MongoDB Security Checklist & Best Practices

Try MongoDB Atlas free

Last Updated: June 16, 2020

Data security is a top concern. News stories about new data breaches make the headlines nearly every week, describing compromises that impact thousands of users.

The good news is that MongoDB has everything you need to ensure security best practices, from encryption to authentication, access control, and auditing.

In-depth documentation and detailed resources such as white papers are available to delve deeper into all of the best practices outlined here. This page provides a brief overview of best practices for MongoDB security, with links for learning more.

Now, let's review some of the ways to keep your MongoDB database secure.

1. Create Separate Security Credentials

To enable authentication, create login credentials for each user or process that accesses MongoDB.

Suppose several users need administrative access to the database. Instead of sharing credentials, which increases the likelihood that the account will be compromised, issue each person their own credential and assign them privileges according to their roles, described next.

2. Use Role-Based Access Control

Instead of giving authorizations to individual users, associate authorizations with roles such as application server, database administrator, developer, BI tool, and more. Predefined roles are available out of the box such as dbAdmin, dbOwner, clusterAdmin, and more. Those roles can be further customized to meet the needs of particular teams and functional areas while ensuring consistent policies across the organization.

3. Limit Connections to the Database

One way that data leaks occur is that an intruder gains remote access to the database. By limiting remote connections to the database, you reduce this risk. The best practice is to allow connections only from specified IP addresses, a practice known as whitelisting.

With MongoDB Atlas, the fully managed service for MongoDB, each Atlas project gets its own VPC. For additional security, customers can enable VPC peering to the private networks housing their applications to prevent access over the public internet.

4. Encrypt Your Data

Encryption can be applied in a number of ways:

- Encrypting data at rest. Encrypt the data where it is stored. At rest encryption is not available for MongoDB Community Edition; it requires MongoDB Enterprise or MongoDB Atlas.
- Encrypting data in transit. By default, with MongoDB, all data is encrypted in transit using TLS.

5. Add Extra Encryption for Sensitive Data

A key feature of the MongoDB 4.2 release is client-side field-level encryption.

Most encryption is applied at the server. This means that if someone has access to the server, they may be able to read that data. Client-side field-level encryption ensures that only relevant parties can read their own data on the client-side using their unique decryption key.

This means, in effect, that only the user can read the encrypted data.

Suppose that Ralph's retirement account includes his social security number. The data is stored in encrypted form, so only Ralph can view it. Not the database administrator, not the developer, not the analyst–only Ralph.

Enabling FLE does not require updating application code; only updating the database driver.

Here's an animation that illustrates how this important feature works:

The Client Side Field Level Encryption FAQ offers additional details about FLE, drawn from a full-length webinar available on demand.

6. Auditing and Logs

Audit trails should track who made changes to the configuration of the database, what those changes were and when the changes were made. With its audit framework, MongoDB Enterprise offers a full audit trail of administrative actions.

7. Community Edition or Enterprise Server?

MongoDB Community Edition is the free and open version of MongoDB. MongoDB Enterprise Server offers additional security and performance features for enterprise use cases at scale. A comparison of the two editions is available here, along with instructions for upgrading from Community Edition to Enterprise Server.

But if you are at the point of considering Enterprise Server for features like LDAP integration and encryption at rest, why not evaluate MongoDB Atlas, the fully-managed database as a service that delivers all of the goodness of MongoDB Enterprise Server along with security best practices out of the box? MongoDB Atlas is available and secure by default on all three major public clouds: AWS, Microsoft Azure, and GCP.

The Bottom Line: Secure Deployment with Confidence

MongoDB is on the front line of security. Security practitioners will appreciate the depth and range of encryption choices offered by MongoDB, as well as the engineering effort invested in features like client-side field-level encryption.

- Consider diving into more detail by downloading a white paper on MongoDB security architecture.
- Learn about MongoDB Atlas and its security configuration on the major public clouds by exploring the Trust Center and downloading a paper on MongoDB Atlas Security Controls.
- Discover how MongoDB enables compliance with regulations such as GDPR and CCPA.
- Watch a webinar on a new approach to data privacy with MongoDB client-side field-level encryption.

Get Started with MongoDB Atlas

MongoDB Atlas offers built-in security features for your database, from the start.

Get Started Free

About

Careers

Investor Relations

Legal

GitHub

Security Information

Trust Center

Connect with Us

Support

Contact Us

Customer Portal

Atlas Status

Customer Support

Manage Cookies

Deployment Options

MongoDB Atlas

Data Basics

Vector Databases

Enterprise Advanced

Community Edition

NoSQL Databases

Document Databases

RAG Database

ACID Transactions

MERN Stack

MEAN Stack