

STORAGE DEVELOPER CONFERENCE



BY Developers FOR Developers

Virtual Conference
September 28-29, 2021

A SNIA[®] Event

Implementing WORM for backup protection

A story about integrating disk storage
with backup application

Presented by Kornel Jakubczyk, 9LivesData

About me

- MS in CS from the University of Warsaw
- Technical leader,
area of integration with backup applications
- 12 years of experience in storage
- Member of 9LivesData team developing NEC HYDRAstor

About 9LivesData

- Contract-based development company, based in Warsaw, Poland
- Advanced R&D in storage
 - Publishing in the best storage conferences like FAST, SYSTOR, SDC
 - Authors of many US patents
- 13 years of experience, millions of storage-related LOCs
- Most important projects
 - NEC HYDRAsor, scalable secondary storage
 - Starfish Storage (Boston, USA)
very large filesystem metadata and content classification

About NEC HYDRAstor

- Scalable secondary storage with global deduplication
 - Massive linear scalability from 1 to 165 nodes
 - Capacity up to 27PB (350PB effective)
 - High performance (up to 5.9PB/h)
 - Erasure-coding, self-healing
- 5th generation on the market
- Several thousands of nodes sold to enterprises on 4 continents
- 9LivesData designed the architecture and developed most of the code

Write Once Read Many

- Once data gets written, it cannot be modified
- Term initially related to physical mediums
 - E.g. CD-R with infinite protection
 - Or magnetic tapes
- Nowadays can be also software-based
 - Protected data cannot be deleted
 - Data is protected until specified date
 - You cannot shorten this protection

Assumed WORM storage

- Can WORM lock individual files
 - Cannot remove filesystem with protected files
 - Nor change their path
- Can set WORM protection period for each file
- Can extend the protection time,
 - At least while protection has not expired

Motivation for WORM protected backup

- Ransomware highlighted the need of backup itself
 - News of a record \$40 million ransom (CNA)
 - WORM as one of defense means
 - Creation of ransomware resistant backups
- Legal compliance

Solution variants

- Simple: Only WORM on storage
- Universal: WORM integration using general interfaces, no deep integration with backup application
- Integrated: Deep integration of WORM in backup application and storage

Requirements

WORM for backup protection

Solution requirements

- 1) WORM protection time on storage should match backup expiration time
- 2) Automatic setting of WORM protection
 - This rules out Simple Solution, due to the need of configuring WORM for each backup
- 3) Don't leave unprotected files
- 4) Don't leak storage space
- 5) Don't break backup application operations

Universal solution

Idea – Two step solution

Universal solution

- Decouple enabling WORM from setting protection time:
 - Enabling WORM at the end of file write
 - Setting protection time after the backup is done
- Use available backup application interface to get needed metadata
- Can be customized to work with many backup applications

First Step

- Enable WORM at the end of file creation for a short period only
 - Long enough to apply Second Step
- Prevents file alternation and deletion as soon as possible
- Need to know when file creation is finished

Second Step

- Set the right protection period off-line,
 - When the backup is finished
 - Performed by a batch script, run by e.g. cron
- Use the backup application's interface to learn the list of files and their expiration time
 - At this step backup is finished, so its metadata should be already available
- Use the storage interface to extend the WORM protection of First Step

When the Universal solution can be used?

Universal solution

Backup application requirements (1/4)

- Provides an access to the backup database, that exposes
 - Backup expiration time
 - Which files belong to a given backup

Backup application requirements (2/4)

- Files are not modified off-line, after backup is finished
- Files belong to one backup only
 - Files do not contain references to other backup's files
- Relations between backups are reflected by their expiration time

- Backup application requirements (3/4)
- Access pattern of backup application should be compatible with WORM on storage
- Potentially incompatible operations on backup files:
 - a) Appends
 - b) Metadata updates
 - c) Overwrites
 - d) Deletes on demand
 - e) Backup copy creations

Backup application requirements (4/4)

- Need to know when the file creation is completed

A) Can be learned by observing the access pattern

- e.g. there can be a fsync at the end

B) There can be metadata written to mark the file as finished

C) Or special interface

- Explicitly guarantees no further updates

How things turned out

Universal solution

Implementation of Second Step can be hard

- A) Need to deal with information that does not normally concern the storage
- B) Need to consider the wide functionality of backup app
- C) Can have many corner cases, like special values
 - Or temporary values as batch script execution can race with normal operations
 - Like copy creation
- D) Puts additional load on the backup database

High maintenance

- Have extensive regression tests
 - To handle upcoming backup application releases
- Need to adapt to the changes in backup application

Complex failure handling

- First Step failure

- Storage has an option to fail some operation
- Potentially causing further backup failure

- Backup failure

- Second Step will not be executed
- The leftover files can be deleted after the protection period of First Step expires
 - This can lead to false-positive clean-up failures

- Second Step failure

- Risk of not extending WORM protection on time
- Need to monitor the script execution

Limitations

1) Functionality not compatible with this solution

- Backup retention related to storage capacity rather than time
- Extending the backup retention until additional operations complete

2) Cases may need manual handling

- Manually extending the retention period of a backup

3) Operations which can be executed by the backup application, but are not reflected on the storage

- Removing image before it is expired
- Shortening backup retention

Pros and cons

- **Pros:**

- Fulfils the requirements, generally
- Can be implemented for many backup applications
- Minimises the window when files are not protected

- **Cons:**

- Complexity
- High maintenance
- Can generate false-positive failures
- Needs customization

Design decisions

- Implement only Second Step
 - But initial protection will be compromised
- Scripts called at the end of backup
 - Can be run while backup metadata is not available
 - It may not be possible to fail backup from the script

Integrated solution

Cooperation of backup application and storage

Integrated Solution

- WORM feature added to the OpenStorage option of Veritas' NetBackup
- And implemented by storage vendors
- WORM related information is available in NetBackup's interface
- NetBackup is aware of WORM locking
 - You cannot delete worm protected file, nor shorten its retention
- Deep integration resolves the previous limitations
 - NetBackup can handle failure to apply worm locking

Universal vs Integrated Solution

	Where implemented	Limitations	Support for multiple backup applications	Maintenance	Administration
Universal	Middleware	More	Possible, after customization	Harder	Harder
Integrated	Backup application and storage vendors	Less	Focused on one	Easier	Easier

Other approach to WORM protected backup

- Use a publicly available interface that supports WORM
 - E.g. S3 interface (Object Lock)
- Can provide a wide range of supported combinations
- Applicable when:
 - Backup application and storage implement this interface
 - Interface can support all the necessary features

Conclusions

- WORM supported by storage only not enough for backups
 - Due to manual setting of retention info in WORM storage
- WORM automation necessary
- The best is deep integration of WORM in the backup application
 - Requires backup application vendor support
 - Works for one specific backup application
- In the absence of deep integration generic solution implemented by middleware possible
 - Works with assumptions of specific backup app behaviour/API availability
 - Customizable to multiple backup apps

Acknowledgments

- Thanks to my colleagues from 9LivesData
- Thanks to NEC development team members

Thank you all for your attention!

- You can contact me at jakubczyk@9livesdata.com
- 9livesdata will be happy to help you with your storage development needs
- Please contact projects@9livesdata.com



Please take a moment to rate this session.

Your feedback is important to us.