D

Datastore: Types, Examples, and Security Basics

Learn what a datastore is, how databases, object storage, warehouses, and data lakes differ, and how to secure stored data.

PUBLISHER
Gridinsoft Support

UPDATED
Oct 8, 2026

READ TIME
4 min

CONTENTS

In this guide

01 Common types of datastores

02 Datastore vs. database

03 Data store or datastore: is there a difference?

04 Datastore vs. warehouse and data lake

05 How to choose a datastore

06 Datastore security controls

07 Backups are not the same as replication

08 Common datastore mistakes

09 Datastore troubleshooting checklist

A **datastore** is a system or repository that keeps data so applications or people can retrieve and manage it later. The term is broader than “database”: files, object storage, key-value services, databases, warehouses, and data lakes can all serve as datastores.

The right choice depends on the data model, access pattern, consistency, scale, retention, recovery, and security requirements—not simply which technology can hold the most bytes.

01 Common types of datastores

- **Relational databases** organize structured records into tables and support transactions and SQL.
- **NoSQL databases** may use key-value, document, column-family, or graph models.
- **File systems and network shares** store data as files and directories.
- **Object storage** keeps objects with identifiers and metadata, often at large scale.
- **Data warehouses** consolidate curated historical data for analytics.
- **Data lakes** retain large volumes of raw structured and unstructured data.
- **In-memory stores** prioritize low latency but may need persistence elsewhere.

02 Datastore vs. database

A database is a type of datastore with a defined data model, query behavior, and management system. “Datastore” can also describe a storage abstraction inside a product or a named cloud service. Check the context: a platform’s product called Datastore may have specific features that the generic term does not imply.

03 Data store or datastore: is there a difference?

Data store is the general two-word term for any repository that persists data. **Datastore** is a common spelling in technical writing and is also used in product names and programming interfaces. The spelling alone does not identify a particular architecture. A database is one kind of

data store; a file system, object bucket, cache with persistence, or data lake may be another.

04 Datastore vs. warehouse and data lake

Operational databases support frequent application reads and writes. Warehouses are optimized for analytical queries across consolidated, structured data. Data lakes accept broader raw formats for later processing. Modern platforms can blur these boundaries, but workloads still differ. Sending heavy analytics to a transactional database may harm application performance.

05 How to choose a datastore

1. Define record size, relationships, schema changes, and query patterns.
2. Set consistency, transaction, latency, and availability requirements.
3. Estimate growth, retention, transfer, and backup costs.
4. Plan regional placement, replication, and recovery objectives.
5. Verify access controls, encryption, logging, and compliance needs.
6. Test failure and recovery with realistic data and load.

No single technology maximizes consistency, availability, scale, simplicity, and cost for every workload.

06 Datastore security controls

Keep private stores off the public Internet, authenticate every workload, and grant separate least-privilege roles for reading, writing, administration, and backups. Encrypt network connections and sensitive stored data, manage keys separately, rotate credentials, and log both data access and administrative changes.

Classify data before choosing retention. Test that deletion, legal hold, and backup policies behave as intended across replicas and exports.

07 Backups are not the same as replication

Replication improves availability by copying current state, but it can quickly copy accidental deletion, corruption, or ransomware changes. Maintain versioned or immutable backups in a separate security boundary and regularly restore them. Record recovery time and recovery point objectives instead of assuming a successful backup job guarantees recovery.

08 Common datastore mistakes

- Public access caused by a permissive firewall or storage policy.
- Shared administrator credentials embedded in application code.
- No capacity alerts, lifecycle policy, or tested restore process.
- Collecting sensitive data without a business need or deletion date.
- Logging secrets or personal data into a second, less protected store.
- Choosing a technology by trend rather than measured access patterns.

09 Datastore troubleshooting checklist

When an application cannot read data, confirm the endpoint, DNS, network path, TLS trust, identity, permission, schema, and query. For performance problems, separate client latency from server execution, check indexes and hot keys, and review throttling or resource saturation. Preserve audit evidence before changing permissions during a suspected breach.

NEED MORE HELP?

We are here when the guide is not enough.

Visit the Gridinsoft Support Center for current product guidance or submit a support request with the details of your case.

Open this guide online at support.gridinsoft.com