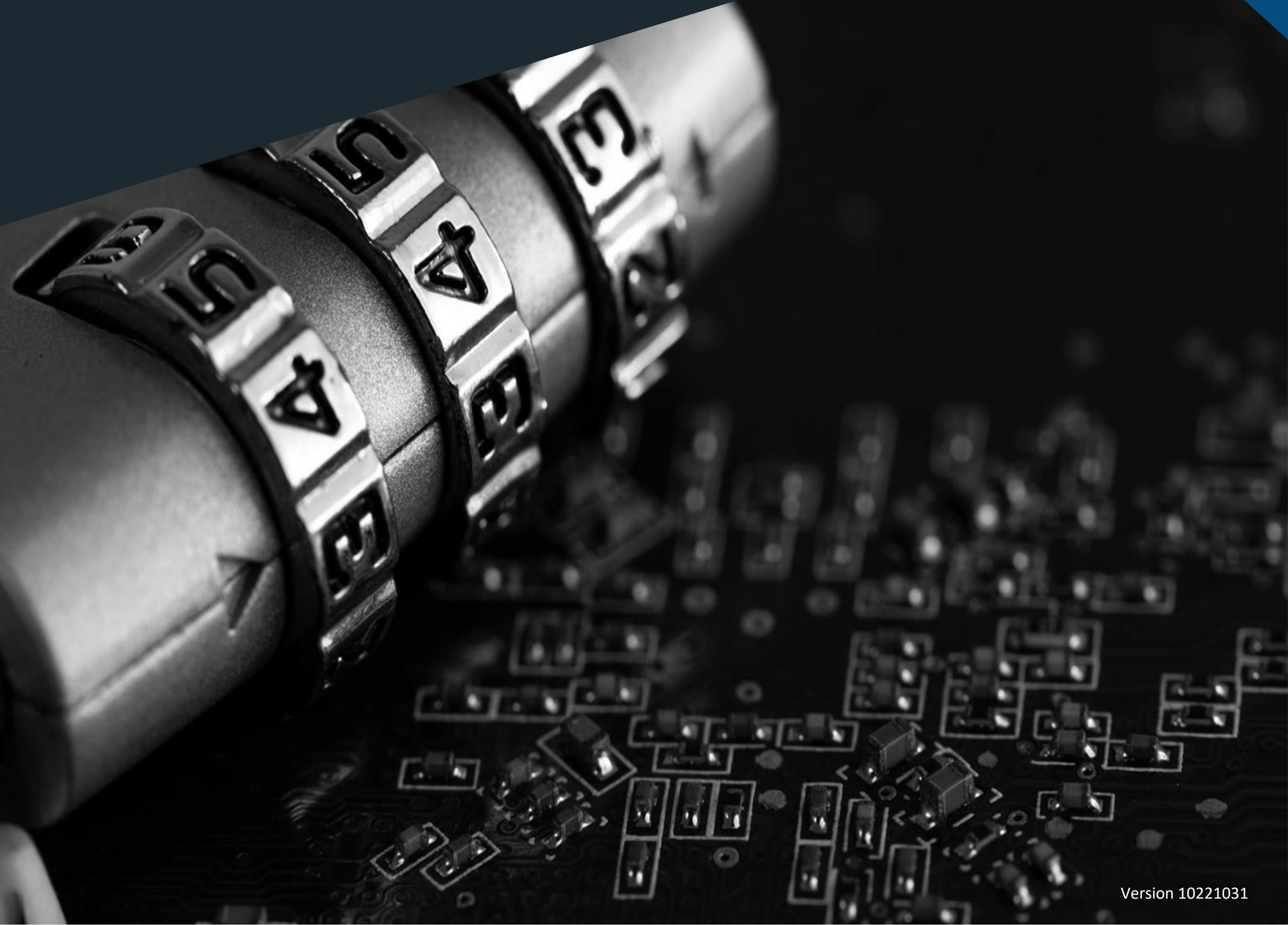
APPLIED INCIDENT RESPONSE

Analyst Reference

Credential Defense



Version 10221031

Contents

Overview

Digital devices incorporate access controls to restrict which actions are permitted on a system depending on which user is accessing that system. To do so, the system must identify which user is interacting with the system, determine which rights and permissions should be allowed for that user, and ultimately grant access to permitted functions while denying access to those that are not allowed. This process involves two separate but interconnected components: authentication and authorization.

Authentication is the process used to identify which user is accessing the system. Credentials such as usernames and passwords, multifactor authentication tokens, access keys, or others can be used to facilitate the authentication process.

Once authentication is completed, the user is frequently issued a digital proof of identity to represent their authenticated state and authorize access to allowed resources throughout the environment. These authorization credentials can include things such as tickets or tokens that are presented by the user to other systems to gain access to additional resources, within the constraints of the permissions granted to their account.

Theft of either an authentication credential or an authorization credential can allow an attacker to impersonate a user to gain access to any resources to which that account has permissions.

Throughout this document are hyperlinks to external sites to explain additional technical and implementation details. These links are valid at the time of this writing, and point to reputable organizations such as MITRE.org and Microsoft.com, but are not controlled or maintained by the author.

Credential Access (MITRE ATT&CK TA006) is an attack tactic used by most threat groups, and with good reason. Possession of a valid credential grants access to additional resources within the network environment, facilitating discovery (TA0007) and lateral movement (TA0008). Furthermore, because this access frequently mimics standard user behavior, it can be more difficult to identify as malicious activity, facilitating defense evasion (TA0005).

Attackers can access and leverage stolen credentials in different ways. Systems that rely on a username and password are subject to brute forcing attacks (T1110), such as credential guessing, credential stuffing, and credential spraying. Attack tools such as Mimikatz (S0002) allow theft of credentials during interactive logon sessions directly from memory of the Local Security Authority Subsystem (LSASS) process. Access keys for cloud service provider accounts, secure shell access, and APIs can be stolen from disk and leveraged by attackers for access to additional systems. Password hashes for domain or local accounts can be stolen from their storage locations on disk (**T1003**) such as the Security Account Manager (SAM) registry hive, the */etc/shadow* file, or the NTDS.dit file on domain controllers. Offline password cracking (T1110) utilities such as hashcat can then be used to determine the original plaintext password correlating to a stolen password hash, or the stolen hash can directly be used to access other systems using pass-the-hash (T1550), overpass-the-hash, and similar attack techniques. In general, authenticated users are typically issued some form of digital proof of identity, such as a ticket (T1558) or token (T1635), that can be used to gain access to authorized resources. These authorization credentials are frequently targeted by attackers because possession of the authorization credential serves a sufficient proof of identity to access systems with whatever permissions the authenticated user has. Possession of such credentials enables a variety of credential attacks including pass-the-ticket (T1550), pass-the-token, and others.

Because credential theft attacks are so popular among adversary groups, the defense of credentials should be a top priority for all organizations. This document will discuss preventive and detective controls to safeguard your credentials and detect when a credential is being abused by an attacker.

Preventing Abuse

Multifactor Authentication

Theft of a credential used to authenticate to a system (such as a username and password) allows an attacker to impersonate the user associated with that credential. In the case of usernames and passwords, this opens up a variety of brute force attacks (T1110) where an attacker can attempt to determine a valid username and password combination on a system to which they otherwise have no access. The frequency of these types of attacks, and the ease with which they can be delivered, necessitates that additional factors of authentication be required so that simple possession of the username and password is not sufficient to successfully impersonate a user.

Multifactor authentication, in which a combination of factors (including something you know, something you are, or something you have) is needed to complete an authentication exchange, should ideally be required for most accounts. Accounts that are exposed to the Internet, and therefore more exposed to brute force attacks (T1110) such as password spraying, credential stuffing, and password guessing, especially should require multifactor authentication. In addition, privileged accounts that grant access to sensitive systems should be required to use multifactor authentication in all cases.

Dedicated Administrative Workstation

Most organizations already take the step of assigning multiple user accounts to individuals with responsibility for administering IT systems. One account is assigned, as with most members of the organization, with minimal privileges and is intended for daily use conducting activities like browsing the Internet and checking email. A separate account is assigned to individuals who have a need for increased account privileges to perform their job functions, such as administering computer systems. One individual may therefore have two separate accounts: a low privileged account for daily use, and an increased privileged account for use administering IT resources. The privileged account should never be used for general-purpose functions, such as web surfing and email access. The privileged account should only ever be utilized when performing tasks that require its enhanced permissions.

A dedicated administrative workstation takes this idea one step further. Employees who require privileged credentials should have access to not one, but at least two, different workstations on which to conduct their activities. One workstation is used for general IT purposes utilizing their low privileged, standard user account. The other workstation, a dedicated administrative workstation, is used exclusively for tasks requiring privileged account logon to administer or otherwise interact with IT systems using the increased permissions of an administrative account.

The reason that the separation of workstations is critical is that credentials are most at risk when they are used to perform an interactive logon, such as logon at the console or default use of Remote Desktop Protocol (RDP). Similarly, credentials such as access keys are frequently stored on workstations used for administrative purposes. These workstations therefore require an elevated level of security to safeguard these sensitive credentials.

Dedicated administrative workstations should utilize all available security controls to harden them against attack. Where possible, they should not allow inbound remote connectivity, but instead require the user to logon at the keyboard to reduce the possibility of an attacker laterally moving to such a workstation. Virtualization based security and application control should be enabled on these systems. Restrictions should be in place to prevent arbitrary web surfing or other potentially dangerous online activities. Additional guidance from Microsoft on the configuration and use of dedicated access workstations, sometimes called secure administrative workstations (SAW) or privileged access workstations (PAW) can be found here and here.

Another significant advantage of having dedicated administrative workstations is that it makes detecting credential abuse, should one be compromised, much easier. Any logon using an administrative credential that does not originate from a dedicated administrative workstation should generate an immediate security alert, as such a logon would indicate either a violation of policy by an administrative user or a compromise of a privileged credential.

Dedicated administrative workstations also enable additional security controls in your environment. Individual endpoints that are not used for administrative purposes can be configured with host-based firewalls to deny remote access unless the connection originates from a dedicated administrative workstation. This enables common remote access technologies such as PowerShell Remoting to be utilized by your authorized administrators, while denying the ability of attackers to leverage these protocols for lateral movement between arbitrary systems.

Tiered Administrative Permissions

In keeping with the concept of least privilege, administrator accounts should only be assigned the permissions necessary to perform a specific set of tasks. When a privileged credential is stolen, the attacker has access to any system that the stolen credential can access. Permission assignment to privileged credentials should always follow the concept of least privilege to restrict collateral damage if one of those credentials is compromised in the future.

One way to approach this is through a tiered administrative model. Microsoft has recommended such a model for many years and has integrated it into a broader enterprise access model and their privileged access security levels. The model recognizes that compromise of privileged credentials can result in extreme damage to any organization. To minimize not only the risk of theft of such credentials, but also the damage should one be compromised, IT assets should be broken up into different tiers, or levels, of sensitivity and separate administrative accounts should be designated for each of these tiers.

For example, Microsoft's original recommendation specified that Tier 0 be reserved for domain controllers and domain administrators. Since domain administrator accounts, and enterprise admin accounts, possess some of the most valuable privileges in an organization, their use should be heavily restricted to only tasks that require these specific permission sets. Accordingly, they should only be utilized when interacting with domain controllers themselves.

Servers and similarly sensitive systems are then placed in Tier 1. Accounts with permissions to administer these systems should have no permissions on general-purpose workstations (Tier 2) nor on domain controllers (Tier 0). This prevents lateral movement between tiers should a privileged credential be compromised. Dedicated administrative workstations used to administer Tier 1 devices are considered a part of Tier 1 as well.

Tier 2 of the Microsoft model is reserved for workstations and other general-purpose computing devices. Accounts and permissions to administer these devices, such as helpdesk accounts, should have no permissions on servers and should only ever be used interact with workstations or other Tier 2 devices. Because general-purpose workstations, and the users who utilize them, represent the largest attack surface for most organizations, these devices should be considered the highest risk. As a result, privileged accounts for Tier 0 and Tier 1 should never be used on any Tier 2 device, and they should explicitly be prohibited by Group Policy from doing so.

While the original tiered administrative model is still applicable for on-premise Windows Active Directory deployments, additional details and implementation considerations for hybrid cloud environments can be found in Microsoft's enterprise access model, their privileged access security levels, and their rapid modernization plan.

In most organizations, local administrator accounts are not necessary and should be disabled. Standard domain users should not be given local administrator permissions on their systems, since doing so allows attackers who compromise a user's credentials to override critical security controls and facilitate additional credential attacks. No accounts should be shared by multiple users, including default accounts like the domain administrator. Each account should be uniquely assigned to only one person.

Network Segmentation

just as the tiered administrative model attempts to provide separation based on the sensitivity of systems, network segmentation provides additional separation at the IP layer. Network segments or boundaries represent potential security controls. Whenever two different networks, or subnetworks, intersect, you can control the types of communication allowed between the two network segments. At its most basic, this would involve subnetting your environment and utilizing firewalls between each subnet to restrict the computers or devices that are allowed to communicate between the subnets. When a credential is stolen in one subnet, if the firewall blocks access to systems in the other subnet, that credential cannot be directly leveraged by the attacker against the systems in the other subnet.

Another way to achieve segmentation is through virtual local area networks (VLANs) or private virtual local area networks (PVLANs) with access control enforced between the VLANs. If an attacker compromises a credential, but is unable to achieve connectivity to a port on another system that would allow use of that credential, then damage from the compromise is greatly reduced.

An alternative is to utilize host-based firewalls to achieve a similar objective. The Windows Defender Firewall can easily be configured in a scalable manner using Group Policy. Other commercial, host-based firewall solutions offer similar configurability. Each endpoint should only be allowed to connect to, and receive communication from, systems for which it has a legitimate business requirement to communicate. As an example, most workstation should not accept incoming connections from other workstations, since most networks are not architected in a manner that requires such communication. However, when an attacker has stolen a credential from one workstation, they will typically try to leverage that credential on other workstations to achieve lateral movement. By using host-based firewalls to deny these connections, we deny the attacker a key technique in advancing their attack.

The more segmentation you can implement within your environment, the harder discovery (TA0007) and lateral movement (TA0008) will be for an attacker. Ideally, your network should be an inhospitable maze for any attacker unlucky enough to find itself tasked with operating in your environment. Network segmentation coupled with the tiered administrative permissions provides multiple barriers between a compromised host and other high value systems within your organization.

Virtualization-based Security

Microsoft has invested heavily in using a hardware root of trust, coupled with a machine hypervisor, to offer a variety of security features that rely upon virtualization technology. These virtualization-based security (VBS) features offer some of the best protections for credentials on individual endpoints and should be utilized by all organizations.

Just as two virtual machines operating on the same hypervisor platform are kept isolated from one another, Microsoft creates virtual secure containers using the Hyper-V hypervisor to isolate critical areas of memory from the rest of the operating system. Sensitive information, such as credentials for the current interactive logons, can be stored in the secure containers with access controlled by the hypervisor. In this way, even if the operating system is compromised to System level, the hypervisor does not grant access to the credentials stored in the secure container to an attacker. This technology has proven effective against tool such as Mimikatz and will be a key component of device security moving forward.

Windows Defender Credential Guard is an important control to enable on any Windows computer that supports it. An optional feature on most versions of Windows 10, this control has seen minimal adoption despite being effective against many of the most common credential theft techniques. Beginning with Windows 11 Enterprise, version 22H2, this feature is turned on by default.

Windows Defender Credential Guard creates a secure virtual container to store credential information including the credentials of the currently logged-on user. If the operating system is compromised, even to administrator or System level, this control prevents an attacker from reaching into system memory to steal those credentials.

Windows Defender Application Guard allows administrators to protect the underlying operating system from some of the riskiest behaviors of their user base. Most intrusions begin with a client-side attack, involving social engineering of an end-user. The sophistication of these attacks makes user security awareness training alone an insufficient control, since users cannot always determine when an attachment or link might be malicious. Windows Defender Application Guard offers a configurable way to run high-risk applications, such as browsers and Microsoft Office programs, in virtual secure containers. Should one of these containers access malicious code, the code is constrained from accessing the primary operating system, thereby reducing the associated damage. Availability of these features depends on your Windows and Office license levels.

Note: Windows Defender Remote Credential Guard, despite the similarity in name, is not a VBS feature. Instead, it changes the way authentication and authorization are handled during Remote Desktop Protocol (RDP) access to prevent the credential from being stored in memory on the remote system. While not related to VBS, it is nonetheless an important control to enable if RDP is used within your organization.

Application Control

Application control solutions provide a manageable means of controlling which programs are allowed to run on an endpoint. Using a rule-based approach, modern application control solutions provide one of the most cost-effective security controls available. Most Windows licenses include Windows Defender Application Control and AppLocker, which together provide an effective means of restricting execution of unauthorized code on a Windows system. While any control can be bypassed, application control significantly increases the difficulty level for attackers to gain access to a system and take malicious actions on that system.

Lock Down Older Protocols

Microsoft has emphasized backward compatibility with each version of the Windows operating system. Over several decades of existence, this has resulted in Windows supporting several inherently insecure, older protocols. While the least secure protocols are disabled by default on all modern Windows systems, attackers still target these protocols by re-enabling them on compromised systems.

Any protocol is subject to attack. Therefore, disabling any protocol that is not necessary within your environment is an important part of hardening your environment. While some protocols such as LanMan (LM), NTLMv1, and SMBv1 have particularly serious flaws that makes their use inherently insecure, even ubiquitous protocols such as NTLMv2 and Kerberos are frequently targeted by attackers. Any protocol that a computer system does not need to support should be disabled on that system.

Disabling protocols in a Windows environment is usually done through Group Policy. Keep in mind, however, that if an attacker achieves local administrator permissions on an individual system, the attacker can override the settings on that system to reenable legacy protocols. Once enabled, the attacker can force authentication exchanges using these older protocols between different processes on that computer to take advantage of cryptographic flaws in protocols such as NTLMv1 and expose the credential information.

To help mitigate this risk, network defenders should take multiple steps. First, ensure protocols that are not necessary for business purposes are disabled using Group Policy. This should include LM, NTLMv1, and SMBv1 but could also include SMBv2, NTLMv2, and support for weaker algorithms by Kerberos on systems where legacy support is not required. In addition, the PowerShell v2 Engine should be removed using Roles and Features if it is in place.

Second, in keeping with the concept of least privilege, users should not be given administrator access to their local workstations unless required for a defined business purpose. Many organizations mistakenly assume that since the files on a user's workstation contain information that the user should be allowed to access, that there is no harm in giving users local administrative permissions. Remember that if a user is tricked into executing malicious code, such as through a social engineering attack, that code runs with whatever permissions have been given to that user's account. If that includes local administrative permissions, attackers have an easier time not only directly stealing credentials from memory but also overriding default security controls such as disabling legacy protocols. With local administrative permissions, an attacker can reenable protocols such as NTLMv1 and force an authentication exchange on the local system to expose the local credential.

Third, all privileged accounts should be placed in the Protected Users Security Group. This group enforces restrictions on the use of some of these legacy protocols, caching of sensitive credential information, and other security controls anywhere the accounts are used. Keep in mind that this can have an operational impact on the ways in which these accounts can be leveraged, so careful testing and possible modification of process might be required to enable this control without impacting operations. Nonetheless, the security benefits usually outweigh any inconvenience that may be encountered.

*Thank you to Jeff McJunkin (@jeffmcjunkin) for contributing to this section.

Detecting Abuse

If a credential is compromised, attackers are likely to leverage that credential to access different resources in the network. The resulting pattern of behavior is normally different from behaviors during normal operation. By monitoring the use of privileged credentials, and understanding normal patterns of behavior, detection of abnormalities resulting from compromise can be achieved.

Active threat hunting, in which a hypothesis formulated around the compromise of a credential is used to guide a methodical search for evidence of such a compromise, is one way in which to detect these types of attacks. User and entity behavior analytics (UEBA) tools, which leverage machine learning algorithms to detect abnormal patterns of behavior, can also help detect these types of attacks in a more automated fashion. These tools require that an identifiable baseline of normal behavior exists within your organization. Unfortunately, many organizations lack disciplined use of their privileged credentials, making these types of patterns difficult to discern. Use of dedicated administrative workstations and tiered administrative permissions as discussed previously in this document help in producing an orderly and predictable pattern for the legitimate use of administrative credentials.

Security Information and Event Management (SIEM) systems, or other big data analytic platforms such as Elastic Stack, can also be leveraged to generate alerts when unusual behavior occurs. Examples include the first time an administrator account logs in to a system; unusual network authentication such as workstation-to-workstation connections; use of an instance credential in cloud environments to authenticate to any new resource; unusually large numbers of logons from a single account in a set period of time; large numbers of failed logons, as would be observed from password-guessing attacks; large numbers of explicit use of credential entries, as would be observed in password-spray attacks; failed logons resulting from policy violations, such as restrictions on where privileged accounts should be used; privileged account logon originating from a system that is not a dedicated administrative workstation; or any other pattern that, in your environment, might be indicative of a compromised credential.

Windows systems record detailed event logs that can help detect these types of attacks. Examples of some of the key Event IDs that can help in this detection are provided below. Additional details can be found in our Event Log Analyst Reference and Lateral Movement Analyst Reference documents.

Useful Event IDs for Detection of Credential Theft

Account Logon is the Microsoft term for authentication event records. Logon is the term used to refer to an account being authorized to access 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 hive rather than as a part of Active Directory) are authenticated by the local system where they exist. Account Logon (authentication) events will be logged by the system that performs the authentication. Logon (authorization) events are recorded on the system to which access was granted. These logs are therefore distributed throughout the network, making centralization and aggregation of these logs into a SIEM or similar system critical to their effective use.

Some of the Event IDs of interest in detecting abuse of credentials include:

Account Logon Events

These events will appear on the system performing the authentication function, which will be a domain controller in the case of a domain account and a member server or client in the case of a local account.

Event ID	Description
4768	A Kerberos Ticket Granting Ticket (TGT) was requested. This event will appear on a domain controller. 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 details about the remote host in the case of a remote access. The Keywords field indicates whether the authentication attempt was successful or failed. In the case 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. These codes are also detailed in our Event Log Analyst Reference.
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.
4769	A service ticket was requested by an account for a specified resource. This event will appear on a domain controller. The 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 Kerberos-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 can be useful for detecting attacks against Kerberos. Tickets that use the weaker RC4 algorithm (type 0x17 or 0x18) are easier to crack and favored for Kerberoast attacks, but are uncommon in normal use.
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 case of a remote access attempt. The Keywords field indicates whether the authentication attempt succeeded or failed. In the case of authentication failure, the error code in the event description provides additional details about the failure. These codes are detailed in our Event Log Analyst Reference. 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. For domain accounts, these events will appear on a domain controller. 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 might in and of itself be cause for further investigation in environments where local account usage is not expected.

Logon Events

These events are logged on individual systems as authorization to access their resources is requested, and will therefore be distributed throughout the organization.

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 might be indicative of relay attacks, where an attacker relays an authentication request from 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. If Remote Credential Guard was used to protect the credential, this fact will be recorded in the event log. Use of Restricted Admin mode may impact the Logon Type, causing a Type 3 logon to be recorded.
4625	A failed logon attempt. Large numbers of these throughout a network might 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 might log as Type 3 rather than Type 10, depending on the systems involved.
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. Attackers will frequently switch their user context to facilitate access to different resources, resulting in these log entries. Large numbers of these entries from one account might be an indicator of password brute forcing attacks, such as password sprays. Event ID 4624 with logon Type 9 (explicit use of credential) might also be recorded.
4672	This event ID is recorded when certain privileges associated with privileged, or administrator, access are granted to a logon. As with all Logon events, the event log will be generated by the system being accessed. Since these events are created when privileged accounts are used, they can be a great source of information about where privileged accounts are being leveraged in the environment and assist in the detection of unusual patterns.