NXPowerLite™
FOR FILE SERVERS

NXPowerLite for File Servers Technical Factsheet

Summary

This document is an introduction to how Neuxpower has designed and built NXPowerLite for File Servers to be a powerful technology, while respecting customer data and taking a safety-first approach to minimize the risk of issues with data integrity.

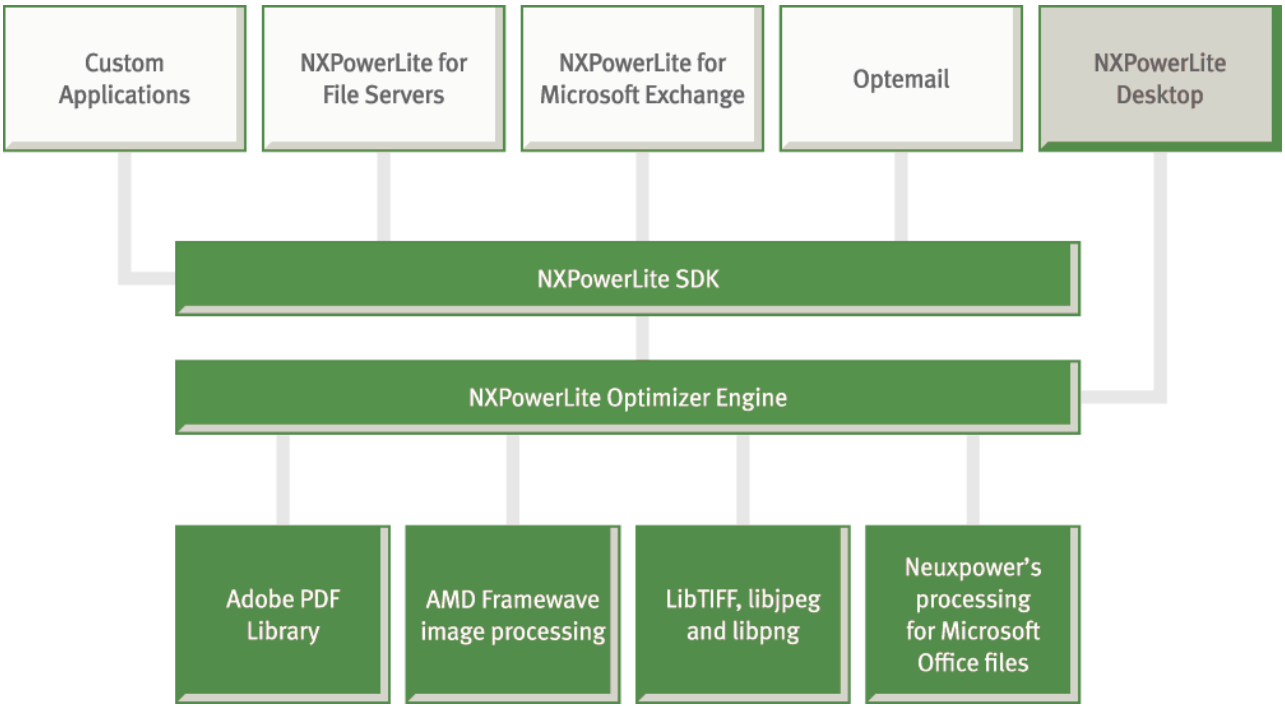
If you require more information that is not included in this document, please contact us and we will be happy to provide you with further detail.

A proven, tried and tested technology

Over 2,000 organizations already rely on NXPowerLite for File Servers to reduce storage, including major institutions such as Nestlé UK, ThyssenKrupp and the Canadian Department of National Defence.

Underlying this software is the NXPowerLite Software Development Kit (SDK), which uses the same optimizer engine (referred to as the Engine) as the NXPowerLite Desktop application for Windows. Between them, these applications have now been used by over 2 million customers worldwide.

As part of the Engine there is a technological baseline of best-of-breed libraries, developed in-house and from established third-party providers, to handle the actual processing of the files:



NXPowerLite™
FOR FILE SERVERS

Safety as standard

A cautious approach to processing files

In use, NXPowerLite for File Servers analyzes files to identify any compatibility issues prior to optimization. Files are copied and optimized separately to the originals, and if any issues are identified prior to or during optimization, or if NXPowerLite fails to reduce the size of the file, the original file is left untouched.

NXPowerLite is a completely transparent process to users with specific features designed to minimise the chance that it will affect their everyday interactions with files. Users can continue to use the same tools and processes without interruption from NXPowerLite. Here's how NXPowerLite achieves this:

- Last-accessed or last-modified dates of files are not altered.
- Files that are in use are skipped
- Files that have recently been modified are skipped by default

During processing, NXPowerLite for File Servers maintains a human-readable rolling log of operations so that it can be clearly identified when a file has been optimized; if not, the reason why is logged, including details of any errors returned by the NXPowerLite SDK. To avoid filling up disk space with log files, this is a 'rolling' log and older log files are periodically removed. Significant events are also logged to the Windows event log.

In extremely rare circumstances, where NXPowerLite for File Servers has identified a system failure where it cannot safely continue, it will log a message to the Windows event log and shut down. For safety reasons, it may need user intervention prior to any restart.

Safety is a priority during development

NXPowerLite products are designed and developed using continually-reviewed best practices to attempt to identify any issues early in a development cycle.

All code changes applied to NXPowerLite products are managed in a versioned source control system, and cross-referenced with an issue tracking system for traceability of changes. Critical to the short development cycles are automated builds on tightly-controlled systems, as well as continuous integration systems running unit and regression testing. Developers identify riskier changes and flag those up for peer review by the team.

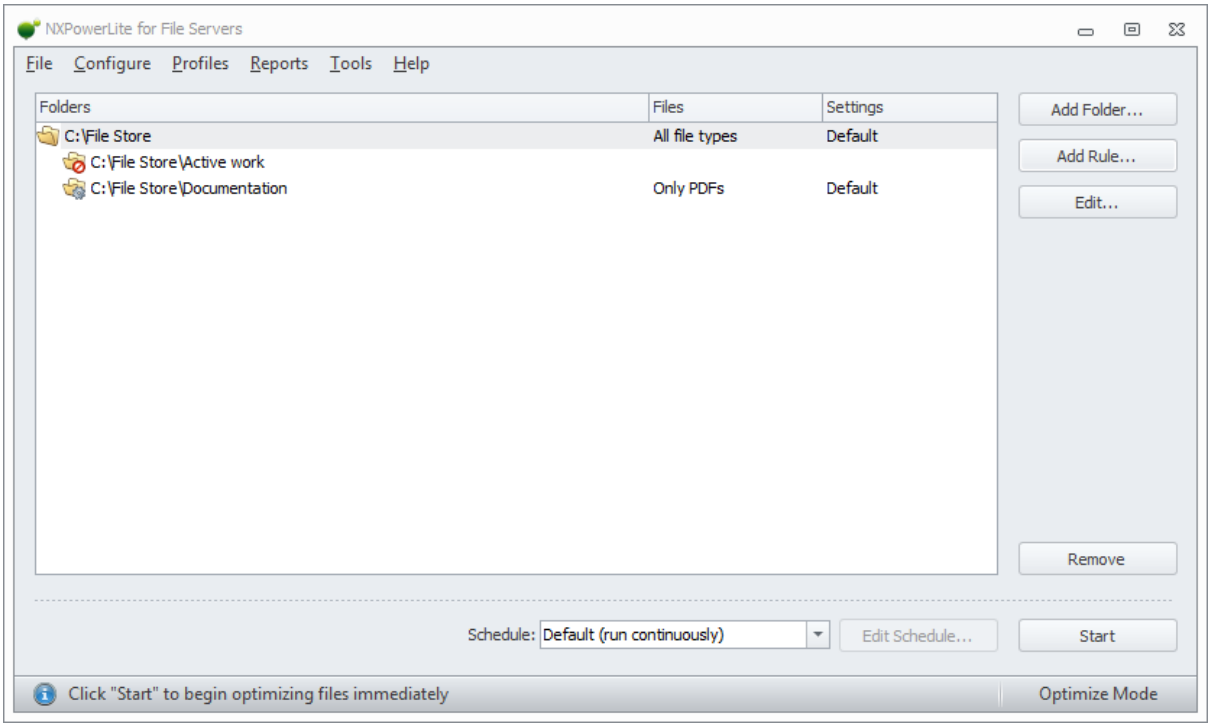
Any changes to the Engine are treated with the utmost caution. Improvements are tried out in publicly released versions of NXPowerLite Desktop first, where any issues with file processing can be quickly identified. When changes have matured, they are then built into the SDK and our server products. On the rare occasions that file integrity issues are identified in shipping products, fixes are released in all products at the earliest possible opportunity.

Release candidate builds are tested within the team, by the product owner, by our partners and, where appropriate, by knowledgeable third parties.

High-level product overview

The product consists of a client-server design, where the client is the NXPowerLite Server Dashboard and the server runs as a Windows service named "NXPowerLite File Server". All file optimization is managed by the service, but it can be controlled either from the Dashboard or—to a lesser extent—by stopping, pausing and starting the service. The high-level roles of each of these components are listed below; see Appendix 2 for a more detailed list of the actual files included in the NXPowerLite for File Servers installation.

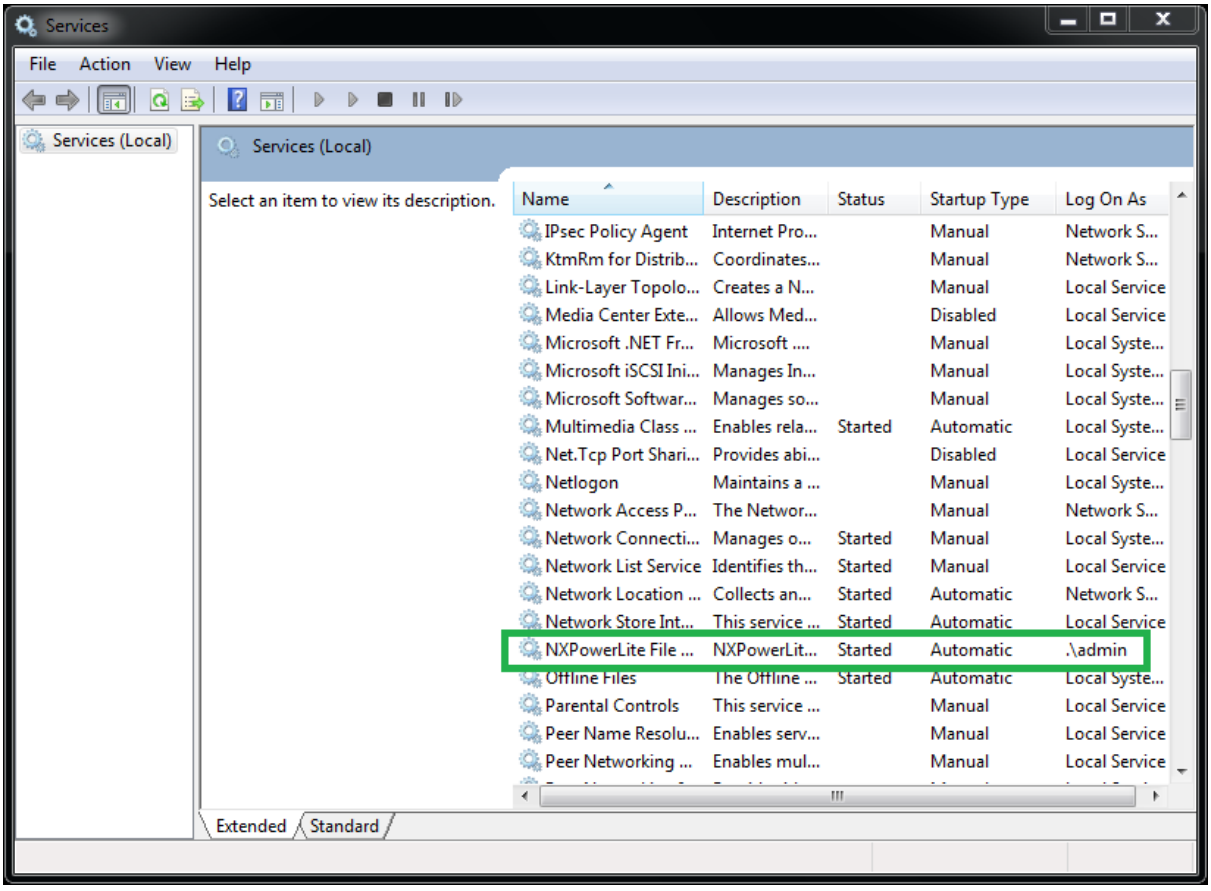
NXPowerLite Server Dashboard



The NXPowerLite Server Dashboard simply acts as a thin client interface to the service element, which handles all configuration and processing. Actions performed via the Dashboard include:

- Starting and stopping evaluation and optimization runs
- Entering the registration code
- Configuring the folders to process
- Managing optimization settings
- Managing file types
- Managing the schedule
- Viewing reports, including the live running report when optimizing
- Viewing help documentation
- Progress notification and user feedback

NXPowerLite File Server Windows Service



The NXPowerLite File Server service manages all aspects of the optimization process, regardless of whether the Dashboard is open or not. This includes, but is not restricted to:

- Starting and stopping operation based on the schedule, as well as the service pause state if the user has chosen to pause NXPowerLite for File Servers

NXPowerLite™
FOR FILE SERVERS

- Accessing optimization locations
- Scanning for optimizable files
- Optimizing at the specified times according to the settings selected by the user
- Replacing files with optimized versions, preserving the attributes of the original file
- Tracking data reduction and generating reports
- General trace logging
- Logging key events and errors to the Windows Application event log

How NXPowerLite works in more detail

Installation & running

NXPowerLite for File Servers is installed following Microsoft installer guidelines to ensure application and configuration files are correctly located and protected. To prove integrity of the installed application, all installed components are digitally signed, either by Neuxpower or where appropriate by the third-party author of the component.

It can be installed on any Windows machine that has access to the local drives, Windows domain and network drives where the data is stored.

As a service, NXPowerLite will shut down and start up with the computer it is running on. You should not need to manually intervene unless you wish to do so, in which case you should set the service to be started manually. Like all services, it can be stopped and started, paused and resumed, and it is designed to respond quickly to messages from the Windows Service Control Manager, to avoid blocking shutdown or reboot.

Aside from files being optimized, NXPowerLite will only write to the following locations:

- Temporary files folder – where all file optimization work takes place
- Program Data folder – all configuration data is written to:
  `%PROGRAMDATA%\Neuxpower\NXPowerLite for File Servers v7`

Controlling access using the service logon account

NXPowerLite for File Servers performs all actions on the file system from within the installed service. As with any service, this has a designated 'logon' account, and it is as this account that optimization takes place. This gives a great deal of reassurance, as it can only ever access files visible to that account, using standard Windows permissions.

NXPowerLite™
FOR FILE SERVERS

By default, the NXPowerLite File Server service is installed to run as the built-in Local System user. It may be that this user cannot see all files selected as locations (this commonly occurs with network share drives). In order to grant access to optimize these locations, you simply need to configure the service to log in as a user with appropriate access rights.

Default settings

NXPowerLite will scan each of the configured locations recursively, producing a list of candidate files. Files of unsupported types, or those that NXPowerLite is not configured to optimize, will be skipped.

By default, the optimization settings are relatively conservative, and could be set to be more aggressive once confidence in the process has been reached. For extra confidence, the following safeguards exist:

- Files that have been modified in the last 7 days are skipped, as these could still be undergoing editing. This 7-day threshold can be configured via the Dashboard.
- Read-only, encrypted and/or password-protected files are skipped; this cannot be overridden.
- File pointers are skipped, e.g. those resulting from journalled backups.
- Files that have already been optimized are skipped. However, if settings are currently more aggressive than those used the last time a file was optimized, or if the file was last optimized by an earlier version of NXPowerLite, re-optimization may be attempted to see if the file can be further reduced.

Additional settings added in version 7:

- Files below 32KB are skipped. These files are highly unlikely to contain the right content for NXPowerLite to be effective.
- Skip '.zip' files over 2GB in size. We have seen instances where NXPowerLite has tried to unzip large zip files containing non-optimizable content, such as a compressed virtual machine. This uses a great deal of computing resources and will fail without reducing files.
- Optimized files are only saved if they reduced by 10KB or more. This is to minimize unnecessary file changes.

NXPowerLite™
FOR FILE SERVERS

File tagging

NXPowerLite embeds a small metadata "tag" into each successfully processed file. The tag describes the version of NXPowerLite that optimized the file and the settings that were used, and enables NXPowerLite to bypass optimizing the file on subsequent runs. Tagging allows NXPowerLite to ensure that it cannot 'over-optimize' images in files, ensuring that the intended original quality is not degraded. Moving or renaming files is completely safe because NXPowerLite will always read this information if it exists.

Processing errors

The architecture of NXPowerLite for File Servers is designed for a high level of availability, whilst ensuring that the risk of data corruption is kept to a minimum. This is achieved by the following safeguards:

- The overall optimization workflow is encapsulated within the NXPowerLite File Server service, which has responsibility for the running state.
- Underneath this, each file is optimized by a new instance of Optimizer.exe. This is a deliberate safeguard to prevent:
    - Any risk of the cross-pollination of data between files
    - Memory leakage as the result of using complex third-party libraries
    - System instability if an unanticipated event occurs due to a rogue file

Any errors are logged to the NXPowerLite for File Servers rolling log, and counters are maintained for categories of failure reasons. In extreme cases, for instance a system failure or lack of disk space, NXPowerLite for File Servers will log the cause and shut down gracefully.

Reporting

NXPowerLite for File Servers has a powerful reporting mechanism.

During evaluation, NXPowerLite generates a report at the end of the evaluation run, showing accurately what space you would have saved if you had been optimizing. Once NXPowerLite has been registered, it continually updates a detailed report showing live information.

As well as information describing the amount of reduction achieved, the report also includes some key information about each run, including:

- The amount of space recovered, broken down by file type
- Configuration details

NXPowerLite
FOR FILE SERVERS

The report serves two purposes: it gives real-time details of how much space NXPowerLite for File Servers has saved, and also gives information as to how that may be improved on subsequent runs; for instance, by identifying inaccessible locations.

You can use the contents of these reports to adjust settings and locations to fully control how your data is optimized.

See below for an example of the live report:

NXPowerLite
by Neuxpower

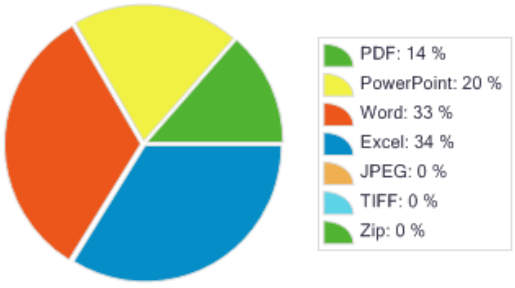
Optimize Report

Version 6.2.15

Report Generated: 17 March 2016 11:06:00

Data Reclaimed Summary

File Types	Reclaimed
PDF	9 MB
PowerPoint	13 MB
Word	22 MB
Excel	23 MB
JPEG	0 MB
TIFF	0 MB
Zip	0 MB
Total Reclaimed	**67**



Last 30 days

Date	Running Time	Files optimized	Reclaimed (MB)
17 March 2016	00:03:36	338	56
08 March 2016	00:00:49	60	11
Total	**00:04:25**	**398**	**67**

Configuration details

Folders	File Types	Optimization Settings
C:\File Store	All file types	Default
C:\File Store\Active work		
C:\File Store\Documentation	Only PDFs	Default

NXPowerLite™
FOR FILE SERVERS

Appendix 1: Using the Windows Audit Facility with NXPowerLite for File Servers

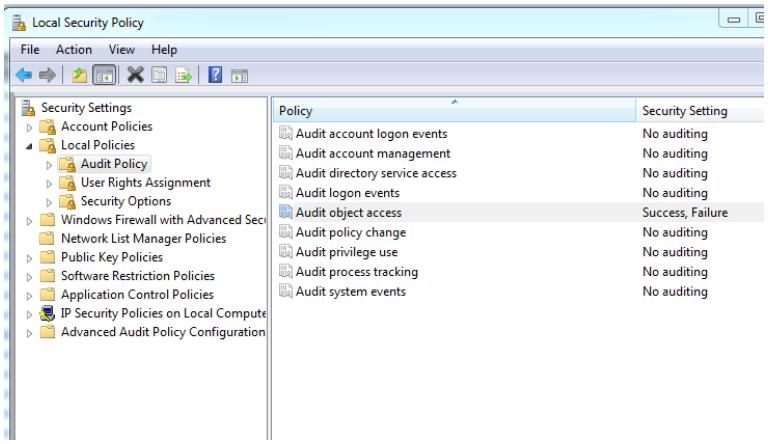
You can use the Windows Audit facility to see in more detail what impact NXPowerLite for File Servers has on your original files. Note that this is quite a detailed and 'noisy' process, and that we would only recommend it for a controlled sample of files.

These instructions are specific to NXPowerLite for File Servers version 7.0.X. They assume you have a single location of unoptimized files.

Setting up auditing

Set up auditing on your machine:

1. Go to the **Local Security Policy** console
2. Expand **Local Policies**, then click on **Audit Policy**
3. Double-click **Audit Object Access** and check both **Success** and **Failure**:

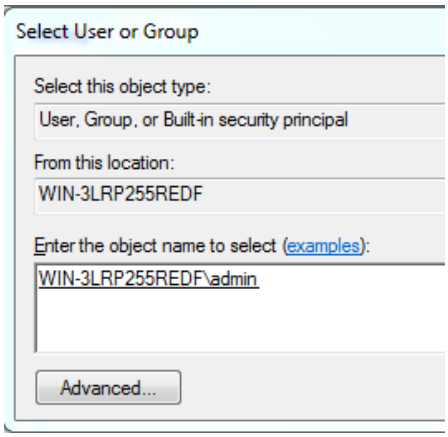


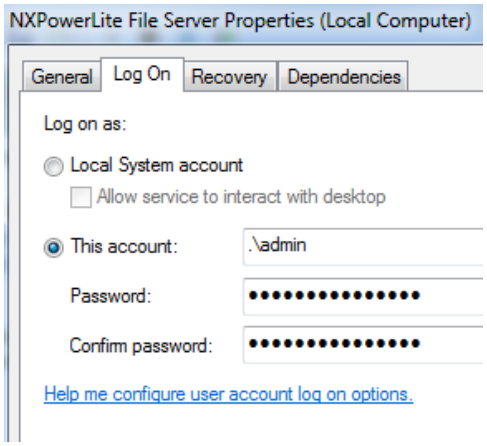
4. Close these windows

Set up auditing of a location to be optimized by NXPowerLite for File Servers:

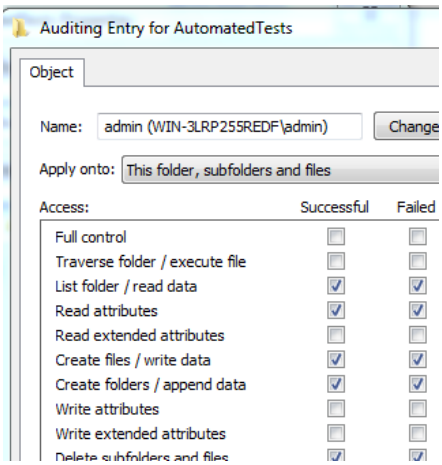
1. Using the NXPowerLite Server Dashboard, setup the folder to be evaluated or optimized as a location
2. In Explorer, right click on this folder and select **Properties**
3. Select the **Security** tab, and click on **Advanced**
4. Select the **Auditing** tab. You may need to grant permission at this point.
5. Click **Edit...** then on the next page **Add...**

6. In the next window, enter the name of the user that the NXPowerLite File Server service is logged on as, e.g. the following should match:



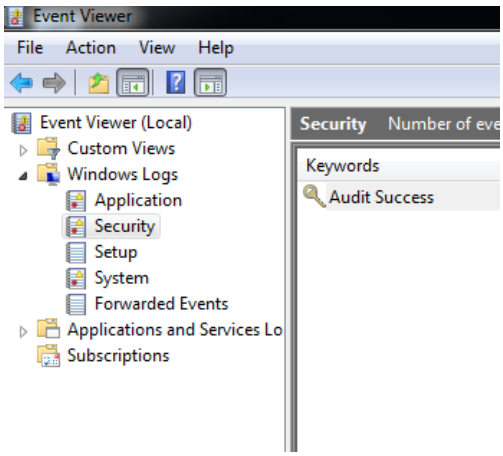


7. In the next Auditing Entry window, turn on both the success and failure checkboxes for the following options:
    a. List folder / read data
    b. Read attributes
    c. Create files / write data
    d. Create folders / append data
    e. Delete subfolders and files
    f. Delete
    g. Read permissions
    h. Change permissions



8. Click OK on all open windows; it is worth closing all Explorer windows on the folder being scanned, otherwise Explorer.exe will generate a lot of noise

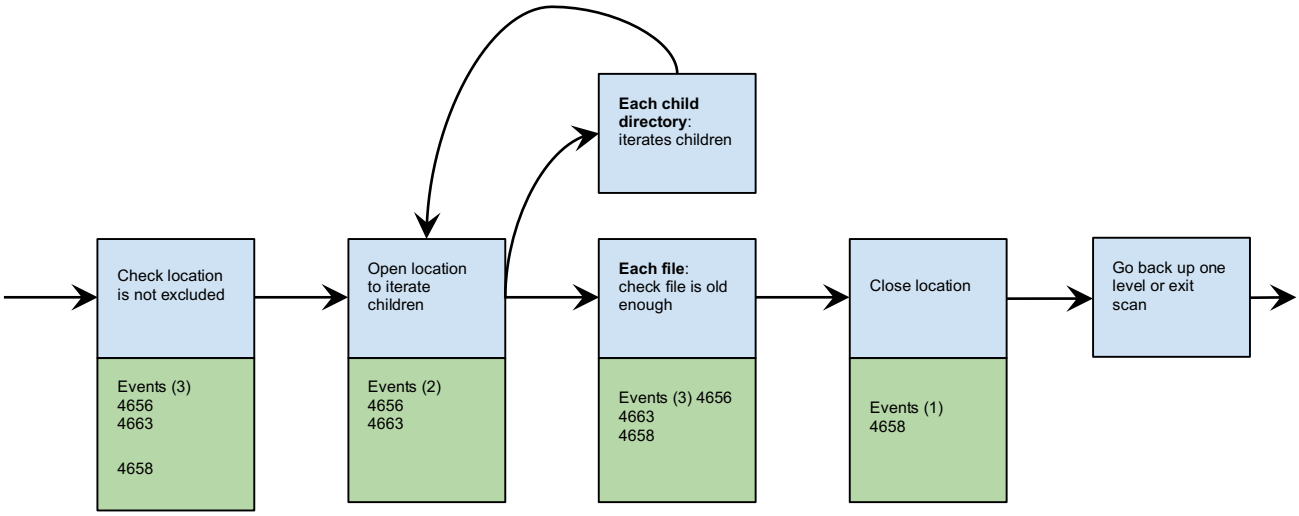
9. Open Event Viewer and navigate to the Security log:



10. Start NXPowerLite for File Servers so that it scans and (if registered) optimizes the files at the location; you will see a sequence of events appear in the Event Log which is summarized below

Scan events

These are the audit events originating from RunCoordinator.exe (the NXPowerLite File Server service) during the scan phase, with an indication of the action that triggered the event.



The audit events are as follows:

4656: Open object

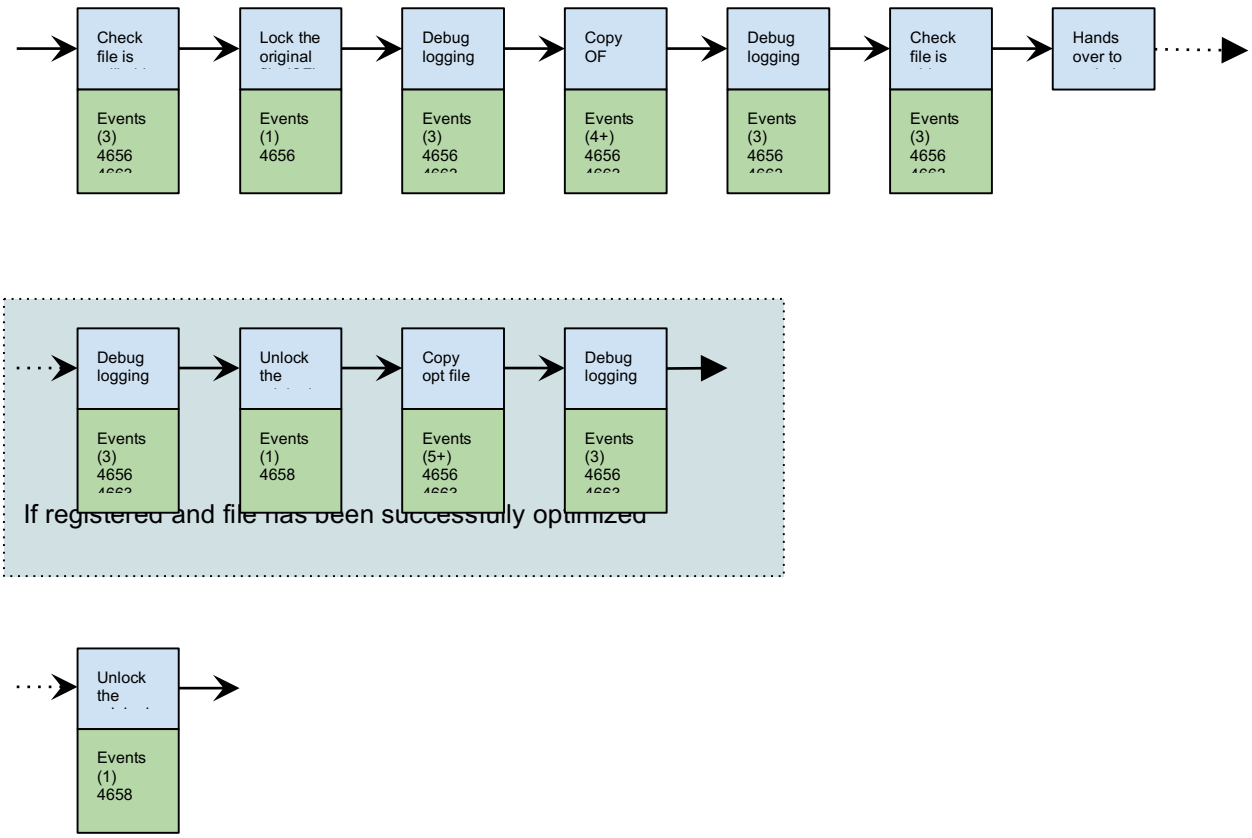
4663: Access object

4658: Close object

NXPowerLite™
FOR FILE SERVERS

Evaluation/Optimization events

Similarly, these are the audit events originating from RunCoordinator.exe during the evaluation or optimization phase, with an indication of the action that triggered the event. Note that additional events would be noted coming from Optimizer.exe if you monitor the temporary folder; these are not included here and would vary from file to file.



The superscript numbers relate to the following:

1 – Copying the original file contents to a temporary file. A handle to the original file is created (4656), the last accessed file tile is reset (4663), then depending on the size of the file there are one or more reads into a fixed-size buffer (4663 for each read). The handle is then closed (4658) although an open handle is retained from an earlier call to lock the file.

2 – Copying the optimized file (which is a temporary file) back to the original file location. A handle to the original file is created (4656), then the file time is read (4663). Depending on the size of the file, there are one or more writes from a fixed-size buffer to the file (4663 per write), then the file time is reset (4663). The handle is then closed (4658).

3 – This only happens if the file handle has previously been closed.

NXPowerLite™ FOR FILE SERVERS

The audit events 4656, 4663 and 4658 are the same as listed under "Scan events".

Appendix 2: NXPowerLite for File Servers Files

This list details the role of each file installed by NXPowerLite for File Servers. The third column indicates whether the components are developed by Neuxpower ('Y' indicates a Neuxpower component, 'N' is a third-party element, although it may have been built by Neuxpower).

Dashboard * Note that these components have no impact on the files being optimized		
FileServer.exe	Main entry point for the NXPowerLite Server Dashboard.	Y
AppModule.InterProcessComm.dll AppModule.NamedPipes.dll	Supports communication between the Dashboard and service components. Built by Neuxpower using externally-sourced code.	Y
DevExpress.*.dll	Third-party user interface and report display elements.	N
LogicNP.FolderView.dll	Third-party user interface element, used to browse for folders to add as locations.	N
Newtonsoft.Json.dll	Formats messages between the Dashboard and service components.	N
QuickScanUtility.exe	Entry point for the Quick Scan application.	Y
ReportViewer.dll	Responsible for formatting and displaying reports.	Y
Documentation subfolder	Contains built-in documentation in a variety of languages	Y
All other subfolders except 'Resource'	Contains resources used for displaying interfaces in different languages. Neuxpower builds all components except DevExpress.*.dll	Y
NXPowerLite File Server Windows Service * This manages all aspects of optimizing the files, and scheduling optimization runs		
RunCoordinator.exe	Main application of the service. It needs to be running for any Dashboard work to take place. The status of the service is not indicative of whether optimization is taking place. It manages identification of files to optimize, and copying the files to and from a temporary location for optimization to take place.	Y
PauseUtility.exe	Manages our internal paused state, and should not be running in normal operation.	Y

NXPowerLite™
FOR FILE SERVERS

Optimizer.exe	The application which actually optimizes the files. Launched for each file that requires optimization.	Y
DL*.dll	Adobe PDF libraries, supplied by DataLogics. They are loaded by nxpdfopt.dll and nxpdftag.dll.	N
nxcapi.dll	Main entry point of the NXPowerLite SDK, loaded by Optimizer.exe.	Y
nxpdfopt.dll nxpdftag.dll	NXPowerLite's PDF optimization engine, loaded by Optimizer.exe.	Y
NXPowerLite*.dll	Message resources for the Windows Event Log.	Y
Resource subfolder	Contains data files required by the Adobe PDF Library.	N
FlxComm.dll FlxCore.dll	Used by the application to communicate with the license portal to check credit allowances. Not used when the software is registered with a perpetual license.	N