Best Practice

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

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:

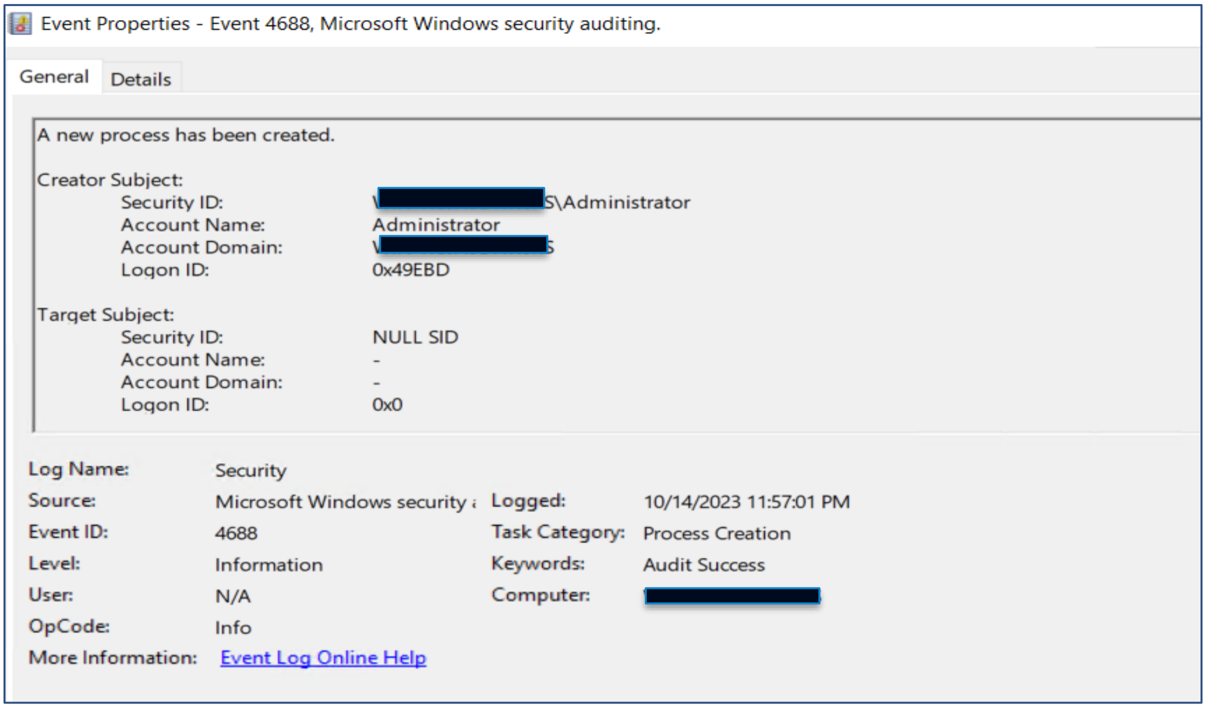
- Widows Event ID: 4688
- Other Windows Security Logs

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.



Event Properties - Event 4688, Microsoft Windows security auditing.

General Details

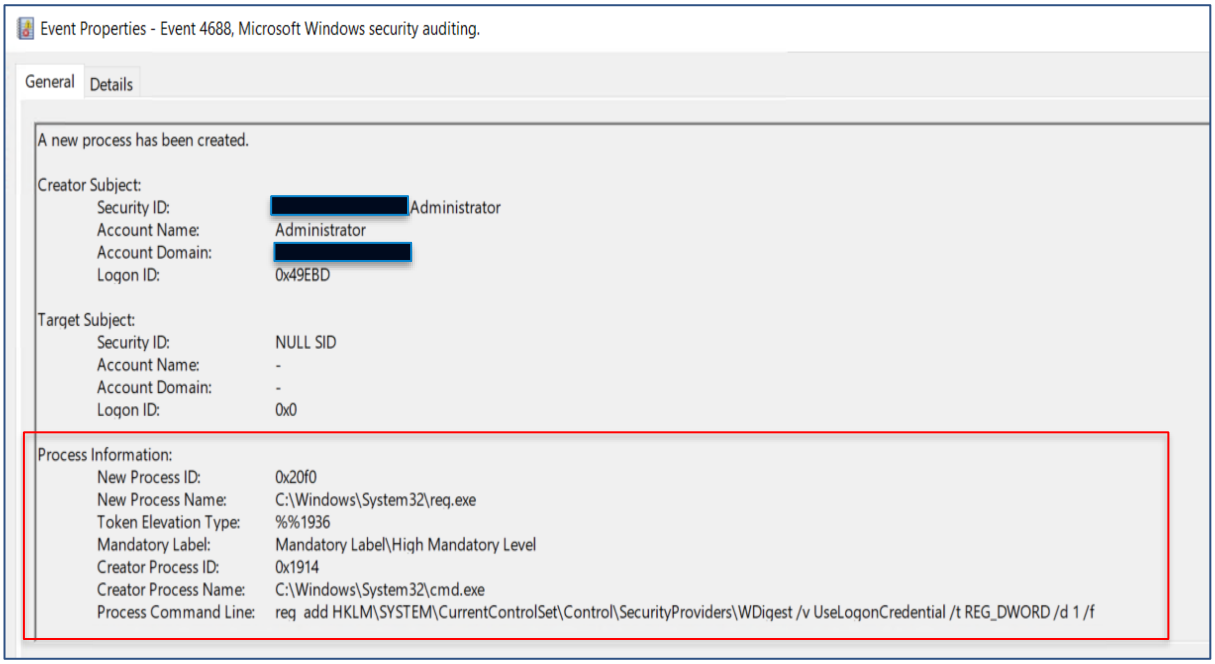
```
A new process has been created.

Creator Subject:
        Security ID:        [redacted]\Administrator
        Account Name:       Administrator
        Account Domain:     [redacted]
        Logon ID:           0x49EBD

Target Subject:
        Security ID:        NULL SID
        Account Name:       -
        Account Domain:     -
        Logon ID:           0x0
```

```
Log Name:           Security
Source:             Microsoft Windows security a   Logged:          10/14/2023 11:57:01 PM
Event ID:           4688                           Task Category:   Process Creation
Level:              Information                    Keywords:        Audit Success
User:               N/A                            Computer:        [redacted]
OpCode:             Info
More Information:   Event Log Online Help
```

Fig 1



Event Properties - Event 4688, Microsoft Windows security auditing.

General Details

```
A new process has been created.

Creator Subject:
        Security ID:        [redacted]Administrator
        Account Name:       Administrator
        Account Domain:     [redacted]
        Logon ID:           0x49EBD

Target Subject:
        Security ID:        NULL SID
        Account Name:       -
        Account Domain:     -
        Logon ID:           0x0

Process Information:
        New Process ID:         0x20f0
        New Process Name:       C:\Windows\System32\reg.exe
        Token Elevation Type:   %%1936
        Mandatory Label:        Mandatory Label\High Mandatory Level
        Creator Process ID:     0x1914
        Creator Process Name:   C:\Windows\System32\cmd.exe
        Process Command Line:   reg  add HKLM\SYSTEM\CurrentControlSet\Control\SecurityProviders\WDigest /v UseLogonCredential /t REG_DWORD /d 1 /f
```

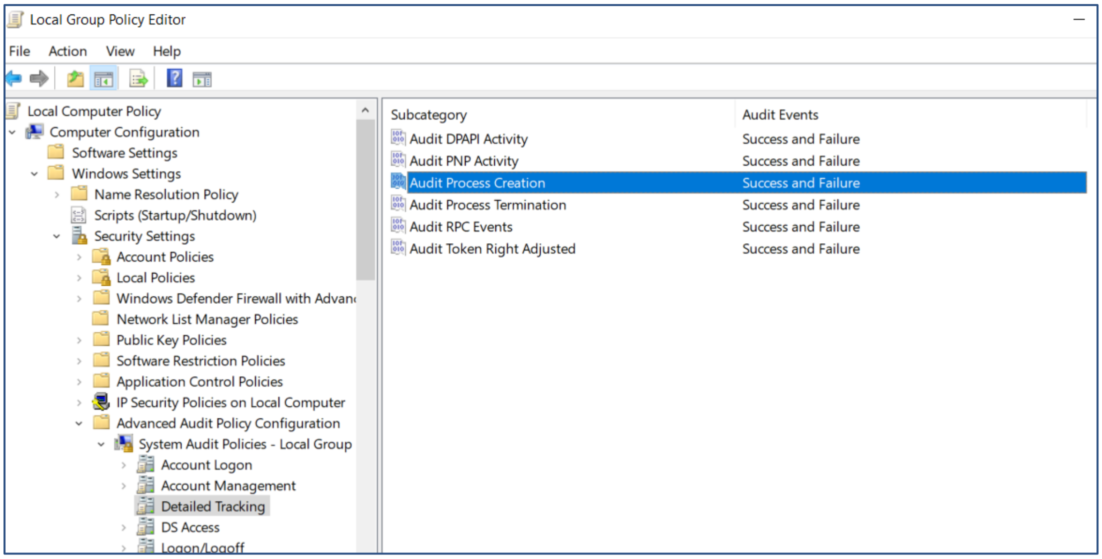
Fig 2

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.


Local Group Policy Editor
File
Action
View
Help
Local Computer Policy
Computer Configuration
Software Settings
Windows Settings
Name Resolution Policy
Scripts (Startup/Shutdown)
Security Settings
Account Policies
Local Policies
Windows Defender Firewall with Advanc
Network List Manager Policies
Public Key Policies
Software Restriction Policies
Application Control Policies
IP Security Policies on Local Computer
Advanced Audit Policy Configuration
System Audit Policies - Local Group
Account Logon
Account Management
Detailed Tracking
DS Access
Logon/Logoff

Subcategory	Audit Events
Audit DPAPI Activity	Success and Failure
Audit PNP Activity	Success and Failure
Audit Process Creation	Success and Failure
Audit Process Termination	Success and Failure
Audit RPC Events	Success and Failure
Audit Token Right Adjusted	Success and Failure

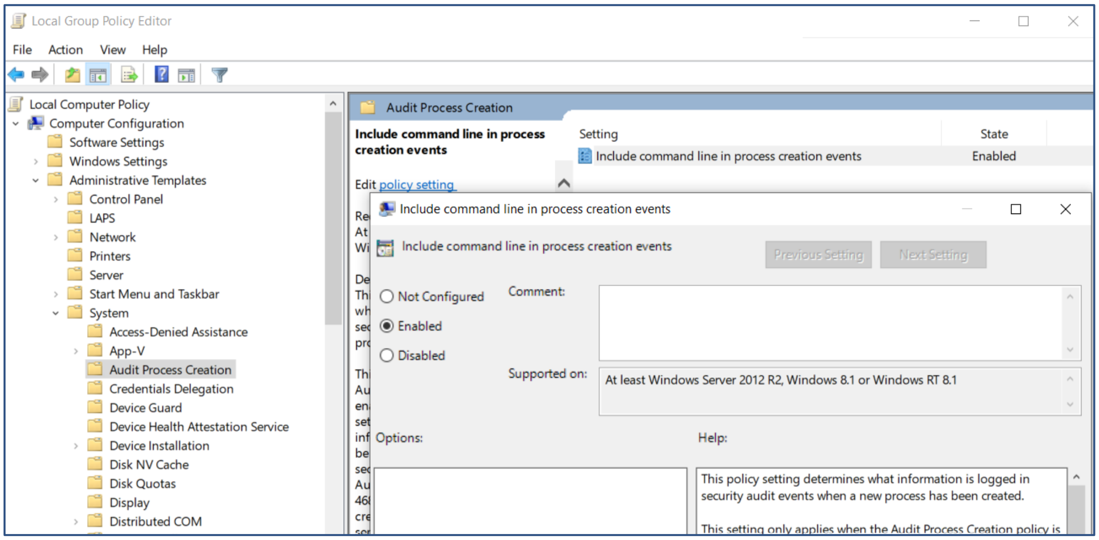
Fig 3

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.



Setting	State
Include command line in process creation events	Enabled

Include command line in process creation events

○ Not Configured
● Enabled
○ Disabled

Comment:

Supported on: At least Windows Server 2012 R2, Windows 8.1 or Windows RT 8.1

This policy setting determines what information is logged in security audit events when a new process has been created.

This setting only applies when the Audit Process Creation policy is

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

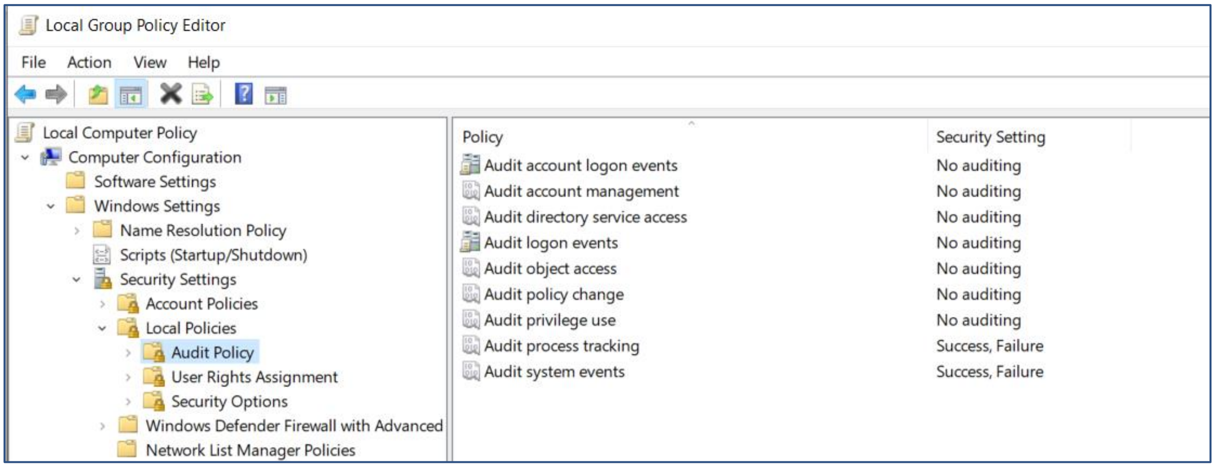
Group Policy client-side extensions that do not process policy on a background update cycle but do process policy at a computer start up.

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.

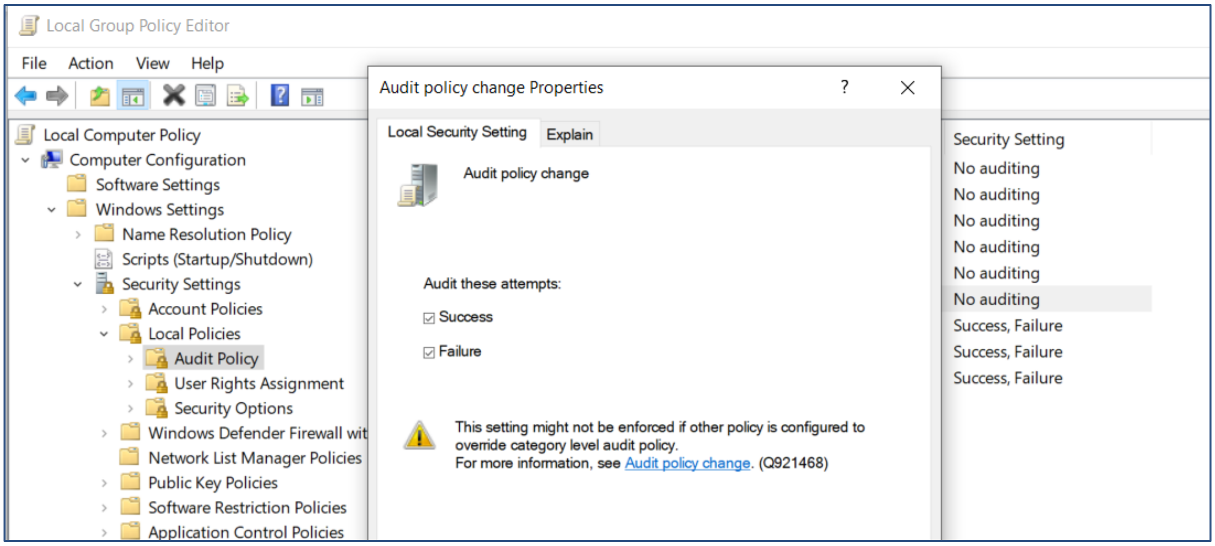
Enable Audit Policy Events in the Group Policy Editor

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.



Policy	Security Setting
Audit account logon events	No auditing
Audit account management	No auditing
Audit directory service access	No auditing
Audit logon events	No auditing
Audit object access	No auditing
Audit policy change	No auditing
Audit privilege use	No auditing
Audit process tracking	Success, Failure
Audit system events	Success, Failure

Fig 5



Audit policy change

Audit these attempts:

☑ Success

☑ Failure

This setting might not be enforced if other policy is configured to override category level audit policy.
For more information, see Audit policy change. (Q921468)

Fig 6

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

PowerShell Logs

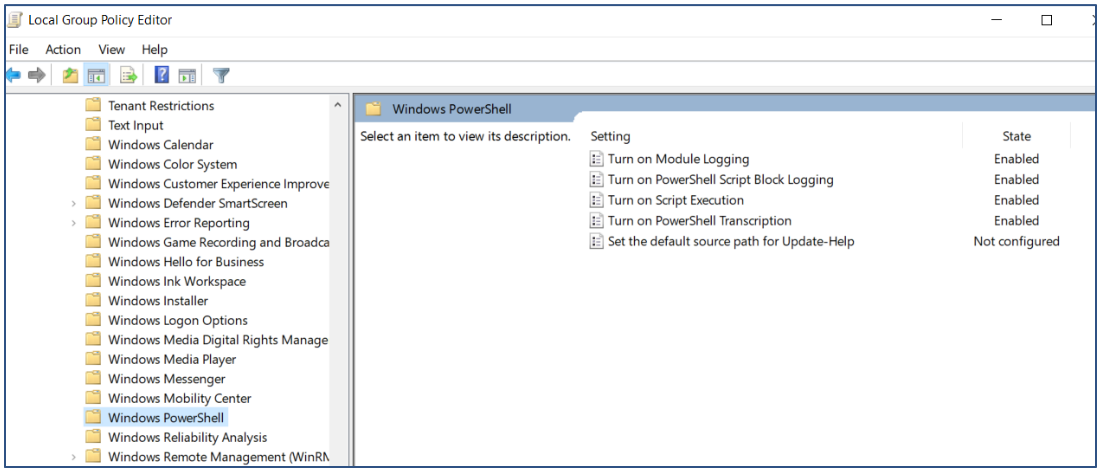
ArcSight Default Content relies on PowerShell logs to trigger alerts to potential threats to your system.

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



Select an item to view its description.

Setting	State
Turn on Module Logging	Enabled
Turn on PowerShell Script Block Logging	Enabled
Turn on Script Execution	Enabled
Turn on PowerShell Transcription	Enabled
Set the default source path for Update-Help	Not configured

Fig 7

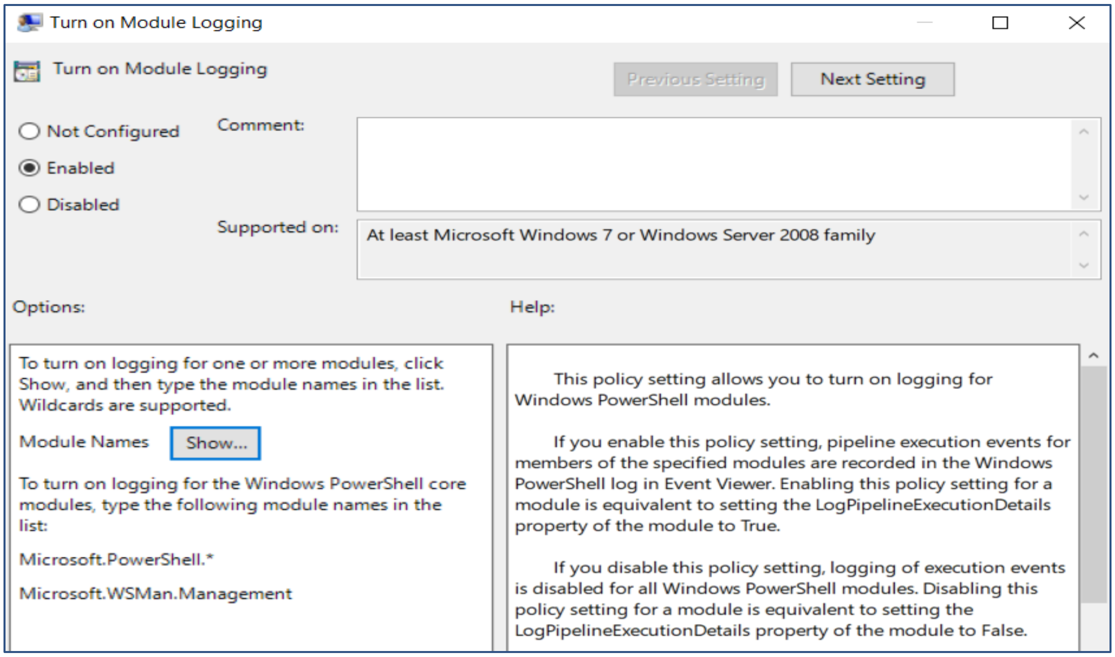
2. Select the settings and enable Module, Script Block and Transcription logging.

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 *



Turn on Module Logging

Turn on Module Logging

○ Not Configured
● Enabled
○ Disabled

Comment:

Supported on: At least Microsoft Windows 7 or Windows Server 2008 family

Options:

To turn on logging for one or more modules, click Show, and then type the module names in the list. Wildcards are supported.

Module Names Show...

To turn on logging for the Windows PowerShell core modules, type the following module names in the list:

Microsoft.PowerShell.*

Microsoft.WSMan.Management

Help:

This policy setting allows you to turn on logging for Windows PowerShell modules.

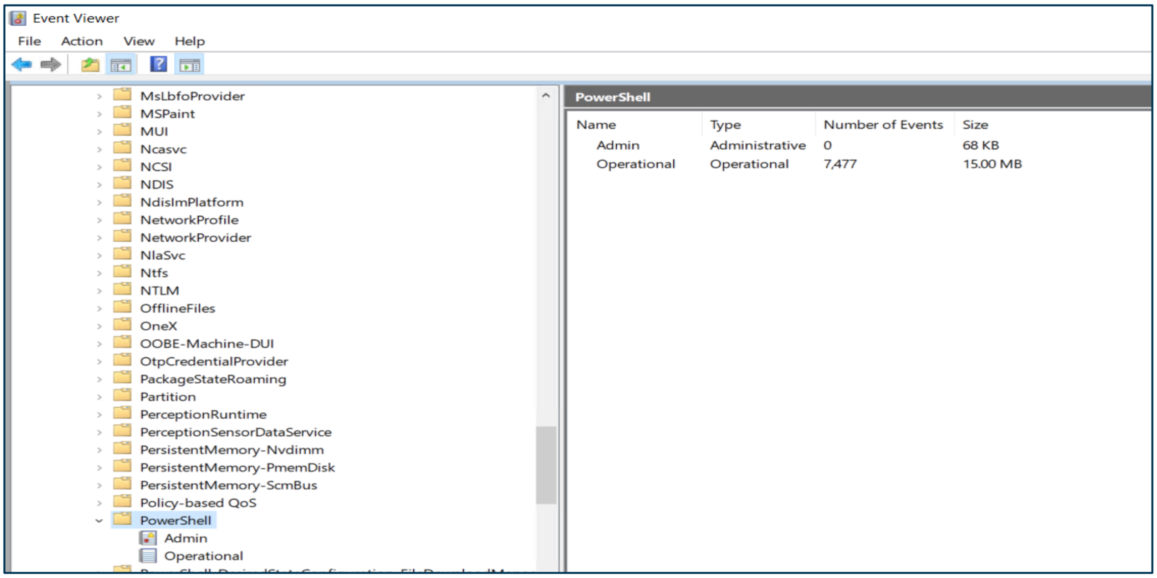
If you enable this policy setting, pipeline execution events for members of the specified modules are recorded in the Windows PowerShell log in Event Viewer. Enabling this policy setting for a module is equivalent to setting the LogPipelineExecutionDetails property of the module to True.

If you disable this policy setting, logging of execution events is disabled for all Windows PowerShell modules. Disabling this policy setting for a module is equivalent to setting the LogPipelineExecutionDetails property of the module to False.

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).

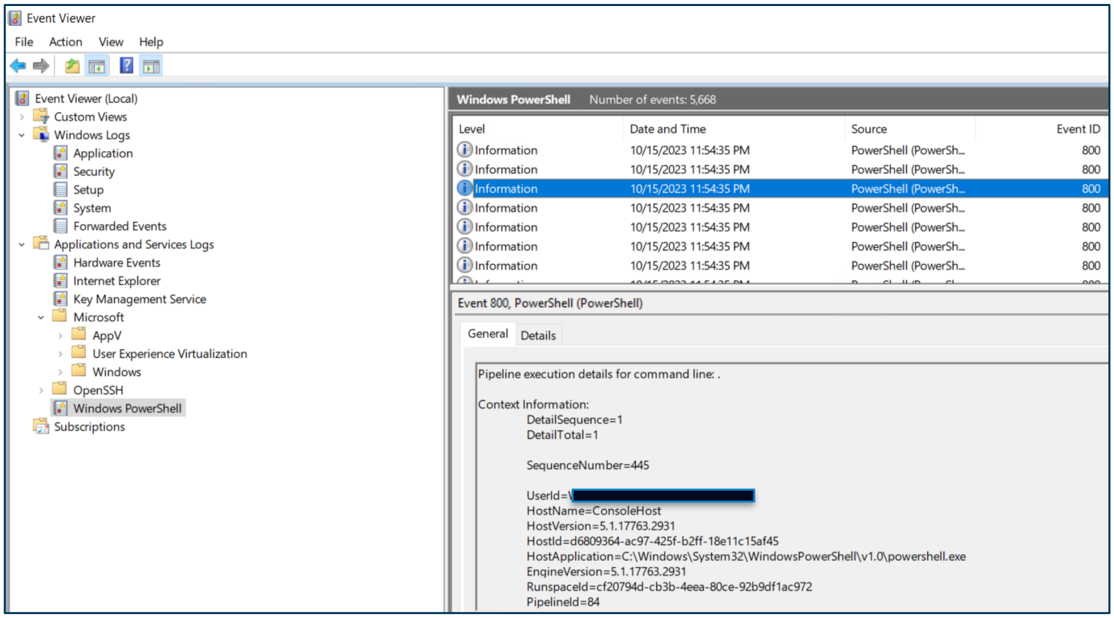


MsLbfoProvider
MSPaint
MUI
Ncasvc
NCSI
NDIS
NdisImPlatform
NetworkProfile
NetworkProvider
NlaSvc
Ntfs
NTLM
OfflineFiles
OneX
OOBE-Machine-DUI
OtpCredentialProvider
PackageStateRoaming
Partition
PerceptionRuntime
PerceptionSensorDataService
PersistentMemory-Nvdimm
PersistentMemory-PmemDisk
PersistentMemory-ScmBus
Policy-based QoS
PowerShell
Admin
Operational

PowerShell

Name	Type	Number of Events	Size
Admin	Administrative	0	68 KB
Operational	Operational	7,477	15.00 MB

Fig 9



Windows PowerShell   Number of events: 5,668

Level	Date and Time	Source	Event ID
Information	10/15/2023 11:54:35 PM	PowerShell (PowerSh...	800
Information	10/15/2023 11:54:35 PM	PowerShell (PowerSh...	800
Information	10/15/2023 11:54:35 PM	PowerShell (PowerSh...	800
Information	10/15/2023 11:54:35 PM	PowerShell (PowerSh...	800
Information	10/15/2023 11:54:35 PM	PowerShell (PowerSh...	800
Information	10/15/2023 11:54:35 PM	PowerShell (PowerSh...	800
Information	10/15/2023 11:54:35 PM	PowerShell (PowerSh...	800

Event 800, PowerShell (PowerShell)

```
Pipeline execution details for command line: .

Context Information:
        DetailSequence=1
        DetailTotal=1

        SequenceNumber=445

        UserId=\[redacted]
        HostName=ConsoleHost
        HostVersion=5.1.17763.2931
        HostId=d6809364-ac97-425f-b2ff-18e11c15af45
        HostApplication=C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe
        EngineVersion=5.1.17763.2931
        RunspaceId=cf20794d-cb3b-4eea-80ce-92b9df1ac972
        PipelineId=84
```

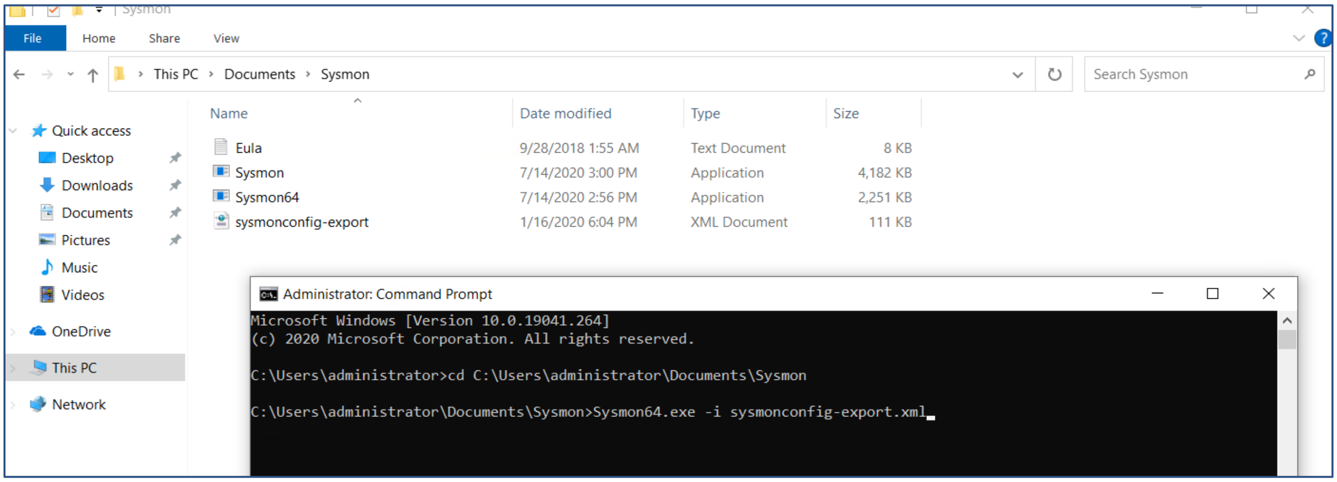
Fig 10

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.



Name	Date modified	Type	Size
Eula	9/28/2018 1:55 AM	Text Document	8 KB
Sysmon	7/14/2020 3:00 PM	Application	4,182 KB
Sysmon64	7/14/2020 2:56 PM	Application	2,251 KB
sysmonconfig-export	1/16/2020 6:04 PM	XML Document	111 KB

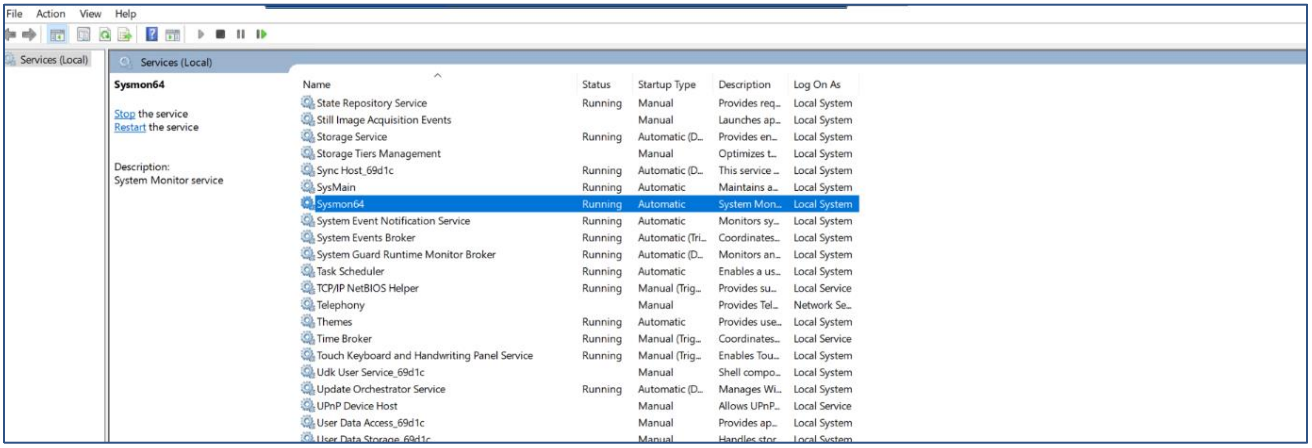
```
Microsoft Windows [Version 10.0.19041.264]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\administrator>cd C:\Users\administrator\Documents\Sysmon

C:\Users\administrator\Documents\Sysmon>Sysmon64.exe -i sysmonconfig-export.xml
```

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.



Sysmon64

Stop the service
Restart the service

Description:
System Monitor service

Name	Status	Startup Type	Description	Log On As
State Repository Service	Running	Manual	Provides req...	Local System
Still Image Acquisition Events		Manual	Launches ap...	Local System
Storage Service	Running	Automatic (D...	Provides en...	Local System
Storage Tiers Management		Manual	Optimizes t...	Local System
Sync Host_69d1c	Running	Automatic (D...	This service ...	Local System
SysMain	Running	Automatic	Maintains a...	Local System
Sysmon64	Running	Automatic	System Mon...	Local System
System Event Notification Service	Running	Automatic	Monitors sy...	Local System
System Events Broker	Running	Automatic (Tri...	Coordinates...	Local System
System Guard Runtime Monitor Broker	Running	Automatic (D...	Monitors an...	Local System
Task Scheduler	Running	Automatic	Enables a us...	Local System
TCP/IP NetBIOS Helper	Running	Manual (Trig...	Provides su...	Local Service
Telephony		Manual	Provides Tel...	Network Se...
Themes	Running	Automatic	Provides use...	Local System
Time Broker	Running	Manual (Trig...	Coordinates...	Local Service
Touch Keyboard and Handwriting Panel Service	Running	Manual (Trig...	Enables Tou...	Local System
Udk User Service_69d1c		Manual	Shell compo...	Local System
Update Orchestrator Service	Running	Automatic (D...	Manages Wi...	Local System
UPnP Device Host		Manual	Allows UPnP...	Local Service
User Data Access_69d1c		Manual	Provides ap...	Local System
User Data Storage_69d1c		Manual	Handles stor...	Local System

Fig 12

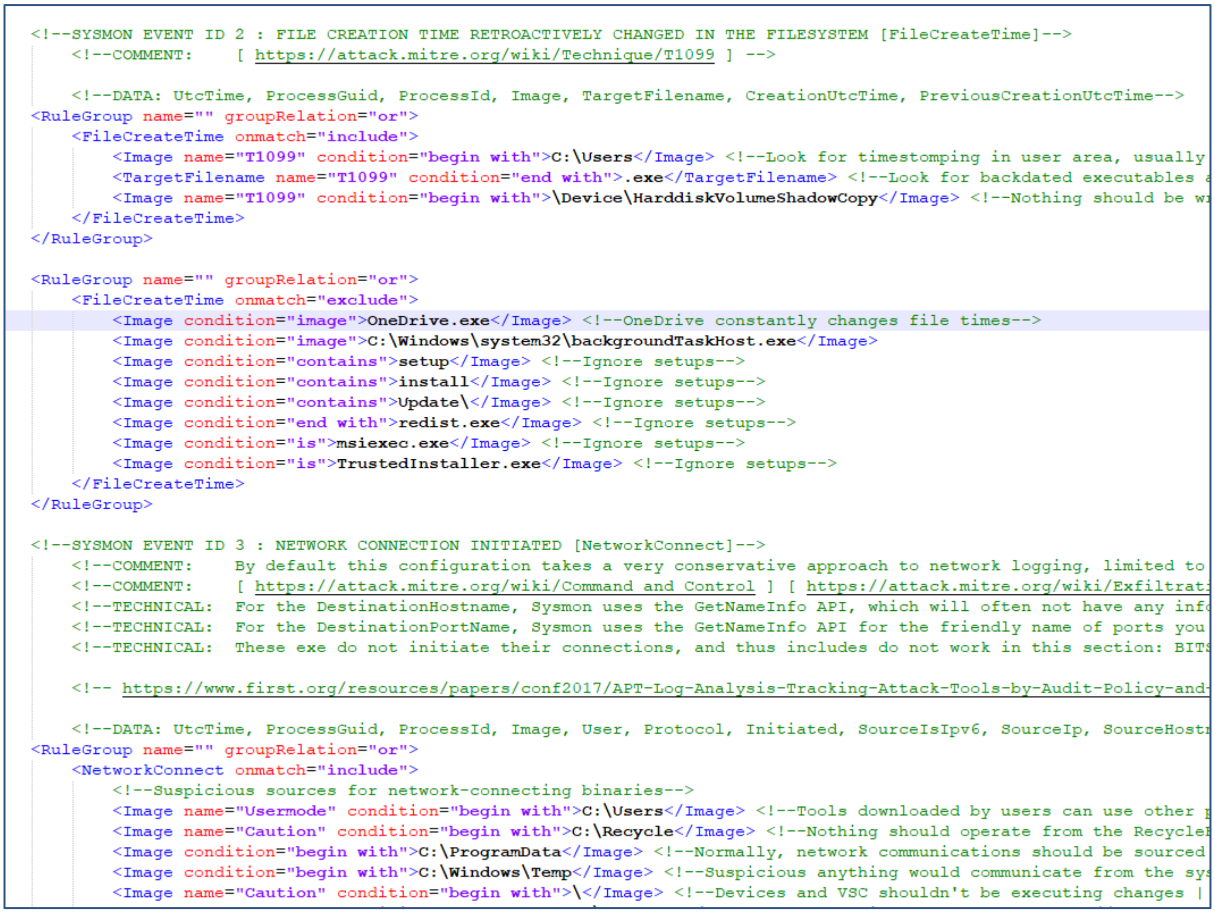
Sysmon allows you to craft the exact logs that you would like to monitor using its configuration file. This capability provides a lot of flexibility and verbosity in the logs recorded. Depending on the user’s environment and requirement, one can craft their own Sysmon logs.

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 a
        <Image name="T1099" condition="begin with">\Device\HarddiskVolumeShadowCopy</Image> <!--Nothing should be wr
    </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/Exfiltrati
    <!--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: BITS

    <!-- 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, SourceHostn
<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 p
        <Image name="Caution" condition="begin with">C:\Recycle</Image> <!--Nothing should operate from the RecycleB
        <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 sys
        <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:

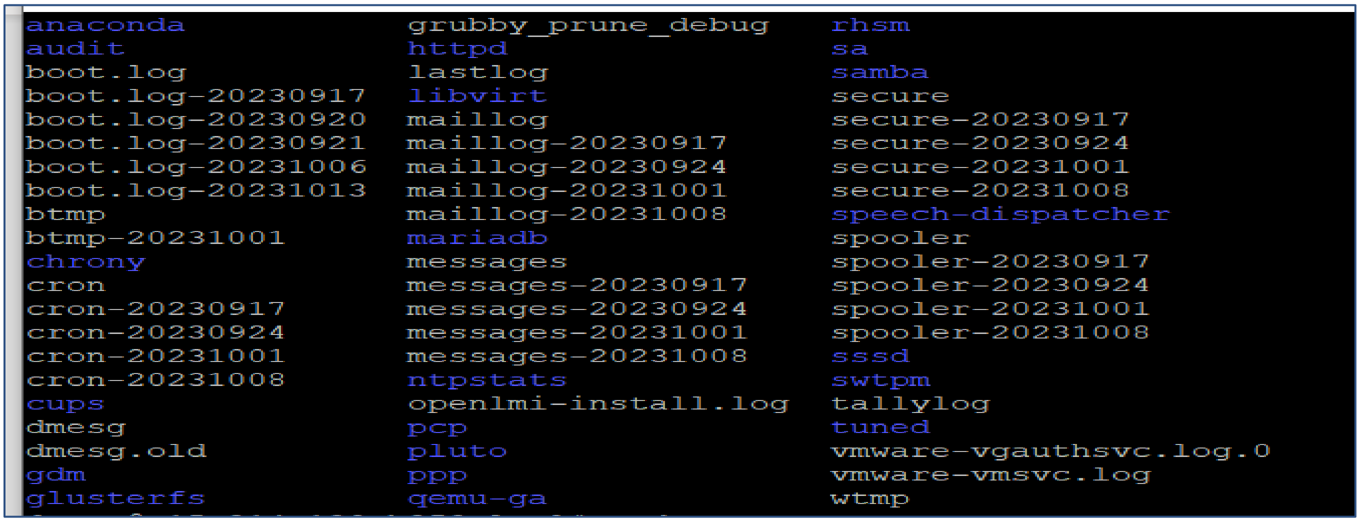
1. Make sure the refuse manual stop is set to no in the .conf file of the auditd.
2. Add the following audit rules in the paths:
   /etc/audit/audit.rules and /etc/audit/rules.d/audit.rules

```
-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 File connector and by providing the appropriate path in the configuration. More details can be found in the ArcSight SmartConnector documentation.

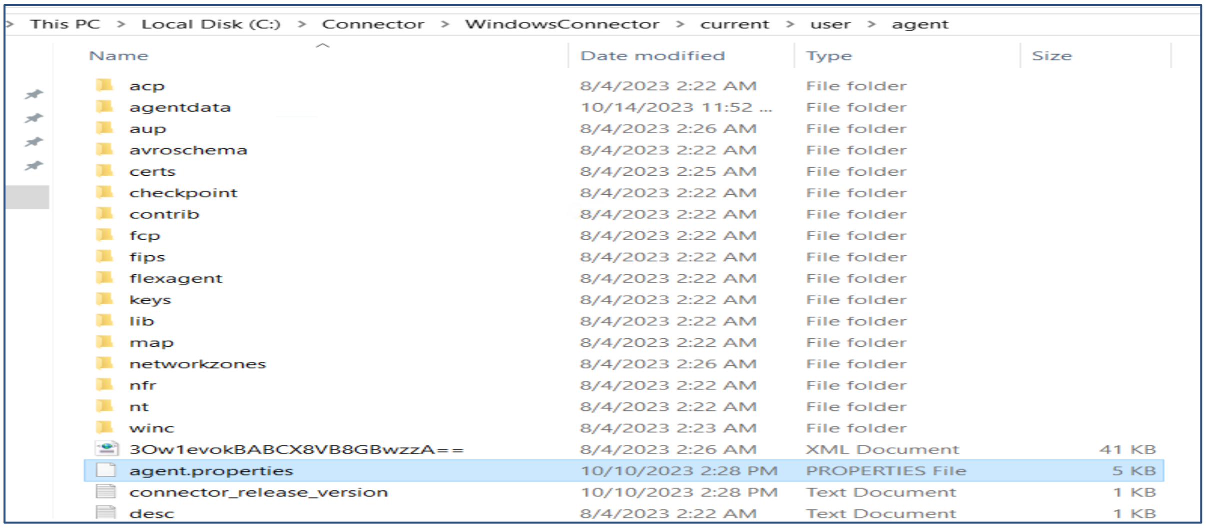
Leveraging Windows ArcSight Connector

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.



This PC > Local Disk (C:) > Connector > WindowsConnector > current > user > agent

Name	Date modified	Type	Size
acp	8/4/2023 2:22 AM	File folder	
agentdata	10/14/2023 11:52 ...	File folder	
aup	8/4/2023 2:26 AM	File folder	
avroschema	8/4/2023 2:22 AM	File folder	
certs	8/4/2023 2:25 AM	File folder	
checkpoint	8/4/2023 2:22 AM	File folder	
contrib	8/4/2023 2:22 AM	File folder	
fcp	8/4/2023 2:22 AM	File folder	
fips	8/4/2023 2:22 AM	File folder	
flexagent	8/4/2023 2:22 AM	File folder	
keys	8/4/2023 2:22 AM	File folder	
lib	8/4/2023 2:22 AM	File folder	
map	8/4/2023 2:22 AM	File folder	
networkzones	8/4/2023 2:26 AM	File folder	
nfr	8/4/2023 2:22 AM	File folder	
nt	8/4/2023 2:22 AM	File folder	
winc	8/4/2023 2:23 AM	File folder	
3Ow1evokBABCX8VB8GBwzzA==	8/4/2023 2:26 AM	XML Document	41 KB
agent.properties	10/10/2023 2:28 PM	PROPERTIES File	5 KB
connector_release_version	10/10/2023 2:28 PM	Text Document	1 KB
desc	8/4/2023 2:22 AM	Text Document	1 KB

Fig 15

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

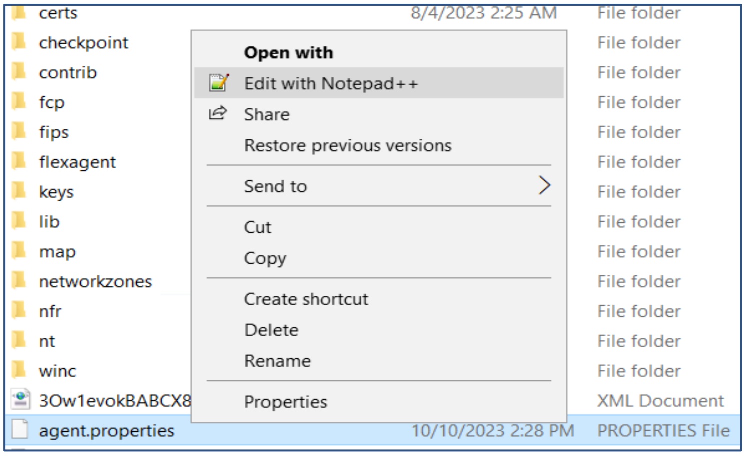
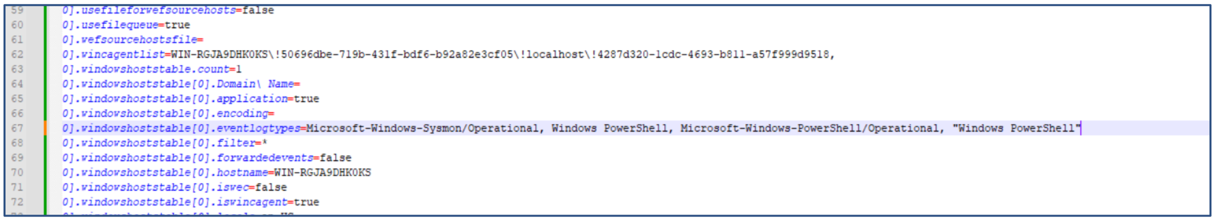


Fig 16

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.



```
0].usefileforwefsourcehosts=false
0].usefilequeue=true
0].wefsourcehostsfile=
0].wincagentlist=WIN-RGJA9DHK0KS\!50696dbe-719b-431f-bdf6-b92a82e3cf05\!localhost\!4287d320-1cdc-4693-b811-a57f999d9518,
0].windowshoststable.count=1
0].windowshoststable[0].Domain\ Name=
0].windowshoststable[0].application=true
0].windowshoststable[0].encoding=
0].windowshoststable[0].eventlogtypes=Microsoft-Windows-Sysmon/Operational, Windows PowerShell, Microsoft-Windows-PowerShell/Operational, "Windows PowerShell"
0].windowshoststable[0].filter=*
0].windowshoststable[0].forwardedevents=false
0].windowshoststable[0].hostname=WIN-RGJA9DHK0KS
0].windowshoststable[0].iswec=false
0].windowshoststable[0].iswincagent=true
```

Fig 17

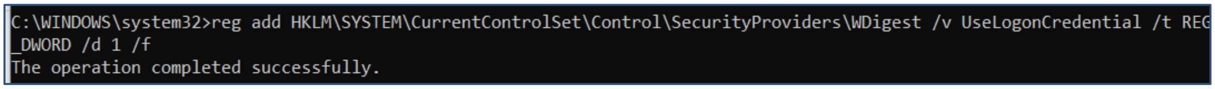
Use Cases

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

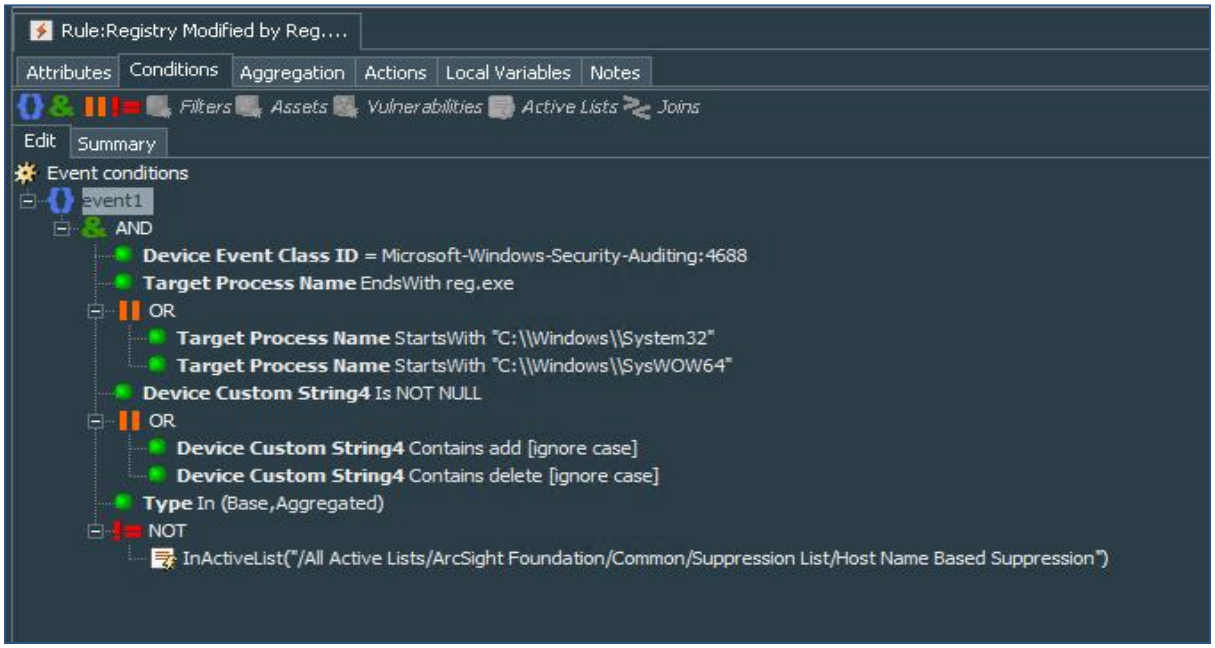
Scenario: The adversary is trying to modify the Registry and running the below commands.



```
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).

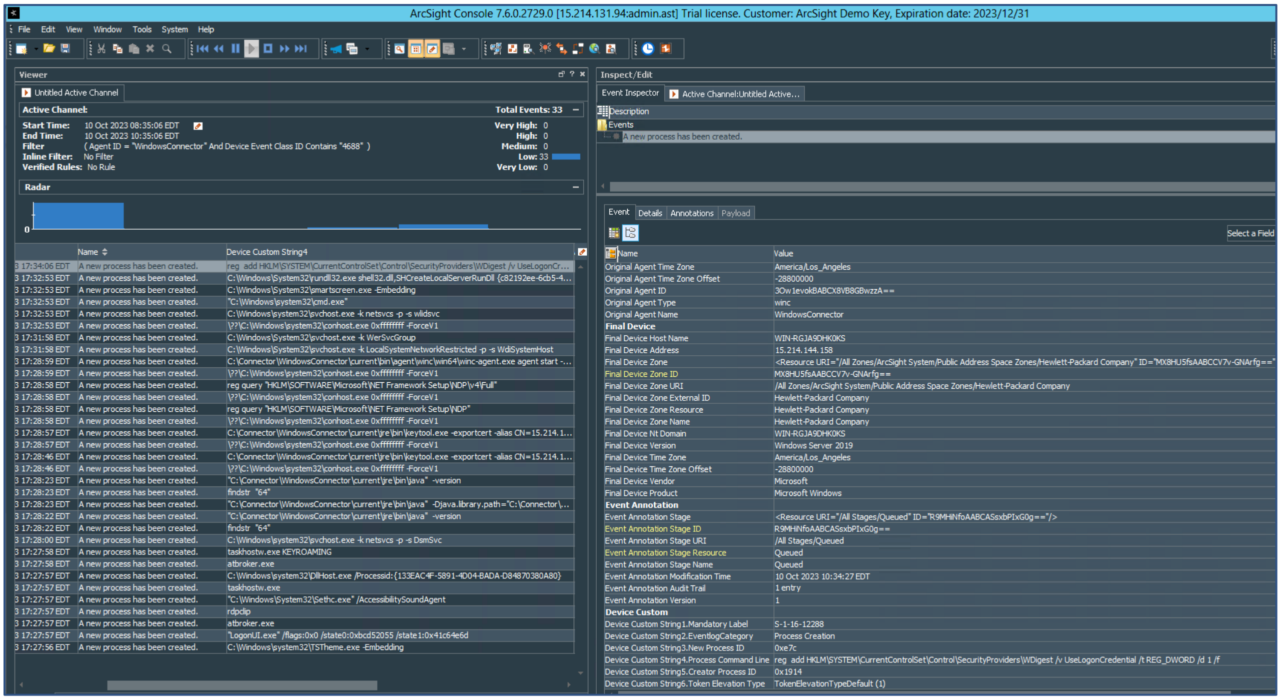


```
Event conditions
  event1
    AND
      Device Event Class ID = Microsoft-Windows-Security-Auditing:4688
      Target Process Name EndsWith reg.exe
      OR
        Target Process Name StartsWith "C:\\Windows\\System32"
        Target Process Name StartsWith "C:\\Windows\\SysWOW64"
      Device Custom String4 Is NOT NULL
      OR
        Device Custom String4 Contains add [ignore case]
        Device Custom String4 Contains delete [ignore case]
      Type In (Base,Aggregated)
      NOT
        InActiveList("/All Active Lists/ArcSight Foundation/Common/Suppression List/Host Name Based Suppression")
```

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.



ArcSight Console 7.6.0.2729.0 [15.214.131.94:admin.ast] Trial license. Customer: ArcSight Demo Key, Expiration date: 2023/12/31

Active Channel: **Total Events: 33**

Start Time: 10 Oct 2023 08:35:06 EDT
End Time: 10 Oct 2023 10:35:06 EDT
Filter (Agent ID = "WindowsConnector" And Device Event Class ID Contains "4688")
Inline Filter: No Filter
Verified Rules: No Rule

Very High: 0
High: 0
Medium: 0
Low: 33
Very Low: 0

	Name	Device Custom String4
3 17:34:06 EDT	A new process has been created.	reg add HKLM\SYSTEM\CurrentControlSet\Control\SecurityProviders\WDigest /v UseLogonCr...
3 17:32:53 EDT	A new process has been created.	C:\Windows\System32\rundll32.exe shell32.dll,SHCreateLocalServerRunDll {c82192ee-6cb5-4...
3 17:32:53 EDT	A new process has been created.	C:\Windows\System32\smartscreen.exe -Embedding
3 17:32:53 EDT	A new process has been created.	"C:\Windows\system32\cmd.exe"
3 17:32:53 EDT	A new process has been created.	C:\Windows\system32\svchost.exe -k netsvcs -p -s wlidsvc
3 17:32:53 EDT	A new process has been created.	\??\C:\Windows\system32\conhost.exe 0xffffffff -ForceV1
3 17:31:58 EDT	A new process has been created.	C:\Windows\System32\svchost.exe -k WerSvcGroup
3 17:31:58 EDT	A new process has been created.	C:\Windows\System32\svchost.exe -k LocalSystemNetworkRestricted -p -s WdiSystemHost
3 17:28:59 EDT	A new process has been created.	C:\Connector\WindowsConnector\current\bin\agent\winc\win64\winc-agent.exe agent start -...
3 17:28:59 EDT	A new process has been created.	\??\C:\Windows\system32\conhost.exe 0xffffffff -ForceV1
3 17:28:58 EDT	A new process has been created.	reg query "HKLM\SOFTWARE\Microsoft\NET Framework Setup\NDP\v4\Full"
3 17:28:58 EDT	A new process has been created.	\??\C:\Windows\system32\conhost.exe 0xffffffff -ForceV1
3 17:28:58 EDT	A new process has been created.	reg query "HKLM\SOFTWARE\Microsoft\NET Framework Setup\NDP"
3 17:28:58 EDT	A new process has been created.	\??\C:\Windows\system32\conhost.exe 0xffffffff -ForceV1
3 17:28:57 EDT	A new process has been created.	C:\Connector\WindowsConnector\current\jre\bin\keytool.exe -exportcert -alias CN=15.214.1...
3 17:28:57 EDT	A new process has been created.	\??\C:\Windows\system32\conhost.exe 0xffffffff -ForceV1
3 17:28:46 EDT	A new process has been created.	C:\Connector\WindowsConnector\current\jre\bin\keytool.exe -exportcert -alias CN=15.214.1...
3 17:28:46 EDT	A new process has been created.	\??\C:\Windows\system32\conhost.exe 0xffffffff -ForceV1
3 17:28:23 EDT	A new process has been created.	"C:\Connector\WindowsConnector\current\jre\bin\java" -version
3 17:28:23 EDT	A new process has been created.	findstr "64"
3 17:28:23 EDT	A new process has been created.	"C:\Connector\WindowsConnector\current\jre\bin\java" -Djava.library.path="C:\Connector\...
3 17:28:22 EDT	A new process has been created.	"C:\Connector\WindowsConnector\current\jre\bin\java" -version
3 17:28:22 EDT	A new process has been created.	findstr "64"
3 17:28:00 EDT	A new process has been created.	C:\Windows\system32\svchost.exe -k netsvcs -p -s DsmSvc
3 17:27:58 EDT	A new process has been created.	taskhostw.exe KEYROAMING
3 17:27:58 EDT	A new process has been created.	atbroker.exe
3 17:27:57 EDT	A new process has been created.	C:\Windows\system32\DllHost.exe /Processid:{133EAC4F-5891-4D04-BADA-D84870380A80}
3 17:27:57 EDT	A new process has been created.	taskhostw.exe
3 17:27:57 EDT	A new process has been created.	"C:\Windows\System32\Sethc.exe" /AccessibilitySoundAgent
3 17:27:57 EDT	A new process has been created.	rdpclip
3 17:27:57 EDT	A new process has been created.	atbroker.exe
3 17:27:57 EDT	A new process has been created.	"LogonUI.exe" /flags:0x0 /state0:0xbcd52055 /state1:0x41c64e6d
3 17:27:56 EDT	A new process has been created.	C:\Windows\system32\TSTheme.exe -Embedding

Description
Events
A new process has been created.

Name	Value
Original Agent Time Zone	America/Los_Angeles
Original Agent Time Zone Offset	-28800000
Original Agent ID	3Ow1evokBABCX8VB8GBwzzA==
Original Agent Type	winc
Original Agent Name	WindowsConnector
Final Device	
Final Device Host Name	WIN-RGJA9DHK0KS
Final Device Address	15.214.144.158
Final Device Zone	<Resource URI="/All Zones/ArcSight System/Public Address Space Zones/Hewlett-Packard Company" ID="MX8HU5fsAABCCV7v-GNArfg=="
Final Device Zone ID	MX8HU5fsAABCCV7v-GNArfg==
Final Device Zone URI	/All Zones/ArcSight System/Public Address Space Zones/Hewlett-Packard Company
Final Device Zone External ID	Hewlett-Packard Company
Final Device Zone Resource	Hewlett-Packard Company
Final Device Zone Name	Hewlett-Packard Company
Final Device Nt Domain	WIN-RGJA9DHK0KS
Final Device Version	Windows Server 2019
Final Device Time Zone	America/Los_Angeles
Final Device Time Zone Offset	-28800000
Final Device Vendor	Microsoft
Final Device Product	Microsoft Windows
Event Annotation	
Event Annotation Stage	<Resource URI="/All Stages/Queued" ID="R9MHiNfoAABCASsxbPIxG0g=="/>
Event Annotation Stage ID	R9MHiNfoAABCASsxbPIxG0g==
Event Annotation Stage URI	/All Stages/Queued
Event Annotation Stage Resource	Queued
Event Annotation Stage Name	Queued
Event Annotation Modification Time	10 Oct 2023 10:34:27 EDT
Event Annotation Audit Trail	1 entry
Event Annotation Version	1
Device Custom	
Device Custom String1.Mandatory Label	S-1-16-12288
Device Custom String2.EventlogCategory	Process Creation
Device Custom String3.New Process ID	0xe7c
Device Custom String4.Process Command Line	reg add HKLM\SYSTEM\CurrentControlSet\Control\SecurityProviders\WDigest /v UseLogonCredential /t REG_DWORD /d 1 /f
Device Custom String5.Creator Process ID	0x1914
Device Custom String6.Token Elevation Type	TokenElevationTypeDefault (1)

Fig 20

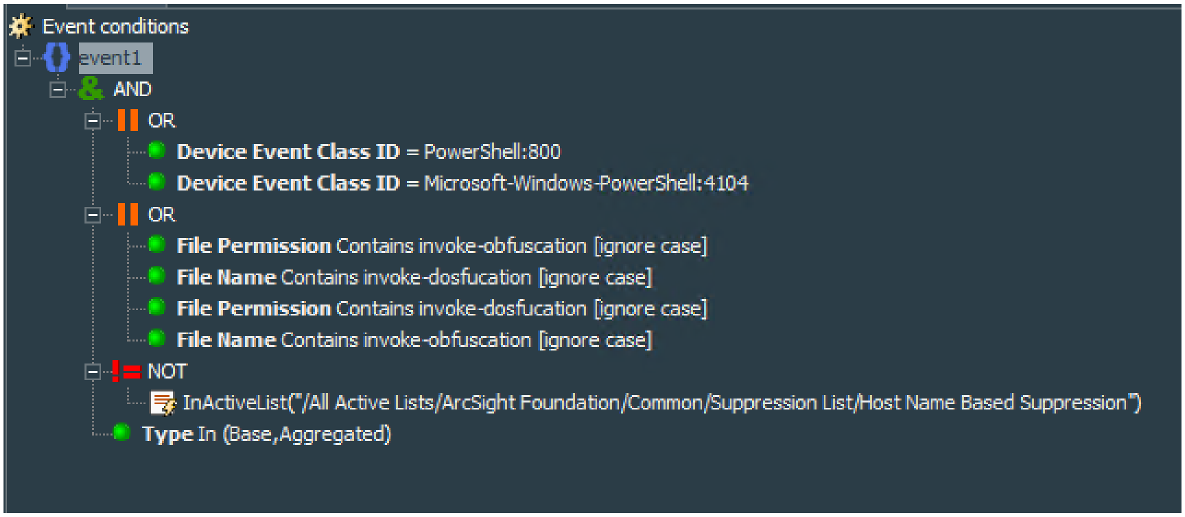
PowerShell Use Case

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
```

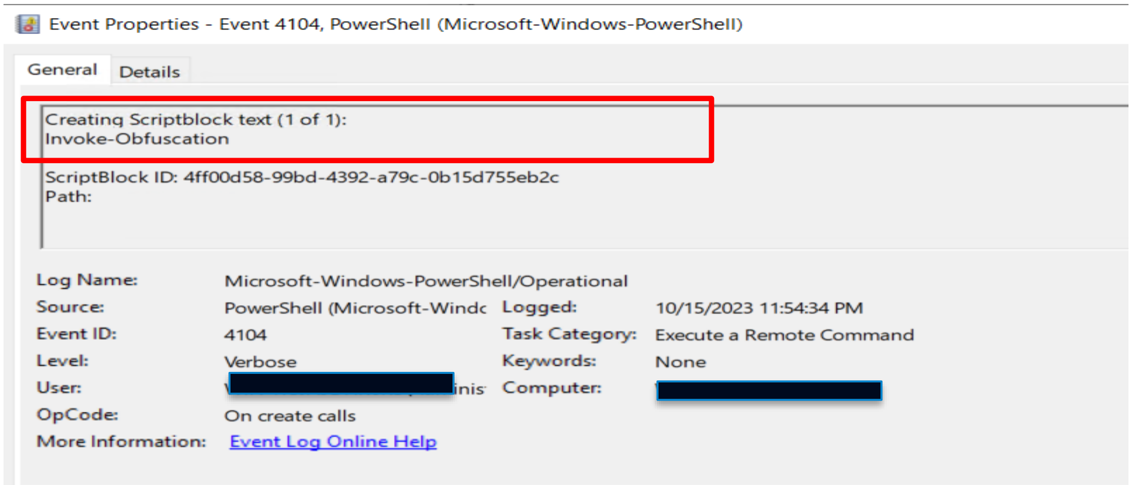
In default content, we have a rule to detect this activity. Rule Path: /All Rules/Real-time Rules/Security Threat Monitoring/Host Monitoring/Command Obfuscation Using PowerShell (Fig 21)



```
Event conditions
  event1
    AND
      OR
        Device Event Class ID = PowerShell:800
        Device Event Class ID = Microsoft-Windows-PowerShell:4104
      OR
        File Permission Contains invoke-obfuscation [ignore case]
        File Name Contains invoke-dosfucation [ignore case]
        File Permission Contains invoke-dosfucation [ignore case]
        File Name Contains invoke-obfuscation [ignore case]
      NOT
        InActiveList("/All Active Lists/ArcSight Foundation/Common/Suppression List/Host Name Based Suppression")
      Type In (Base,Aggregated)
```

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).



Event Properties - Event 4104, PowerShell (Microsoft-Windows-PowerShell)

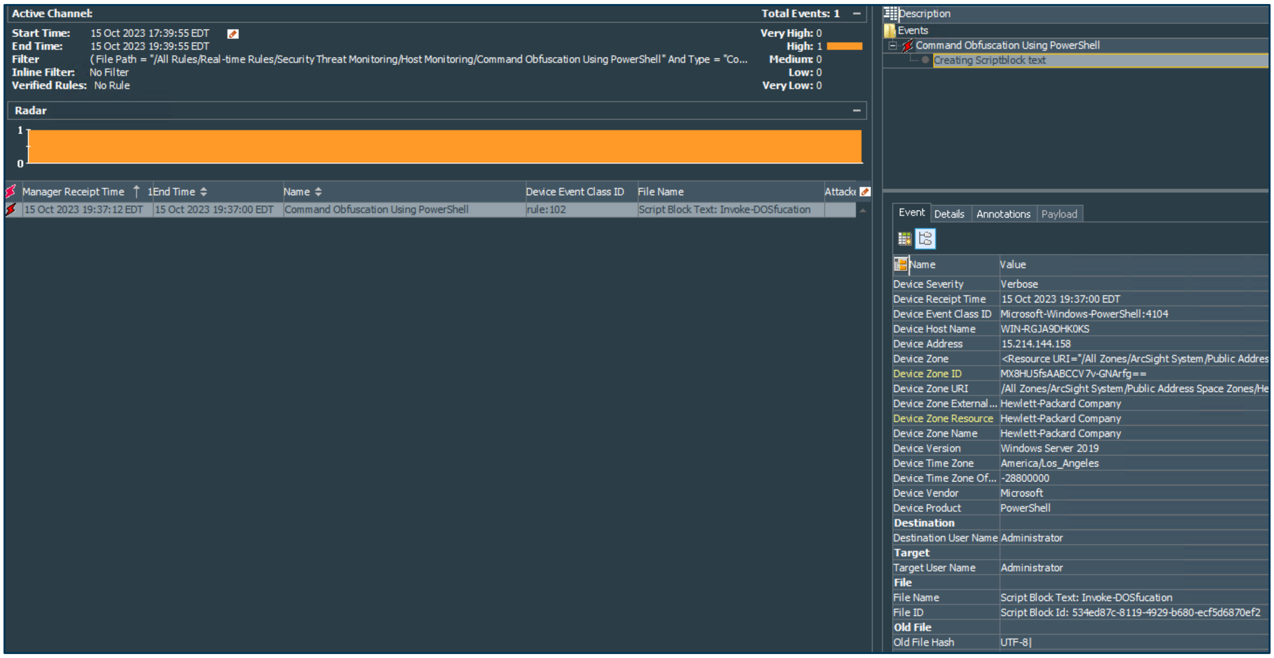
Creating Scriptblock text (1 of 1):
Invoke-Obfuscation

ScriptBlock ID: 4ff00d58-99bd-4392-a79c-0b15d755eb2c
Path:

Log Name:	Microsoft-Windows-PowerShell/Operational		
Source:	PowerShell (Microsoft-Windc	Logged:	10/15/2023 11:54:34 PM
Event ID:	4104	Task Category:	Execute a Remote Command
Level:	Verbose	Keywords:	None
User:	[redacted]	Computer:	[redacted]
OpCode:	On create calls		
More Information:	Event Log Online Help		

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).



Active Channel: Total Events: 1

Start Time:	15 Oct 2023 17:39:55 EDT	Very High:	0
End Time:	15 Oct 2023 19:39:55 EDT	High:	1
Filter	(File Path = "/All Rules/Real-time Rules/Security Threat Monitoring/Host Monitoring/Command Obfuscation Using PowerShell" And Type = "Co...	Medium:	0
Inline Filter:	No Filter	Low:	0
Verified Rules:	No Rule	Very Low:	0

Manager Receipt Time	End Time	Name	Device Event Class ID	File Name	Attacke
15 Oct 2023 19:37:12 EDT	15 Oct 2023 19:37:00 EDT	Command Obfuscation Using PowerShell	rule:102	Script Block Text: Invoke-DOSfucation	

Description
Events
Command Obfuscation Using PowerShell
Creating Scriptblock text

Name	Value
Device Severity	Verbose
Device Receipt Time	15 Oct 2023 19:37:00 EDT
Device Event Class ID	Microsoft-Windows-PowerShell:4104
Device Host Name	WIN-RGJA9DHK0KS
Device Address	15.214.144.158
Device Zone	<Resource URI="/All Zones/ArcSight System/Public Addres
Device Zone ID	MX8HU5fsAABCCV7v-GNArfg==
Device Zone URI	/All Zones/ArcSight System/Public Address Space Zones/He
Device Zone External...	Hewlett-Packard Company
Device Zone Resource	Hewlett-Packard Company
Device Zone Name	Hewlett-Packard Company
Device Version	Windows Server 2019
Device Time Zone	America/Los_Angeles
Device Time Zone Of...	-28800000
Device Vendor	Microsoft
Device Product	PowerShell
Destination	
Destination User Name	Administrator
Target	
Target User Name	Administrator
File	
File Name	Script Block Text: Invoke-DOSfucation
File ID	Script Block Id: 534ed87c-8119-4929-b680-ecf5d6870ef2
Old File	
Old File Hash	UTF-8|

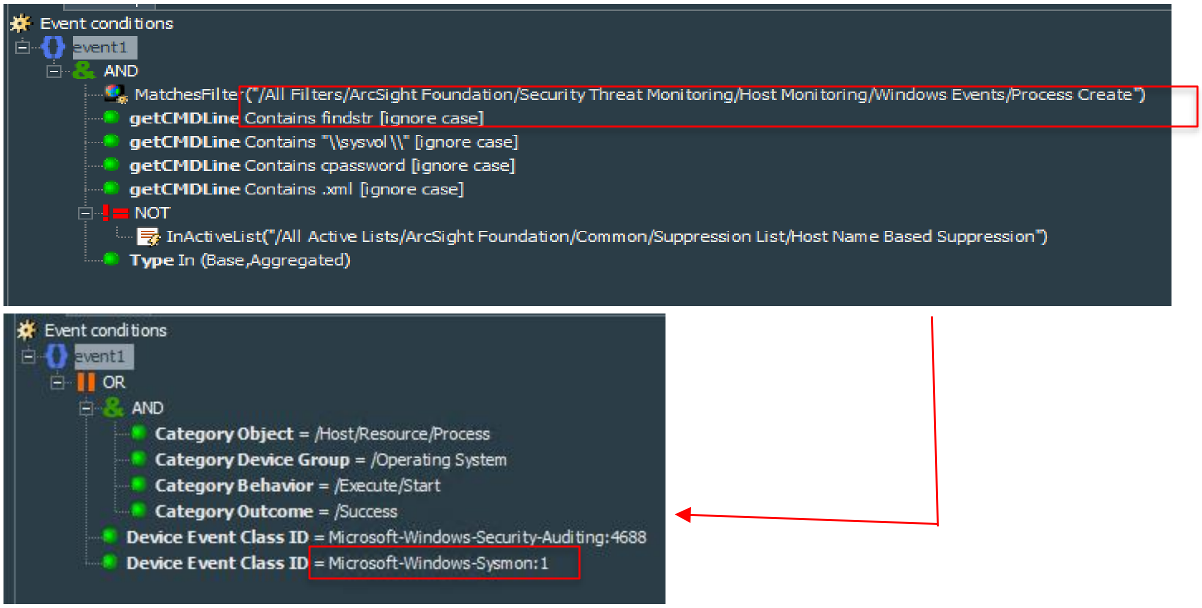
Fig 23

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.

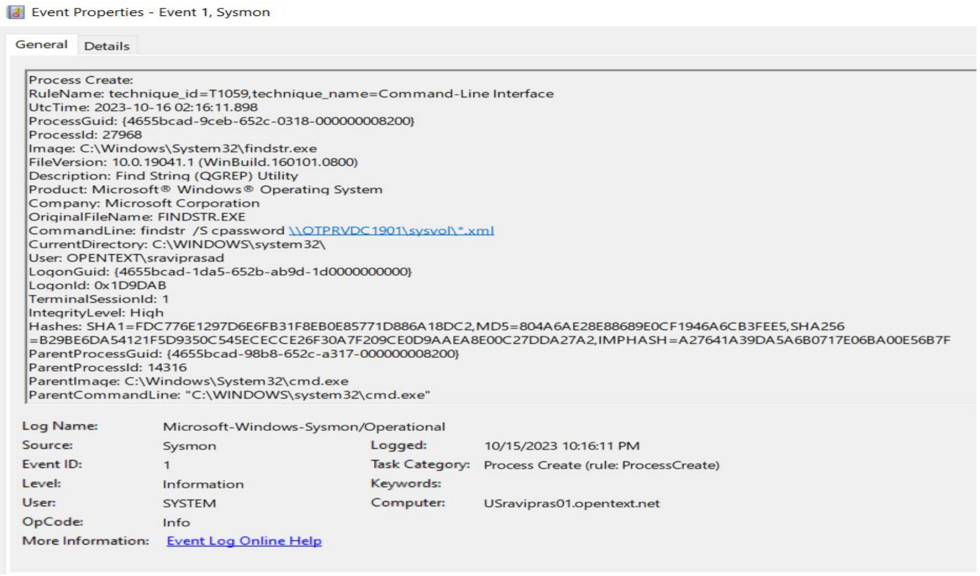


```
Event conditions
  event1
    AND
      MatchesFilter("/All Filters/ArcSight Foundation/Security Threat Monitoring/Host Monitoring/Windows Events/Process Create")
      getCMDLine Contains findstr [ignore case]
      getCMDLine Contains "\\sysvol\\" [ignore case]
      getCMDLine Contains cpassword [ignore case]
      getCMDLine Contains .xml [ignore case]
      NOT
        InActiveList("/All Active Lists/ArcSight Foundation/Common/Suppression List/Host Name Based Suppression")
      Type In (Base,Aggregated)
```

```
Event conditions
  event1
    OR
      AND
        Category Object = /Host/Resource/Process
        Category Device Group = /Operating System
        Category Behavior = /Execute/Start
        Category Outcome = /Success
      Device Event Class ID = Microsoft-Windows-Security-Auditing:4688
      Device Event Class ID = Microsoft-Windows-Sysmon:1
```

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.



```
Process Create:
RuleName: technique_id=T1059,technique_name=Command-Line Interface
UtcTime: 2023-10-16 02:16:11.898
ProcessGuid: {4655bcad-9ceb-652c-0318-000000008200}
ProcessId: 27968
Image: C:\Windows\System32\findstr.exe
FileVersion: 10.0.19041.1 (WinBuild.160101.0800)
Description: Find String (QGREP) Utility
Product: Microsoft® Windows® Operating System
Company: Microsoft Corporation
OriginalFileName: FINDSTR.EXE
CommandLine: findstr  /S cpassword \\OTPRVDC1901\sysvol\*.xml
CurrentDirectory: C:\WINDOWS\system32\
User: OPENTEXT\sraviprasad
LogonGuid: {4655bcad-1da5-652b-ab9d-1d0000000000}
LogonId: 0x1D9DAB
TerminalSessionId: 1
IntegrityLevel: High
Hashes: SHA1=FDC776E1297D6E6FB31F8EB0E85771D886A18DC2,MD5=804A6AE28E88689E0CF1946A6CB3FEE5,SHA256
=B29BE6DA54121F5D9350C545ECECCE26F30A7F209CE0D9AAEA8E00C27DDA27A2,IMPHASH=A27641A39DA5A6B0717E06BA00E56B7F
ParentProcessGuid: {4655bcad-98b8-652c-a317-000000008200}
ParentProcessId: 14316
ParentImage: C:\Windows\System32\cmd.exe
ParentCommandLine: "C:\WINDOWS\system32\cmd.exe"
```

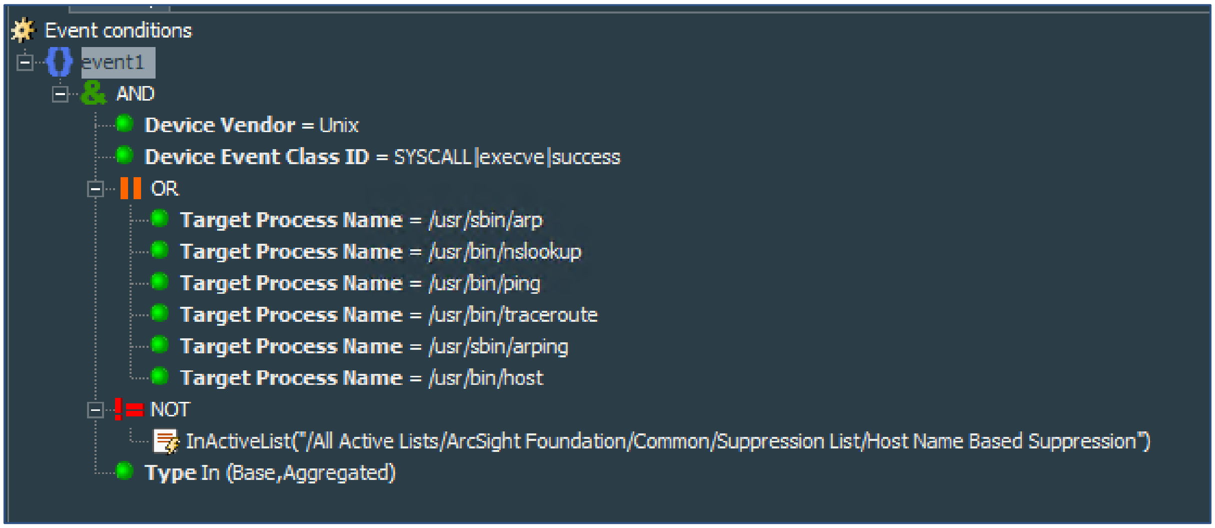
Log Name:	Microsoft-Windows-Sysmon/Operational		
Source:	Sysmon	Logged:	10/15/2023 10:16:11 PM
Event ID:	1	Task Category:	Process Create (rule: ProcessCreate)
Level:	Information	Keywords:	
User:	SYSTEM	Computer:	USravipras01.opentext.net
OpCode:	Info		
More Information:	Event Log Online Help		

Fig 25

Linux Use Case

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

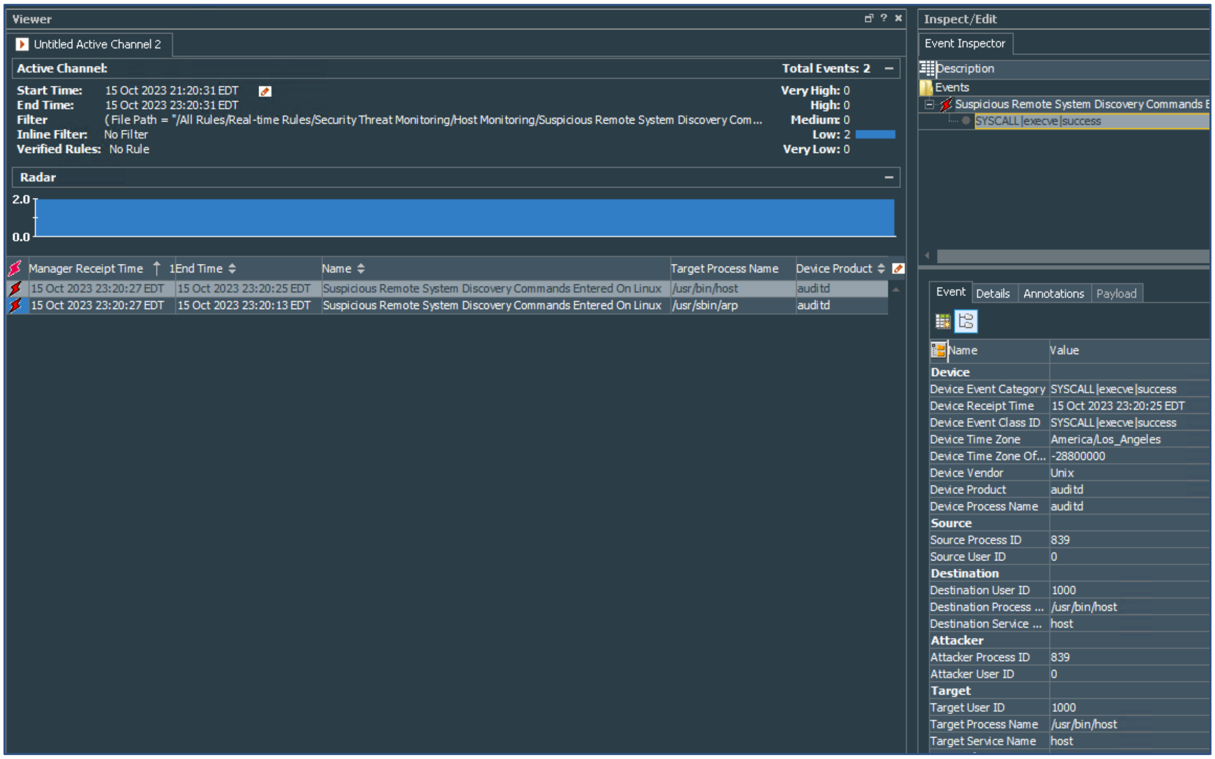
In default content, we have rule to detect this activity. Rule Path: /All Rules/ArcSight Foundation/MITRE ATT&CK/TA0007 Discovery/T1018-Remote System Discovery/Suspicious Remote System Discovery Commands Entered on Linux (Fig 26).



```
Event conditions
  event1
    AND
      Device Vendor = Unix
      Device Event Class ID = SYSCALL|execve|success
      OR
        Target Process Name = /usr/sbin/arp
        Target Process Name = /usr/bin/nslookup
        Target Process Name = /usr/bin/ping
        Target Process Name = /usr/bin/traceroute
        Target Process Name = /usr/sbin/arping
        Target Process Name = /usr/bin/host
      NOT
        InActiveList("/All Active Lists/ArcSight Foundation/Common/Suppression List/Host Name Based Suppression")
      Type In (Base,Aggregated)
```

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.



Active Channel: Total Events: 2

Start Time: 15 Oct 2023 21:20:31 EDT
End Time: 15 Oct 2023 23:20:31 EDT
Filter (File Path = "/All Rules/Real-time Rules/Security Threat Monitoring/Host Monitoring/Suspicious Remote System Discovery Com...
Inline Filter: No Filter
Verified Rules: No Rule

Very High: 0
High: 0
Medium: 0
Low: 2
Very Low: 0

Manager Receipt Time	End Time	Name	Target Process Name	Device Product
15 Oct 2023 23:20:27 EDT	15 Oct 2023 23:20:25 EDT	Suspicious Remote System Discovery Commands Entered On Linux	/usr/bin/host	auditd
15 Oct 2023 23:20:27 EDT	15 Oct 2023 23:20:13 EDT	Suspicious Remote System Discovery Commands Entered On Linux	/usr/sbin/arp	auditd

Description
Events
Suspicious Remote System Discovery Commands E
SYSCALL|execve|success

Name	Value
Device	
Device Event Category	SYSCALL|execve|success
Device Receipt Time	15 Oct 2023 23:20:25 EDT
Device Event Class ID	SYSCALL|execve|success
Device Time Zone	America/Los_Angeles
Device Time Zone Of...	-28800000
Device Vendor	Unix
Device Product	auditd
Device Process Name	auditd
Source	
Source Process ID	839
Source User ID	0
Destination	
Destination User ID	1000
Destination Process ...	/usr/bin/host
Destination Service ...	host
Attacker	
Attacker Process ID	839
Attacker User ID	0
Target	
Target User ID	1000
Target Process Name	/usr/bin/host
Target Service Name	host

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

The only warranties for products and services of Open Text and its affiliates and licensors (“Open Text”) are as may be set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. Open Text shall not be liable for technical or editorial errors or omissions contained herein. The information contained herein is subject to change without notice.