International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Using MongoDB to Understand the Underlying Methods Techniques Encryption in NoSQL database

Shahida B[a]

[a],*CSE Department, PDIT, HospetIndia*

DOI: *https://doi.org/10.55248/gengpi.2022.3.9.23*

ABSTRACT

There is a need for appropriate encryption methods, which must be adhered by the concerned parties, in order to provide a high level of security for the secret data. In this study, an examination of the different encryption algorithms and their performance in the management of private data with authentication, access control, secure configuration, and data encryption is presented. It includes enhancing MongoDB's level-based access protected model and adding privacy keys for security and monitoring purposes respectively. The NoSQL data stores, also known as highly compressed data on non-relational database management systems, which provide support for data management of internet user programs, do not now offer this service. Data security, encryption techniques, MongoDB, and NoSQL are some of the keywords that come to mind.The proper encryption methods used to various crucial data fields provide data protection without adversely affecting the database's performance in terms of speed or memory usage. Due to the use of appropriate encryption techniques, this is achievable. Since document-oriented and unstructured data are generally kept in databases, it is frequently used to handle various sorts of data. Examples of such databases include MongoDB, Cassandra, CouchDB, Redis, Hyper-table, and others. Since they are open source, there is a critical necessity to offer good security and protect the user's private information when it is in transit or at rest. There is a need for a single solution that can improve the security of data transfer by providing an improved encryption method that speeds up processes and uses less memory when maintaining databases.

Keywords:NoSQL database, MongoDB, CouchDB, Column Database, Key-value Database, Big Data, Big Data Analytics, RDBMS, and NoSQL

1. Introduction

Databases based on documents are what MongoDB offers. MongoDB is a non-relational database that allows documents to hold data of any kind, in contrast to traditional relational databases. However, the present MongoDB solutions provide very little protection for users' privacy and security. In this, we presented a privacy access policy, which involves obtaining some credential from the user and encrypting it. This ensures a high level of security for sensitive information pertaining to the user while maintaining a high level of speed in MongoDB. Therefore, ensuring the safety of one's data has emerged as one of the most important needs for all users who publish their data on any medium. Cryptography is the technique that ensures information security in a variety of domains, including but not limited to computer science, information technology, and electronic commerce. The practice of cryptography refers to the generation of an encrypted output by combining data known as the plain text as an input with a user-defined password or key in order to produce what is known as the cipher text. The key is a series of symbols that controls the cryptographic operations such as encrypting, decrypting, and generating or verifying signatures. These are all examples of cryptographic work. Methods of Encryption In order to ensure that data transmission is carried out in a safe manner, a number of different security algorithms are implemented alongside the information that is carried out. The process of encrypting plain text involves using several mathematical transformation methods in order to change the plain text into an unreadable format known as cipher text. According to the comment made by Kircchoff, the fact that the key is kept secret provides the greater level of security than the encryption technique that is used. In this article, we have conducted research on and studied a variety of encryption methods, including DES, triple DES, RSA, AES, ECC, BLOWFISH, and RC5 techniques. These algorithms are accessible to construct a secret data transfer [1] [2][3].RSA and ECC are asymmetric key algorithms, whereas the other cryptographic techniques are symmetric key algorithms. These encryption algorithms are listed in alphabetical order. In symmetric algorithms, the sender and receiver use the same key for both encryption and decryption, while in asymmetric key algorithms, two keys are used: a public key for encryption and a private key for decryption. symmetric algorithms are more secure than asymmetric key algorithms.Analysis and Design Make use of the

* *Corresponding author.*

E-mail address: *shahida@pdit.ac.in*

query optimizer to decide which indexes should be suggested in the end. Our strategy involves the creation of virtual indexes, which eliminates the need for any modifications to be made to the database. Using the method on a database that is document-based rather than using SQL. In a typical scenario, there are two users involved: one user obtains information from the other user, and either person may provide the information that was sought. Finally, there is a clash between the sharing of information and protecting individuals' privacy. while the sensitive data has to be kept secret since the owners could be eager to divulge it or might be compelled to do so[4][5].The overarching strategy for the regulation of access control into NoSQL data storage while keeping privacy in mind is a very essential aim. Users are able to carry out operations in order to get access to resources for which they have the necessary authorization. Users and roles may each have their own purpose authorizations attached to their accounts. The data storage and network transfer type for documents in MongoDB are both straightforward and lightning fast.

2. Techniques of Applying MongodB:

The procedures involved with MongoDB BSON is the format that Mongo DB uses to store its data. The server has many databases, each of which contains a significant number of collections. In a relational database, they function similarly to tables. For the purpose of modeling our data, we simply require a single collection. After inputting some data, we would see BSON returned as the representation of our data if we questioned the Post collection using the shell. Flow of data Step one is to start the mongo server from the command prompt, and then go to the bin directory, which is where the mongo server starts the port [11][12] [13].The mongo server will then be started by the monog.exe program. Step 2: The next step requires the client to verify themselves by logging into the system using their user ID and password. The application server determines whether the client has been granted authorization before deciding whether to provide the client access to the database. Step Three: The third step provides two different sorts of access from which we may upload images with access control as well as other types of files. Additionally, this grants access to the admin panel. Step 4: In order to upload a picture, the necessary parameter is retrieved from the client, and then an algorithm is used to build an encrypted key. After that, the file is chopped up into bits and saved on the mongo server. Step 5: In order to perform an insertion operation, the application server must first save the encrypted key for the data in one collection of the database. Next, it must get the encrypted key for the data from another collection of the mongo database. Step 6: The identical procedures are carried out for each of the other file formats; however, the output of these operations is directly a document type. The process of encryption On the one hand, private information must be kept secret, but on the other, the owners of the data may be ready to reveal it (or they may be compelled to do so). In NoSQL data storage, achieving a generic method to the rule of security is an extremely significant aim. [3] We suggested a method to safeguard data or a message while maintaining its validity and integrity. Their method involves encrypting the confidential communication prior to beginning the process of inserting it. The concealed message is encrypted using a miniature algorithm that makes use of a secret key, and the DCT method is employed for both embedding and extracting the file. [4] explored a variety of encryption schemes, including the RSA (Rivest Shamir and Adleman) Algorithm, the Digital Signature Algorithm, the Diffie–Hellman Algorithm, the Data Encryption Standard, and the Advanced Encryption Standard (Advanced Encryption Standard). By using various real-time encryption strategies, the primary security objectives, which include secrecy, integrity, authentication, and non-repudiation, may be attained. Each method has its own set of applications, and some of them could even be appropriate for the specific setting in question in order to achieve an elevated level of safety. A very essential goal to work toward is developing a generic approach to the rule of privacy-aware access control [5][6][7].

3. Literature Review

Security measures for non-relational database systems (NOSQL). Comparing the sharding architecture of various existing databases, such as MongoDB, Redis, HBase, Cassandra, CouchDB, and Couch base, on the basis of defined assessment criteria was done in order to analyze the various security features offered by NoSQL databases and improve the security controls of various NoSQL databases. Specifically, this was done to improve the security controls of various NoSQL databases. An evaluation criterion that is proposed here is one that is made up of many different types of safety measures. In order to increase the security controls of a variety of The authors did a comparison of the NoSQL databases based on the evaluation criteria, which included authentication, Access Control, Secure Configuration, data Encryption, and Auditing. According to the findings of their investigation, MongoDB satisfies just one of the evaluation criteria with a high level of support, and that is access control. The other evaluation criteria are only satisfied to a low or medium degree[8][9][10].MONGODB Server and Safety Measures The front end of the MongoDB server communicates with many MongoDB clients via the exchange of messages. Mem functions as a proxy between a MongoDB server and its clients, supervising and, if necessary, modifying the flow of messages that are sent back and forth between the parties involved. In the event that the message being intercepted encodes a query, it will create the query in such a manner that it will only be able to access documents for which the policies that have been defined are met. The incorporation of data into a MongoDB installation is a simple process that only needs a basic setup to be carried out. System administrators are not needed to engage in any kind of programming. In addition, Meme was developed to be compatible with a wide variety of MongoDB drivers as well as various MongoDB versions.

Experiments that were run on a MongoDB server of a scale that is really applicable yielded a minimal enforcement overhead that did not in any way impact the query usability. The researchers Saurabh Singh et al. [14] highlighted the security flaws that are present in Mongo DB databases and suggested several cryptographic solutions that use elliptic curve and RSA for data encryption and decryption in order to cut down on the number of security breaches. They devised a hybrid protocol design, in which the client makes a request for server authentication, the SSH protocol employs RSA for authentication, and in parallel, it uses ECC to ensure integrity and secrecy for the transaction of data. The performance of ECC is contingent on the accuracy of the calculation, often known as the elliptic curve discrete logarithm problem or scalar multiplication. The new security protocol has been developed with the goal of providing increased safety [15].It is a hybrid of symmetric and asymmetric cryptography approaches brought together into one system. Integrity, confidentiality, and authentication are the three fundamental tenets of cryptography that are supported by the protocol. These three primitives may be accomplished with the assistance of ECC, Dual-RSA, and Message Digest MD5, and based on the findings of these algorithms, the amount of time needed for encryption and decryption using ECC is much less than that required by RSA and its upgraded version. They have also used the ECC approach in order to generate the secure shell while data was being sent across the communication channel. ECC additionally guarantees the user's privacy and data's integrity. It is thought that elliptic curves may give high security with reduced key sizes, which is something that is extremely beneficial in a lot of different applications. On the basis of the security problems, namely data file encryption, client/server authentication/encryption, inter cluster encryption and script injection and Denial of Service attacks, comparisons were made between the two sets. According to the findings of their investigation, MongoDB, CouchDB, and Cassandra are the databases that are safe from data capturing and sniffing during the communication from the servers, whereas Redis and Hyper-tables are safe from attacks that are launched by users of the internet.

The researchers discovered that none of these databases encrypt their data files, thus they proposed several helpful solutions, such as encrypting sensitive data at the application level via the usage of stun tunnels, in order to facilitate safer communication. The Encryption of Mongo DB The only kind of assaults that are impossible in a Mongo DB database are denial of service attacks (also known as DoS attacks). It is recommended that the program should execute data encryption at the application level itself before recoding sensitive data such as passwords and credit card information in order to prevent the data from being hacked. Running MongoDB in standalone mode or replica-set mode offers a higher level of security compared to running it in shared mode due to the activation of authentication using a pre-shared secret [21][22] [23] Despite this, the pre-shard secret can be cracked by hackers who have access to the system data. As a result, in order to strengthen the safety of the key file, the permissions on an operating system level should be adjusted accordingly (e.g. using chmod command). In the same manner, according to the author's statement, we are able to prevent this kind of attack on MongoDB by terminating the following symbols: (:), (), and (). This will prevent the attacking input from getting into the web server, which is the frontage of the database server. In other words, we will be able to stop the attack. Due to these factors, the developers had to compose an additional script in order to identify and get rid of these extra symbols before they can enter the database. Matthew Trudeau and colleagues [16] spoke about the risk of hacking in NoSQL database technology. They are primarily concerned with the Mongo DB database and the security features that are pre-installed on it. These features include authorization, authentication, and TLS/SSL encryption. They emphasized how important it was to make use of the built-in security mechanisms; if not, severe security risks would be tried on the personal data, like the assault that took place in January 2017[17][18][19][20].The Features of MongoDB Standards for data encryption: The current version of Mongo DB (version 3.4) comes equipped with built-in features that are intended to provide authentication, authorization, encryption, auditing, network exposure, injection prevention, and other similar services; however, all of these features are ineffective because they slow down the database. The Advanced Encryption Standard (AES) when operating in Cipher Block Chaining mode is one of the encryption standards that Mongo DB supports.

The other encryption standard that Mongo DB supports is called AES256-CBC. In addition, MongoDB supports the AES256-GCM encryption algorithm, which is sometimes referred as as Galois/Counter Mode. The process of encrypting requires both master keys and database keys. The data stored in the database is encrypted using the database keys, and then the master key is used to encrypt the database keys. In addition, the authors' research revealed that MongoDB does not provide any built-in functionality for application-level encryption. The documentation for MongoDB proposes either implementing a bespoke encryption/decryption function or making use of a solution developed by one of their partners [18] in order to encrypt each field or document. In addition, MongoDB is compatible with transport encryption protocols like TLS and SSL, which are used to encrypt network communication. The implementation of TLS/SSL makes use of Open SSL libraries, only employing SSL ciphers that use a key that is at least 128 bits in length [19], presented an enforcement monitor called Mem (MongoDB enforcement monitor) to implement security by acting as a proxy between the MongoDB user and server and enforce access control [20], proposed a survey on various schemes for database encryption and the future need for the complete solution of providing a more securely protected environment and so on.

4. Summary

A Multiple Document Interface (MDI) application may include numerous child windows inside it. This window will display the very first time the user interacts with the system. Applications that use a single document interface (SDI) only allow for the manipulation of a single document at a time. Notepad is an example of an application that uses a Standard Document Interface, while Visual Studio is an example of an application that uses a Multiple Document Interface (MDI) [15] [16] [17] [18]. The Window menu option in MDI applications allows users to navigate between the many windows and documents that are open. After the data has been stored, the next step is to read data from the database table, then save it as a bitmap once again, and finally see the bitmap on the form. An picture may be seen by using the Graphics tool. Draw Image technique or utilizing an image box This study also evaluates the major NoSQL databases such as MongoDB, Cassandra, Redis, CouchDB, Hypertable, etc. as a result of the fact that the data they use are stored in an unstructured manner and may be found in the form of papers, emails, and so on. The majority of NoSQL databases are vulnerable to external security assaults launched by malicious or accidental intruders, and researchers have discovered that these databases are lacking in many key areas such as authentication, script injection, denial of service attacks, and others. Almost none of the NoSQL databases provide appropriate encryption and decryption procedures to authenticate user data while it is being stored locally or being sent elsewhere. According to the results of the survey, it is abundantly clear that there is a serious need to handle confidential data safely without any loss while it is in transit or vulnerability to sniffing or injection attacks. This can be accomplished by providing a suitable secured environment and adhering to encryption and compression techniques. Implementing the SHA-3 algorithm or adopting the business version of the MongoDB database are both potential alternatives for warding off hacking attempts while working with that database. In order to prevent data from being snatched in transit, application-level encryption has to be put into place.

Implementing all of the database's built-in security features is an absolute must for every database that aspires to be successful, and there are specific recommendations to encrypt all of the fields and adhere to the best practices in order to provide database users a risk-free environment. Security breaches that have happened since the beginning of 2017 may be traced back to the problematic selection of default settings and the failure to follow recommended practices like opting out. The Purpose concepts and associated offer tools to govern access at the document level on the basis of purpose and key-based policies. In order to put the suggested security measure into effect, a compliance monitor has been developed. It fulfills the role of a middleman between a MongoDB user and a MongoDB server, and it regulates access by keeping an eye on the flow of messages that are being sent back and forth between the two parties. In addition, we want to expand the method that was provided in order to enable several NoSQL data stores. The strengthening of data security is one of the major issues that must be considered while transmitting data between users. Even if the use of data in an unstructured format has expanded in a variety of domains, there is still a substantial need for the provision of access criteria such as authentication, access control, data encryption, secure configuration, and auditing. The right encryption techniques applied to different essential fields of data provide data protection without negatively impacting the performance of the database in terms of speed or memory consumption. This is possible because suitable encryption methods are used. NoSQL databases, which include MongoDB, Cassandra, CouchDB, Redis, Hyper-table, and others, are often used to manage document-oriented and unstructured data since these types of data are typically stored in the databases. Because they are open source, there is a major need to provide strong security and preserve the user's sensitive data without allowing any manipulation when the data is either at rest or in transit. There is a need for a single solution that can increase the safety of data transfer by offering an enhanced encryption approach that reduces the amount of time a process takes and the amount of memory that is used while managing databases.

REFERENCES

[1] Abramova, V., & Bernardino, J. (2013, July). NoSQL databases: MongoDB vs Cassandra. In Proceedings of the international C* conference on computer science and software engineering (pp. 14-22).

[2] Ali, W., Shafique, M. U., Majeed, M. A., & Raza, A. (2019). Comparison between SQL and NoSQL Databases and Their Relationship with Big Data Analytics. Asian Journal of Research in Computer Science, 4(2), 1-10

[3] Becker, M. Y., & Sewell, P. (2004, June). Cassandra: Flexible trust management, applied to electronic health records. In Proceedings. 17th IEEE Computer Security Foundations Workshop, 2004. (pp. 139-154). IEEE.

[4] Berg, K. L., Seymour, T., & Goel, R. (2013). History of databases. International Journal of Management & Information Systems (IJMIS), 17(1), 29-36.

[5] Bjeladinovic, S., Marjanovic, Z., & Babarogic, S. (2020). A proposal of architecture for integration and uniform use of hybrid SQL/NoSQL database components. Journal of Systems and Software, 168, 110633.

[6] Chandra, D. G. (2015). BASE analysis of NoSQL database. Future Generation Computer Systems, 52, 13-21.

[7] Chen, J. K., & Lee, W. Z. (2019). An introduction of NoSQL databases based on their categories and application industries. Algorithms, 12(5), 106.

[8] Cuzzocrea, A., & Shahriar, H. (2017, December). Data masking techniques for NoSQL database security: A systematic review. In 2017 IEEE International Conference on Big Data (Big Data) (pp. 4467-4473). IEEE.

[9] de Oliveira, V. F., Pessoa, M. A. D. O., Junqueira, F., & Miyagi, P. E. (2021). SQL and NoSQL Databases in the Context of Industry 4.0. Machines, 10(1), 20.

[10] Deka, G. C. (2013). A survey of cloud database systems. It Professional, 16(2), 50-57. IEEE.

[11] Di Martino, S., Fiadone, L., Peron, A., Riccabone, A., & Vitale, V. N. (2019, June). Industrial Internet of Things: Persistence for Time Series with NoSQL Databases. In 2019 IEEE 28th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE) (pp. 340-345). IEEE.

[12] dos Santos Ferreira, G., Calil, A., & dos Santos Mello, R. (2013, December). On providing DDL support for a relational layer over a document NoSQL database. In Proceedings of International Conference on Information Integration and Web-based Applications & Services (pp. 125-132).

[13] Gessert, F., Wingerath, W., Friedrich, S., & Ritter, N. (2017). NoSQL database systems: a survey and decision guidance. Computer Science-Research and Development, 32(3), 353-365.

[14] Guimaraes, V., Hondo, F., Almeida, R., Vera, H., Holanda, M., Araujo, A., ... & Lifschitz, S. (2015, November). A study of genomic data provenance in NoSQL document-oriented database systems. In 2015 IEEE International Conference on Bioinformatics and Biomedicine (BIBM) (pp. 1525-1531). IEEE.

[15] Rodriguez, K. M., Reddy, R. S., Barreiros, A. Q., & Zehtab, M. (2012, June). Optimizing Program Operations: Creating a Web-Based Application to Assign and Monitor Patient Outcomes, Educator Productivity and Service Reimbursement. In DIABETES (Vol. 61, pp. A631-A631). 1701 N BEAUREGARD ST, ALEXANDRIA, VA 22311-1717 USA: AMER DIABETES ASSOC.

[16] Kwon, D., Reddy, R., & Reis, I. M. (2021). ABCMETAapp: R shiny application for simulation-based estimation of mean and standard deviation for meta-analysis via approximate Bayesian computation. Research synthesis methods, 12(6), 842–848. https://doi.org/10.1002/jrsm.1505

[17] Reddy, H. B. S., Reddy, R. R. S., Jonnalagadda, R., Singh, P., & Gogineni, A. (2022). Usability Evaluation of an Unpopular Restaurant Recommender Web Application Zomato. Asian Journal of Research in Computer Science, 13(4), 12-33.

[18] Reddy, H. B. S., Reddy, R. R. S., Jonnalagadda, R., Singh, P., & Gogineni, A. (2022). Analysis of the Unexplored Security Issues Common to All Types of NoSQL Databases. Asian Journal of Research in Computer Science, 14(1), 1-12.

[19] Singh, P., Williams, K., Jonnalagadda, R., Gogineni, A., &; Reddy, R. R. (2022). International students: What's missing and what matters. Open Journal of Social Sciences, 10(02),

[20] Jonnalagadda, R., Singh, P., Gogineni, A., Reddy, R. R., & Reddy, H. B. (2022). Developing, implementing and evaluating training for online graduate teaching assistants based on Addie Model. Asian Journal of Education and Social Studies, 1-10.

[21] Sarmiento, J. M., Gogineni, A., Bernstein, J. N., Lee, C., Lineen, E. B., Pust, G. D., & Byers, P. M. (2020).Alcohol/illicit substance use in fatal motorcycle crashes. Journal of surgical research, 256, 243-250.

[22] Brown, M. E., Rizzuto, T., & Singh, P. (2019). Strategic compatibility, collaboration and collective impact for community change. Leadership & Organization Development Journal.

[23] Sprague-Jones, J., Singh, P., Rousseau, M., Counts, J., & Firman, C. (2020). The Protective Factors Survey: Establishing validity and reliability of a self-report measure of protective factors against child maltreatment. Children and Youth Services Review, 111, 104868