International Journal for Rapid Research in Engineering Technology & Applied Science
Vol 5 Issue 10 Oct 2019
ISSN Online: (2455-4723)

Privacy Preserving Mining using Data Encryption scheme for Hadoop Ecosystem

Sonal Jain* Mohit Jain**
M. Tech Scholar, CSE Department* Assistant Professor, CSE Department**
BM College of Technology Indore
imsonaljain@gmail.com* bmctmohitcs@gmail.com**

Abstract: Nowadays, explosive amount of data is being generated every day. Data from sensors, mobile devices, social networking websites, scientific data & enterprises – all are contributing to this huge explosion in data. This sudden bombardment can be grasped by the fact that we have created a vast volume of data in the last two years. Big Data- as these large chunks of data is generally called- Big Data and has become one of the hottest research trends today. Research suggests that tapping the potential of this data can benefit businesses, scientific disciplines and the public sector contributing to their economic gains as well as development in every sphere. Security is one of the important features to keep information safe and secure from unwanted and unintended data. Study of existing work concludes that HDFS does not have any security framework or algorithm to keep data safe and secure. This work proposed a solution to perform encryption of large data going to be put into HDFS as safe and secure.

Keywords:*Hadoop, Map Reduce, Blowfish, Data Encryption*

IIntroduction

Big Data is the aggregation of bulk quantity of data and that data can be in any form, may be in structured form or unstructured form. It is widely popular in several fields due to its storage capacity of relational and non-relational, structured and unstructured data. For big organizations and business development it is an opportunity to enhance business. Data is generated in large amount due to the communication and transmission of data and big data is needed to be processed for data mining algorithms. Big data consist of three V's, called as: Volume, Variety and Velocity.

The need is to develop efficient systems that can exploit this potential to the maximum, keeping in mind the current challenges associated with its analysis, structure, scale, timeliness and privacy. There has been a shift in the architecture of data-processing systems today, from the centralized architecture to the distributed architecture.

The Big Data research orientation, invariably encounter Hadoop. Hadoop is designed to process large amount of data, regardless of its structure. The core of Hadoop is MapReduce framework, created by Google to solve the problem of web search indexes. The nonprofit organization [2] Apache Software Foundation (ACF) maintain and manage Hadoop framework and Hadoop environment technology. The framework such as Mongo DB, NoSql, Pig and many other are introduce in big data environment to manage massive amount of sensitive data at any given time. Several technologies related to Hadoop [3] include the HDFS which is used for distributed file system. The Hive component is developed to maintain data warehouse application with Hadoop server. The MapReduce is a programming model of Hadoop. The Pig is used for querying language in Hadoop, which is similar to SQL language but SQL is for relational database. The Sqoop, provide connectivity to upload data to HDFS and to Hive from MySql. There are several other technologies developed in the Hadoop environment to play with BigData and expert one's own skills.

The MapReduce framework has been widely adopted by various companies and organization to process huge volume of datasets thus, it solve the problem of data that is being too large. The Hadoop is integrated in Linux environment lessens cost-effectively for computing array.

To support distributed file system design, Hadoop Distributed File System (HDFS) was developed. It is java based file system. It is reliable and scalable to data storage. HDFS is designed using low cost hardware and is highly fault tolerant. HDFS replicate the data across the cluster. It continues computation without aborting the process in case of single server failure. HDFS has no restriction on dataset storage. It support both structured and schema-less data.

The challenges of big data are its distributed environment and thus it is more complicated and vulnerable to attack. Without right security and encryption big data means big problem. BigData environment may include dataset with personal identifiable information. Therefore it is important to address the information ownership and classify the data according to its criticality. A block representation of Hadoop ecosystem and core Hadoop is shown in figure 1 respectively.

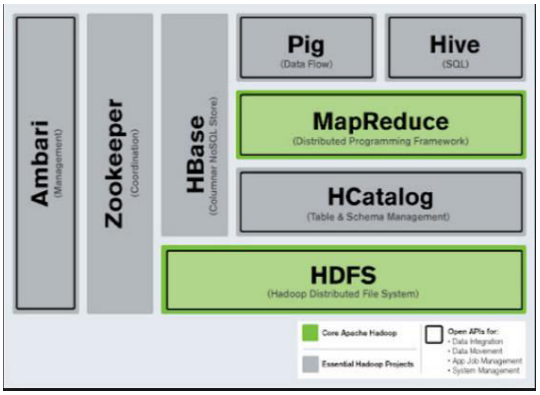


Figure 1: Block Representation of Core Apache Hadoop Server

II EXISTING WORK

Parmar et. al.[1] explore that data generation growth has been raise with rapid and exponential way. Storing and processing of such large data is becoming most hilarious task and need separate advance level algorithm to process. Hadoop ecosystem has been developed to store and process valuable data. They observe security is one of the measure concerns in HDFS because there is no provision to main privacy or safety from information leakage. They provide a security model using Kerberos and AES algorithm to keep data safe and secure. Proposed solution is implemented and evaluated using hadoop 2.7.1 ecosystem based on processing speed.

Vulnerability in Hadoop is a big question in terms of security of confidential information which is stored in Hadoop. Author inquires about the associated issues which are identified in the framework. Also a solution is proposed for the determined limitations. A method is proposed to eliminate the observed vulnerabilities in framework. For every individual, information security is a vital role and necessity. This distribution is not specialized for everyone by vendors. This technique is cost-effective which is used with Hadoop cluster by anyone for security.

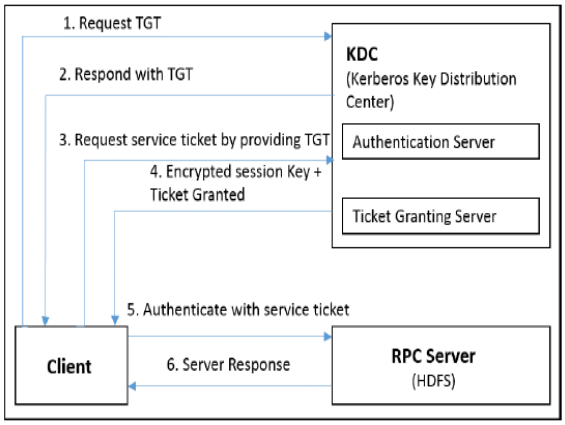


Figure 2: Block Representation of Proposed Solution

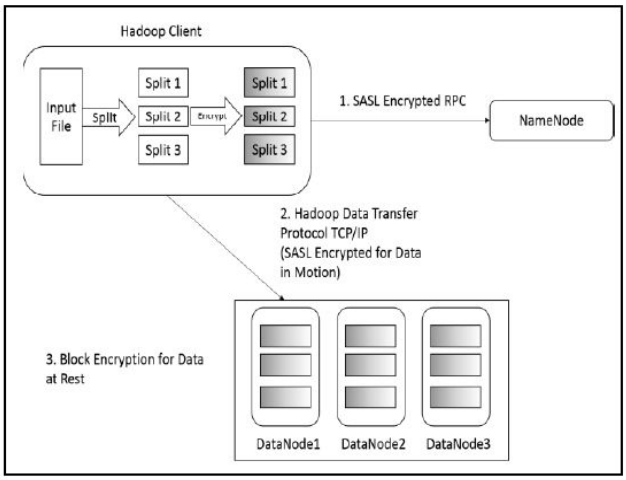


Figure 3: Block Representation-2 of Proposed Solution

Mathur et al. [2] address that different security algorithms are proposed to apply data encryption of plain text information. Here, they implement all encryption algorithms and evaluate computation time for different plain text input. This paper presents the comparison in performance of six most useful algorithms: DES, 3DES, AES, RC2, RC6 and BLOWFISH. Performance of different algorithms is different according to data loads Performance evaluation of proposed solution is shown in Table 1.

Input size in (Kbytes)	AES	3DES	DES	RC6	Blow Fish	RC2
49	56	54	29	41	36	57
59	38	48	33	24	36	60
100	90	81	49	60	37	91
247	112	111	47	77	45	121
321	164	167	82	109	45	168
694	210	226	144	123	46	262
899	258	299	240	162	64	268
963	208	283	250	125	66	295
5345.28	1237	1466	1296	695	122	1570
7310.336	1366	1786	1695	756	107	1915
Average Time	374	452	389	217	60.3	480.7
Throughput (Megabytes/sec)	4.174	3.45	4.01	7.19	25.892	3.247

Table 1: Performance comparison of security algorithms

Study of Table 1 concludes that Blowfish is one of the most effective and efficient algorithm for data encryption. Afterwards, BLOWFISH gives better result

IIIPROBLEM STATEMENT

Hadoop is changing the perception of handling Big Data especially the unstructured data. Let's know how Apache Hadoop software library, which is a framework, plays a vital role in handling Big Data. Apache Hadoop enables surplus data to be streamlined for any distributed processing system across clusters of computers using simple programming models. It truly is made to scale up from single servers to a large number of machines, each and every offering local computation, and storage space. Instead of depending on hardware to provide high-availability, the library itself is built to detect and handle breakdowns at the application layer, so providing an extremely available service along with a cluster of computers, as both versions might be vulnerable to failures.

Big data is massive and messy, and it's coming at you uncontrolled. Data are gathered to be analyzed to discover patterns and correlations that could not be initially apparent, but might be useful in making business decisions in an organization. These data are often personal data, which are useful from a marketing viewpoint to understand the desires and demands of potential customers and in analyzing and predicting their buying tendencies. Security and privacy of this massive data is biggest challenge because they are always residing into sharable storage place known as HDFS. Thus, basic need behind development of BigData analysis tool is to overcome these challenges and retrieve information with best possible solution. Study of previous research work and BigData applications generate issue of insecurity and privacy leakage problem. Subsequently, previous work uses AES algorithm which is proven as low performance than blowfish which can be improved. The complete study observes certain problem in existing work which can be listed as below;

1. Data encryption is one of the primary requirements in hadoop ecosystem to keep data private and sage from unauthorized use.
2. Encryption algorithms always come with issue of extra overhead which should try to reduce as possible as.
3. AES algorithm in Existing work can be replaced by blowfish to improve performance of hadoop ecosystem during data encryption.
4. There is scope of improvement in existing solution where AES can be replaced by other security algorithms.
5. Subsequently, key size can also be raise from 128 bit to 192 bit length for symmetric key cryptography.

IVPROPSOED SOLUTION

Methodology used in proposed work is based on a proper system. This work proposed replacement of AES algorithm with RC6 algorithm and also implement mining approach using association rules to perform mining of ciphered text. Mathur et. al. prove that RC6 perform better and consume low computation and memory overhead. BLOWFISH is a symmetric key block cipher derived for supports key sizes of 128, 192, and 256 bits up to 2040-bits, Proposed solution would perform BLOWFISH encryption first before moving into HDFS and Data Mining algorithm with Mapper class to perform parallel processing of encryption along with mining on ciphered data.

The complete study proposed that Hadoop Server with MapReduce Framework will be configured to process large data set.

1. Initally a dataset is stored on Hadoop Distributed File System (HDFS).
2. Now, data is divided into block on which MD5 algorithm is applied. Then the hash and the value of chunk are added to form another block.
3. At last, Blowfish algorithm is applied on all the blocks. Here Blowfish is applied because it is much faster than any other symmetric key algorithm.
4. Implementation scheme works as follows:
    - The file is chosen and it is placed into hadoop distributed file system.

- The integrity is calculated and data is encrypted by the data encryption code using MD5 and Blowfish algorithms.
- The encrypted data is now distributed to data nodes.
- At the time of decryption, Blowfish algorithm is applied. Again, data is divided into blocks and MD5 is calculated. This new MD5 and the previously calculated MD5 is matched. If it matches it means that integrity of the data is not compromised.
- Along with this, Blowfish and AES algorithms are also compared which clearly shows that Blowfish algorithm take less encryption/decryption time than AES algorithms which means Blowfish algorithm performs better than AES algorithm.

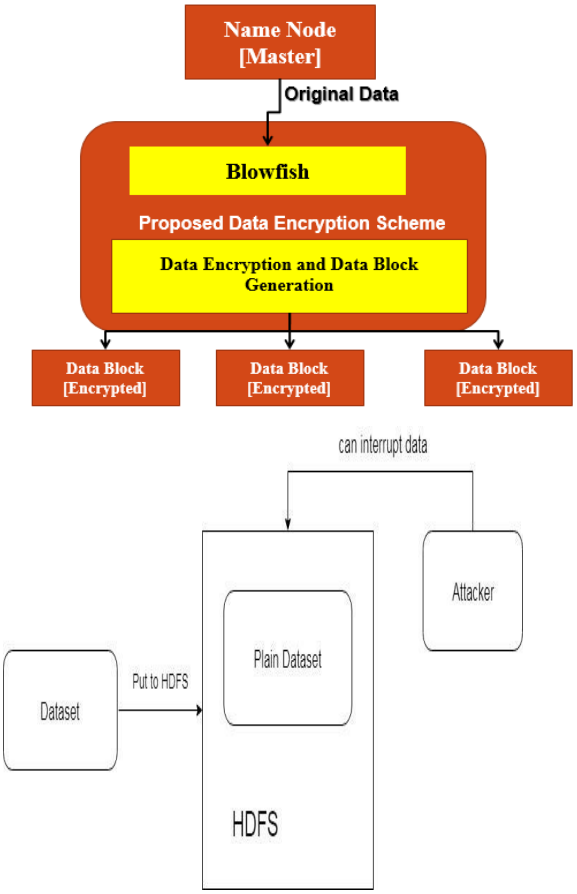


Figure 4: System Architecture

Whenever client uploads data in HDFS, client should definitely use security algorithm, so user fetching that data will get ciphered text.

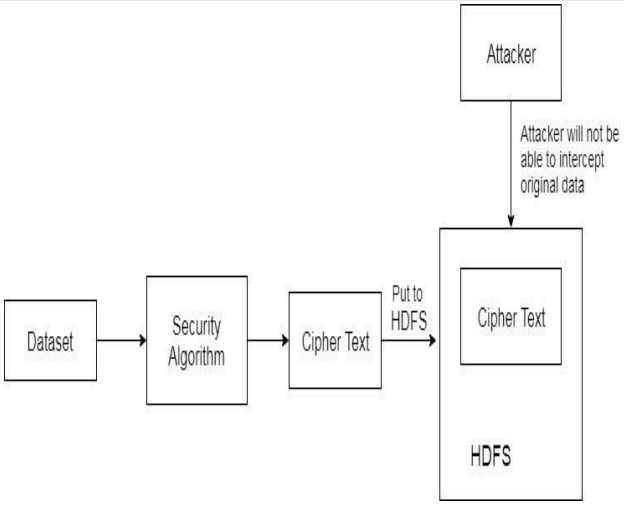


Figure 6: With Security Architecture

Proposed work will replace ARIA and AES with RC6 & Blowfish Algorithms Study of security algorithms address that Blowfish and RC6 can perform better and strong with respect to ARIA and AES.

- BLOWFISH perform better as comparison to AES.
- Key size increase up to 448 bits.
- Improves computation time and memory overheads.

A snipping of implementation is shown in figure 5

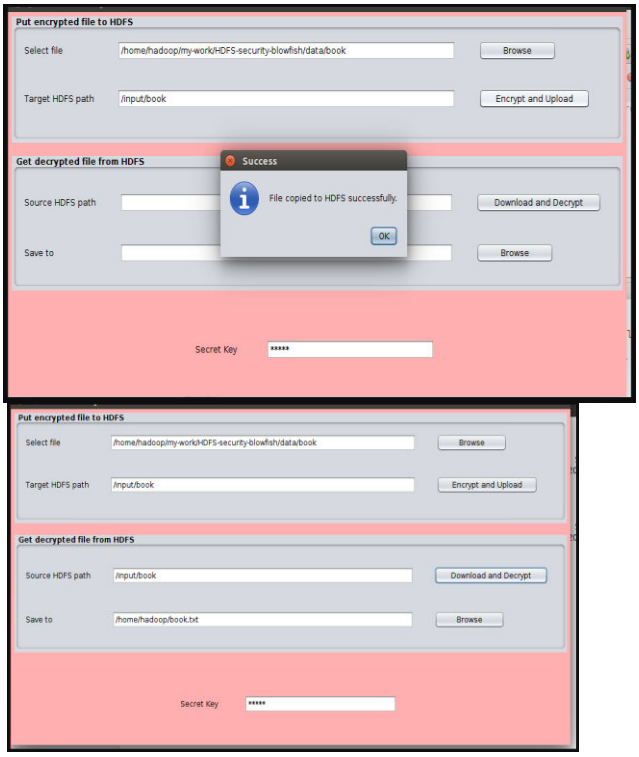
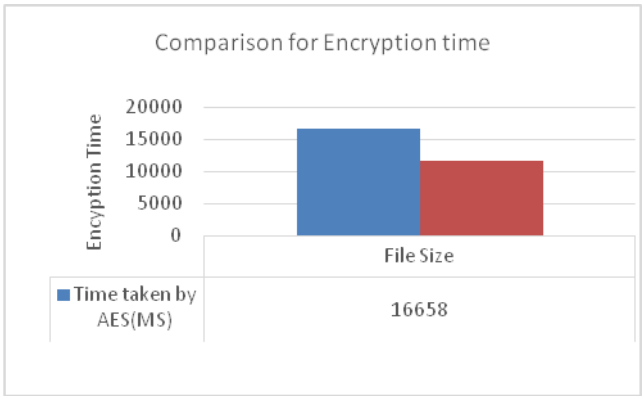


Figure 4.6: Decrypted File

Table 5.1: Comparison for Encryption time

Filesize(MB)	**Time taken by AES(MS)**	**Time taken by Blowfish(MS)**
Total time	16658 MS	11659 MS



Comparison of previous work with existing work is performed, which calculate and compare time to encrypt file using AES and BLOWFISH algorithm. File size is in MB and time is calculated in millisecond.

VICONCLUSION & FUTURE WORK

The complete work concludes that there is strong need to provide data encryption solution over plain text in hadoop ecosystem. Proposed solution would used blowfish algorithm for encryption purpose help to replace use of AES algorithm.

The complete work concludes that proposed solution implements confidentiality, authentication and access control on Hadoop server. It evaluates the user with rights and access permission before mining. This process ensures not only authorization of access but also filter the unwanted and extra mining effort along with privacy layer between actual owner data and user.

Three node Hadoop servers give low computation time then single node. So, proposed solution can be used with Hadoop server to maintain security in HDFS for large data based supermarket application.

References

[1] Raj R. Parmar, Sudipta Roy, Debnath Bhattacharyya, Samir kumar bandyopadhyay, “Large-Scale Encryption in the Hadoop Environment: Challenges and Solutions”. Published on 9 May 2017, pp. 7156-7163, vol. 5, IEEE Access.

[2] Milind mathur, ayushkesarwani, “comparison between DES, 3DES, RC2, RC6, Blowfish and AES”. Proceedings of National Conference on New Horizons in IT - NCNHIT 2013.

[3] S. Ghemawat, H. Gobioff, and S.-T. Leung, ‘‘The Google file system,’’ in Proc. 19th ACM Symp. Oper. Syst. Principles (SOSP), 2003, pp. 29–43.

[4] D. Borthakur, ‘‘The Hadoop distributed file system: Architecture and design,’’ Hadoop Project Website, vol. 11, p. 21, Aug. 2007

[5] B. Lakhe, Practical Hadoop Security. New York, NY, USA: Apress, 2014, pp.19-46.

[6]. P. P. Sharma and C. P. Navdeti, ‘‘Securing big data Hadoop: A review of security issues, threats and solution,’’ Int. J. Comput. Sci. Inf. Technol., vol. 5, no. 2, pp. 2126–2131, 2014.

[7] D. J. Bernstein, ‘‘ChaCha, a variant of Salsa20,’’ in Proc. Workshop Rec SASC, 2008, pp. 1–6.

[8]. Seonyoung Park and Youngseok Lee, Secure Hadoop with Encrypted HDFS, Springer-Verlag Berlin Heidelberg in 2013Prof.

[9] Zerfos, Petros, Hangu Yeo, Brent D. Paulovicks, and Vadim Sheinin. "SDFS: Secure distributed file system for data-at-rest security for Hadoop-as-a-service." In Big Data (Big Data), 2015 IEEE International Conference on, pp. 1262-1271. IEEE, 2015.

[10] Cheng, Zhonghan, Diming Zhang, Hao Huang, and Zhenjiang Qian. "Design and Implementation of Data Encryption in Cloud based on HDFS." International Workshop on Cloud Computing and Information Security (CCIS 2013), pp. 274-277. 2013.

[11] Shehzad, Danish, Zakir Khan, Hasan Dag, and ZekiBozkus. "A Novel Hybrid Encryption Scheme to Ensure Hadoop Based Cloud Data Security."International Journal of Computer Science and Information Security 14, no. 4 (2016): 480.

[12] S. Kaisler et al. “Big Data: Issues and Challenges Moving Forward”. English. In: System Sciences (HICSS), 2013 46th Hawaii International Conference on. IEEE, Jan. 2013, pp. 995–1004.

[13] Fhom, H. S. (2015) ‘Big Data: Opportunities and Privacy Challenges’, arXiv.org, p. 823.

[14] Quan, Q., Tian-Hong, W., Rui, Z. and Ming-jun, X. (2013) ‘A model of cloud data secure storage based on HDFS’. IEEE, pp. 173–178. doi: 10.1109/ICIS.2013.6607836.

[15] Owen O’Malley, Kan Zhang, Sanjay Radia, Ram Marti, and Christopher Harrell. Hadoop Security Design, Technical Report, 2009.10