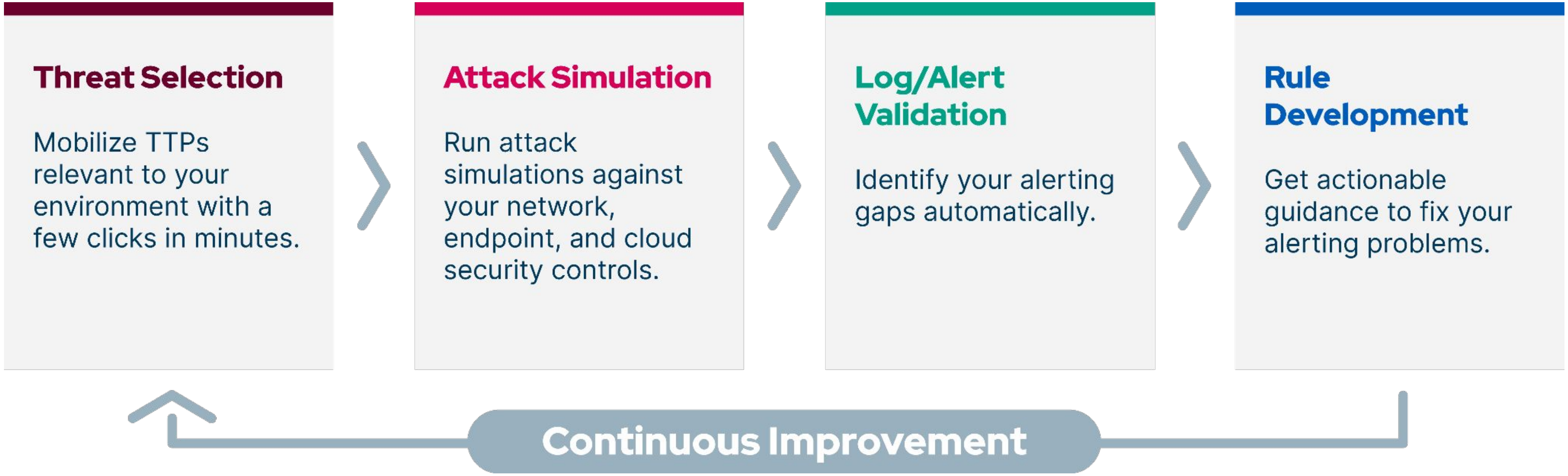
Picus Detection Content Library

800+ detection rules (+log source recommendations), mapped to TTP’s and ATT&CK category mappings.



Rule Id	Rule Name	Severity	Release Date	Update Date ↓	MITRE ATT&CK	Action Name
3918	Process Termination via PowerShell	Medium	01-09-2020	04-11-2021	Impact	Kill Specific Processes using Powershell ... Terminate Specific Process via Powershell
6105	Execution of Encoded String or Command via...	Medium	14-09-2020	04-11-2021	Defense Evasion Execution Initial Access Privilege Escalation	Execute Encoded PowerShell Command Execute Encoded PowerShell Command ... Execute Powershell Script by using VBA ... Execute Shellcode in Winword.exe Proc...
5104	Persistence via File Transport to Word Startu...	Low	14-10-2021	14-10-2021	Persistence Privilege Escalation	Copy a File "winlog.wll" in MS Word Star...

Detection Development Life-Cycle Leveraging Attack Simulation



Threat Selection

Mobilize TTPs relevant to your environment with a few clicks in minutes.

Attack Simulation

Run attack simulations against your network, endpoint, and cloud security controls.

Log/Alert Validation

Identify your alerting gaps automatically.

Rule Development

Get actionable guidance to fix your alerting problems.

Continuous Improvement

Summary

- Detection content is critical to detection success
- Detection content creation and management requires a structured process
- Attack Simulation enables the DDLC by supporting multiple phases of the process, from identification to measurement
- Pre-built detection content and tests accelerates time-to-value and reduces implementation costs

THANK YOU!