TECHNICAL WHITE PAPER

How It Works: Protecting MongoDB with Rubrik

Rubrik Technical Marketing

November 2023
RWP-0631

Table of Contents

ABSTRACT 3

AUDIENCE 3

RECENT HISTORY OF BACKUP & RECOVERY SOLUTIONS 3

WHAT IS RUBRIK FOR NATIVE MONGODB PROTECTION? 3

TECHNICAL OVERVIEW 4

DAY-ZERO SETUP 5

- RBS – Rubrik Backup Service
- Adding a MongoDB Source
- Discovery

DATA PROTECTION 8

- SLA Policy Engine
- SLA Domain Assignment

BACKUP 16

- Oplog Backups
- Full backup
- Forever Incremental backups
- On-Demand Snapshot

RECOVERY 19

SOLUTION ARCHITECTURE 20

- Centralized Management
- Reporting
- Protection Against Ransomware

KEY BENEFITS 24

CONCLUSION 24

NEXT STEPS 25

VERSION HISTORY 25

ABSTRACT

MongoDB is a versatile, NoSQL database management system known for handling large volumes of data. It is used across various industries and applications to store, retrieve, and manage data efficiently. MongoDB offers flexibility and scalability for web applications, mobile apps, e-commerce platforms, or big data analytics. It is a popular choice for developers and organizations seeking a robust database solution. As a result, protecting data stored in MongoDB is critical for IT operations to help ensure business continuity, compliance requirements, ransomware threats, and disaster recovery. Rubrik integrates with MongoDB to offer a policy-driven approach to database protection, providing secure management of the backup data lifecycle. Rubrik also brings data resilience and remediation to your MongoDB databases, allowing you to recover and restore business operations rapidly.

AUDIENCE

This white paper aims to assist Backup and Database Administrators (DBAs) in comprehending the Rubrik implementation to protect MongoDB.

RECENT HISTORY OF BACKUP & RECOVERY SOLUTIONS

Typically, organizations have more than one database, such as Oracle, Db2, SAP HANA, and MongoDB, in their environment to meet all their application requirements. The next logical step for their IT team is to enforce data and application protection and have recovery options ready should they become necessary because of natural disasters, infrastructure outages or failures, user errors, or cyber-attacks. A comprehensive data backup and recovery strategy is essential to minimize any downtime and potential data loss.

Oftentimes, native tools lack comprehensive functionality and have limitations. These tools:

- Do not protect against cyber/ransomware attacks
- Require full backups every time, leading to inefficiency across compute and storage
- Increase operational complexity because of silos of different databases
- Do not offer a single automated SaaS platform to create and manage data protection across on-premises, public cloud, and hybrid infrastructure
- Are either not designed for data protection or not scalable solutions

WHAT IS RUBRIK FOR NATIVE MONGODB PROTECTION?

Rubrik Security Cloud is a Software-as-a-Service (SaaS) platform that keeps your data secure and quickly recovers it wherever it lives—across the enterprise, in the cloud, and in SaaS applications. Rubrik offers many data protection and security solutions, such as Enterprise Data Protection for your databases, VMs, physical machines, etc., with air-gapped, immutable, access-controlled backups. Visit the Rubrik Security Cloud website for more details.

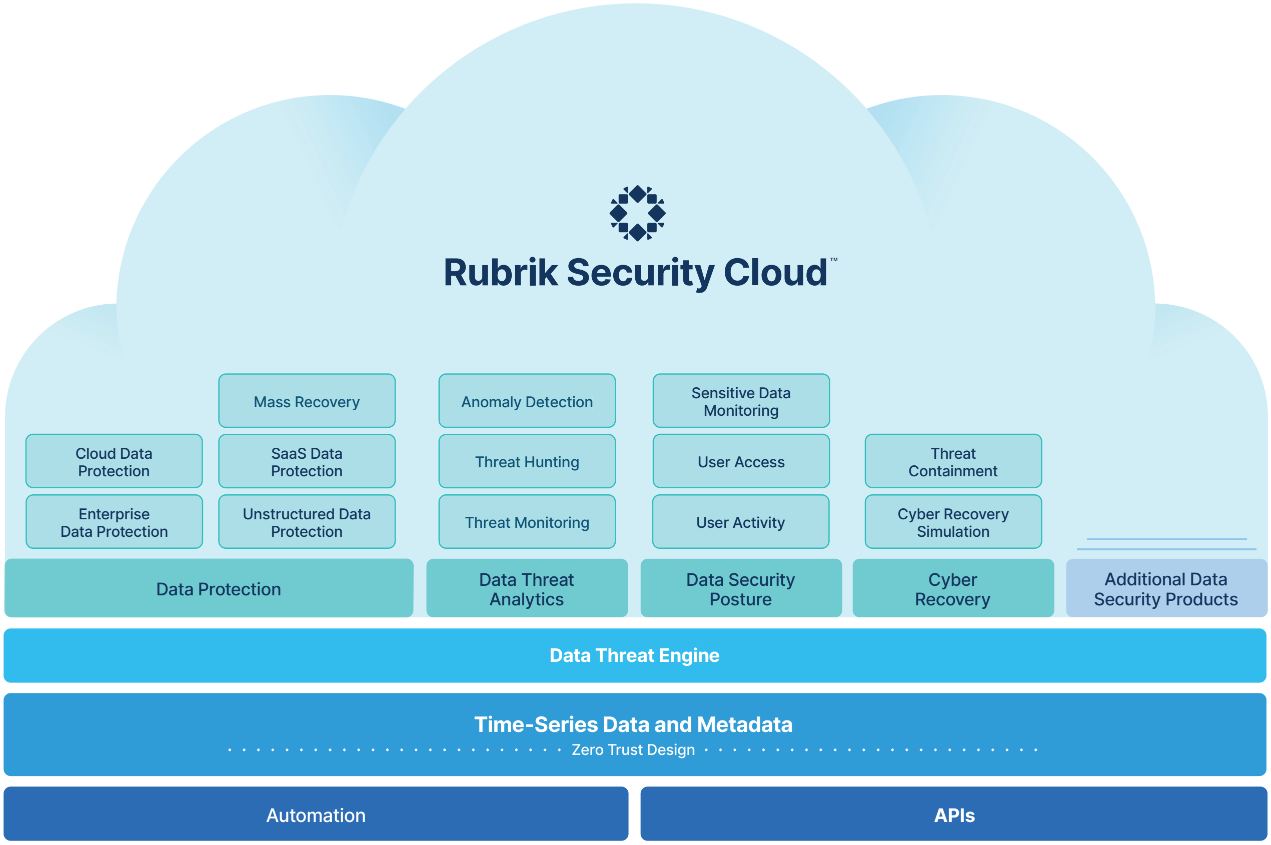


Figure 1 – RSC Overview

Rubrik provides MongoDB data protection and data management for on-premises infrastructure. This document focuses on the data protection solution for on-premises infrastructure and IaaS systems.

TECHNICAL OVERVIEW

When customers configure a MongoDB environment, they can create a MongoDB cluster using either sharding or a replica set setup. Replica sets are commonly employed to ensure high availability, enabling data copies to be distributed across multiple nodes within the cluster.

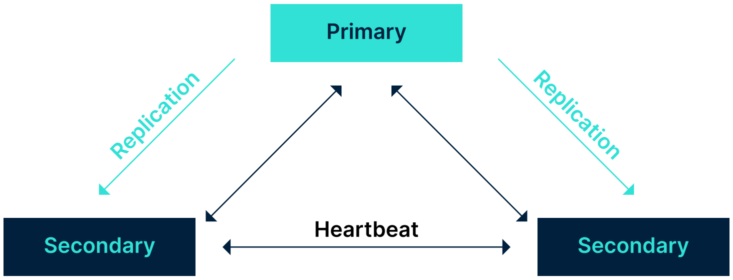


Figure 2 – MongoDB Replica set config

Alternatively, customers can opt for horizontal scaling through sharding, a strategy mainly suitable for larger environments that demand high throughput beyond the capacity of a single system.

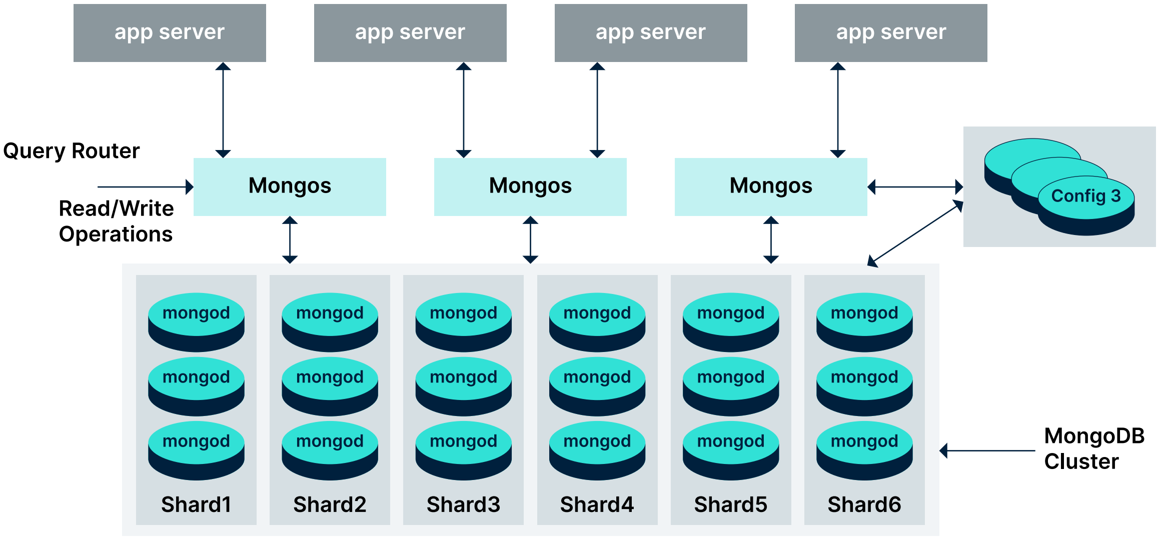


Figure 3 – MongoDB sharded config

In Rubrik, a MongoDB Source serves as an identifier for your MongoDB cluster, which typically comprises multiple nodes. To ensure proper backup and management, each node should install the Rubrik Backup Service. If the primary node fails, then Rubrik can continue backup from the secondary nodes. The hierarchical structure is outlined as follows:

- A MongoDB Source is a way to identify a MongoDB cluster in RSC
- A MongoDB Source consists of a list of databases
- A MongoDB Database consists of a list of collections
- A MongoDB Collection contains a list of documents
- A MongoDB Document is a key value pair list array or nested document

MongoDB stores data records as BSON documents. BSON is a binary representation of JSON documents.

DAY-ZERO SETUP

Before Rubrik starts protecting your data, certain day-zero setup prerequisites must be addressed. Let's examine them.

RBS – RUBRIK BACKUP SERVICE

All communication between Rubrik and a MongoDB Source, including its nodes, is channeled through the Rubrik Backup Service (RBS). RBS is a lightweight connector service running on the Linux host, ensuring secure communication between the Rubrik Cluster and the Linux host. For each MongoDB node, it is necessary to have the RBS host installed via the .deb/.rpm package and subsequently added to Rubrik using the RSC UI.

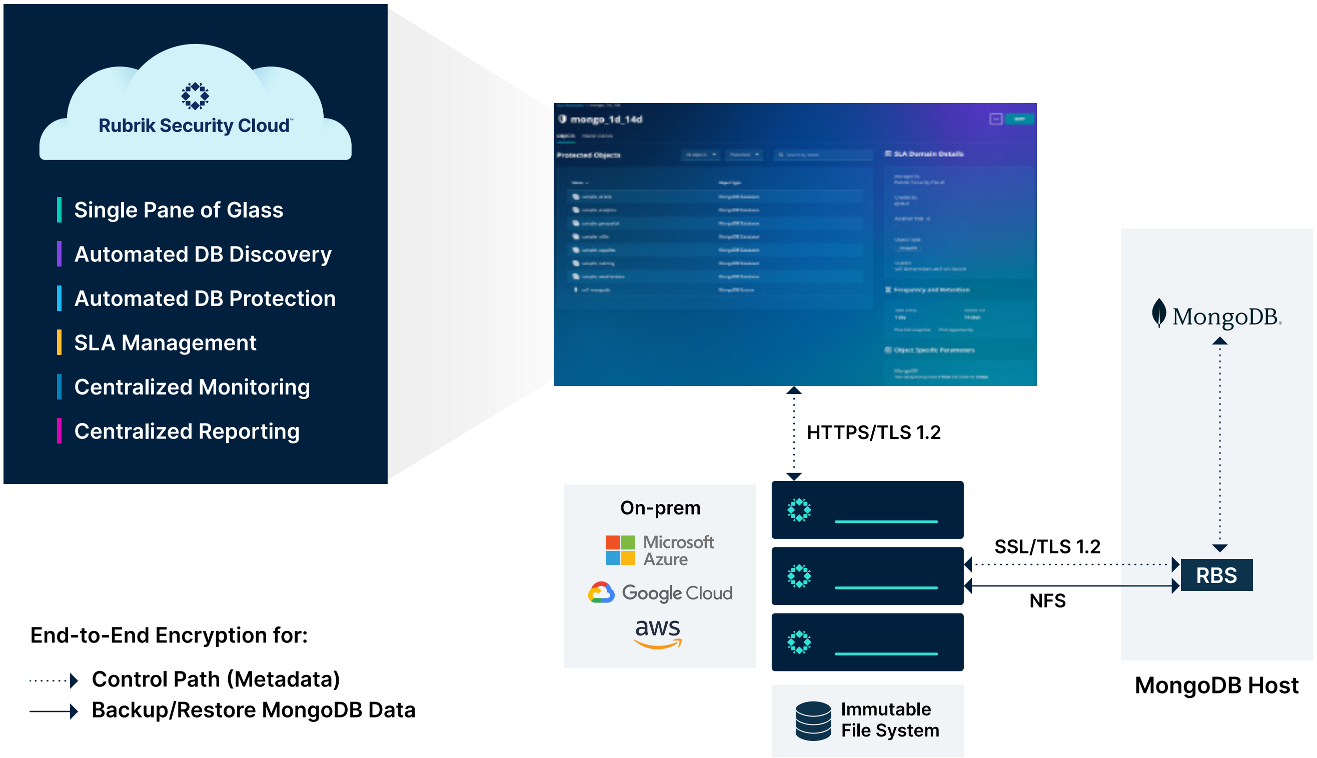


Figure 4 – Solution Overivew

ADDING A MONGODB SOURCE

Before Rubrik can begin protecting a MongoDB cluster, adding the cluster to the RSC is essential. When adding a new MongoDB Source, provide the following details:

- **MongoDB Source Name:** This is a user-defined identifier for the MongoDB cluster.
- **Rubrik Cluster:** This designates the Rubrik Cluster to which backups will be directed. It is the same cluster from which you downloaded the RBS.
- **Deployment Type:** Currently, only the “Replica Set” deployment type is supported at the time of this writing.
- **Host Details:** At least one of the hosts where you installed the RBS must be supplied. While Rubrik can discover the entire cluster topology by connecting to one node, providing details for all nodes ensures full cluster discovery in case of connection issues with the first node.

When configuring your MongoDB Source, you can include additional parameters:

- **Ignore Nodes:** Enabling this option instructs Rubrik to exclude the specified node(s) from any discovery, backup, or recovery operations. While an ignored node will not be directly used for recovery purposes, data on the ignored node will be recovered as MongoDB will replicate from the node being written to the ignored node.

- **SSL Authentication:** This feature permits the incorporation of SSL authentication into the cluster. It lets you define source certificate validation requirements and specify paths for CA certificates and SSL key files.
- **Driver Authentication:** This functionality enables you to supply Rubrik with a username and password to authenticate access to the MongoDB cluster.

Database Credentials for Driver Authentication

If your MongoDB cluster requires Driver Authentication, you must provide a username and password with appropriate permissions to the MongoDB Source. Various permissions are essential at distinct levels within the environment to enable Rubrik to explore, back up, and restore the environment. The specific permissions needed can be found in our user guide.

As of the latest update, the following list outlines the permissions that Rubrik requires:

Operation	Resource	Actions
Backup	Cluster	getCmdLineOpts, listDatabases, replSetGetConfig, replSetGetStatus
Backup	Database Collection	collStats, find, dbStats, listCollections, listIndexes
Backup	Database Local Collection oplog.rs	collStats find
Backup	rubrik_backup_agent_db*	convertToCapped createCollection createIndex insert update dropCollection dropDatabase
Restore	anyResource	anyAction

When adding the MongoDB Source, a new database will be created called **rubrik_backup_agent_db**. This database has capped collections with a fixed file size of 200 MB, with newer records overwriting the older ones. The above permissions must be set on the database once the MongoDB Source is added.

DISCOVERY

When adding the MongoDB Source, Rubrik follows a sequence of steps:

- Rubrik initiates a connection to the seed node(s) to uncover the topology of the MongoDB cluster. If a single seed node is provided, Rubrik will identify all other nodes within the cluster. If all nodes are supplied, Rubrik will validate the accuracy of this information. Additionally, Rubrik checks if the RBS is installed and reachable from the Rubrik Service Console (RSC).

- Rubrik proceeds to verify its ability to authenticate with the MongoDB cluster using the provided database credentials. It also confirms whether it is authorized to conduct discovery, backup, and recovery operations.

- The subsequent step involves the discovery of databases and collections within the cluster. Rubrik records this information as part of the metadata it collects.

This process repeats every 15 minutes. As the environment changes, Rubrik dynamically adapts to these modifications. This implies that newly created databases or collections automatically inherit a Service Level Agreement (SLA) and are backed up by the SLA rules.

DATA PROTECTION

The next step would be to start protecting the MongoDB using Rubrik Global SLA Domain Policy to ensure the reliability and availability of data backups. The goal is to ensure your organization's data can be quickly and easily restored during a cyberattack or natural disaster.

SLA POLICY ENGINE

A Rubrik SLA Domain Policy is a declarative policy that captures the core objectives for backup and recovery. It translates the Recovery Point Objective (RPO) requirements for data protection and manages the data from the cradle to the grave. More importantly, it eliminates the need to manually configure jobs, tasks, and other activities to maintain a data protection scheme.

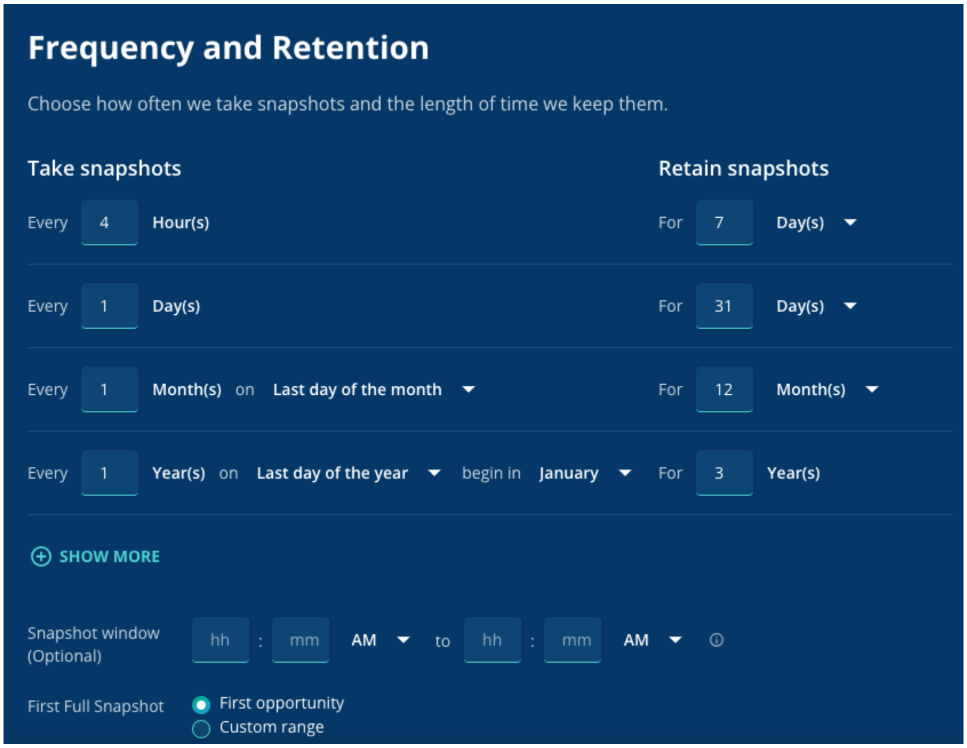
SLA Domains are a core part of the Rubrik architecture. You can add any workload type Rubrik supports to an SLA Domain. This provides a simple mechanism to control data protection for different workloads across your on-prem, edge, or cloud environments.

Regular Use SLA Domains

You should configure a custom SLA Domain Policy depending on the RPO and SLA requirements. Let's walk through the pieces required to configure an SLA Domain Policy for any object:

- **Backup Frequency:** also known as the Recovery Point Objective (RPO). Simply put, how often are backups taken?
  - For databases, this determines how often a database restore point is taken. Each restore point is synthesized from incremental blocks in each backup to maintain an incremental forever scheme. If a database is in the Full Recovery model, the RPO is further reduced by Oplog backups.

- **Backup Retention:** indicates the length of time backups are held for a particular backup frequency.
  - You can optionally select the beginning time for the snapshot window. The first full backup is initiated within the specified window. The first full backup occurs when the MongoDB source or the database or the collection has been added by default.



Frequency and Retention

Choose how often we take snapshots and the length of time we keep them.

Take snapshots | Retain snapshots

Every 4 Hour(s) | For 7 Day(s)

Every 1 Day(s) | For 31 Day(s)

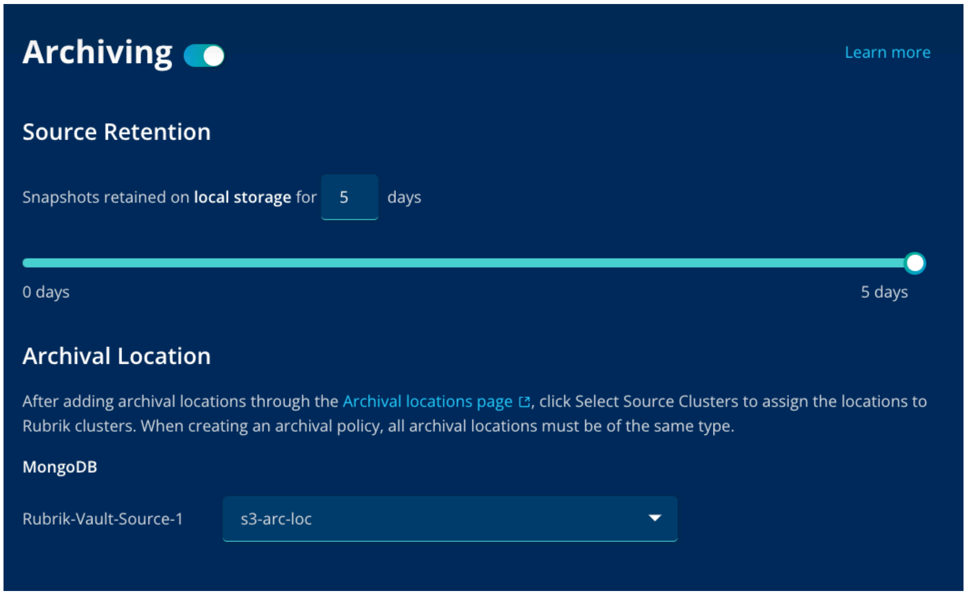
Every 1 Month(s) on Last day of the month | For 12 Month(s)

Every 1 Year(s) on Last day of the year begin in January | For 3 Year(s)

Snapshot window (Optional): hh : mm AM to hh : mm AM ⓘ

First Full Snapshot: ● First opportunity ○ Custom range

- **Archival Policy:** this defines the life cycle of backups during the archival process. It defines when backups are sent to the archive, where they are stored, and the retention of data archived. Archives create an offsite copy of your data in cheaper storage tiers, offering significantly reduced costs. Archive targets can be public cloud (AWS, Azure, GCP, or Rubrik Cloud Vault) or on-premises (S3 compatible object stores, NFS, or tape).
  - For databases, long-term archiving is often stored in a cloud archive such as Rubrik Cloud Vault for regulatory or compliance reasons.
  - Rubrik Cloud Vault offers account-isolated, offsite, immutable copies of your data with authenticated, fully encrypted Azure Blob integration. You can set up your secure, isolated cloud environment directly from RSC in minutes. Visit the Rubrik Cloud Vault webpage to read about the archive solution.



Archiving

Learn more

Source Retention

Snapshots retained on **local storage** for 5 days

0 days 5 days

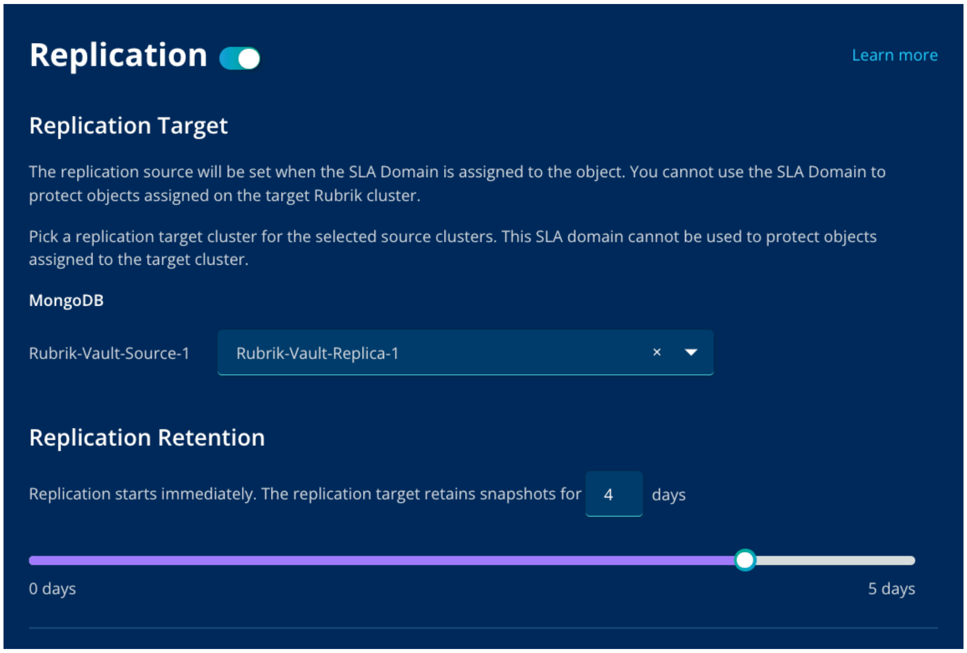
Archival Location

After adding archival locations through the Archival locations page, click Select Source Clusters to assign the locations to Rubrik clusters. When creating an archival policy, all archival locations must be of the same type.

MongoDB

Rubrik-Vault-Source-1 s3-arc-loc

- **Replication Policy:** this relates to disaster recovery (DR). Effectively, how long should backups replicated to another Rubrik cluster be kept at a DR site?
  - For databases, this often is a shorter time frame. In a DR situation, recovery of the most recent state of a database is most common.



Replication

Learn more

Replication Target

The replication source will be set when the SLA Domain is assigned to the object. You cannot use the SLA Domain to protect objects assigned on the target Rubrik cluster.

Pick a replication target cluster for the selected source clusters. This SLA domain cannot be used to protect objects assigned to the target cluster.

MongoDB

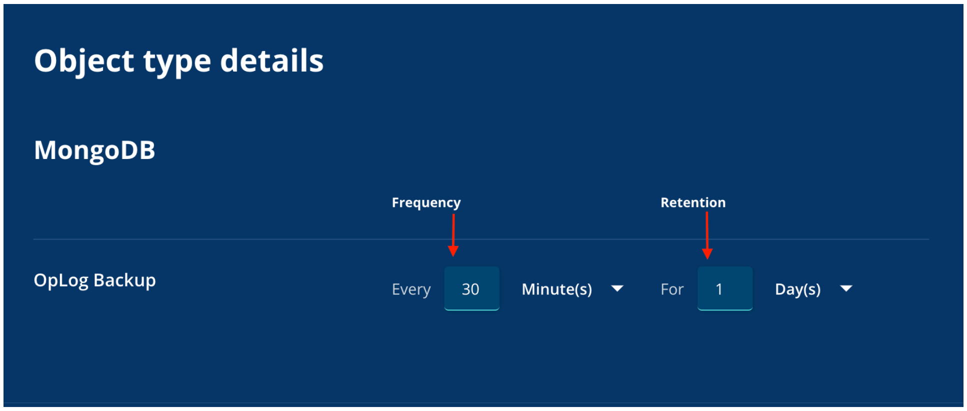
Rubrik-Vault-Source-1 Rubrik-Vault-Replica-1 ×

Replication Retention

Replication starts immediately. The replication target retains snapshots for 4 days

0 days 5 days

- **Log Backup:** This relates to the point-in-time or granular recovery of the MongoDB database or collections. You need to enable Oplog backup as part of the SLA Policy.
  - The combination of a snapshot of the database and the Oplog backups from the database permits Rubrik to recover a database to the state it was in at a selected point in time.
  - Rubrik automatically replays the Oplog to ensure granular recovery.



Object type details

MongoDB

	Frequency	Retention
OpLog Backup	Every 30 Minute(s)	For 1 Day(s)

A visual example of the above SLA Domain policy applied to a MongoDB database would look something like this:

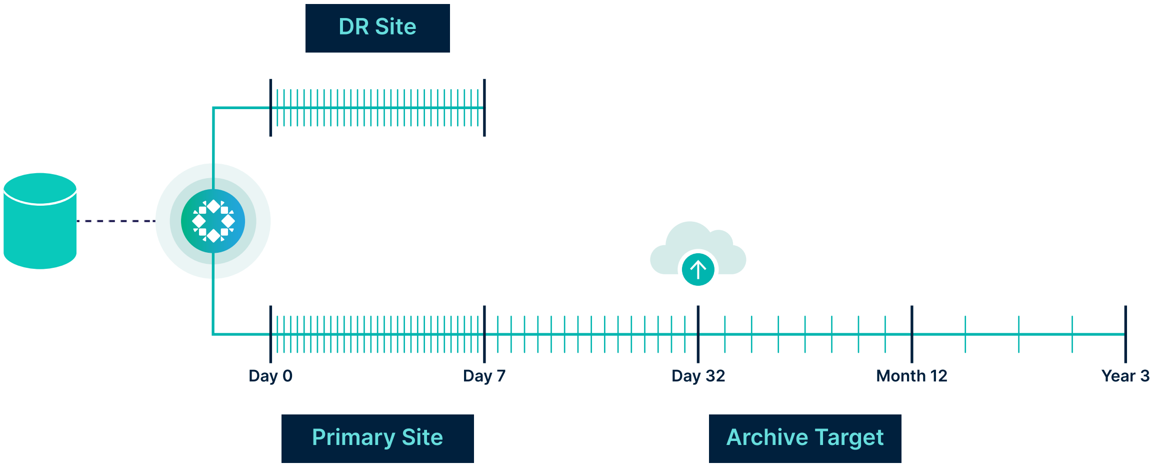


Figure 5 – Example of SLA Domain Policy

As illustrated by the screenshots above, the policy architecture is straightforward to configure yet powerful. Please see the Rubrik User Guide for a more thorough walkthrough of SLA Domain details.

Special Use Case SLAs

As a Database Administrator (DBA), you may encounter various one-time events where you must take on-demand snapshots before proceeding. These events can include but are not limited to:

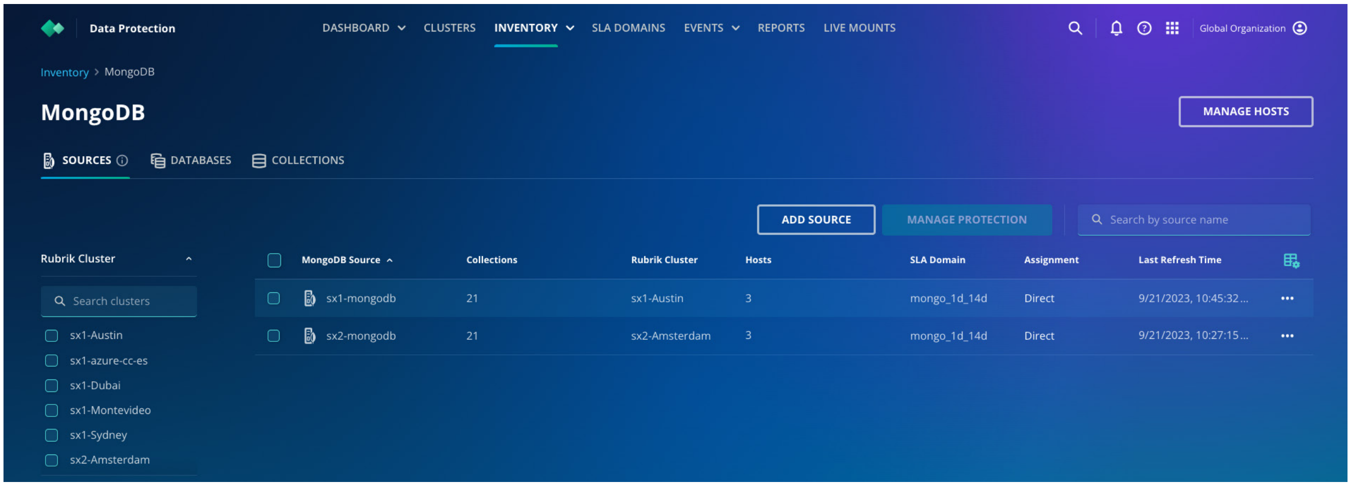
- Code Deployments
- Database Changes
- Patching
- Database or Instance Decommission

Typically, databases or hosts with regular or customized SLAs have longer retention periods for their snapshots. However, for specific events, on-demand snapshots taken just before the event are considered special. This raises the question of whether it's necessary to retain the on-demand snapshot for the entire SLA period, such as 7 years. Utilizing a Special Use SLA Domain is an appropriate solution in such cases. This domain enables the on-demand snapshot to be retained for a shorter period, such as 7 days, instead of the regular SLA period of 7 years. Rubrik automatically expires and deletes the on-demand snapshot after 7 days, saving storage costs.

SLA DOMAIN ASSIGNMENT

As noted above, SLA Domain Policies can be applied at the MongoDB source or database level. The following visual walkthrough illustrates this concept and showcases how to configure MongoDB-specific options.

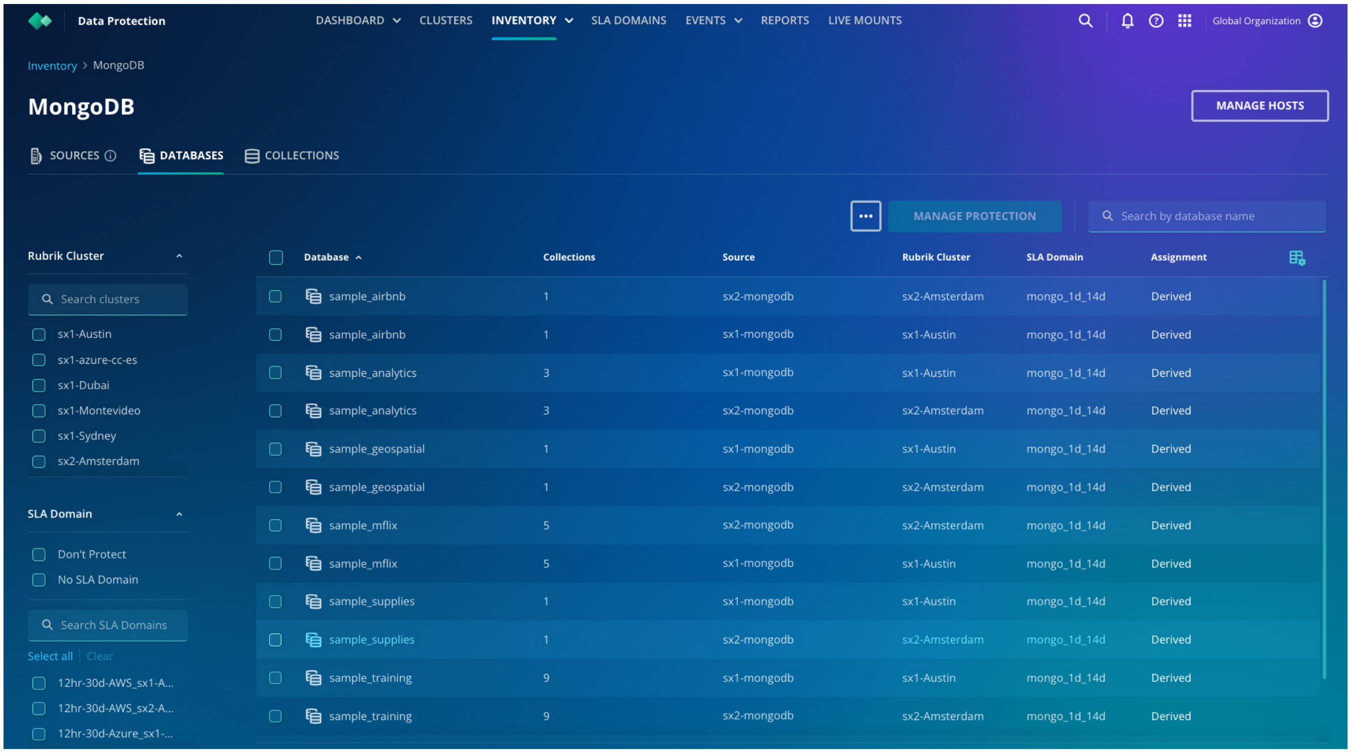
From the *Inventory* menu, select *MongoDB* Databases. A list of added MongoDB sources and counts of their discovered collections will be displayed:



□ sx1-Austin
□ sx1-azure-cc-es
□ sx1-Dubai
□ sx1-Montevideo
□ sx1-Sydney
□ sx2-Amsterdam

□	MongoDB Source	Collections	Rubrik Cluster	Hosts	SLA Domain	Assignment	Last Refresh Time	
□	sx1-mongodb	21	sx1-Austin	3	mongo_1d_14d	Direct	9/21/2023, 10:45:32 ...	•••
□	sx2-mongodb	21	sx2-Amsterdam	3	mongo_1d_14d	Direct	9/21/2023, 10:27:15 ...	•••

Database level details:



MongoDB

Rubrik Cluster

- □ sx1-Austin
- □ sx1-azure-cc-es
- □ sx1-Dubai
- □ sx1-Montevideo
- □ sx1-Sydney
- □ sx2-Amsterdam

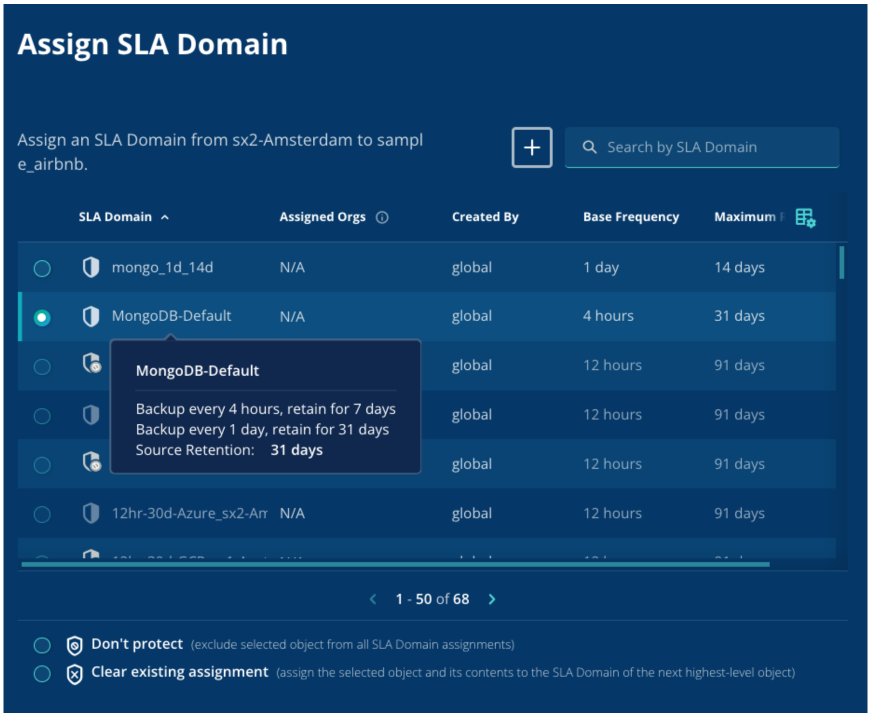
SLA Domain

- □ Don't Protect
- □ No SLA Domain

- □ 12hr-30d-AWS_sx1-A...
- □ 12hr-30d-AWS_sx2-A...
- □ 12hr-30d-Azure_sx1-...

□	Database	Collections	Source	Rubrik Cluster	SLA Domain	Assignment
□	sample_airbnb	1	sx2-mongodb	sx2-Amsterdam	mongo_1d_14d	Derived
□	sample_airbnb	1	sx1-mongodb	sx1-Austin	mongo_1d_14d	Derived
□	sample_analytics	3	sx1-mongodb	sx1-Austin	mongo_1d_14d	Derived
□	sample_analytics	3	sx2-mongodb	sx2-Amsterdam	mongo_1d_14d	Derived
□	sample_geospatial	1	sx1-mongodb	sx1-Austin	mongo_1d_14d	Derived
□	sample_geospatial	1	sx2-mongodb	sx2-Amsterdam	mongo_1d_14d	Derived
□	sample_mflix	5	sx2-mongodb	sx2-Amsterdam	mongo_1d_14d	Derived
□	sample_mflix	5	sx1-mongodb	sx1-Austin	mongo_1d_14d	Derived
□	sample_supplies	1	sx1-mongodb	sx1-Austin	mongo_1d_14d	Derived
□	sample_supplies	1	sx2-mongodb	sx2-Amsterdam	mongo_1d_14d	Derived
□	sample_training	9	sx1-mongodb	sx1-Austin	mongo_1d_14d	Derived
□	sample_training	9	sx2-mongodb	sx2-Amsterdam	mongo_1d_14d	Derived

After selecting a source or database, clicking the *Manage Protection* button brings up the policy assignment screen, where users can assign a policy and set options around copy-only backups, Oplog backup frequency, and Oplog backup retention.



Assign SLA Domain

Assign an SLA Domain from sx2-Amsterdam to sampl e_airbnb.

	SLA Domain	Assigned Orgs	Created By	Base Frequency	Maximum R
○	mongo_1d_14d	N/A	global	1 day	14 days
●	MongoDB-Default	N/A	global	4 hours	31 days
○			global	12 hours	91 days
○			global	12 hours	91 days
○			global	12 hours	91 days
○	12hr-30d-Azure_sx2-Am	N/A	global	12 hours	91 days

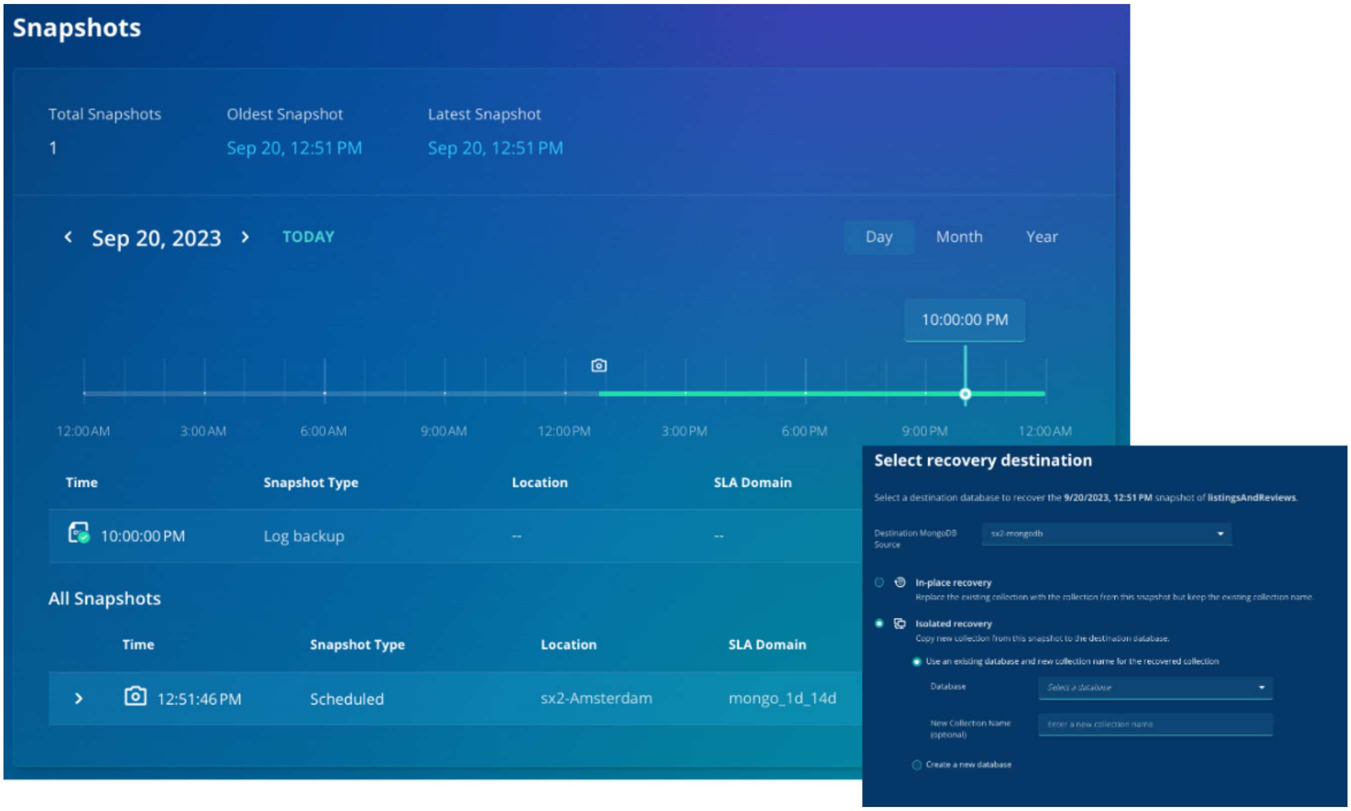
MongoDB-Default

Backup every 4 hours, retain for 7 days
Backup every 1 day, retain for 31 days
Source Retention: **31 days**

○ **Don't protect** (exclude selected object from all SLA Domain assignments)

○ **Clear existing assignment** (assign the selected object and its contents to the SLA Domain of the next highest-level object)

Once protected, the restore process is a simple slider for point-in-time restore options:



Snapshots

Total Snapshots	Oldest Snapshot	Latest Snapshot
1	Sep 20, 12:51 PM	Sep 20, 12:51 PM

‹ Sep 20, 2023 › TODAY

Day Month Year

10:00:00 PM

Time	Snapshot Type	Location	SLA Domain
10:00:00 PM	Log backup	--	--

All Snapshots

Time	Snapshot Type	Location	SLA Domain
› 12:51:46 PM	Scheduled	sx2-Amsterdam	mongo_1d_14d

Select recovery destination

Select a destination database to recover the **9/20/2023, 12:51 PM** snapshot of **listingsAndReviews**.

Destination MongoDB Source: sx2-mongodb

○ **In-place recovery**
Replace the existing collection with the collection from this snapshot but keep the existing collection name.

● **Isolated recovery**
Copy new collection from this snapshot to the destination database.

● Use an existing database and new collection name for the recovered collection

Database: *Select a database*

New Collection Name (optional): Enter a new collection name

○ Create a new database

Granular Database Protection

SLA domain policies can be assigned at the MongoDB source, cluster, or database level. It is recommended to assign an SLA Domain at the highest level: in this case, at the source level. All lower levels will automatically receive the same protection by applying an SLA Domain at the highest level, meaning Source → Database → Collection. Assigning an SLA Domain to the sources will ensure all collections inherit the same policy. Similarly, databases will inherit the policy from their respective clusters. Rubrik automatically detects and secures newly added databases to a source.

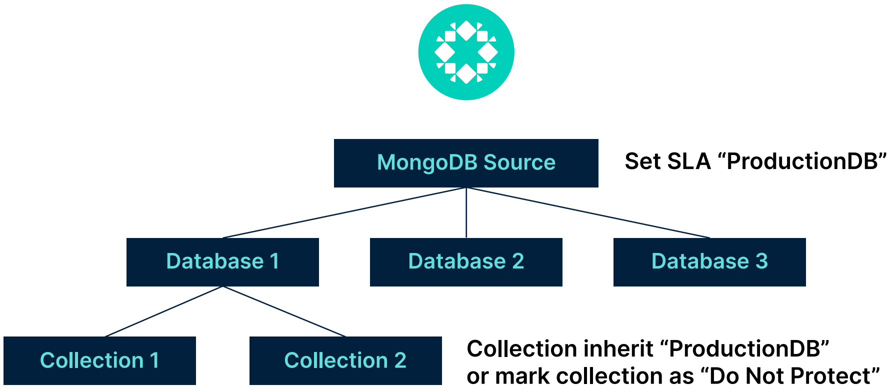
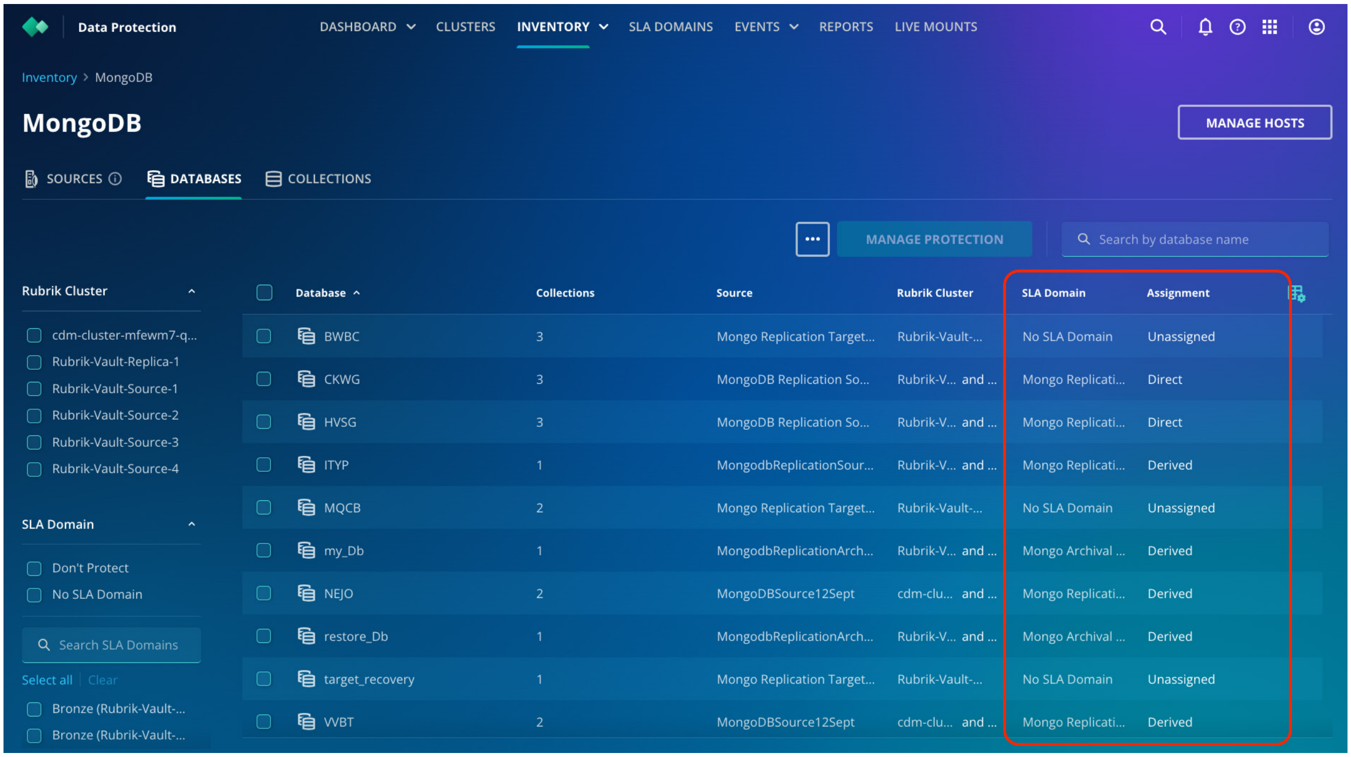


Figure 6 – Granular Data Protection

You can assign a different policy for each source to meet your business requirements when you have multiple sources. Granular SLAs can be configured and assigned at cluster or database for more stringent RPO and RTO requirements.



MongoDB

Rubrik Cluster

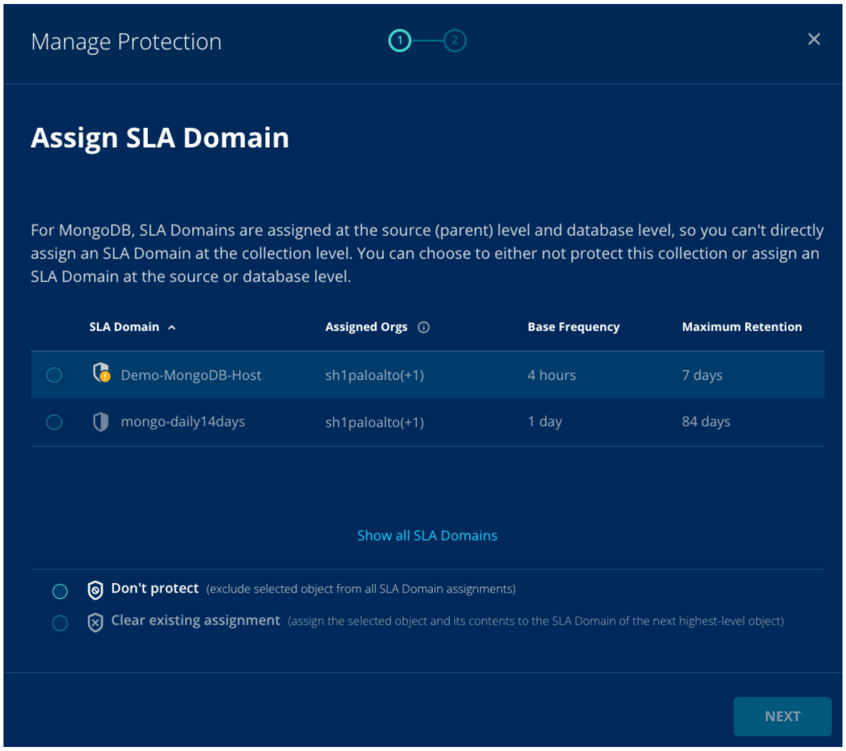
- □ cdm-cluster-mfewm7-q...
- □ Rubrik-Vault-Replica-1
- □ Rubrik-Vault-Source-1
- □ Rubrik-Vault-Source-2
- □ Rubrik-Vault-Source-3
- □ Rubrik-Vault-Source-4

SLA Domain

- □ Don't Protect
- □ No SLA Domain
- □ Bronze (Rubrik-Vault-...
- □ Bronze (Rubrik-Vault-...

Database	Collections	Source	Rubrik Cluster	SLA Domain	Assignment
BWBC	3	Mongo Replication Target...	Rubrik-Vault-...	No SLA Domain	Unassigned
CKWG	3	MongoDB Replication So...	Rubrik-V... and ...	Mongo Replicati...	Direct
HVSG	3	MongoDB Replication So...	Rubrik-V... and ...	Mongo Replicati...	Direct
ITYP	1	MongodbReplicationSour...	Rubrik-V... and ...	Mongo Replicati...	Derived
MQCB	2	Mongo Replication Target...	Rubrik-Vault-...	No SLA Domain	Unassigned
my_Db	1	MongodbReplicationArch...	Rubrik-V... and ...	Mongo Archival ...	Derived
NEJO	2	MongoDBSource12Sept	cdm-clu... and ...	Mongo Replicati...	Derived
restore_Db	1	MongodbReplicationArch...	Rubrik-V... and ...	Mongo Archival ...	Derived
target_recovery	1	Mongo Replication Target...	Rubrik-Vault-...	No SLA Domain	Unassigned
VVBT	2	MongoDBSource12Sept	cdm-clu... and ...	Mongo Replicati...	Derived

When it comes to collection-level protection, Rubrik offers support for inherited SLAs (at the source or database level), or you have the option to opt out of protection entirely.



Assign SLA Domain

For MongoDB, SLA Domains are assigned at the source (parent) level and database level, so you can't directly assign an SLA Domain at the collection level. You can choose to either not protect this collection or assign an SLA Domain at the source or database level.

	SLA Domain	Assigned Orgs	Base Frequency	Maximum Retention
○	Demo-MongoDB-Host	sh1paloalto(+1)	4 hours	7 days
○	mongo-daily14days	sh1paloalto(+1)	1 day	84 days

○ **Don't protect** (exclude selected object from all SLA Domain assignments)

○ **Clear existing assignment** (assign the selected object and its contents to the SLA Domain of the next highest-level object)

BACKUP

MongoDB has a native backup solution for granular recovery and point-in-time recovery, however, those APIs are not exported to third parties. To solve this challenge, Rubrik has developed an innovative approach to back up MongoDB using the native client library interface to collect documents and Oplogs.

To ensure the creation of a consistent backup image, initiating the Oplog backup before commencing the full backup in MongoDB is essential. This process captures any changes made during the full backup, preserving data integrity and consistency.

Rubrik uses a "First Full, Forever Incremental" approach to back up MongoDB databases. This methodology involves performing an initial full backup followed by perpetual incremental backups. This approach enables faster, more network-efficient, and storage-efficient snapshots of the databases.

OPLOG BACKUPS

Oplog backups are executed on every node within the MongoDB Cluster, irrespective of whether they are primary or secondary nodes. This comprehensive approach guarantees the continuity of backups, even during a node failure during the backup process. The Oplogs are transferred to the Rubrik Cluster, where Rubrik reads the log backups to ensure it has a consistent set of logs, and selects the node from wherever it finds the required consistent set first. The node responsible for transmitting this marker indicating the completion of the log backup is the one Rubrik selects for performing the log backup, while simultaneously discarding the log backup data received from other nodes.

The frequency of log snapshots can be configured according to the SLA protection policy. After the log backup process is finished, Rubrik initiates post-processing activities to prepare the snapshot chain. This post-completion phase makes the backups accessible and updates the recoverable range accordingly.

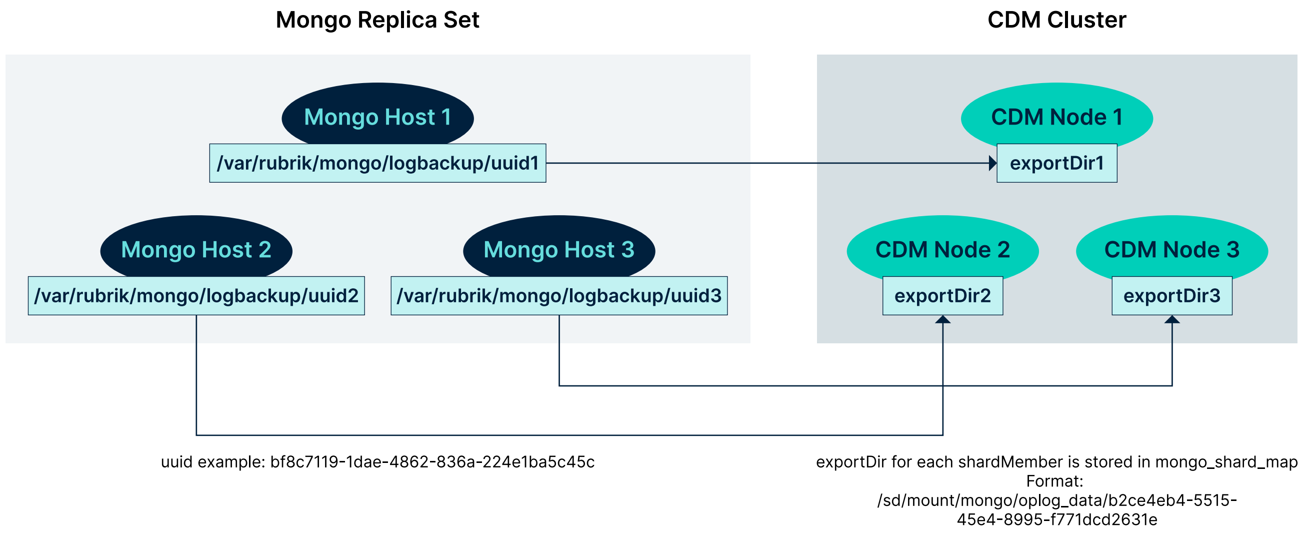

Mongo Replica Set
CDM Cluster
Mongo Host 1
/var/rubrik/mongo/logbackup/uuid1
Mongo Host 2
/var/rubrik/mongo/logbackup/uuid2
Mongo Host 3
/var/rubrik/mongo/logbackup/uuid3
CDM Node 1
exportDir1
CDM Node 2
exportDir2
CDM Node 3
exportDir3
uuid example: bf8c7119-1dae-4862-836a-224e1ba5c45c
exportDir for each shardMember is stored in mongo_shard_map
Format:
/sd/mount/mongo/oplog_data/b2ce4eb4-5515-45e4-8995-f771dcd2631e

Figure 7 – Oplog Backup

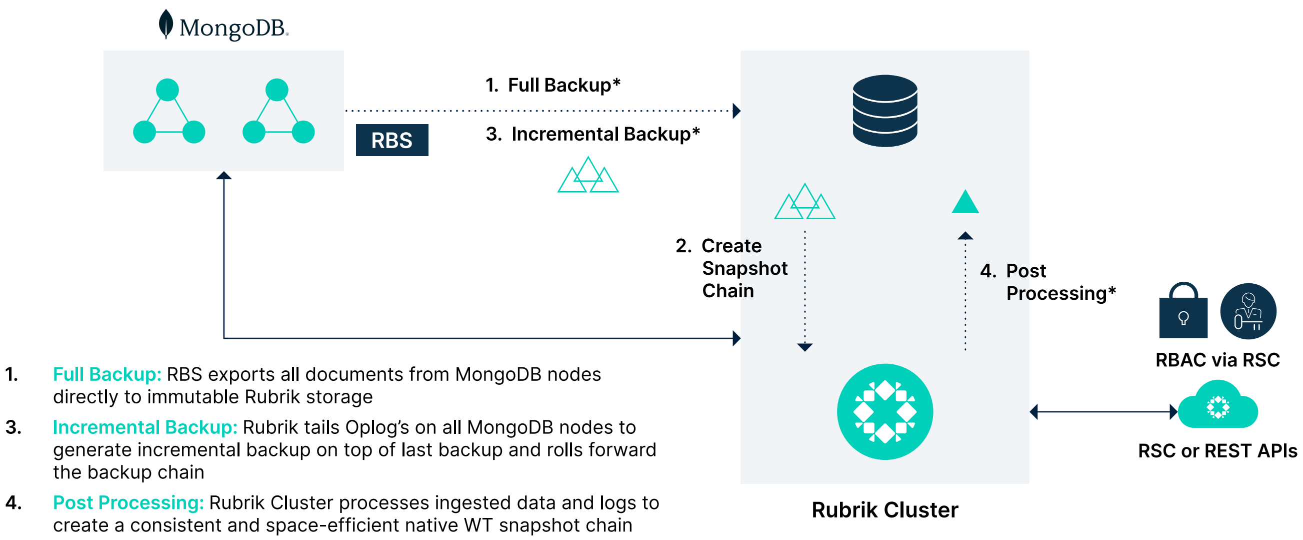
FULL BACKUP

Rubrik employs a three-step process for generating a full backup:

- **Document Extraction:** In this step, documents from each collection are extracted and streamed in their native WiredTiger format to the Rubrik Cluster
- **Log Backup:** Simultaneously with document extraction and image creation, Oplog backups are initiated. These Oplogs are then transmitted to the Rubrik Cluster.
- **Replay:** Rubrik initiates a MongoDB process on its cluster and replays the Oplogs onto the document extraction and database image. This process generates a data file that is readable by MongoDB and, most critically, produces a transactionally consistent snapshot of the database.

FOREVER INCREMENTAL BACKUPS

Rubrik captures periodic snapshots of the OpLogs and then transfers them to the Rubrik Cluster. These OpLog snapshots are crucial for generating a Forever Incremental backup. Subsequently, Rubrik initiates multiple MongoDB processes on the Rubrik Cluster, replaying the log backups atop the most recent full or incremental snapshot. This process maintains a snapshot chain, allowing the ability to restore at the individual collection level easily.



1. **Full Backup:** RBS exports all documents from MongoDB nodes directly to immutable Rubrik storage
3. **Incremental Backup:** Rubrik tails Oplog's on all MongoDB nodes to generate incremental backup on top of last backup and rolls forward the backup chain
4. **Post Processing:** Rubrik Cluster processes ingested data and logs to create a consistent and space-efficient native WT snapshot chain

Figure 8 – Forever Incremental backup

Post Processing / Replay: Rubrik will start a MongoDB process on the Rubrik Cluster and load the previous snapshot created. Rubrik will then take all of the OpLog backups taken from the previous snapshot until now and replay each OpLog backup to create a new WiredTiger backup file. This becomes a new snapshot that includes all data from the previous snapshot and the OpLog data replayed until now. It is stored in a space-efficient WiredTiger snapshot chain on the Rubrik storage.

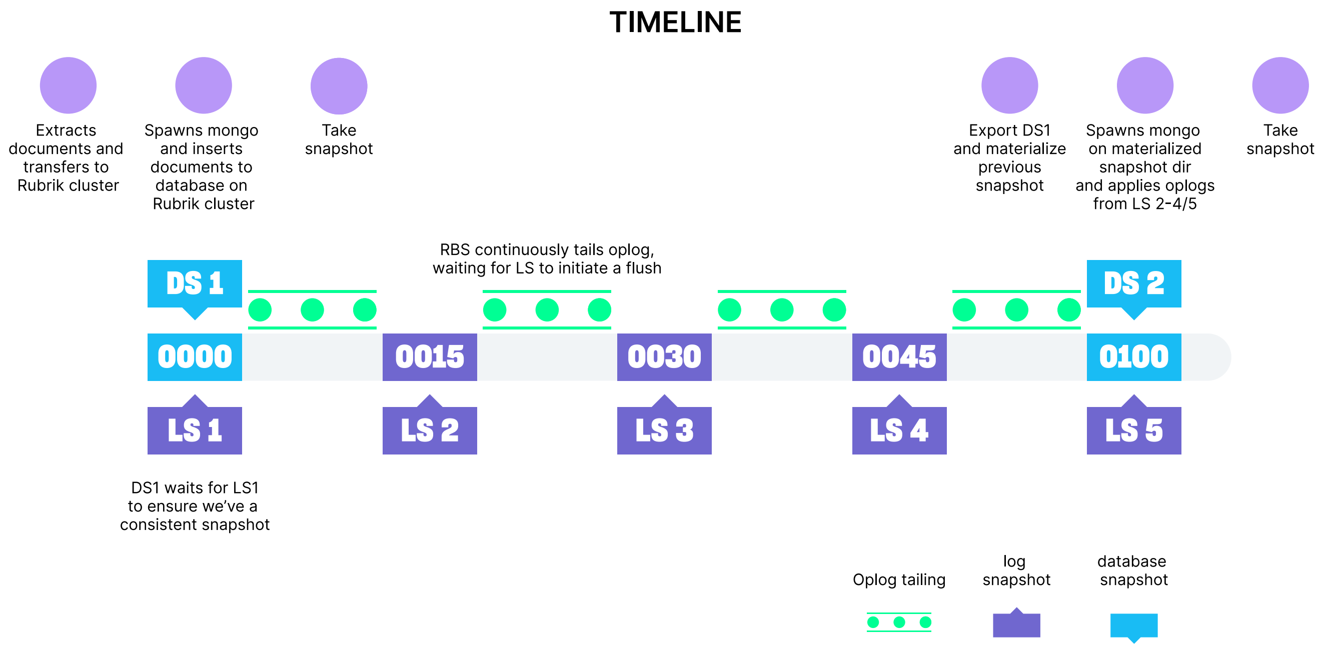

TIMELINE
Extracts documents and transfers to Rubrik cluster
Spawns mongo and inserts documents to database on Rubrik cluster
Take snapshot
Export DS1 and materialize previous snapshot
Spawns mongo on materialized snapshot dir and applies oplogs from LS 2-4/5
Take snapshot
RBS continuously tails oplog, waiting for LS to initiate a flush
DS 1
DS 2
0000
0015
0030
0045
0100
LS 1
LS 2
LS 3
LS 4
LS 5
DS1 waits for LS1 to ensure we've a consistent snapshot
Oplog tailing
log snapshot
database snapshot

Figure 9 – Example of Forever Incremental backup

Examining the image provided, if DS1 is the foundational snapshot, Rubrik will sequentially replay LS1, LS2, LS3, LS4, and LS5 on top of DS1. This process results in the creation of DS2 in the WiredTiger format.

DS2 then assumes the role of the base snapshot for future operations, meaning that the next snapshot will be built upon DS2, and any subsequent Oplog backups will be replayed on DS2 as well. This iterative process ensures the continuity of efficient and consistent data snapshots.

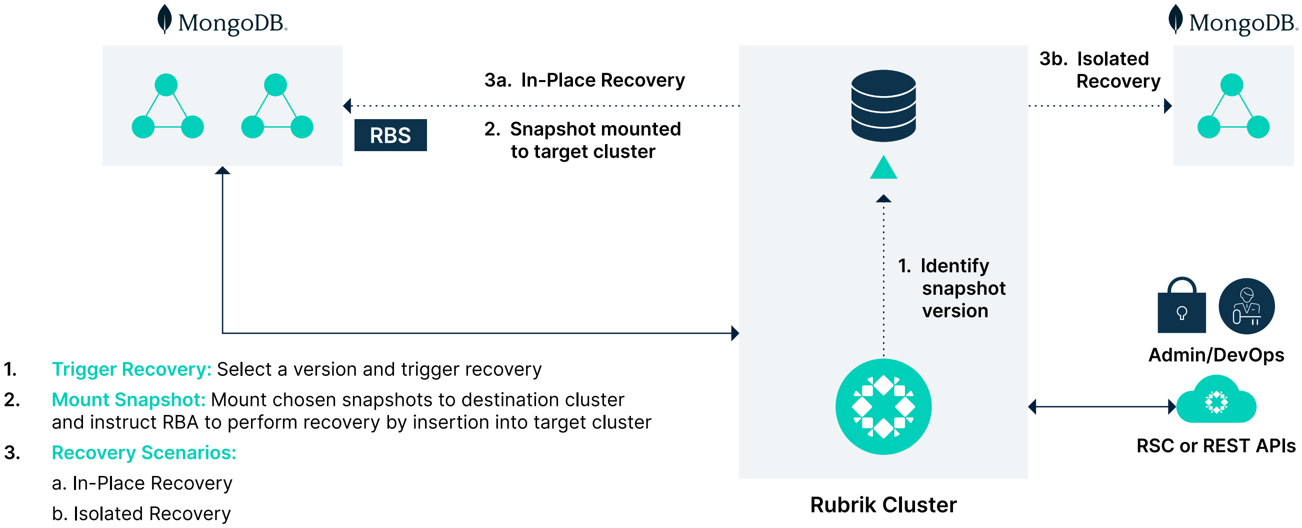
ON-DEMAND SNAPSHOT

On-demand snapshots (ODS) can exclusively be executed at the database level. Furthermore, ODS is limited to databases currently protected by an SLA Domain. Upon clicking the ODS button, you will be presented with the choice to initiate either a Full Snapshot or an Incremental Snapshot. Additionally, you will need to select the SLA that should be applied to the ODS. The ODS retention will be based on the SLA Domain.

When an ODS request is initiated, Rubrik will assess whether an ongoing snapshot operation is in progress for the same database. If an existing snapshot is already in execution, the ODS request will be queued and scheduled to commence once the ongoing snapshot concludes.

Conversely, if an ODS is underway and a scheduled snapshot begins, the latter will be deferred until the ongoing ODS snapshot has been completed. This careful synchronization ensures the preservation of backup consistency and prevents Rubrik from causing performance disruptions to the MongoDB Source.

RECOVERY



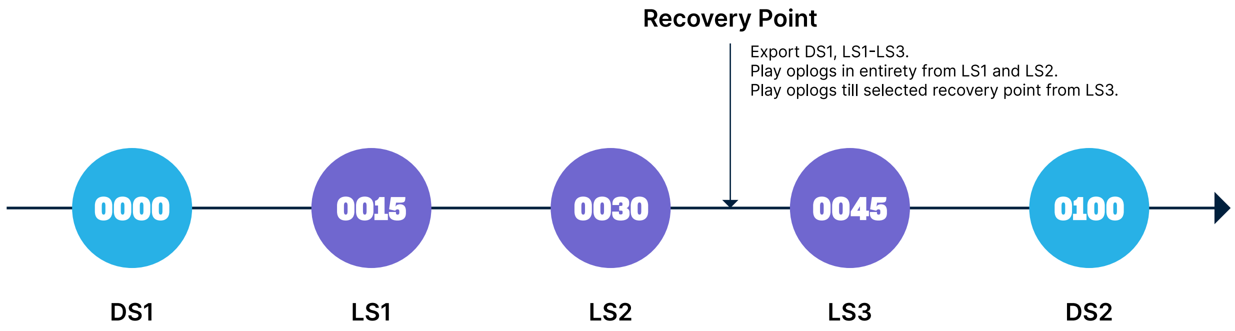
1. **Trigger Recovery:** Select a version and trigger recovery
2. **Mount Snapshot:** Mount chosen snapshots to destination cluster and instruct RBA to perform recovery by insertion into target cluster
3. **Recovery Scenarios:**
   a. In-Place Recovery
   b. Isolated Recovery

Figure 10 – Data Recovery

Rubrik offers users the flexibility to perform recovery operations at both the collection and database levels. It also provides the convenience of bulk recovery for multiple collections within a database or multiple databases concurrently. Regardless of the chosen recovery scope, the user must specify a point in time to recover to, which can either be the latest point or a specific moment on the timeline.

The next option in the recovery process involves configuring the target. You need to designate a target MongoDB Source and then choose between two recovery methods: “In-place” or “Isolated.” An “In-place” recovery will overwrite the target with the data from the selected backup.

Conversely, an “Isolated” recovery is intended for creating a duplicate copy of the data on the selected target, leaving the original data intact.



Export DS1, LS1-LS3.
Play oplogs in entirety from LS1 and LS2.
Play oplogs till selected recovery point from LS3.

Figure 11 – Example of Data Recovery

The depicted image serves as an illustration of a recovery process. DS1 represents a database snapshot, while LSx symbolizes Oplog backups. During a recovery operation, Rubrik initially transfers and restores DS1. Subsequently, it replays LS1, LS2, and segments of LS3 to reach the specified point in time. Although simplified, this illustration applies to bulk recoveries of collections or entire databases, demonstrating the consistent approach Rubrik employs to ensure data recovery accuracy.

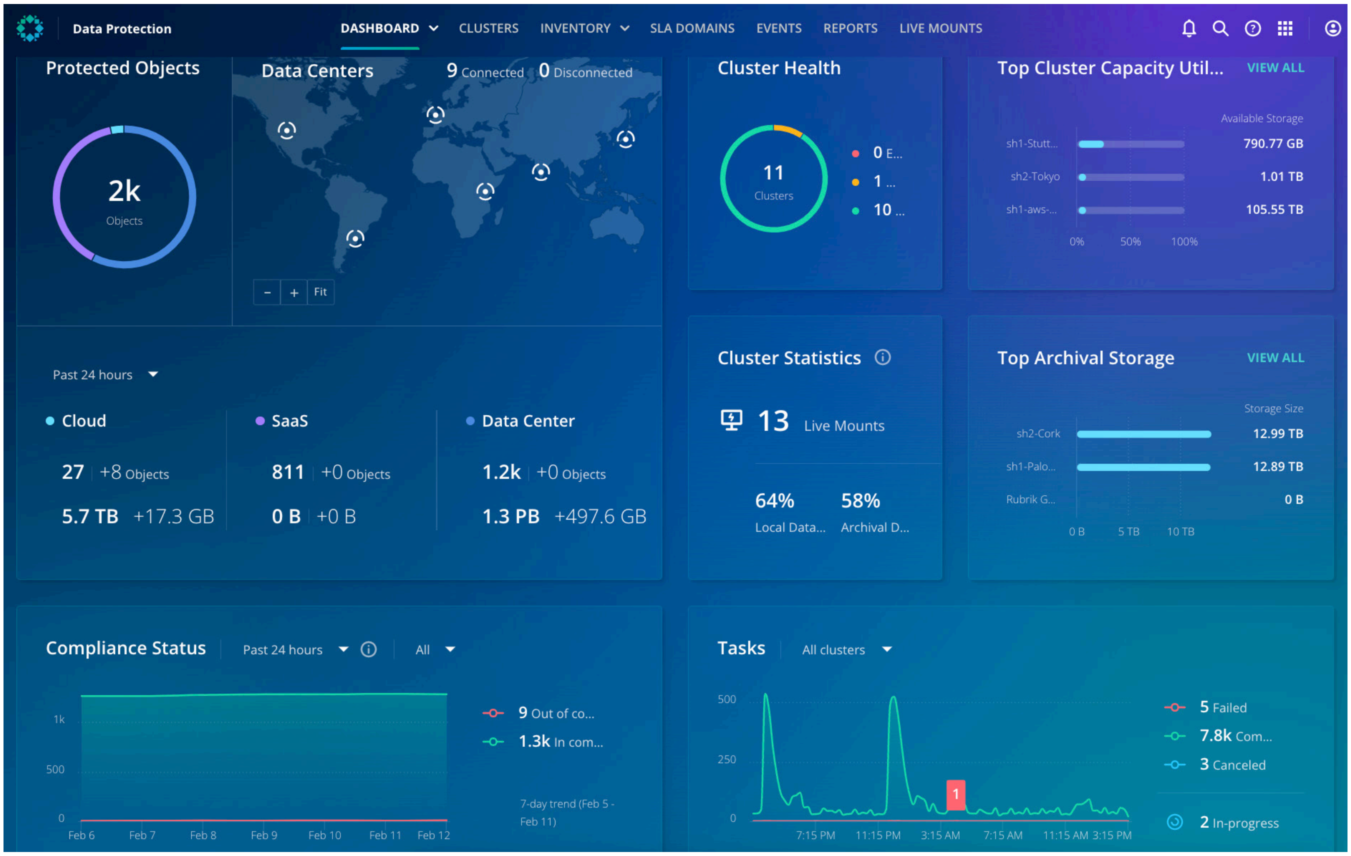
SOLUTION ARCHITECTURE

CENTRALIZED MANAGEMENT

Rubrik provides centralized management for your global, distributed Rubrik environment, focusing on delivering a seamless user experience. By providing a comprehensive view of your physical, virtual, and cloud topologies, Rubrik simplifies management tasks elegantly and intuitively.

Data Protection Dashboard

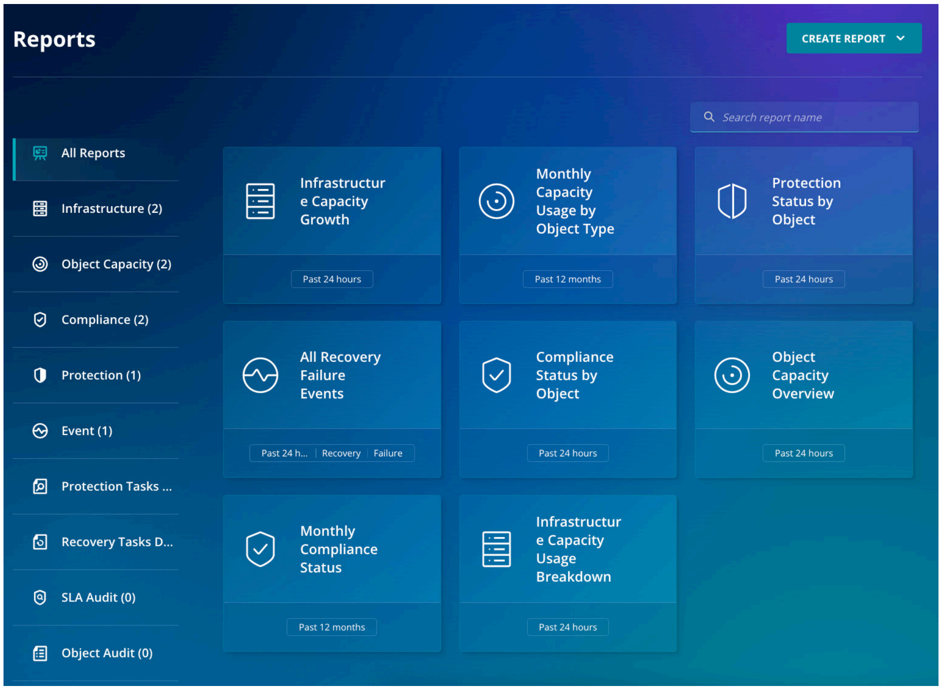
RSC includes the Data Protection Dashboard, which offers analytics on data management, compliance, and capacity utilization for your entire infrastructure - on-premises, at the edge, and in the cloud. This dashboard provides an at-a-glance view of SLA-compliant applications and lets you display all set-up events aggregated over a configurable time range.



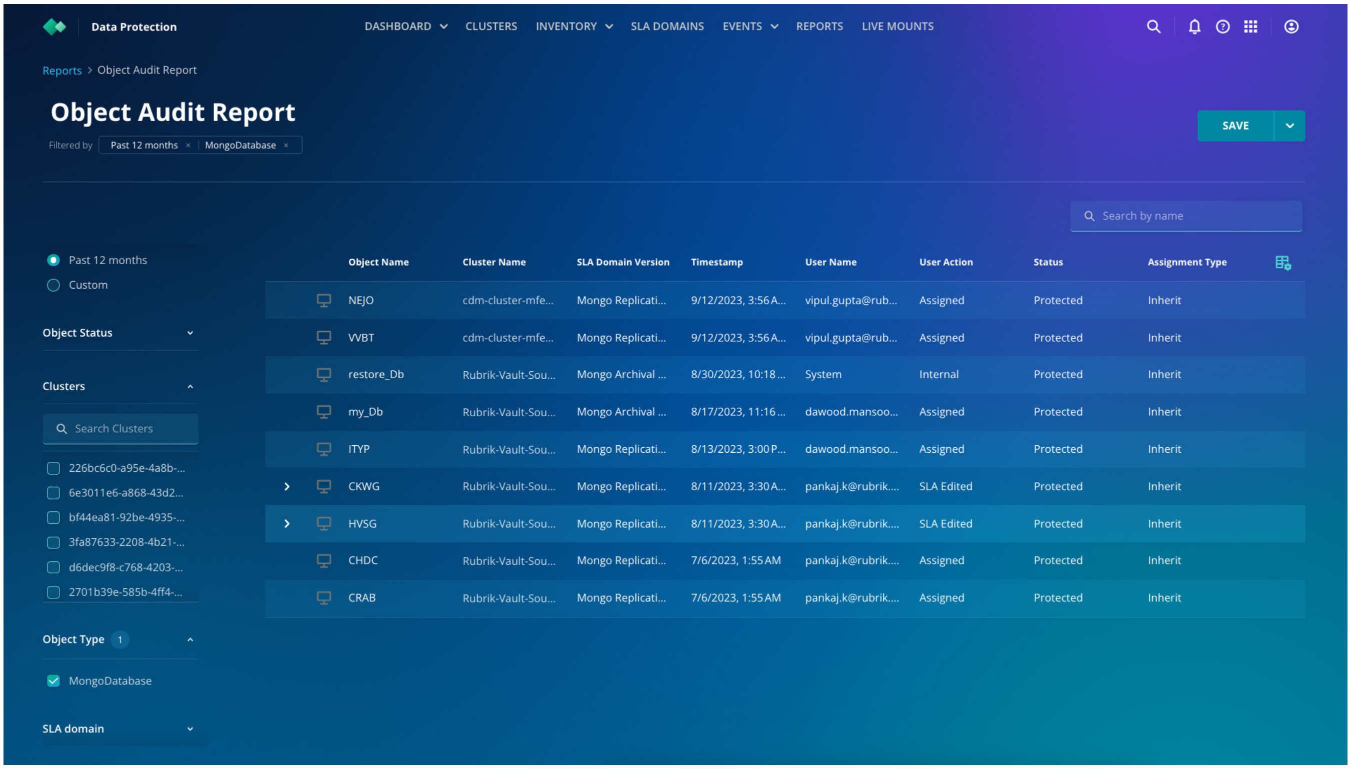
REPORTING

In addition, RSC offers reports that can be customized to display information on application data protection and the underlying infrastructure. Utilizing the on-demand infrastructure health and behavior insights allows you to maximize cost savings and performance while sharing rich data visualizations and customized reports that promote operational efficiencies.

Furthermore, a responsive HTML5-based interface enables the creation of custom reporting workflows using commonly used system metrics. The data collected in these reports can be used for audits and data management planning.



For example, RSC offers audit reports with granular historical information on SLA Domains and workload objects managed by Rubrik clusters to help manage regulatory requirements. RSC generates the SLA Audit report once an SLA Domain is created, while an entry for the related details is added to the Object Audit report once an SLA Domain is assigned to a workload object such as MongoDB.



Filtered by Past 12 months × | MongoDatabase ×

● Past 12 months
○ Custom

□ 226bc6c0-a95e-4a8b-...
□ 6e3011e6-a868-43d2...
□ bf44ea81-92be-4935-...
□ 3fa87633-2208-4b21-...
□ d6dec9f8-c768-4203-...
□ 2701b39e-585b-4ff4-...

Object Type 1

☑ MongoDatabase

Object Name	Cluster Name	SLA Domain Version	Timestamp	User Name	User Action	Status	Assignment Type
NEJO	cdm-cluster-mfe...	Mongo Replicati...	9/12/2023, 3:56 A...	vipul.gupta@rub...	Assigned	Protected	Inherit
VVBT	cdm-cluster-mfe...	Mongo Replicati...	9/12/2023, 3:56 A...	vipul.gupta@rub...	Assigned	Protected	Inherit
restore_Db	Rubrik-Vault-Sou...	Mongo Archival ...	8/30/2023, 10:18 ...	System	Internal	Protected	Inherit
my_Db	Rubrik-Vault-Sou...	Mongo Archival ...	8/17/2023, 11:16 ...	dawood.mansoo...	Assigned	Protected	Inherit
ITYP	Rubrik-Vault-Sou...	Mongo Replicati...	8/13/2023, 3:00 P...	dawood.mansoo...	Assigned	Protected	Inherit
CKWG	Rubrik-Vault-Sou...	Mongo Replicati...	8/11/2023, 3:30 A...	pankaj.k@rubrik....	SLA Edited	Protected	Inherit
HVSG	Rubrik-Vault-Sou...	Mongo Replicati...	8/11/2023, 3:30 A...	pankaj.k@rubrik....	SLA Edited	Protected	Inherit
CHDC	Rubrik-Vault-Sou...	Mongo Replicati...	7/6/2023, 1:55 AM	pankaj.k@rubrik....	Assigned	Protected	Inherit
CRAB	Rubrik-Vault-Sou...	Mongo Replicati...	7/6/2023, 1:55 AM	pankaj.k@rubrik....	Assigned	Protected	Inherit

PROTECTION AGAINST RANSOMWARE

Backup data is the last line of defense and the key to recovering from a ransomware attack. The founding engineers of Rubrik made security a core design principle from the beginning of product development. The secure-by-design approach of Rubrik makes it easy for customers to implement a superior security posture related to backups and data management by reducing manual work post-deployment. As part of Zero Trust Data Security™, this methodology gives customers confidence not only that their data is safe but that they will also be able to recover from an attack quickly.

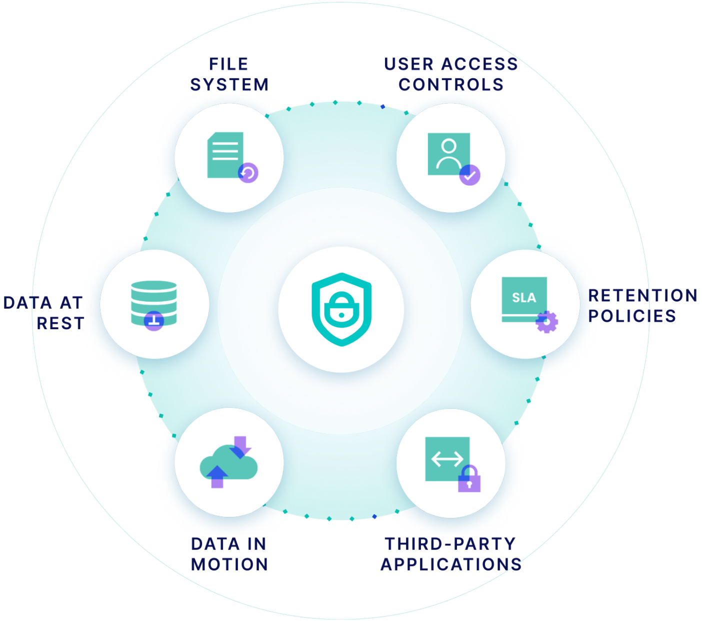


Figure 12 – Zero Trust By Design

The Zero Trust model, extends to how Rubrik offers the following features to minimize data loss in the event of a cyber attack:

Immutable by Design

Rubrik has created a custom append-only file system. Once the object backups are written to the Rubrik file system, they can't be modified or deleted, making the backups immutable. This feature ensures the preservation of backup integrity, allowing for their use in recovery even in case of a compromise to production data.

Encryption Everywhere

Rubrik requires strong ciphers with forward secrecy for all connections and uses a strong access key to authenticate Rubrik clusters. Data transmission between RSC and Rubrik clusters is encrypted with the Transport Layer Security (TLS) protocol. RSC accesses and manipulates data on Rubrik clusters exclusively over HTTPS, using TLS v1.2 only.

RSC uses TLS v1.2 and strong ciphers with forward secrecy for internal and external communication.

RSC stores and encrypts data at rest with AES-256 encryption using a symmetric key algorithm and keys sized at 256 bits. Data is encrypted before being written to disk and decrypted during read operations.

Comprehensive Identity and Access Management (IAM)

Rubrik enforces strong authentication via complex passwords, required multi-factor authentication (MFA), and granular access control. Let's discuss each of these in detail:

MFA/TOTP

Compromised directory service platforms and individual accounts are hallmarks of a ransomware attack. Privileged accounts and directory services are high-value targets, and attackers will focus on compromising them to gain further control of an environment. To defend against these vulnerabilities, Rubrik enables MFA, by default, that can be used natively with Rubrik Time-based One Time Passwords (TOTP). When configured, access through all system interfaces (GUI, CLI, and API) requires the end-user to perform a secondary authentication process before granting access. This additional layer of security provides a robust defense against any compromised accounts in directory services (such as Microsoft's Active Directory). Since this is native to Rubrik, there is no dependency on third-party identity providers, allowing customers to be up and running with minimal administrative effort. Should you already have one, Rubrik also supports additional third-party MFA providers via 3rd party Identity Provider (IdP) services that support SAML 2.0.

To enhance security against compromised accounts within the Rubrik system, - local accounts can be configured to meet the same authentication requirements and provide secondary authentication to access the system. All multi-factor authentication (MFA) solutions adhere to the account lockout and lockout duration policies defined within the Rubrik system.

Furthermore, - authentication-related activities, such as configuration changes, resynchronization, and password resets, are meticulously logged. These logs aid incident and event management, and they can be seamlessly forwarded to a Security Information and Event Management (SIEM) system for in-depth analysis. By correlating these logged events, any potential malicious actors attempting to launch brute-force attacks on passwords while posing as legitimate users can be thoroughly scrutinized and investigated.

SINGLE SIGN-ON

Single Sign-On can greatly simplify the authentication process for application users and improve administrators' onboarding and off-boarding process, enhancing security by removing the chance of individual credentials being missed when an employee moves on. Rubrik Security Cloud supports SAML 2.0 based Identity Providers, which commonly include multi-factor authentication capabilities.

GLOBAL RBAC

RSC is built on Zero Trust principles, one of which is that users should receive only the required access to fulfill their job requirements. Rubrik provides several pre-built roles for the most common access requirements, but customers can create custom roles scoped to their exact needs. Service Accounts for programmatic access can be assigned to a scoped RBAC role. Additionally, the Organizations feature is available for customers where multi-tenancy is required.

Intelligent Data Lock (IDL)

IDL is a feature that protects your account from malicious activity by delaying the deletion of the snapshots of the data sources backed up by Rubrik clusters. Based on your history, if Rubrik identifies any data deletion event as suspicious, then the snapshots are retained by IDL. If you want to delete the snapshots retained by IDL before the expiration period, contact Rubrik Support.

KEY BENEFITS

Rubrik's native MongoDB protection offers similar ease of use and out-of-the-box experience to the native protection of VMware, Oracle, SQL, SAP HANA, IBM Db2, and other such workloads. Let's take a look at them:

Automated Discovery

- After RBS installation and as MongoDB clusters are added to Rubrik as sources, all the underlying MongoDB database and collections are automatically discovered during day 0 operations
- Rubrik automates the protection of auto-discovered and new objects post-SLA assignment

Declarative SLA Policy Engine

- Streamline the protection of MongoDB databases by assigning SLA policies that configure backup frequency, retention, archiving, and replication using the same engine

Native Protection Benefits

- Reduce storage cost via semantic deduplication where Rubrik ensures writing only a golden copy of MongoDB data from its internally replicated data copies across the replicas
- Consistent snapshots across MongoDB nodes ensure a repair-free and, thus faster recovery of data
- Flexibility to recover data to unlike topology target MongoDB cluster ensures seamless movement of data across dev-QA clusters

Point-in-Time Granularity

- Leverage data and log backups to enable collection-level point-in-time recovery
- Roll forward log backups on top of data backups for granular control over recovery points.

Unified Management and Reporting Platform

- Keep your MongoDB, other databases, and other datacenter objects protected across on-premises and cloud with centralized visibility and control.
- **Active Monitoring:** View the status of your MongoDB data backup from a centralized activities pane.
- **Comprehensive SLA Compliance Reporting:** View backup summary information and the latest recovery points, and identify which backups have failed across your environment.

CONCLUSION

Rubrik delivers a comprehensive and automated data protection solution for your MongoDB databases while providing instant access with near-zero recovery time objectives (RTOs). Designed to manage all your physical and virtualized databases with a single, user-friendly interface, Rubrik can handle both on-premises and cloud environments. With fast object-level recovery and the ability to create unlimited database clones for application development purposes within seconds, Rubrik is an ideal choice for data protection and management.

NEXT STEPS

To learn more, check out our website or contact your sales representative.

VERSION HISTORY

Version	Date	Summary of Changes	Authors
1.0	November 2023	Initial Release	Alpika Singh, Chris Lumnah

rubrik

Global HQ
3495 Deer Creek Road
Palo Alto, CA 94304
United States

1-844-4RUBRIK
inquiries@rubrik.com
www.rubrik.com

rwp-hiw-protecting-mongodb-with-rubrik / 20231115