Synology

Synology WriteOnce (WORM) White Paper

Last updated: Mar 27, 2025

Table of Contents

Introduction 2
WriteOnce: Synology's WORM solution 2
WriteOnce Fundamentals 4
Enabling WriteOnce for shared folders 4
Compliance mode vs. Enterprise mode 4
WriteOnce File-lock Mechanism 6
Locking files 6
Auto Lock 6
Setting retention periods 7
Configuring lock states 8
Tamper-Proof Clock 9
Utilizing protocol integration 9
WriteOnce Functionality 12
WriteOnce folder management 12
Immutable snapshots and replication 12
WriteOnce behavior in different scenarios 14
Conclusion 16

Introduction

Ensuring the safety of crucial information against unforeseen disasters, unauthorized access, and compliance risks is essential in today's digital landscape, regardless of the storage device or system you're using. This is where WORM technologies may prove useful.

WORM (Write Once, Read Many) is a data storage technology that allows data to be written to a storage device only once and prevents it from being wiped or changed. Any data stored on a WORM-compliant device is **immutable**, meaning that after the data is written to storage, changes can no longer be made to it. This plays a key role when it comes to addressing data security and compliance requirements, as well as protecting against ransomware and other attacks.

Why implement WORM technology?

There are a number of reasons why organizations implement WORM data storage, including:

- **Additional security**: WORM serves as an additional layer of protection, shielding valuable assets like intellectual property and trade secrets from unauthorized access and potential data breaches.
- **Regulatory compliance**: WORM ensures compliance in industries like finance and healthcare, where immutable data storage is mandated for enhanced security and privacy.
- **Active file protection**: WORM acts as a robust defense against unauthorized alterations or accidental changes to critical business documents, ensuring that only authorized personnel can access and modify files.
- **Archive integrity**: WORM preserves the integrity of important historical documents and records, safeguarding them from any modifications while stored.

WriteOnce: Synology's WORM solution

To address the evolving needs of businesses for robust WORM data storage, Synology presents **WriteOnce**, a WORM-based solution for Synology NAS designed to ensure the permanence and integrity of essential files. With this feature, you have the power to configure files on your NAS with WORM functionality, essentially locking them once they are written, and preventing any future alterations or deletions.

By embracing Synology WriteOnce, businesses can gain the confidence to fortify their data against unauthorized access, accidental alterations, and compliance risks, guaranteeing that critical information remains invulnerable to tampering. Whether you're safeguarding sensitive financial records, confidential customer data, or invaluable intellectual property, WriteOnce offers a solution to keep your data secure.

This document introduces some of WriteOnce's features, explains how they work, as well as outlines different implementation strategies and integrated services.

To see if your device is supported, refer to which NAS models support WriteOnce and immutable snapshots.

For tutorials and other information on WriteOnce, refer to Synology's Knowledge Center.

WriteOnce Fundamentals

Enabling WriteOnce for shared folders

When setting up a shared folder in DSM, you have the option to protect the shared folder with **WriteOnce**, Synology's WORM-based file protection solution. This allows you to ensure file immutability within the folder via the file-lock function to safeguard your data for however long you need. WriteOnce settings can be further fine-tuned to tailor protection according to your organization's unique needs.

WriteOnce is implemented at the shared folder level on btrfs file systems and thus makes use of btrfs's **Copy-on-write** mechanism. This means that any modifications to a file will be made on a new data block, keeping the original data block intact and unchanged to be used as reference while tracking changes on the new block. This helps prevent data inconsistencies or partial updates that could occur due to system failures, crashes, or interruptions during write processes.

Furthermore, to ensure data integrity, WriteOnce implements **CRC32 checksum**, a mathematical algorithm that is calculated to create a unique value that represents a WriteOnce file's content. During read operations, the system compares the checksum to check for data discrepancies and prevent potential corruption.

Notes:

- WriteOnce can only be enabled for new shared folders, and once created, they cannot be converted back to a regular shared folder at any time in the future.
- To help organizations comply with certain data regulation standards in regard to data removal, the recycle bin is automatically disabled for WriteOnce shared folders.

Compliance mode vs. Enterprise mode

WriteOnce allows you to choose between two different modes: **Compliance mode** and **Enterprise mode**. Both modes allow you to lock files so that they cannot be deleted or rewritten to provide data permanence. However, these two modes are distinct in terms of what kind of protection they provide, and you should choose the mode that works best for you or your organization's needs.

For more detailed information, refer to the differences between Compliance and Enterprise modes of WriteOnce folders.

Compliance mode

Compliance mode provides a high level of data protection by preventing the deletion of the shared folder by anyone, including administrators. As its name implies, this mode allows businesses to comply with certain regulatory requirements, such as for safeguarding critical financial records or sensitive healthcare information where **data immutability** is required.

Files in WriteOnce folders in this mode can never be altered or modified, and can only be deleted after all of the files in the folder reach the end of their retention periods. To prevent any tampering, operations that could compromise data directly in the WriteOnce shared folder are not permitted, meaning that volume or storage pool deletion from within DSM is not possible.

Enterprise mode

Enterprise mode strikes a balance between data security and administrative control. In this mode, the shared folder or its volume and storage pool can only be deleted by authorized administrators, granting a degree of data management flexibility without compromising data integrity.

This also allows organizations to meet self-regulated regulations and best practices for protecting their data by keeping it invulnerable from any unauthorized changes. With this in mind, it's important to note that while Enterprise mode offers enhanced data protection, it may not fulfill specific regulatory compliance requirements, so its suitability should be considered depending on the scenario.

WriteOnce File-lock Mechanism

Locking files

WriteOnce shared folders offer two ways to lock files, either automatically or manually. With the **Auto Lock** function, you can effortlessly have files automatically lock upon creation and data upload. Alternatively, you also have the option to manually lock files from **File Station** or via command lines after data have been added to the WriteOnce folder.

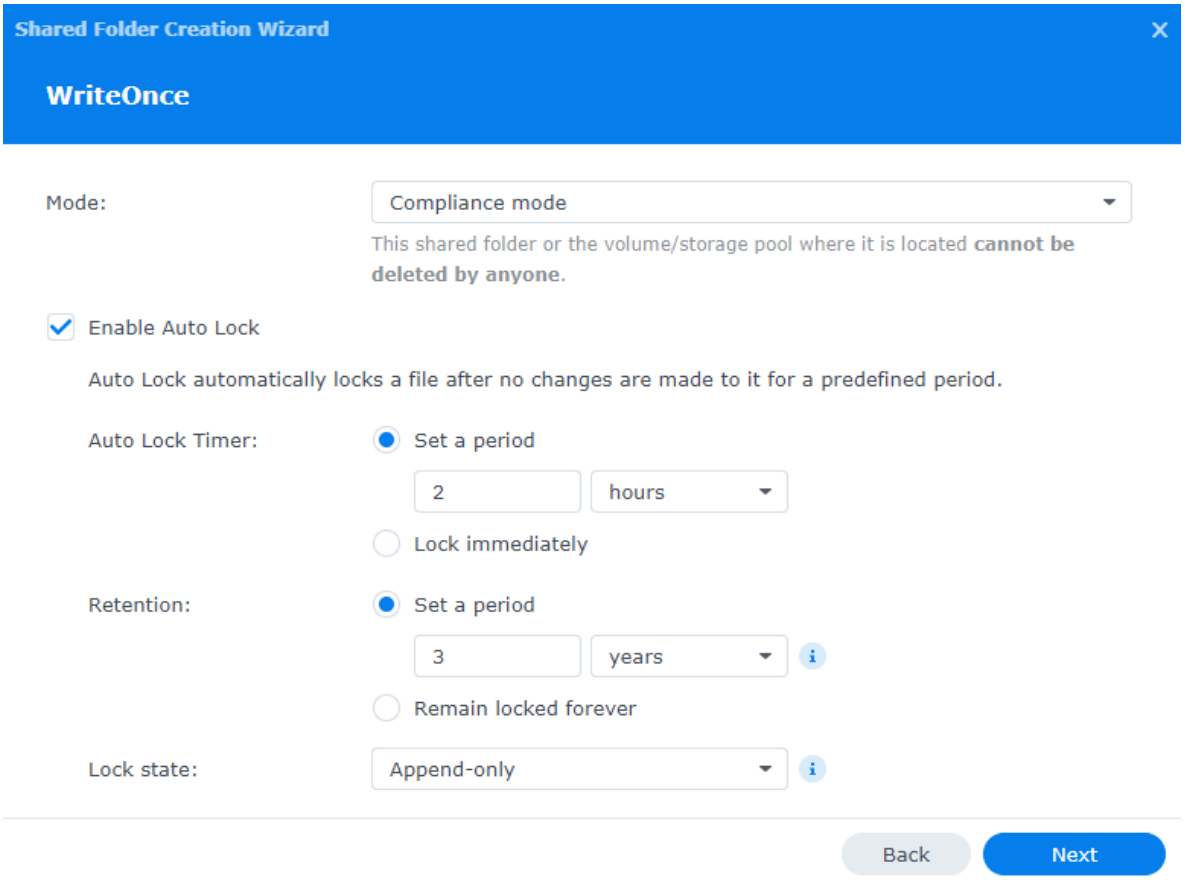
Upon locking a file within the WriteOnce folder, you gain full control over its retention and integrity. You can set a retention period, specifying how long the file should remain locked and what kind of state to lock it in to safeguard against modifications or deletions.

Once you've configured the retention period and locked the file, it remains protected and inaccessible under the WriteOnce status until the set retention period expires.

Auto Lock

During WriteOnce shared folder setup, the **Auto Lock** function can be enabled to automatically lock files after a certain period of inactivity. You can set a specific time period before the file locks or opt for immediate locking, depending on your organization's needs and security requirements. This feature helps streamline the file locking process and overall file management within

WriteOnce shared folders.



Auto Lock settings are specific to each WriteOnce shared folder. You can modify these settings after creating the shared folder, but any changes will only apply to files that are not locked (Open) at that time.

The **Auto Lock Timer** calculates the locking time based on the file's most recent update. If any modifications are made before the designated locking time, the timer will extend and recalculate from the point when you stop editing. This ensures that all changes are accounted for without disrupting file modifications. However, if no changes occur within the predefined period, the file will be locked according to the original Auto Lock Timer setting.

Even if you've configured Auto Lock Timer, you can still manually lock files before the Timer's expiration for immediate protection of vital data whenever necessary. If Auto Lock is not enabled, administrators will need to manually initiate the locking process for each individual file in order to trigger WriteOnce protection.

Setting retention periods

Whether you're configuring Auto Lock for a WriteOnce folder or locking a WriteOnce file manually, you will have to set a retention period to determine how long you want the file to remain locked. This prevents the data from being deleted or tampered with during the specified period of time. No matter when you decide to lock a file, the lock state you choose, or the number of files you're locking, it's essential to understand how the retention period works.

Each file can be locked with a unique retention period, tailored to your organization's needs. This gives you the freedom to set individual retention periods and specify the exact duration that each file will remain under the WriteOnce lock. Once you've configured a retention period and locked the file, it cannot be unlocked until the retention period expires. Therefore, careful consideration must be given when selecting the retention period for your WriteOnce files.

While the retention period can be extended, it cannot be shortened once locked. Consequently, if you choose to lock a file forever, it can never be unlocked at any point in the future. With these factors in mind, it's important to take caution when configuring the retention settings for files in your WriteOnce folders.

Configuring lock states

Right after configuring the retention period, you must choose a lock state to determine how the file is locked. You can see the lock state next to the WriteOnce file, and there are a number of different statuses to make note of.

Before locking a file, it is in an **Open** state, allowing it to be freely modified. Once it is locked, it will be set to either **Immutable** or **Append-only** based on either the WriteOnce shared folder settings or your configurations when locking the file manually. After the retention period expires, the file will transition to the **Expired** state.

Depending on your settings, there may be files in different states simultaneously within the WriteOnce shared folder:

Open

- The file has the same characteristics as files in a regular shared folder and is not protected by WriteOnce.
- Depending on your permission settings, you can edit, delete, and rename the file.

Locked

- The file is protected by WriteOnce for a predetermined retention period. Refer to how to know when the retention period expires for related details.
- Directory and parent directories containing the file cannot be deleted.
- There are two possible lock states:
  - **Immutable**
    - The file cannot be edited, deleted, or renamed.
    - Administrators can manually convert an empty file to Append-only if needed.
  - **Append-only**

    - The file has the same characteristics as Immutable files, except that new content can be added to the end.
    - Append-only files are protected in 64KB units. These units are locked and cannot be modified, except for the last unit remaining writable at any position until it is full. Once the unit is fully written and data extends into the next one, it becomes protected by WriteOnce.
    - Administrators can manually convert the file to Immutable at any time during the retention period.

Expired

- The file's retention period is expired. The file can be deleted, but its original content cannot be edited.
- For Append-only files, you can add new content to the end.
- Administrators can manually change this file to Immutable or Append-only to maintain protection:
    - Expired Immutable files can be relocked as Immutable. If the file size is 0 KB, the file can also be relocked as Append-only.
    - Expired Append-only files can be relocked as Immutable or Append-only.

Tamper-Proof Clock

In a data compliance environment where precise timing is crucial, simply relying on the system clock could put data at risk. Recognizing the susceptibility of the system clock to modification by administrators or malicious actors, Synology has designed a **Tamper-Proof Clock** for WriteOnce. This specialized timekeeping mechanism independently determines the expiration time of the retention period. By not relying on the system clock, arbitrary time adjustments can be prevented to ensure the accuracy of the retention period.

Each WriteOnce shared folder is equipped with its own tamper-proof clock. When a WriteOnce folder is created, the Tamper-Proof Clock initializes and synchronizes with the system clock to serve as a reference for future operations. To ensure consistent and accurate timekeeping, the Tamper-Proof Clock writes its value to disk at least once a day.

However, some events may cause Tamper-Proof Clock operations to pause, resulting in a longer retention period. For more information on these events, refer to this Tamper-Proof Clock article.

Utilizing protocol integration

Other than locking files directly via DSM File Station, WriteOnce also provides integration with various file protocols, such as SMB, NFS, AFP, FTP, and WebDAV, for accessing WriteOnce folders

or locking the files within them.

WriteOnce file management capabilities differ for each protocol. NFS and SMB provide advanced file management capabilities, such as file-locking and changing the lock state, whereas the other protocols primarily focus on error reporting. For this reason, this section will focus on the NFS and SMB protocols.

Before using commands to manage WriteOnce files over NFS or SMB, it's important to note the changes that have been made to the timestamps for WriteOnce files. Refer to the following table for a basic overview:

Timestamp	Meaning for ordinary files	Meaning for WriteOnce files
ctime	Last time changes were made to file metadata	**Start of retention period**
atime	Last time file was accessed	**End of retention period**

For WriteOnce files, the **start of the retention period** is stored on the timestamp for the **last metadata change (ctime)**, and the **end of the retention period** is stored on the **last file access time (atime)**. This means that when setting the retention start date, the file's *ctime* should be changed to the desired start of the retention period before locking. After locking the file, the *ctime* becomes immutable, whereas the *atime* can be extended as needed.

Manually lock files

NFS

The following command can be used to lock files manually via NFS:[1]

```
touch -a -t [atime] [file to be locked]
```

With root access (uid 0), the file can then be locked as read-only (immutable) via the following command:[2]

```
chmod a-w [file to be locked]
```

Notes:

1. If no *atime* (end of retention period) is specified, or if the current time has already surpassed the *atime*, the locking operation will fail.
2. After executing this command, the *ctime* (start of retention period) will be updated to the current system time.

SMB

In a Windows environment, the *atime* can be specified using the Win32 API SetFileTime() or the following command in PowerShell:[1]

```
(Get-Item [file to be locked]).LastAccessTime=("atime")
```

The file can then be locked as read-only (immutable) via the following command in PowerShell or Windows File Explorer:

```
attrib +r [file to be locked]
```

Notes:

1. If no *atime* (end of retention period) is specified, or if the current time has already surpassed the *atime*, the locking operation will fail.
2. These examples highlight the manual file locking operations for the SMB protocol in a Windows environment. Steps and commands may vary depending on the specific implementation, version, and configuration of the protocol.

Manually change lock status to append-only

Files can be changed from **Open** to **Immutable** or **Append-only** to **Immutable**. However, to keep data intended to be immutable protected against any unauthorized changes, only **empty** immutable files are permitted to be changed to **Append-only**. This approach aims to maintain a balance between file management flexibility and the security of immutable data.

The following commands can be used to change empty **Immutable** files to **Append-only**:

NFS

```
chmod a+w [file to be locked]
```

SMB

```
attrib -r [file to be locked]
```

WriteOnce Functionality

WriteOnce folder management

Folder settings

WriteOnce shared folders can be managed by system administrators in **Control Panel** > **Shared Folder**. Configurable settings after the WriteOnce folder has been created include enabling and disabling Auto Lock, as well as configuring the Auto Lock Timer, the retention period, and the default lock state.

File settings

System administrators can manage files within WriteOnce shared folders individually using **File Station**. Administrators can manually lock files, extend their retention periods, or change their lock states between "Append-only" and "Immutable". Refer to the Configuring lock states section for more details.

Notes:

- The mobile version of File Station, **DS file**, does **not** support file management within WriteOnce shared folders. You can use the app to browse WriteOnce files, but cannot perform any other actions on them.

Folder directory access

In both **Compliance** and **Enterprise** mode, the path to a file in a locked state is also locked to protect files from modifications. This means that you cannot delete or rename a directory within a WriteOnce folder. The only exception to this is if the directory is completely empty.

Within the WriteOnce folder, there are some specific directories and sub-directories that are reserved for storing DSM system-specific files. These directories are intentionally exempt from the WriteOnce retention policy to ensure that the system functions smoothly.

Immutable snapshots and replication

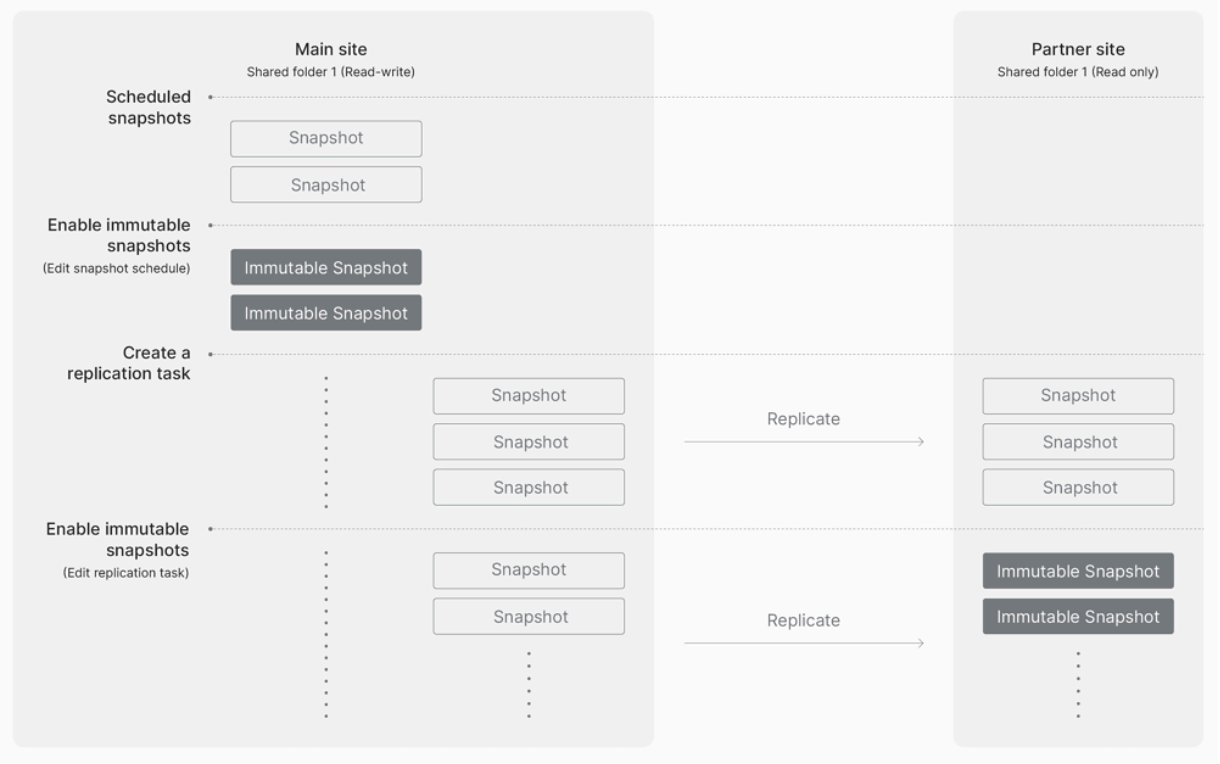
In addition to protecting WriteOnce data from unauthorized access, you may want to strengthen its security even further by using snapshots, point-in-time copies of your data that are stored on

the same volume. This acts as a backup, allowing you to restore files to the exact state saved in the snapshot when needed.

Snapshots of shared folders, including WriteOnce folders, can also be **immutable**. This protects them against malicious or accidental deletions and unauthorized alterations, ensuring the integrity of the snapshot versions.

You can also add a second layer of protection to your data by replicating snapshots to other local volumes, or remote partner sites through Snapshot Replication. Immutability can be enabled for scheduled replications, or manually for each snapshot on the partner site.

When enabling immutable snapshots, a regular snapshot will be created on the main site and then replicated to the partner site as immutable. Consequently, if you do not enable this feature, the snapshots on both the main and partner site will be regular snapshots. This is displayed in the following visual representation.



To learn how to create immutable snapshots locally or replicate immutable snapshots to a partner site, refer to this immutable snapshot article.

Notes:

- To utilize the immutable snapshot feature for replications, both the main site and the partner site must support WriteOnce.
- Snapshot immutability applies solely to the snapshots themselves and can be set for a maximum of 30 days.

WriteOnce behavior in different scenarios

When encountering different operational scenarios, a common question that comes up is how WriteOnce continues to uphold data immutability during these processes. Whether you're migrating data, performing backups, initiating system resets, or wiping data, understanding WriteOnce's behavior within these contexts is essential. With this knowledge, you will be well-equipped to make informed decisions while safeguarding your organization's critical data and maintaining regulatory compliance.

The following table provides an overview of WriteOnce and immutable snapshot behavior during or after certain operations:

<table>
<tr><th>Category</th><th>Function</th><th>Behavior</th><th>WriteOnce Shared Folders</th><th>Immutable Snapshots</th></tr>
<tr><td rowspan="3">Migration</td><td>HDD Migration</td><td>Moving entire drives to another NAS</td><td rowspan="4">WriteOnce properties will remain if supported. If not, volume becomes read-only.</td><td rowspan="4">Immutable snapshots will remain if supported. If not, the volume containing the snapshot becomes read-only.[1]</td></tr>
<tr><td>Migration Assistant</td><td>Migrating volumes and system configs to another NAS</td></tr>
<tr><td>Online Assemble</td><td>Assembling an available pool on another NAS without restarting system</td></tr>
<tr><td rowspan="2">Backup</td><td rowspan="2">Hyper Backup</td><td>Back up and restore entire system</td></tr>
<tr><td>Back up and restore shared folders</td><td>Restored as regular shared folder, WriteOnce is not retained.</td><td>Immutable snapshots are not included in backups.</td></tr>
</table>

Reset	Factory Reset	Reinstallation, data volume reformatted, user data cleared	Compliance mode = Factory reset is not possible. Enterprise mode = WriteOnce folder removal prompt.	Factory reset is not possible.
	Reset button (8 sec)	Reinstallation, system reset, data volume not cleared	WriteOnce properties remain, but files are not deleted.	Immutable snapshots remain and are not deleted.
Data wipe	Secure Erase[2]	Completely wipes data, similar to physical destruction of drive	WriteOnce data cleared: Modification and deletion is prevented on the same system but cannot be prevented on other systems.	Immutable snapshots cleared: Modification and deletion is prevented on the same system but cannot be prevented on other systems.

Notes:

1. To avoid unexpected data loss or complications, only migrate drives to devices that support WriteOnce. If you've already moved drives to an unsupported device, move them back to a WriteOnce-supported device to re-protect the data.
2. To execute Secure Erase, the drive must be deactivated first.

Conclusion

When it comes to meeting regulatory compliance and safeguarding sensitive data against emerging threats, WriteOnce proves to be a valuable asset for organizations. With its seamless integration, secure file-locking mechanism, and reliable tamper-proof clock, WriteOnce reflects Synology's commitment to comprehensive data security solutions.

By understanding WriteOnce's capabilities, configuring settings to match your organization's needs, and utilizing its power across various scenarios, it empowers businesses to confidently navigate the complexities of data protection. Synology's WriteOnce equips organizations to tackle the challenges of an evolving digital landscape, ensuring the integrity and resilience of their critical data.