APPLIED INCIDENT RESPONSE

Analyst Reference

Windows Event Log Analysis

Version 20191223

Contents

Introduction 2
Event Log Format 3
Account Management Events 4
Account Logon and Logon Events 5
Access to Shared Objects 11
Scheduled Task Logging 12
Object Access Auditing 13
Audit Policy Changes 16
Auditing Windows Services 17
Wireless LAN Auditing 18
Process Tracking 19
Additional Program Execution Logging 21
Auditing PowerShell Use 24

Introduction

Microsoft has gradually increased the efficiency and effectiveness of its auditing facilities over the years. Modern Windows systems can log vast amounts of information with minimal system impact. With the corresponding decrease in the price of storage media, excuses to not enable and retain these critical pieces of evidence simply don't stand up to scrutiny. Configuring adequate logging on Windows systems, and ideally aggregating those logs into a SIEM or other log aggregator, is a critical step toward ensuring that your environment is able to support an effective incident response.

This document provides an overview of some of the most important Windows logs and the events that are recorded there. As with all of our Analyst Reference documents, this PDF is intended to provide more detail than a cheat sheet while still being short enough to serve as a quick reference. The PDF also contains links to external resources for further reference.

Event Log Format

Modern Windows systems store logs in the %SystemRoot%\System32\winevt\logs directory by default in the binary XML Windows Event Logging format, designated by the .evtx extension. Logs can also be stored remotely using log subscriptions. For remote logging, a remote system running the Windows Event Collector service subscribes to subscriptions of logs produced by other systems. The types of logs to be collected can be specified at a granular level and transport occurs over HTTPS on port 5986 using WinRM. GPO's can be used to configure the remote logging facilities on each computer.

Events can be logged in the Security, System and Application event logs or, on modern Windows systems, they may also appear in several other log files. The Setup event log records activities that occurred during installation of Windows. The Forwarded Logs event log is the default location to record events received from other systems. But there are also many additional logs, listed under Applications and Services Logs in Event Viewer, that record details related to specific types of activities. Since these log files are much more targeted than the Security log, they often retain information about events that occurred well before the current Security log has been overwritten. Always look for multiple sources of log information, and don't forget to look for older log files that may be captured by backup systems or volume shadow copies.

Event IDs have several fields in common:

- Log Name: The name of the Event Log where the event is stored. Useful when processing numerous logs pulled from the same system.
- Source: The service, Microsoft component or application that generated the event.
- Event ID: A code assigned to each type of audited activity.
- Level: The severity assigned to the event in question.
- User: The user account involved in triggering the activity or the user context that the source was running as when it logged the event. Note that this field often indicates "System" or a user that is not the cause of the event being recorded.
- OpCode: Assigned by the source generating the log. It's meaning is left to the source.
- Logged: The local system date and time when the event was logged.
- Task Category: Assigned by the source generating the log. It's meaning is left to the source.
- Keywords: Assigned by the source and used to group or sort events.
- Computer: The computer on which the event was logged. This is useful when examining logs collected from multiple systems, but should not be considered to be the device that caused an event (such as when a remote logon is initiated, the Computer field will still show the name of the system logging the event, not the source of the connection).
- Description: A text block where additional information specific to the event being logged is recorded. This is often the most significant field for the analyst.

Account Management Events

The following events will be recorded on the system where the account was created or modified, which will be the local system for a local account or a domain controller for a domain account.

Event ID	Description
4720	A user account was created.
4722	A user account was enabled.
4723	A user attempted to change an account's password.
4724	An attempt was made to reset an account's password.
4725	A user account was disabled.
4726	A user account was deleted.
4727	A security-enabled global group was created.
4728	A member was added to a security-enabled global group.
4729	A member was removed from a security-enabled global group.
4730	A security-enabled global group was deleted.
4731	A security-enabled local group was created.
4732	A member was added to a security-enabled local group.
4733	A member was removed from a security-enabled local group.
4734	A security-enabled local group was deleted.
4735	A security-enabled local group was changed.
4737	A security-enabled global group was changed.
4738	A user account was changed.
4741	A computer account was created.
4742	A computer account was changed.
4743	A computer account was deleted.
4754	A security-enabled universal group was created.
4755	A security-enabled universal group was changed.
4756	A member was added to a security-enabled universal group.
4757	A member was removed from a security-enabled universal group.
4758	A security-enabled universal group was deleted.
4798	A user's local group membership was enumerated. Large numbers of these events may be indicative of adversary account enumeration.
4799	A security-enabled local group membership was enumerated. Large numbers of these events may be indicative of adversary group enumeration.

Account Logon and Logon Events

Account Logon is the Microsoft term for authentication. Logon is the term used to refer to an account gaining access to a resource. Both Account Logon and Logon events will be recorded in the Security event log. Authentication (account logon) of domain accounts is performed by a domain controller within a Windows network. Local accounts (those that exist within a local SAM file rather than as a part of Active Directory) are authenticated by the local system where they exist. Account logon events will be logged by the system that performs the authentication. Auditing of Account Logon and Logon events is easily set by Group Policy. While Microsoft continues to enable more logging by default as new versions of Windows are released, administrators should review their audit policies on a regular basis to ensure that all systems are generating adequate logs. The ability to store event logs on remote systems (either using the native Microsoft remote logging features or third-party SIEM or other tools) helps safeguard logs from alteration or destruction.

The domain controllers in your network should therefore be able to provide a fairly centralized accounting of which accounts where authenticated throughout the domain. Remember that to get a full picture, you will need to query each of your DCs since the one that performs the authentication creates the associated event log. On the other hand, if you find that member servers or workstations are performing their own authentication, that is a good indicator that local user accounts are being used. As this is not normally done in most environments, account logon events on non-domain controllers can often be an indicator of compromise. By contrast, logon event logs are generated by the system that is being accessed, so logon events will be generated by systems across the network, providing another reason to aggregate logs to a central location.

Event IDs of particular interest on domain controllers, which authenticate domain users, include:

Event ID	Description
4768	The successful issuance of a TGT shows that a user account was authenticated by the domain controller. The Network Information section of the event description contains additional information about the remote host in the event of a remote logon attempt. The Keywords field indicates whether the authentication attempt was successful or failed. In the event of a failed authentication attempt, the result code in the event description provides additional information about the reason for the failure, as specified in RFC 4120. Some of the more commonly encountered codes are:

Common Event ID 4768 result codes

Decimal	Hex	Meaning
6	0x6	Username not valid.
12	0xC	Policy restriction prohibiting this logon (such as a workstation restriction or time-of-day restriction).
18	0x12	The account is locked out, disabled, or expired.
23	0x17	The account's password is expired.
24	0x18	The password is incorrect.
32	0x20	The ticket has expired (common on computer accounts).
37	0x25	The clock skew is too great.

Source: https://docs.microsoft.com/en-us/windows/security/threat-protection/auditing/event-4768

Event ID	Description
4769	A service ticket was requested by a user account for a specified resource. This event description shows the source IP of the system that made the request, the user account used, and the service to be accessed. These events provide a useful source of evidence as they track authenticated user access across the network. The Keywords field indicates whether the request for the service ticket was successful or failed. In the case of a failure, the result code indicates the reason for the failure. The ticket encryption type is also recorded, which might be useful for detecting attacks against Kerberos.
4770	A service ticket was renewed. The account name, service name, client IP address, and encryption type are recorded.
4771	Depending on the reason for a failed Kerberos logon, either Event ID 4768 or Event ID 4771 is created. In either case, the result code in the event description provides additional information about the reason for the failure.
4776	This event ID is recorded for NTLM authentication attempts. The Network Information section of the event description contains additional information about the remote host in the event of a remote logon attempt. The Keywords field indicates whether the authentication attempt succeeded or failed. In the event of authentication failure, the error code in the event description provides additional details about the failure, as described in Table 8.3. A series of failed 4776 events with Error Code C000006A (the password is invalid) followed by an Error Code C0000234 (the account is locked out) may be indicative of a failed password guessing attack (or a user who has simply forgotten the account password). Similarly, a series of failed 4776 events followed by a successful 4776 event may show a successful password guessing attack. The presence of Event ID 4776 on a member server or client is indicative of a user attempting to authenticate to a local account on that system and may in and of itself be cause for further investigation.

Common Event ID 4776 error code descriptions

Error Code	Meaning
0xC0000064	The username is incorrect.
0xC000006A	The password is incorrect.
0xC000006D	Generic logon failure. Possibly bad username or password or mismatch in the LAN Manager Authentication Level between the source and target computers.
0xC000006F	Account logon outside authorized hours.
0xC0000070	Account logon from unauthorized workstation.
0xC0000071	Account logon with expired password.
0xC0000072	Account logon to account disabled by administrator.
0xC0000193	Account logon with expired account.
0xC0000224	Account logon with Change Password At Next Logon flagged.
0xC0000234	Account logon with account locked.
0xc0000371	The local account store does not contain secret material for the specified account.

Source: https://docs.microsoft.com/en-us/windows/security/threat-protection/auditing/event-4776

On systems being accessed, Event IDs of note include:

Event ID	Description
4624	A logon to a system has occurred. Type 2 indicates an interactive (usually local) logon, whereas a Type 3 indicates a remote or network logon. The event description will contain information about the host and account name involved. For remote logons, focus on the Network Information section of the event description for remote host information. Correlation with the associated 4768, 4769, or 4776 events may yield additional details about a remote host. Discrepancies in the record entry between the recorded hostname and its assigned IP address may be indicative of Server Message Block (SMB) relay attacks, where an attacker relays a request from one system using an IP address not associated with that system. The Caller Process Name and Caller Process ID fields in the Process Information section of the event description can provide additional details about the process initiating the logon. Successful Remote Desktop Protocol (RDP) connections usually log as Logon Type 10 in Event ID 4624. This records a successful remote interactive logon and may result in the user's credentials being cached in RAM and possibly on disk. Use of Restricted Admin mode may impact this. Failed RDP logons usually result in Logon Type 3.

Logon events contain a Type code in the event description:

Logon events contain a Type code in the event description:

Logon event type code descriptions

Logon Type	Description
2	Interactive, such as logon at keyboard and screen of the system, or remotely using third-party remote access tools like VNC, or psexec with the -u switch. Logons of this type will cache the user's credentials in RAM for the duration of the session and may cache the user's credentials on disk.
3	Network, such as access to a shared folder on this computer from elsewhere on the network. This represents a noninteractive logon, which does not cache the user's credentials in RAM or on disk.
4	Batch (indicating a scheduled task). Batch logon type is used by batch servers, where processes may be executing on behalf of a user without their direct intervention.
5	Service indicates that a service was started by the Service Control Manager.
7	Unlock indicates that an unattended workstation with a password protected screen is unlocked
8	NetworkCleartext indicates that a user logged on to this computer from the network and the user's password was passed to the authentication package in its unhashed form. The built-in authentication packages all hash credentials before sending them across the network. The credentials do not traverse the network in plaintext (also called cleartext). Most often indicates a logon to Internet Information Services (IIS) with basic authentication.
9	NewCredentials indicates that a user logged on with alternate credentials to perform actions such as with RunAs or mapping a network drive. If you want to track users attempting to log on with alternate credentials, also look for Event ID 4648.
10	RemoteInteractive indicates that Terminal Services, Remote Desktop, or Remote Assistance for an interactive logon. See the note on RDP at the end of this section for more details.
11	CachedInteractive (logon with cached domain credentials such as when logging on to a laptop when away from the network). The domain controller was not contacted to verify the credential, so no account logon entry is generated.

Table includes details from:
www.ultimatewindowssecurity.com/securitylog/encyclopedia/event.aspx?eventid=4624 and https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2003/cc787567(v=ws.10).

Event ID	Description
4625	A failed logon attempt. Large numbers of these throughout a network may be indicative of password guessing or password spraying attacks. Again, the Network Information section of the event description can provide valuable information about a remote host attempting to log on to the system. Note that failed logons over RDP may log as Type 3 rather than Type 10, depending on the systems involved. You can determine more about the reason for the failure by consulting the Failure Information section of the event description.

The status code found in Event ID 4625 provides additional details about the event:

Common logon failure status codes

Status code	Description
0XC000005E	Currently no logon servers are available to service the logon request.
0xC0000064	User logon with misspelled or bad user account.
0xC000006A	User logon with misspelled or bad password.
0XC000006D	This is either due to a bad username or incorrect authentication information.
0XC000006E	Unknown username or bad password.
0xC000006F	User logon outside authorized hours.
0xC0000070	User logon from unauthorized workstation.
0xC0000071	User logon with expired password.
0xC0000072	User logon to account disabled by administrator.
0XC00000DC	Indicates the Server was in the wrong state to perform the desired operation.
0XC0000133	Clocks between domain controller and other computer too far out of sync.
0XC000015B	The user has not been granted the requested logon type (also known as logon right) at this machine.
0XC000018C	The logon request failed because the trust relationship between the primary domain and the trusted domain failed.
0XC0000192	An attempt was made to log on, but the Netlogon service was not started.
0xC0000193	User logon with expired account.
0XC0000224	User is required to change password at next logon.
0XC0000225	Evidently a bug in Windows and not a risk.
0xC0000234	User logon with account locked.
0XC00002EE	Failure Reason: An error occurred during logon.
0XC0000413	Logon Failure: The machine you are logging on to is protected by an authentication firewall. The specified account is not allowed to authenticate to the machine.

Source: https://docs.microsoft.com/en-us/windows/security/threat-protection/auditing/event-4625

Event ID	Description
4634/4647	User logoff is recorded by Event ID 4634 or Event ID 4647. The lack of an event showing a logoff should not be considered overly suspicious, as Windows is inconsistent in logging Event ID 4634 in many cases. The Logon ID field can be used to tie the Event ID 4624 logon event with the associated logoff event (the Logon ID is unique between reboots on the same computer). Type 3 (Network) logons will typically disconnect shortly after a request is complete and do not indicate the actual amount of time that a user was engaged in any particular activity. Interactive logons (primarily type 2, but also types 10 and 11 where they exist) can provide a better sense of session duration, but Windows is not overly consistent in logging Event ID 4634 and may disconnect sessions due to inactivity well after a user stopped actively interacting with a session.
4648	A logon was attempted using explicit credentials. When a user attempts to use credentials other than the ones used for the current logon session (including bypassing User Account Control [UAC] to open a process with administrator permissions), this event is logged.
4672	This event ID is recorded when certain privileges associated with elevated or administrator access are granted to a logon. As with all logon events, the event log will be generated by the system being accessed.
4778	This event is logged when a session is reconnected to a Windows station. This can occur locally when the user context is switched via fast user switching. It can also occur when a session is reconnected over RDP. The initial connection over RDP is logged with Event ID 4624 as mentioned earlier. To differentiate between RDP versus local session switching, look at the Session Name field within the event description. If local, the field will contain Console, and if remote, it will begin with RDP. For RDP sessions, the remote host information will be in the Network Information section of the event description.
4779	This event is logged when a session is disconnected. This can occur locally when the user context is switched via fast user switching. It can also occur when a session is reconnected over RDP. A full logoff from an RDP session is logged with Event ID 4637 or 4647 as mentioned earlier. To differentiate between RDP versus local session switching, look at the Session Name field within the event description. If local, the field will contain Console, and if remote, it will begin with RDP. For RDP sessions, the remote host information will be in the Network Information section of the event description.

Additional information about RDP Sessions can be found in the %SystemRoot%\System32\winevt\Logs\Microsoft-Windows-TerminalServices-LocalSessionManager%4Operational log file. Event ID 21 in this log shows session logon events, both local and remote, including the IP from which the connection was made if remote. Event ID 24 in this log shows session disconnection, including the IP from which the connection was made if remote. For local logons, the Source Network Address field of the event description will read LOCAL rather than provide the remote IP.

Information about RDP Sessions can also be found in the %SystemRoot%\System32\winevt\Logs\Microsoft-Windows-TerminalServices-RemoteConnectionManager%4Operational log file. Event ID 1149 in this log will show the user account and source IP used to initiate an RDP session.

Access to Shared Objects

Attackers frequently leverage valid credentials to remotely access data through user created or administrative shares. Doing so will generate Account Logon and Logon events as mentioned above, but additional logging can also be enabled in the Group Policy Management Console by navigating to Computer Configuration -> Policies -> Windows Settings -> Security Settings -> Advanced Audit Policy Configuration -> Audit Policies -> Object Access -> Audit File Share. Once enabled, the following Event IDs will be logged in the Security Log:

Network share event IDs

Event ID	Description
5140	A network share object was accessed. The event entry provides the account name and source address of the account that accessed the object. Note that this entry will show that the share was accessed but not what files in the share were accessed. A large number of these events from a single account may be an indicator of an account being used to harvest or map data on the network.
5142	A network share object was added.
5143	A network share object was modified.
5144	A network share object was deleted.
5145	A network share object was checked to see whether client can be granted desired access. Failure is only logged if the permission is denied at the file share level. If permission is denied at the NTFS level then no entry is recorded.

If detailed file share auditing is enabled in the Group Policy Management Console by navigating to Computer Configuration -> Policies -> Windows Settings -> Security Settings -> Advanced Audit Policy Configuration -> Audit Policies -> Object Access -> Audit Detailed File Share, then each file within each share that is accessed will generate an Event ID 5145 log entry. As you can imagine, this level of logging may generate a large volume of results.

The system initiating the access may also show evidence of the connections in the registry key NTUSER\Software\Microsoft\Windows\CurrentVersion\Explorer\MountPoints2.

Scheduled Task Logging

If history is enabled in the Task Scheduler application, through Event Viewer, or with the wevtutil command (see here for more details), then the %SystemRoot%\System32\winevt\Logs\Microsoft-Windows-TaskScheduler%4Operational log will record activity relating to scheduled tasks on the local system as follows:

Scheduled task activity event IDs

Event ID	Description
106	Scheduled Task Created. The entry shows the user account that scheduled the task and the name the user assigned to the task. The Logged date and time show when the task was scheduled. Look for the associated Event ID 200 and 201 for additional information.
140	Scheduled Task Updated. The entry shows the user account that updated the task and the name of the task. The Logged date and time show when the task was updated. Look for the associated Event ID 200 and 201 for additional information.
141	Scheduled Task Deleted. The entry shows the user account that deleted the task and the name of the task.
200	Scheduled Task Executed. Shows the task name and the full path to the executable on disk that was run (listed as the Action). Correlate this with the associated Event ID 106 to determine the user account that scheduled the task.
201	Scheduled Task Completed. Shows the task name and the full path to the executable on disk that was run (listed as the Action). Correlate this with the associated Event ID 106 to determine the user account that scheduled the task.

Also, see the Object Access Auditing section for additional Event IDs that may be recorded in relation to scheduled tasks.

Object Access Auditing

Object access auditing is not enabled by default but should be enabled on sensitive systems. To do so, simply set use the Local Security Policy to set Security Settings -> Local Policies -> Audit Policy -> Audit object access to Enabled for Success and Failure. When object access auditing is enabled, some activities are logged by default and others need to be explicitly configured. The reason for this is that object access occurs constantly on a system, so this log is designed to be more granular to allow objects of importance to receive extra auditing without overwhelming the logs trying to record all object access on the system. Object access audit events are stored in the Security log. If object access auditing is enabled, scheduled tasks get additional logging. The Event IDs related to scheduled tasks are:

Scheduled task event IDs

Event ID	Description
4698	A scheduled task was created. The event description contains the user account that created the task in the Subject section. XML details of the scheduled task are also recorded in the event description under the Task Description section and includes the Task Name. Additional tags of interest include the following: • `<Date>` shows the time of the logged event and matches the Logged field of the event itself. • `<Author>` shows the user that originally created the task, this does not change if another user later updates the task (see Event ID 4702 for additional information about how to determine whether a scheduled task was updated). • `<Description>` shows the description entered by the user. • `<Triggers>` provides information on when the task is scheduled to run. • `<User ID>` shows the user context under which the task will run, which may be different than the account used to schedule the task. If `<Logon Type>` shows Password, then the password for the account listed in `<User ID>` was entered at the time the task was scheduled, which may indicate an additionally compromised account. • `<Command>` shows the path to the executable that will run. Any arguments specified will be listed in the `<Arguments>` tag.
4699	A scheduled task was deleted. The Subject section of the event description contains the Account Name that deleted the task as well as the Task Name.
4700	A scheduled task was enabled. See Event ID 4698 for additional details.
4701	A scheduled task was disabled. See Event ID 4698 for additional details.
4702	A scheduled task was updated. The user who initiated the update appears in the Subject section of the event description. The details of the task after its modification are listed in the XML in the event description. Compare with previous Event ID 4702 or 4698 entries for this task to determine what changes were made. See Event ID 4698 for additional details.

Aside from scheduled tasks, individual file objects are frequently audited for object access. In addition to enabling the option for Success and/or Failure for Audit Object Access as mentioned earlier, to audit access to individual files or folders you also need to explicitly set the auditing rules in the file or folder's Properties dialog box by selecting the Security tab, clicking Advanced, selecting the Auditing tab, and setting the type of audit and the user account(s) for which auditing should be set. Detailed instructions can be found here:

https://docs.microsoft.com/en-us/windows/security/threat-protection/auditing/apply-a-basic-audit-policy-on-a-file-or-folder

For a process to use a system object, such as a file, it must obtain a handle to that object. Once auditing is enabled, the event IDs described below can be used to view access to important files and folders by tracking the issuance and use of handles to those objects.

Object handle event IDs

Event ID	Description
4656	A handle to an object was requested. When a process attempts to gain a handle to an audited object, this event is created. The details of the object to which the handle was requested and the handle ID assigned to the handle are listed in the Object section of the event description. Success or failure of the handle request will be indicated in the Keywords field. The account used to request the handle, as well as that account's associated Logon ID, is recorded in the Subject section of the event description. The details of the process requesting the handle are listed under the Process Information section of the event description. The Access Request Information shows the type of access requested. Note that obtaining a handle to an object does not mean that all the permissions requested were actually used. Look for additional Event ID 4663 entries with the same Handle ID (which is kept unique between reboots) to determine which permissions were used. You can also try to determine other actions taken by the same user during that session by searching for occurrences of the Logon ID (which is also unique between reboots).
4657	A registry value was modified. The user account and process responsible for opening the handle are listed in the event description. The Object section contains details of the modification, including the Object Name field, which indicates the full path and name of the registry key where the value was modified. The Object Value Name field contains the name of the modified registry key value. Note that this event generates only when a key value is modified, not if the key itself is modified.
4658	The handle to an object was closed. The user account and process responsible for opening the handle are listed in the event description. To determine the object itself, refer to the preceding Event ID 4656 with the same Handle ID.
4660	An object was deleted. The user account and process responsible for opening the handle are listed in the event description. To determine the object itself, refer to the preceding Event ID 4656 with the same Handle ID.
4663	An attempt was made to access an object. This event is logged when a process attempts to interact with an object, rather than just obtain a handle to the object. This can be used to help determine what types of actions may have been taken on an object (for example, read only or modify data). See Event ID 4656 for additional details.

Since Windows 8/Server 2012, additional logging can also be enabled in the Group Policy Management Console by navigating to Computer Configuration -> Policies -> Windows Settings -> Security Settings -> Advanced Audit Policy Configuration -> Audit Policies -> Object Access -> Audit Removeable Storage. Once enabled, Windows will create additional Event ID 4663 entries (see above) whenever an account access a file system object that is on removable storage. This can help identify when users are copying data to or from external media.

Audit Policy Changes

When audit policy changes, it impacts the evidence available to investigators and incident handlers, whether the change was done maliciously by an attacker or legitimately by an administrator. Fortunately, modern Windows systems do a good job of logging these changes when they occur. The Event ID used for this auditing is 4719:

- 4719 – System audit policy was changed. The Audit Policy Change section will list the specific changes that were made to the audit policy. The Subject section of the event description may show the account that made the change, but often (such as when the change is made through Group Policy) this section simply reports the name of the local system. Unfortunately, auditing Directory Services access is one area where Windows is still less than clear. You can find additional information here and here, and there are a number of third-party tools that provide additional visibility and accountability in modifications to Group Policy Objects.
- 1102 - Regardless of the settings in the audit policy, if the Security event log is cleared, Event ID 1102 will be recorded as the first entry in the new, blank log. You can tell the name of the user account that cleared the log in the details of the entry. A similar event, with ID 104, is generated in the System log if it is cleared.

Auditing Windows Services

Many attacks rely on Windows services either for executing commands remotely or for maintaining persistence on systems. While most of the events we have mentioned so far have been found in the Security Event Log, Windows records events related to starting and stopping of services in the System Event Log. The following events are often noteworthy:

- 6005 – The event log service was started. This will occur at system boot time, and whenever the system is manually started. Since the event log service is critical for security, it gets is own Event ID.
- 6006 – The event log service was stopped. While this obviously occurs at system shutdown or restart, its occurrence at other times may be indicative of malicious attempts to avoid logging of activity or to modify the logs.
- 7034 – A service terminated unexpectedly. The event description will display the name of the services and may display the number of times that this service has crashed.
- 7036 – A service was stopped or started. While the event log service has its own Event ID, other services are logged under the same Event ID. The event description provides the name of the service, but no details of which user account requested the service to stop is provided. The description will indicate that the service entered the running state when it is started or entered the stopped state when it is stopped.
- 7040- The start type for a service was changed. The event description will display the name of the service that was changed and describe the change that was made.
- 7045 – A service was installed by the system. The name of the service is found in the Service Name field of the event description, and the full path to the associated executable is found in the Service File Name field. This can be a particularly important event as many tools, such as psexec, create a service on the remote system to execute commands. Many of these tools will create a randomly named service (which stands out in the logs as highly unusual) or will run an executable from locations like the Temp folder. It is worth noting that some legitimate services, like Windows Defender, may also use names that look in part randomized, so it is worth examining any odd entries carefully to determine if they are malicious.

If you have enabled Advanced Audit Policy Configuration > System Audit Policies > System > Audit Security System Extension in your GPOs, Windows 10 and Server 2016/2019 systems will also record Event ID 4697 in the Security event log.

Wireless LAN Auditing

Windows maintains an event log dedicated to wireless local area network (WLAN) activity, and with rogue access points being a common attack vector for man-in-the-middle and malware attacks, it may be worth looking at unusual connections on devices with Wi-Fi capability, particularly those allowed to leave your environment. The log is located at %SystemRoot%\System32\winevt\Logs\Microsoft-Windows-WLAN-AutoConfig%4Operational.evtx. Event IDs of interest are:

Wi-Fi connection event IDs

Event ID	Description
8001	WLAN service has successfully connected to a wireless network. The event description provides the Connection Mode indicating if this was an automatic connection based on a configured profile (and the associated Profile Name) or a manual connection. The SSID of the access point, its authentication mechanism, and its encryption mechanism are also recorded.
8002	WLAN service failed to connect to a wireless network. Once again, the event description will contain the Connection Mode, associated Profile Name, and the SSID along with a Failure Reason field.

Process Tracking

Unlike many Linux shells (such as bash) the Windows cmd.exe shell does not maintain a history of commands run by users. This has created a noticeable gap in the ability of incident handlers to understand the actions that an attacker takes on a compromised host. The rise of "Living of the Land" attacks that do not rely on malware but instead use built-in Windows commands has only made this blind spot more damaging. While in the early days of Windows, auditing process creation was considered far too system intensive, modern Windows systems have greatly increased the efficiency of their auditing facilities, allowing for process tracking to be used to great effect. The addition of the ability to log full command lines in process creation events has gone a long way to remove the blinders from incident handlers and provide a trail which we can follow to uncover the actions taken by an attacker.

While not always required on every system, enabling this feature on key systems is increasingly becoming standard practice in security-conscious environments. This requires setting two separate Group Policy settings. The first is of course Computer Configuration -> Windows Settings -> Security Settings -> Local Policies -> Audit Policy -> Audit process tracking. However, to fully benefit from process tracking you should also enable the ability to capture the command line in those events. This requires a second setting located at Computer Configuration -> Administrative Templates -> System -> Audit Process Creation -> Include command line in process creation events. Keep in mind that some command line arguments may contain sensitive information such as passwords, so secure access to such logs accordingly and make users aware of the change in audit policy. Once enabled, Event ID 4688 in the Security log provides a wealth of information regarding processes that have been run on the system:

Event ID	Description
4688	A new process has been created. The event description provides the Process ID and Process Name, Creator Process ID, Creator Process Name, and Process Command Line (if enabled separately, as outlined earlier in this section). In addition to the details about the process, details about the user account used to launch the process are recorded in the Subject section. In pre–Windows 10/Server 2016 systems there is only one Subject. However, in Windows 10 and Server 2016/2019, we now receive details about the Creator Subject and the Target Subject. The Creator Subject (which is the same as the pre–Windows 10/Server 2016 Subject) lists the user context under which the Creator Process was running. The Target Subject lists the user context under which the newly created process is running. In addition to the details of the user context, we get information in the Token Elevation Type field about the user's administrative privileges that may have been assigned to the process. A Type 1 token indicates a full token, with all privileges available to that user account, such as when the user is the built-in administrator account or User Access Control (UAC) is disabled. Type 2 indicates that a full token was issued by the user specifying to bypass UAC, such as through the Run As Administrator option. A Type 3 token indicates that administrator privileges were removed due to UAC.

In addition the Event ID 4688, activation of process tracking may also result in additional Security log entries from the Windows Filtering Platform related to network connections and listening ports as follows:

Windows Filtering Platform (WFP) event IDs

Event ID	Description
5031	The Windows Firewall Service blocked an application from accepting incoming connections on the network.
5152	The WFP blocked a packet.
5154	The WFP has permitted an application or service to listen on a port for incoming connections.
5156	The WFP has allowed a connection.
5157	The WFP has blocked a connection.
5158	The WFP has permitted a bind to a local port.
5159	The WFP has blocked a bind to a local port.

The event descriptions of the Windows Filtering Platform events are self explanatory and detailed, including information about the local and remote IPs and port numbers as well as the Process ID and Process Name involved.

As can be seen, the information logged by enabling process tracking auditing can be of immense value, but can also generate a large amount of data. Experiment with your test environment to come up with a balance that can appropriately increase security auditing in your production environment.

Additional Program Execution Logging

If AppLocker is configured in your environment (a step that can help frustrate an adversary and should be considered), dedicated AppLocker event logs will be generated as well. Presented in Event Viewer under Application and Services Logs\Microsoft\Windows\AppLocker, these event logs are stored with the other event logs in C:\Windows\System32\winevt\Logs and have names such as Microsoft-Windows-AppLocker%4EXE and DLL.evtx. There are separate logs covering executables and dynamic-link libraries (DLLs), Microsoft installers (MSI) and scripts, packaged app deployment, and packaged app execution. The event logs generated will vary depending on whether AppLocker is set to audit-only mode or blocking mode. Details of the specific event IDs that may apply to your situation can be found at here.

Remember also that your antivirus or other endpoint detection and response systems may generate useful logs that may record files scanned and/or blocked. For example, Windows Defender maintains an event log located at C:\Windows\System32\winevt\Logs\Microsoft-Windows-Windows Defender%4Operational.evtx and Microsoft-Windows-Windows Defender%4WHC.evtx that contains information about potential malware that was detected and suspicious scripts that were run (as reported by the Antimalware Scan Interface [AMSI]). Event IDs of potential interest in this log include:

Windows Defender suspicious event IDs

Event ID	Description
1006	The antimalware engine found malware or other potentially unwanted software.
1007	The antimalware platform performed an action to protect your system from malware or other potentially unwanted software.
1008	The antimalware platform attempted to perform an action to protect your system from malware or other potentially unwanted software, but the action failed.
1013	The antimalware platform deleted history of malware and other potentially unwanted software.
1015	The antimalware platform detected suspicious behavior.
1116	The antimalware platform detected malware or other potentially unwanted software.
1117	The antimalware platform performed an action to protect your system from malware or other potentially unwanted software.
1118	The antimalware platform attempted to perform an action to protect your system from malware or other potentially unwanted software, but the action failed.
1119	The antimalware platform encountered a critical error when trying to take action on malware or other potentially unwanted software.
5001	Real-time protection is disabled.
5004	The real-time protection configuration changed.
5007	The antimalware platform configuration changed.
5010	Scanning for malware and other potentially unwanted software is disabled.
5012	Scanning for viruses is disabled.

Additional details on Windows Defender event log records can be found here.

Windows exploit protection is a feature of Windows 10 that can provide excellent defense against a range of adversary exploitation techniques. This feature can protect both the operating system and individual applications from common attack vectors, blocking the exploitation when it otherwise would have resulted in system compromise. Although some features of exploit protection are enabled by default, many are disabled due to their potential to interfere with legitimate software. When enabled, this feature logs its activities in the C:\Windows\System32\winevt\Logs\Microsoft-Windows-Security-Mitigations%4KernelMode.evtx and Microsoft-Windows-Security-Mitigations%4UserMode.evtx log files. More details can be found here.

Another option to enhance visibility into processes that run on systems in your environment is to implement Sysmon, a free utility by Sysinternals, which is now a part of Microsoft. Sysmon can be freely downloaded here.

When deployed on a system, Sysmon installs as a system service and device driver to generate event logs related to processes, network connections, and modifications to file creation times. It creates a new category of logs that are presented in Event Viewer under Applications and Services Logs\Microsoft\Windows\Sysmon\Operational and is stored in C:\Windows\System32\winevt\Logs\Microsoft-Windows-Sysmon%4Operational.evtx. An example of useful event IDs generated by Sysmon include:

Event IDs generated by Sysmon

Event ID	Description
1	Process creation (includes many details such as process ID, path to executable, hash of executable, command line used to launch, user account used to launch, parent process ID, path and command line for parent executable, and more).
2	A process changed a file creation time.
3	Network connection.
4	Sysmon service state changed.
5	Process terminated.
6	Driver loaded.
7	Image loaded (records when a module is loaded in a specific process).
8	CreateRemoteThread (creating a thread in another process).
9	RawAccessRead (raw access to drive data using \\.\ notation).
10	ProcessAccess (opening access to another process's memory space).
11	FileCreate (creating or overwriting a file).
12	Registry key or value created or deleted.
13	Registry value modification.
14	Registry key or value renamed.
15	FileCreateStreamHash (creation of an alternate data stream).
16	Sysmon configuration change.
17	Named pipe created.

18	Named pipe connected.
19	WMIEventFilter activity detected.
20	WMIEventConsumer activity detected.
21	WMIEventConsumerToFilter activity detected.
22	DNS query event (Windows 8 and later)
255	Sysmon error

Auditing PowerShell Use

Microsoft continues to increase the amount of logs available surrounding PowerShell to help combat its nefarious use. Once again, these logging facilities must be enabled via Group Policy, specifically at Computer Configuration -> Policies -> Administrative Templates -> Windows Components -> Windows PowerShell. There are three basic categories of logging that may be available, depending on the version of Windows in question.

- Module Logging
  - Logs pipeline execution events;
  - Logs to event logs.
- Script Block Logging
  - Captures de-obfuscated commands sent to PowerShell;
  - Captures the commands only, not the resulting output;
  - Logs to event logs.
- Transcription
  - Captures PowerShell input and output;
  - Will not capture output of outside programs that are run, only PowerShell;
  - Logs to text files in user specified location.

Once enabled, these logs can provide a wealth of information concerning the use of PowerShell on your systems. If you routinely run lots of PowerShell scripts, this can produce a large volume of data, so be sure to test and tune the audit facilities to strike a balance between visibility and load before deploying such changes in production.

PowerShell event log entries appear in different event logs. Inside of %SystemRoot%\System32\winevt\Logs\Microsoft-Windows-PowerShell%4Operational.evtx you will find two events of particular note:

Event ID	Description
4103	Shows pipeline execution from the module logging facility. Includes the user context used to run the commands. Hostname field will contain Console if executed locally or will show if run from a remote system.
4104	Shows script block logging entries. Captures the commands sent to PowerShell, but not the output. Logs full details of each block only on first use to conserve space. Will show as a Warning level event if Microsoft deems the activity Suspicious.

Additional entries can be found in the %SystemRoot%\System32\winevt\Logs\Windows PowerShell.evtx log:

Event ID	Description
400	Indicates the start of command execution or session. Hostname field shows if (local) Console or the remote session that caused the execution.

800	Shows pipeline execution details. UserID shows account used. Hostname field shows if (local) Console or the remote session that caused the execution. Since many malicious scripts encode options with Base64, check the HostApplication field for options encoded with the -enc or -EncodedCommand parameter.

Remember that PowerShell Remoting requires authenticated access, so look for the associated Account Logon and Logon events as well.