Log Files On Windows

Bensoula Khaoula CS-M1

🔐 1. Track Failed Login Attempts (Wrong Password)

```
Get-WinEvent -FilterHashtable @{LogName='Security'; Id=4625} |
Select-Object TimeCreated, Message |
Sort-Object TimeCreated -Descending
```

📘 Command Breakdown:

- `Get-WinEvent` : Retrieves events from event logs.
- `-FilterHashtable` : Lets you specify filters in a hash table format.
- `LogName='Security'` : We're looking into the Security log.
- `Id=4625` : This is the event ID for **failed login attempts** (wrong password).
- `Select-Object TimeCreated, Message` : Shows only the time and message of each event.
- `Sort-Object TimeCreated -Descending` : Sorts results from most recent to oldest.

✅ 2. Track Successful Logins + Admin Privileges

```
Get-WinEvent -FilterHashtable @{
  LogName = 'Security';
  Id = 4624, 4672;
  StartTime = (Get-Date).Date
} |
Select-Object TimeCreated, Id, Message |
Sort-Object TimeCreated -Descending
```

📘 Command Breakdown:

- `Id = 4624, 4672` :
  - `4624` → Successful login

- `4672` → User was granted admin privileges
- `StartTime = (Get-Date).Date`: Only show today's events.
- The rest is like before: filtering, selecting important fields, sorting.

🗒️ 3. List All Audit Categories Available

```
auditpol /list /subcategory:*
```

📘 Command Breakdown:

- `auditpol`: Built-in command to manage audit policy.
- `/list`: Show available audit categories.
- `/subcategory:*`: Show **all** detailed subcategories (e.g., Logon, File System, Process Creation...).

🔍 4. Check If Logging is Enabled for Process Creation

```
auditpol /get /subcategory:"Création du processus"
```

📘 Command Breakdown:

- `/get`: Query the current audit policy.
- `/subcategory:"Création du processus"`: Check logging status for **Process Creation** events (event ID 4688).
- Shows whether **Success** and/or **Failure** logging is enabled.

🔧 5. Enable Process Creation Logging

```
auditpol /set /subcategory:"Création du processus" /success:enable
/failure:enable
```

📘 Command Breakdown:

- `/set` : Change audit settings.
- `/success:enable` : Enable logging when a process **successfully** starts.
- `/failure:enable` : Enable logging if starting a process **fails**.

👀 6. View Created Processes (Who Ran What?)

```
Get-WinEvent -FilterHashtable @{LogName='Security'; Id=4688} |
ForEach-Object {
    $msg = $_.Message
    [PSCustomObject]@{
        Time = $_.TimeCreated
        NewProcess = ($msg -split "New Process Name:\s+")[1] -split "`r?`n"
| Select-Object -First 1
        Creator = ($msg -split "Creator Process Name:\s+")[1] -split "`r?`n"
| Select-Object -First 1
    }
}
```

📘 Command Breakdown:

- `Id=4688` : Event ID for every process created.
- `ForEach-Object` : Loop through each event to extract useful data.
- `$msg = $_.Message` : Save the event's message text.
- `-split "New Process Name:\s+"` :
  - Split the message to isolate the process name.
- `Select-Object -First 1` : Get the first line after splitting (the process path).
- The result: A table showing **when** the process ran, **what** was run, and **who** ran it.

🛡️ 7. Track Audit Policy Changes

Basic:

```
Get-WinEvent -FilterHashtable @{LogName='Security'; Id=4719} |
Select-Object TimeCreated, Message |
Sort-Object TimeCreated -Descending
```

Detailed:

```
Get-WinEvent -FilterHashtable @{LogName='Security'; Id=4719} |
Sort-Object TimeCreated -Descending |
Select-Object -First 5 |
ForEach-Object {
    $_.TimeCreated
    $_.Message
    ""
}
```

📘 Command Breakdown:

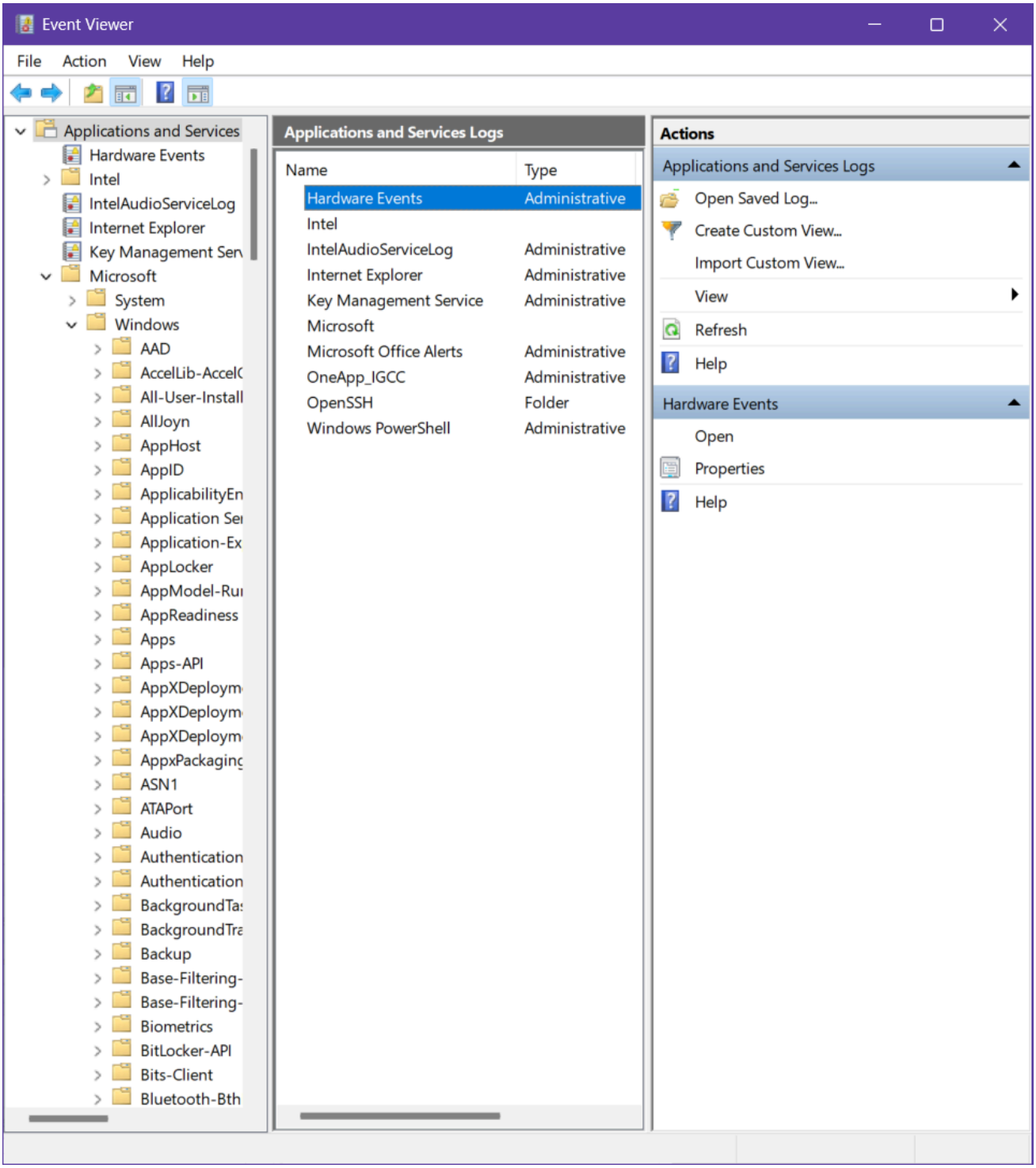
- `Id=4719`: Indicates a change in **audit policy** (security settings).
- `Select-Object -First 5`: Only show the **5 most recent** changes.
- The loop prints the time and message for each.

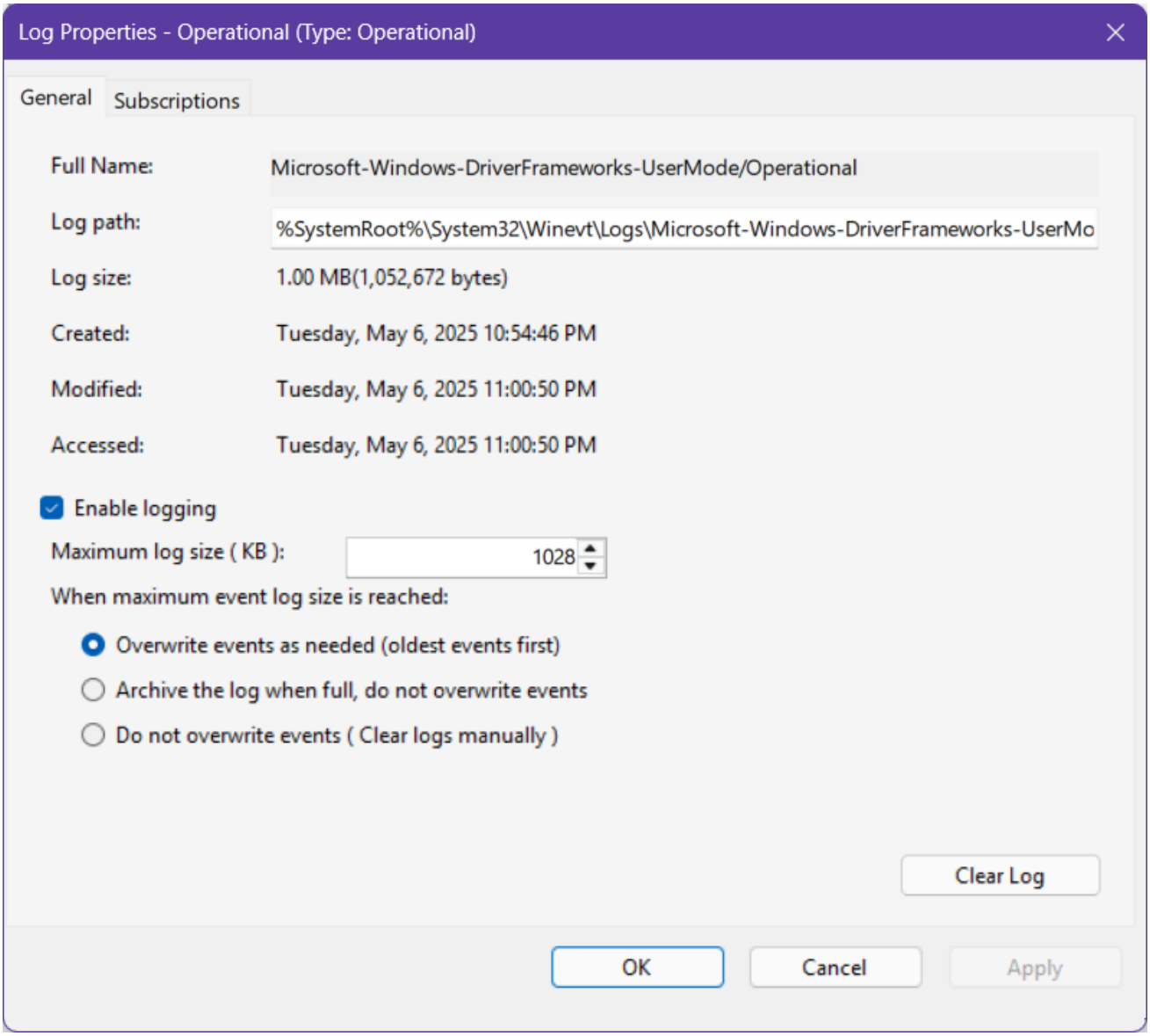
🔌 8. Enable USB Logging (Only if Not Active)

🧭 **Manual steps**:

1. Open **Event Viewer**.
2. Navigate to:
   `Applications and Services Logs > Microsoft > Windows > DriverFrameworks-UserMode > Operational`

3. Right-click **Operational** → Click **Enable Log**.





💾 9. Detect USB Plug-in and Unplug Events

```
$today = (Get-Date).Date

Get-WinEvent -FilterHashtable @{
  LogName   = "Microsoft-Windows-DriverFrameworks-UserMode/Operational"
  Id        = 2003, 2100, 2102
  StartTime = $today
} |
Select-Object TimeCreated, Id, Message |
Sort-Object TimeCreated -Descending
```

📘 Command Breakdown:

- `$today = (Get-Date).Date` : Get today's date (midnight).
- `LogName = ...Operational` : USB plug/unplug logs are saved here.
- `Id = 2003` : USB device plugged in.
- `Id = 2100, 2102` : PnP (plug and play) or power operations for USB.
- Filters and sorts logs to show USB activity for the current day.

REMEMBER :

What Is an Audit Log (Audit Trail)?

An audit log (or audit trail) is a chronological record of events that show:

```
Who performed an action (e.g., username, process)

What action was performed (e.g., login attempt, file access)

When the action occurred (timestamp)

Where it happened (machine name, IP address)

Whether it was successful or failed
```

Command line to know who is the user of the Security ID that you have found :

$objSID = New-ObjectSystem.Security.Principal.SecurityIdentifier("S-1-5-...")$
objSID.Translate([System.Security.Principal.NTAccount])