Windows File Access Monitoring

In this post, we will dive into how to configure file access auditing on a Windows file server and explore the challenges with interpreting critical access events.

Background

The first step in developing an effective audit strategy is to gain a good understanding of your systems, use cases and business needs, so you can avoid the impact of configuring a wider audit scope than is actually needed. The more audit policy settings you choose and the more files and folders you audit, the more work the file server has to do to log the events, the more storage is required to accommodate the volume of events, and the more data admins have to parse through to understand who is accessing what.

Accordingly, before enabling an audit policy, make sure to:

- Determine where your organization’s most critical data is stored and prioritize which files and folders need to be audited.
- Determine the amount of storage that will be required to support the chosen audit settings.

Handpicked related content:
Windows Audit Policy Best Practices

Configuring File Access Auditing on a Windows File Server

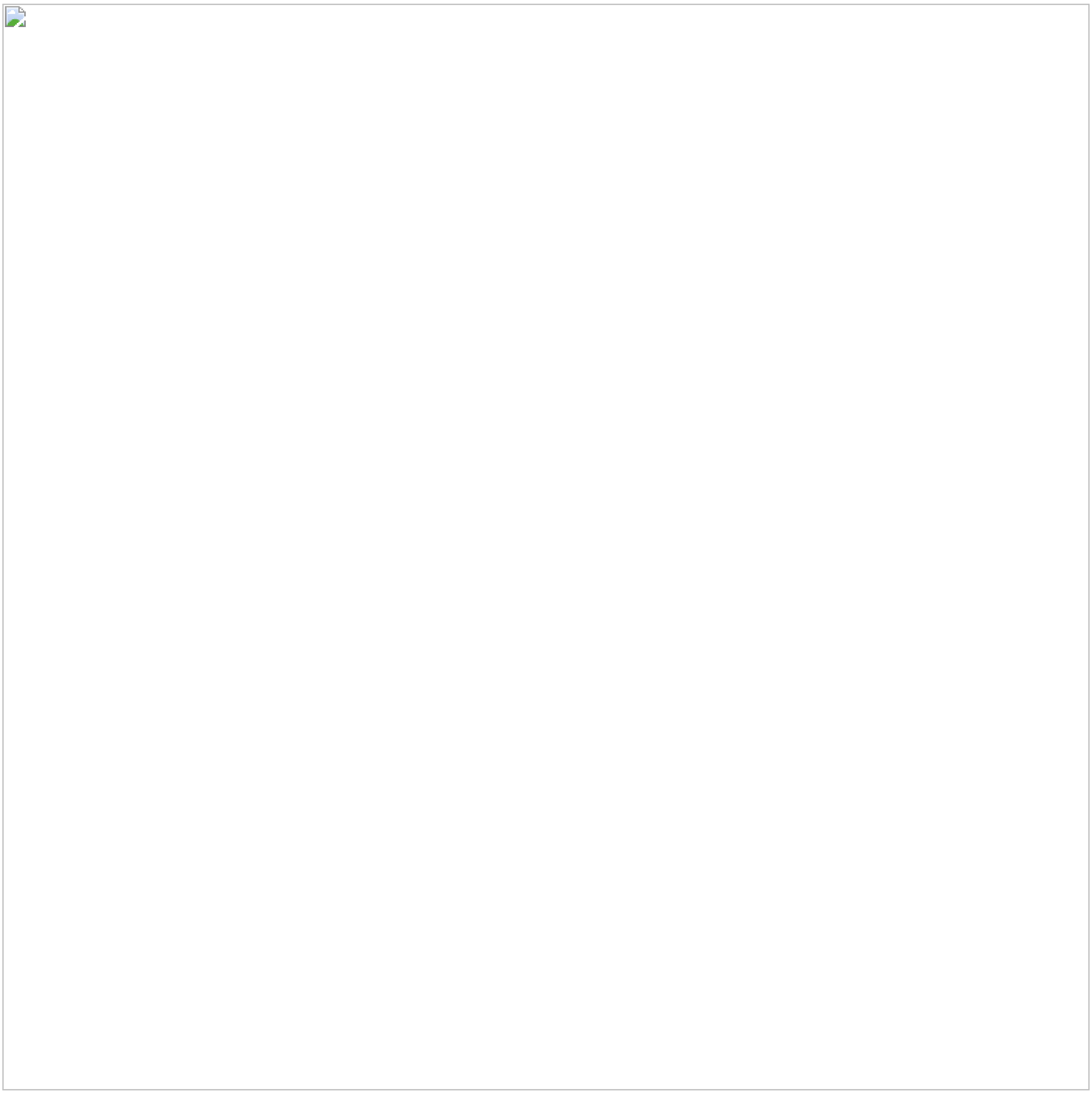
In this blogpost, I will show how to enable an advanced audit policy through Group Policy on a domain controller running Windows Server 2016 R2. (If you have just a single file server, you could use Local Security Policy instead.)

Advanced audit policy allows administrators to be more selective in the types and number of events to be returned than they can with the basic audit policy settings. In particular, when it comes to auditing file access, basic audit policy provides a single setting while advanced policy provides 14 subcategories. In this example, we will enable the following options:

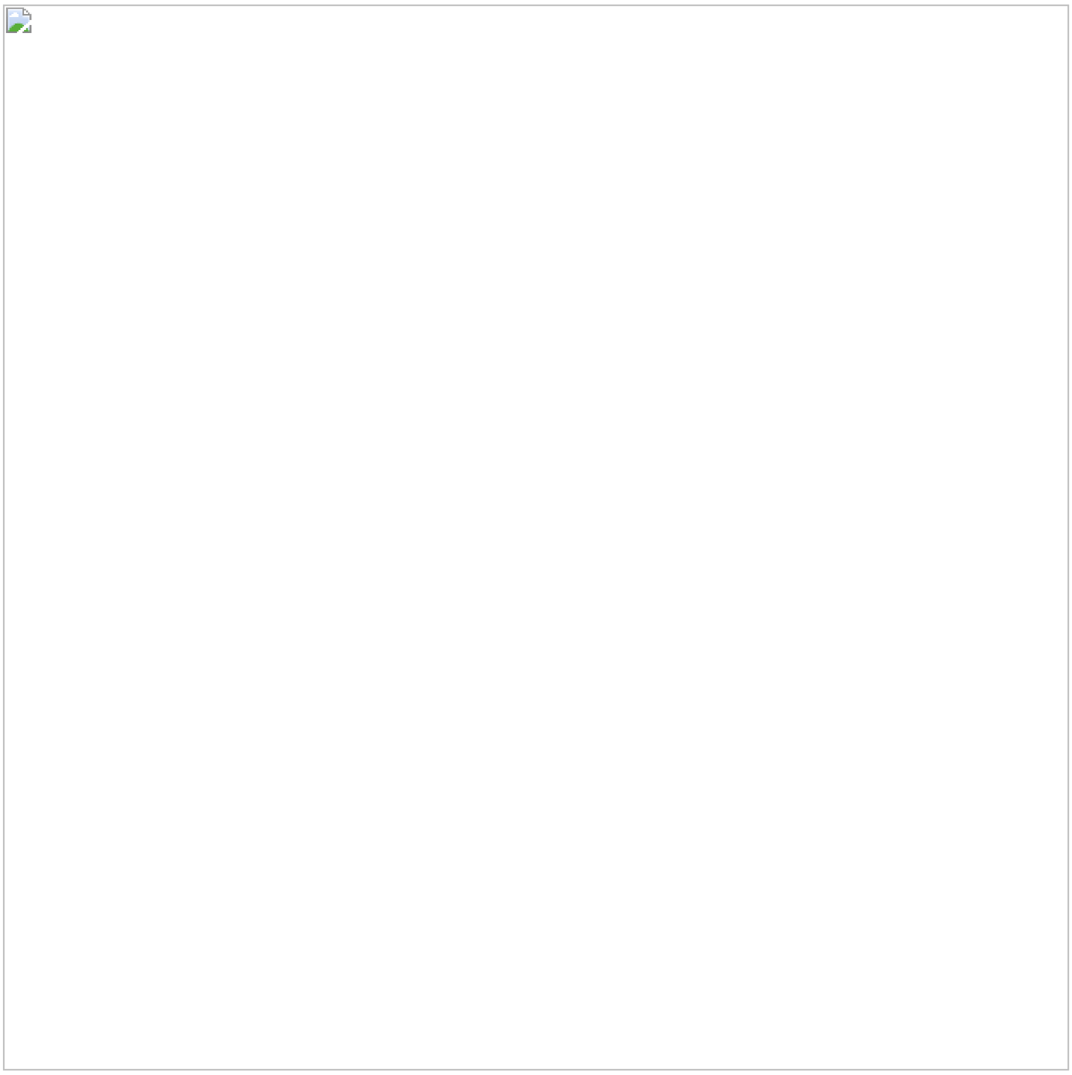
- **Audit File System** — Audits user attempts to access file system objects.
- **Audit Handle Manipulation** — Adds visibility into failed access attempts.
- Create a new Group Policy object (GPO) through Group Policy Management and enter a suitable name.

Right-click the new GPO to launch the Group Policy Management Editor window.

Navigate to **Computer Configuration –> Windows Settings –> Advanced Audit Policy Configuration –> Audit Policies –> Object Access**.

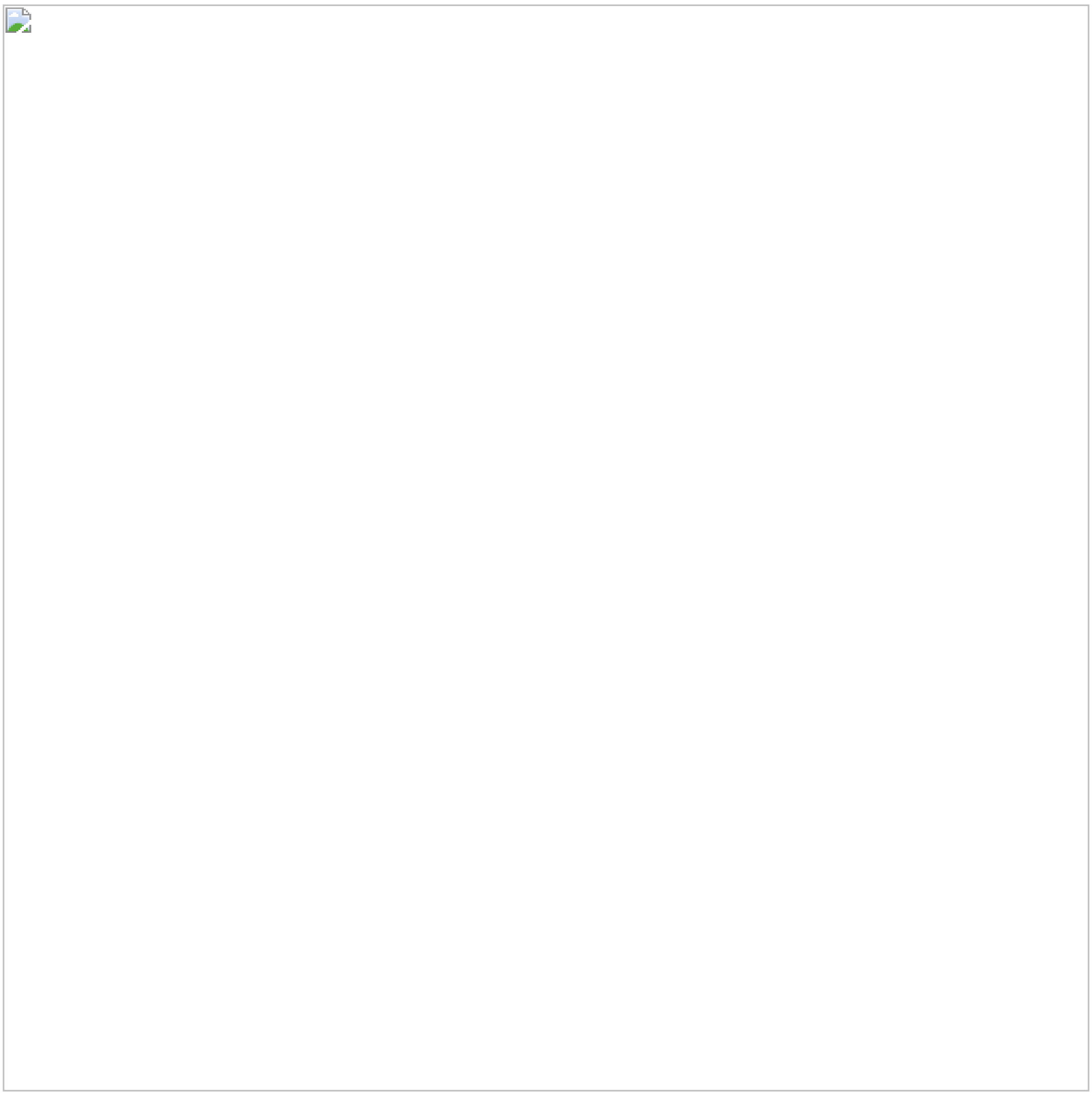


Double-click **Audit File System**. Then select **Configure the following audit events** and choose both **Success** and **Failure.** Save your changes by clicking **Apply** and then click **OK**.



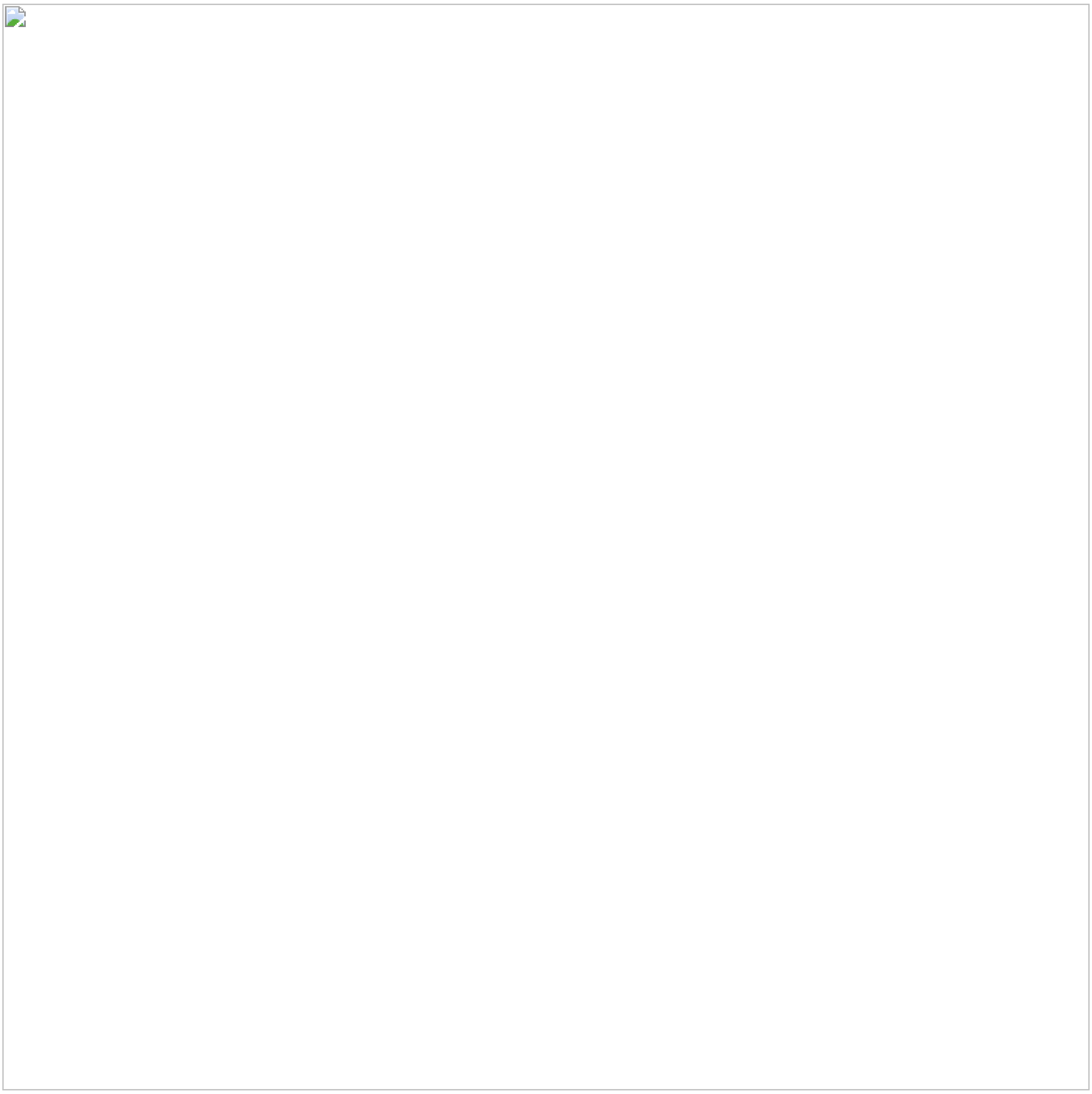
Double-click **Audit Handle Manipulation**. Select **Configure the following audit events** and choose both **Success** and **Failure.** Then click **Apply** and then **OK.**

Now we need to link the new GPO with the OU that contains the file servers. In Group Policy Management, right-click the OU, select **Link an existing GPO…**, select the GPO we created (File System Access Policy) and click **OK** to apply it to the selected OU. Then force the file servers to check the new group policy: Right-click the OU in Group Policy Management again, click **Group Policy Update** and follow the steps in the wizard.



Navigate to the properties of the Security log on the target Windows file server. Then configure the **Maximum log size (KB)** and the action to be taken if the maximum event log size is reached.

Navigate to the security tab of each target folder’s properties –> click **Advanced** –> navigate to the **Auditing** tab –> click **Add** –>configure the auditing settings. Assuming these folders contain your organization’s most critical assets, you will likely want to monitor access events from all users by selecting the “Everyone” principal.



Challenges with File Access Auditing

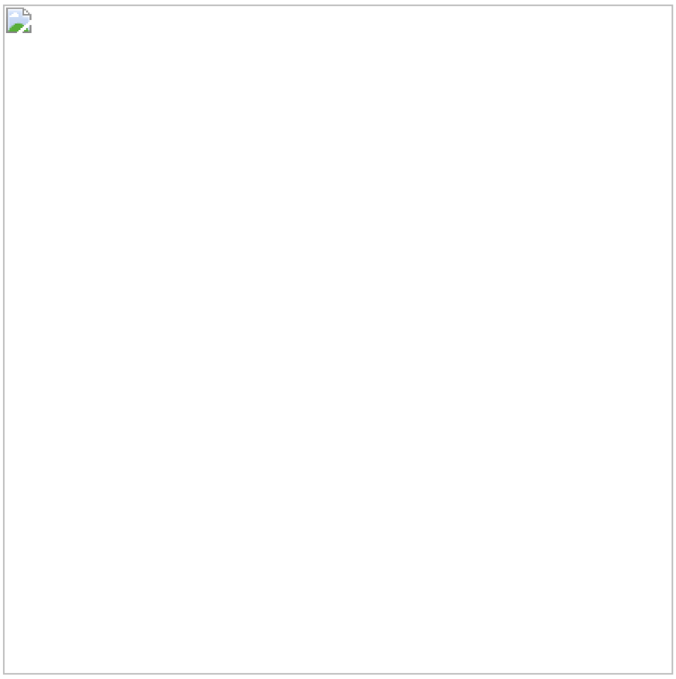
Now let's review some of the top challenges in file access auditing.

High Event Volume

Administrators often struggle to effectively manage the huge volume of audit data that is produced. For example, let's examine the events created by the following common scenario:

1. A user opens a Microsoft Word document on a file share.
2. The user edits the document.
3. The user saves and closes the document.

This ordinary behavior would result in over 200 events being written to the event log, as shown below. Multiply this by the number of times this activity will be performed by users every day, and it's easy to see how quickly the task of monitoring file access becomes unwieldy!



Here are more details on the events that could be returned in our simple scenario:

Event ID	Description	Details
4656	A handle to an object was requested	This is the first event recorded when a user attempts to access a file; it includes the type of access that is being requested.
4658	The handle to an object was closed	This event logs when a handle to an object was closed. It is useful in determining how long a file was open.
4660	An object was deleted	This event is logged when an object was deleted. To find which object it was, you must relate it to a corresponding 4656 event.
4663	An attempt was made to access an object	This event identifies the operation attempted against a file or folder, such as ReadData, WriteData or Delete.
4670	Permissions on an object were changed	This event is logged when permissions to a file or folder were changed. It shows who made the change and the before and after values.

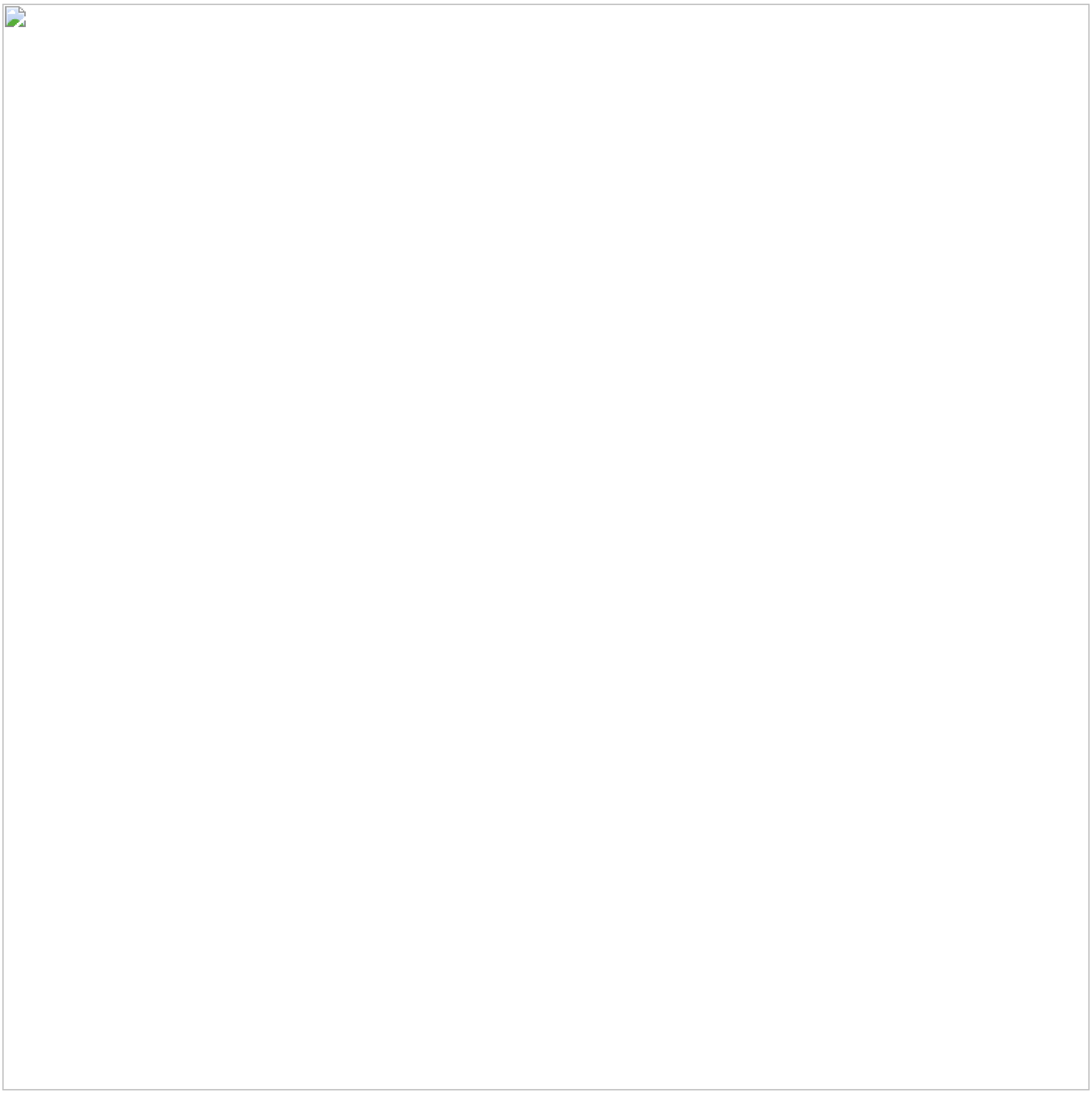
Noise from Temporary Files

The example above returned 230 events — but nearly half of them are logged against temporary files that existed only for a short time.

Microsoft Office uses temporary files for multiple purposes, including to auto-save data during editing, free up memory and prevent data loss. While this provides users with a better experience, it’s a huge headache for the admin tasked with managing the audit trail: In order to make sense of which objects are being accessed, they not only have to correlate several different Event IDs, they also gave to identify and discard the events related to temporary files.

Difficulty Understanding Changes to Permissions

Event 4670 is logged when a permission on a file or folder is changed. It’s vital to monitor these events since they can put sensitive information at risk, such as when a folder permission is added to the Domain Users group. Here is a sample event:



Event 4670 can be difficult to work with for several reasons:

- The security descriptor is represented using Security Descriptor Definition Language (SDDL), so the admin needs to translate it into a readable format.
- Once translated, the admin needs to painstakingly compare the original and new security descriptors in order to identify the changed permission.

Correlating Events to Understand File Movement

Understanding the movement of files from one location to another can be critical, for example, when a user’s documents are missing and need to be found. But moving a file, whether via drag-and-drop or cut-and-paste, generates multiple events, many of which are 4663 events. In order to determine where a file was moved, admins have to manually filter out the noise events and correlate the 4663 events that have a matching Handle ID.

How can Netwrix help?

As we’ve seen, native file access auditing overwhelms admins with so much event data and manual filtering and correlation effort that it is not a viable way to answer crucial questions about file access, permission changes and file movement.

Data access governance software from Netwrix provides an effective and scalable approach to file activity monitoring. Moreover, it will help you reduce the risk of cybersecurity incidents by enabling you to understand who has access to what and strictly limit access to sensitive data. You can:

- Audit activity across your IT ecosystem.
- Reduce access to sensitive data to the required minimum to reduce the risk of insider threats and minimize the damage from ransomware and other attacks.
- Streamline regular privilege attestations by data owners.
- Protect sensitive data whenever it goes with accurate and consistent tagging of content.

FAQ

What is a file activity monitoring tool?

It’s a software solution that tracks activity around sensitive data across your network.

How do you see who has accessed a file?

To track this activity, organizations need to either enable native file access auditing or download a file activity monitoring product that will monitor activity around your files and folders.

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.