FORWARD DEFENSE

Analyst Reference

Windows Event Log Analysis

Version 20181019

Table of Contents

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. Windows logging is a robust capability and exhaustive treatments of the topic are very hard to find. Two references that provide additional details are Randy Franklin Smith's Ultimate Windows Security site and the book Mastering Windows Network Forensics and Investigation.

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.

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

Malicious actors may create rogue accounts on either local systems or at the domain level. 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.

- 4720 – A user account was created
- 4722 – A use account was enabled
- 4723 – An 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

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 all 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:

- 4768 – The successful issuance of a Ticket Granting Ticket (TGT) shows that a particular user account was authenticated by the domain controller. The Network Information section of the event description will contain additional information about the remote host in the event of a remote logon attempt. The Keywords field will indicate if the authentication attempt was successful or failed. In the event of a failed authentication attempt, the Result Code in the event description will provide additional informa tion about the reason for the failure, as specified in RFC 4120. Some of the more commonly encountered codes are:

Decimal	Hex	Meaning
6	0x6	Username not valid
12	0xC	Policy restriction prohibiting this logon (such as a work-station 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: Mastering Windows Network Forensics and Investigation, by Steve Anson et al., 2nd ed., John Wiley & Sons, Inc., 2012, p. 458..

A table decoding other possible Result Codes can be found here.

- 4769 – A service ticket was requested by a user account for a specified resource. This event description will show 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 will indicate if the request for the service ticket was successful or failed. In the case of a failure, the Result Code will indicate the reason for the failure. A table decod ing each possible Result Code can be found here.

- 4771 - Depending on the reason for a failed Kerberos logon, either Event ID 4768 or Event ID 4771 will be created. In either case, the Result Code in the event description will pro vide additional information about the reason for the failure. A table decoding each possi ble Result Code can be found here.

- 4776 – While less common in a domain environment, NTLM may still be used for authenti cation. Additionally, many attack tools downgrade authentication attempts to NTLM when authenticating. While these types of authentication do frequently occur with legiti mate traffic, such as some authentication requests originating by IP address rather than computer name, their presence may also indicate a non-standard tool being used to authenticate. The Network Information section of event description will contain addi tional information about the remote host in the event of a remote logon attempt. The Keywords field will indicate if the authentication attempt was successful or failed. In the event of authentication failure, the Error Code in the event description will provide addi tional details about the reason for the failure as follows:

Error Code	Meaning
C0000064	The username is invalid
C000006A	The password is invalid
C000006F	The attempt violates a time-of-day policy restriction
C0000070	The account is not allowed to log on from this workstation because of a security policy restriction.
C0000071	The account's password has expired
C0000072	The account is disabled
C0000193	The account has expired
C0000224	The user is required to change password at next logon
C0000234	The account is locked out

Source: Mastering Windows Network Forensics and Investigation, by Steve Anson et al., 2nd ed., John Wiley & Sons, Inc., 2012, p. 438.

A series of failed 4776 events with Error Code C000006A followed by an Error Code C0000234 may be indicative of a password guessing attack (or a user who has simply forgotten the account password).

- If a new account is created, Event ID 4720 will be created on a domain controller for a do main account or on the local system for a local account.

On systems being accessed, Event IDs of note include:

- 4624 – A logon to a system has occurred. Type 2 indicates an interactive (local) logon, while 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 between the recorded host name and IP address may be indicative of SMB Relay attacks, where an attacker relays a request from one system from an IP address not associated with that system.

Logon events will contain a Type code in the event description. The meaning of this code is as follows:

Logon Type	Description
2	Interactive, such as logon at keyboard and screen of 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 (i.e. connection to shared folder on this computer from elsewhere on network). This represents a non-interactive logon, which does not cache the user's credentials in RAM or on disk.
4	Batch (i.e. 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 (A service was started by the Service Control Manager).
7	Unlock (i.e. unattended workstation with password protected screen saver)
8	NetworkCleartext A user logged on to this computer from the network. 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 IIS with "basic authentication."
9	NewCredentials such as with RunAs or mapping a network drive with alternate credentials. This logon type does not seem to show up in any events. If you want to track users attempting to logon with alternate credentials look for Event ID 4648.
10	RemoteInteractive (Terminal Services, Remote Desktop or Remote Assistance). See 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 credentials.

(Table includes details from https://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))

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.

For additional information about this event and tips for proactive security monitoring based on this Event ID see this Microsoft article.

- 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 logon 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 there provides additional details about the event as follows:

Status and Sub Status Codes	Description (not checked against "Failure Reason:")
0xC0000064	user name does not exist
0xC000006A	user name is correct but the password is wrong
0xC0000234	user is currently locked out
0xC0000072	account is currently disabled
0xC000006F	user tried to logon outside his day of week or time of day restrictions
0xC0000070	workstation restriction, or Authentication Policy Silo violation (look for event ID 4820 on domain controller)
0xC0000193	account expiration
0xC0000071	expired password
0xC0000133	clocks between DC and other computer too far out of sync
0xC0000224	user is required to change password at next logon
0xC0000225	evidently a bug in Windows and not a risk
0xc000015b	The user has not been granted the requested logon type (aka logon right) at this machine

(source: https://www.ultimatewindowssecurity.com/securitylog/encyclopedia/event.aspx?eventID=4625)

- 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 4634 in many cases. The Logon ID field can be used to tie the 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 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 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 ad ministrator access are granted to a logon. As with all logon events, the event log will be generated by the system being accessed.
- 4776 – An NTLM-based authentication has occurred. When found on a non-domain controller this indicates the use of a local user account. Since most domains are designed to use domain rather than local user accounts, the presence of this Event ID on member servers or client workstations is frequently suspicious. See the above description of Do main Controller events for additional detail on Event ID 4776.
- 4778 – This event is logged when a session is reconnected to a Windows station. This can occur locally when the user context is switched vis 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 above. To differentiate between RDP vs local session switch ing, look at the Session Name field within the event description. If local, it will be "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 vis 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 above. To differentiate between RDP vs local session switching, look at the Session Name field within the event description. If local, it will be "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%/Sys tem32/winevt/Logs/Microsoft-Windows-TerminalServices-LocalSessionManager%4Opera tional 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.
- Additional information about RDP Sessions can also be found in the %SystemRoot%/Sys tem32/winevt/Logs/Microsoft-Windows-TerminalServices-RemoteConnectionMan ager%4Operational log file. Event ID 1149 in this log will show the user account and source IP used to initiate an RDP session.

Note: Successful Remote Desktop Protocol connections will log as with 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. Restricted Admin mode may impact this, but also opens the possibility of pass-the-hash attacks. For more information, see this article. Failed RDP logons will usually result in Logon Type 3.

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:

- 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.
- 5142 – A network share object was added.
- 5143 – A network share object was modified.
- 5144 – A network share object was deleted.

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:

- 106 – Scheduled Task Created. The entry will show the user account that scheduled the task and the name the user assigned to the task. The Logged date and time will show when the task was scheduled. Look for the associated Event ID 200 and 201 for additional information.
- 140 – Scheduled Task Updated. The entry will show the user account that updated the task and the name of the task. The Logged date and time will show when the task was updated. Look for the associated Event ID 200 and 201 for additional information.
- 141 – Scheduled Task Deleted. The entry will show 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"). Note that the 200 event lists the action first and the 201 event lists the task name first. 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:

- 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:
  - <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 users later updates the task.
  - <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 the <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 event description contains the user account that created the task in the Subject section as well as the task name.
- 4700 – A scheduled task was enabled. See Event ID 4698 above for additional details.
- 4701 – A scheduled task was disabled. See Event ID 4698 above for additional details.
- 4702 – A scheduled task was updated. The user who initiated the update will appear 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 above for additional details.

Aside from scheduled tasks, individual file objects are frequently audited for object access. In addition to enabling the Success and/or Failure for Audit object access as mentioned above, to audit access to individual files or folders you need to explicitly set the auditing rules in the file or folder's Properties dialogue box, System tab, under Advanced settings. As with other permissions, these too can be pushed to child objects or inherited from parent objects. Once auditing is enabled, the following Event IDs can be used to track access to important files and folders:

- 4656 – A handle to an object was requested. When a process attempts to gain a handle to an audited object, this event is created. Success of failure will be logged depending on the permissions under which the requesting process was running (which is logged in the event description under the Subject section). The details of the object are listed in the Object section of the event description, and the details of the process requesting the handle is listed under Process Information section of the event description. The Access Request Information shows the type of access requested. Note that simply obtaining a handle to an object does not mean that all the permissions requested were actually used. Look for additional Event ID 4663 with the same Handle ID (which is kept unique between reboots). You can also try to determine other actions take by the same user during that session by searching for occurrences of the Logon ID (which is also unique between reboots).
- 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 with a tool like EventLogEdit.
- 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.

Wireless LAN Auditing

Windows now maintains an Event Log dedicated to WLAN activity, and with rogue access points being a common attack vector for man-in-the-middle and malware attacks it is worth looking at unusual connections on devices with Wi-Fi capability. The log is located at Additional information about RDP Sessions can be found in the %SystemRoot%/System32/winevt/Logs/Microsoft-Windows-WLAN-AutoConfig%4Operational.evtx. Event IDs of interest include:

- 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 Encryption mechanism are also recorded.
- 8002 – WLAN service faile dot connect to a wireless network. Once again the event description will contain Connection Mode, associated Profile Name and SSID are listed 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:

- 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 outline above). In addition to the details about the process, we also get details about the user involved in the Subject lines. In pre-Windows 10/2016 systems there is only one Subject. However, in Windows 10 and 2016 we now receive details about the Creator Subject and the Target Subject. The Creator Subject (which is the same as the pre-Windows 10/2016 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 also get information in the Token Elevation Type field regarding 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. You can read more about UAC here.

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:

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

Auditing PowerShell Use

PowerShell remoting is enabled by default for members of the Administrators group and Remote Management Users group on Windows Server 2012 and later. If your network connection is set to a Private network, Windows firewall will likewise allow PowerShell remoting. While disabling PowerShell remoting may help prevent its use by attackers, it also disables one of the most powerful tools in the administrator's arsenal for daily administrative tasks as well as baselining and incident handling.

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:

- 4103
  - Shows pipeline execution from the module logging facility;
  - Includes the user context used to run the commands;
  - Hostname field will show "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 are located in the %SystemRoot%/System32/winevt/Logs/Windows PowerShell.evtx log, as follows:
- Event 400
  - Indicates the start of command execution or session;
  - Hostname field shows if (local) console or remote session caused the execution.
- Event 800
  - Shows pipeline execution details;
  - UserID shows account used;
  - Hostname field shows if (local) console or remote session caused the execution;
  - Since many malicious scripts encode options with Base64, check the HostApplication field for options encoded with -enc or -EncodedCommand parameter.

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