Best Practice

Verbose Logging: Best Practices for Default Content's Effective Utilization

opentext™

Contents

Introduction

To a cyber security professional, logs provide information about events happening in an environment. ArcSight Default Content utilizes the information provided by logs to detect attacks to your environment. However, logs do not provide sufficient information for Default Content resources unless you configure certain systems or application settings to make the information more verbose. Verbose logging is the practice of recording as much information as possible about events that occur on your system. When you enable verbose logging, the additional information recorded is essential for optimizing the Default Content resources for ArcSight ESM using the MITRE ATT&CK Framework.

Windows Logging

The Windows event logs are a detailed and chronological record of system, security, and application notifications stored by the Windows operating system. Default Content resources use these logs to identify potential attacks to your system. However, these Windows logs do not provide detailed information to trigger alerts unless you configure them.

Below are the different types of logs that must be enabled for verbose logging to optimize security alerts from Default Content Resources. Each section contains information about the log and how to enable verbose logging.

- Widows Security Logs
- PowerShell Logs
- Sysmon Logs

Windows Security Logs

ArcSight Default Content relies on many Windows Security logs for optimal alerts.

This section covers:

- Widows Event ID: 4688
- Other Windows Security Logs

Windows Event ID: 4688

Windows Event ID: 4688 logs process creation events. There are four principal events that cause process creation:

- System initialization
- Execution of a process creation system call by running a process.
- A user request to create a new process.
- Initiation of a batch job

Windows Event ID: 4688 is not enabled in Windows by default. However, multiple ArcSight Default Content resources require Windows Event ID:4688 to trigger an alert for a potential attack to your system. Also, once 4688 event is enabled, it should be ensured that the process command line is also enabled. If you enable the command line for 4688, more details will be recorded, including the New Process ID, New Process Name, Token Elevation Type, Mandatory Label, and more. Fig 1 shows process event ID before enabling the command line. Fig 2 shows the verbose event after command line is enabled.

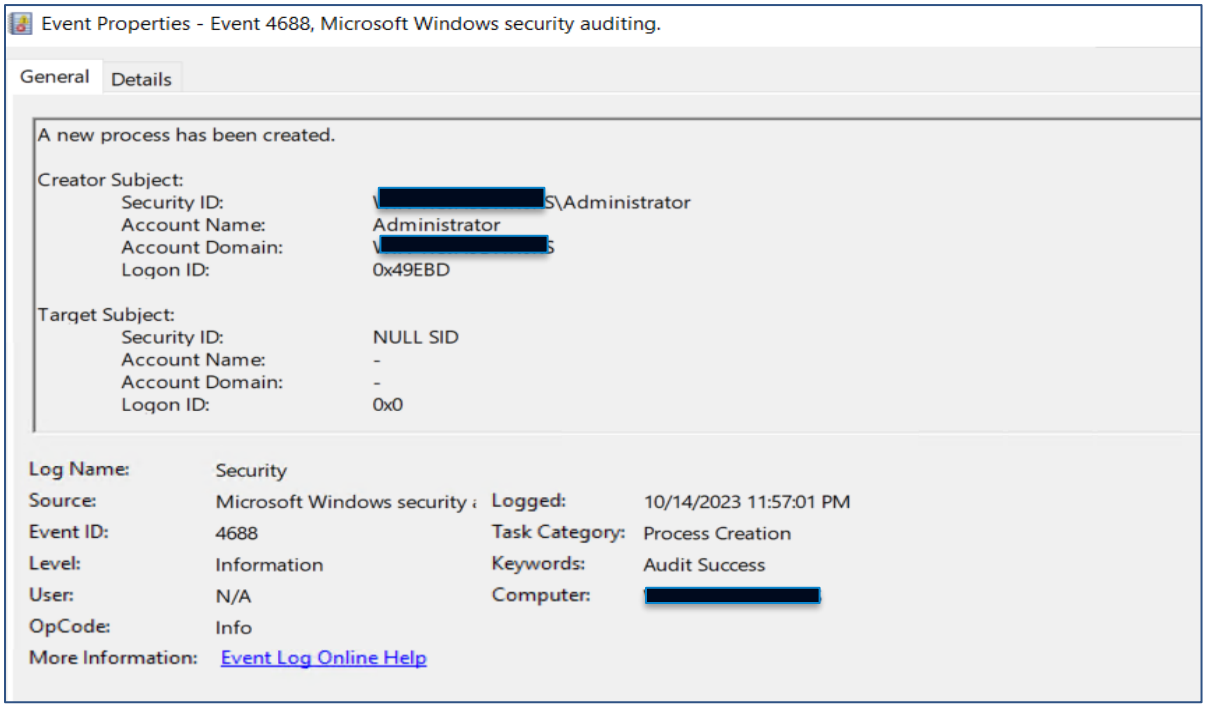


Fig 1

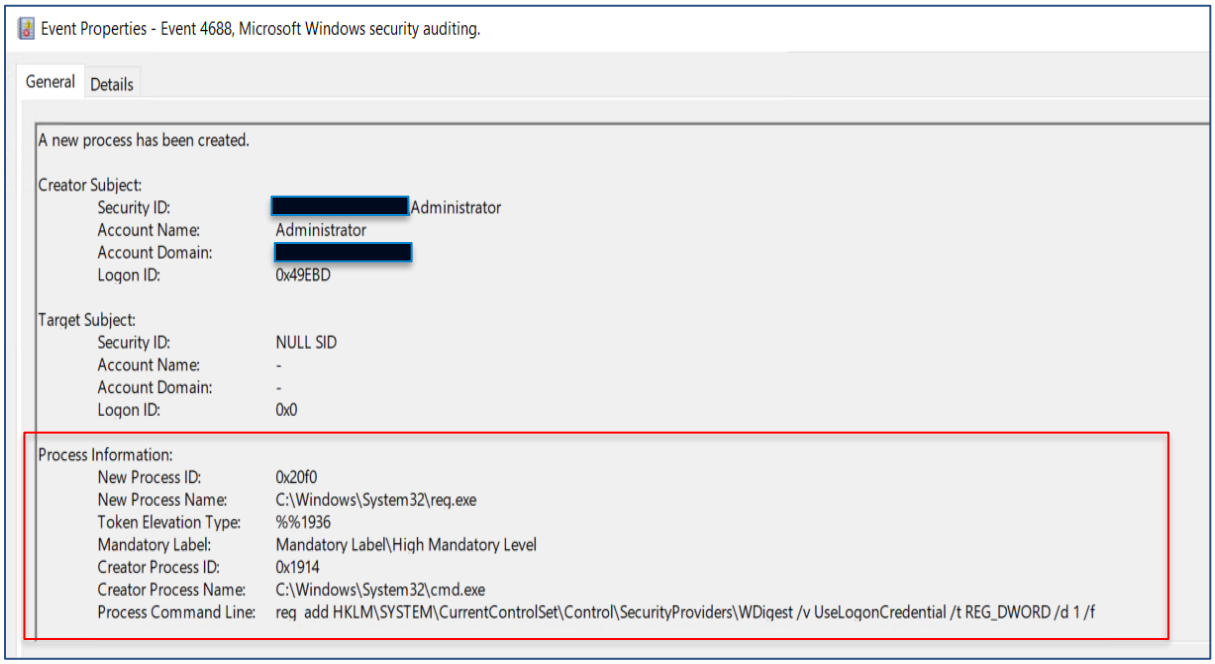


Fig 2

Enable Windows Event ID:4688 in the Group Policy Editor

1. In Group Policy Editor, follow this path to the Detailed Tracking folder:

   Windows Settings>Security Settings> Advanced Audit Policy Configuration>Audit Policies>Detailed Tracking

   In this document, a Windows Server 2019 is being used as an example. Please follow the configuration settings as per the windows version you are using in your environment.

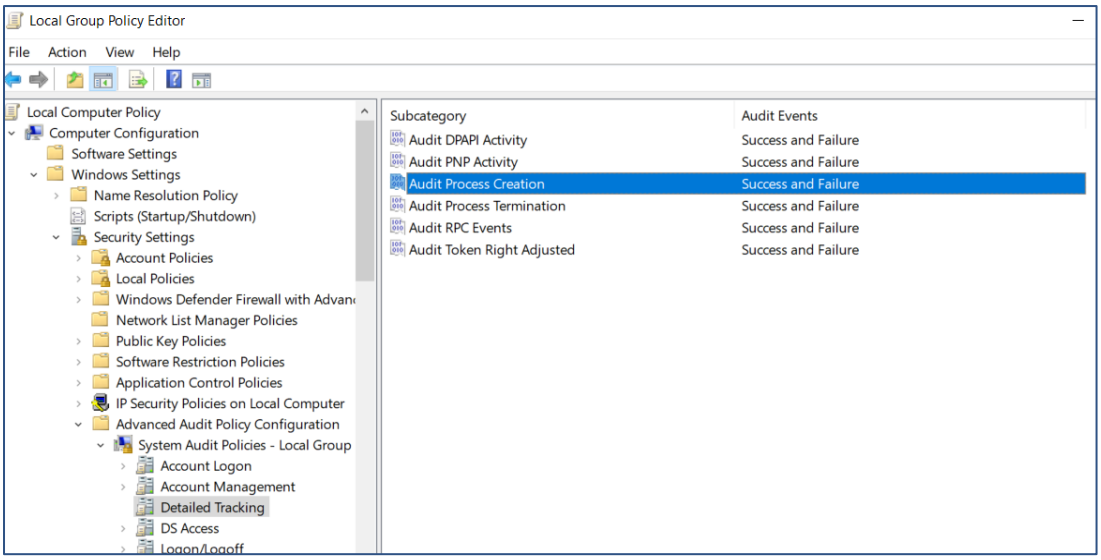


Fig 3

2. As per Fig 3, in the Detailed Tracking Folder, select Audit Process Creation.
3. In the new window, select Success and Failure.
4. Click OK.

Then enable the command line:

5. Open the Group Policy Editor on the Windows machine you want to monitor and follow this path to the Audit Process Creation folder:

   Administrative Templates>System>Audit Process Creation
6. Select Edit policy setting.
7. In the new window, select Enabled.

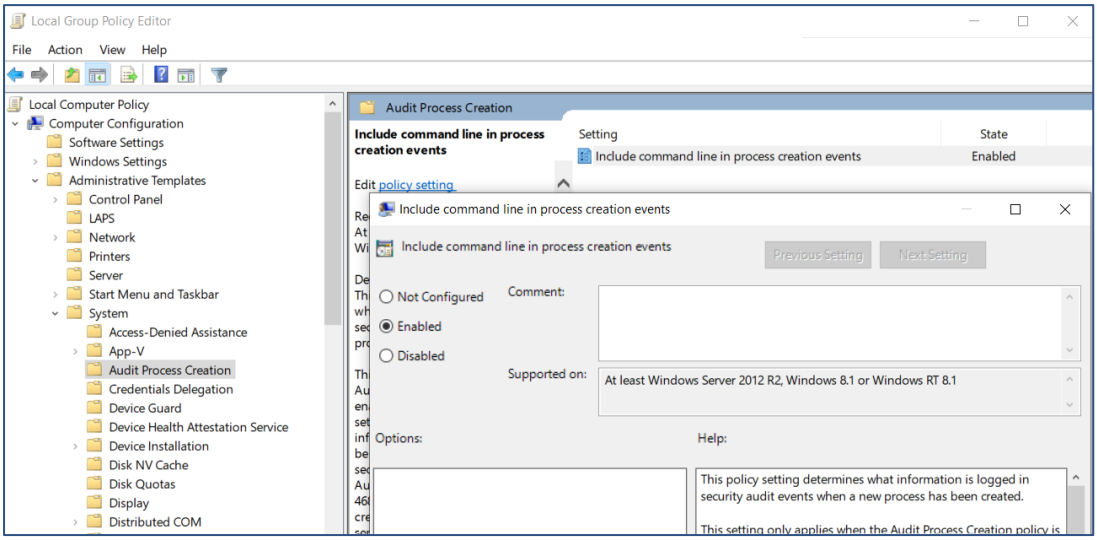


Fig 4

8. Select OK.

To save the new settings in Group Policy Object, it's important to run the gpupdate /force command to force a background update of all Group Policy settings, regardless of if they have changed. By default, Windows will update group policy settings every 90 minutes or during a computer reboot. However, this is one of those times when an immediate update is necessary, by using the gpupdate command, you can force a policy update. A system restart may be required for those

Group Policy client-side extensions that do not process policy on a background update cycle but do process policy at a computer start up.

Other Windows Security Logs

To optimize ArcSight Default Content resources, it is necessary to enable Audit Policy events for event IDs such as 4624, 4719, etc. It provides information about basic audit policies that are available in Windows and links to information about each setting.

Enable Audit Policy Events in the Group Policy Editor

1. In Group Policy Editor, follow this path to the Detailed Tracking folder:
   Windows Settings>Security Settings> Local Policy>Audit Policies
2. In the Audit Policy folder, select each of these policies shown in Fig. 5 and enable them to alert for successes and failures.

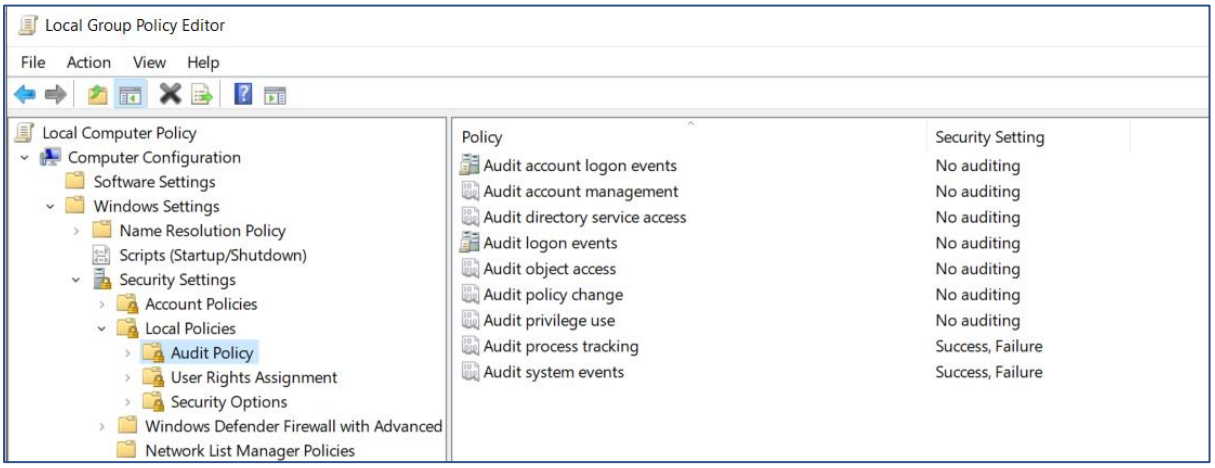


Fig 5

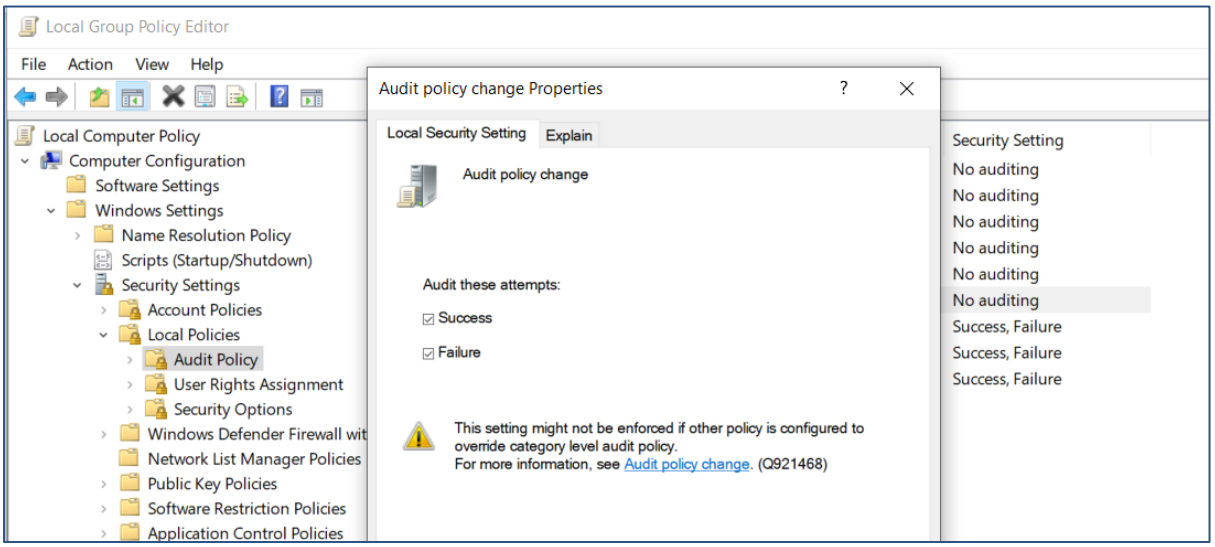


Fig 6

3. In the new window, select Success and Failure.
4. Click OK.

PowerShell Logs

ArcSight Default Content relies on PowerShell logs to trigger alerts to potential threats to your system.

PowerShell supports three types of logging: module, script block, and transcription. PowerShell events are written to the PowerShell operational log: Microsoft-Windows-PowerShell Operational.

- Module logging records pipeline execution details as PowerShell executes, including variable initialization and command invocations. This type of log captures some details missed by other PowerShell logging sources, though it may not reliably capture the commands executed.
- Script block logging records blocks of code as they are executed by the PowerShell engine, thereby capturing the full contents of codes executed by an attacker, including scripts and commands.
- Transcription logging creates a unique record of every PowerShell session, including all input and output, exactly as it appears in the session.

Enable the Above Logging Capabilities

1. In Group Policy Editor, follow this path to the Windows PowerShell settings folder:
   Computer Configuration> Administrative Templates> Windows Components>Windows PowerShell

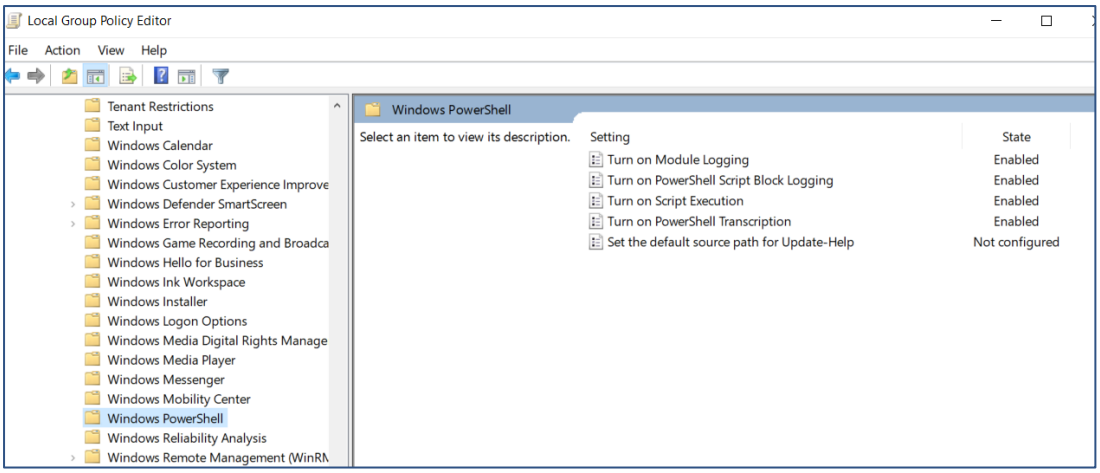


Fig 7

2. Select the settings and enable Module, Script Block and Transcription logging.

Below is an example of enabling module logging.

1. In the Options pane, click Module Name as per Fig 8. In the Module Name window, enter * to record all modules.
2. Click OK.

Alternatively, setting the following registry values using command line will have the same effect. For example, the below commands can be run to enable PowerShell Module logging.

- reg add HKLM\SOFTWARE\Wow6432Node\Policies\Microsoft\Windows\PowerShell\ModuleLogging /d 1
- reg add HKLM\SOFTWARE\Wow6432Node\Policies\Microsoft\Windows\PowerShell\ModuleLogging /v \ModuleNames /d *

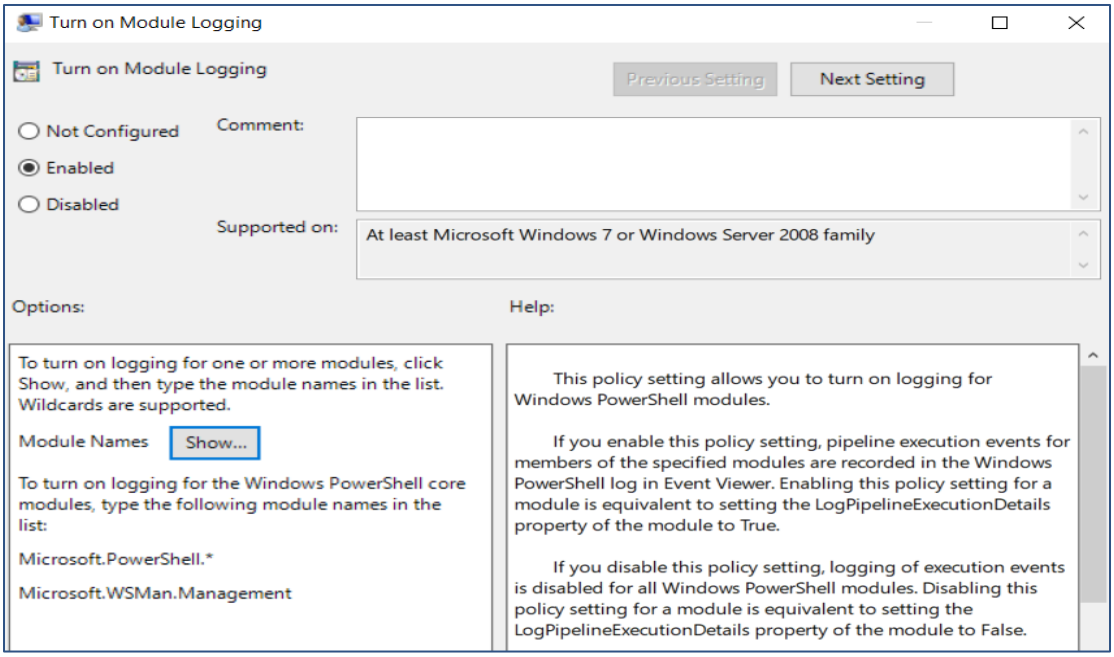


Fig 8

There are two locations where the PowerShell Logs are recorded in the windows event viewer.

- Application and Services Logs> Windows PowerShell records events such as 800 (Fig 9).
- Application Service Logs> Microsoft>Windows>PowerShell records events such as 4104 (Fig 10).

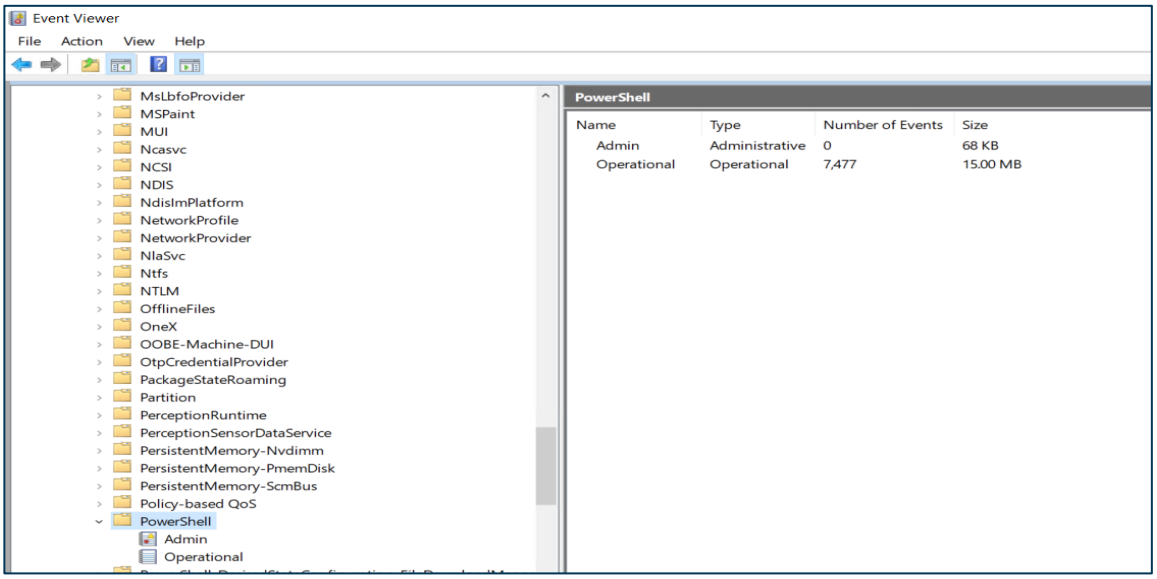


Fig 9

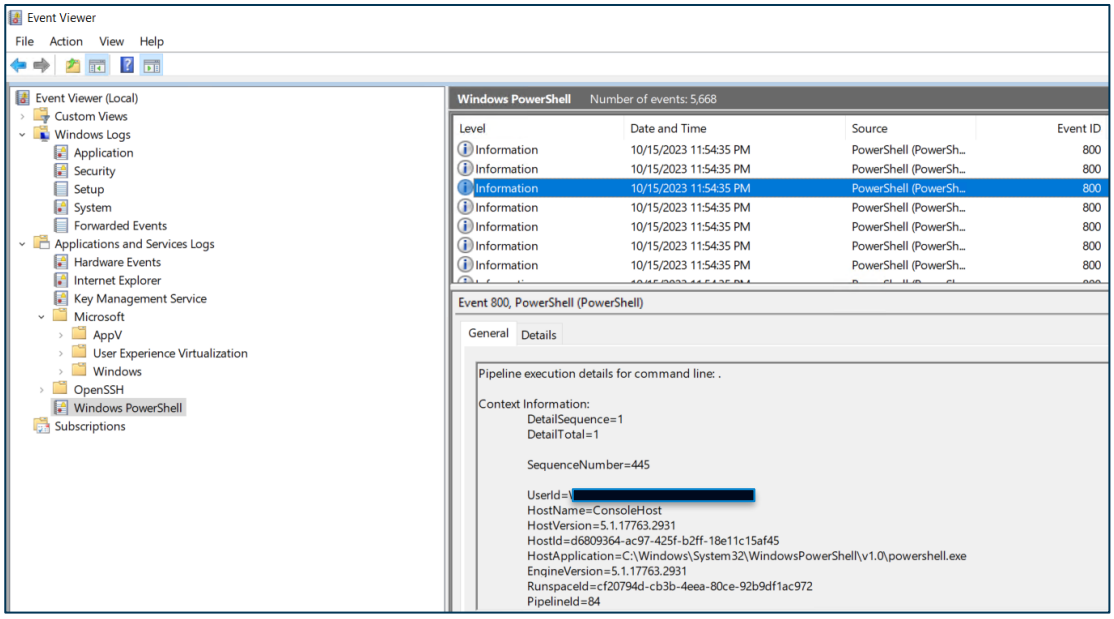


Fig 10

Sysmon Logs

ArcSight Default Content relies on Sysmon logs to trigger alerts to potential threats in your environment, such as: Microsoft-Windows-Sysmon 1, 3, 7, 8, 10, 11, 12, 13, 15, 17, 19, 20 and 21.

Sysmon is part of the Sys-internals software package, now owned by Microsoft, and enriches the standard Windows logs by producing some higher-level monitoring of events such as process creations, network connections, and changes to the file system.

The latest release of Sysmon can be downloaded from the Microsoft page (https://learn.microsoft.com/en-us/sysinternals/downloads/sysmon) and installed.

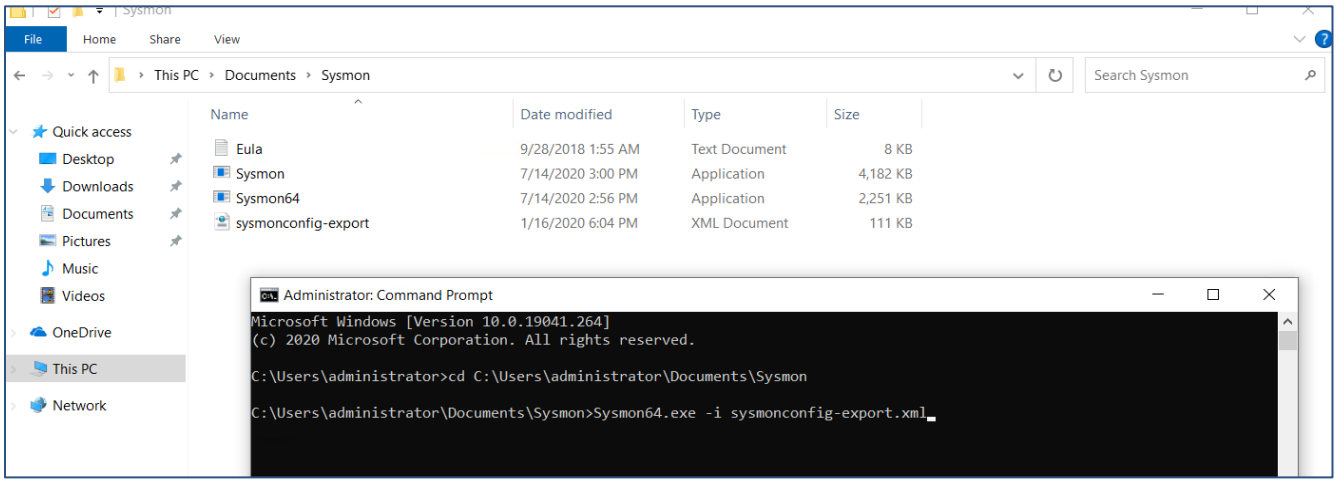


Fig 11

Once the Sysmon is installed, you can verify if Sysmon service is up and running in the services app on your Windows Machine as per Fig 12.

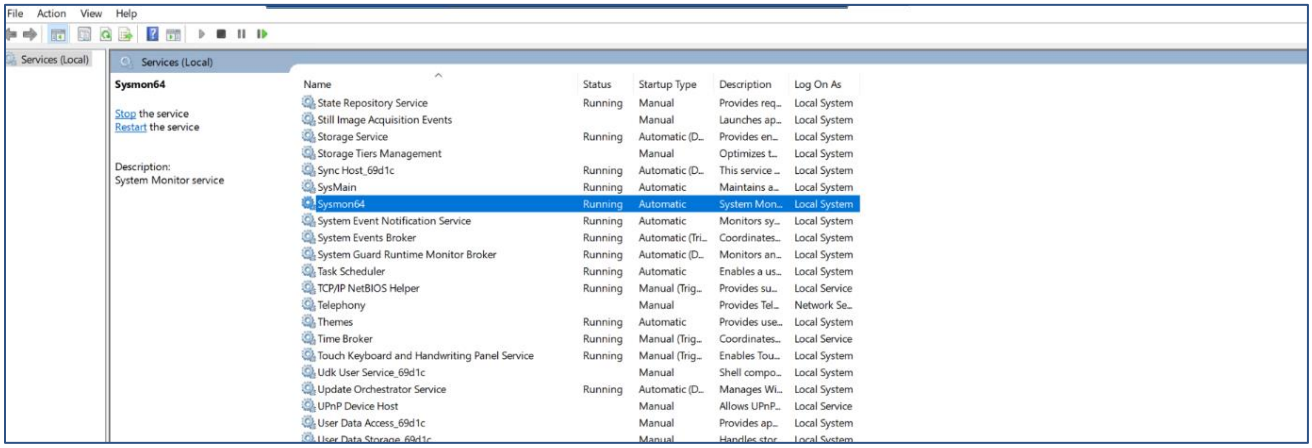


Fig 12

Sysmon allows you to craft the exact logs that you would like to monitor using its configuration file. This capability provides a lot of flexibility and verbosity in the logs recorded. Depending on the user's environment and requirement, one can craft their own Sysmon logs.

All the details on how the configuration file is modified can be understood by running C:\Windows>Sysmon64.exe -? config. The important filter tags that should be included in the configuration file are listed in the table below.

Event ID	XML Tag	Event Function
Microsoft-Windows-Sysmon 1	ProcessCreate	Process Create
Microsoft-Windows-Sysmon 3	NetworkConnect	Network connection detected
Microsoft-Windows-Sysmon 7	ImageLoad	Image loaded
Microsoft-Windows-Sysmon 10	ProcessAccess	Process accessed
Microsoft-Windows-Sysmon 11	FileCreate	File created
Microsoft-Windows-Sysmon 12	RegistryEvent	Registry object added or deleted
Microsoft-Windows-Sysmon 13	RegistryEvent	Registry value set
Microsoft-Windows-Sysmon 15	FileCreateStreamHash	File stream created
Microsoft-Windows-Sysmon 17	PipeEvent	Pipe Created
Microsoft-Windows-Sysmon 18	PipeEvent	Pipe Connected
Microsoft-Windows-Sysmon 19	WmiEvent	WmiEventFilter activity detected

Microsoft-Windows-Sysmon 20	WmiEvent	WmiEventConsumer activity detected
Microsoft-Windows-Sysmon 21	WmiEvent	WmiEventConsumerToFilter activity detected

Below is an example of a snippet from Sysmon configuration file that can be added for monitoring certain Sysmon events.

```
<!--SYSMON EVENT ID 2 : FILE CREATION TIME RETROACTIVELY CHANGED IN THE FILESYSTEM [FileCreateTime]-->
    <!--COMMENT:    [ https://attack.mitre.org/wiki/Technique/T1099 ] -->

    <!--DATA: UtcTime, ProcessGuid, ProcessId, Image, TargetFilename, CreationUtcTime, PreviousCreationUtcTime-->
<RuleGroup name="" groupRelation="or">
    <FileCreateTime onmatch="include">
        <Image name="T1099" condition="begin with">C:\Users</Image> <!--Look for timestomping in user area, usually
        <TargetFilename name="T1099" condition="end with">.exe</TargetFilename> <!--Look for backdated executables
        <Image name="T1099" condition="begin with">\Device\HarddiskVolumeShadowCopy</Image> <!--Nothing should be w
    </FileCreateTime>
</RuleGroup>

<RuleGroup name="" groupRelation="or">
    <FileCreateTime onmatch="exclude">
        <Image condition="image">OneDrive.exe</Image> <!--OneDrive constantly changes file times-->
        <Image condition="image">C:\Windows\system32\backgroundTaskHost.exe</Image>
        <Image condition="contains">setup</Image> <!--Ignore setups-->
        <Image condition="contains">install</Image> <!--Ignore setups-->
        <Image condition="contains">Update\</Image> <!--Ignore setups-->
        <Image condition="end with">redist.exe</Image> <!--Ignore setups-->
        <Image condition="is">msiexec.exe</Image> <!--Ignore setups-->
        <Image condition="is">TrustedInstaller.exe</Image> <!--Ignore setups-->
    </FileCreateTime>
</RuleGroup>

<!--SYSMON EVENT ID 3 : NETWORK CONNECTION INITIATED [NetworkConnect]-->
    <!--COMMENT:    By default this configuration takes a very conservative approach to network logging, limited to
    <!--COMMENT:    [ https://attack.mitre.org/wiki/Command and Control ] [ https://attack.mitre.org/wiki/Exfiltrat
    <!--TECHNICAL:  For the DestinationHostname, Sysmon uses the GetNameInfo API, which will often not have any inf
    <!--TECHNICAL:  For the DestinationPortName, Sysmon uses the GetNameInfo API for the friendly name of ports you
    <!--TECHNICAL:  These exe do not initiate their connections, and thus includes do not work in this section: BIT

    <!-- https://www.first.org/resources/papers/conf2017/APT-Log-Analysis-Tracking-Attack-Tools-by-Audit-Policy-and-

    <!--DATA: UtcTime, ProcessGuid, ProcessId, Image, User, Protocol, Initiated, SourceIsIpv6, SourceIp, SourceHost
<RuleGroup name="" groupRelation="or">
    <NetworkConnect onmatch="include">
        <!--Suspicious sources for network-connecting binaries-->
        <Image name="Usermode" condition="begin with">C:\Users</Image> <!--Tools downloaded by users can use other
        <Image name="Caution" condition="begin with">C:\Recycle</Image> <!--Nothing should operate from the Recycle
        <Image condition="begin with">C:\ProgramData</Image> <!--Normally, network communications should be sourced
        <Image condition="begin with">C:\Windows\Temp</Image> <!--Suspicious anything would communicate from the sy
        <Image name="Caution" condition="begin with">\</Image> <!--Devices and VSC shouldn't be executing changes |
```

Fig 13

Linux Logging

Typically, you will find Linux server logs in the /var/log directory and sub-directory. This is where syslog daemons are normally configured to write.

Auditd Logs

These logs are a special case of kernel messages designed for auditing actions such as file access. By default, services like auditd, write messages to /var/log/audit/audit.log. Auditd logs by default contain limited information and certain settings must be configured to make it more verbose.

Configure Linux Logs:

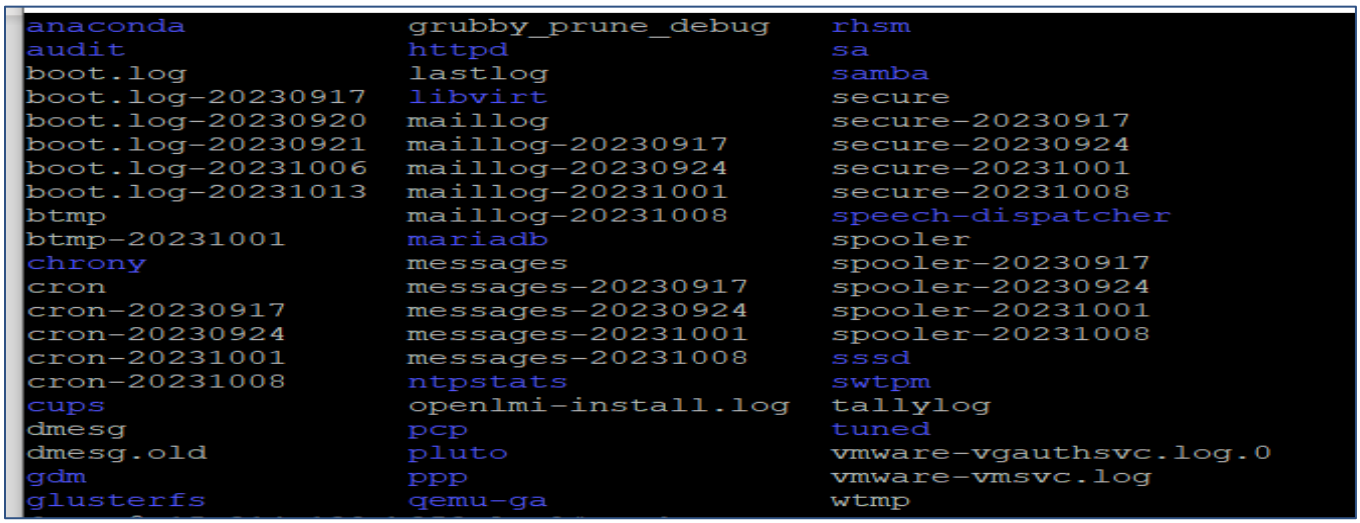
1. Make sure the refuse manual stop is set to no in the .conf file of the auditd.
2. Add the following audit rules in the paths:
   /etc/audit/audit.rules and /etc/audit/rules.d/audit.rules

```
-a exit,always -F arch=b64 -F euid=0 -S execve
-a exit,always -F arch=b32 -F euid=0 -S execve
-w /etc/hosts -p r -k hosts_file_access
-w /etc/login.defs -p w -k password_policy_modified
-w /etc/pam.d/system-auth -p w -k password_policy_modified
-w /etc/sudoers -p w -k sudoers_file_modified
```

3. Restart the auditd service. The path of the auditd service is in /usr/lib/system/system/auditd.service.

Other Linux Logs

There are also some commands run on Linux machines that cannot be monitored using auditd logs. Depending on the Linux Distribution, the logs are recorded in the appropriate output locations. Please refer to your Linux documentation for the exact output location path.



```
anaconda              grubby_prune_debug    rhsm
audit                 httpd                 sa
boot.log              lastlog               samba
boot.log-20230917     libvirt               secure
boot.log-20230920     maillog               secure-20230917
boot.log-20230921     maillog-20230917      secure-20230924
boot.log-20231006     maillog-20230924      secure-20231001
boot.log-20231013     maillog-20231001      secure-20231008
btmp                  maillog-20231008      speech-dispatcher
btmp-20231001         mariadb               spooler
chrony                messages              spooler-20230917
cron                  messages-20230917     spooler-20230924
cron-20230917         messages-20230924     spooler-20231001
cron-20230924         messages-20231001     spooler-20231008
cron-20231001         messages-20231008     sssd
cron-20231008         ntpstats              swtpm
cups                  openlmi-install.log   tallylog
dmesg                 pcp                   tuned
dmesg.old             pluto                 vmware-vgauthsvc.log.0
gdm                   ppp                   vmware-vmsvc.log
glusterfs             qemu-ga               wtmp
```

Fig 14

To increase the logging verbosity, install open-source tools like Snoopy Command Logger (not affiliated with Open Text). The snoopy logs can be monitored using a Syslog NG Deamon connector and by providing the appropriate path in the configuration. More details can be found in the ArcSight SmartConnector documentation.

Leveraging Windows ArcSight Connector

There are the following types of default Windows Event Logs:

- Application log, which tracks events that occur in a registered application.
- Security log, which tracks security changes and possible breaches in security.
- System log, which tracks system events.

Once you install the connector in your environment (refer to ArcSight SmartConnectors documentation), there are a few configuration changes that must be made, to monitor different types of logs under Windows connector.

1. Navigate to the connector path and then to the current\user\agent folder.

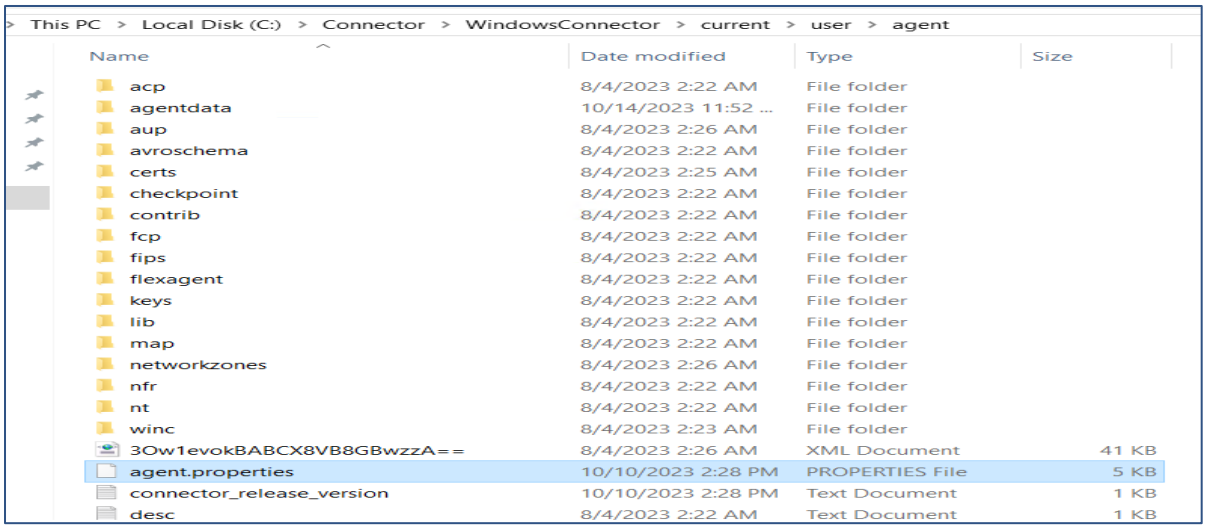


Fig 15

2. Right-click on the agent.properties file and open with Notepad++ under administrator mode.

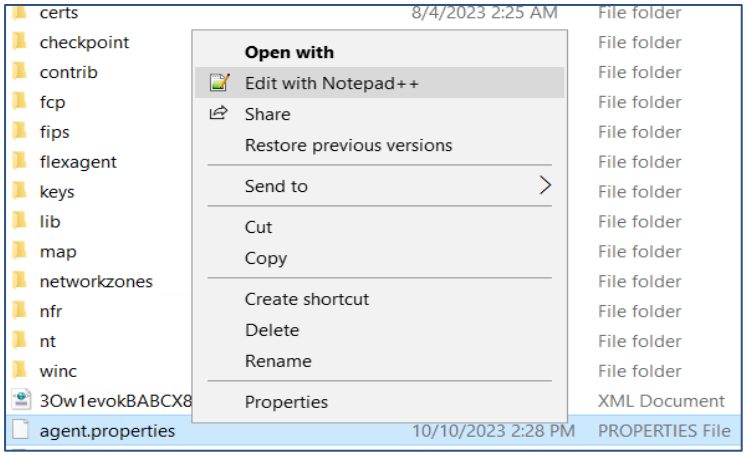


Fig 16

3. Add the following line: **Microsoft-Windows-Sysmon/Operational, Windows PowerShell, Microsoft-Windows-PowerShell/Operational, "Windows PowerShell"** in the agents[0].windowshoststable[0].eventlogtypes parameter and save it. Once completed, restart the connector.

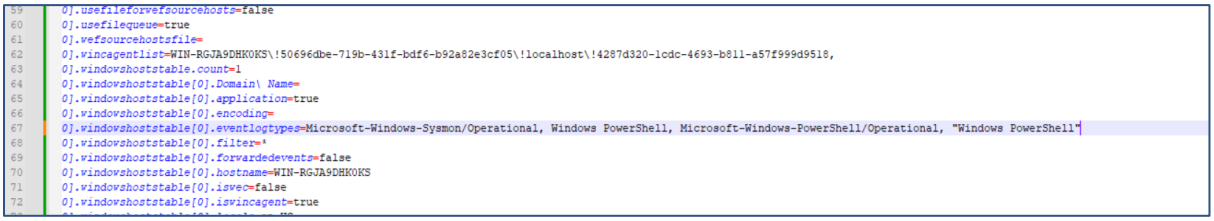


Fig 17

Use Cases

The following use cases give specific examples of ArcSight Default Content resources that rely on verbose logging.

- Windows Use Case
- PowerShell Use Case
- Sysmon Use Case
- Linux Use Case

Windows Use Case

Scenario: The adversary is trying to modify the Registry and running the below commands.

```
C:\WINDOWS\system32>reg add HKLM\SYSTEM\CurrentControlSet\Control\SecurityProviders\WDigest /v UseLogonCredential /t REG
_DWORD /d 1 /f
The operation completed successfully.
```

Fig 18

In default content, we have a rule to detect the above activity. Rule Path: /All Rules/ArcSight Foundation/MITRE ATT&CK/TA0005 Defense Evasion/T1112-Modify Registry/Registry Modified by Reg.exe (Fig 18).

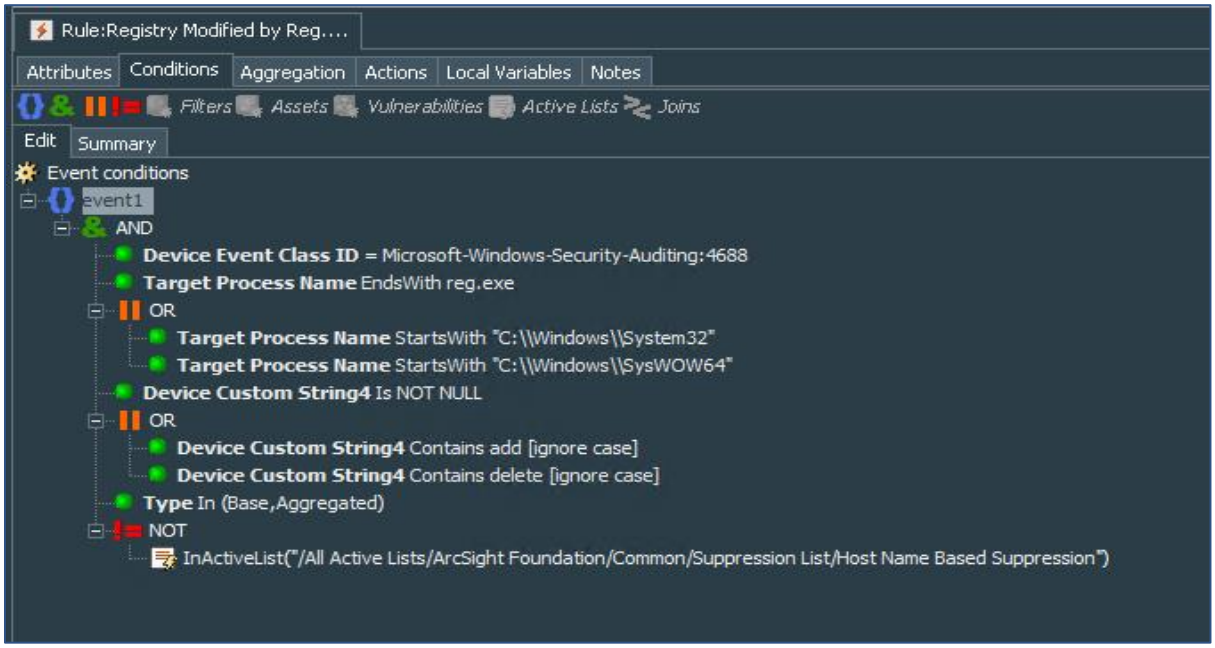


Fig 19

In this rule, the condition uses 4688 Event ID, and to capture the specific event, the fields like Target Process Name and Device Custom String4 are used.

However, the Device Custom String4 field in the ESM will not be populated unless we configure verbose logging by enabling Detailed Tracking and Audit Process Creation events. This event won't have sufficient information and the rule can never trigger despite the registry being modified. Hence, its important to enable the appropriate settings as mentioned in Windows Security Logs section of this document. Fig 20 shows the details of the event recorded in the ESM.

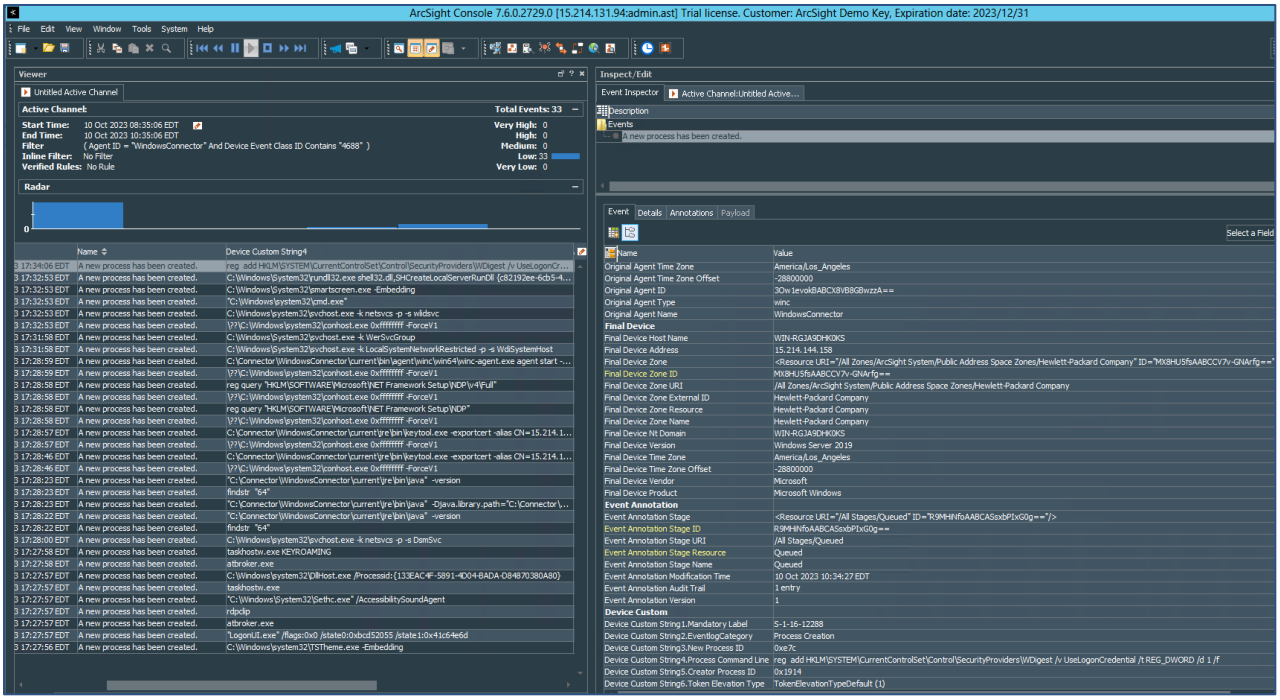


Fig 20

PowerShell Use Case

Scenario: The adversary is obfuscating the content during command execution to impede detection and is running the below commands in PowerShell.

```
Import-Module ./Invoke-Obfuscation.psd1

Invoke-Obfuscation
```

In default content, we have a rule to detect this activity. Rule Path: /All Rules/Real-time Rules/Security Threat Monitoring/Host Monitoring/Command Obfuscation Using PowerShell (Fig 21)

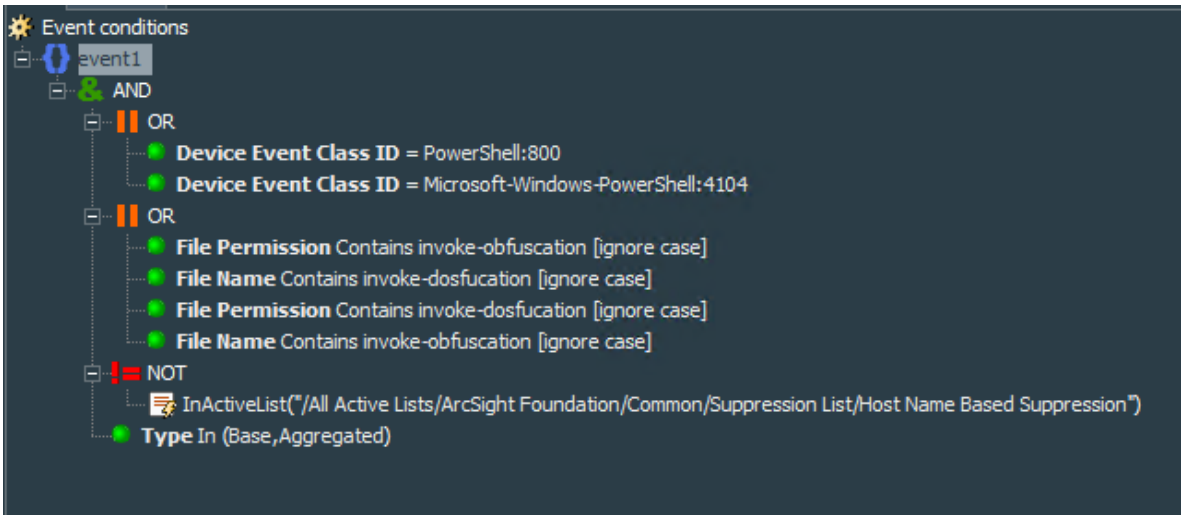


Fig 21

This rule condition uses PowerShell event IDs like 800 and 4104. These event IDs are not logged in the event viewer unless enabled. Refer to the section [PowerShell Logging](). The below screenshot shows the details recorded for PowerShell event ID 4104 in windows event viewer (Fig 22).

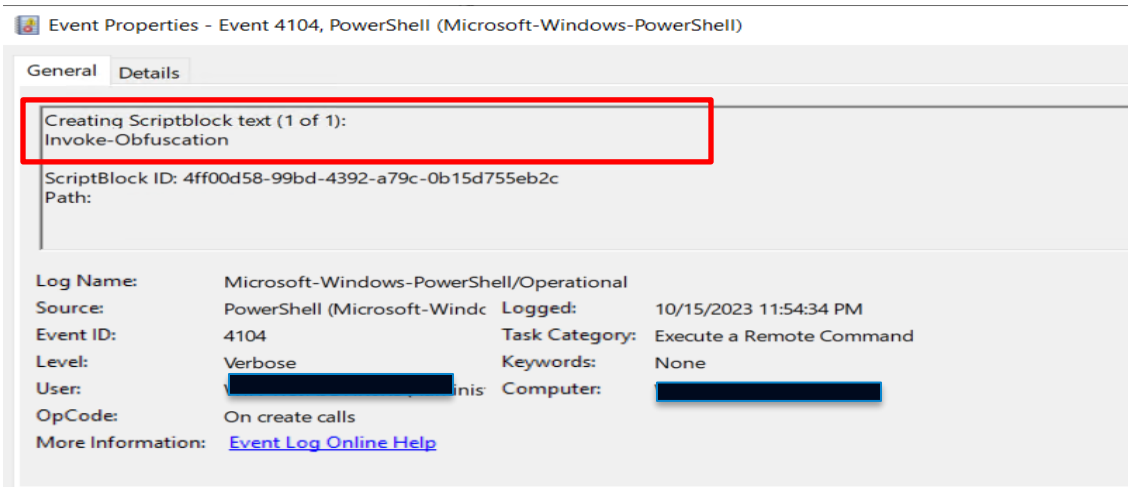


Fig 22

This in turn will populate the fields like Device Event Class ID, File Permission, and File Name in the ESM and trigger the rule (Fig 23).

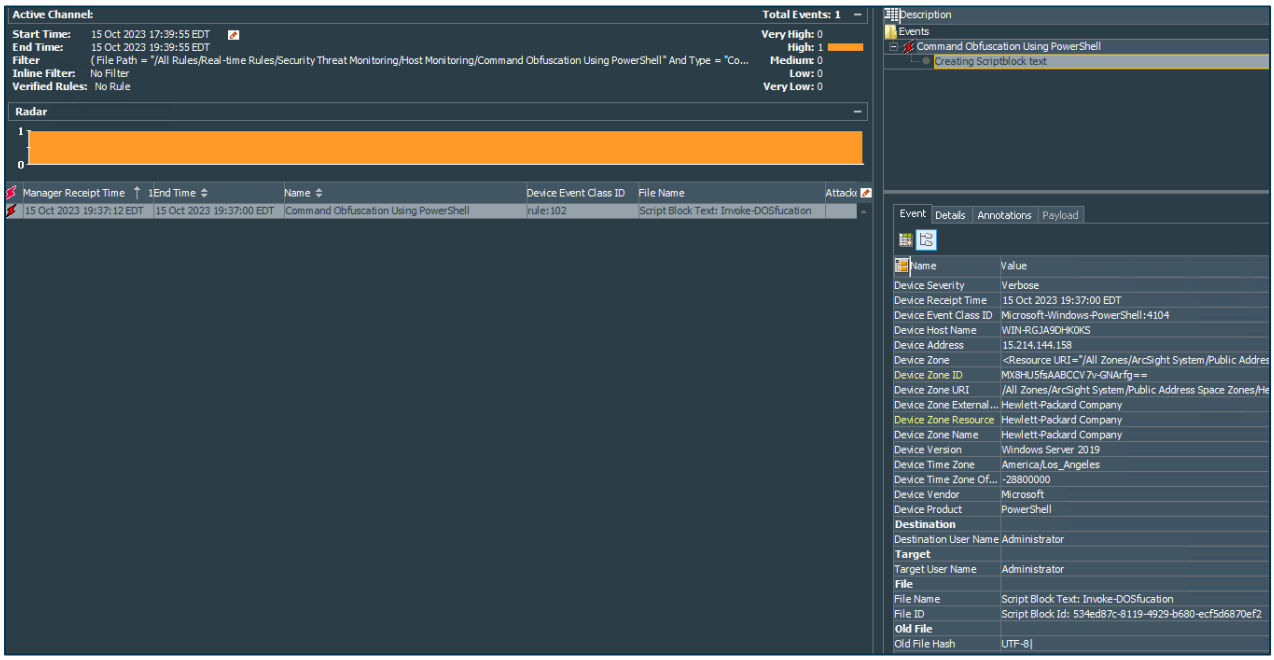


Fig 23

Sysmon Use Case

Scenario: The adversary is trying to gain access to credentials stored in group policy preferences in Windows and is trying to run the below commands to view the credentials.

```
findstr /S /I cpassword \\sysvol\policies\*.xml
```

In default content, we have a rule to detect this activity. Rule Path: /All Rules/ArcSight Foundation/MITRE ATT&CK/TA0006 Credential Access/T1552-Unsecured Credentials/T1552.006-Group Policy Preferences/Credentials in Group Policy Preferences.

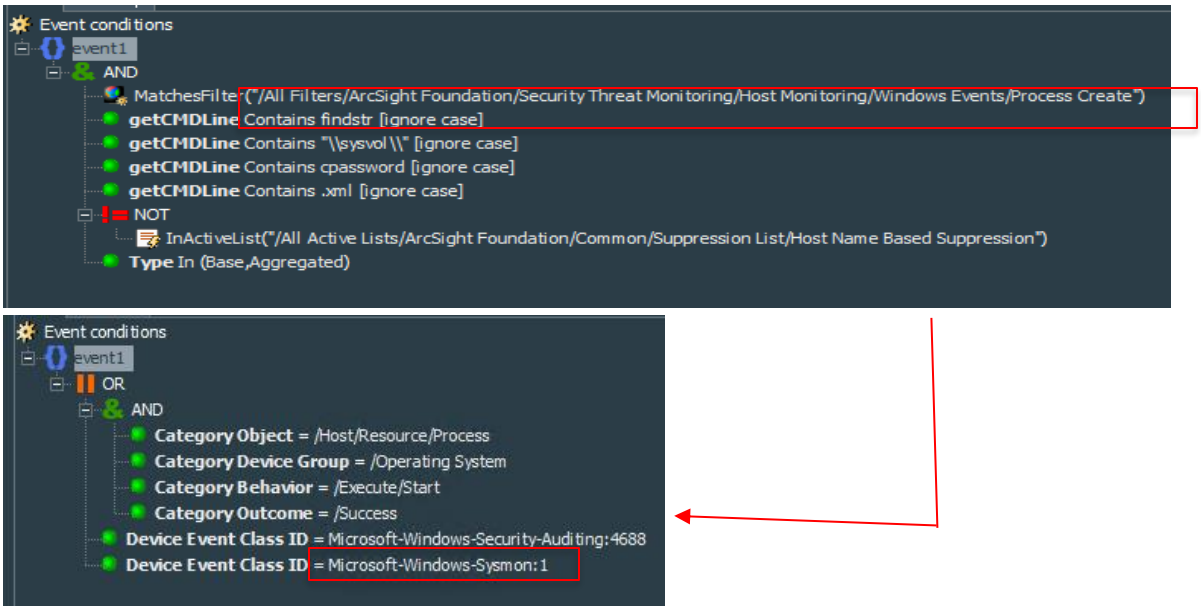


Fig 24

The rule contains a filter called 'process create' which in turn uses Sysmon event ID 1 in an either-or condition. Sysmon must be installed to record Sysmon events. Please refer Sysmon Logging for further reading. Fig 25 are the details recorded when the above Sysmon event is recorded in the windows event viewer.

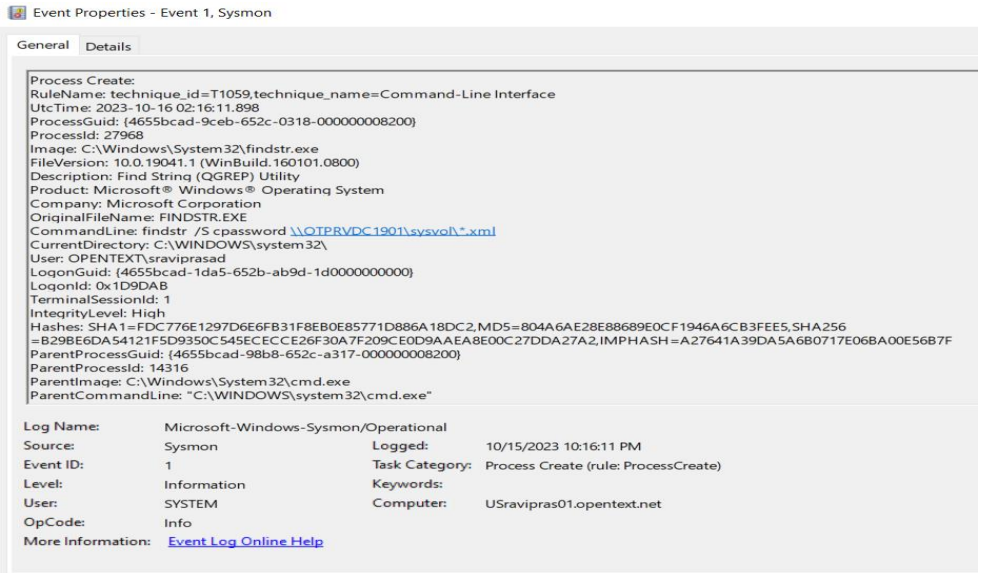


Fig 25

Linux Use Case

Scenario: The adversary is trying to discover information about remote systems using Linux commands.

In default content, we have rule to detect this activity. Rule Path: /All Rules/ArcSight Foundation/MITRE ATT&CK/TA0007 Discovery/T1018-Remote System Discovery/Suspicious Remote System Discovery Commands Entered on Linux (Fig 26).

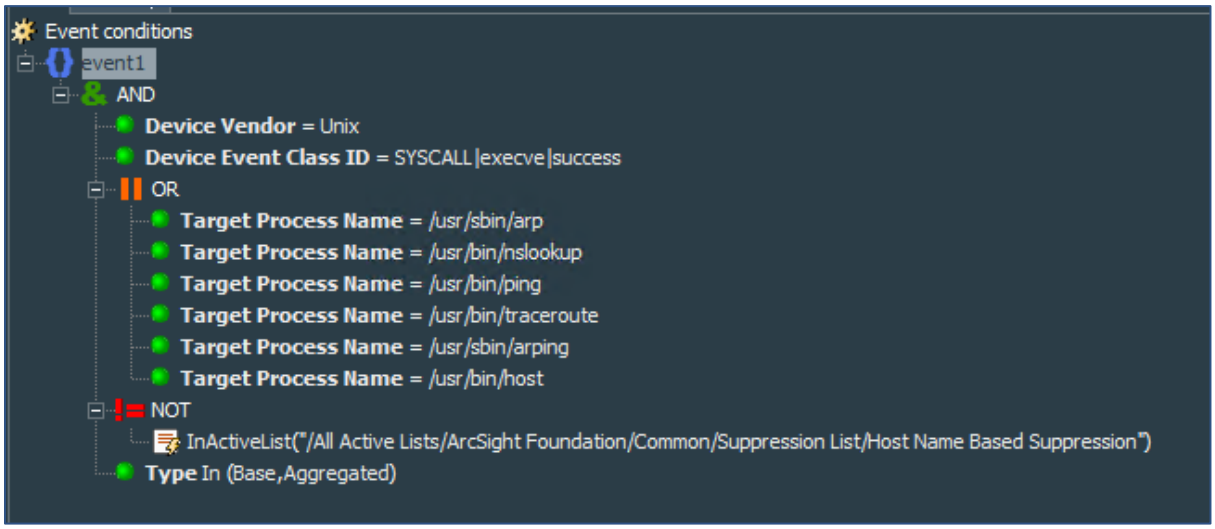


Fig 26

Most of the Linux-based rules in Default Content depend on auditd logs. However, the target process name field is empty unless verbose logging is enabled in the Linux Machine. Refer to [Linux Logging] for more information. When the audit rules are enabled, the appropriate fields get populated, which in turn triggers the rule.

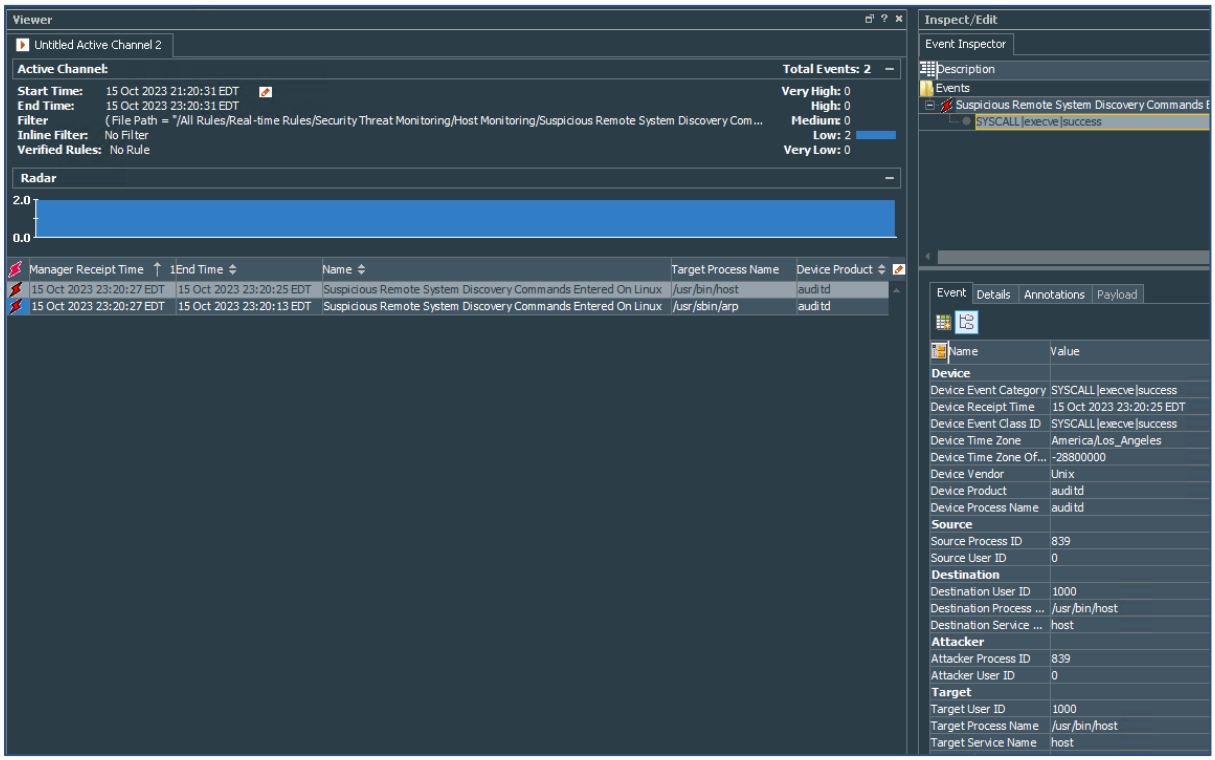


Fig 27

Conclusion

It's always a challenge to understand the level of verbosity needed in the logs to ensure the right and sufficient information is recorded. It is also important to strike a balance as to what needs to be enabled and disabled to keep the performance of the system unaffected. Hence, to ease some part of this detailed process, this document helps as an easy guide to set up the logs and make use of MITRE ATT&CK related default content in the most effective manner possible.

Useful Links:

- ESM Default Content ArcSight Marketplace
- ArcSight Connectors Documentation
- MITRE Coverage in Default Content
- Contact Us

Legal Notice