International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-8 Issue-6, March 2020

A Novel Encryption Scheme for Securing Data in HDFS Environment Inspired By DNA

Shahrabanu Haidari , Amiya Kumar Dash, Jitendra Kumar Rout

Abstract: ***Today, the majority of industries used Hadoop for processing their data. Hadoop is an open-source and programming based framework that has many components. One of them is HDFS (Hadoop Distributed Files System) that is used to stored data. Hadoop by default does not have any security mechanism. According to the previous study authentication, authorization, and Data encryption are the principal techniques to enhance the security in HDFS. As huge volume of data is stored in HDFS, encryption of massive data will consume more time and need more resources for operations. In this paper we have developed one DNA based that used confusion and Diffusion for securing data in HDFS. This proposed algorithm is efficient as compared to other encryption algorithm.***

Keywords **:** ***Hadoop, Security, Big Data, Encryption .***

I. INTRODUCTION

Big Data is huge data, this definition to specify the Big Data not perfect but another glance Data specification that have to mention them. Here we refer to three important characteristics.

Volume: The voluminosity is one of the defining features of Big Data.

Velocity: A key factor is the rate of data generation.

Variety: In world we have different format of data: Structure, unstructured and semi structure[11].

Hadoop is a software platform that enables running of big data application and is implemented in Java. Data are stored in the Hadoop distributed file system as blocks which are the smallest unit of HDFS. During the storing of data into HDFS, large files broken down into small chunks. These chunks are called blocks.

Manuscript received on March 15, 2020.
Revised Manuscript received on March 24, 2020.
Manuscript published on March 30, 2020.

* Correspondence Author

Shahrabanu Haidari*, M.Tech,computer science and security information brunch,School of Computer Engineering, Kalinga Institute of Industrial Technology, Bhubaneswar , India.

Amiya Kumar Dash , Assistant Professor in School of Computer Engineering, Kalinga Institute of Industrial Technology, Bhubaneswar , India

Jitendra Kumar Rout, Assistant Professor in School of Computer Engineering, Kalinga Institute of Industrial Technology, Bhubaneswar India.

The size of each block is 128 MB. HDFS includes a master node called Name node, where the orginal data is partitioned and assigned to the data nodes based on defined rules[1]. In definition, for high availability, each data block repeated three times, and

each cluster data node stores a small fragment of the entire data set. Name node manages the meta

data and is aware of which blocks of data belong to which files, And where the capacity for storage is filled. Name node by the help of the heartbeat can realize which data nodes are still working. If the Name node does not receive any heartbeat signal from any data node, it means there is an error and it will remove the failed data node. Name node tries to evenly distribute the data load across the available data nodes. Name node guarantees that the number of copies of information is always accessible[12]. Hadoop has one another important part that is called Map Reduce. Map Reduce same to HDFS, also have master and slave parts and work according to that. Master (Job Tracker) take problem and divided into tasks. Those tasks have been processed by slaves (Task Tracker)[2]. Information is an important asset for all industrial and individual and must be protected from unauthorized access. Security information for protecting data have some security models, Confidentiality, Integrity, and Availability (CIA) is one security model which according that we can take a security mechanism for protecting our information. Data authentication and Data encryption are part of data confidentiality. In Hadoop, also are some security mechanism like Kerberos, Data node, sentry, and Data encryption.

II. RELATED WORK

Typically, Hadoop does not have any security features[4]. We need to used to security algorithms to resolve the security issues[2]. To enhance the security of Hadoop many researchers proposed different algorithms.

Devi et al have discussed Big data issues and focused more on security issues arises in the base layer of Hadoop Architecture[3]. Finding out the issues in HDFS can help us to try to solve these and enhance the security of HDFS. Kadre et al proposed AES-MR(Advanced encryption standard using Map reduce[9]. They have introduced a new technique for encryption of data in HDFS in parallel mode. The period time in which this technique devotes to encryption and decryption is very less and noticeable. Their work is very effective in protecting user's sensitive data.

Tondon et al have used a distributed detection system to detect attacks[6]. They run a cod of Java in Hadoop audit files at the data node. The result of their work is: all suspicious packets will remove and also, the user who is the sender of those packets will block.

Singh Bhathal et al have discussed regarding different vulnerabilities and possible solutions are provided to reducing those vulnerabilities[1]. According to this fact, which each security mechanism invents for a specific environment and Hadoop does not have any security mechanism from itself.

they got results, Hadoop has to upgrade to a new version of itself with all security features, without needs to any installation and configure plugins separately. Young Song et al they have proposed an HDFS data encryption scheme using the ARIA encryption on Hadoop[5]. They designed a variable-length data processing component, that can check the size of the last block, it is 128 bits or not. If the size of the block was 128 bits, data will be encrypted, if the size of the block was less than 128 bits, variable-length will add dummy bits to blocks and then the block will be encrypted.

Balaraju et al have designed one Hadoop protocol (Secure-HDNANode) which answers the purpose of authentication and meta data security[10]. As we know Hadoop does not have any security mechanism. They tried to help this system to reduce Hadoop collapses which eventually improve performance and data security. HDFS is different from a normal Distributed System. Kerberos protocol designed based on the normal Distributed System. HDFS now used Kerberos for authentication and authorization. So this protocol will not be suitable for HDFS.

Security approaches in HDFS:

Authentication, authorization and Data Encryption are the requirements level for security. Overall security architecture of an application depends on each level of security. In HDFS we can use different approaches to achieve to that level of security[6].

Kerberos mechanism: Kerberos is a protocol for network authentication designed as part of the Athena Project at MIT[14]. This uses secrete-key cryptography to provide encryption across the open network.Users proceed with three steps to access services which each involves the exchange of message with the server. Authentication: Each client for connecting to Name Node, first must verify itself with the authentication server and obtain the time-stamped Ticket-Granting Ticket(TGT). Authorization: That client, who could get TGT, asks service tickets from Ticket Granting Server (TGS). Service Request: clients by using service ticket can authenticate itself to the server that provide service. In Hadoop, that server will be name node or job tracker[3].

The Advanced Encryption Standard (AES): Advanced Encryption Standard is one of the popular method of encrypting of data, that introduced by the US government to keep save the sensitive data at storage level[15]. Now this technology became a popular technique in the world. In the AES encryption algorithm, data will encrypt in a fixed size of blocks in rounds with sub-keys, those generated by the key generator. Block cipher is an encryption algorithm that operate on single block of data at one time. In AES encryption algorithm, the size of each block is 128 bits or 16 bytes in length.

Bull Eye: In big data Hadoop, enterprises will store their sensitive data such as credit card numbers, passwords, account numbers, personal details and much more. In Hadoop, a new approach is used these days known as Bull Eye. This algorithm designed to optimize security in Hadoop[13]. Bull Eye use to look at all sensitive data in 360 ° to ensures all data are secure and stored safely. Users can be able to store their sensitive information in a correct manner[6].

AES-MR: Advanced Encryption Standard is an encryption and decryption technique by the map-reduce framework for security in HDFS. The aim of this technique is enhancing security and growing the rate of operations to prevent time-consuming in HDFS Hadoop environment[2].

III. PROPOSED WORK

Enterprises store massive data in Hadoop. We need to secure our sensitive data in the Hadoop Framework. One of the securing level is encryption of data. Encryption of massive data will take more time, and it seems time-consuming. In this paper have porposed one technique that mixed an encryption algorithm inspired from DNA with the MapReduce. Map Reduce process data parallelly. This algorithm is a private-key block cipher that with 128 bits keys encrypt a block of 128 bits plain text inspired by DES and AES. Encryption algorithm comprises three parts: initial part, iteration part, and final part. The key will change to sub-keys by one operation like AES key generator. In this algorithm also used Shannon's principles of Confusion and diffusion to enhance security and complexity of algorithm. It has a step that imitates from the idea of transcription(transfer from DNA to mRNA) and translation (RNA to amino acids)[7].

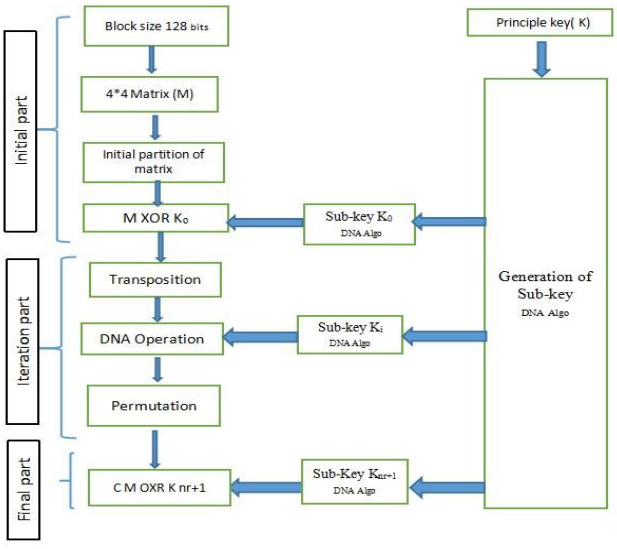

Fig.1.Encryption algorithm inspired by DNA [6].

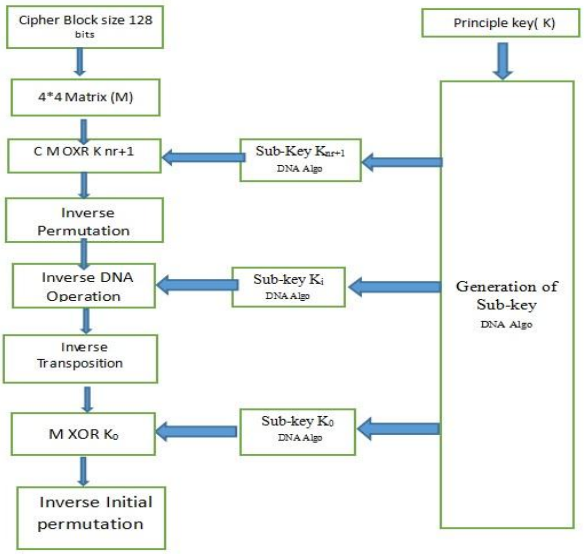

Fig.2. Decryption algorithm inspired by DNA

A. Encryption and Decryption operations with Map Reduce process

The advantage of processing data in parallel processes will be very fast and simple. Map Reduce has two-phase Map function and Reduce function. Map-reduce framework will make a set of map functions and set of reduce functions and splits data to fix size chunks to process by map function. The output of mapper is a key value pair. <block key, object>.

Mapper will changed to key-value pair <block key, object> model and map function read one record in one time from each block and encrypt data by encryption algorithm inspired by DNA as shown in Fig1. Also, the mapper sends data in key-value pairs into the reducer part. Here we have used a simple code for check the if last block of data was less than of 128 bits, so encryption algorithm will add dummy bits to blocks until it became 128 bits and then block will encrypt as other data.

Reducer assemble encrypted data as key-value pair <block id; object> from different mappers and combine it in one HDFS file.

In the proposed work size of each input data is equal to block size and in each data node, one mapper worked. Map function reads one record of each block in one round and encrypts it by encryption algorithm.

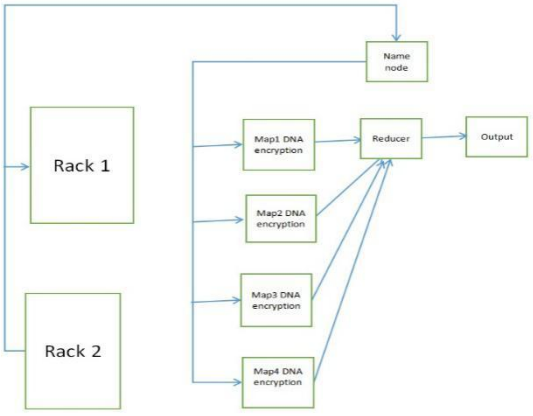

Fig.3. Operation of encrypt data DNA and Map Reduce.

B. Execution of DNA based encryption and decryption algorithm DNA with MapReduce

Encryption:

Encryption data with the encryption algorithm inspired by DNA with Map Reduce. In this work we can with the strength of Map Reduce in processing data in parallel mod, encrypt a huge amount of data in Hadoop. The velocity of operations is significant and also we can enhance the security of data in HDFS.

According to Map Reduce has two-phase Mapper and Reducer, in the mapper, all data will divide into a fixed chunk of data and then encryption operation will do on data. Reducer will combine all encrypted chunks and combined into one HDFS file.

The steps of our work will be these steps:

- Mapper will get data from HDFS as fixed-size chunks.
- The map function comprises the encryption code, that will encrypt data with code in parallel. Map function change each data chunk into encrypted chunk.
- All encrypted chunks will gather by Reducer and combine to a single encrypted file.
- Finally, the single encrypted file will store in HDFS.

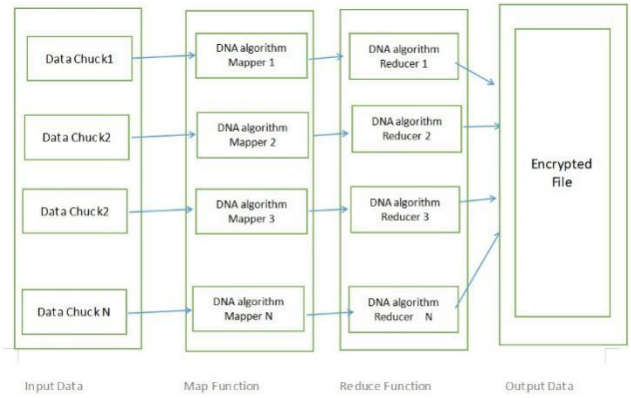

Fig.4. Encryption of Data

Decryption:

The decryption part will be the operation inverse of the encryption part. As the encryption phase, the decryption process also is very important. We just got the result equal to the first plain text. We explain the steps of decryption bellow:

I. The input encrypted file taken from HDFS.
II. This file will be broken to chunks and forward to the Mapper map function.
III. Map function contains the code of decryption and will decrypt data into plain text.
IV. Those decrypt chunks will forward to Reducer to combine into one HDFS file.
V. Finally, the plain text file will store in HDFS.

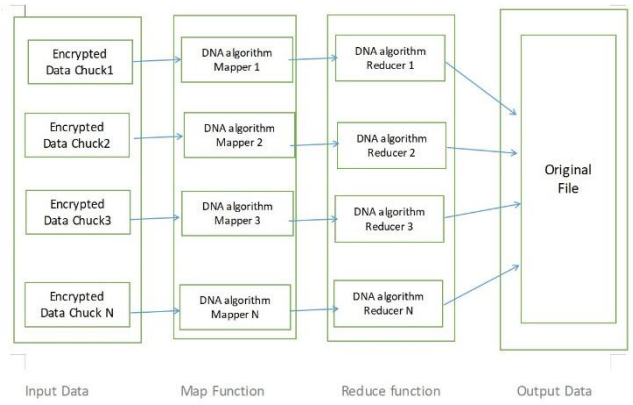

Fig.5. Decryption of Data

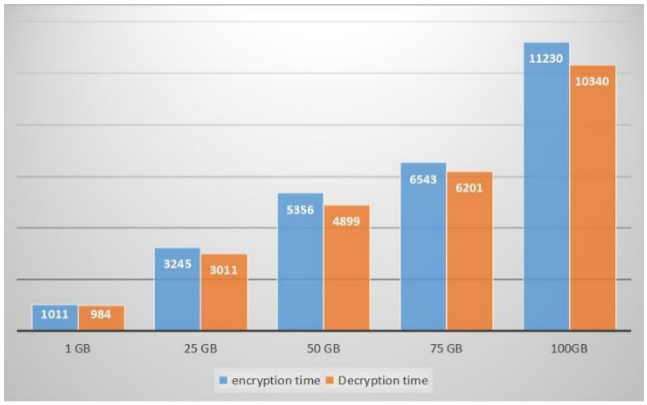

Fig.6. Time of Encryption and Decryption

IV. EXPERIMENTAL RESULTS AND ANALYSIS

We have implemented our work on the Cloudera platform. We have written Java code for encryption and decryption in Eclipse. Then for combining all classes, we exported as a JAR file format. Eventually, we have executed it on Cloudera.

A. Evaluation parameter

According to security model CIA we have to take the mechanisms, techniques and approaches for enhance security in our system. In addition Authentication, authorization and Data Encryption are 3 principles, that we can according them robust security of data in Hadoop. In Hadoop we have Kerberos and Name node for Authentication and Authorization and for encrypting of data it is better to use a proper encryption algorithm for achieving a robust security.

B. Experimental Result

An encryption algorithm inspired by DNA with MapReduce encrypt data in HDFS, Encryption data in HDFS is one effective method to enhance security against the attacks. By Encrypting data attacker could not access to data easily. Here in Fig 6 shown the time of encryption and decryption data in HDFS with map reduce. That is very significant for the speed of operations.

C. Performance of our technique

The entire performance of the algorithm has scaled by the time taken of all steps, encryption, and decryption. Here we can see all the results in table 1 and Fig 6. Thus using encryption data in Hadoop helps to secure the Hadoop system. As we mentioned before, for securing systems according to the security models we have to take different security mechanisms, techniques and approaches to can have robust security. In Hadoop also we have different techniques for security such as Kerberos for authentication and Name node for authorization but at the storage level, also we need to have robust security mechanisms. Using of an encryption algorithm inspired by DNA, that is more secure with the map reduce can be effective. This algorithm by a proper speed can encrypt data.

Table.1 Time of Encryption and Decryption.

Size of data	*Encryption time(Second)*	*Decryption time(Second)*
1 GB	1011	984
25 GB	3245	3011
50 GB	5356	4899
75 GB	6543	6201
100 GB	11230	10340

V. CONCLUSION

In Hadoop, Kerberos is used for authentication and Name Node and Sentry are used for authorization. Hadoop typically does not have any security mechanism at storage level. In this paper we encrypt data at the storage level of Hadoop. We have encrypted data with an encryption algorithm inspired DNA with MapReduce in HDFS. This encryption algorithm is one robust algorithm for encrypting data and a private-key block cipher that with 128 bits keys encrypt a block of 128 bits plain text inspired by DES and AES. The timing plot of the performance shows the acceptable speed for encryption and decryption data in HDFS. Therefore, by encrypting data at storage level in Hadoop by using our proposed algorithm we can robust security in Hadoop.

REFERENCES

1. Bhathal, Gurjit Singh and Singh, Amardeep: Big data: Hadoop framework vulnerabilities, security issues and attacks. In: Array, pp. 100002. Elsevier, India (2019)
2. Kadre, Viplove and Chaturvedi, Sushil: AES–MR: A Novel Encryption Scheme for securing Data in HDFS Environment using MapReduce. In: International Journal of Computer Applications, pp. 12–19. Foundation of Computer Science, India (2015).
3. Saraladevi, B, Pazhaniraja, N, Paul, P.V, Basha,M.S.,& Dhavachelvan, P: Big Data and Hadoop-A study in security perspective. In: Procedia computer science, pp.596- 601. Procedia computer science, India (2015).
4. Jam, Masoumeh Rezaei, et al: A survey on security of Hadoop. In:4th International Conference on Computer and Knowledge Engineering (ICCKE), .IEEE, (2014).

5. Song, Youngho, et al. "Design and implementation of HDFS data encryption scheme using ARIA algorithm on Hadoop." 2017 IEEE International Conference on Big Data and Smart Computing (BigComp). IEEE,(2017).
6. Tondon, Devika, and Monika Khurana: Security of Big Data in Hadoop Using AESMR with Auditing. In:International Journal of Advanced Research in Computer Science and Software Engineering,(2017).
7. Babaei, Majid.: A novel text and image encryption method based on chaos theory and DNA computing. In:Natural computing, pp. 101-107.(2013).
8. Jain, Priyank, Manasi Gyanchandani, and Nilay Khare. "Big data privacy: a technological perspective and review." Journal of Big Data 3.1 (2016): 25.
9. Canbay, Yavuz, Yilmaz Vural, and Seref Sagiroglu. "Privacy preserving big data publishing." 2018 International Congress on Big Data, Deep Learning and Fighting Cyber Terrorism (IBIGDELFT). IEEE, 2018.
10. Balaraju.J,Dr.p.v.R.D.Prasada Rao: enhanced security for hadoop distributed file system by using dna cryptography. In:International Journal of Pure and Applie Mathmatics,pp.8127-8142.(2018).
11. Choudhary, Mehak and Chandra, Dimple and Tyagi, Twinkle: A review on security measures of Hadoop. In: 2017 International Conference on Innovations in Control, Communication and Information Systems (ICICCI), pp. 1–4. IEEE, India (2017).
12. Parmar, Raj R and Roy, Sudipta and Bhattacharyya, Debnath and Bandyopadhyay, Samir Kumar and Kim, Tai-Hoon:Large-scale encryption in the Hadoop
13. Roy, Nilabja and Shankaran, Nishanth and Schmidt, Douglas C: Bulls-Eye–a resource provisioning service for enterprise distributed real-time and embedded systems. In:OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"pp.1843–1861,Springer(2006).
14. Dongpo Zhang : Big Data Security and Privacy Protection . In:8th International Conference on Management and Computer Science (ICMCS 2018),Advances in Computer Science Research(2017).
15. Mehak, Gagan: Improving Data Storage Security in Cloud using Hadoop. In:Int. J. Eng. Res. Appl,pp.133–138,(2014).

AUTHORS PROFILE

Shahrbanu Haidari is currently student of M.Tech in computer science and security information brunch in School of Computer Engineering, Kalinga Institute of Industrial Technology, Bhubaneswar , India. Her research interest includes Cryptography, Big Data and privacy in social networks.

Amiya Kumar Dash is currently serves as Assistant Professor in School of Computer Engineering, Kalinga Institute of Industrial Technology, Bhubaneswar , India. He has completed his Masters Degree from Department of Computer Science & Engineering, National Institute of Technology Rourkela, India in 2015. He received the Institute Silver Medal as branch topper in M.tech for the academic year 2013-2015, NIT Rourkela. He is a life member of The Indian Science Congress Association and Institution of Engineering and Technology (IET). His research interest includes natural language processing, sentiment analysis, machine learning.

Jitendra Kumar Rout currently serves as Assistant Professor in School of Computer Engineering, Kalinga Institute of Industrial Technology, Bhubaneswar India. He has completed his Ph.D Degree from Department of Computer Science & Engineering, National Institute of Technology Rourkela, India in 2019. He is a life member of The Indian Science Congress Association and Institution of Engineering and Technology (IET). His research interests include privacy in social networks, cryptography, natural language processing, and multimedia data mining.