CHAPTER 8

# Event Log Analysis

# Chapter Overview

01

## Understanding Event Logs

Key log files, audit policy, and event structure

02

## Account-Related Events

Logon, logoff, account changes, and privilege use

03

## Object Access & Process Auditing

File access, process creation, and lateral movement artifacts

04

## PowerShell Logging

Module, script block, and transcription logging

# Understanding Event Logs

Windows audit logs as forensic evidence

# Key Log Files & Audit Policy

**Security.evtx**: logon/logoff, account management, object access (most important)

**System.evtx**: OS events, service start/stop, driver load

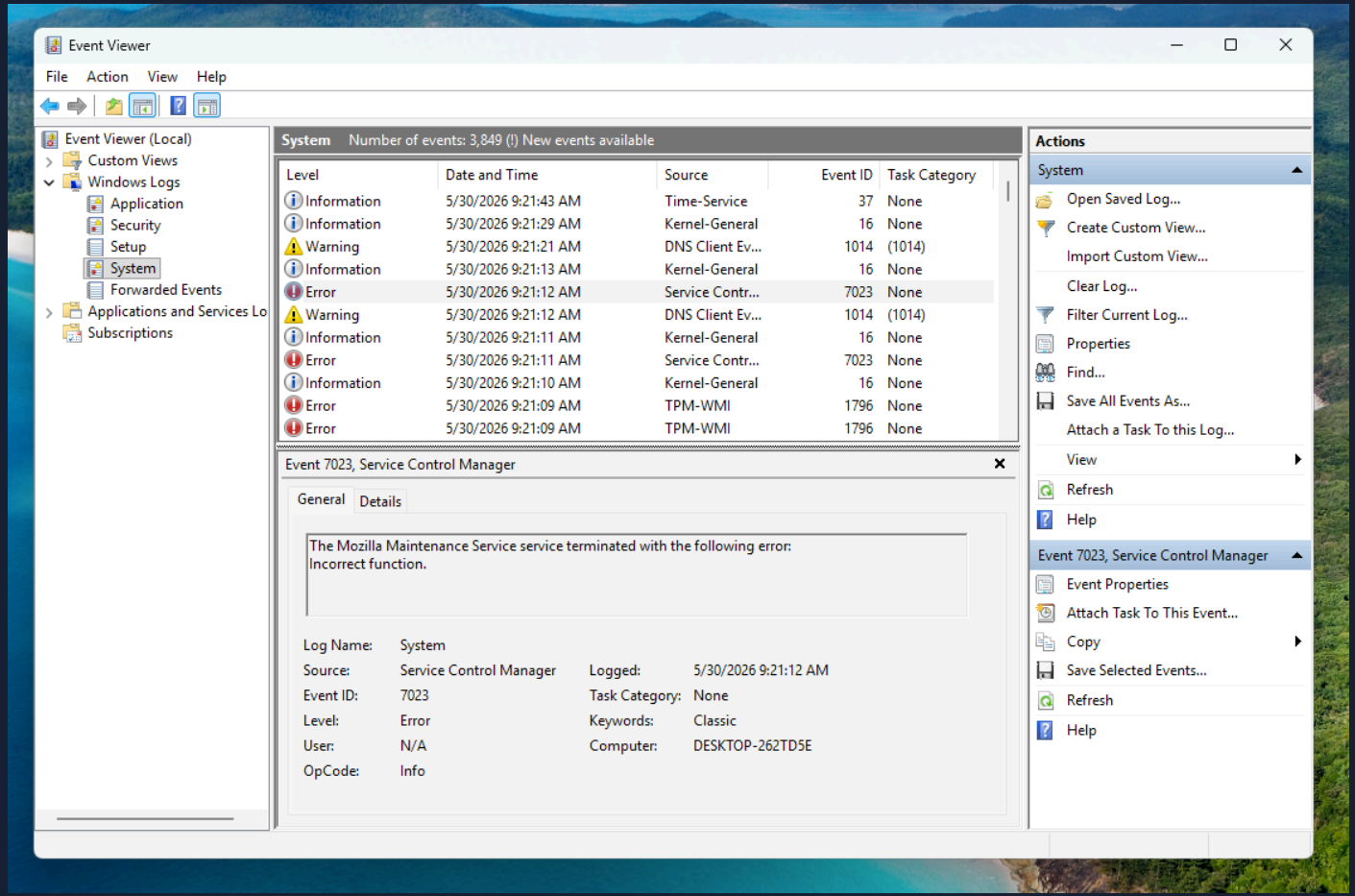
**Application.evtx**: application-generated events

**Microsoft-Windows-PowerShell/Operational.evtx**: PowerShell activity

**Microsoft-Windows-TaskScheduler/Operational.evtx**: scheduled task events

Default audit policy is insufficient — enable advanced audit policy via GPO

**Log clearing events**: 1102 (Security log cleared), 104 (System log cleared) — always investigate




*Windows Event Viewer*

# Enabling Advanced Audit Policy via GPO

**Default audit policy**: minimal — only basic success/failure for some categories

**Advanced Audit Policy**: granular control over 57 individual subcategories

**Path**: Computer Config → Windows Settings → Security Settings → Advanced Audit Policy

**Key subcategories to enable**: Logon, Logoff, Process Creation, Object Access, Privilege Use

**Auditpol.exe**: auditpol /get /category:* — view current policy settings

**Force with auditpol**: auditpol /set /subcategory:'Process Creation' /success:enable

**Command line logging (4688)**: enable via GPO — Computer Config → Admin Templates → System → Audit Process Creation

**Test deployment**: trigger a test logon and verify Event 4624 appears in Security log

# Sysmon — Enhanced Windows Logging

**Sysmon (System Monitor)**: Sysinternals tool adding 29+ event types to Windows logging

**Event ID 1**: Process creation with full command line, parent PID, and file hashes

**Event ID 3**: Network connections with process name, source, and destination

**Event ID 7**: Image (DLL) loaded with hash — detect DLL hijacking and injection

**Event ID 8**: CreateRemoteThread — detect cross-process code injection

**Event ID 10**: ProcessAccess — detect LSASS credential dumping attempts

**Event ID 11**: FileCreate — detect dropper activity and persistence files

**Configuration**: SwiftOnSecurity or Olaf Hartong sysmon-modular configs recommended

# Account-Related Events

Who logged in, when, and how

# Critical Logon Event IDs

## Logon Events

**4624**: Successful logon — always check LogonType

**4625**: Failed logon — brute force indicator

**4634 / 4647**: Logoff

**4648**: Logon with explicit credentials (runas, pass-the-hash)

**4768 / 4769**: Kerberos TGT / TGS requests

## Logon Types

**Type 2**: Interactive (keyboard at console)

**Type 3**: Network (SMB, mapped drives)

**Type 4**: Batch (scheduled tasks)

**Type 5**: Service

**Type 10**: Remote Interactive (RDP)

**Type 7**: Unlock workstation

# Account Changes & Privilege Use

**4720**: A user account was created

**4722**: A user account was enabled

**4728 / 4732**: Member added to global / local group (watch for Administrators)

**4756**: Member added to Universal Security Group

**4672**: Special privileges assigned to new logon (admin-equivalent rights)

**4688**: Process creation — CRITICAL — enable command line logging (KB3004375)

**4697 / 7045**: Service installed — common malware persistence mechanism

# Object Access & Process Tracking Events

**4663**: An attempt was made to access an object — file, registry key, or directory

**Requires**: SACL (System Access Control List) on the object — set per object via properties

**4656**: A handle to an object was requested — precedes 4663

**4688**: A new process was created — CRITICAL for tracking attacker tool execution

**4688 includes command line (if enabled)**: reveals full attacker command and arguments

**4689**: A process has exited — pair with 4688 to get execution duration

**5140**: A network share object was accessed — lateral movement via SMB

**5145**: A network share object was checked for access — detailed SMB audit

# Lateral Movement Event Signatures

**PsExec**: Event 7045 (service install) with ServiceName='PSEXESVC', then 4624 Type 3

**Pass-the-Hash**: 4624 Type 3 + NTLM AuthenticationPackage from unusual source IP

**DCSync**: 4662 — Replication rights (1131f6aa GUID) exercised by non-DC account

**Scheduled task creation**: 4698 (task created) on REMOTE system from attacker's IP

**WMI execution**: parent wmiprvse.exe spawning cmd.exe or powershell.exe (4688)

**RDP access**: 4778 (session reconnect) + 4779 (disconnect) timeline on target system

**Net use connection**: 5140 (share accessed) + 4624 Type 3 from workstation to server

**Kerberos anomalies**: 4769 RC4 encryption type, 4768 from unexpected source IP

# Windows Event Forwarding (WEF)

**WEF**: built-in Windows mechanism to forward events to a central Windows Event Collector (WEC)

**WEC**: typically a Windows Server collecting events from all endpoints via WS-Man

**GPO configuration**: Computer Config → Admin Templates → Windows Components → Event Forwarding

**Subscription**: defines WHICH events to forward — use XPath filters for efficiency

**Pull mode**: WEC pulls from endpoints (firewall-friendly for workstations)

**Push mode**: endpoints push to WEC (lower latency, good for servers)

**Forwarded logs appear in**: Windows Logs → Forwarded Events on the collector

**SIEM integration**: pull from central WEC instead of 10,000 individual endpoints

# PowerShell Logging

Capturing attacker script execution

# PowerShell Audit Logging

**Module Logging (4103)**: records which PowerShell modules were invoked

**Script Block Logging (4104)**: records exact content of executed script blocks

4104 captures obfuscated AND de-obfuscated code — attackers cannot hide this

**Transcription logging**: full session transcript saved to disk

**Enable via GPO**: Computer Config → Admin Templates → Windows Components → PowerShell

**AMSI (Antimalware Scan Interface)**: feeds script content to AV engines at runtime

**Combined**: SBL + AMSI = comprehensive visibility into PowerShell attacks

# EVTX Parsing Tools & Log Analysis Workflow

**EvtxECmd (Eric Zimmerman)**: fast command-line EVTX parser — CSV/JSON output

**Chainsaw**: SIGMA rule-based hunting across EVTX files — fast bulk triage

**Hayabusa**: timeline generator from EVTX — produces visual attack timeline

**LogParser**: SQL-like queries — SELECT * FROM Security WHERE EventID=4624

**Step 1 — Collect**: export EVTX files or query SIEM for target timeframe

**Step 2 — Filter**: start with 4624/4625/4688/7045 — highest-signal events

**Step 3 — Correlate**: link logon events to process creation to service install

**Step 4 — Timeline**: chronological sequence of attacker actions by timestamp

# Detecting Log Tampering & Clearing

**Event 1102**: The audit log was cleared (Security log) — always investigate

**Event 104**: The System log was cleared — equivalent for System.evtx

Event 104 is logged in new System.evtx even when the old one is cleared

**Log service stop**: Event 7035/7036 (Service Control Manager) showing EventLog stopped

**Gaps in log sequence**: Event IDs are sequential — missing range indicates deletion

**Time gaps**: hour-long gap in a normally busy Security log = log clearing or manipulation

**SIEM forwarding defense**: forwarded events cannot be deleted from collector after delivery

**Threat**: attackers with admin rights can wipe local logs — forward logs before they arrive

# Event Log Analysis at Scale — SIEM Queries

**SIEM purpose**: centralize, correlate, and search logs from thousands of systems simultaneously

**Splunk**: index=wineventlog EventCode=4624 LogonType=10 | stats count by src_ip, user

**Elastic KQL**: event.code:4688 AND process.command_line:*powershell* AND process.parent.name:word*

**Time correlation**: attacker activity often spans 10–30 minute windows — zoom in on those periods

**Anomaly detection**: ML-based baselines flag users logging in at abnormal times or from new hosts

**Alert tuning**: high-volume event IDs (4624, 4634) need filters to be useful — filter out normal

**Retention**: logs older than 90 days may be archived — plan ahead for long-dwell investigations

**SIGMA rules**: compile vendor-neutral detection rules to any SIEM query language automatically

# Conclusion

Event logs are the single richest source of attacker activity in Windows environments

Default audit policy is insufficient — deploy advanced audit policy via GPO

Logon type 3 (network) with NTLM authentication is a lateral movement indicator

Script block logging (4104) captures PowerShell attacks even through obfuscation

Sysmon extends Windows logging with process hashes, network connections, and injection detection

Centralize logs to a SIEM — local logs can be cleared by attackers with admin rights

# Knowledge Check