SIEM/SOAR TESTBED IMPLEMENTATION FOR USE IN UNDERGRAD AND POSTGRAD MODULES

Final Project Report

Name: Darragh Murphy
Student ID: C00135047
Project Supervisor: James Egan

24/04/2022

Abstract

This document will provide an overview of the project implementation, a breakdown of the created lab exercises and the takeaways that the author has gained throughout the course of the year.

Table of Contents

Abstract i

Introduction 1

Project Description 2

Lab Exercises 2

Lab Exercises Summary 8

Lab 1: Obfuscated Files or Information 8

Lab 2: Command and Scripting Interpreter 10

Lab 3: Windows Management Instrumentation 12

Lab 4: Ingress Tool Transfer 14

Lab 5: Signed Binary Proxy Execution 16

The Test Environment 18

Project Screenshots 22

Learning Outcomes 27

Personal 27

Technical 27

Project Review 30

Problems Encountered 30

Design Differences 31

Further Research 32

Future Projects 32

Recommendations 32

Acknowledgements 34

References 35

Introduction

The purpose of the Final Project Report Document is to discuss the steps taken to implement an environment where students from the Year 2 Incident Handling Project module and the upcoming MSc. Advanced Incident Response module can gain experience analyzing security event logs that are generated by students performing various tests within that environment.

The project's major goal was to demonstrate that acquiring the skills required in log analysis and getting experience via the use of a SIEM tool are useful skills to learn for students who would soon be joining the workforce.

The lab exercises were developed based on the most common MITRE ATTACK framework techniques observed being used by adversaries and documented in Red Canary's 2022 Threat Detection Report.[1] The lab exercises involve performing research on a specific technique, using Atomic Red Team's atomics tests for that technique to perform testing, and finally using Splunk to search for the logs associated with the test performed.

The implementation of the project is achieved through a deployment of Chris Long's Detection Lab[2], and this comes pre-configured with a host of security tools. A client Window's 10 virtual machine is configured to forward logs to Splunk which is hosted on an Ubuntu Server virtual machine.

In addition to discussing the steps taken to create the lab exercises and implementing the virtual machines, this document will also cover any issues I encountered throughout the year, achievements made and what I learned during the process from start to finish. This document will also contain any recommendations and will outline what will be handed over to the lecturers teaching the related modules.

[1] (Red Canary, 2022)

[2] (Long, 2022)

Project Description

The main objective of the project was to test an environment that could be used to generate traffic and alerts for analysis by undergrad and postgrad students in the Year 2 Incident Handling Project module and the upcoming MSc. Advanced Incident Response module.

Over the course of my research, I discovered the Atomic Red Team[3] GitHub repository. This is a library of tests that are mapped to the techniques contained in the MITRE ATTACK framework. This library is extensive and the tests can be reproduced consistently. the ease of executing these tests would become a building block for the lab exercises that were developed.

Throughout the research phase, I also discovered the Detection Lab. The Detection Lab is a repository containing scripts that allow the user to quickly spin up four virtual machines that are configured to be insecure and designed for testing purposes. The machines come packaged with various security software including Splunk, Suricata and Zeek and they are preconfigured to forward log data to Splunk for storage and analysis.

The Detection Lab comes with many tools installed; however, the primary focus of my project was to generate traffic that can be analysed in Splunk, the SIEM that comes pre-configured with the lab. The Atomic Red Team's Atomic Tests repository is installed on the Windows 10 machine during the configuration and executing these tests will generate log events that can be analyzed by students.

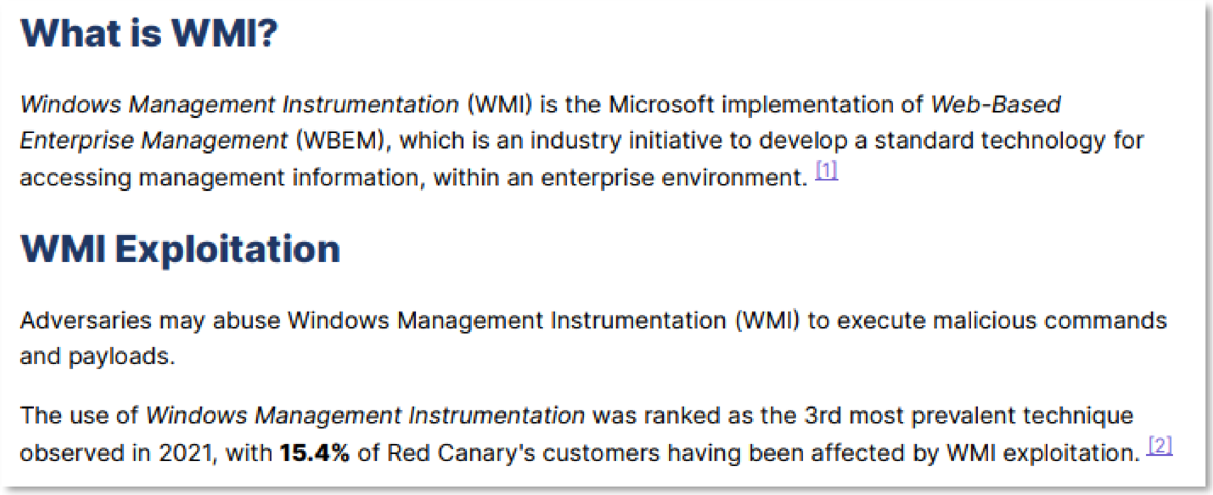
Lab Exercises

The MITRE ATTACK is a database that contains the tactics, techniques, and procedures that adversaries use. Gaining knowledge about these TTPs is valuable to a student's career, and I felt that this would be an excellent starting point for the lab exercises.

On March 22nd Red Canary published their 2022 Threat Detection report. This report contained a list of the top techniques observed in 2021.[4] This became a valuable resource when creating the lab exercises.

[3] (Atomic Red Team, 2022)
[4] (Red Canary, 2022)

The lab exercises were created in such a way that the students are provided with some basic theory information about a technique from the MITRE ATTACK framework. The next section contains a series of questions that provide topics surround the techniques that the students will be required to research before the final section. The final section consists of performing the tests using the Invoke-AtomicTest PowerShell module and then searching for the log events in Splunk.



What is WMI?

Windows Management Instrumentation (WMI) is the Microsoft implementation of *Web-Based Enterprise Management* (WBEM), which is an industry initiative to develop a standard technology for accessing management information, within an enterprise environment. [1]

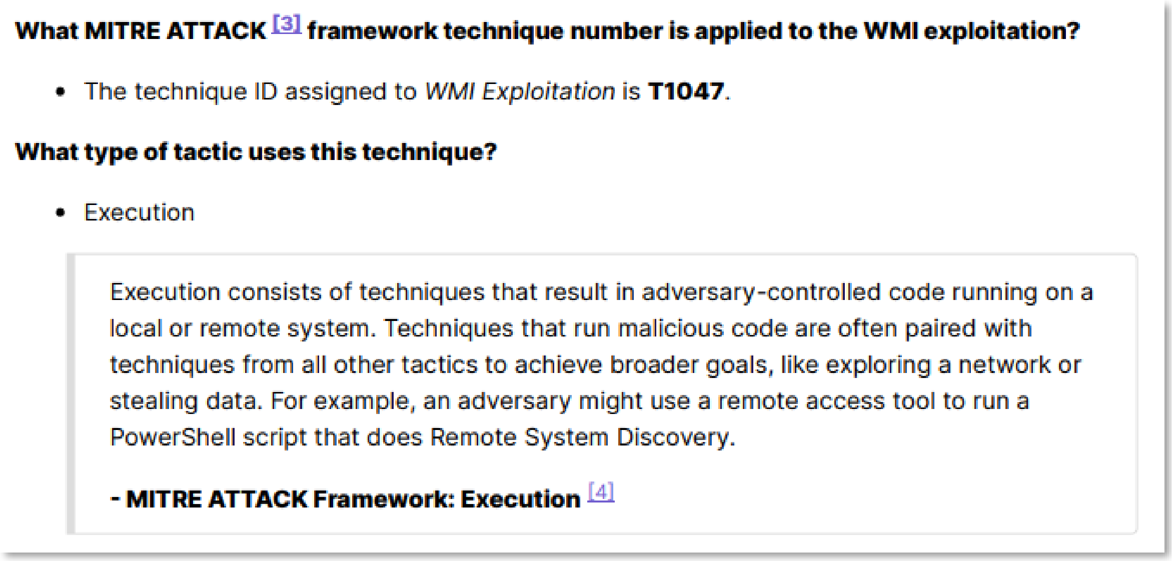
WMI Exploitation

Adversaries may abuse Windows Management Instrumentation (WMI) to execute malicious commands and payloads.

The use of *Windows Management Instrumentation* was ranked as the 3rd most prevalent technique observed in 2021, with **15.4%** of Red Canary's customers having been affected by WMI exploitation. [2]

Figure 1: Theory Introduction relating to WMI

The screenshot below contains an example of the types of questions that are asked in the lab exercises. The questions start with one that will bring them to the MITRE ATTACK website and here they will have the database available to them to perform their research. That website contains a wealth of information and references to specific examples of where the techniques have been used. The answers are provided in the exercises; however, the lecturer can choose what information to redact before passing the assignment to the students.



What MITRE ATTACK [3] framework technique number is applied to the WMI exploitation?

- The technique ID assigned to *WMI Exploitation* is **T1047**.

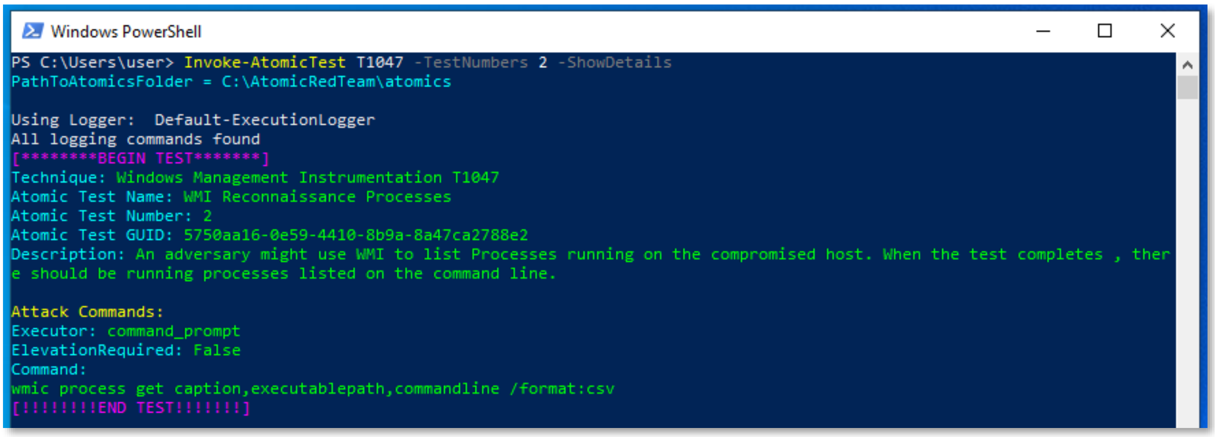
What type of tactic uses this technique?

- Execution

> Execution consists of techniques that result in adversary-controlled code running on a local or remote system. Techniques that run malicious code are often paired with techniques from all other tactics to achieve broader goals, like exploring a network or stealing data. For example, an adversary might use a remote access tool to run a PowerShell script that does Remote System Discovery.
>
> **- MITRE ATTACK Framework: Execution** [4]

Figure 2: Two types of the questions to prompt research.

Before executing a test, it is recommended to use the *-ShowDetails* switch. This enables the student to see a description of the test and to view what commands the test will execute. These details will help the students when it comes to searching for logs. A link is also provided to the relevant Atomic Test's GitHub page. The students will be able to use this for further research.

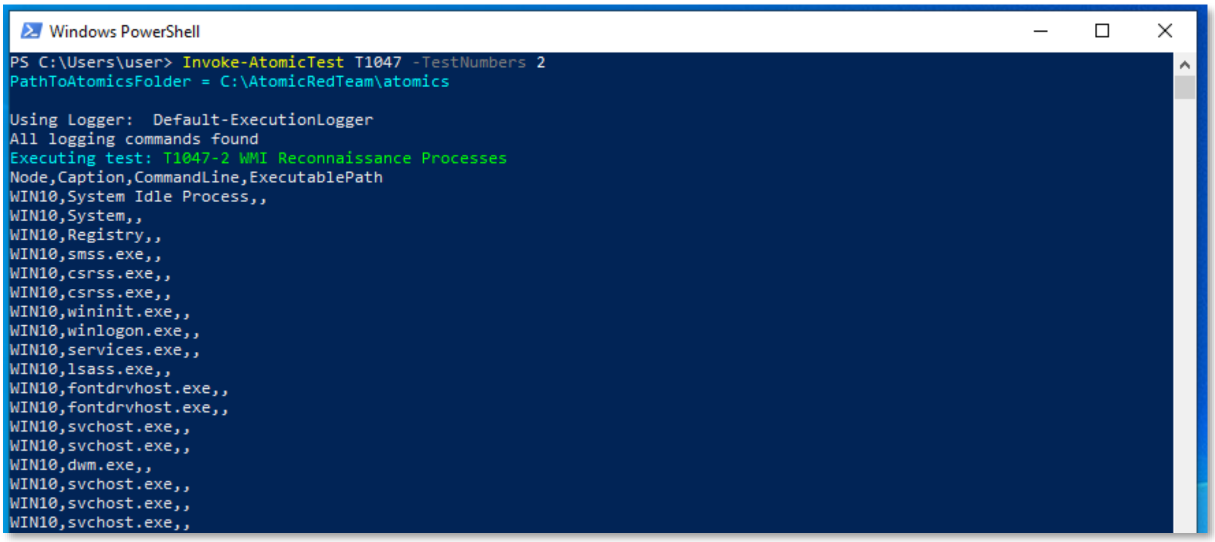


```
PS C:\Users\user> Invoke-AtomicTest T1047 -TestNumbers 2 -ShowDetails
PathToAtomicsFolder = C:\AtomicRedTeam\atomics

Using Logger:  Default-ExecutionLogger
All logging commands found
[********BEGIN TEST*******]
Technique: Windows Management Instrumentation T1047
Atomic Test Name: WMI Reconnaissance Processes
Atomic Test Number: 2
Atomic Test GUID: 5750aa16-0e59-4410-8b9a-8a47ca2788e2
Description: An adversary might use WMI to list Processes running on the compromised host. When the test completes , ther
e should be running processes listed on the command line.

Attack Commands:
Executor: command_prompt
ElevationRequired: False
Command:
wmic process get caption,executablepath,commandline /format:csv
[!!!!!!!!END TEST!!!!!!!]
```

Figure 3: The -ShowDetails Switch displays the test details.

In the screenshot below I have executed the Atomic Test T1047.002, which leverages the Windows Management Instrumentation to list the running processes on a target machine. WMI exploitation was ranked 3rd in the top 10 techniques observed. WMI can be used to execute scripts and processes, perform reconnaissance, and move laterally within a target environment.

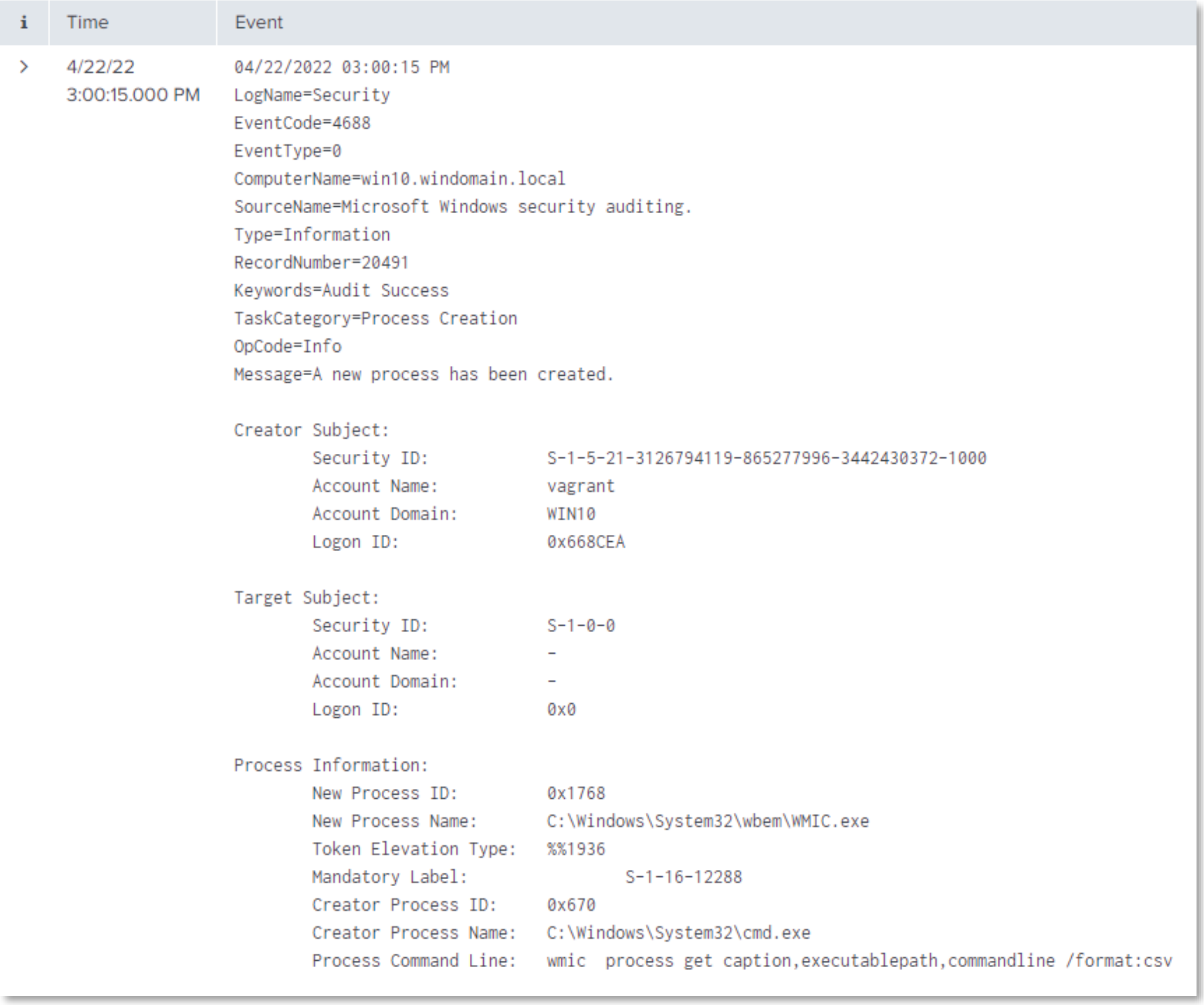
As per the screenshot below, the test was executed successfully and a list of the running processes is displayed on the console.



```
PS C:\Users\user> Invoke-AtomicTest T1047 -TestNumbers 2
PathToAtomicsFolder = C:\AtomicRedTeam\atomics

Using Logger:  Default-ExecutionLogger
All logging commands found
Executing test: T1047-2 WMI Reconnaissance Processes
Node,Caption,CommandLine,ExecutablePath
WIN10,System Idle Process,,
WIN10,System,,
WIN10,Registry,,
WIN10,smss.exe,,
WIN10,csrss.exe,,
WIN10,csrss.exe,,
WIN10,wininit.exe,,
WIN10,winlogon.exe,,
WIN10,services.exe,,
WIN10,lsass.exe,,
WIN10,fontdrvhost.exe,,
WIN10,fontdrvhost.exe,,
WIN10,svchost.exe,,
WIN10,svchost.exe,,
WIN10,dwm.exe,,
WIN10,svchost.exe,,
WIN10,svchost.exe,,
WIN10,svchost.exe,,
```

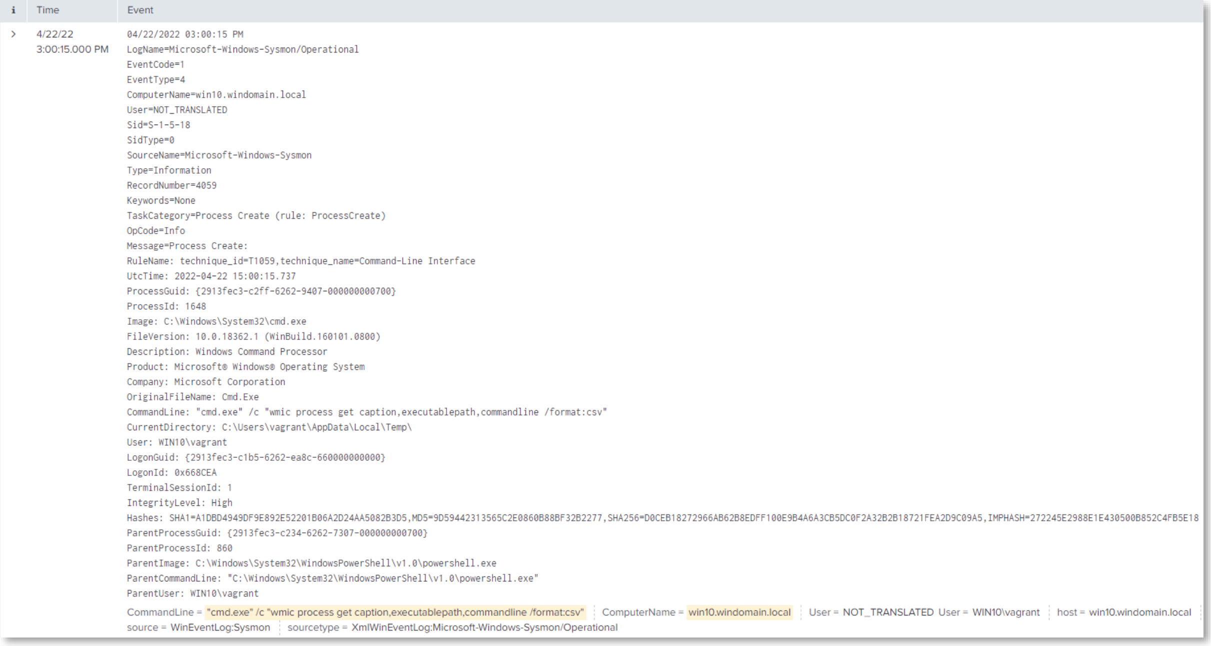
Figure 4: Executing the test

In the screenshots below, we can see the valuable information that was gathered in just two event logs that were generated. This information is valuable. The data displayed in the logs may vary, depending how the SIEM is configured but the core important data will be the same.



```
i   Time                 Event
>   4/22/22              04/22/2022 03:00:15 PM
    3:00:15.000 PM       LogName=Security
                         EventCode=4688
                         EventType=0
                         ComputerName=win10.windomain.local
                         SourceName=Microsoft Windows security auditing.
                         Type=Information
                         RecordNumber=20491
                         Keywords=Audit Success
                         TaskCategory=Process Creation
                         OpCode=Info
                         Message=A new process has been created.

                         Creator Subject:
                                 Security ID:            S-1-5-21-3126794119-865277996-3442430372-1000
                                 Account Name:           vagrant
                                 Account Domain:         WIN10
                                 Logon ID:               0x668CEA

                         Target Subject:
                                 Security ID:            S-1-0-0
                                 Account Name:           -
                                 Account Domain:         -
                                 Logon ID:               0x0

                         Process Information:
                                 New Process ID:         0x1768
                                 New Process Name:       C:\Windows\System32\wbem\WMIC.exe
                                 Token Elevation Type:   %%1936
                                 Mandatory Label:                S-1-16-12288
                                 Creator Process ID:     0x670
                                 Creator Process Name:   C:\Windows\System32\cmd.exe
                                 Process Command Line:   wmic  process get caption,executablepath,commandline /format:csv
```

Figure 5: Windows Event Log in Splunk



```
i   Time                 Event
>   4/22/22              04/22/2022 03:00:15 PM
    3:00:15.000 PM       LogName=Microsoft-Windows-Sysmon/Operational
                         EventCode=1
                         EventType=4
                         ComputerName=win10.windomain.local
                         User=NOT_TRANSLATED
                         Sid=S-1-5-18
                         SidType=0
                         SourceName=Microsoft-Windows-Sysmon
                         Type=Information
                         RecordNumber=4059
                         Keywords=None
                         TaskCategory=Process Create (rule: ProcessCreate)
                         OpCode=Info
                         Message=Process Create:
                         RuleName: technique_id=T1059,technique_name=Command-Line Interface
                         UtcTime: 2022-04-22 15:00:15.737
                         ProcessGuid: {2913fec3-c2ff-6262-9407-000000000700}
                         ProcessId: 1648
                         Image: C:\Windows\System32\cmd.exe
                         FileVersion: 10.0.18362.1 (WinBuild.160101.0800)
                         Description: Windows Command Processor
                         Product: Microsoft® Windows® Operating System
                         Company: Microsoft Corporation
                         OriginalFileName: Cmd.Exe
                         CommandLine: "cmd.exe" /c "wmic process get caption,executablepath,commandline /format:csv"
                         CurrentDirectory: C:\Users\vagrant\AppData\Local\Temp\
                         User: WIN10\vagrant
                         LogonGuid: {2913fec3-c1b5-6262-ea8c-660000000000}
                         LogonId: 0x668CEA
                         TerminalSessionId: 1
                         IntegrityLevel: High
                         Hashes: SHA1=A1DBD4949DF9E892E52201B06A2D24AA5082B3D5,MD5=9D59442313565C2E0860B88BF32B2277,SHA256=D0CEB18272966AB62B8EDFF100E9B4A6A3CB5DC0F2A32B2B18721FEA2D9C09A5,IMPHASH=272245E2988E1E430500B852C4FB5E18
                         ParentProcessGuid: {2913fec3-c234-6262-7307-000000000700}
                         ParentProcessId: 860
                         ParentImage: C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe
                         ParentCommandLine: "C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe"
                         ParentUser: WIN10\vagrant
                         CommandLine = "cmd.exe" /c "wmic process get caption,executablepath,commandline /format:csv"  ComputerName = win10.windomain.local  User = NOT_TRANSLATED User = WIN10\vagrant  host = win10.windomain.local
                         source = WinEventLog:Sysmon  sourcetype = XmlWinEventLog:Microsoft-Windows-Sysmon/Operational
```

Figure 6: Sysmon Event Log in Splunk

An analyst in a SOC will use logs such as the one displayed in order to determine the series of events that triggered an alert. The events can help identify if the alert is a false positive, or if something malicious has occurred. Some of the most important information that a user can use to determine the legitimacy are as follows:

- Timestamps
- Event Types
- Event Codes
- Command Line
- User
- Host Machine
- IP Address
- File Hash
- Parent Process
- File Path

Splunk's SIEM tool currently has a market share of 63.76%, with over 14,000 customers using this product.[5] This means that most, if not all graduates will eventually use this tool or one of its competitors over the course of their careers. From my own experience, the skills in one tool are easily transferrable to the next. This makes learning these skills invaluable.

The exercises may be completed over a number of weeks and will provide ample opportunities for a student to research these techniques and to gain valuable experience using Splunk to build search queries in order to locate the events.

[5] (Slintel, 2022)

Lab Exercises Summary

Lab 1: Obfuscated Files or Information

Description:

Adversaries may attempt to make an executable or file difficult to discover or analyze by encrypting, encoding, or otherwise obfuscating its contents on the system or in transit. This is common behaviour that can be used across different platforms and the network to evade defences.[6]

Research Questions:

The students will be asked to answer and research a series of questions relating to the topic that the lab exercise is based on.

1. What MITRE ATTACK framework technique ID is assigned to this Obfuscated Files or Information?
2. What MITRE ATTACK framework tactic uses this technique?
3. Why do malicious actors use Obfuscated Files or Information?
4. What can Malicious Actors use Obfuscated Files or Information for?
5. Name and research groups and instances that have used this technique.
6. What are some mitigations against this type of technique?
7. How can this technique be detected?
8. What logs can be used to identify this technique?

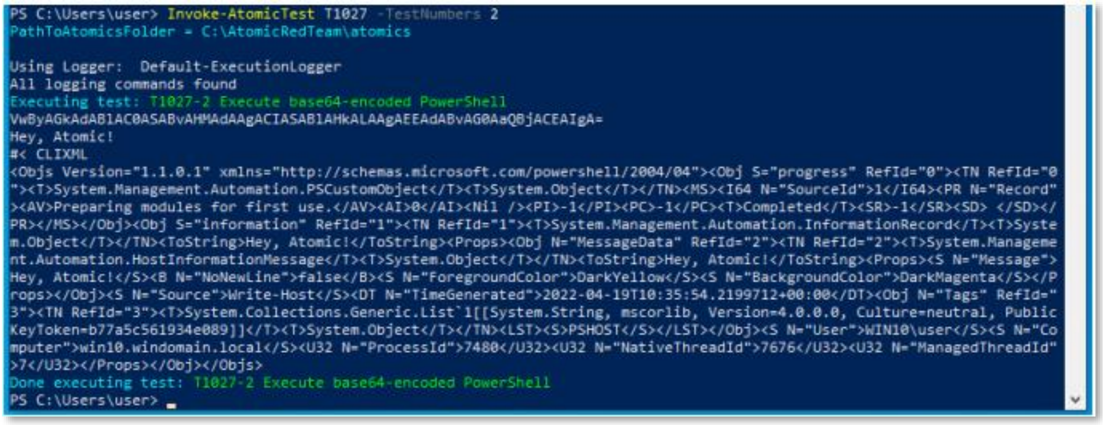
Practical:

The practical element of the lab exercise requires the student to perform some testing using the Atomic Red Team atomics tests. The students may execute any tests related to this technique. Each test is divided into two sections: Executing the Test and Searching for the Logs.

Test #2 - Execute base64-encoded PowerShell

This test shows how code may be encoded, in the hopes of avoiding detection. The code is then executed. Successful execution of this test should display 'Hey, Atomic!'.

[6] (MITRE ATTACK, 2021)

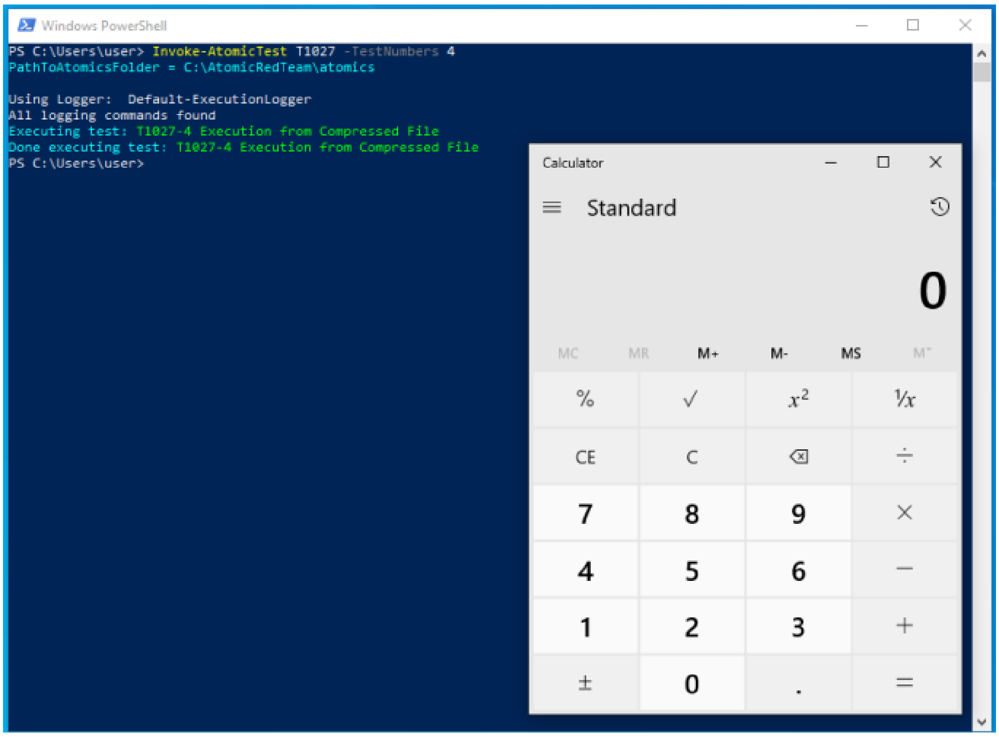


```
PS C:\Users\user> Invoke-AtomicTest T1027 -TestNumbers 2
PathToAtomicsFolder = C:\AtomicRedTeam\atomics

Using Logger:  Default-ExecutionLogger
All logging commands found
Executing test: T1027-2 Execute base64-encoded PowerShell
VwByAGkAdABlAC0ASABvAHMAdAAgACIASABlAHkALAAgAEEAdABvAG0AaQBjACEAIgA=
Hey, Atomic!
#< CLIXML
<Objs Version="1.1.0.1" xmlns="http://schemas.microsoft.com/powershell/2004/04"><Obj S="progress" RefId="0"><TN RefId="0
"><T>System.Management.Automation.PSCustomObject</T><T>System.Object</T></TN><MS><I64 N="SourceId">1</I64><PR N="Record"
><AV>Preparing modules for first use.</AV><AI>0</AI><Nil /><PI>-1</PI><PC>-1</PC><T>Completed</T><SR>-1</SR><SD> </SD></
PR></MS></Obj><Obj S="information" RefId="1"><TN RefId="1"><T>System.Management.Automation.InformationRecord</T><T>Syste
m.Object</T></TN><ToString>Hey, Atomic!</ToString><Props><Obj N="MessageData" RefId="2"><TN RefId="2"><T>System.Manageme
nt.Automation.HostInformationMessage</T><T>System.Object</T></TN><ToString>Hey, Atomic!</ToString><Props><S N="Message">
Hey, Atomic!</S><B N="NoNewLine">false</B><S N="ForegroundColor">DarkYellow</S><S N="BackgroundColor">DarkMagenta</S></P
rops></Obj><S N="Source">Write-Host</S><DT N="TimeGenerated">2022-04-19T10:35:54.2199712+00:00</DT><Obj N="Tags" RefId="
3"><TN RefId="3"><T>System.Collections.Generic.List`1[[System.String, mscorlib, Version=4.0.0.0, Culture=neutral, Public
KeyToken=b77a5c561934e089]]</T><T>System.Object</T></TN><LST><S>PSHOST</S></LST></Obj><S N="User">WIN10\user</S><S N="Co
mputer">win10.windomain.local</S><U32 N="ProcessId">7480</U32><U32 N="NativeThreadId">7676</U32><U32 N="ManagedThreadId"
>7</U32></Props></Obj></Objs>
Done executing test: T1027-2 Execute base64-encoded PowerShell
PS C:\Users\user> _
```

Figure 7: T1027 Test #2

Test #4 - Execution from Compressed File

The purpose of this test is to demonstrate how an adversary may run an executable from a compressed folder. The test itself should launch *calc.exe*.



```
PS C:\Users\user> Invoke-AtomicTest T1027 -TestNumbers 4
PathToAtomicsFolder = C:\AtomicRedTeam\atomics

Using Logger:  Default-ExecutionLogger
All logging commands found
Executing test: T1027-4 Execution from Compressed File
Done executing test: T1027-4 Execution from Compressed File
PS C:\Users\user>
```

Figure 8: T1027 Test #4

Technologies Used:

PowerShell, Atomic Red Team, Splunk, MITRE ATTACK

Learning Outcomes:

- The student will learn about the MITRE ATTACK technique T1027.
- The student will learn about log events that are triggered by this technique.
- The student will learn how to use Splunk Search and Reporting.
- The student will gain valuable experience analysing the log data returned in searches.
- Learn about the information included in logs and how it can determine if the activity is legitimate or malicious.

Lab 2: Command and Scripting Interpreter

Description:

Adversaries may abuse command and script interpreters to execute commands, scripts, or binaries. These interfaces and languages provide ways of interacting with computer systems and are a common feature across many different platforms. Most systems come with some built-in command-line interface and scripting capabilities, for example, macOS and Linux distributions include some *flavour* of Unix Shell while Windows installations include the Windows Command Shell and PowerShell.[7]

Research Questions:

The students will be asked to answer and research a series of questions relating to the topic that the lab exercise is based on.

1. What MITRE ATTACK framework technique ID is assigned to the Command and Scripting Interpreter?
2. What MITRE ATTACK framework tactic uses this technique?
3. Why do malicious actors use the Command and Scripting Interpreter?
4. What can Malicious Actors use the Command and Scripting Interpreter for?
5. Name and research groups and instances that have used this technique.
6. What are some mitigations against this type of technique?
7. How can this technique be detected?
8. What logs can be used to identify this technique?

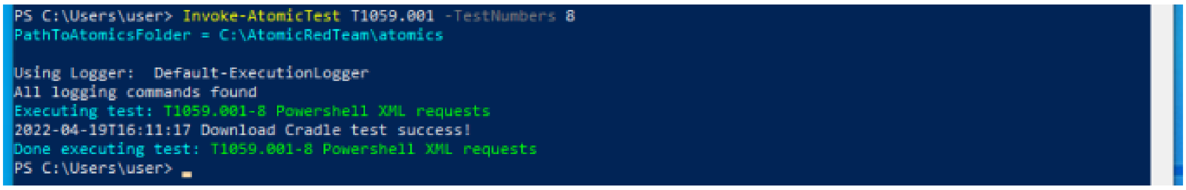
[7] (MITRE ATTACK, 2021)

Practical:

The practical element of the lab exercise requires the student to perform some testing using the Atomic Red Team atomics tests. This lab is focused on two sub-techniques: PowerShell and Command Prompt. The students may execute any tests related to these techniques. Each test is divided into two sections: Executing the Test and Searching for the Logs.

T1059.001 Test #8 - PowerShell XML Requests

This test covers a PowerShell method used to download and execute an XML from the internet. Upon a successful execution of the test, this should display 'Download Cradle test success!'.

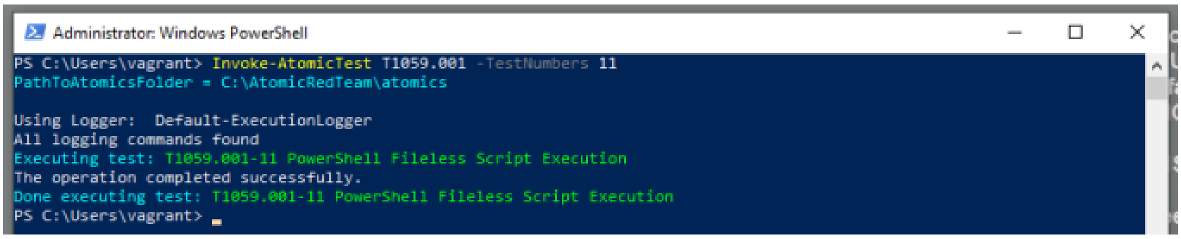


```
PS C:\Users\user> Invoke-AtomicTest T1059.001 -TestNumbers 8
PathToAtomicsFolder = C:\AtomicRedTeam\atomics

Using Logger:  Default-ExecutionLogger
All logging commands found
Executing test: T1059.001-8 Powershell XML requests
2022-04-19T16:11:17 Download Cradle test success!
Done executing test: T1059.001-8 Powershell XML requests
PS C:\Users\user> _
```

Figure 9: T1059.001 Test #8

T1059.001 Test #11 - PowerShell Fileless Script Execution

Execution of a PowerShell payload from the Windows Registry similar to that seen in fileless malware infections. Upon execution, open "C:\Windows\Temp" and verify that art-marker.txt is in the folder.



```
PS C:\Users\vagrant> Invoke-AtomicTest T1059.001 -TestNumbers 11
PathToAtomicsFolder = C:\AtomicRedTeam\atomics

Using Logger:  Default-ExecutionLogger
All logging commands found
Executing test: T1059.001-11 PowerShell Fileless Script Execution
The operation completed successfully.
Done executing test: T1059.001-11 PowerShell Fileless Script Execution
PS C:\Users\vagrant> _
```

Figure 9: T1059.001 Test #11

Technologies Used:

PowerShell, Atomic Red Team, Splunk, MITRE ATTACK, Command Prompt, The Registry

Learning Outcomes:

- The student will learn about the MITRE ATTACK technique T1059.
- The student will learn about log events that are triggered by this technique.

- The student will learn how to use Splunk Search and Reporting.
- The student will gain valuable experience analysing the log data returned in searches.
- Learn about the information included in logs and how it can determine if the activity is legitimate or malicious.

Lab 3: Windows Management Instrumentation

Description:

Adversaries may abuse Windows Management Instrumentation (WMI) to execute malicious commands and payloads. WMI is an administration feature that provides a uniform environment to access Windows system components. The WMI service enables both local and remote access, though the latter is facilitated by Remote Services such as Distributed Component Object Model (DCOM) and Windows Remote Management (WinRM). Remote WMI over DCOM operates using port 135, whereas WMI over WinRM operates over port 5985 when using HTTP and 5986 for HTTPS.[8]

Research Questions:

The students will be asked to answer and research a series of questions relating to the topic that the lab exercise is based on.

1. What MITRE ATTACK framework technique ID is assigned to the Windows Management Instrumentation?
2. What MITRE ATTACK framework tactic uses this technique?
3. Why do malicious actors use the Windows Management Instrumentation?
4. What can Malicious Actors use the Windows Management Instrumentation for?
5. Name and research groups and instances that have used this technique.
6. What are some mitigations against this type of technique?
7. How can this technique be detected?
8. What logs can be used to identify this technique?

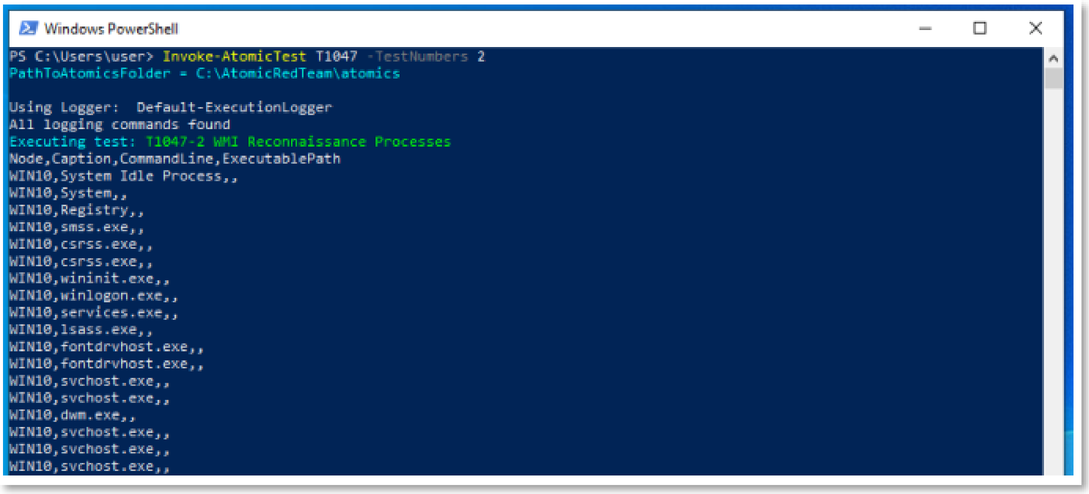
[8] (MITRE ATTACK, 2021)

Practical:

The practical element of the lab exercise requires the student to perform some testing using the Atomic Red Team atomics tests. The students may execute any tests related to these techniques. Each test is divided into two sections: Executing the Test and Searching for the Logs.

T1047 Test #2 - WMI Reconnaissance Processes

An adversary might use WMI to list Processes running on the compromised host. When the test completes, there should be running processes listed on the command line.

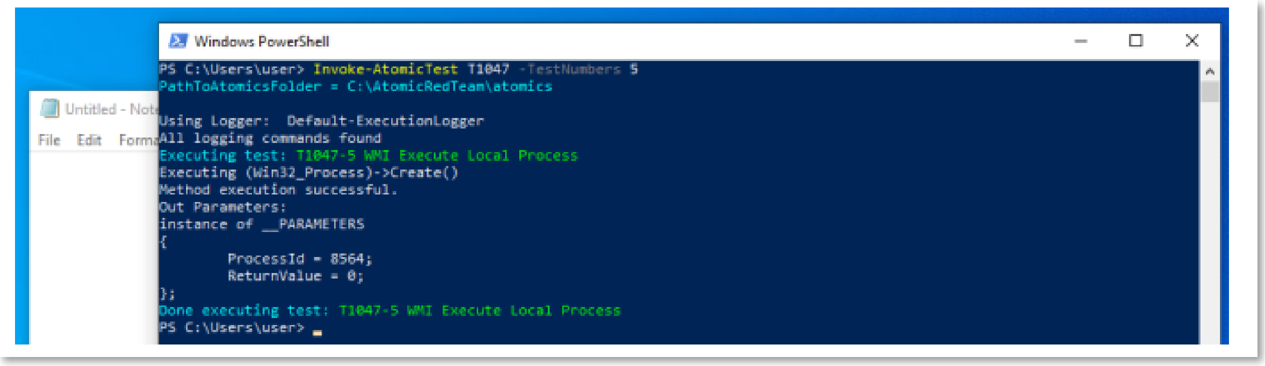


```
PS C:\Users\user> Invoke-AtomicTest T1047 -TestNumbers 2
PathToAtomicsFolder = C:\AtomicRedTeam\atomics

Using Logger:  Default-ExecutionLogger
All logging commands found
Executing test: T1047-2 WMI Reconnaissance Processes
Node,Caption,CommandLine,ExecutablePath
WIN10,System Idle Process,,
WIN10,System,,
WIN10,Registry,,
WIN10,smss.exe,,
WIN10,csrss.exe,,
WIN10,csrss.exe,,
WIN10,wininit.exe,,
WIN10,winlogon.exe,,
WIN10,services.exe,,
WIN10,lsass.exe,,
WIN10,fontdrvhost.exe,,
WIN10,fontdrvhost.exe,,
WIN10,svchost.exe,,
WIN10,svchost.exe,,
WIN10,dwm.exe,,
WIN10,svchost.exe,,
WIN10,svchost.exe,,
WIN10,svchost.exe,,
```

Figure 9: T1047 Test #2

T1047 Test #5 – WMI Execute Local Process

This test uses wmic.exe to execute a process on the local host. When the test completes, a new process will be started locally. A notepad application will be started when input is left on default.



```
PS C:\Users\user> Invoke-AtomicTest T1047 -TestNumbers 5
PathToAtomicsFolder = C:\AtomicRedTeam\atomics

Using Logger:  Default-ExecutionLogger
All logging commands found
Executing test: T1047-5 WMI Execute Local Process
Executing (Win32_Process)->Create()
Method execution successful.
Out Parameters:
instance of __PARAMETERS
{
        ProcessId = 8564;
        ReturnValue = 0;
};
Done executing test: T1047-5 WMI Execute Local Process
PS C:\Users\user> _
```

Figure 10: T1047 Test #5

Technologies Used:

PowerShell, Atomic Red Team, Splunk, MITRE ATTACK, WMI

Learning Outcomes:

- The student will learn about the MITRE ATTACK technique T1047.
- The student will learn about log events that are triggered by this technique.
- The student will learn how to use Splunk Search and Reporting.
- The student will gain valuable experience analysing the log data returned in searches.
- Learn about the information included in logs and how it can determine if the activity is legitimate or malicious.

Lab 4: Ingress Tool Transfer

Description:

Adversaries may transfer tools or other files from an external system into a compromised environment. Files may be copied from an external adversary-controlled system through the command-and-control channel to bring tools into the victim network or through alternate protocols with another tool such as FTP. Files can also be copied over on Mac and Linux with native tools like scp, rsync, and sftp.[9]

Research Questions:

The students will be asked to answer and research a series of questions relating to the topic that the lab exercise is based on.

1. What MITRE ATTACK framework technique ID is assigned to Ingress Tool Transfer?
2. What MITRE ATTACK framework tactic uses this technique?
3. Why do malicious actors use Ingress Tool Transfer?
4. What can Malicious Actors use Ingress Tool Transfer?
5. Name and research groups and instances that have used this technique.
6. What are some mitigations against this type of technique?
7. How can this technique be detected?

[9] (MITRE ATTACK, 2020)

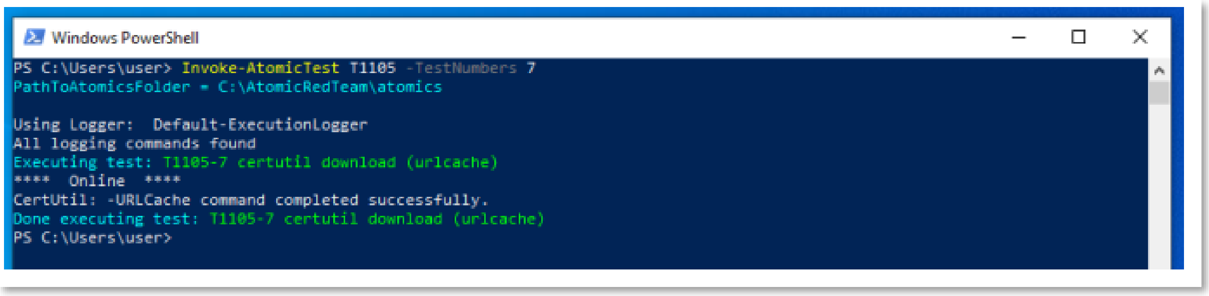
8. What logs can be used to identify this technique?

Practical:

The practical element of the lab exercise requires the student to perform some testing using the Atomic Red Team atomics tests. The students may execute any tests related to these techniques. Each test is divided into two sections: Executing the Test and Searching for the Logs.

T1105 Test #7 certutil download (urlcache)

This test will use certutil -urlcache to download a certificate from the internet.

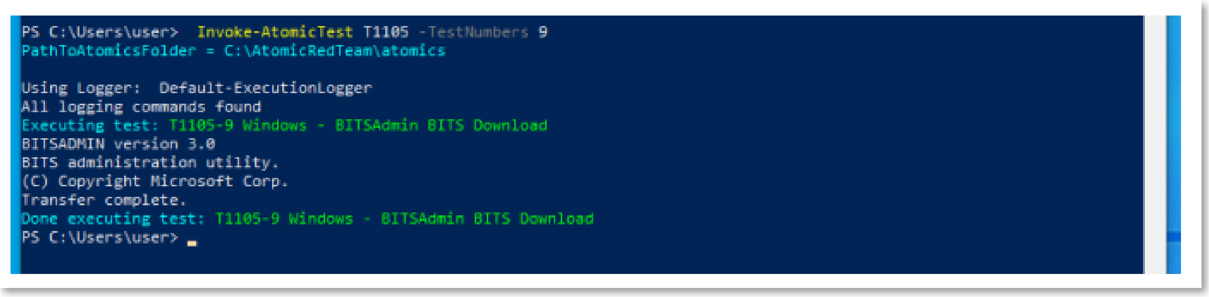


```
PS C:\Users\user> Invoke-AtomicTest T1105 -TestNumbers 7
PathToAtomicsFolder = C:\AtomicRedTeam\atomics

Using Logger:  Default-ExecutionLogger
All logging commands found
Executing test: T1105-7 certutil download (urlcache)
****  Online  ****
CertUtil: -URLCache command completed successfully.
Done executing test: T1105-7 certutil download (urlcache)
PS C:\Users\user>
```

Figure 11: T1105 Test #7

T1105 Test #9 Windows - BITSAdmin BITS Download

This test used BITSAdmin.exe to schedule a job to download a file. - BITSAdmin.exe is a tool that can be used to create, download or upload jobs and to monitor their progress.



```
PS C:\Users\user>  Invoke-AtomicTest T1105 -TestNumbers 9
PathToAtomicsFolder = C:\AtomicRedTeam\atomics

Using Logger:  Default-ExecutionLogger
All logging commands found
Executing test: T1105-9 Windows - BITSAdmin BITS Download
BITSADMIN version 3.0
BITS administration utility.
(C) Copyright Microsoft Corp.
Transfer complete.
Done executing test: T1105-9 Windows - BITSAdmin BITS Download
PS C:\Users\user> _
```

Figure 12: T1105 Test #9

Technologies Used:

PowerShell, Atomic Red Team, Splunk, MITRE ATTACK, certutil, BITSAdmin

Learning Outcomes:

- The student will learn about the MITRE ATTACK technique T1047.

- The student will learn about log events that are triggered by this technique.
- The student will learn how to use Splunk Search and Reporting.
- The student will gain valuable experience analysing the log data returned in searches.
- Learn about the information included in logs and how it can determine if the activity is legitimate or malicious.

Lab 5: Signed Binary Proxy Execution

Description:

Adversaries may bypass process and/or signature-based defences by proxying execution of malicious content with signed, or otherwise trusted, binaries. Binaries used in this technique are often Microsoft-signed files, indicating that they have been either downloaded from Microsoft or are already native in the operating system. Binaries signed with trusted digital certificates can typically execute on Windows systems protected by digital signature validation. Several Microsoft signed binaries that are default on Windows installations can be used to proxy execution of other files or commands.[10]

Research Questions:

The students will be asked to answer and research a series of questions relating to the topic that the lab exercise is based on.

9. What MITRE ATTACK framework technique ID is assigned to Ingress Tool Transfer?
10. What MITRE ATTACK framework tactic uses this technique?
11. Why do malicious actors use Ingress Tool Transfer?
12. What can Malicious Actors use Ingress Tool Transfer?
13. Name and research groups and instances that have used this technique.
14. What are some mitigations against this type of technique?
15. How can this technique be detected?
16. What logs can be used to identify this technique?

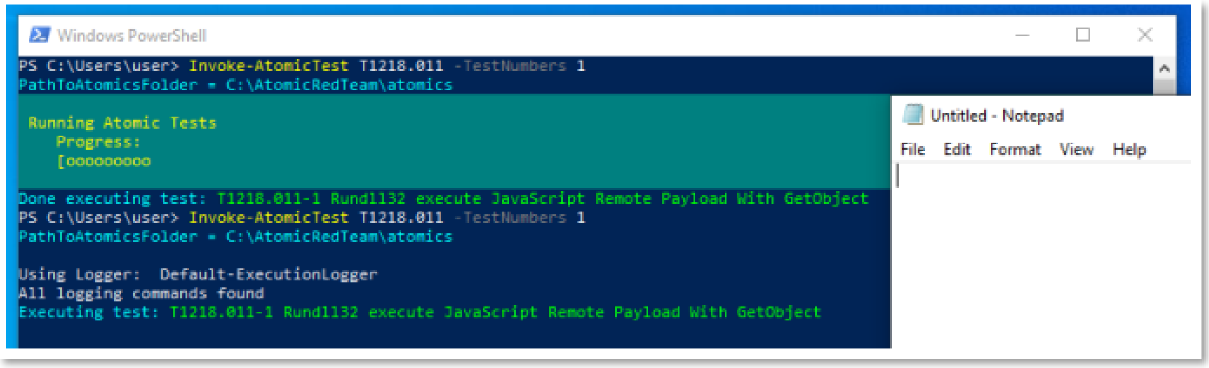
[10] (MITRE ATTACK, 2022)

Practical:

The practical element of the lab exercise requires the student to perform some testing using the Atomic Red Team atomics tests. We will be focusing on learning about the sub-technique for *Rundll32.* The students may execute any tests related to this technique. Each test is divided into two sections: Executing the Test and Searching for the Logs.

T1218.011 Test # 1

Test execution of a remote script using rundll32.exe. Upon execution notepad.exe will be opened.

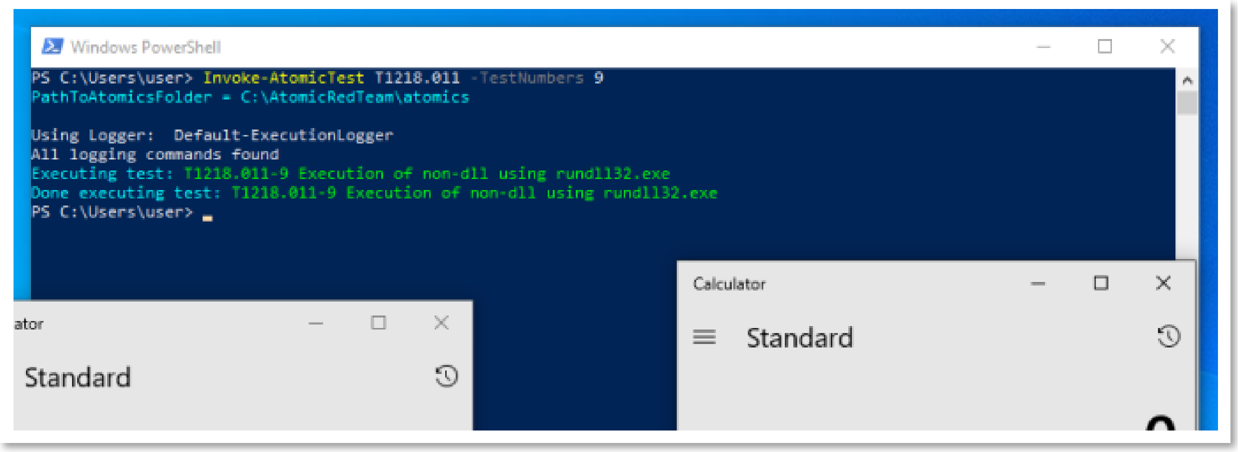


```
PS C:\Users\user> Invoke-AtomicTest T1218.011 -TestNumbers 1
PathToAtomicsFolder = C:\AtomicRedTeam\atomics

 Running Atomic Tests
    Progress:
    [oooooooooo

Done executing test: T1218.011-1 Rundll32 execute JavaScript Remote Payload With GetObject
PS C:\Users\user> Invoke-AtomicTest T1218.011 -TestNumbers 1
PathToAtomicsFolder = C:\AtomicRedTeam\atomics

Using Logger:  Default-ExecutionLogger
All logging commands found
Executing test: T1218.011-1 Rundll32 execute JavaScript Remote Payload With GetObject
```

Figure 13: T1218 Test #1

T1218.011 Test #9 Execution of non-dll using rundll32.exe

This test seeks to call Rundll32's StartW export function to load a DLL from the command line, using a non-dll file. Adversaries often use this technique in conjunction with Cobalt Strike to download17 malicious DLLs.



```
PS C:\Users\user> Invoke-AtomicTest T1218.011 -TestNumbers 9
PathToAtomicsFolder = C:\AtomicRedTeam\atomics

Using Logger:  Default-ExecutionLogger
All logging commands found
Executing test: T1218.011-9 Execution of non-dll using rundll32.exe
Done executing test: T1218.011-9 Execution of non-dll using rundll32.exe
PS C:\Users\user> _
```

Figure 14: T1218 Test #9

Technologies Used:

PowerShell, Atomic Red Team, Splunk, MITRE ATTACK, JavaScript, rundll32.exe

Learning Outcomes:

- The student will learn about the MITRE ATTACK technique T1047.
- The student will learn about log events that are triggered by this technique.
- The student will learn how to use Splunk Search and Reporting.
- The student will gain valuable experience analysing the log data returned in searches.
- Learn about the information included in logs and how it can determine if the activity is legitimate or malicious.

The Test Environment

The Detection lab consists of four virtual machines:

- An Ubuntu machine configured as a log collector.
- A Windows Server 2016 machine configured as a Domain Controller.
- A Windows Server 2016 machine configured as a Windows Event Forwarder.
- A Windows 10 machine configured to simulate a user of the system.

The Prerequisites:

- Clone the repository from the Detection Lab GitHub page.
- Install Vagrant 2.2.9+
- Install VirtualBox 6.0+

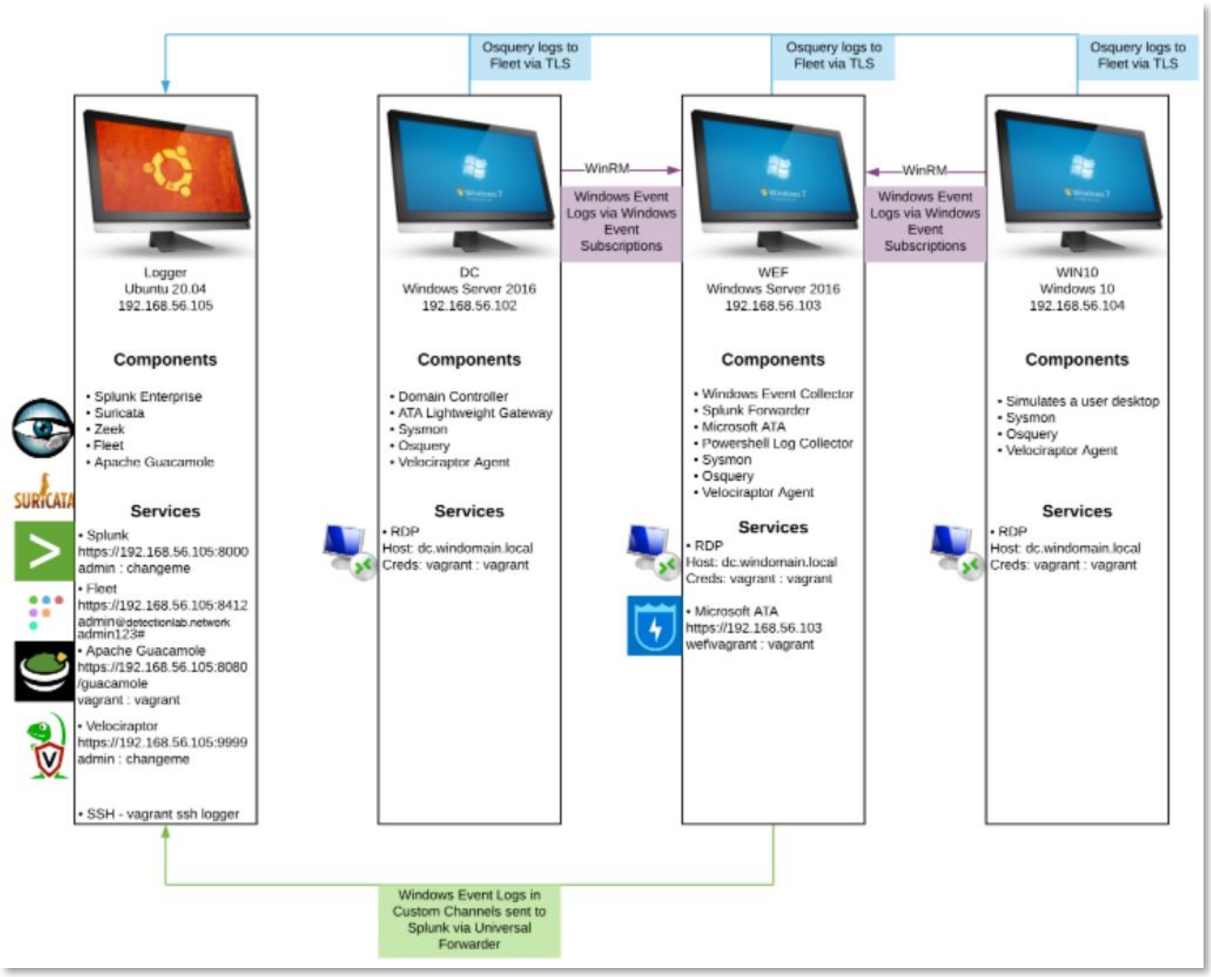
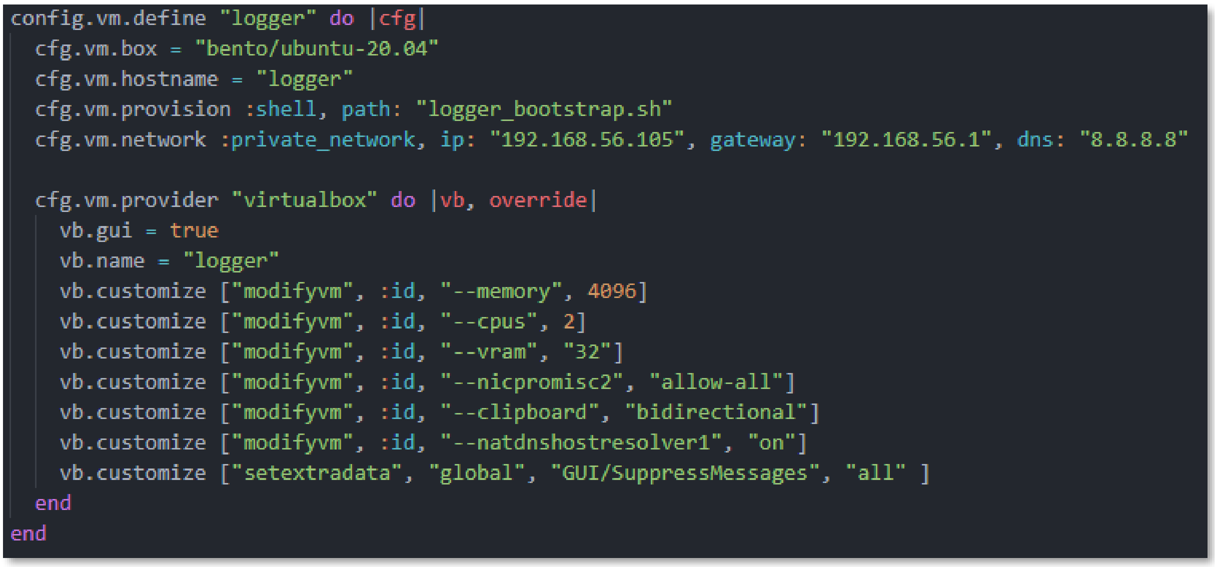


Figure 15: Detection Lab Configuration

The Detection Lab is deployed using PowerShell and Vagrant Scripts held within the repository. Vagrant is defined as a tool that is used for managing virtual machine environments. It is operated from the command-line and allows a user to interact with their virtual machines, including creating machines, suspending machines, shutting them down and destroying them. By using preconfigured scripts and boxes hosted online, this allows for easy replication of the environment. If an error occurs, entering two commands with Vagrant will destroy the machines, and create a new instance and allow you to start fresh.

Vagrant configures the virtual machines based on the configuration file, which is named the *Vagrantfile.* This file contains the location of the virtual machines we wish to deploy and the relevant system configuration, including any necessary scripts that will run during installation. The Vagrantfile is written in Ruby.



```
config.vm.define "logger" do |cfg|
  cfg.vm.box = "bento/ubuntu-20.04"
  cfg.vm.hostname = "logger"
  cfg.vm.provision :shell, path: "logger_bootstrap.sh"
  cfg.vm.network :private_network, ip: "192.168.56.105", gateway: "192.168.56.1", dns: "8.8.8.8"

  cfg.vm.provider "virtualbox" do |vb, override|
    vb.gui = true
    vb.name = "logger"
    vb.customize ["modifyvm", :id, "--memory", 4096]
    vb.customize ["modifyvm", :id, "--cpus", 2]
    vb.customize ["modifyvm", :id, "--vram", "32"]
    vb.customize ["modifyvm", :id, "--nicpromisc2", "allow-all"]
    vb.customize ["modifyvm", :id, "--clipboard", "bidirectional"]
    vb.customize ["modifyvm", :id, "--natdnshostresolver1", "on"]
    vb.customize ["setextradata", "global", "GUI/SuppressMessages", "all" ]
  end
end
```

Figure 16: Log Collector Configuration contained in the Vagrantfile

From the screenshot above:

- The name of the machine is defined as *"logger"*.
- The machine is hosted on Vagrant Cloud and is named *"bento/ubuntu-20.04"*.
- Network settings are configured.
- The shell script *"logger_bootstrap.sh"* contains commands that will install and configure relevant software on the Ubuntu Server.
- Resources are also assigned to the machines.

The other machines are configured in a similar manner and are provisioned with a series of PowerShell Scripts.

Splunk Universal Forwarders are installed on the Windows machines and they are configured to collect and send log data to Splunk on the Ubuntu server for storage and analysis.

The Windows Auditing configuration is configured through Group Policy objects via a PowerShell script as Vagrant provisioned the virtual machines. Some of the most important aspects of these GPOs is the enabling of the command line process monitoring and PowerShell transcript logging.

From the Detection Lab, the GPOs that were applied were as follows:

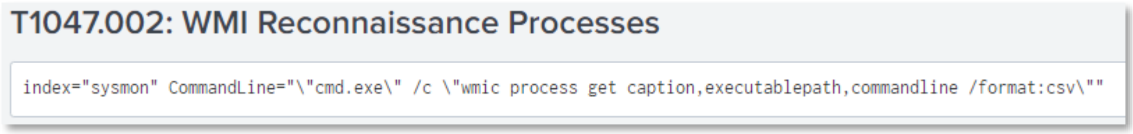
- Custom Event Channel Permissions - https://rawgit.com/clong/DetectionLab/master/Vagrant/resources/GPO/reports/Custom%20Event%20Channel%20Permissions.htm
- Default Domain Controllers Policy - https://rawgit.com/clong/DetectionLab/master/Vagrant/resources/GPO/reports/Default%20Domain%20Controllers%20Policy.htm
- Default Domain Policy - https://rawgit.com/clong/DetectionLab/master/Vagrant/resources/GPO/reports/Default%20Domain%20Policy.htm
- Domain Controllers Enhanced Auditing Policy - https://rawgit.com/clong/DetectionLab/master/Vagrant/resources/GPO/reports/Domain%20Controllers%20Enhanced%20Auditing%20Policy.htm
- PowerShell Logging – https://rawgit.com/clong/DetectionLab/master/Vagrant/resources/GPO/reports/Powershell%20Logging.htm
- Servers Enhanced Auditing Policy – https://rawgit.com/clong/DetectionLab/master/Vagrant/resources/GPO/reports/Servers%20Enhanced%20Auditing%20Policy.htm
- Windows Event Forwarding Server – https://rawgit.com/clong/DetectionLab/master/Vagrant/resources/GPO/reports/Windows%20Event%20Forwarding%20Server.htm
- Workstations Enhanced Auditing Policy - https://rawgit.com/clong/DetectionLab/master/Vagrant/resources/GPO/reports/Workstations%20Enhanced%20Auditing%20Policy.htm

The Windows Event Forwarder (WEF) server is configured with Palantir's WEF Subscriptions, which is a repository that enables the collection of logs relating to security events on a device and allows for rapid configuration of WEF.[11]

For further information regarding the deployment of the Detection Lab and it's configuration, please see the Lab Introduction hosted on the Showcase website.

[11] (Palantir, 2022)

Project Screenshots

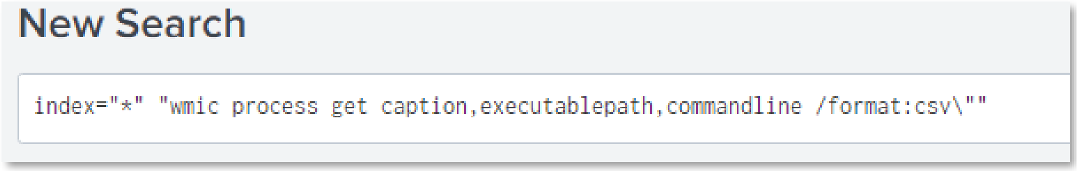


T1047.002: WMI Reconnaissance Processes

```
index="sysmon" CommandLine="\"cmd.exe\" /c \"wmic process get caption,executablepath,commandline /format:csv\""
```

Figure 17: Search Criteria for the Alert

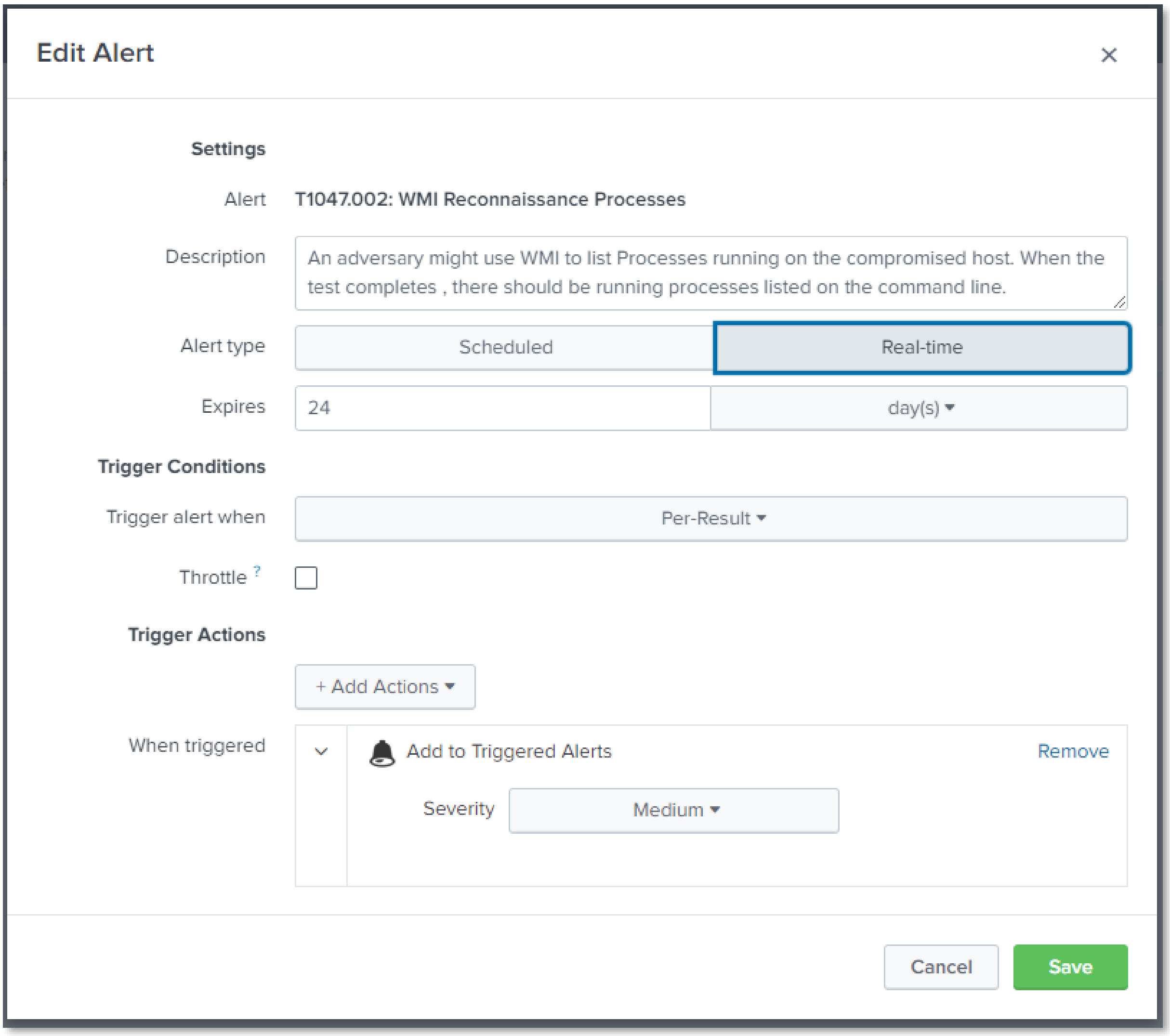
The screenshot above contains the Splunk search query to locate one of the logs generated by the atomic test for *T1047.002: WMI Reconnaissance Processes.* This search query will return the Sysmon logs relating to this alert. The screenshot below will return all logs associated with this event.



New Search

```
index="*" "wmic process get caption,executablepath,commandline /format:csv\""
```

Figure 18: Search Splunk for all occurrences of the command executed



Edit Alert

Settings

Alert: T1047.002: WMI Reconnaissance Processes

Description: An adversary might use WMI to list Processes running on the compromised host. When the test completes , there should be running processes listed on the command line.

Alert type: Scheduled [✓ Real-time]

Expires: 24 day(s)

Trigger Conditions

Trigger alert when: Per-Result

Throttle ? □

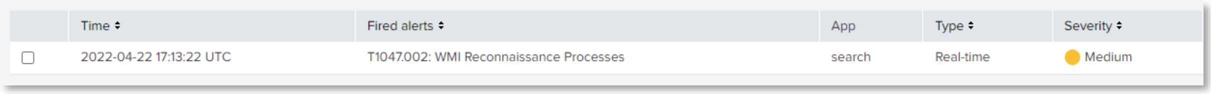
Trigger Actions

When triggered: Add to Triggered Alerts

Severity: Medium

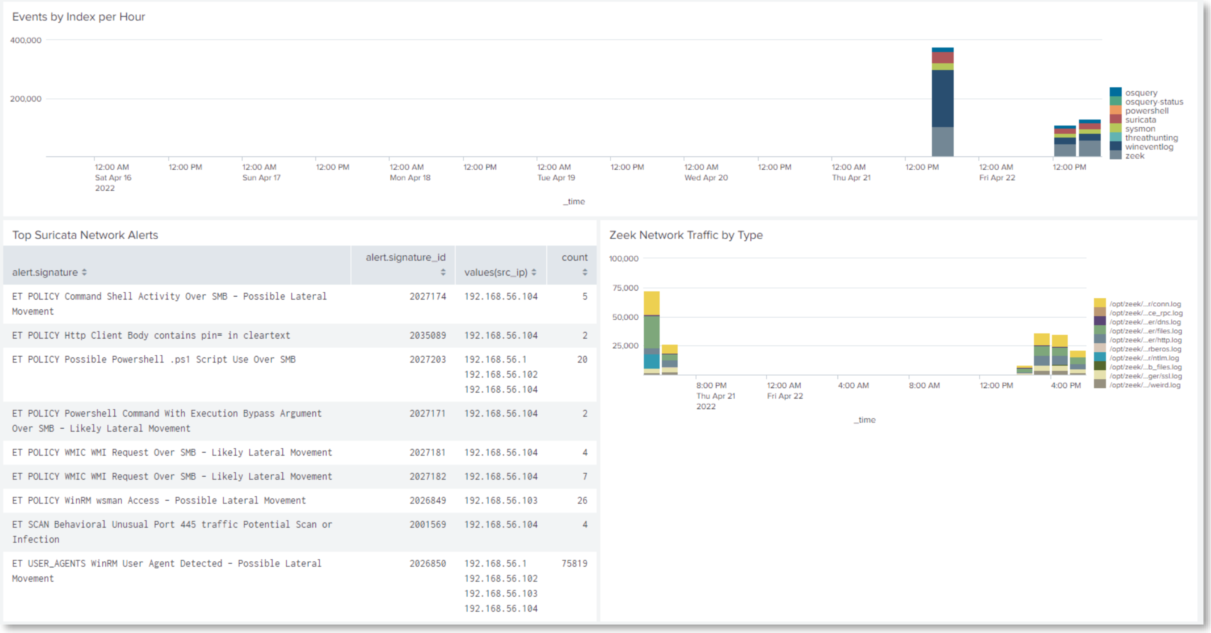
Figure 19: Alert Creation Menu

The screenshot above displays the Alert creation screen. To create an alert, you are required to build a search query, and click to save it as an Alert. This is the menu that opens. Simply input the details as such. When a test is performed, all alerts relating to that test will trigger.



	Time ⇕	Fired alerts ⇕	App	Type ⇕	Severity ⇕
□	2022-04-22 17:13:22 UTC	T1047.002: WMI Reconnaissance Processes	search	Real-time	● Medium

Figure 20: Triggered Alert



Top Suricata Network Alerts

alert.signature ⇕	alert.signature_id ⇕	values(src_ip) ⇕	count ⇕
ET POLICY Command Shell Activity Over SMB - Possible Lateral Movement	2027174	192.168.56.104	5
ET POLICY Http Client Body contains pin= in cleartext	2035089	192.168.56.104	2
ET POLICY Possible Powershell .ps1 Script Use Over SMB	2027203	192.168.56.1 192.168.56.102 192.168.56.104	20
ET POLICY Powershell Command With Execution Bypass Argument Over SMB - Likely Lateral Movement	2027171	192.168.56.104	2
ET POLICY WMIC WMI Request Over SMB - Likely Lateral Movement	2027181	192.168.56.104	4
ET POLICY WMIC WMI Request Over SMB - Likely Lateral Movement	2027182	192.168.56.104	7
ET POLICY WinRM wsman Access - Possible Lateral Movement	2026849	192.168.56.103	26
ET SCAN Behavioral Unusual Port 445 traffic Potential Scan or Infection	2001569	192.168.56.104	4
ET USER_AGENTS WinRM User Agent Detected - Possible Lateral Movement	2026850	192.168.56.1 192.168.56.102 192.168.56.103 192.168.56.104	75819

Figure 21: Pre-configured Dashboards (1)

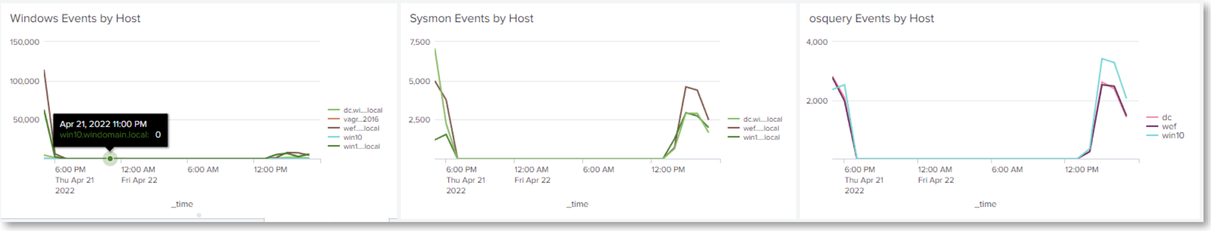
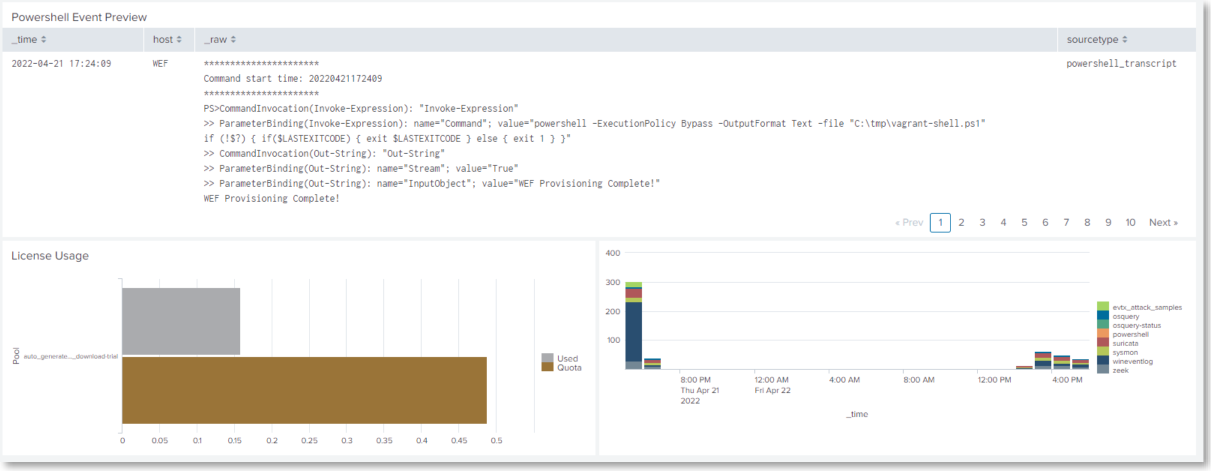


Figure 22: Pre-configured Dashboards (2)

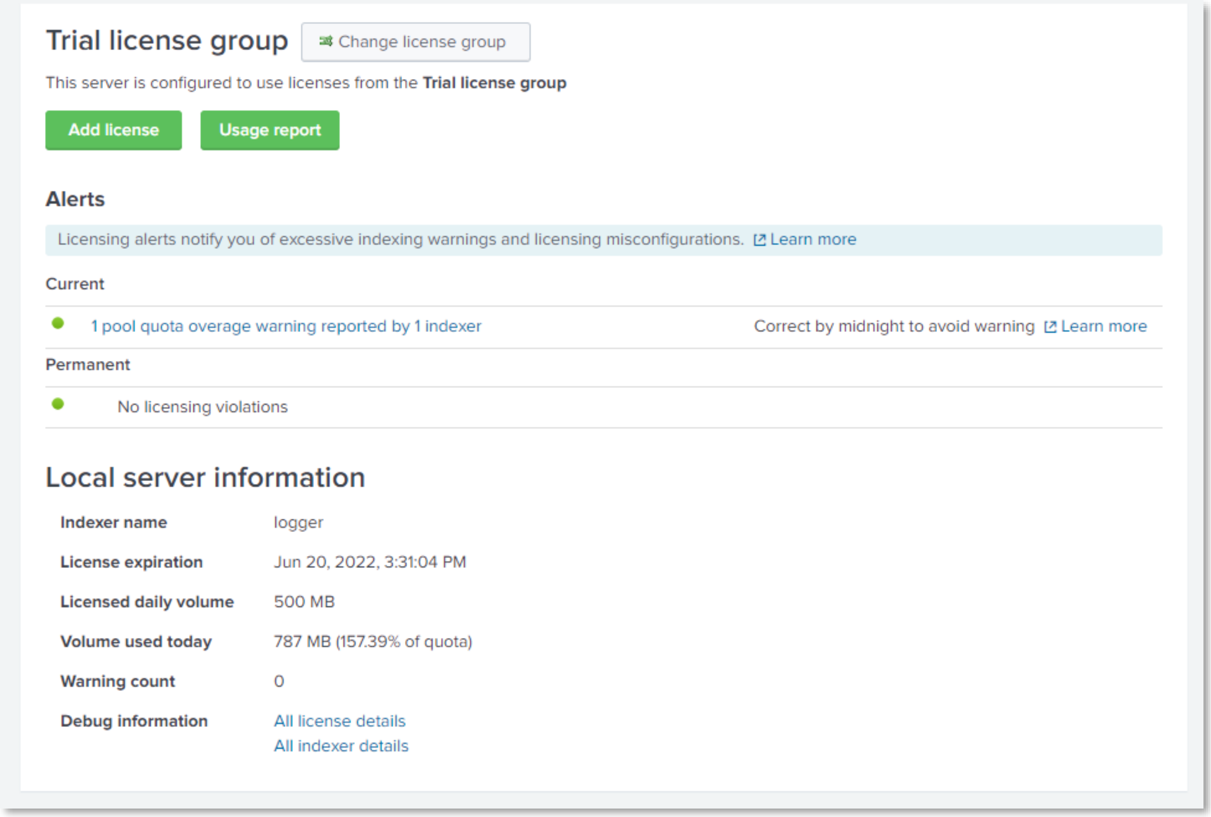

Powershell Event Preview

_time ⇕	host ⇕	_raw ⇕	sourcetype ⇕

2022-04-21 17:24:09	WEF	`**********************`
		`Command start time: 20220421172409`
		`**********************`
		`PS>CommandInvocation(Invoke-Expression): "Invoke-Expression"`
		`>> ParameterBinding(Invoke-Expression): name="Command"; value="powershell -ExecutionPolicy Bypass -OutputFormat Text -file "C:\tmp\vagrant-shell.ps1"`
		`if (!$?) { if($LASTEXITCODE) { exit $LASTEXITCODE } else { exit 1 } }"`
		`>> CommandInvocation(Out-String): "Out-String"`
		`>> ParameterBinding(Out-String): name="Stream"; value="True"`
		`>> ParameterBinding(Out-String): name="InputObject"; value="WEF Provisioning Complete!"`
		`WEF Provisioning Complete!`
			powershell_transcript

Figure 23: Pre-configured Dashboards (3)

When a user logs into Splunk, they are presented with a series of pre-configured Dashboards to display metrics of the various logs that are ingested. Figure 13 contains dashboards relating to the total amount of logs ingested per hour, Top Network Alerts detected by Suricata, and they type of Network traffic logged by Seek. Figure 14 breaks down the total Windows Events, Sysmon Events and osquery Events by machine type (DC, WEF, Win10) and they are graphed over time. Figure 15 displays PowerShell logs for quick analysis and also displays how much data you ingested on that day. The free Splunk licence limits data ingestion to 500MB per day. It is possible to exceed this limit, however I did receive warnings. Due to the level of testing that I was doing, I destroyed the virtual machines and reprovisioned them once a week.



Trial license group

This server is configured to use licenses from the **Trial license group**

Alerts

Licensing alerts notify you of excessive indexing warnings and licensing misconfigurations. Learn more

Current

● 1 pool quota overage warning reported by 1 indexer — Correct by midnight to avoid warning Learn more

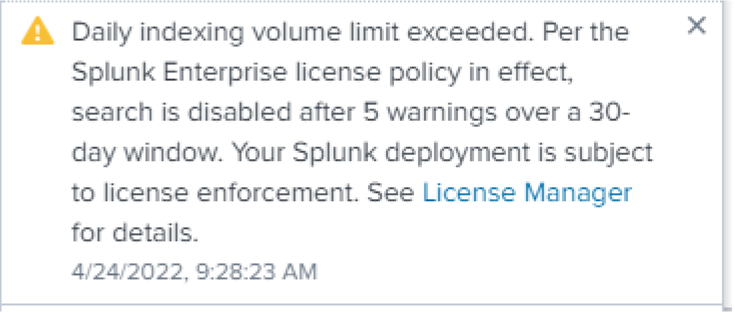
Permanent

● No licensing violations

Local server information

Indexer name	logger
License expiration	Jun 20, 2022, 3:31:04 PM
Licensed daily volume	500 MB
Volume used today	787 MB (157.39% of quota)
Warning count	0
Debug information	All license details All indexer details

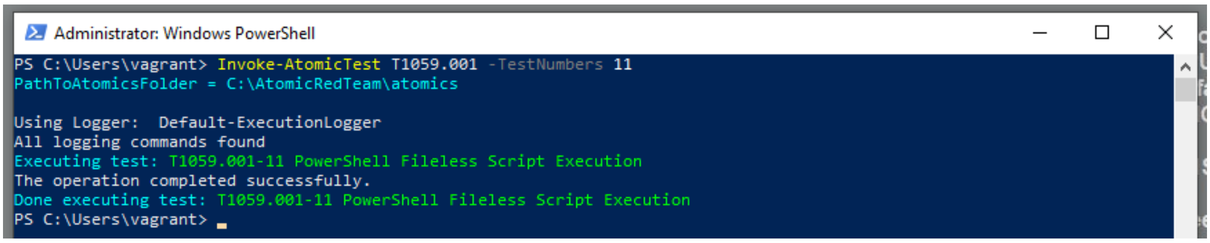
Figure 24: License Details



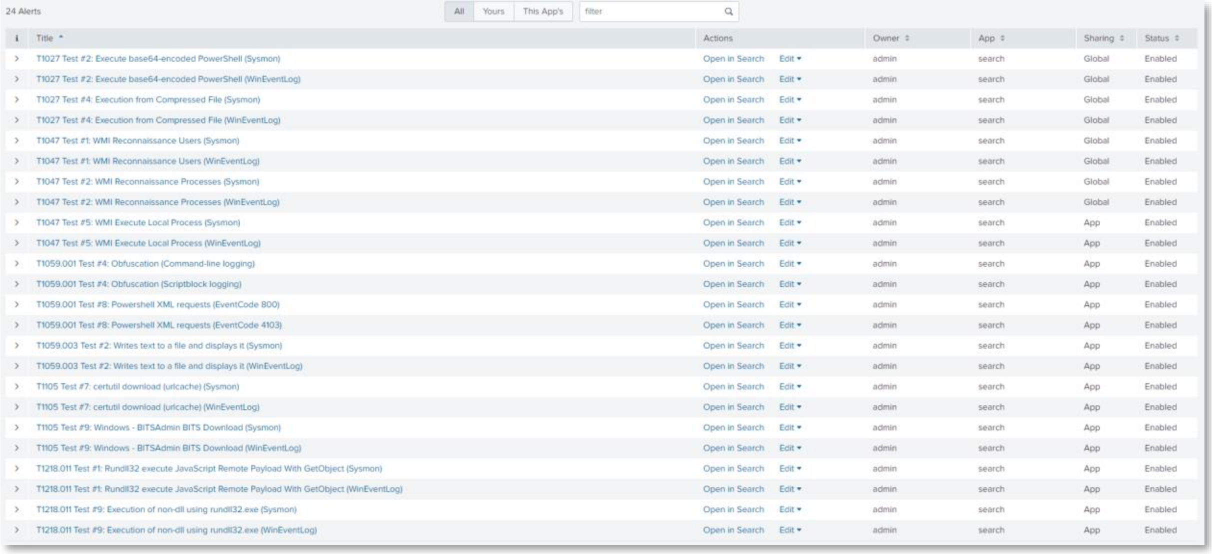
Daily indexing volume limit exceeded. Per the Splunk Enterprise license policy in effect, search is disabled after 5 warnings over a 30-day window. Your Splunk deployment is subject to license enforcement. See License Manager for details.

4/24/2022, 9:28:23 AM

Figure 25: Licence Warning



```
PS C:\Users\vagrant> Invoke-AtomicTest T1059.001 -TestNumbers 11
PathToAtomicsFolder = C:\AtomicRedTeam\atomics

Using Logger:  Default-ExecutionLogger
All logging commands found
Executing test: T1059.001-11 PowerShell Fileless Script Execution
The operation completed successfully.
Done executing test: T1059.001-11 PowerShell Fileless Script Execution
PS C:\Users\vagrant> _
```



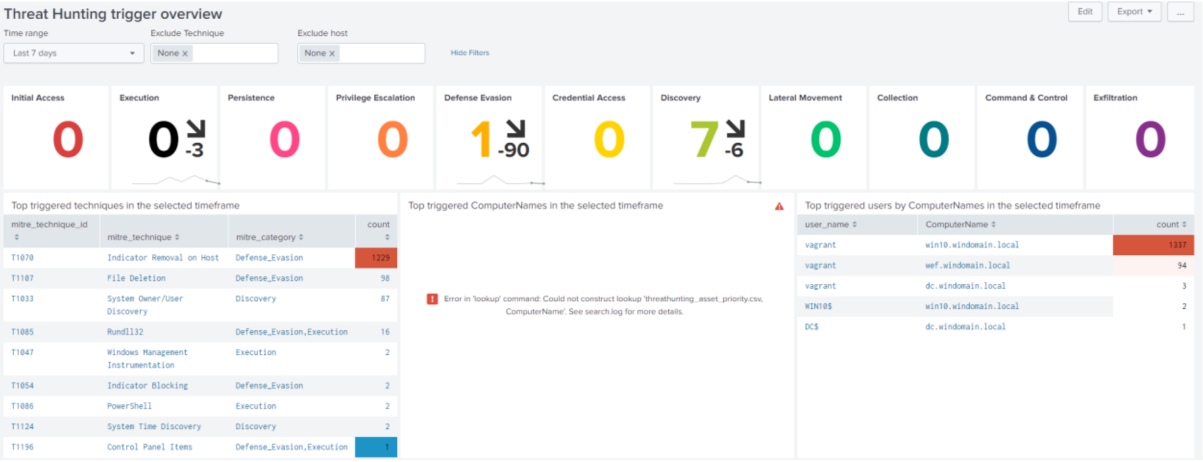
i	Title	Actions		Owner	App	Sharing	Status
>	T1027 Test #2: Execute base64-encoded PowerShell (Sysmon)	Open in Search	Edit ▾	admin	search	Global	Enabled
>	T1027 Test #2: Execute base64-encoded PowerShell (WinEventLog)	Open in Search	Edit ▾	admin	search	Global	Enabled
>	T1027 Test #4: Execution from Compressed File (Sysmon)	Open in Search	Edit ▾	admin	search	Global	Enabled
>	T1027 Test #4: Execution from Compressed File (WinEventLog)	Open in Search	Edit ▾	admin	search	Global	Enabled
>	T1047 Test #1: WMI Reconnaissance Users (Sysmon)	Open in Search	Edit ▾	admin	search	Global	Enabled
>	T1047 Test #1: WMI Reconnaissance Users (WinEventLog)	Open in Search	Edit ▾	admin	search	Global	Enabled
>	T1047 Test #2: WMI Reconnaissance Processes (Sysmon)	Open in Search	Edit ▾	admin	search	Global	Enabled
>	T1047 Test #2: WMI Reconnaissance Processes (WinEventLog)	Open in Search	Edit ▾	admin	search	Global	Enabled
>	T1047 Test #5: WMI Execute Local Process (Sysmon)	Open in Search	Edit ▾	admin	search	App	Enabled
>	T1047 Test #5: WMI Execute Local Process (WinEventLog)	Open in Search	Edit ▾	admin	search	App	Enabled
>	T1059.001 Test #4: Obfuscation (Command-line logging)	Open in Search	Edit ▾	admin	search	App	Enabled
>	T1059.001 Test #4: Obfuscation (Scriptblock logging)	Open in Search	Edit ▾	admin	search	App	Enabled
>	T1059.001 Test #8: Powershell XML requests (EventCode 800)	Open in Search	Edit ▾	admin	search	App	Enabled
>	T1059.001 Test #8: Powershell XML requests (EventCode 4103)	Open in Search	Edit ▾	admin	search	App	Enabled
>	T1059.003 Test #2: Writes text to a file and displays it (Sysmon)	Open in Search	Edit ▾	admin	search	App	Enabled
>	T1059.003 Test #2: Writes text to a file and displays it (WinEventLog)	Open in Search	Edit ▾	admin	search	App	Enabled
>	T1105 Test #7: certutil download (urlcache) (Sysmon)	Open in Search	Edit ▾	admin	search	App	Enabled
>	T1105 Test #7: certutil download (urlcache) (WinEventLog)	Open in Search	Edit ▾	admin	search	App	Enabled
>	T1105 Test #9: Windows - BITSAdmin BITS Download (Sysmon)	Open in Search	Edit ▾	admin	search	App	Enabled
>	T1105 Test #9: Windows - BITSAdmin BITS Download (WinEventLog)	Open in Search	Edit ▾	admin	search	App	Enabled
>	T1218.011 Test #1: Rundll32 execute JavaScript Remote Payload With GetObject (Sysmon)	Open in Search	Edit ▾	admin	search	App	Enabled
>	T1218.011 Test #1: Rundll32 execute JavaScript Remote Payload With GetObject (WinEventLog)	Open in Search	Edit ▾	admin	search	App	Enabled
>	T1218.011 Test #9: Execution of non-dll using rundll32.exe (Sysmon)	Open in Search	Edit ▾	admin	search	App	Enabled
>	T1218.011 Test #9: Execution of non-dll using rundll32.exe (WinEventLog)	Open in Search	Edit ▾	admin	search	App	Enabled

Figure 26: Alerts Configured for the Lab Exercises

Figure 17 displays the alerts that I created using Splunk search queries that I built using log telemetry generated from executing the tests contained in the lab exercises. These alerts are intended to be a guiding point for students to aid them in finding all the relevant logs. After the alerts have been created, and the student executes a test, they can navigate to the triggered alerts page. The student should open the related alert and note any important details and attempt to add or remove parameters to the search in order to locate further logs.

The free version of Splunk does not include the Splunk Enterprise application, which provides advanced analytics capabilities and incident response features. It combines automation and orchestration to collate relevant log data and thus enables a more thorough investigation. This is application requires a paid licence.[12]

[12] (Splunk, 2022)



Threat Hunting trigger overview

Time range: Last 7 days

Exclude Technique: None ×

Exclude host: None ×

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Command & Control	Exfiltration
0	0 ↘ -3	0	0	1 ↘ -90	0	7 ↘ -6	0	0	0	0

Top triggered techniques in the selected timeframe

mitre_technique_id	mitre_technique	mitre_category	count
T1070	Indicator Removal on Host	Defense_Evasion	1229
T1107	File Deletion	Defense_Evasion	98
T1033	System Owner/User Discovery	Discovery	87
T1085	Rundll32	Defense_Evasion,Execution	16
T1047	Windows Management Instrumentation	Execution	2
T1054	Indicator Blocking	Defense_Evasion	2
T1086	PowerShell	Execution	2
T1124	System Time Discovery	Discovery	2
T1196	Control Panel Items	Defense_Evasion,Execution	1

Top triggered ComputerNames in the selected timeframe

Error in 'lookup' command: Could not construct lookup 'threathunting_asset_priority.csv, ComputerName'. See search.log for more details.

Top triggered users by ComputerNames in the selected timeframe

user_name	ComputerName	count
vagrant	win10.windomain.local	1337
vagrant	wef.windomain.local	94
vagrant	dc.windomain.local	3
WIN10$	win10.windomain.local	2
DC$	dc.windomain.local	1

Figure 27: Threat Hunting Application

Figure 27 shows an overview of the Threat Hunting applications. Students can navigate to this screen and use it to investigate any testing activity they have completed. The threat hunting app receives its data from the Sysmon application. This does not come fully configured for all the MITRE ATTACK framework techniques, but I believe that students will find it useful as part of learning how to use a SIEM.

Learning Outcomes

Personal

I have learnt many things throughout the year and this project has been a great learning experience. I learnt a great deal this year about managing stress. There are just somethings that are out of your control. I learnt that if I stopped focusing on the negative aspects of the things that I can't change, that I could think clearly about those that needed to be completed.

I also learnt a lot about time management. This is where I feel that I could improve the most if I was to start the project again.

Technical

Prior to beginning this project, I had some experience using SIEM tools but I did not have the same understanding of how they worked as I do now. The research for this project was extensive and it did take me much longer to complete this section than I expected.

I began the project with optimism that I would be able to create a test environment with the relevant security tools and logging in place. My project supervisor, James, provided me with an image of an open-source SOC from a Siemplify webinar that had some of the required aspects of my project. My experience from working in a SOC was telling me that this is an excellent way for students to gain the relevant experience before entering the workplace and creating a prototype would be a good first step.

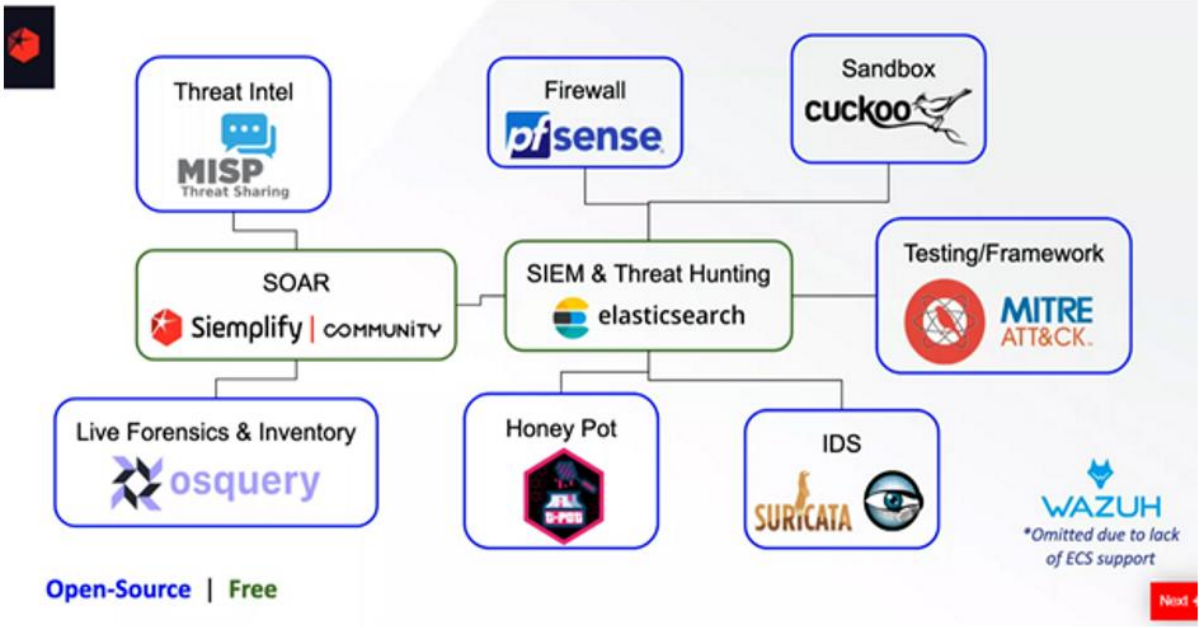


Figure 28: Siemplify Open-Source SOC

As I began my research, I discovered a paper published in Cal Poly Pomona, a public university in California titled *Exploring Depth in Cybersecurity Education Through the Lens of a SIEM.*[13] This paper discusses the requirements for a more hands on approach with regards to learning about SIEM technology, as with increased regulations (NIST, GDPR) comes a requirement to use a SIEM to manage your data. The paper also discusses how the students have access to a student SOC, where they can volunteer and gain much needed hands-on experience. These ideas resonated with me, and I felt I could bring them forward in my project.

As time went on, and I completed further research, I realised that the scope was too large and that I needed to change direction in order to have a demonstrate the importance of learning about a SIEM. I had no prior experience building an environment that complex and I was unsure if I would be able to deliver if I stayed on that track.

[13] (MacDonald, et al., 2020)

I had already completed research on similar projects and I had discovered the Detection Lab. This was set up very similar to the Siemplify Open-Source SOC and used some of the same tools. The SIEM used by Detection Lab was Splunk instead of the ELK Stack (Elasticsearch, Logstash, Kibana).

A takeaway from the technical side of my project is that if I was to start the project again, I would begin the implementation sooner. I would also research a way to better manage the machines, similar to using Packer and Vagrant as Chris Long did for the Detection Lab.

Vagrant

I had never used this tool before and even with a basic knowledge, I discovered that it was easy to install. The commands are straight forward, and the once I analysed the Vagrantfile, it was easier to understand the sequence of events when vagrant is used to deploy a machine.

Splunk

I had some experience using Splunk, so the UI was familiar to me. However, I was unfamiliar with the alert creation screens. I was familiar with using the Splunk Enterprise application, and this was missing due to the free license. I gained a lot of experience in using the searching feature and now I understand how to build search queries and to search for the exact information that I required.

Atomic Red Team

The tests provided by the Atomic Red Team that are mapped to the MITRE ATTACK framework enabled me to create and develop the lab exercises within the timeframe that I had set out in my new project plan. The test details provided me with enough information so that I could search Splunk for the logs created.

MITRE ATTACK

I learnt a great deal about the MITRE ATTACK framework and about how adversaries may leverage specific techniques.

Project Review

Problems Encountered

One of the major problems that I encountered during the implementation stage of my project was with installing virtual machines and configuring the software. I received four virtual machines that were hosted on IT Carlow's vSphere server, however when I began updating the Ubuntu machine, I found that some packages had been deprecated and at the time I was unable to find a solution. I made a decision to attempt to build the required machines on my personal PC and write myself a guide for the installation that I would later be able to replicate on the vSphere machines. However, as anyone that was worked with virtual machines knows, installations on different hardware, with different software can cause any number of problems.

I began with the installation of the Ubuntu machine, that would be used as the Log Collector, using VirtualBox on my personal computer. The installation went smoothly, so I took a snapshot and began the installation of the SIEM I had planned to use, the ELK Stack (Elasticsearch, Logstash, Kibana). I was unable to get Kibana running after the installation.

At the same time as the Kibana issues, I was also having problems installing a Windows 10 virtual machine, and the two Windows Server machines. This problem baffled me, as I had installed Windows 10 virtual machines without any problems in the past, and currently had a Windows 10 machine installed for the Reverse Engineering and Malware Analysis module. I confirmed that my machine was set up to run Windows virtual machines. A problem can occur sometimes if the *Virtual Machine Platform* and *Windows hypervisor Platform* are enabled in the optional features menu. They were disabled.

The other major problem that I had this year was time. I had a number of setbacks in my personal life that affected how I could allocate my time and with the increased workload after the Christmas break, I struggled to catch up on the work.

After a number of failed attempts at installing a Windows virtual machine, the last being an install that took 24 hours, only to fail at 95% I performed a fresh install of Windows. I was unable to find what the problem was that caused the installation to fail, I figured it must be conflicting with something on my machine. As the presentation was drawing close, I needed to be able to demonstrate what I hoped to achieve. When the fresh Windows installation completed, I decided to spin up the Detection Lab to see if it would work. Everything installed correctly and I could access all the services that I required to demonstrate what I had intended to create as a project.

My solution to the problems I had encountered was to change my project's direction. The core goal would stay the same, however the way I implemented it would change. It was during this redirection phase that I decided that creating lab exercises as supplemental material would be an excellent way to demonstrate the value of a SIEM tool and allow student to gain valuable experience in a structured manner.

Design Differences

My original plan involved creating an opensource SOC environment on virtual machines that would allow students to use SIEM and SOAR tools. In hindsight, I think that was too large a goal. I had been caught up in this great idea and with everything going on in my personal life, I did not give much thought to the implementation until after Christmas. As time progressed, the gravity of the situation hit me and I decided to come up with a new plan. One that I was sure I could implement. Many of the core elements that I had planned to implement came packaged with Detection Lab and it removed the time I would need to spend configuring machines and allowed me to focus my time on creating lab exercises that will demonstrate a SIEM tools value to students.

The fact that the Detection Lab installed correctly on my personal machine played a big factor in using this instead of using the Detection Lab ELK during the new planning stage. The Detection Lab deployment was consistent, and easily replicable. It takes approximately two hours to bring the lab online.

I had originally planned to use the ELK stack for the SIEM as it was opensource, however Splunk is provided with the Detection Lab and has a license that allows for the ingestion of 500MB per day.

Further Research

I had researched the Detection Lab before Christmas during a stage when I was investigating to see if there were similar projects to my own out there. I proceeded to read the documentation again and familiarise myself with some of the tools included with the lab.

My plan for the lab exercises was to create something that is relevant to what is observed in the industry. I began researching common MITRE ATTACK framework techniques that adversaries use and their corresponding atomic tests that were developed by the Atomic Red Team. Red Canary published their 2022 Threat Detection Report[14] in March, and this contained a list of the top 10 techniques observed in 2021.

I also researched a number of free courses provided by Splunk which I think would be hugely beneficial to students. The free online videos require students to register, and they can enrol in free classes that were once packaged under the Splunk Fundamentals Part 1 course. I completed parts of this prior to my internship and having a basic knowledge of a SIEM tool helped me hugely when it began.

Future Projects

The brief of the project mentioned implementing a SOAR tool alongside a SIEM tool. This was something that I did not achieve due to the workload and time management issues. I discovered a nice tool by a company called Tines. They are an Irish company that provide automation services without the need to write code. I was excited to attempt to integrate this with my project but I think it would be a nice addition in any future iterations of this project.

Developing training surrounding the other services included in the Detection Lab could become a project. I did not have time to investigate the use of Fleet as a manager for osquery, or to use Microsoft’s Advanced Threat Analytics.

Recommendations

Based on the research I completed throughout the year, my own work experience and from comments from companies that will be hiring students from IT Carlow I would recommend

[14] (Red Canary, 2022)

implementing some form of education surrounding Splunk. As recently as August 2021, Unum were looking to hire a Senior Splunk Engineer[15], State Street mentioned that they use Splunk during a talk they gave to the IT Carlow, and I have used Splunk during my internship.

From the additional research that I performed; I would recommend assigning the students a task of completing the following free online courses:

- **What is Splunk?** - https://education.splunk.com/course/what-is-splunk
- **Intro to Splunk** - https://education.splunk.com/course/intro-to-splunk-elearning
- **Using Fields** – https://education.splunk.com/course/using-fields
- **Scheduling Reports and Alerts** – https://education.splunk.com/course/scheduling-reports-alerts-elearning
- **Visualizations** - https://education.splunk.com/course/visualizations-elearning

The students would be required to register with the Splunk website and enrol in the courses. They total 6 and a half hours of video. The knowledge gained through watching these training videos will enable the student to complete the lab exercises more thoroughly.

I am also making a recommendation to use the five lab exercises I have created for the two new modules that are being created.

- Year 2: Incident Handling Project Module
- MSc: Advanced Incident response

I will be presenting these documents to the lecturers involved prior to the project demonstration. The documents will also be hosted on the Showcase website. I will also include some brief video demonstrations of the lab exercises.

The lab exercises cover the following techniques from the MITRE ATTACK framework:

- T1027: Obfuscated Files of Information
- T1059: Command and Scripting Interpreter
- T1047: Windows Management Instrumentation
- T1105: Ingress Tool Transfer
- T1218: Signed Binary Proxy Execution

[15] (Unum, 2021)

Acknowledgements

I would like to thank my project supervisor James Egan for his continued support throughout the year. I would also like to thank all the IT Carlow staff and lecturers who have taught and supported me over the past four years. There were many challenges to overcome, with Covid being the biggest obstacle, but we got there in the end. I would also like to thank Chris Long and his dedication to creating the Detection Lab, it provided the perfect testing environment to demonstrate the effectiveness of using a SIEM for log analysis.

References

Atomic Red Team, 2022. *Atomic Red Team.* [Online]

Available at: https://github.com/redcanaryco/atomic-red-team

[Accessed 22nd April 2022].

Long, C., 2022. *DetectionLab.* [Online]

Available at: https://github.com/clong/detectionlab

[Accessed 22nd April 2022].

MacDonald, M. R., Pike, N. D. & Pike, R. E., 2020. *Exploring Depth in Cybersecurity Education,* California: Cal Poly Pomona.

MITRE ATTACK, 2020. *Ingress Tool Transfer.* [Online]

Available at: https://attack.mitre.org/techniques/T1105/

[Accessed 25th April 2022].

MITRE ATTACK, 2021. *Command and Scripting Interpreter.* [Online]

Available at: https://attack.mitre.org/techniques/T1059/

[Accessed 25th April 2022].

MITRE ATTACK, 2021. *Obfuscated Files or Information.* [Online]

Available at: https://attack.mitre.org/techniques/T1027/

[Accessed 25th April 2022].

MITRE ATTACK, 2021. *Windows Management Instrumentation.* [Online]

Available at: https://attack.mitre.org/techniques/T1047/

[Accessed 25th April 2022].

MITRE ATTACK, 2022. *Signed Binary Proxy Execution.* [Online]

Available at: https://attack.mitre.org/techniques/T1218/

[Accessed 25th April 2022].

Palantir, 2022. *Windows Event Forwarding Guidance.* [Online]

Available at: https://github.com/palantir/windows-event-forwarding

[Accessed 22nd April 2022].

Red Canary, 2022. *2022 Threat Detection Report.* [Online]

Available at: https://resource.redcanary.com/rs/003-YRU-

314/images/2022_ThreatDetectionReport_RedCanary.pdf
[Accessed 22nd April 2022].

Slintel, 2022. *Market Share of Splunk.* [Online]
Available at: https://www.slintel.com/tech/security-information-and-event-management-siem/splunk-market-share
[Accessed 22nd April 2022].

Splunk, 2022. *Splunk Enterprise Security.* [Online]
Available at: https://splunkbase.splunk.com/app/263/#/overview
[Accessed 24th April 2022].

Unum, 2021. *Splunk Senior Infrastructure Engineer.* [Online]
Available at: https://www.irishjobs.ie/Jobs/Splunk-Senior-Infrastructure-Engineer-8648280.aspx
[Accessed 22nd April 2022].