Realtime publishers

The Essentials Series: Fundamentals of Effective File Server Security

Auditing File and Folder Access

sponsored by

SCRIPTLOGIC®

by Greg Shields

Copyright Statement

Auditing File and Folder Access

The third rail of effective file server security is all about *assurance*. Assuring that the right users have access to data preserves *availability*. Assuring that everyone else stays out preserves *security*. Assuring that permissions on files and folders are always correctly set preserves *compliance*.

Auditing access to Windows Server 2008 file servers is the primary mechanism through which this assurance is achieved. Auditing enables administrators to verify that security controls put in place are working properly, all the while logging access and modifications to controlled files. Auditing can be enabled via the Windows Explorer GUI, command prompt tools, and Group Policy. Like reporting on access controls, the auditing process is per server and the logging of controlled events is done to the local event log.

A correctly-developed auditing system provides a number of benefits to the organization. It assists in securing the enterprise by determining inappropriate access to files or folders. It provides for the maintenance of a modification history across data, applications, and operating system (OS) configurations. And it creates the necessary documentation for meeting regulatory standards.

Auditing Considerations

The first step in deciding to audit file server access is determining what type of events to audit. For many IT organizations, the selection of auditing categories is often defined by internal security organizations in cooperation with applicable rules of regulatory compliance. Although each compliance regulation is uniquely different in its guidance, all generally require that user and administrator actions are tracked into an auditable database. For some, that database can be your Windows servers' Event Logs.

The first step in this process is to designate a purpose to each audit rule. Auditing rules have several typical purposes. They can assist in securing the enterprise by determining inappropriate access to files or folders. Auditing also allows the maintenance of a modification history outside of any application-specific modification tracking.

This linkage between auditing rules and business goals is critically important, as there can be unintended effects when purposes are not designated to an audit rule. The first of these relates to collected events that do not further the organization's goals. Collecting these events consumes resources, leads to log maintenance issues, and makes event filtering dramatically more difficult. Therefore, discretion is required when developing an audit policy so that only the appropriate types of access tracking are monitored.

The second unintended effect stems from possible legal exposure. Your organization's legal counsel should review your auditing strategy. Their legally-focused review helps to ensure that the auditing purpose covers potential exposure and that the policy is defensible.

Note

In short, your audit policy should collect the minimum amount of auditing information that is necessary to accomplish your business' goals.

Windows Server 2008 provides for auditing on folders as well as individual files. File and folder auditing can monitor access to both simple as well as special permissions as discussed in the first article of this series. For each object, assigned auditing rules can monitor for success and/or failure in exercising the users' permissions.

As with permissions, it is important to remember that auditing configurations are also inheritable. Thus, if extensive auditing is set up and allowed to pass down the settings to a large number of files, a great number of audit entries could be generated.

Configuring Auditing

Auditing must first be globally enabled before setting auditing rules on individual files and folders. Doing this across multiple machines in an environment is most commonly accomplished via Group Policy. Navigate to Computer Configuration | Policies | Windows Settings | Security Settings | Local Policies | Audit Policy, as shown in Figure 1. Here, edit the Audit Object Access policy to allow the tracking of Success and/or Failure events.

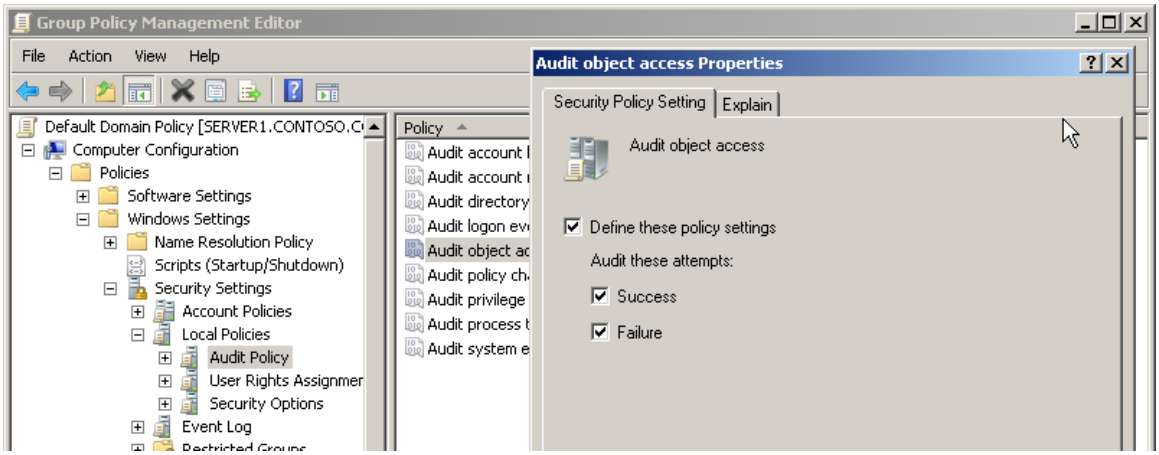

Figure 1: Enabling auditing via Group Policy.

Once the global audit policy is enabled, configuring auditing on individual files and folders can be performed using Group Policy, Windows Explorer, or command-line tools. As with the global policy, leveraging Group Policy for individual file and folder configuration ensures a comprehensive approach. Right-click Computer Configuration | Policies | Windows Settings | Security Settings | File System, and select Add File to add a file or folder to the policy (see Figure 2).

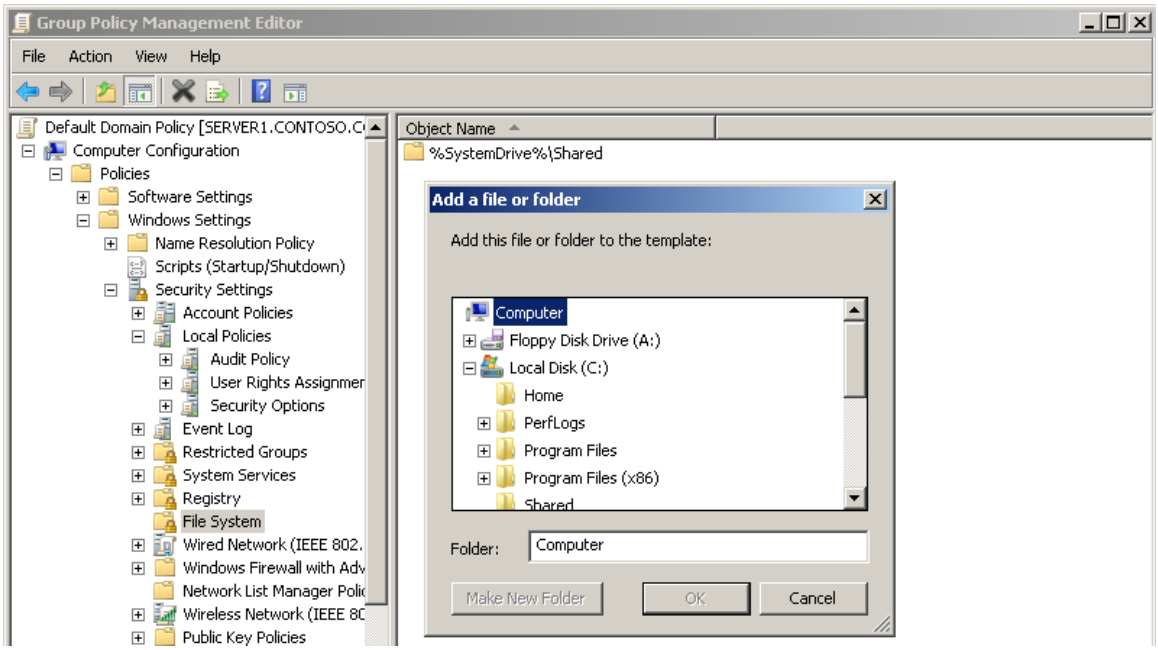

Figure 2: Configuring auditing of NTFS access.

After entering the file path, click Edit Security to bring forward the same Security wizard you're used to seeing when managing permissions directly in the file system. Click Advanced, and select the Auditing tab of the Advanced Security Settings window to configure audit settings using the GUI.

Command-Line Auditing

Audit configuration from the command line is possible using the same Windows PowerShell Get-ACL cmdlet discussed earlier in this series. Get-ACL is used to retrieve the existing audit policy using PowerShell by supplying the -audit command-line switch:

```
Get-ACL c:\Users\Administrator\Tools\ -audit | Select-Object –ExpandProperty Audit
```

To use PowerShell to create an audit rule, the Set-ACL cmdlet is required:

```
$AclToModify = Get-ACL –Path 'c:\Users\Administrator\Tools' –Audit

$NewAudit = New-Object
System.Security.AccessControl.FileSystemAuditRule("MyLocalDomain\gshields","ReadDa
ta","Success")

$ AclToModify.AddAuditRule($NewAudit)

Set-ACL –Path 'c:\Users\Administrator\Tools' –ACLObject $AclToModify
```

PowerShell provides a rich mechanism for scripting the creation of audit rules; however, effectively using it requires familiarity with the .NET Framework classes that manage access to the file system rules.

Mining the Security Event Log

After auditing has been configured, success or failure events will be stored in the Security event log. Event log entries are stored per server, so be conscious of each server's maximum log size and how the event log is configured to react when the log size is reached. Depending on the size of the log file and the number of events, there is a danger of losing audit entries due to log size maximums being reached.

> **Note**
>
> Windows Server 2008 includes a feature called Event Log Forwarding, which allows file servers to centralize event log data onto a single server. This server can be configured to pull the event logs from the other servers or those servers can pass selected events to the central server. More information on Event Log Forwarding can be found at http://technet.microsoft.com/en-us/library/cc748890.aspx.

Gathering log information is only the first step. Effectively mining event log data for meaningful events requires extra effort. With the release of Windows Server 2008, Event Viewer provides several filtering options that limit the data being presented In Figure 3, you can see how a few specific settings can greatly enhance the quality of information viewed from the Security log:

- **Logged.** For a general-purpose file system auditing log, this can be set at *Any time*.
- **Event Level.** All security audit entries will be Informational, so this filter is of relatively little use.
- **Event Logs.** All the entries dealing with auditing are contained in the Security event log.
- **Event Sources.** *Microsoft Windows security auditing* should be selected here.
- **Task Category.** Here, select the File System option.
- **Keywords.** The Keywords option is of little use, as all security audit entries contain the keywords Audit Success or Audit Failure.

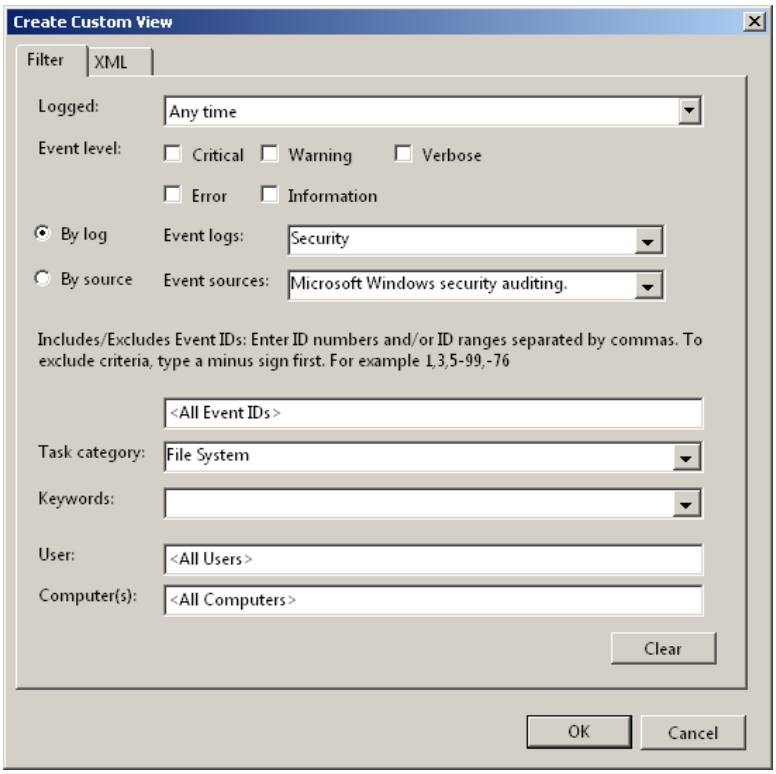

Figure 3: Event Viewer's filtering options.

Once filtered, reporting from the Event Viewer is quite limited. A selection of events can be saved to XML, text, or comma-separated value files, but there is no facility for rich reporting. Additionally, reporting efforts can be hampered by the limitations of event log storage. In the most gracious scenario where all old event logs cannot be maintained in the current view, the events are archived and those archives would need to be searched individually in order to obtain information from them.

Centralized Auditing and Reporting via Third-Party Tools

Auditing is all about assurance, but assuring effective auditing with native tools alone is a challenge. As you can see, configuring auditing via Windows Explorer or Windows PowerShell requires a number of steps and careful coordination to be effective. Each and every file share must be managed as an individual item, which increases the chance for errors or omissions in auditing. Although Group Policy assists this process, the natural dynamics of an IT environment mean that Group Policies must be regularly verified to ensure their policies remain correct over time.

IT environments with large numbers of file shares, large amounts of file storage, or high-security requirements may find that native solutions are insufficient for their needs. To meet regulatory compliance and provide timely security information, you may find the need to turn to third-party toolsets. Their extended capabilities enable the central configuration of an audit plan, central storage of audit data, customizable reporting, alerting in the case of unauthorized access, and enhanced search features that are often necessary as environments scale.