

# Threat Hunt Scenario Assistance

## Threat Hunt Scenario

Assistance



**\*\*Date:\*\*** October 1st - 15th, 2025

**\*\*Analyst:\*\*** Grisham DelRosario

**\*\*Environment:\*\*** Microsoft - Log Analytics Workspace (LAW - Cyber Range)

**\*\*Attack Type:\*\*** Fake Remote Session

**\*\*Attacker IP:\*\*** 192.168.50.255 (Kali)

**\*\*Target IP:\*\*** 192.168.50.10 (Windows 10 Server)

## Scenario

A routine support request should have ended with a reset and reassurance. Instead, the so-called "help" left behind a trail of anomalies that don't add up. What was framed as troubleshooting looked more like an audit of the system itself probing, cataloging, leaving subtle traces in its wake. Actions chained together in

suspicious sequence: first gaining a foothold, then expanding reach, then preparing to

linger long after the session ended. And just when the activity should have raised questions, a neat explanation appeared – a story planted in plain sight, designed to justify the very behavior that demanded scrutiny. This wasn't remote assistance. It was a misdirection. Your mission this time is to reconstruct the timeline, connect the scattered remnants of this "support session", and decide what was legitimate, and what was staged. The evidence is here. The question is whether you'll see through the story or believe it.

### Starting Point

Before you officially begin the flags, you must first determine where to start hunting. Identify where to start hunting with the following intel given:

1. Multiple machines in the department started spawning processes originating from the **download** folders. This unexpected scenario occurred during the **first half of October**.
2. Several machines were found to share the same types of files – similar executables, naming patterns, and other traits.
3. Common keywords among the discovered files included **"desk," "help," "support,"** and **"tool."**
4. Intern operated machines seem to be affected to certain degree.

Identify the most suspicious machine based on the given conditions \*

gab-intern-vm

```
//-----  
let start = datetime(2025-10-01T00:00:00Z);  
let end   = datetime(2025-10-31T23:59:59Z);  
let keywords = dynamic(["desk","help","support","tool"]);  
DeviceFileEvents  
| where TimeGenerated between (start .. end)  
//| where FolderPath has @"C:\Users\" and FolderPath has @"\Downloads\"  
| where FileName has_any (keywords)  
| project TimeGenerated, DeviceName, FileName, FolderPath,  
          InitiatingProcessAccountDomain, InitiatingProcessFolderPath, InitiatingProcessId,  
          InitiatingProcessFileName, InitiatingProcessCommandLine, SHA1  
| order by TimeGenerated desc
```

1. Spawning process originating from the download folder. Occurred in the first half of October, so sometime between October 1st -15th?
2. Similar executables, naming patterns, and other traits.
3. Common keywords, "desk", "help", "support", and "tool"

TimeGenerated [Local Time] ↑	DeviceName	FileName	FolderPath
> 10/8/2025, 5:48:16.739 PM	amb15	7-Zip Help.Ink	C:\ProgramData\Microsoft\Windows\Start Menu\Programs\7-Zip\7-Zip Help.Ink
> 10/8/2025, 5:49:07.775 PM	windows-target-1	7-Zip Help.Ink	C:\ProgramData\Microsoft\Windows\Start Menu\Programs\7-Zip\7-Zip Help.Ink
> 10/8/2025, 8:02:16.902 PM	st-th-1	Memory Diagnostics Tool.Ink	C:\\$WinREAgent\Scratch\Mount\ProgramData\Microsoft\Windows\Start Menu\Programs\Administrative Tools\Memory Diagnostics Tool.Ink
> 10/8/2025, 8:02:16.902 PM	st-th-1	Memory Diagnostics Tool.Ink	C:\\$WinREAgent\Scratch\Mount\Windows\WinSxS\amd64_microsoft-windows-m_diagnostics-schedule_31bf3856ad364e35_10.0.0.0_x-ww_31bf3856ad364e35_x-ww\Memory Diagnostics Tool.Ink
> 10/8/2025, 8:11:07.285 PM	test-ar-win-10	help.png	C:\Users\va1ex1313\AppData\Local\Microsoft\OneDrive\25.179.0914.0003\Bundle\Assets\common\assets\images\help.png
> 10/8/2025, 8:29:44.361 PM	graham-mde	help.png	C:\Users\labuser\AppData\Local\Microsoft\OneDrive\25.179.0914.0003\Bundle\Assets\common\assets\images\help.png
> 10/8/2025, 8:42:50.767 PM	graham-mde	YourPhone.Contracts.Help.xml	C:\Program Files\WindowsApps\Microsoft.YourPhone_1.25072.79.0_x64__8wekyb3d8bbwe\YourPhone.Contracts.Help.xml
> 10/8/2025, 8:42:50.833 PM	graham-mde	YourPhone.Help.xml	C:\Program Files\WindowsApps\Microsoft.YourPhone_1.25072.79.0_x64__8wekyb3d8bbwe\YourPhone.Help.xml
> 10/8/2025, 9:49:17.901 PM	windows-target-1	7-Zip Help.Ink	C:\ProgramData\Microsoft\Windows\Start Menu\Programs\7-Zip\7-Zip Help.Ink
> 10/9/2025, 1:39:42.221 AM	sam-ftwr	tool_close.bmp	C:\Program Files (x86)\Yahoo!\Messenger\skins\Default\theme\tool_close.bmp
> 10/9/2025, 1:39:42.237 AM	sam-ftwr	tool_closebutton.bmp	C:\Program Files (x86)\Yahoo!\Messenger\skins\Default\theme\tool_closebutton.bmp
> 10/9/2025, 1:49:05.682 AM	wmx-portfolio	7-Zip Help.Ink	C:\ProgramData\Microsoft\Windows\Start Menu\Programs\7-Zip\7-Zip Help.Ink
> 10/9/2025, 3:00:08.533 AM	salehsalehvm	help.png	C:\Users\labuser\AppData\Local\Microsoft\OneDrive\25.179.0914.0003\Bundle\Assets\common\assets\images\help.png
> 10/9/2025, 4:58:10.912 AM	gab-intern-vm	help.png	C:\Users\g4br3lintern\AppData\Local\Microsoft\OneDrive\25.179.0914.0003\Bundle\Assets\common\assets\images\help.png
> 10/9/2025, 5:05:38.728 AM	gab-intern-vm	Support_701.txt	C:\Users\g4br3lintern\Documents\HelpdeskLogs\Support_701.txt
> 10/9/2025, 5:48:52.609 AM	salehsalehvm	7-Zip Help.Ink	C:\ProgramData\Microsoft\Windows\Start Menu\Programs\7-Zip\7-Zip Help.Ink
> 10/9/2025, 5:49:02.417 AM	gab-intern-vm	7-Zip Help.Ink	C:\ProgramData\Microsoft\Windows\Start Menu\Programs\7-Zip\7-Zip Help.Ink
> 10/9/2025, 5:49:14.737 AM	aafbc1288	7-Zip Help.Ink	C:\ProgramData\Microsoft\Windows\Start Menu\Programs\7-Zip\7-Zip Help.Ink
> 10/9/2025, 7:06:05.966 AM	na-zero-1	help.png	C:\Users\Nalab\AppData\Local\Microsoft\OneDrive\25.174.0907.0003\Bundle\Assets\common\assets\images\help.png
> 10/9/2025, 9:48:50.750 AM	salehsalehvm	7-Zip Help.Ink	C:\ProgramData\Microsoft\Windows\Start Menu\Programs\7-Zip\7-Zip Help.Ink

```
//-----
let start = datetime(2025-10-01T00:00:00Z);
let end   = datetime(2025-10-31T23:59:59Z);
let keywords = dynamic(["desk","help","support","tool"]);
DeviceFileEvents
| where TimeGenerated between (start .. end)
//| where FolderPath has @"C:\Users\" and FolderPath has @"\Downloads\"
| where DeviceName == "gab-intern-vm"
| where FileName has_any (keywords)
| project TimeGenerated, DeviceName, FileName, FolderPath,
InitiatingProcessAccountDomain, InitiatingProcessFolderPath, InitiatingProcessId,
InitiatingProcessFileName, InitiatingProcessCommandLine, SHA1
| order by TimeGenerated desc
```

Ideally, another way I could have found this device without having to think so hard was to have queried the term 'Intern' for 'DeviceName' in order to find the suspicious device,

'gab-intern-vm'.

This too would have been an easier method to find in order to narrow down the suspicious device.

Results Chart				
TimeGenerated [Local Time] ↑	DeviceName	FileName	FolderPath	InitiatingProcessAccountDo...
> 10/6/2025, 5:49:22.672 AM	gab-intern-vm	7-Zip Help.Ink	C:\ProgramData\Microsoft\Windows\Start Menu\Programs\7-Zip\7-Zip Help.Ink	nt authority
> 10/6/2025, 9:49:21.328 AM	gab-intern-vm	7-Zip Help.Ink	C:\ProgramData\Microsoft\Windows\Start Menu\Programs\7-Zip\7-Zip Help.Ink	nt authority
> 10/6/2025, 1:49:32.486 PM	gab-intern-vm	7-Zip Help.Ink	C:\ProgramData\Microsoft\Windows\Start Menu\Programs\7-Zip\7-Zip Help.Ink	nt authority
> 10/6/2025, 5:49:17.419 PM	gab-intern-vm	7-Zip Help.Ink	C:\ProgramData\Microsoft\Windows\Start Menu\Programs\7-Zip\7-Zip Help.Ink	nt authority
> 10/6/2025, 9:49:14.478 PM	gab-intern-vm	7-Zip Help.Ink	C:\ProgramData\Microsoft\Windows\Start Menu\Programs\7-Zip\7-Zip Help.Ink	nt authority
> 10/7/2025, 5:49:14.051 AM	gab-intern-vm	7-Zip Help.Ink	C:\ProgramData\Microsoft\Windows\Start Menu\Programs\7-Zip\7-Zip Help.Ink	nt authority
> 10/7/2025, 9:49:03.251 AM	gab-intern-vm	7-Zip Help.Ink	C:\ProgramData\Microsoft\Windows\Start Menu\Programs\7-Zip\7-Zip Help.Ink	nt authority
> 10/7/2025, 1:49:03.358 PM	gab-intern-vm	7-Zip Help.Ink	C:\ProgramData\Microsoft\Windows\Start Menu\Programs\7-Zip\7-Zip Help.Ink	nt authority
> 10/7/2025, 5:49:11.164 PM	gab-intern-vm	7-Zip Help.Ink	C:\ProgramData\Microsoft\Windows\Start Menu\Programs\7-Zip\7-Zip Help.Ink	nt authority
> 10/7/2025, 9:48:56.985 PM	gab-intern-vm	7-Zip Help.Ink	C:\ProgramData\Microsoft\Windows\Start Menu\Programs\7-Zip\7-Zip Help.Ink	nt authority
> 10/9/2025, 4:58:10.912 AM	gab-intern-vm	help.png	C:\Users\g4bn3Intern\AppData\Local\Microsoft\OneDrive\25.179.0914.0003\Bundle\Assets\common\assets\images\help.png	gab-intern-vm
> 10/9/2025, 5:05:38.728 AM	gab-intern-vm	Support_701.txt	C:\Users\g4bn3Intern\Documents\HelpdeskLogs\Support_701.txt	gab-intern-vm
> 10/9/2025, 5:49:02.417 AM	gab-intern-vm	7-Zip Help.Ink	C:\ProgramData\Microsoft\Windows\Start Menu\Programs\7-Zip\7-Zip Help.Ink	nt authority
> 10/9/2025, 9:49:00.718 AM	gab-intern-vm	7-Zip Help.Ink	C:\ProgramData\Microsoft\Windows\Start Menu\Programs\7-Zip\7-Zip Help.Ink	nt authority
> 10/9/2025, 1:48:54.396 PM	gab-intern-vm	7-Zip Help.Ink	C:\ProgramData\Microsoft\Windows\Start Menu\Programs\7-Zip\7-Zip Help.Ink	nt authority
> 10/9/2025, 5:49:06.718 PM	gab-intern-vm	7-Zip Help.Ink	C:\ProgramData\Microsoft\Windows\Start Menu\Programs\7-Zip\7-Zip Help.Ink	nt authority
> 10/9/2025, 9:48:53.682 PM	gab-intern-vm	7-Zip Help.Ink	C:\ProgramData\Microsoft\Windows\Start Menu\Programs\7-Zip\7-Zip Help.Ink	nt authority

## Flag 1 – Initial Execution Detection

### Objective:

Detect the earliest anomalous execution that could represent an entry point.

### What to Hunt:

Look for atypical script or interactive command activity that deviates from normal user behavior or baseline patterns.

### Thought:

Pinpointing the first unusual execution helps you anchor the timeline and follow the actor's parent/child process chain.

### Hint:

1. Downloads
2. Two

What was the first CLI parameter name used during the execution of the suspicious program? ★

-ExecutionPolicy

Throughout the threat hunt, the table 'DeviceProcessEvents' was very key in order to examine the logs.

For Flag 1, we're looking at Initial Execution Detection

When I read what to hunt and saw 'script', the first thing that came to mind was PowerShell and Command Prompt.

Further on, the question asked "What was the first CLI (command line interface) parameter name used during the execution of the suspicious program?"

After looking back and forth at what was being asked of the flag and examining logs "unusual execution" was key in order to find this flag.

The earliest anomalous execution of powershell being executed was October 9th, 2025 @ 5:22AM

```
//-----FLAG 1-----
DeviceProcessEvents
| where DeviceName == "gab-intern-vm"
| where AccountName == "g4bri3lintern"
| where FileName == "powershell.exe"
| where TimeGenerated between (datetime(2025-10-01T00:00:00Z) .. datetime(2025-10-20T23:59:59Z))
| project TimeGenerated, DeviceName, AccountName, FileName, FolderPath, ProcessCommandLine, InitiatingProcessFileName, SHA1
```

Results	Chart	EARLIEST ANOMALOUS EXECUTION	FIRST CLI PARAMETER NAME	ATYPICAL SCRIPT			
TimeGenerated [UTC]	ActionType	DeviceName	AccountName	FileName	FolderPath	ProcessCommandLine	InitiatingProcessFileName
> 10/6/2025, 6:00:48.754 AM	ProcessCreated	gab-intern-vm	g4bri3lintern	powershell.exe	C:\Windows\System32\WindowsPowerShell\1.0\powershell.exe	powershell.exe -ExecutionPolicy Bypass -NoProfile -Command "Add-Type -using System.Diagnostics; using System.Diagnostics.Tracing; names...	cmd.exe
> 10/9/2025, 11:59:09.521 AM	ProcessCreated	gab-intern-vm	g4bri3lintern	powershell.exe	C:\Windows\System32\WindowsPowerShell\1.0\powershell.exe	"powershell.exe"	runtimebroker.exe
> 10/9/2025, 12:22:21.352 PM	ProcessCreated	gab-intern-vm	g4bri3lintern	powershell.exe	C:\Windows\System32\WindowsPowerShell\1.0\powershell.exe	"powershell.exe"	runtimebroker.exe
> 10/9/2025, 12:22:27.658 PM	ProcessCreated	gab-intern-vm	g4bri3lintern	powershell.exe	C:\Windows\System32\WindowsPowerShell\1.0\powershell.exe	"powershell.exe" -ExecutionPolicy Bypass -File C:\Users\g4bri3lintern\Downloads\SupportTool.ps1	powershell.exe
> 10/9/2025, 12:23:38.905 PM	ProcessCreated	gab-intern-vm	g4bri3lintern	powershell.exe	C:\Windows\System32\WindowsPowerShell\1.0\powershell.exe	"powershell.exe" -Command "if([Get-ExecutionPolicy] -ne 'AllSigned') { Set-ExecutionPolicy -Scope Process Bypass }; & 'C:\Users\g4bri3lintern\Downloads\...	explorer.exe
> 10/9/2025, 12:34:57.750 PM	ProcessCreated	gab-intern-vm	g4bri3lintern	powershell.exe	C:\Windows\System32\WindowsPowerShell\1.0\powershell.exe	"powershell.exe" -NoProfile -WindowStyle Hidden -Command Write-Output 'Set-MpPreference -DisableRealtimeMonitoring \$true'   Out-File -FilePath 'C:\Us...	powershell.exe
> 10/9/2025, 12:50:16.419 PM	ProcessCreated	gab-intern-vm	g4bri3lintern	powershell.exe	C:\Windows\System32\WindowsPowerShell\1.0\powershell.exe	"powershell.exe"	runtimebroker.exe
> 10/9/2025, 12:50:39.955 PM	ProcessCreated	gab-intern-vm	g4bri3lintern	powershell.exe	C:\Windows\System32\WindowsPowerShell\1.0\powershell.exe	"powershell.exe" -NoProfile -Sta -Command "try { Get-Clipboard } Out-Null } catch {}"	powershell.exe
> 10/9/2025, 1:12:46.202 PM	ProcessCreated	gab-intern-vm	g4bri3lintern	powershell.exe	C:\Windows\System32\WindowsPowerShell\1.0\powershell.exe	"powershell.exe" -NoProfile -ExecutionPolicy Bypass -WindowStyle Hidden -File C:\Users\g4bri3lintern\Downloads\SupportTool.ps1	svchost.exe
> 10/9/2025, 1:13:12.526 PM	ProcessCreated	gab-intern-vm	g4bri3lintern	powershell.exe	C:\Windows\System32\WindowsPowerShell\1.0\powershell.exe	"powershell.exe" -NoProfile -ExecutionPolicy Bypass -WindowStyle Hidden -File 'C:\Users\g4bri3lintern\Downloads\SupportTool.ps1'	explorer.exe

Upon looking at the log activity for powershell executables we can see the first CLI parameter is set to '-ExecutionPolicy'. First time it was executed was on October 6th, 2025 at 6:00:48 AM

This eventually occurred again for a powershell.exe process called 'SupportTool.ps1' for October 9th, 2025 during 12:22:27 PM

## Flag 2 – Defense Disabling

### Objective:

Identify indicators that suggest attempts to imply or simulate changing security posture.

### What to Hunt:

Search for artifact creation or short-lived process activity that contains tamper-related content or hints, without assuming an actual configuration change occurred.

### Thought:

A planted or staged tamper indicator is a signal of intent — treat it as intent, not proof of actual mitigation changes.

### Hint:

1. File was manually accessed

What was the name of the file related to this exploit? \*

DefenderTamperArtifact.Ink

Further on, I decided to pivot back into DeviceProcessEvents table and look back into more power shell activity.

I kept noticing this command scrolling through the logs and noticed the string when querying for

```
' Artifact ' and ' Out-File -FilePath  
'C:\Users\Public\DefenderTamperArtifact.txt' '
```

The query used in Flag 1 to understand the CLI parameter ' -ExecutionPolicy ', was key into understanding the timeline of events

that showed another powershell command outputting a file called

```
' DefenderTamperArtifact.txt '
```

As I kept querying for the term artifact and I kept on encountering the file name

```
' ReconArtifacts.zip. '
```

It was the closest thing I can find but it was not the official tampered artifact.

Still needed to find something related to either this or the ' DefenderTamperArtifact.txt ' file.

Somehow I knew these were related to Defense Disabling but could not make the linkage as to how it was all connected.

Results		Chart					
TimeGenerated [UTC]		ActionType	DeviceName	AccountName	FileName	ProcessCommandLine	InitiatingProcessFileName
> 10/6/2025, 6:00:48.754 AM		ProcessCreated	gab-intern-vm	gab33intern	powershell.exe	powershell.exe -ExecutionPolicy Bypass -NoProfile -Command "Add-Type -using System; using System.Diagnostics; using System.Diagnostics.Tracing; namespace Sense { [EventDataName = 'Onboarding']public struct Onbo...	cmd.exe
> 10/9/2025, 11:59:00.521 AM		ProcessCreated	gab-intern-vm	gab33intern	powershell.exe	"powershell.exe"	runtimebroker.exe
> 10/9/2025, 12:22:21.352 PM		ProcessCreated	gab-intern-vm	gab33intern	powershell.exe	"powershell.exe"	runtimebroker.exe
> 10/9/2025, 12:22:27.658 PM		ProcessCreated	gab-intern-vm	gab33intern	powershell.exe	"powershell.exe" -ExecutionPolicy Bypass -File C:\Users\gab33intern\Downloads\SupportTools1	powershell.exe
> 10/9/2025, 12:23:30.905 PM		ProcessCreated	gab-intern-vm	gab33intern	powershell.exe	"powershell.exe" "-Command '&([Get-ExecutionPolicy] -ne 'AllSigned') { Set-ExecutionPolicy -Scope Process Bypass }; & 'C:\Users\gab33intern\Downloads\SupportTools1'"	explorer.exe
> 10/9/2025, 12:34:57.758 PM		ProcessCreated	gab-intern-vm	gab33intern	powershell.exe	"powershell.exe" -NoProfile -WindowStyle Hidden -Command Write-Output 'Set-MpPreference -DisableRealtimeMonitoring \$true'   Out-File -FilePath 'C:\Users\Public\DefenderTamperArtifact.txt' -Encoding ASCII -Append	powershell.exe
> 10/9/2025, 12:50:16.419 PM		ProcessCreated	gab-intern-vm	gab33intern	powershell.exe	"powershell.exe"	runtimebroker.exe
> 10/9/2025, 12:50:39.955 PM		ProcessCreated	gab-intern-vm	gab33intern	powershell.exe	"powershell.exe" -NoProfile -Sta -Command 'try { Get-Clipboard   Out-Null } catch { }'	powershell.exe
> 10/9/2025, 1:12:46.202 PM		ProcessCreated	gab-intern-vm	gab33intern	powershell.exe	"powershell.exe" -NoProfile -ExecutionPolicy Bypass -WindowStyle Hidden -File C:\Users\gab33intern\Downloads\SupportTools1	svchost.exe
> 10/9/2025, 1:13:12.526 PM		ProcessCreated	gab-intern-vm	gab33intern	powershell.exe	"powershell.exe" -NoProfile -ExecutionPolicy Bypass -WindowStyle Hidden -File 'C:\Users\gab33intern\Downloads\SupportTools1'	explorer.exe

TimeGenerated [UTC]	2025-10-09T12:34:57.7652337Z
ActionType	ProcessCreated
DeviceName	gab-intern-vm
FileName	conhost.exe
FolderPath	C:\Windows\System32\conhost.exe
InitiatingProcessCommandLine	"powershell.exe" -NoProfile -WindowStyle Hidden -Command Write-Output 'Set-MpPreference -DisableRealtimeMonitoring \$true'   Out-File -FilePath 'C:\Users\Public\DefenderTamperArtifact.txt' -Encoding ASCII -Append
InitiatingProcessFolderPath	c:\windows\system32\windowspowershell\v1.0\powershell.exe
InitiatingProcessParentFileName	powershell.exe

I decided to check DeviceFileEvents table and query for ' artifact ' in the FileName column.

```
//-----FLAG 2-----
DeviceFileEvents
| where DeviceName == "gab-intern-vm"
| where ActionType == "FileCreated"
| where FileName contains "Artifact"
| where TimeGenerated between (datetime(2025-10-09T00:00:00Z) .. datetime(2025-10-15T23:59:59Z))
| project TimeGenerated, ActionType, DeviceName, FileName, FolderPath, InitiatingProcessCommandLine, InitiatingProcessFolderPath, InitiatingProcessParentFileName
```

TimeGenerated [UTC]	ActionType	DeviceName	FileName	FolderPath	InitiatingProcessCommandLine
> 10/9/2025, 12:34:59.126 PM	FileCreated	gab-intern-vm	DefenderTamperArtif...	C:\Users\gab33intern\AppData\LR...	Explorer.EXE
> 10/9/2025, 12:58:17.436 PM	FileCreated	gab-intern-vm	ReconArtifacts.zip	C:\Users\Public\ReconArtifacts.zip	"powershell.exe"
> 10/9/2025, 12:59:05.680 PM	FileCreated	gab-intern-vm	ReconArtifacts.zip	C:\Users\gab33intern\Documents...	Explorer.EXE

For the query, I kept using ' Artifact ' and used this information to see if there was another file name related to the term.

I found ' ReconArtifacts.zip ' and then saw that there was a

'DefenderTamperArtifact.lnk ' file.

The timestamp matches with process creation from the 'DeviceProcessEvents' table

The '.lnk ' file extension is a shortcut of the filename. Upon researching '.LNK ' files, they are often the trigger for malicious scripts and can be used for malicious purposes.



TimeGenerated (UTC) ↑	ActionType	DeviceName	FileName	FolderPath	InitiatingProcessCommandLine	InitiatingProcessFolderPath	InitiatingProcessParentFileName
> 10/9/2025, 12:15:13.885 PM	FileCreated	gab-intern-vm	energy-report.html	C:\ProgramData\Microsoft\Win...	taskhost.exe - ExploitGuardPolicy	c:\windows\system32\taskhost.exe	schtost.exe
> 10/9/2025, 12:22:22.013 PM	FileCreated	gab-intern-vm	..._PSScriptPolicyTest_k1n41yzi...	C:\Users\gabril3lntern\AppData...	"powershell.exe"	c:\windows\system32\windowspowershell\v1.0\powershell.exe	RuntimeBroker.exe
> 10/9/2025, 12:22:22.013 PM	FileCreated	gab-intern-vm	..._PSScriptPolicyTest_2czfyo.ho...	C:\Users\gabril3lntern\AppData...	"powershell.exe"	c:\windows\system32\windowspowershell\v1.0\powershell.exe	RuntimeBroker.exe
> 10/9/2025, 12:22:27.651 PM	FileCreated	gab-intern-vm	SupportTool.ps1	C:\Users\gabril3lntern\Downloa...	"powershell.exe"	c:\windows\system32\windowspowershell\v1.0\powershell.exe	RuntimeBroker.exe
> 10/9/2025, 12:22:27.739 PM	FileCreated	gab-intern-vm	..._PSScriptPolicyTest_5x1pk3co...	C:\Users\gabril3lntern\AppData...	"powershell.exe" - ExecutionPoli...	c:\windows\system32\windowspowershell\v1.0\powershell.exe	powershell.exe
> 10/9/2025, 12:22:27.739 PM	FileCreated	gab-intern-vm	..._PSScriptPolicyTest_kcfpubs.d...	C:\Users\gabril3lntern\AppData...	"powershell.exe" - ExecutionPoli...	c:\windows\system32\windowspowershell\v1.0\powershell.exe	powershell.exe
> 10/9/2025, 12:22:56.670 PM	FileCreated	gab-intern-vm	SupportToolLink	C:\Users\gabril3lntern\AppData...	Explorer.EXE	c:\windows\explorer.exe	userinit.exe
> 10/9/2025, 12:22:56.923 PM	FileCreated	gab-intern-vm	DownloadsLink	C:\Users\gabril3lntern\AppData...	Explorer.EXE	c:\windows\explorer.exe	userinit.exe
> 10/9/2025, 12:23:31.361 PM	FileCreated	gab-intern-vm	..._PSScriptPolicyTest_x4gcmk.a...	C:\Users\gabril3lntern\AppData...	"powershell.exe" - "Command" ...	c:\windows\system32\windowspowershell\v1.0\powershell.exe	explorer.exe
> 10/9/2025, 12:23:31.362 PM	FileCreated	gab-intern-vm	..._PSScriptPolicyTest_rniekdb.5...	C:\Users\gabril3lntern\AppData...	"powershell.exe" - "Command" ...	c:\windows\system32\windowspowershell\v1.0\powershell.exe	explorer.exe
> 10/9/2025, 12:23:31.613 PM	FileCreated	gab-intern-vm	..._PSScriptPolicyTest_ibwzagi.1...	C:\Users\gabril3lntern\AppData...	"powershell.exe" - "Command" ...	c:\windows\system32\windowspowershell\v1.0\powershell.exe	explorer.exe
> 10/9/2025, 12:23:31.613 PM	FileCreated	gab-intern-vm	..._PSScriptPolicyTest_gg5nba.u...	C:\Users\gabril3lntern\AppData...	"powershell.exe" - "Command" ...	c:\windows\system32\windowspowershell\v1.0\powershell.exe	explorer.exe
> 10/9/2025, 12:28:32.929 PM	FileCreated	gab-intern-vm	..._PSScriptPolicyTest_h4dpym...	C:\Windows\ServiceProfiles\Loc...	powershell.exe - ExecutionPolic...	c:\windows\system32\windowspowershell\v1.0\powershell.exe	Sensell.exe
> 10/9/2025, 12:28:32.929 PM	FileCreated	gab-intern-vm	..._PSScriptPolicyTest_d8kkbk3.v...	C:\Windows\ServiceProfiles\Loc...	powershell.exe - ExecutionPolic...	c:\windows\system32\windowspowershell\v1.0\powershell.exe	Sensell.exe
> 10/9/2025, 12:34:58.130 PM	FileCreated	gab-intern-vm	..._PSScriptPolicyTest_0qyptw.1...	C:\Users\gabril3lntern\AppData...	"powershell.exe" -NoProfile -W...	c:\windows\system32\windowspowershell\v1.0\powershell.exe	powershell.exe
> 10/9/2025, 12:34:58.130 PM	FileCreated	gab-intern-vm	..._PSScriptPolicyTest_es3qphq...	C:\Users\gabril3lntern\AppData...	"powershell.exe" -NoProfile -W...	c:\windows\system32\windowspowershell\v1.0\powershell.exe	powershell.exe
> 10/9/2025, 12:34:59.126 PM	FileCreated	gab-intern-vm	DefenderTamperArtifactLink	C:\Users\gabril3lntern\AppData...	Explorer.EXE	c:\windows\explorer.exe	userinit.exe
> 10/9/2025, 12:34:59.135 PM	FileCreated	gab-intern-vm	PublicLink	C:\Users\gabril3lntern\AppData...	Explorer.EXE	c:\windows\explorer.exe	userinit.exe
> 10/9/2025, 12:40:19.458 PM	FileCreated	gab-intern-vm	mpasdlta.vdm	C:\Windows\Temp\05650F3A-5...	MpSigStub.exe /stub 1.1.24010...	c:\windows\system32\mpsigstub.exe	AM_Delta_Patch_1439.37.0.exe
> 10/9/2025, 12:40:19.596 PM	FileCreated	gab-intern-vm	mpavdlta.vdm	C:\Windows\Temp\05650F3A-5...	MpSigStub.exe /stub 1.1.24010...	c:\windows\system32\mpsigstub.exe	AM_Delta_Patch_1439.37.0.exe
> 10/9/2025, 12:48:55.871 PM	FileCreated	gab-intern-vm	..._PSScriptPolicyTest_er11xyfo...	C:\Windows\SystemTemp\PS_...	powershell.exe -NoProfile -Non...	c:\windows\system32\windowspowershell\v1.0\powershell.exe	cmd.exe
> 10/9/2025, 12:48:55.871 PM	FileCreated	gab-intern-vm	..._PSScriptPolicyTest_yendind.c...	C:\Windows\SystemTemp\PS_...	powershell.exe -NoProfile -Non...	c:\windows\system32\windowspowershell\v1.0\powershell.exe	cmd.exe
> 10/9/2025, 12:48:55.960 PM	FileCreated	gab-intern-vm	script42.ps1	C:\Packages\Plugins\Microsof...	RunCommandExtension.exe "e...	c:\packages\plugins\microsoft.rpld.exe\runcommandextension\1.1.22\bin\runco...	cmd.exe
> 10/9/2025, 12:48:56.243 PM	FileCreated	gab-intern-vm	..._PSScriptPolicyTest_yuvsorb.L...	C:\Windows\SystemTemp\PS_...	powershell - ExecutionPolicy Un...	c:\windows\system32\windowspowershell\v1.0\powershell.exe	cmd.exe
> 10/9/2025, 12:48:56.244 PM	FileCreated	gab-intern-vm	..._PSScriptPolicyTest_m43kz3p...	C:\Windows\SystemTemp\PS_...	powershell - ExecutionPolicy Un...	c:\windows\system32\windowspowershell\v1.0\powershell.exe	cmd.exe
> 10/9/2025, 12:48:57.198 PM	FileCreated	gab-intern-vm	..._PSScriptPolicyTest_g4kylo.1.t...	C:\Windows\SystemTemp\PS_...	powershell.exe - ExecutionPolic...	c:\windows\system32\windowspowershell\v1.0\powershell.exe	cmd.exe

> 10/9/2025, 12:57:33.279 PM	FileCreated	gab-intern-vm	WireServerLostPluginVersions.xml	D:\CollectGuestLogs\Temp\Wire...	"CollectGuestLog.exe" - Mode...	c:\windowsazure\guestagent\2.7.41491.1172-2025-10-06_054131\collectguestlo...	WaAppAgent.exe
> 10/9/2025, 12:57:36.160 PM	FileCreated	gab-intern-vm	..._PSScriptPolicyTest_oqsgw.1.npr.ps1	C:\Windows\SystemTemp\PS_...	gc_worker.exe -a AzureWindow...	c:\packages\plugins\microsoft.guestconfiguration.configurationforwindows\1.29...	gc_service.exe
> 10/9/2025, 12:57:36.160 PM	FileCreated	gab-intern-vm	..._PSScriptPolicyTest_20bk2nrcu.ps1	C:\Windows\SystemTemp\PS_...	gc_worker.exe -a AzureWindow...	c:\packages\plugins\microsoft.guestconfiguration.configurationforwindows\1.29...	gc_service.exe
> 10/9/2025, 12:58:15.329 PM	FileCreated	gab-intern-vm	..._PSScriptPolicyTest_y2bhl0Qj6.ps1	C:\Windows\SystemTemp\PS_...	gc_worker.exe -a AuditSecureRe...	c:\packages\plugins\microsoft.guestconfiguration.configurationforwindows\1.29...	gc_service.exe
> 10/9/2025, 12:58:15.329 PM	FileCreated	gab-intern-vm	..._PSScriptPolicyTest_2ogpakm.c5c.ps1	C:\Windows\SystemTemp\PS_...	gc_worker.exe -a AuditSecureRe...	c:\packages\plugins\microsoft.guestconfiguration.configurationforwindows\1.29...	gc_service.exe
> 10/9/2025, 12:58:16.801 PM	FileCreated	gab-intern-vm	SupportTool.ps1	C:\Users\Public\RemoteAssistB...	"powershell.exe"	c:\windows\system32\windowspowershell\v1.0\powershell.exe	RuntimeBroker.exe
> 10/9/2025, 12:58:17.436 PM	FileCreated	gab-intern-vm	ReconArtifacts.zip	C:\Users\Public\ReconArtifacts...	"powershell.exe"	c:\windows\system32\windowspowershell\v1.0\powershell.exe	RuntimeBroker.exe
> 10/9/2025, 12:59:05.680 PM	FileCreated	gab-intern-vm	ReconArtifacts.zip	C:\Users\gabril3lntern\Docume...	Explorer.EXE	c:\windows\explorer.exe	userinit.exe
> 10/9/2025, 12:59:24.830 PM	FileCreated	gab-intern-vm	..._PSScriptPolicyTest_hdjytm.dyh.ps1	C:\Windows\SystemTemp\PS_...	powershell.exe - ExecutionPolic...	c:\windows\system32\windowspowershell\v1.0\powershell.exe	Sensell.exe
> 10/9/2025, 12:59:24.830 PM	FileCreated	gab-intern-vm	..._PSScriptPolicyTest_y54voitc.tps1	C:\Windows\SystemTemp\PS_...	powershell.exe - ExecutionPolic...	c:\windows\system32\windowspowershell\v1.0\powershell.exe	Sensell.exe
> 10/9/2025, 12:59:51.459 PM	FileCreated	gab-intern-vm	\$184KVF.zip	C:\\$Recycle.Bin\S-1-5-21-1259...	Explorer.EXE	c:\windows\explorer.exe	userinit.exe
> 10/9/2025, 12:59:56.900 PM	FileCreated	gab-intern-vm	..._PSScriptPolicyTest_0mcyau.yuc.ps1	C:\Windows\SystemTemp\PS_...	gc_worker.exe -a MinimumPass...	c:\packages\plugins\microsoft.guestconfiguration.configurationforwindows\1.29...	gc_service.exe
> 10/9/2025, 12:59:56.900 PM	FileCreated	gab-intern-vm	..._PSScriptPolicyTest_ke5ovqz.gbn.ps1	C:\Windows\SystemTemp\PS_...	gc_worker.exe -a MinimumPass...	c:\packages\plugins\microsoft.guestconfiguration.configurationforwindows\1.29...	gc_service.exe
> 10/9/2025, 13:00:36.639 PM	FileCreated	gab-intern-vm	..._PSScriptPolicyTest_cagid2.2dy.ps1	C:\Windows\SystemTemp\PS_...	gc_worker.exe -a PasswordMus...	c:\packages\plugins\microsoft.guestconfiguration.configurationforwindows\1.29...	gc_service.exe
> 10/9/2025, 13:01:17.616 PM	FileCreated	gab-intern-vm	..._PSScriptPolicyTest_oymekc.vfk.ps1	C:\Windows\SystemTemp\PS_...	gc_worker.exe -a PasswordMus...	c:\packages\plugins\microsoft.guestconfiguration.configurationforwindows\1.29...	gc_service.exe
> 10/9/2025, 13:01:17.616 PM	FileCreated	gab-intern-vm	..._PSScriptPolicyTest_3v5pkcq.gwr.ps1	C:\Windows\SystemTemp\PS_...	gc_worker.exe -a MaximumPas...	c:\packages\plugins\microsoft.guestconfiguration.configurationforwindows\1.29...	gc_service.exe
> 10/9/2025, 13:01:17.616 PM	FileCreated	gab-intern-vm	..._PSScriptPolicyTest_u0lbp4dv21.ps1	C:\Windows\SystemTemp\PS_...	gc_worker.exe -a MaximumPas...	c:\packages\plugins\microsoft.guestconfiguration.configurationforwindows\1.29...	gc_service.exe
> 10/9/2025, 13:01:56.202 PM	FileCreated	gab-intern-vm	..._PSScriptPolicyTest_ur42oah.wc2.ps1	C:\Windows\SystemTemp\PS_...	gc_worker.exe -a MinimumPass...	c:\packages\plugins\microsoft.guestconfiguration.configurationforwindows\1.29...	gc_service.exe
> 10/9/2025, 13:01:56.202 PM	FileCreated	gab-intern-vm	..._PSScriptPolicyTest_24n25n.zc2.ps1	C:\Windows\SystemTemp\PS_...	gc_worker.exe -a MinimumPass...	c:\packages\plugins\microsoft.guestconfiguration.configurationforwindows\1.29...	gc_service.exe
> 10/9/2025, 13:02:35.390 PM	FileCreated	gab-intern-vm	..._PSScriptPolicyTest_b22kv3.3b0.ps1	C:\Windows\SystemTemp\PS_...	gc_worker.exe -a EnforcePassw...	c:\packages\plugins\microsoft.guestconfiguration.configurationforwindows\1.29...	gc_service.exe
> 10/9/2025, 13:02:35.390 PM	FileCreated	gab-intern-vm	..._PSScriptPolicyTest_14tnvyv.cof.ps1	C:\Windows\SystemTemp\PS_...	gc_worker.exe -a EnforcePassw...	c:\packages\plugins\microsoft.guestconfiguration.configurationforwindows\1.29...	gc_service.exe



### Flag 3 – Quick Data Probe

**Objective:**

Spot brief, opportunistic checks for readily available sensitive content.

**What to Hunt:**

Find short-lived actions that attempt to read transient data sources common on endpoints.

**Thought:**

Attackers look for low-effort wins first; these quick probes often precede broader reconnaissance.

**Hint:**

1. Clip

**Side Note: 1/2**

1. has query

Provide the command value tied to this particular exploit \*

```
"powershell.exe" -NoProfile -Sta -Command "tr
```

For this flag I imagined the command value had something to do with copy and paste actions as it is a common short-lived action.

The other part to this was the term ' query '

I decided to check the ' InitiateProcessCommandLine ' column and find syntax and flags that looked like it was written as a query.

Upon looking I kept my focus on the timeline of the script and tried to match up the time .

The InitiatingProcessCommandLine showed this command below when querying for 'clip'

```
"powershell.exe" -NoProfile -Sta -Command "try { Get-Clipboard | Out-Null } catch { }"
```

This specific activity related to powershell has the syntax for a query such as

```
"try { Get-Clipboard | Out-Null } catch { }"
```

```
//-----FLAG 3-----  
DeviceFileEvents  
| where DeviceName == "gab-intern-vm"  
| where InitiatingProcessCommandLine contains "clip"  
| where TimeGenerated between (datetime(2025-10-09T00:00:00Z) .. datetime(2025-10-15T23:59:59Z))  
| project TimeGenerated, ActionType, DeviceName, FileName, FolderPath, InitiatingProcessCommandLine, InitiatingProcessFolderPath, InitiatingProcessFileName, InitiatingProcessParentFileName
```

TimeGenerated [UTC]	ActionType	DeviceName	FileName	FolderPath	InitiatingProcessCommandLine
> 10/9/2025, 12:50:40.032 PM	FileCreated	gab-intern-vm	__PSScriptPolicyTest_...	C:\Users\g4bri3lntern\AppData\L...	"powershell.exe" -NoProfile -Sta -Command "try { Get-Clipboard   Out-Null } catch {}"
10/9/2025, 12:50:40.032 PM	FileCreated	gab-intern-vm	__PSScriptPolicyTest_...	C:\Users\g4bri3lntern\AppData\L...	"powershell.exe" -NoProfile -Sta -Command "try { Get-Clipboard   Out-Null } catch {}"
TimeGenerated [UTC]	2025-10-09T12:50:40.0328347Z				
ActionType	FileCreated				
DeviceName	gab-intern-vm				
FileName	__PSScriptPolicyTest_zvcqknciv5n.psm1				
FolderPath	C:\Users\g4bri3lntern\AppData\Local\Temp\_PSScriptPolicyTest_zvcqknciv5n.psm1				
InitiatingProcessCommandLine	"powershell.exe" -NoProfile -Sta -Command "try { Get-Clipboard   Out-Null } catch {}"				
InitiatingProcessFolderPath	c:\windows\system32\windowspowershell\v1.0\powershell.exe				
InitiatingProcessFileName	powershell.exe				
InitiatingProcessParentFileName	powershell.exe				

## Flag 4 – Host Context Recon

### Objective:

Find activity that gathers basic host and user context to inform follow-up actions.

### What to Hunt:

Telemetry that shows the actor collecting environment or account details without modifying them.

### Thought:

Context-gathering shapes attacker decisions — who, what, and where to target next.

### Hint:

1. qwi

Point out when the last recon attempt was \*

2025-10-09T12:51:44.3425653Z

While going through the logs, and reading this flag I recall seeing an executable called ' qwinsta.exe ' I had to look up this program and it is a command on windows that can:

```
'Display information about sessions on a Remote Desktop Session Host server'
```

This made sense in terms of gathering host and user context information.

Working within the timeline of 12:51:44 PM we can see that this was the last recon attempt for the query session for the attacker to enumerate.

```
//-----FLAG 4-----
DeviceProcessEvents
| where DeviceName == "gab-intern-vm"
| where AccountName == "g4bri3lintern"
| where ProcessCommandLine contains "qwi"
| where TimeGenerated between (datetime(2025-10-01T00:00:00Z) .. datetime(2025-10-20T23:59:59Z))
| project TimeGenerated, AccountDomain, AccountName, ActionType, DeviceName, FileName, InitiatingProcessCommandLine, InitiatingProcessFileName
```

>	10/9/2025, 12:51:44.308 PM	gab-intern-vm	g4bri3lintern	ProcessCreated	cmd.exe	"cmd.exe" /c query session	"powershell.exe"	powershell.exe
>	10/9/2025, 12:51:44.327 PM	gab-intern-vm	g4bri3lintern	ProcessCreated	query.exe	query session	"cmd.exe" /c query session	cmd.exe
>	10/9/2025, 12:51:44.342 PM	gab-intern-vm	g4bri3lintern	ProcessCreated	qwinsta.exe	"qwinsta.exe"	query session	query.exe

## Flag 5 – Storage Surface Mapping

### Objective:

Detect discovery of local or network storage locations that might hold interesting data.

### What to Hunt:

Look for enumeration of filesystem or share surfaces and lightweight checks of available storage.

### Thought:

Mapping where data lives is a preparatory step for collection and staging.

### Hint:

1. Storage assessment

Provide the 2nd command tied to this activity \*

"cmd.exe" /c wmic logicaldisk get name,freespace

After looking at the 'qwinsta.exe' process that was created in the logs.

I noticed the command prompt executable that showed logical disk that comes after the 'qwinsta.exe' executable.

This made sense in terms of data as to where it lives and the data that can be discovered such as 'storage'. Decided to search for 'WMIC.exe' command and found out that the 'logical disk'

We can see the 'TimeGenerated' column is still within 12:50:00 PM-12:51:00 PM.

```
"cmd.exe" /c wmic logicaldisk get name,freespace,size
```

```
//-----FLAG 5-----
DeviceProcessEvents
| where DeviceName == "gab-intern-vm"
| where AccountName == "g4bri3lintern"
| where TimeGenerated between (datetime(2025-10-01T00:00:00Z) .. datetime(2025-10-20T23:59:59Z))
| project TimeGenerated, AccountDomain, AccountName, ActionType, FileName, ProcessCommandLine, InitiatingProcessCommandLine, InitiatingProcessFileName
```

TimeGenerated [UTC] ↑	AccountDomain	AccountName	ActionType	FileName	ProcessCommandLine
> 10/9/2025, 12:49:32.243 PM	gab-intern-vm	g4bri3lintern	ProcessCreated	smartscreen.exe	smartscreen.exe -Embedding
> 10/9/2025, 12:50:16.419 PM	gab-intern-vm	g4bri3lintern	ProcessCreated	powershell.exe	"powershell.exe"
> 10/9/2025, 12:50:16.451 PM	gab-intern-vm	g4bri3lintern	ProcessCreated	conhost.exe	conhost.exe 0xffffffff -ForceV1
> 10/9/2025, 12:50:39.955 PM	gab-intern-vm	g4bri3lintern	ProcessCreated	powershell.exe	"powershell.exe" -NoProfile -Sta -Command "try { Get-Clipboard   Out-Null } catch {}"
> 10/9/2025, 12:50:58.317 PM	gab-intern-vm	g4bri3lintern	ProcessCreated	cmd.exe	"cmd.exe" /c quser
> 10/9/2025, 12:50:58.364 PM	gab-intern-vm	g4bri3lintern	ProcessCreated	quser.exe	quser
> 10/9/2025, 12:50:59.344 PM	gab-intern-vm	g4bri3lintern	ProcessCreated	cmd.exe	"cmd.exe" /c qwinsta
> 10/9/2025, 12:50:59.369 PM	gab-intern-vm	g4bri3lintern	ProcessCreated	qwinsta.exe	qwinsta
> 10/9/2025, 12:51:17.366 PM	gab-intern-vm	g4bri3lintern	ProcessCreated	cmd.exe	"cmd.exe" /c net use
> 10/9/2025, 12:51:17.389 PM	gab-intern-vm	g4bri3lintern	ProcessCreated	net.exe	net use
> 10/9/2025, 12:51:18.384 PM	gab-intern-vm	g4bri3lintern	ProcessCreated	cmd.exe	"cmd.exe" /c wmic logicaldisk get name,freespace,size
> 10/9/2025, 12:51:18.562 PM	gab-intern-vm	g4bri3lintern	ProcessCreated	WMIC.exe	wmic logicaldisk get name,freespace,size
> 10/9/2025, 12:51:31.569 PM	gab-intern-vm	g4bri3lintern	ProcessCreated	cmd.exe	"cmd.exe" /c ipconfig /all
> 10/9/2025, 12:51:31.582 PM	gab-intern-vm	g4bri3lintern	ProcessCreated	ipconfig.exe	ipconfig /all
> 10/9/2025, 12:51:32.590 PM	gab-intern-vm	g4bri3lintern	ProcessCreated	cmd.exe	"cmd.exe" /c nslookup helpdesk-telemetry.remoteassist.invalid
> 10/9/2025, 12:51:32.622 PM	gab-intern-vm	g4bri3lintern	ProcessCreated	nslookup.exe	nslookup helpdesk-telemetry.remoteassist.invalid
> 10/9/2025, 12:51:44.308 PM	gab-intern-vm	g4bri3lintern	ProcessCreated	cmd.exe	"cmd.exe" /c query session
> 10/9/2025, 12:51:44.327 PM	gab-intern-vm	g4bri3lintern	ProcessCreated	query.exe	query session
> 10/9/2025, 12:51:44.342 PM	gab-intern-vm	g4bri3lintern	ProcessCreated	qwinsta.exe	"qwinsta.exe"
> 10/9/2025, 12:51:57.639 PM	gab-intern-vm	g4bri3lintern	ProcessCreated	cmd.exe	"cmd.exe" /c tasklist /v
> 10/9/2025, 12:51:57.686 PM	gab-intern-vm	g4bri3lintern	ProcessCreated	tasklist.exe	tasklist /v
> 10/9/2025, 12:52:14.313 PM	gab-intern-vm	g4bri3lintern	ProcessCreated	cmd.exe	"cmd.exe" /c whoami /groups
> 10/9/2025, 12:52:14.364 PM	gab-intern-vm	g4bri3lintern	ProcessCreated	whoami.exe	whoami /groups
> 10/9/2025, 12:52:15.322 PM	gab-intern-vm	g4bri3lintern	ProcessCreated	cmd.exe	"cmd.exe" /c whoami /priv
> 10/9/2025, 12:52:15.339 PM	gab-intern-vm	g4bri3lintern	ProcessCreated	whoami.exe	whoami /priv

## Flag 6 – Connectivity & Name Resolution Check

### Objective:

Identify checks that validate network reachability and name resolution.

### What to Hunt:

Network or process events indicating DNS or interface queries and simple outward connectivity probes.

### Thought:

Confirming egress is a necessary precondition before any attempt to move data off-host.

### Side Note: 2/2

1. session

Provide the File Name of the initiating parent process \*

RuntimeBroker.exe

What was key to this question was network related events.  
Especially when it comes to DNS and outbound connections.

I decided to check the 'InitiatingProcessParentFileName' column in the 'DeviceNetworkEvents' table and try to narrow down unusual PowerShell activity.

I made sure to stay focused on October 9th 2025 during the time of 12:50-12:51 PM as other events from DeviceProcessEvents and DeviceFileEvents were very important in relation to SupportToolScript.ps1. Powershell has been very prevalent throughout the hunt.

```
//-----FLAG 6-----  
DeviceNetworkEvents  
| where DeviceName == "gab-intern-vm"  
| where ActionType == "ConnectionSuccess"  
| where InitiatingProcessFileName == "powershell.exe"  
| where TimeGenerated between ((datetime(2025-10-09T00:00:00Z)) .. datetime(2025-10-09T23:59:59Z))  
| project TimeGenerated, ActionType, AdditionalFields, DeviceName, InitiatingProcessFileName, InitiatingProcessFolderPath, InitiatingProcessId, InitiatingProcessParentFileName, Protocol, RemoteIP, RemoteIPType, RemotePort, InitiatingProcessRemoteSessionIP
```

TimeGenerated (UTC)	ActionType	AdditionalFields	DeviceName	InitiatingProcessFileName	InitiatingProcessFolderPath	InitiatingProcessId	InitiatingProcessParentFileName	Protocol	RemoteIP	RemoteIPType
> 10/7/2025, 12:25:11.994 PM	ConnectionSuccess		gab-intern-vm	powershell.exe	c:\windows\system32\windowspowershell\1...	3808	cmd.exe	Tcp	185.199.109.133	Public
> 10/7/2025, 3:52:57.636 PM	ConnectionSuccess		gab-intern-vm	powershell.exe	c:\windows\system32\windowspowershell\1...	7296	SenseI.exe	Tcp	20.10.127.193	Public
> 10/7/2025, 4:37:54.433 PM	ConnectionSuccess		gab-intern-vm	powershell.exe	c:\windows\system32\windowspowershell\1...	9056	cmd.exe	Tcp	10.0.0.5	Private
> 10/7/2025, 4:37:58.752 PM	ConnectionSuccess		gab-intern-vm	powershell.exe	c:\windows\system32\windowspowershell\1...	9056	cmd.exe	Tcp	10.0.0.5	Private
> 10/7/2025, 4:38:07.573 PM	ConnectionSuccess		gab-intern-vm	powershell.exe	c:\windows\system32\windowspowershell\1...	9056	cmd.exe	Tcp	10.0.0.5	Private
> 10/7/2025, 4:38:18.325 PM	ConnectionSuccess		gab-intern-vm	powershell.exe	c:\windows\system32\windowspowershell\1...	9056	cmd.exe	Tcp	10.0.0.5	Private
> 10/7/2025, 4:48:58.538 PM	ConnectionSuccess		gab-intern-vm	powershell.exe	c:\windows\system32\windowspowershell\1...	5076	cmd.exe	Tcp	185.199.110.133	Public
> 10/7/2025, 4:49:02.156 PM	ConnectionSuccess		gab-intern-vm	powershell.exe	c:\windows\system32\windowspowershell\1...	6604	cmd.exe	Tcp	20.60.181.193	Public
> 10/7/2025, 4:49:09.049 PM	ConnectionSuccess		gab-intern-vm	powershell.exe	c:\windows\system32\windowspowershell\1...	6604	cmd.exe	Tcp	20.60.133.132	Public
> 10/9/2025, 4:37:05.119 AM	ConnectionSuccess		gab-intern-vm	powershell.exe	c:\windows\system32\windowspowershell\1...	96	cmd.exe	Tcp	185.199.108.133	Public
> 10/9/2025, 12:13:00.509 PM	ConnectionSuccess		gab-intern-vm	powershell.exe	c:\windows\system32\windowspowershell\1...	8272	cmd.exe	Tcp	185.199.110.133	Public
> 10/9/2025, 12:25:03.796 PM	ConnectionSuccess		gab-intern-vm	powershell.exe	c:\windows\system32\windowspowershell\1...	6636	cmd.exe	Tcp	185.199.108.133	Public
> 10/9/2025, 12:37:39.443 PM	ConnectionSuccess		gab-intern-vm	powershell.exe	c:\windows\system32\windowspowershell\1...	2256	cmd.exe	Tcp	10.0.0.5	Private
> 10/9/2025, 12:37:43.845 PM	ConnectionSuccess		gab-intern-vm	powershell.exe	c:\windows\system32\windowspowershell\1...	2256	cmd.exe	Tcp	10.0.0.5	Private
> 10/9/2025, 12:37:52.623 PM	ConnectionSuccess		gab-intern-vm	powershell.exe	c:\windows\system32\windowspowershell\1...	2256	cmd.exe	Tcp	10.0.0.5	Private
> 10/9/2025, 12:38:03.433 PM	ConnectionSuccess		gab-intern-vm	powershell.exe	c:\windows\system32\windowspowershell\1...	2256	cmd.exe	Tcp	10.0.0.5	Private
> 10/9/2025, 12:49:01.040 PM	ConnectionSuccess		gab-intern-vm	powershell.exe	c:\windows\system32\windowspowershell\1...	6736	cmd.exe	Tcp	20.60.181.193	Public
> 10/9/2025, 12:49:07.962 PM	ConnectionSuccess		gab-intern-vm	powershell.exe	c:\windows\system32\windowspowershell\1...	6736	cmd.exe	Tcp	20.60.133.132	Public
> 10/9/2025, 12:55:05.765 PM	ConnectionSuccess		gab-intern-vm	powershell.exe	c:\windows\system32\windowspowershell\1...	8824	RuntimedBroker.exe	Tcp	23.218.218.182	Public
> 10/9/2025, 1:00:39.393 PM	ConnectionSuccess		gab-intern-vm	powershell.exe	c:\windows\system32\windowspowershell\1...	8824	RuntimedBroker.exe	Tcp	23.192.228.80	Public
> 10/9/2025, 1:00:40.045 PM	ConnectionSuccess		gab-intern-vm	powershell.exe	c:\windows\system32\windowspowershell\1...	8824	RuntimedBroker.exe	Tcp	100.29.147.161	Public
> 10/9/2025, 8:12:47.638 PM	ConnectionSuccess		gab-intern-vm	powershell.exe	c:\windows\system32\windowspowershell\1...	7928	cmd.exe	Tcp	185.199.111.133	Public
> 10/9/2025, 10:43:35.788 PM	ConnectionSuccess		gab-intern-vm	powershell.exe	c:\windows\system32\windowspowershell\1...	7844	SenseI.exe	Tcp	20.10.127.192	Public

## Flag 7 – Interactive Session Discovery

### Objective:

Reveal attempts to detect interactive or active user sessions on the host.

### What to Hunt:

Signals that enumerate current session state or logged-in sessions without initiating a takeover.

### Thought:

Knowing which sessions are active helps an actor decide whether to act immediately or wait.

What is the unique ID of the initiating process \*

2533274790397065

Keywords: Session, Initiate Process, Unique

Had to get a little help with this one.

It was actually really simple. When I read the question "What is the unique ID of the initiating process?" I kept focus for the column InitiatingProcessID

I was so stumped that I feel the process identification task number was staring at me. Luckily I got the hint from a user to project UniqueProcessId

I should have considered the term ' Unique ' in order to find the InitiatingProcessUniqueId number

'2533274790397065'

```
DeviceProcessEvents
| where DeviceName == "gab-intern-vm"
| where AccountName == "g4br131intern"
| where TimeGenerated between (datetime(2025-10-09T00:00:00Z) .. datetime(2025-10-10T23:59:59Z))
| project TimeGenerated, AccountName, ActionType, DeviceName, FileName, ProcessCommandLine, InitiatingProcessFileName, ProcessUniqueId, ProcessId, InitiatingProcessUniqueId, InitiatingProcessId, InitiatingProcessParentId
```

TimeGenerated [UTC]	AccountName	ActionType	DeviceName	FileName	ProcessCommandLine	InitiatingProcessFileName	ProcessUniqueId	ProcessId	InitiatingProcessParentId
> 10/9/2025, 12:50:16.419 PM	g4br131intern	ProcessCreated	gab-intern-vm	powershell.exe	"powershell.exe"	runtimebroker.exe	2533274790397065	8824	2533274790397075
> 10/9/2025, 12:50:16.451 PM	g4br131intern	ProcessCreated	gab-intern-vm	conhost.exe	conhost.exe /d:fffff -ForceV1	powershell.exe	2533274790397066	8416	2533274790397065
> 10/9/2025, 12:50:39.955 PM	g4br131intern	ProcessCreated	gab-intern-vm	powershell.exe	"powershell.exe" -NoProfile -Sta -Command "try {Get-Clipboard   Out-Null} catch {}"	powershell.exe	2533274790397067	8396	2533274790397065
> 10/9/2025, 12:50:58.317 PM	g4br131intern	ProcessCreated	gab-intern-vm	cmd.exe	"cmd.exe" /c qwer	powershell.exe	2533274790397068	4312	2533274790397065
> 10/9/2025, 12:50:58.364 PM	g4br131intern	ProcessCreated	gab-intern-vm	qwer.exe	qwer	cmd.exe	2533274790397069	1960	2533274790397068
> 10/9/2025, 12:50:59.344 PM	g4br131intern	ProcessCreated	gab-intern-vm	cmd.exe	"cmd.exe" /c qwersta	powershell.exe	2533274790397070	7726	2533274790397065
> 10/9/2025, 12:50:59.369 PM	g4br131intern	ProcessCreated	gab-intern-vm	qwersta.exe	qwersta	cmd.exe	2533274790397071	878	2533274790397070
> 10/9/2025, 12:51:17.366 PM	g4br131intern	ProcessCreated	gab-intern-vm	cmd.exe	"cmd.exe" /c net use	powershell.exe	2533274790397073	2524	2533274790397065
> 10/9/2025, 12:51:17.389 PM	g4br131intern	ProcessCreated	gab-intern-vm	net.exe	net use	cmd.exe	2533274790397074	8012	2533274790397073
> 10/9/2025, 12:51:18.384 PM	g4br131intern	ProcessCreated	gab-intern-vm	cmd.exe	"cmd.exe" /c wmic logicaldisk get name, freespace, size	powershell.exe	2533274790397075	3556	2533274790397065
> 10/9/2025, 12:51:18.562 PM	g4br131intern	ProcessCreated	gab-intern-vm	WMIC.exe	wmic logicaldisk get name, freespace, size	cmd.exe	2533274790397076	5012	2533274790397075
> 10/9/2025, 12:51:31.569 PM	g4br131intern	ProcessCreated	gab-intern-vm	cmd.exe	"cmd.exe" /c ipconfig /all	powershell.exe	2533274790397078	8660	2533274790397065
> 10/9/2025, 12:51:31.582 PM	g4br131intern	ProcessCreated	gab-intern-vm	ipconfig.exe	ipconfig /all	cmd.exe	2533274790397079	4328	2533274790397078
> 10/9/2025, 12:51:32.590 PM	g4br131intern	ProcessCreated	gab-intern-vm	cmd.exe	"cmd.exe" /c nslookup helpdesk telemetry.remoteassist.invalid	powershell.exe	2533274790397082	7544	2533274790397065
> 10/9/2025, 12:51:32.622 PM	g4br131intern	ProcessCreated	gab-intern-vm	nslookup.exe	nslookup helpdesk telemetry.remoteassist.invalid	cmd.exe	2533274790397083	8500	2533274790397082
> 10/9/2025, 12:51:44.308 PM	g4br131intern	ProcessCreated	gab-intern-vm	cmd.exe	"cmd.exe" /c query session	powershell.exe	2533274790397087	7304	2533274790397065
> 10/9/2025, 12:51:44.327 PM	g4br131intern	ProcessCreated	gab-intern-vm	query.exe	query session	cmd.exe	2533274790397088	2528	2533274790397087
> 10/9/2025, 12:51:44.342 PM	g4br131intern	ProcessCreated	gab-intern-vm	qwersta.exe	"qwersta.exe"	query.exe	2533274790397089	6984	2533274790397088
> 10/9/2025, 12:51:57.639 PM	g4br131intern	ProcessCreated	gab-intern-vm	cmd.exe	"cmd.exe" /c tasklist /v	powershell.exe	2533274790397090	8412	2533274790397065
> 10/9/2025, 12:51:57.666 PM	g4br131intern	ProcessCreated	gab-intern-vm	tasklist.exe	tasklist /v	cmd.exe	2533274790397091	8792	2533274790397090
> 10/9/2025, 12:52:14.313 PM	g4br131intern	ProcessCreated	gab-intern-vm	cmd.exe	"cmd.exe" /c whoami /groups	powershell.exe	2533274790397092	4860	2533274790397065
> 10/9/2025, 12:52:14.364 PM	g4br131intern	ProcessCreated	gab-intern-vm	whoami.exe	whoami /groups	cmd.exe	2533274790397093	4692	2533274790397092
> 10/9/2025, 12:52:15.322 PM	g4br131intern	ProcessCreated	gab-intern-vm	cmd.exe	"cmd.exe" /c whoami /priv	powershell.exe	2533274790397094	4684	2533274790397065
> 10/9/2025, 12:52:15.339 PM	g4br131intern	ProcessCreated	gab-intern-vm	whoami.exe	whoami /priv	cmd.exe	2533274790397095	8548	2533274790397094

```
DeviceProcessEvents
| where DeviceName == "gab-intern-vm"
| where AccountName == "g4br131intern"
| where InitiatingProcessUniqueId contains "2533274790397065"
| where TimeGenerated between (datetime(2025-10-09T00:00:00Z) .. datetime(2025-10-10T23:59:59Z))
| project TimeGenerated, AccountName, ActionType, DeviceName, FileName, ProcessCommandLine, InitiatingProcessFileName, ProcessUniqueId, ProcessId, InitiatingProcessUniqueId, InitiatingProcessId, InitiatingProcessParentId
```

TimeGenerated [UTC]	AccountName	ActionType	DeviceName	FileName	ProcessCommandLine	InitiatingProcessFileName	InitiatingProcessUniqueId	InitiatingProcessId	InitiatingProcessParentId
> 10/9/2025, 12:50:16.451 PM	g4br131intern	ProcessCreated	gab-intern-vm	conhost.exe	conhost.exe /d:fffff -ForceV1	powershell.exe	2533274790397065	8824	6844
> 10/9/2025, 12:51:44.308 PM	g4br131intern	ProcessCreated	gab-intern-vm	cmd.exe	"cmd.exe" /c query session	powershell.exe	2533274790397065	8824	6844
> 10/9/2025, 12:51:57.639 PM	g4br131intern	ProcessCreated	gab-intern-vm	cmd.exe	"cmd.exe" /c tasklist /v	powershell.exe	2533274790397065	8824	6844
> 10/9/2025, 12:52:14.313 PM	g4br131intern	ProcessCreated	gab-intern-vm	cmd.exe	"cmd.exe" /c whoami /groups	powershell.exe	2533274790397065	8824	6844
> 10/9/2025, 12:52:15.322 PM	g4br131intern	ProcessCreated	gab-intern-vm	cmd.exe	"cmd.exe" /c whoami /priv	powershell.exe	2533274790397065	8824	6844
> 10/9/2025, 12:54:53.547 PM	g4br131intern	ProcessCreated	gab-intern-vm	cmd.exe	"cmd.exe" /c whoami /groups	powershell.exe	2533274790397065	8824	6844
> 10/9/2025, 12:54:54.559 PM	g4br131intern	ProcessCreated	gab-intern-vm	cmd.exe	"cmd.exe" /c whoami /priv	powershell.exe	2533274790397065	8824	6844
> 10/9/2025, 12:50:39.955 PM	g4br131intern	ProcessCreated	gab-intern-vm	powershell.exe	"powershell.exe" -NoProfile -Sta -	powershell.exe	2533274790397065	8824	6844
> 10/9/2025, 12:50:59.344 PM	g4br131intern	ProcessCreated	gab-intern-vm	cmd.exe	"cmd.exe" /c qwersta	powershell.exe	2533274790397065	8824	6844
> 10/9/2025, 12:50:58.317 PM	g4br131intern	ProcessCreated	gab-intern-vm	cmd.exe	"cmd.exe" /c qwer	powershell.exe	2533274790397065	8824	6844
> 10/9/2025, 12:51:18.384 PM	g4br131intern	ProcessCreated	gab-intern-vm	cmd.exe	"cmd.exe" /c wmic logicaldisk g_	powershell.exe	2533274790397065	8824	6844
> 10/9/2025, 12:51:17.366 PM	g4br131intern	ProcessCreated	gab-intern-vm	cmd.exe	"cmd.exe" /c net use	powershell.exe	2533274790397065	8824	6844
> 10/9/2025, 12:51:31.569 PM	g4br131intern	ProcessCreated	gab-intern-vm	cmd.exe	"cmd.exe" /c ipconfig /all	powershell.exe	2533274790397065	8824	6844
> 10/9/2025, 12:51:32.590 PM	g4br131intern	ProcessCreated	gab-intern-vm	cmd.exe	"cmd.exe" /c nslookup helpdes...	powershell.exe	2533274790397065	8824	6844
> 10/9/2025, 10:128.770 PM	g4br131intern	ProcessCreated	gab-intern-vm	schtasks.exe	"schtasks.exe" /Create /SC ONL...	powershell.exe	2533274790397065	8824	6844
> 10/9/2025, 10:129.781 PM	g4br131intern	ProcessCreated	gab-intern-vm	schtasks.exe	"schtasks.exe" /Query /TN Sup...	powershell.exe	2533274790397065	8824	6844



## Flag 8 – Runtime Application Inventory

### Objective:

Detect enumeration of running applications and services to inform risk and opportunity.

### What to Hunt:

Events that capture broad process/process-list snapshots or queries of running services.

### Thought:

A process inventory shows what's present and what to avoid or target for collection.

### Hint:

1. Task
2. List
3. Last

Provide the file name of the process that best demonstrates a runtime process enumeration event on the target host. \*

tasklist.exe

They want the *file name* of the process that shows:

- “runtime process enumeration”
- “process-list snapshots”
- “queries of running services”

And the hint:

1. Task
2. List
3. Last

This is pointing directly at:

tasklist.exe

```
//-----FLAG 8-----
DeviceProcessEvents
| where DeviceName == "gab-intern-vm"
| where AccountName == "gabint31intern"
| where ProcessCommandLine contains "tasklist"
| where TimeGenerated between (datetime(2025-10-01T00:00:00Z) .. datetime(2025-10-15T23:59:59Z))
project TimeGenerated, AccountName, ActionType, DeviceName, FileName, ProcessCommandLine, InitiatingProcessFileName, InitiatingProcessUniqueId, InitiatingProcessId, InitiatingProcessParentId
```

TimeGenerated [UTC]	AccountName	ActionType	DeviceName	FileName	ProcessCommandLine	InitiatingProcessFileName	InitiatingProcessUniqueid	InitiatingProcessId	InitiatingProcessParentId
> 10/9/2025, 12:51:57.639 PM	gabrc3intern	ProcessCreated	gab-intern-vm	cmd.exe	"cmd.exe" /c tasklist /v	powershell.exe	2533274790397065	8824	6844
> 10/9/2025, 12:51:57.686 PM	gabrc3intern	ProcessCreated	gab-intern-vm	tasklist.exe	tasklist /v	cmd.exe	2533274790397090	8412	8824

## Flag 9 – Privilege Surface Check

### Objective:

Detect attempts to understand privileges available to the current actor.

### What to Hunt:

Telemetry that reflects queries of group membership, token properties, or privilege listings.

### Thought:

Privilege mapping informs whether the actor proceeds as a user or seeks elevation.

### Hint:

1. Who

Identify the timestamp of the very first attempt \*

2025-10-09T12:52:14.3135459Z

## Objective

Detect attempts to understand privileges available to the current actor.

This means: **we're hunting for commands that ask "who am I?" or "what privileges do I have?"**

## What to Hunt

Queries of group membership, token properties, or privilege listings.

That's `whoami` territory.

## Hint:

1. Who

Identify the timestamp of the very first attempt.

So we're *not* giving a filename.

We're giving **the timestamp of the earliest privilege-checking event.**

2025-10-09T12:52:14.3135459Z

```
//-----FLAG 9-----
DeviceProcessEvents
| where DeviceName == "gab-intern-vm"
| where AccountName == "g4bri3lintern"
| where ProcessCommandLine contains "who"
| where TimeGenerated between (datetime(2025-10-01T00:00:00Z) .. datetime(2025-10-15T23:59:59Z))
| project TimeGenerated, AccountName, ActionType, DeviceName, FileName, ProcessCommandLine, InitiatingProcessFileName, InitiatingProcessUniqueId, InitiatingProcessId, InitiatingProcessParentId
```

>	10/9/2025, 12:52:14.313 PM	g4bri3lintern	ProcessCreated	gab-intern-vm	cmd.exe	"cmd.exe" /c whoami /groups
>	10/9/2025, 12:52:14.364 PM	g4bri3lintern	ProcessCreated	gab-intern-vm	whoami.exe	whoami /groups
>	10/9/2025, 12:52:15.322 PM	g4bri3lintern	ProcessCreated	gab-intern-vm	cmd.exe	"cmd.exe" /c whoami /priv
>	10/9/2025, 12:52:15.339 PM	g4bri3lintern	ProcessCreated	gab-intern-vm	whoami.exe	whoami /priv
>	10/9/2025, 12:54:53.547 PM	g4bri3lintern	ProcessCreated	gab-intern-vm	cmd.exe	"cmd.exe" /c whoami /groups
>	10/9/2025, 12:54:53.559 PM	g4bri3lintern	ProcessCreated	gab-intern-vm	whoami.exe	whoami /groups
>	10/9/2025, 12:54:54.559 PM	g4bri3lintern	ProcessCreated	gab-intern-vm	cmd.exe	"cmd.exe" /c whoami /priv
>	10/9/2025, 12:54:54.575 PM	g4bri3lintern	ProcessCreated	gab-intern-vm	whoami.exe	whoami /priv

## Flag 10 – Proof-of-Access & Egress Validation

### Objective:

Find actions that both validate outbound reachability and attempt to capture host state for exfiltration value.

### What to Hunt:

Look for combined evidence of outbound network checks and artifacts created as proof the actor can view or collect host data.

### Thought:

This step demonstrates both access and the potential to move meaningful data off the host...

### Side Note: 1/3

1. support

Which outbound destination was contacted first? ★

[www.msftconnecttest.com](http://www.msftconnecttest.com)

Outbound Contact = Anything the host reaches OUT to

In other words:

- DNS lookups
- HTTP(S) requests
- TCP/IP connections to external hosts
- Ping / ICMP echo requests
- Anything that leaves the VM and touches the internet or another host

Defender logs this as DeviceNetworkEvents.

```
//-----FLAG 10-----
DeviceNetworkEvents
| where DeviceName == "gab-intern-vm"
| where InitiatingProcessAccountName == "g4br3lintern"
| where InitiatingProcessFileName == "powershell.exe"
| where TimeGenerated between (datetime(2025-10-01T00:00:00Z) .. datetime(2025-10-15T23:59:59Z))
| project TimeGenerated, ActionType, DeviceName, InitiatingProcessAccountName, InitiatingProcessCommandLine, InitiatingProcessFileName, RemoteIP, RemoteUrl, InitiatingProcessFolderPath, InitiatingProcessUniqueId
| order by TimeGenerated asc
```

TimeGenerated [UTC]	ActionType	DeviceName	InitiatingProcessAccountNa...	InitiatingProcessCommandL...	InitiatingProcessFileName	RemoteIP	RemoteUrl
> 10/9/2025, 12:55:05.765 PM	ConnectionSuccess	gab-intern-vm	g4br3lintern	"powershell.exe"	powershell.exe	23.218.218.182	www.msftconnecttest.com
> 10/9/2025, 1:00:39.393 PM	ConnectionSuccess	gab-intern-vm	g4br3lintern	"powershell.exe"	powershell.exe	23.192.228.80	example.com
> 10/9/2025, 1:00:40.045 PM	ConnectionSuccess	gab-intern-vm	g4br3lintern	"powershell.exe"	powershell.exe	100.29.147.161	httpbin.org

Flag 11 – Bundling / Staging Artifacts

**Objective:**  
Detect consolidation of artifacts into a single location or package for transfer.

**What to Hunt:**  
File system events or operations that show grouping, consolidation, or packaging of gathered items.

**Thought:**  
Staging is the practical step that simplifies exfiltration and should be correlated back to prior recon.

**Hint:**  
1. Include the file value

Provide the full folder path value where the artifact was first dropped into \*

C:\Users\Public\ReconArtifacts.zip

Dropped at:

C:\Users\Public\ReconArtifacts.zip

And the logs confirm it perfectly:

- First created → **12:58:17.436 PM**, in *Public*
- Then copied or moved → *Documents*
- But they specifically ask for "first dropped", meaning the public directory.

Exactly the kind of staging behavior attackers love:

- Public is world-writable
- No elevation required
- No user desktop pop-ups
- Easy to exfiltrate quietly

```
//-----FLAG 11-----
DeviceFileEvents
| where DeviceName == "gab-intern-vm"
| where FolderPath contains "artifact"
| where TimeGenerated between (datetime(2025-10-01T00:00:00Z) .. datetime(2025-10-15T23:59:59Z))
| project TimeGenerated, ActionType, DeviceName, FileName, FolderPath
| order by TimeGenerated asc
```

TimeGenerated [UTC]	ActionType	DeviceName	FileName	FolderPath
> 10/9/2025, 12:34:59.126 PM	FileCreated	gab-intern-vm	DefenderTamperArtifactLnk	C:\Users\g4bri3lntern\AppData\Roaming\Microsoft\Windows\Recent\DefenderTamperArtifact.Lnk
> 10/9/2025, 12:58:17.436 PM	FileCreated	gab-intern-vm	ReconArtifacts.zip	C:\Users\Public\ReconArtifacts.zip
> 10/9/2025, 12:59:05.680 PM	FileCreated	gab-intern-vm	ReconArtifacts.zip	C:\Users\g4bri3lntern\Documents\ReconArtifacts.zip

## Flag 12 – Outbound Transfer Attempt (Simulated)

### Objective:

Identify attempts to move data off-host or test upload capability.

### What to Hunt:

Network events or process activity indicating outbound transfers or upload attempts, even if they fail.

### Thought:

Succeeded or not, attempt is still proof of intent — and it reveals egress paths or block points.

### Side Note: 2/3

1. chat

Provide the IP of the last unusual outbound connection \*

100.29.147.161

```
//-----FLAG 12-----
DeviceNetworkEvents
| where DeviceName == "gab-intern-vm"
| where InitiatingProcessAccountName == "g4br3lintern"
| where InitiatingProcessFileName == "powershell.exe"
| where TimeGenerated between (datetime(2025-10-01T00:00:00Z) .. datetime(2025-10-15T23:59:59Z))
| project TimeGenerated, ActionType, DeviceName, InitiatingProcessAccountName, InitiatingProcessCommandLine, InitiatingProcessFileName, RemoteIP, RemoteUrl
| order by TimeGenerated asc
```

Recall the same query from Flag 10. The IP of the last unusual outbound connection was listed to a website called `httpbin.org` .

The `RemoteIP` column showed the IP, `100.29.147.161` , of the outbound connection

TimeGenerated [UTC]	ActionType	DeviceName	InitiatingProcessAccountNa...	InitiatingProcessCommandL...	InitiatingProcessFileName	RemoteIP	RemoteUrl
> 10/9/2025, 12:55:05.765 PM	ConnectionSuccess	gab-intern-vm	g4br3lintern	"powershell.exe"	powershell.exe	23.218.218.182	www.msftconnecttest.com
> 10/9/2025, 1:00:39.393 PM	ConnectionSuccess	gab-intern-vm	g4br3lintern	"powershell.exe"	powershell.exe	23.192.228.80	example.com
> 10/9/2025, 1:00:40.045 PM	ConnectionSuccess	gab-intern-vm	g4br3lintern	"powershell.exe"	powershell.exe	100.29.147.161	httpbin.org

## Flag 13 – Scheduled Re-Execution Persistence

### Objective:

Detect creation of mechanisms that ensure the actor's tooling runs again on reuse or sign-in.

### What to Hunt:

Process or scheduler-related events that create recurring or logon-triggered executions tied to the same actor pattern.

### Thought:

Re-execution mechanisms are the actor's way of surviving beyond a single session — interrupting them reduces risk.

Provide the value of the task name down below \*

SupportToolUpdater

The question asks for task name

```
//-----FLAG 13-----  
DeviceProcessEvents  
| where DeviceName == "gab-intern-vm"  
| where AccountName == "g4bri3lintern"  
| where InitiatingProcessUniqueid contains "2533274790397065"  
| where TimeGenerated between (datetime(2025-10-09T00:00:00Z) .. datetime(2025-10-10T23:59:59Z))  
| project TimeGenerated, AccountName, ActionType, DeviceName, FileName, ProcessCommandLine, InitiatingProcessFileName, InitiatingProcessUniqueid, InitiatingProcessId, InitiatingProcessParentId  
| order by TimeGenerated asc
```

TimeGenerated [UTC]	2025-10-09T13:01:28.7700443Z
AccountName	g4bri3lintern
ActionType	ProcessCreated
DeviceName	gab-intern-vm
FileName	schtasks.exe
ProcessCommandLine	"schtasks.exe" /Create /SC ONLOGON /TN SupportToolUpdater /TR "powershell.exe -NoProfile -ExecutionPolicy Bypass -WindowStyle Hidden -File "C:\Users\g4bri3lintern\Downloads\SupportTool.ps1"" /RL LIMITED /F
InitiatingProcessFileName	powershell.exe
InitiatingProcessUniqueid	2533274790397065
InitiatingProcessId	8824
InitiatingProcessParentId	6844

We can see in the output of "schtasks.exe" that the task name /TN flag is part of the process command line. We can see the value of the task name is SupportToolUpdater



## Flag 14 – Autorun Fallback Persistence

### Objective:

Spot lightweight autorun entries placed as backup persistence in user scope.

### What to Hunt:

Registry or startup-area modifications that reference familiar execution patterns or repeat previously observed commands.

### Thought:

Redundant persistence increases resilience; find the fallback to prevent easy re-entry.

### Side Note: 3/3

1. log



If table returned nothing: **RemoteAssistUpdater**

DM the CTF admin should you wish to see how it would normally look like

What was the name of the registry value \*

RemoteAssistUpdater

The table RemoteAssistUpdater returned nothing.

---

## Flag 15 – Planted Narrative / Cover Artifact

### Objective:

Identify a narrative or explanatory artifact intended to justify the activity.

### What to Hunt:

Creation of explanatory files or user-facing artifacts near the time of suspicious operations; focus on timing and correlation rather than contents.

### Thought:

A planted explanation is a classic misdirection. The sequence and context reveal deception more than the text itself.

### Hint:

1. The actor opened it for some reason

Identify the file name of the artifact left behind \*

SupportChat\_log.lnk

The actor **left a cover story behind**, and the hint gives it away:

**Hint:** The actor opened it for some reason.
















That means we're hunting for a file the attacker **manually opened**, likely something meant to *explain* or *justify* what they were doing.

The attacker delivered `SupportTool.ps1` to the victim's Downloads folder and then executed it via the Windows shell, causing Explorer to create `SupportTool.lnk` in the Recent items directory. This ties the script to an interactive session (likely the `g4bri3Intern` profile) and demonstrates user-level execution (MITRE ATT&CK T1204 – User Execution).

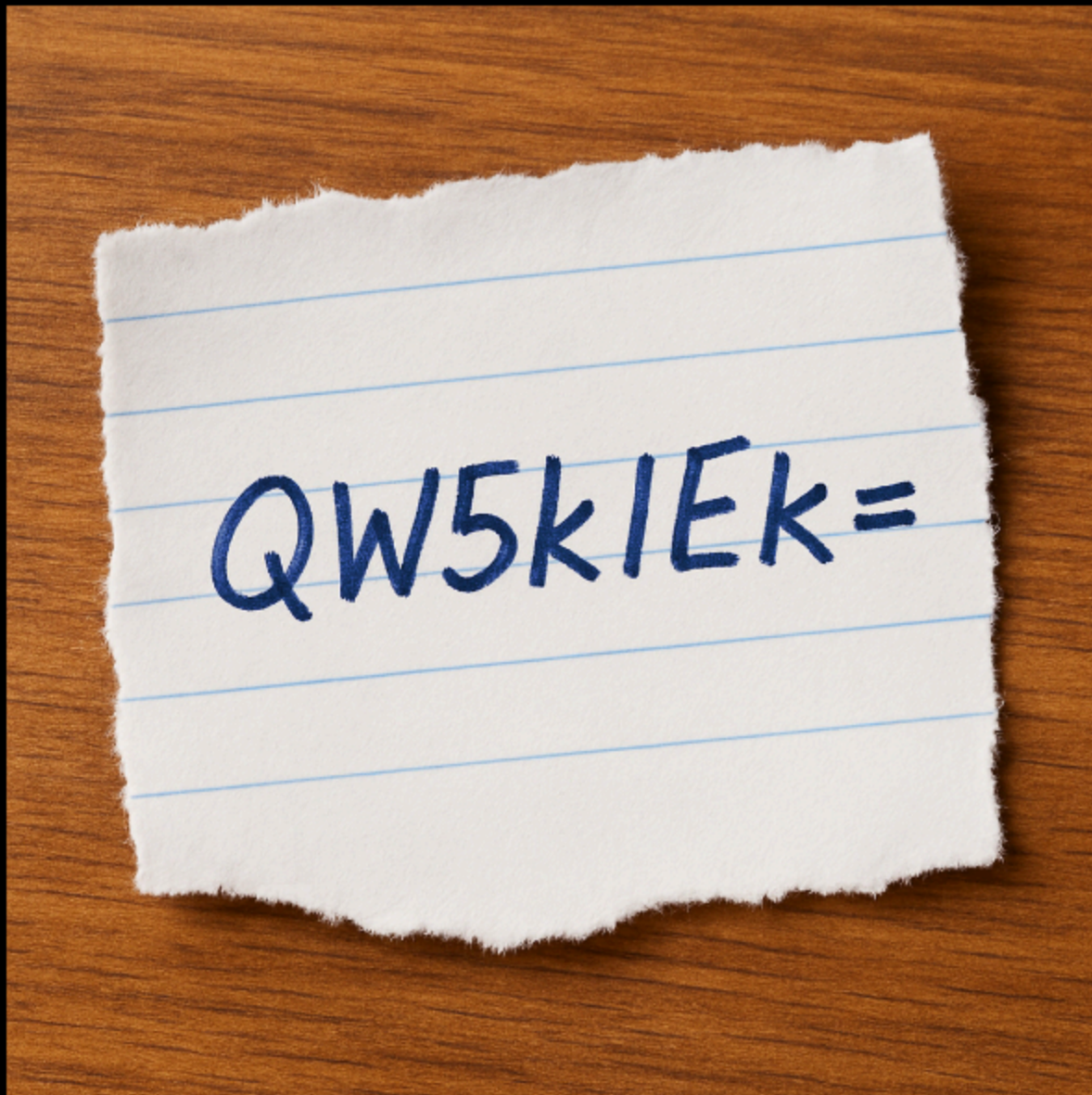
```
//-----FLAG 15-----
DeviceFileEvents
| where DeviceName == "gab-intern-vm"
| where FileName contains "Support"
| where TimeGenerated between (datetime(2025-10-01T00:00:00Z) .. datetime(2025-10-15T23:59:59Z))
| project TimeGenerated, ActionType, DeviceName, FileName, FolderPath, InitiatingProcessCommandLine, InitiatingProcessFileName, InitiatingProcessFolderPath
| order by TimeGenerated asc
```

TimeGenerated [UTC]	ActionType	DeviceName	FileName
> 10/7/2025, 5:46:27.775 AM	FileCreated	gab-intern-vm	api-ms-win-core-rtlsupport-l1-...
> 10/7/2025, 5:46:30.934 AM	FileCreated	gab-intern-vm	Qt5PrintSupport.dll
> 10/8/2025, 3:00:16.488 AM	FileRenamed	gab-intern-vm	servicemonikersupport.dll
> 10/8/2025, 3:00:16.491 AM	FileRenamed	gab-intern-vm	servicemonikersupport.dll
> 10/8/2025, 3:00:54.319 AM	FileRenamed	gab-intern-vm	ServiceMonikerSupport.dll
> 10/8/2025, 3:00:54.320 AM	FileRenamed	gab-intern-vm	ServiceMonikerSupport.dll
> 10/8/2025, 8:00:58.120 AM	FileCreated	gab-intern-vm	ServiceMonikerSupport.dll
> 10/8/2025, 8:01:00.390 AM	FileCreated	gab-intern-vm	ServiceMonikerSupport.dll
> 10/9/2025, 11:57:43.798 AM	FileCreated	gab-intern-vm	api-ms-win-core-rtlsupport-l1-...
> 10/9/2025, 11:58:02.607 AM	FileCreated	gab-intern-vm	Qt5PrintSupport.dll
> 10/9/2025, 12:04:57.108 PM	FileCreated	gab-intern-vm	api-ms-win-core-rtlsupport-l1-...
> 10/9/2025, 12:05:38.728 PM	FileCreated	gab-intern-vm	Support_701.txt
> 10/9/2025, 12:22:27.651 PM	FileCreated	gab-intern-vm	SupportTool.ps1
> 10/9/2025, 12:22:56.670 PM	FileCreated	gab-intern-vm	SupportTool.lnk
> 10/9/2025, 12:58:16.801 PM	FileCreated	gab-intern-vm	SupportTool.ps1
> 10/9/2025, 1:02:41.569 PM	FileCreated	gab-intern-vm	SupportChat_log.lnk
> 10/9/2025, 1:03:11.516 PM	FileCreated	gab-intern-vm	SupportChat_log.txt
> 10/9/2025, 1:03:20.122 PM	FileModified	gab-intern-vm	SupportChat_log.txt
> 10/9/2025, 1:03:20.682 PM	FileModified	gab-intern-vm	SupportChat_log.txt

## Logical Flow & Analyst Reasoning

- 0 → 1**  : An unfamiliar script surfaced in the user's Downloads directory. Was this *SupportTool.ps1* executed under the guise of IT diagnostics?
- 1 → 2**  : Initial execution often precedes an attempt to weaken defenses. Did the operator attempt to tamper with security tools to reduce visibility?
- 2 → 3**  : With protections probed, the next step is quick data checks. Did they sample clipboard contents to see if sensitive material was immediately available?
- 3 → 4**  : Attackers rarely stop with clipboard data. Did they expand into broader environmental reconnaissance to understand the host and user context?
- 4 → 5**  : Recon of the system itself is followed by scoping available storage. Did the attacker enumerate drives and shares to see where data might live?
- 5 → 6**  : After scoping storage, connectivity is key. Did they query network posture or DNS resolution to validate outbound capability?
- 6 → 7**  : Once network posture is confirmed, live session data becomes valuable. Did they check active users or sessions that could be hijacked or monitored?
- 7 → 8**  : Session checks alone aren't enough — attackers want a full picture of the runtime. Did they enumerate processes to understand active applications and defenses?
- 8 → 9**  : Process context often leads to privilege mapping. Did the operator query group memberships and privileges to understand access boundaries?
- 9 → 10**  : With host and identity context in hand, attackers often validate egress and capture evidence. Was there an outbound connectivity check coupled with a screenshot of the user's desktop?
- 10 → 11**  : After recon and evidence collection, staging comes next. Did the operator bundle key artifacts into a compressed archive for easy movement?
- 11 → 12**  : Staging rarely stops locally — exfiltration is tested soon after. Were outbound HTTP requests attempted to simulate upload of the bundle?
- 12 → 13**  : Exfil attempts imply intent to return. Did the operator establish persistence through scheduled tasks to ensure continued execution?
- 13 → 14**  : Attackers rarely trust a single persistence channel. Was a registry-based Run key added as a fallback mechanism to re-trigger the script?
- 14 → 15**  : Persistence secured, the final step is narrative control. Did the attacker drop a text log resembling a helpdesk chat to possibly justify these suspicious activities?

## Collect Your Cipher (4/5)



Finally done? \*

Yes