# Four Challenges with Monitoring Active Directory Security

blog.netwrix.com/2023/01/20/active-directory-security-event-logs-html

Joe Dibley

With attackers constantly developing new tactics to compromise credentials and data, it is increasingly important to monitor critical systems such as Active Directory (AD) for signs of malicious activity.

Many organizations turn to security information and event management (SIEM) products for help. But while these solutions can be extremely powerful, they ultimately depend on Windows event logs — which are complicated to work with and do not provide the information needed to monitor several key AD attack vectors.

This blog explores four of the top challenges in securing Active Directory and explains the limitation of using event logs to address them.

Handpicked related content:

Active Directory Security Best Practices

## Challenge 1. Monitoring Changes to Group Membership

Active Directory security groups are the primary way that users are granted access to IT resources, including data, machines and applications. As attackers move laterally through your environment, they often add the accounts they have compromised to new security groups to gain additional privileges, so it is vital to track changes to security group membership.

Tracking changes to groups that provide privileged access is especially important. This includes built-in groups such as Domain Admins, Enterprise Admins and Schema Admins, as well as any security groups your organization has created that provide elevated access rights.

### What the Event Log Captures — and Its Limitations

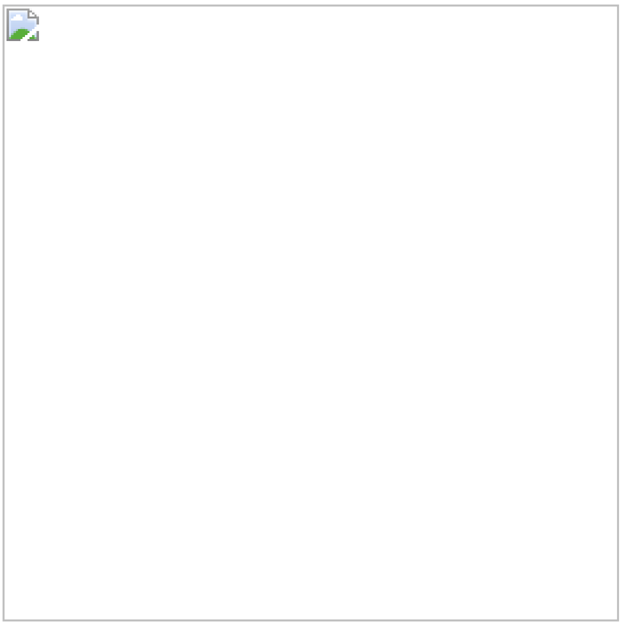
Active Directory does track changes to security groups in the event log. For example, if a user is added to the Domain Admins group, AD will generate an event like the one shown below, which includes the following key details:

- The ID of the user who performed the change
- The DN and class of the object that was changed
- The type of change (in this case, that a member was added)

*Figure 1. Sample event showing that a security group was modified*

However, the event log has several important limitations:

- · **No record of where the change came from** — Event logs do not record the source of a change to group membership. Although a change to the Domain Admins group from a jump server or domain controller (DC) may be normal, a change from a non-administrative workstation or other internet-facing machine can be a telltale sign of an attack. Without details about the source of the change, it’s impossible to alert on changes from abnormal locations.



- · **No monitoring of effective group membership** — Active Directory only logs changes to the direct membership of a group. However, groups can contain other groups as members. Therefore, to truly monitor changes to a group’s membership, you must monitor the group itself and every group nested within it.
- · **Inconsistencies between events** — The logged events will be different depending on how a group was changed. For example, if a user is added to a group using Active Directory Service Interfaces (ADSI), the event log will show one removal event for each existing group member, followed by one event adding back each group member, followed by an event adding the new user; therefore, adding a user to a group with 50 members will generate 101 event log entries. And if the change was made using LDAP, the object’s GUID may be listed instead of its distinguished name. These inconsistencies can cause confusion and misinformation in SIEM products, and make it extremely difficult to build effective rules act on the data collected.

## Challenge 2. Monitoring Changes to Group Policy

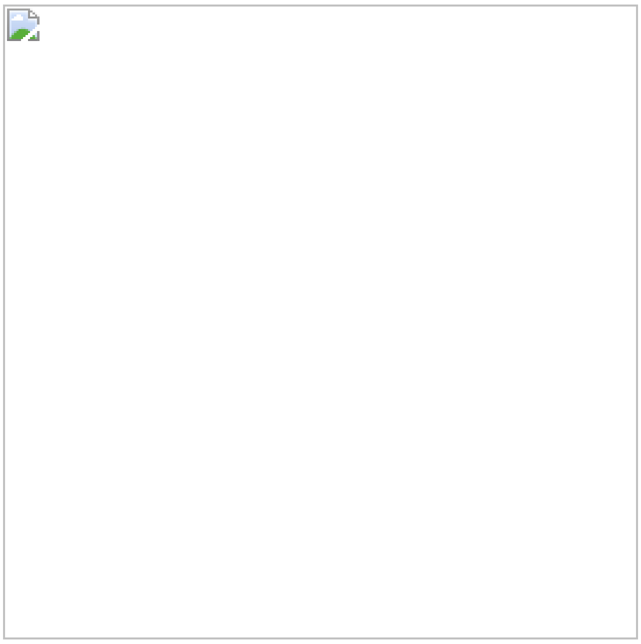
Group Policy settings affect users and computers across the Active Directory domain, including, for example, who has administrative access to systems. A single change to a Group Policy object (GPO) can have severe security impacts or cause production outages, so monitoring these changes is critical.

### What the Event Log Captures — and Its Limitations

When a Group Policy is changed, an event like the one shown in Figure 2 will be logged. This event provides useful information, such as who made the change and the identifier of the GPO.

*Figure 2. Event logged for a change to Group Policy*

However, these events lack the following critical information:



- · **Details of what setting was changed and its before and after values** — GPOs support hundreds of out-of-the-box and custom settings. A change could modify the default browser homepage for users, or provide all users with administrative control of a critical machine. However, the event log does not capturethe altered setting and what it was changed to.
- · **Source of the change** — As with changes to security groups, events recording changes to Group Policy do not indicate where the change came from. Most GPO changes should come from a select few locations, and being able to identify changes that come from abnormal locations is vital to detecting attacks quickly.

## Challenge 3. Monitoring Directory Reads

Another key task in securing your Active Directory is to monitor how user accounts read and enumerate AD objects. Attackers looking to gain a foothold in your network often enumerate critical accounts, groups and servers in order to discover attack paths that lead to escalated privileges and, ultimately, to sensitive data. By monitoring for suspicious read events, you can detect this reconnaissance activity and stop an attack before it is too late.

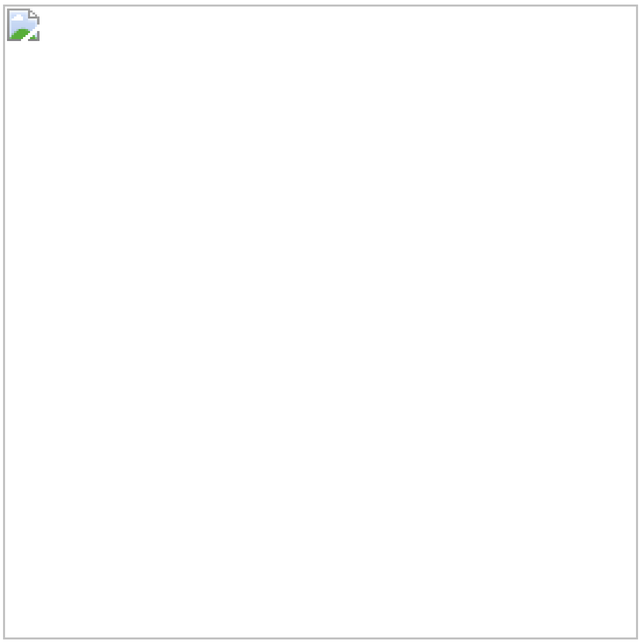
### What the Event Log Captures — and Its Limitations

To help you see who is exploring Active Directory, the event log captures read activity. The event shows details about the user account, the object being read, and the type of operation being performed, as illustrated in Figure 3:

*Figure 3. An event showing that a property of Domain Admins was read*

However, trying to use these events to watch for suspicious activity has multiple drawbacks:

- **Too much noise** — Logging read events results in so much noise in the event log that it is nearly impossible to find any valuable information. Indeed, a single instance of a user looking at a group can generate dozens or even hundreds of events in the log, which makes it nearly impossible to find suspicious activity among all the legitimate events.



- **No record of where the read came from** — Moreover, there is no way to know where the read event originated. As we saw with changes to both security groups and GPOs, knowing which computer a read event came from is crucial information for determining whether it is a harmless read or a malicious act of reconnaissance.
- **Too many access denied events** — It’s also vital to spot users who are attempting to information they do not have the right to see. For instance, Active Directory can store clear-text passwords for administrative accounts in computer attributes, so a user account that tries to read these attributes might be trying to compromise privileged credentials. Unfortunately, every time any account views an object for any purpose, the action generates access failure events for all the attributes they do not have rights to read, even if they did not intend to read those attributes. It’s simply not a viable strategy to try to find truly suspicious failed access events among the ocean of innocent events.
- **No easy way to monitor LDAP queries** — LDAP queries are commonly used to explore Active Directory to discover users, groups and computers. Unfortunately, Microsoft provides no easy way to monitor LDAP queries to see the issued query and where it came from. Because of this issue, even turning on diagnostic-level LDAP monitoring provides little value; in fact, it is not advised by Microsoft, as it will generate a tremendous amount of noise in the event logs.

## Challenge 4. Tracking Authentication Events

With the recent surge of credential-based attacks, monitoring authentication patterns is critical to identify compromised accounts, signs of pass-the-hash and pass-the-ticket attacks, forged Kerberos tickets, or other exploits used to gain privileges and access to sensitive data.

### What the Event Log Captures — and Its Limitations

Active Directory captures events to monitor user logon and authentication activity on domain controllers, member servers and workstations, including those listed in the following table:

| Event ID | Description | Logged to |
|---|---|---|
| 4768 | A Kerberos authentication ticket (TGT) was requested. | Domain controller |
| 4769 | A Kerberos service ticket was requested. | Domain controller |
| 4773 | A Kerberos service ticket request failed. | Domain controller |
| 4776 | The domain controller attempted to validate the credentials for an account. | Domain controller |
| 4771 | Kerberos pre-authentication failed. | Domain controller |
| 4624 | An account successfully logged on. | Server or workstation |
| 4625 | An account failed to log on. | Server or workstation |
| 4634 | An account logged off. | Server or workstation |

While these events capture some useful information, they do not provide an effective way to spot authentication-based attacks because of the following shortcomings:

- **Too much noise** — Events are created every time user logs into any computer, which is typically a tremendous amount of activity. Many other events are created behind the scenes. For example, when a user logs into a member server joined to an AD domain, the server initiates a connection to a DC that retrieves Group Policy information, which results in logon/logoff events appearing in DC's event log. There is no way to disable the logging of normal user login activity without ignoring critical login activity to your domain controllers.
- **No record of logon type on DCs** — The logs do not track the logon type for logon events on DCs, context that is invaluable for determining whether accounts are being used in the appropriate way. For instance, there is no easy way to differentiate between a user logging in through Remote Desktop and a network logon through a mapped network drive; you need to collect logs from every member server and attempting to correlate them with the logs from the DC.
- **Lack of protocol-specific details** — The events are also missing other valuable details. For instance, Kerberos authentication events do not record the ticket life and renewal life timestamps, which are valuable indicators of forged tickets used in the Golden Ticket exploit. Similarly, NTLM logs do not specify the NTLM version that was used, information that is valuable for determining whether you can disable older NTLM versions in favor of more secure protocols.

## How Netwrix Can Help

As we have seen, event logs are inadequate for promptly detecting attacks and responding effectively. For end-to-end protection, consider the Netwrix Active Directory security solution. It will help you:

- Proactively identify security gaps through an in-depth risk assessment.
- Minimize costly downtime and business disruptions.
- Promptly spot even advanced threats in time and respond quickly.

Joe Dibley
Security Researcher at Netwrix and member of the Netwrix Security Research Team. Joe is an expert in Active Directory, Windows, and a wide variety of enterprise software platforms and technologies, Joe researches new security risks, complex attack techniques, and associated mitigations and detections.