INDUSTRIAL GANTRY OVERHEAD CRANE

C BEHERA[1], B MALLA[2,] S. CHANDAN[3],C BEHERA[4], K K DASH[5]

Department of Mechanical Engineering, Raajdhani Engineering College

Abstract— **The fast development and use of Internet, like the social networks, has to handle huge amounts of data. The social networks example the Facebook required high availability and scalability, billions of writes/hour and high write throughput. RDBMS does not support such features. To solve this problem of processing and storing large amounts of data robustly and efficiently, various kinds of distributed database systems have been developed and designed. Cassandra database has the features to handle very large amount of data. As Cassandra stores sensitive data, the security and management of data is important.**

This paper gives comparative performance analysis of Cassandra with regards to encryption standards asymmetric and symmetric considering various parameters such as Write Time, Read Time and Throughput. Experiments results are given to analyses the effectiveness of symmetric and asymmetric algorithms on performance of Cassandra.

Keywords— Asymmetric Algorithm, Symmetric Algorithm, Cassandra.

I. INTRODUCTION

Cassandra is a distributed storage system which manages huge amount of structured data. It has features like high scalability and availability with fault - tolerant capability. Cassandra was designed to boost the Facebook inbox search feature by Prashant Malik and Avinash Lakshman. The Facebook inbox search feature required to handle billions of writes per day and provide high scalability to handle huge number of users in different geographic locations. Cassandra has a customized data model as per the user requirements. Cassandra architecture is based on peer-to-peer distributed system. Each node captures write activity to ensure data durability with the help of commit log. The data is also written to memTable and when memory structure memTable is full the data is also written to SSTable which is a data file on disk. Partition and replication of all writes are automatically done throughout the cluster. Cassandra being a row-oriented database with peer-peer architecture allows any user to access any node in any data center and data is accessed using the CQL language. Any node in the peer-peer architecture can be selected for the client read or write requests. The selected node to which the client connects serves as the coordinator for that particular client operation. Between the nodes that own the data being requested and client application the coordinator acts as a proxy. Based on how the cluster is configured the coordinator decides which nodes in the peer-to- peer ring should get the request [12].

Cassandra is used in application where user related sensitive information is stored and hence the privacy and confidentiality of this data is important. Thus this raise the concern for the security provided by these systems [11].

For security a science known as Cryptography is most widely used because of its various key features for secure data like security and privacy, identification and authentication, trust and verification. There are many algorithms developed to secure the data. There are three types of algorithm:[13].

- Secret Key Cryptography (SKC)/ Symmetric Key Encryption: This type of algorithm for both encryption and decryption uses a single key.
- Public Key Cryptography (PKC)/ Asymmetric: This type of algorithm uses one key called as public key for encryption and another key called as private key is used for decryption.
- Hash Functions: are algorithms that, use no key. A fixed-length hash value is computed using a mathematical transformation based upon the plaintext.

This paper examines a method for evaluating performance of Cassandra with regards to asymmetric and symmetric algorithms. Different size of data blocks are used for a comparative analysis for those encryption algorithms. The paper is organized as follows, Section 1 presents introduction. Related work is presented in Section 2. The experimental design is presented in Section 3. In Section 4 result analysis is performed. Finally Section 5 provides the conclusion and future work.

II. RELATED WORK

Related work includes papers aimed on significant research in Cassandra database system and cryptography algorithms [11].

A study in [2] describes how Cassandra was introduced, implemented, and operated as a storage system. The Cassandra features like high performance, scalability and applicability are described. The experimental results show a very high write throughput for Cassandra. In [1] the study reviews the security features and the main functionality of two of the most popular non-relational databases (NoSQL: Cassandra and MongoDB). Both systems have the problem that include weak authentication in client-server and between server and lack of data files encryption. The study [3] gives a comparison of performance analyses between three databases, two NoSQL databases (MongoDB and Cassandra) and SQL database (PostgreSQL) with regards to sensor database on virtual and physical server with four operations a single read, multiple reads in one statement ,a single write and multiple writes in one statement. In [4] the study describes a monitoring tool for performance analysis of Cassandra.

The study [5] was conducted to analyze different encryption techniques in terms of memory usage, response time and efficiency. In [6] a study which compares the various cryptographic algorithms. The result is evaluated on different video files and a comparison of encryption and decryption time of different cryptographic algorithms is given. The study [7] compares three encryption algorithms AES, Blowfish and DES and the results of comparison is based on parameters such as memory required, throughput and execution time. The study concludes that the best performing algorithm is Blowfish. In [8] a study is conducted for security analysis of Blowfish based on correlation coefficient and avalanche criteria. The results give a good non-linear relation between plaintext and ciphertext for correlation coefficient and good avalanche for Blowfish algorithm. In [9] a study two symmetric encryption algorithm (Blowfish and Rijndael) are analyzed based on experimental results Blowfish performance is better for all test cases. From the study [10] a reviews performance of four symmetric key algorithms: Blowfish, AES 3DES and DES based on the parameters like CPU time, encryption and decryption time, key size and different data block size. The experimental result shows good result for Blowfish in comparison to the other algorithms.

III. Experimental Design

Different number of records is used to conduct the experiments, where a Comparisons analysis of the results of the selected algorithms asymmetric and symmetric is performed to analyze the performance of Cassandra. The proposed algorithm is a symmetric algorithm Blowfish which uses a variable key size which is generated randomly. The standard Blowfish algorithm uses variable key (32 bits to 448 bits), the random key generation algorithm in the proposed algorithm generates key size greater than 448 bits.

A. Evaluation Parameters

Performance of Cassandra is evaluated considering the following parameters [14].

- Write Time

The time taken to encrypt and insert data records into the database.

- Read Time

The time taken to read data records from the database.

- ThroughPut

The throughput is calculated as the total plaintext in bytes encrypted divided by the write/read time.

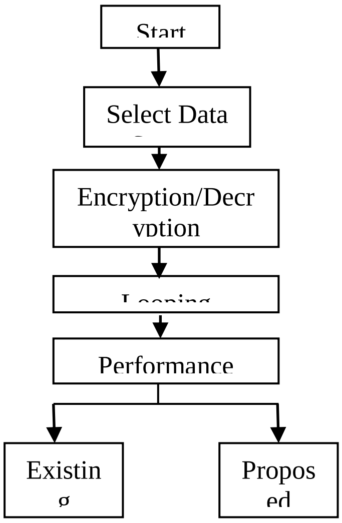

Fig. 1 Proposed Architecture

IV. Experimental Results And Analysis

The performance is analyzed with regards to the parameters: Write Time, Read Time and Throughput. Experimental result for Cassandra performance with respect to encryption algorithm is shown in Table 1 and Table 2 using different number of records. Analyzing the Table 1 and Table 2 it is noticed that the write and read time taken by asymmetric algorithm is higher compare to the write and read time taken by symmetric algorithm. The table also shows that the throughput of symmetric algorithm is higher than asymmetric algorithm.

TABLE 1 PERFORMANCE WITH RESPECT TO ASYMMETRIC ALGORITHM.

Sr. No	Data Records	Write Time (ms)	Read Time (ms)	Throughput (bytes/sec)
1	100000	980	690	946436.29
2	50000	483	324	600562.63
3	5000	431	316	992985.16
4	1000	412	267	509355.41
5	500	231	115	45426.16

TABLE 2 PERFORMANCE WITH RESPECT TO SYMMETRIC ALGORITHM.

Sr. No	Data Records	Write Time (ms)	Read Time (ms)	Throughput (bytes/sec)
1	100000	535	552	558602.68
2	50000	449	312	633555.22
3	5000	402	287	999517.18
4	1000	336	167	527772.92
5	500	228	133	460192.77

Fig. 1. gives a graphical representation of the time taken for writing different number of records with respect to asymmetric and symmetric algorithm. Fig. 2. gives a graphical representation of the time taken for reading the different number of records with respect to asymmetric and symmetric algorithm.

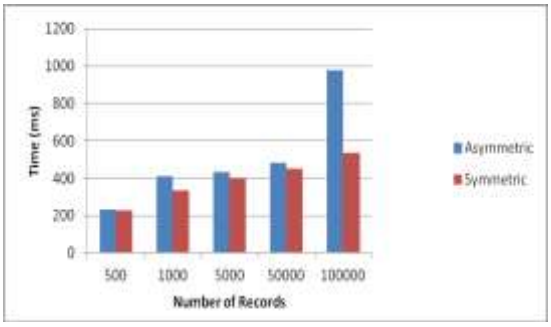

Fig. 1. *Write Time*

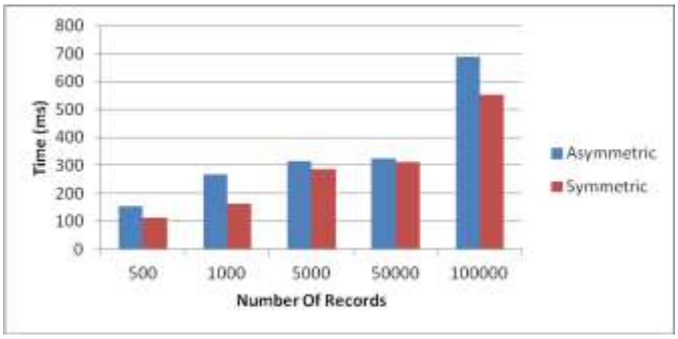

Fig. 2. Read Time

Table 3,4 and 5 shows the write and read time taken by symmetric algorithm when variable key size is used. Analyzing the table 3, table 4 and table 5 it is noticed that the small key size take less time to write and read as compared to large key size.

TABLE 3 PERFORMANCE WITH RESPECT TO SYMMETRIC ALGORITHM FOR KEY SIZE 4 BYTES.

TABLE 4 PERFORMANCE WITH RESPECT TO SYMMETRIC ALGORITHM FOR KEY SIZE 70 BYTES.

TABLE 5 PERFORMANCE WITH RESPECT TO SYMMETRIC ALGORITHM FOR KEY SIZE 80 BYTES.

Sr. No	Data Records	Write Time (ms)	Read Time (ms)	Throughput (bytes/sec)
1	100000	980	690	946436.29
2	50000	958	699	863442.01
3	5000	880	714	639534.10
4	1000	336	597	834307.79

Fig.3. gives a graphical representation of the time taken for writing different number of records with respect to different key size for symmetric algorithm. Fig.4. gives a graphical representation of the time taken for reading the different number of records with respect to different key size for symmetric algorithm.

Sr. No	Data Records	Write Time (ms)	Read Time (ms)	Throughput (bytes/sec)
1	100000	843	594	958148.05
2	50000	448	420	947284.76
3	5000	402	556	999517.18
4	1000	155	543	761365.74

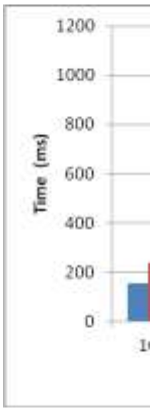

Fig.4

. Write Time

Sr. No	Data Records	Write Time (ms)	Read Time (ms)	Throughput (bytes/sec)
1	100000	970	690	978529.65
2	50000	965	699	962287.03
3	5000	535	592	558602.68
4	1000	240	588	709163.9

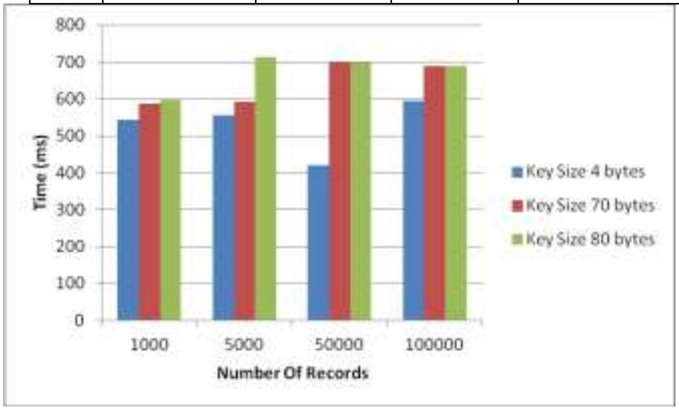

Fig.5. Read Time

V. CONCLUSION AND FUTURE WORK

The performance of Cassandra is analyzed for the existing algorithm and proposed algorithm. Simulation results shows that the write and read time taken by asymmetric algorithm is higher compare to the write and read time taken by symmetric algorithm. The result also show that the throughput of symmetric algorithm is higher than asymmetric algorithm. Simulation result for write and read time taken by symmetric algorithm when variable key size shows that the small key size take less time to write and read as compared to large key size.

In the future, the performance analysis can be done with other encryption algorithm which can help to use the advantage of each algorithm in a combined fashion. The future work also aims to analyze performance of Cassandra with regards to images dataset.

Acknowledgment

We would like to thanks all the authors whom we referred for giving their suggestions and making the material available for us to refer.

References

[1] J. Guillory, J. Spiegel, M. Drislane, B. Weiss, W. Donner, and J. T. Hancock, “Security Issues in NoSQL Databases,” 2011 International Joint Conference of IEEE TrustCom-11/IEEE ICESS-11/FCST-11.

[2] PrashantMalik, AvinashLakshman, “Cassandra - a decentralized structured storage system,” in The 3rd ACM SIGOPS International Workshop on Large Scale Distributed Systems and Middleware (LADIS 09), October 2009.

[3] Jan Sipke van der Veen, Bram van der Waaij , Robert J. Meijer, “Sensor Data Storage Performance: SQL or NoSQL, Physical or Virtual ,” 2012 IEEE Fifth International Conference on Cloud Computing.

[4] PrasannaBagade, Ashish Chandra, and Aditya B.Dhende,“Designing Performance Monitoring Tool for NoSQLCassandra Distributed Database” in 2012 International Conference on Education and e-Learning Innovations.

[5] Malik Sikander Hayat Khiyal, Aihab Khan, and KhansaShabbir,“Performance Evaluation of Encryption Techniques for Confidentiality of Very Large Databases” in International Journal of Computer Theory and Engineering, Vol. 3, No. 6, December 2011.

[6] S. Pavithra and Mrs. E. Ramadevi, “Performance Evaluation of Symmetric Algorithms”, in Journal of Global Research in Computer Science, 3 (8), August 2012, 43-45.

[7] A.Ramesh and .A.Suruliandi, “Performance Analysis of Encryption Algorithms for Information Security”, in 2013 International Conference on Circuits, Power and Computing Technologies [ICCPCT-2013].

[8] Ashwak ALabaichi,Faudziah Ahmad and RamlanMahmod, “Security Analysis of Blowfish algorithm”, in ISBN: 978-1-4673-5256-7/13 ©2013 IEEE.

[9] M.Anand Kumar and Dr.S.Karthikeyan, “Investigating the Efficiency of Blowfish and Rejindael (AES) Algorithms”, in I.J. Computer Network and Information Security, 2012, 2, 22-28.Published Online March 2012 in MECS.

[10] PratapChnadraMandal,“ Superiority of Blowfish Algorithm”, in International Journal of Advanced Research in Computer Science and Software Engineering 2(9), September - 2012, pp. 196-201.

[11] Trupti B. Bhosale, Prof. G. P .Potdar, "Enhancing Blowfish Cryptography Technique for Cassandra", in Volume 6 in 2011 International Conference on Computational and Information Sciences coimbatore institute of information technology international journal ISSN: 0974 – 9640..