Usecase 3: PowerShell Usage

Objective: Detect usage of PowerShell

Log Source: Windows Sysmon

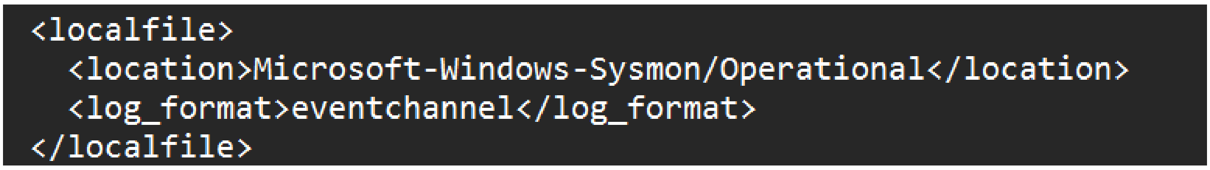
Event ID: Sysmon ID 1: Process Creation

Example Target Log From Event Viewer:

```
Process Create:
RuleName: technique_id=T1204,technique_name=User Execution
UtcTime: 2025-06-30 22:13:58.574
ProcessGuid: {0430377a-0c26-6863-fd10-000000001d00}
ProcessId: 22660
Image: C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe
FileVersion: 10.0.26100.3323 (WinBuild.160101.0800)
Description: Windows PowerShell
Product: Microsoft® Windows® Operating System
Company: Microsoft Corporation
OriginalFileName: PowerShell.EXE
CommandLine: "C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe"
CurrentDirectory: C:\Users\<username>\
User: <system name>\<username>
LogonGuid: {0430377a-884e-685d-13c1-620300000000}
LogonId: 0x362C113
TerminalSessionId: 2
IntegrityLevel: Medium
Hashes: SHA1= <hash string>
ParentProcessGuid: {0430377a-8850-685d-4a06-000000001d00}
ParentProcessId: 17992
ParentImage: C:\Windows\explorer.exe
ParentCommandLine: C:\WINDOWS\Explorer.EXE
```

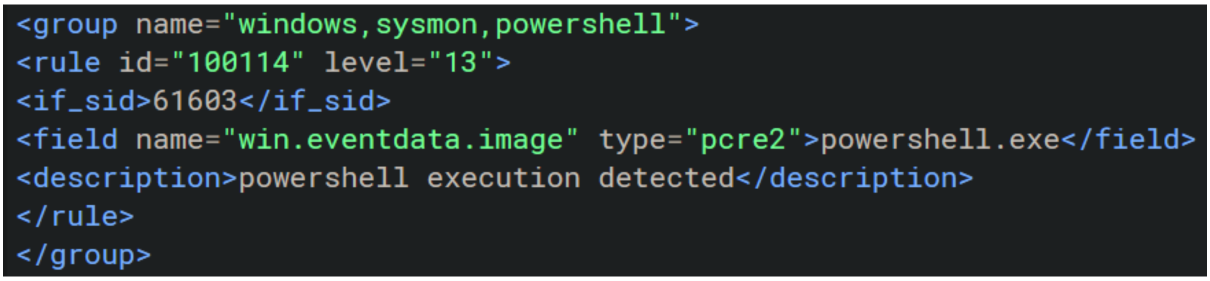
Pre-configuration: Sysmon set up, after installation run these commands as admin in powershell: cd ‘C:\Sysmon\’ (or wherever Sysmon is installed) > wget https://wazuh.com/resources/blog/emulation-of-attack-techniques-and-detection-with-wazuh/sysmonconfig.xml -Outfile sysmonconfig.xml > then .\Sysmon.exe (or wherever Sysmon.exe is installed) -accepteula -I .\sysmonconfig.xml (or wherever sysmonconfig.xml is installed)

Then need to configure agent log collection for Sysmon by adding this to the local agent configuration



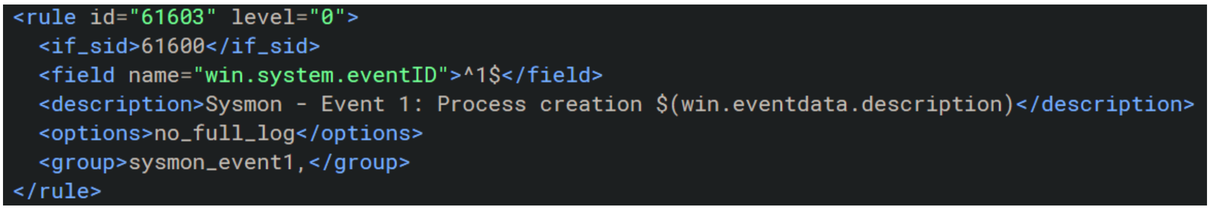
```
<localfile>
  <location>Microsoft-Windows-Sysmon/Operational</location>
  <log_format>eventchannel</log_format>
</localfile>
```

Custom Wazuh Rule:



```
<group name="windows,sysmon,powershell">
<rule id="100114" level="13">
<if_sid>61603</if_sid>
<field name="win.eventdata.image" type="pcre2">powershell.exe</field>
<description>powershell execution detected</description>
</rule>
</group>
```

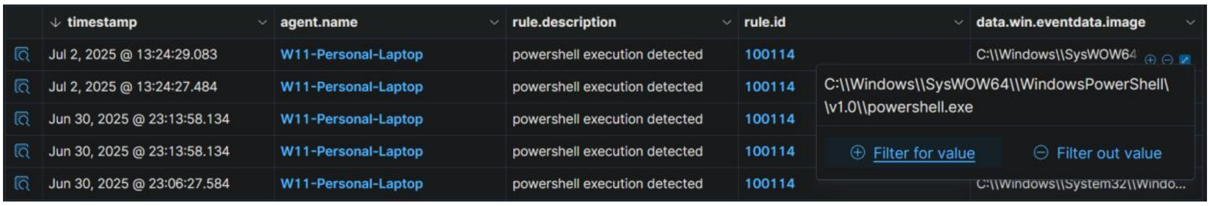
Filters the default 61603 for powershell occurrences using eventdata.image contains powershell.exe



```
<rule id="61603" level="0">
  <if_sid>61600</if_sid>
  <field name="win.system.eventID">^1$</field>
  <description>Sysmon - Event 1: Process creation $(win.eventdata.description)</description>
  <options>no_full_log</options>
  <group>sysmon_event1,</group>
</rule>
```

Because 61603 already filters for event ID 1 events, no further filtering is required other than for the image

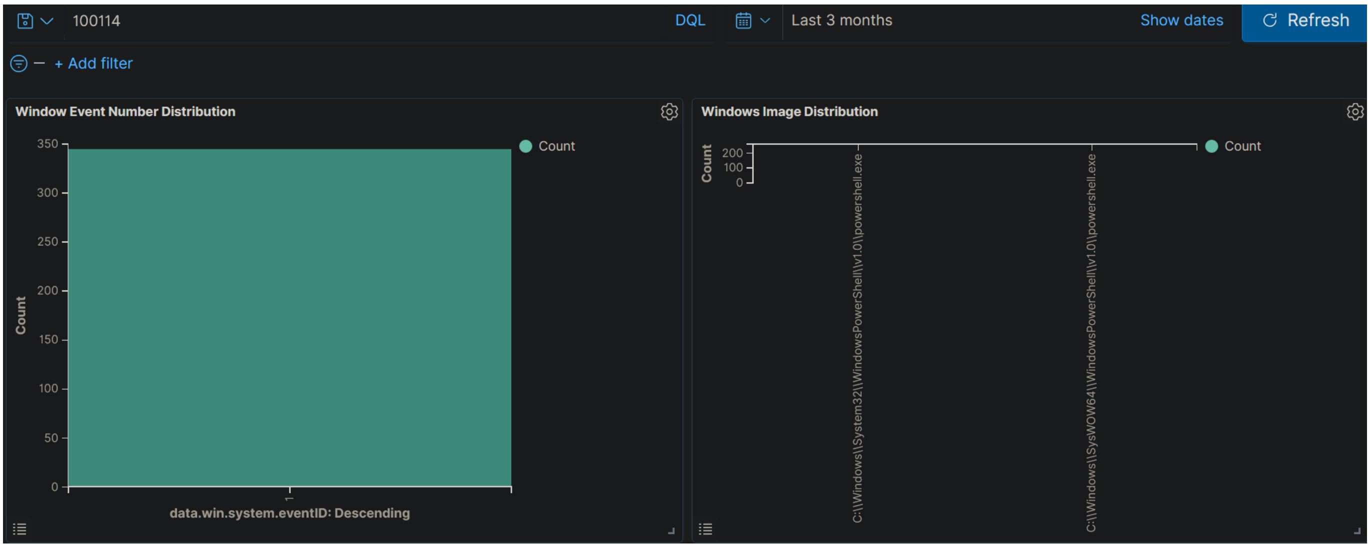
Example Alerts:



	↓ timestamp	agent.name	rule.description	rule.id	data.win.eventdata.image
	Jul 2, 2025 @ 13:24:29.083	W11-Personal-Laptop	powershell execution detected	100114	C:\\Windows\\SysWOW64
	Jul 2, 2025 @ 13:24:27.484	W11-Personal-Laptop	powershell execution detected	100114	
	Jun 30, 2025 @ 23:13:58.134	W11-Personal-Laptop	powershell execution detected	100114	
	Jun 30, 2025 @ 23:13:58.134	W11-Personal-Laptop	powershell execution detected	100114	
	Jun 30, 2025 @ 23:06:27.584	W11-Personal-Laptop	powershell execution detected	100114	C:\\Windows\\System32\\Windo...

C:\\Windows\\SysWOW64\\WindowsPowerShell\\v1.0\\powershell.exe

Verification of fields:



100114
DQL
Last 3 months
Show dates
Refresh
+ Add filter
Window Event Number Distribution
Count
350
300
250
200
150
100
50
0
1
data.win.system.eventID: Descending
● Count
Windows Image Distribution
Count
200
100
0
C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe
C:\\Windows\\SysWOW64\\WindowsPowerShell\\v1.0\\powershell.exe
● Count

While it seems the image filtering is unsuccessful on first glance we can see both are different paths to powershell so both are successful.