IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

ADVANCED ALGORITHM : BIG DATA (HADOOP DISTRIBUTION FILE SYSTEM) SECURITY

Manisha[1], Dr. Akhilesh Kumar Bhardwaj[2]

(M.Tech Student)[1], (Assistant Professor)[2]

Department of Computer Science & Engineering,

Shri Krishan Institute of Engineering & Technology, Kurukshetra

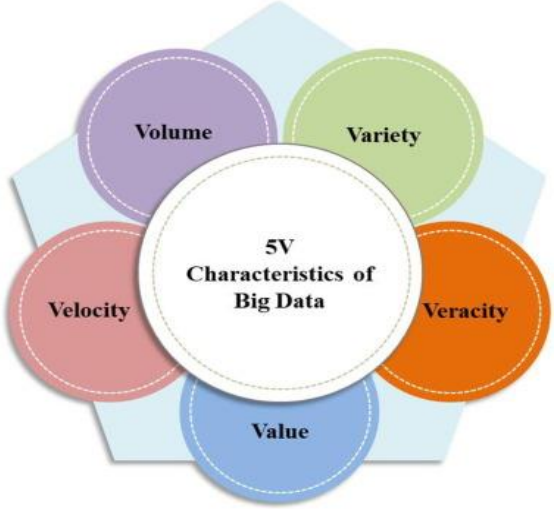
ABSTRACT

The current digital era is transitioning into the information era, where people will need to adapt to the emerging technology that is evolving quickly. Technology's remarkable and rapid development has made life easier for individuals, yet this may have come at the expense of their security and privacy, adding to the burden for those attempting to utilize these technologies. Information security is the research community's focus due to the increasing interdependencies of current computer hardware and the dramatic rise in user numbers.The practicality and viability of obtaining data confidentiality and integrity are shown in the HDFS both empirically and conceptually. The experiments show that the suggested algorithm works well, complies with security guidelines, and ensures data confidentiality and integrity. The research effort also provides a bird's eye view of the future orientations of big data security.

Keywords: Big Data, Hadoop Distribution File System, Advanced Encryption Algorithm

INTRODUCTION

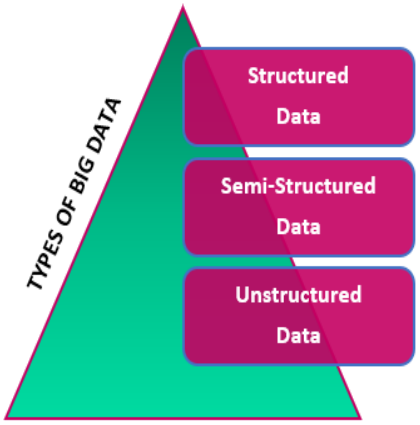
1. Big Data: Characteristics and Types



An assortment of informational indexes or a combination of tremendous information volumes is alluded to as large information. Starting from the start of figuring, the possibility of Huge information has penetrated advanced correspondence and data science. Taking into account that information is produced by each person and all over the place, including cell phones and advanced gadgets, IoT gadgets, online entertainment, servers, and so on. Huge volume informational indexes that are trying for regularly utilized programming devices to accumulate, make due, process, and examine in an adequate measure of time are turning out to be essential for the

steadily extending field of "large information," (1).

The fundamental piece of large information is to gather, make due, store, and control tremendous volumes of information at the appropriate speed and second to secure the legitimate experiences (2). Likewise, Large information makers should create versatile measures of information (Volume) of various configuration types (Assortment) at a controlled recurrence (Speed), while maintaining the critical properties of the crude information (Veracity), which the information gathered can add to the arranged cycle, activity, and extrapolative investigation/speculation (3). As a matter of fact, "Huge information" has no exact definition. In view of a portion of its qualities, it has been characterized. Subsequently, Large information, otherwise called the 5Vs, has been characterized utilizing these five characteristics (3) as displayed.



- **Structured:** "Organised" information is any form that can be found, processed, and managed using a predefined design. In establishing methods to deal with this type of data (where the organisation is predetermined) and deriving value from it, software engineering knowledge has improved more significantly throughout time. Regardless, when the volume of this data increases to massive proportions-typical sizes are now nearing multiple zetta bytes - we are already anticipating challenges (2).
- **Semi-Structured**:Semi-structured data hasn't been put into a database or other specialized repository but contains related information like metadata that makes it easier to analyze than raw data (2).
- **Unstructured:** Unstructured information is defined as any data having a convoluted organization or design. Despite its enormous size, unstructured information creates a number of challenges in processing it and extracting value from it. A diverse information resource with mix of plain text documents, images, audio files, and other types of content serves as a typical example of unstructured information. Nowadays, organizations have access to a wealth of information, but regrettably, they have no idea how to extract value from it because the information is organized in an unorganized or rudimentary manner (2).

2. **Encryption:** Encryption is a computational technique that converts plain text/clear text (decoded, decipherable information) into figure text (scrambled information), which is simply accessible to approved clients with the right cryptographic key (4). Just characterized, encryption changes intelligible arrangement of information into another organization that must be decoded and seen by those with the legitimate secret word (5).Data is changed over into figure text through encryption utilizing a code (an encryption method) and an encryption key (6). Whenever it has been communicated to the beneficiary side, a key (a similar key for symmetric encryption; an alternate incentive for deviated encryption) is utilized to interpret the code information back into the first worth.. Since encryption keys capability comparably to actual keys, just clients who have the fitting key can "discharge " or decode the encoded information (7).
3. **Decryption:** Decryption is the cycle by which information that has been delivered disjointed design by encryption is changed back over completely to its plaintext state. The framework takes the muddled information, switches it over completely to texts and visuals, and afterward unscrambles it to such an extent that both the per user and the framework can promptly decipher it. Both human and robotized decoding techniques are accessible. Moreover, to perform it, a couple of keys or a secret key could be utilized (5). One of the essential contentions for executing an encryption-decoding framework is protection. Unapproved people or gatherings can peruse and get to data that is imparted by means of the Internet (8). Accordingly, information is encoded to lessen information burglary and misfortune. Mail, text archives, pictures, client data, and records are a portion of the items that are regularly encoded. To get to scrambled information, a brief or window mentioning for a secret key is shown to the decoding client (9).

LITERATURE REVIEW

Table 1 Literature survey on various existing algorithm

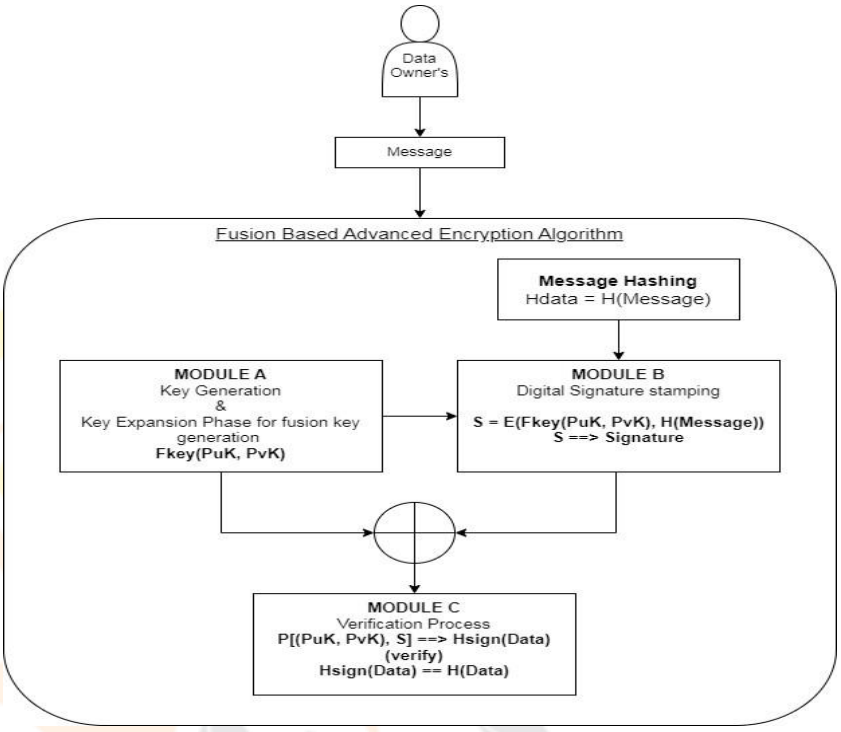
S. No.	Title and year of publications	Observations	Limitations
Hybrid Encryption Approach in Cloud Environment			
1	***Hybrid Encryption Algorithm for Big Data Security in the Hadoop Distributed File System [CAMES 2023]***	A hybrid data encryption algorithm is presented using CP-ABE and AES. Comparative analysis of the proposed algorithm with traditional encryption algorithms such as DES, 3DES, and Blowfish	Selection of attributes must be appropriate else leads to deviations in the performance, which considered as minor limitations in the proposed work
2	***Secure Data Storage in Cloud Using Encryption Algorithm[IEEE 2022]***	The Proposed framework such as S- Box and Feistel algorithm improves the security in the cloud storage framework using different encryption algorithms, which inaddition proposes acryptographic algorithm to store data securely.	The structure of the cloudis addressed based on the transferring, cutting, ordering, encryption, merging, unscrambling and recovery cycle to make sure about the enormous information
3	***An Effective Hybrid Encryption Algorithm for Ensuring Cloud Data Security [IEEE 2019]***	This paper designed a new hybrid algorithm for securing cloud data using three different security policiesfor three different types of sensitive data to maximize control of data owner on storing, processing, and accessing it. The proposed hybrid algorithm is found to be highly secure for all types of sensitive data cloud environments.	Time is a major metric to address and ensure testing the various attack for all processing data and information. Biometric traits can be combined along with the cryptographic processes to strongly connect the security issues in the cloud environment
4	***Secure file storage in cloud computing using a hybrid cryptography algorithm[IJCRT2020]***	LSB steganography technique is used to securely store key information. Files are encrypted withthe help ofthe multithreading technique.	To strongly recommend the integrity and confidentialityof the data storage in thecloud, we can add public key cryptography to avoid attacks. Cloud storage issues of data security are solved using cryptography,steganography techniques.

5	***Enhanced RSA Algorithm with Varying Key Sizes for Data Security in Cloud [IEEE 2020]***	The Key size can be varied to make the encryption process strong. Increasing key size correspondingly increases the time takenfor the encryption anddecryption process. The proposed algorithm reduces the time of encryption and decryption processes and enhances the strength of the algorithm by increasing the key size.	The Efficiency is addressed with the metrics such as speed and time, it can be enhanced still to improve the security of data in the cloud using an addition chaining process
6	***Secure the Cloud Data Transmission using an Improved RSA Algorithm [IEEE 2021]***	The RSA algorithm can be defined as an asymmetric key algorithm that is used to develop a strong security model. In the proposed model the RSA is used to build a new security model because it is a tightly secured algorithm. RSA encryption algorithm with the longest common subsequence of a string (LCS).	Limitations of the algorithm, Using small prime numbers • Using a very close prime Finding the encrypted data is not an easy task for the intruder.
7	***Security analysis and performance evaluation of a new light weight cryptographic algorithm for cloud computing [IEEE 2019]***	The main objective of this study is a comprehensive security analysis and performance evaluation of a lightweight cryptographic algorithm designed to enhance data security incloud computing.	The proposed algorithm is limited to only five rounds of iteration. This decision is based on the understanding that each round necessitates cryptographic mathematical operations that involve 4 bits of data. By reducing the number of rounds, the algorithm aims to minimize computational overheadand conserve energy.

PROPOSED METHODOLOGY/ ALGORITHM

The recommended fusion algorithm includes private and publickeys for encryption and decryption. Two keys - a private key and a public key—are used by Module A to develop a digital signature. Unit Y, on the contrary end, produces a 1024-bit private key using a for loop and 256-bit keys in order to produce a digital signature. Subsequently, the private keys from Unit X and Y are merged and transferred using the XOR algorithm. The XOR output (Unit X, Unit Y) generates secret key, is transformed from bit to byte for digitally signed validation. Thereafter, a message with a hash function is applied. In Encryption process both the message and the secret key processed with hash function .Verification, which takes place in the final stage, by decoding he hash value and contrasting the output with the previous results. As a result, both processing and checking the parts are involved in the proposed method. a thorough method for merging the Unit X, Unit Y, and Unit Z. The proposed fusion algorithm aims to merge Unit X, Unit Y, and Unit Z using private and public keys for encryption and decryption, as well as digital signatures

1. **A Fusion Key Generation:** Data storage and security challenges are increasing with the amount of data generated. The popular Hadoop framework is currently used to analyze and store large amounts of data. Because security was not a top priority when Hadoop was designed, data stored in the Hadoop Distributed File System (HDFS) is vulnerable to attacks. To protect these data, numerous encryption methods have been developed. This paper compares and contrasts the Rivest-Shamir-Adleman (RSA) asymmetric cypher with the Advanced Encryption Standard (AES) symmetric cypher. simply encrypt data in cypher text based on the greatest common divisor. Identification must be studied by a large number of researchers in order to improve the effectiveness of simultaneous interaction between multiple users. The Gen key algorithm approach can be applied to tracking public key revocation methods, network authentication, and distant data sensing sequentially. This fusion based approach's primary objective is to loosen the limits on how easily it may be employed across a broad range of distributionin order to increase consistency. The main benefits of fusion-based algorithms are their scalability, security. The scalability of the proposed approach allows for the addition of any variety of new features to the system. Security is essential for both encryption and hybrid cryptography. Symmetric-key cryptography is used for data encryption, and asymmetric-key cryptography is used for attribute encryption. Efficiency utilizes both asymmetric and symmetric key encryption, making it especially secure.



2. **Advanced Encryption Algorithm Based on Fusion Key to Improve HDFS Big Data Security:** The recommended fusion algorithm includes private and publickeys for encryption and decryption. Two keys - a private key and a public key are used by Module A to develop a digital signature. Unit Y, on the contrary end, produces a 1024-bit private key using a for loop and 256-bit keys in order to produce a digital signature. Subsequently, the private keys from Unit X and Y are merged and transferred using the XOR algorithm. The XOR output (Unit X, Unit Y) generates secret key, is transformed from bit to byte for digitally signed validation. Thereafter, a message with a hash function is applied. In Encryption process both the message and the secret key processed with hash function Verification, which takes place in the final stage, by decoding the hash value and contrasting the output with the previous results. As a result, both processing and checking the parts are involved in the proposed method. athorough method for merging the Unit X, Unit Y, and Unit Z.

   The proposed fusion algorithm aims to merge Unit X, Unit Y, and Unit Z using private and public keys for encryption and decryption, as well as digital signatures for authentication. Here is a detailed explanation of the merging process:

2.1 Private and Public Key Generation:

- Unit X generates a private key and a corresponding public key for encryption and digital signature generation.
- Unit Y also generates a 1024-bit private key using a for loop and 256-bit keys for digital signature generation.

2.2 Digital Signature Generation:

- Unit X utilizes its private key to develop a digital signaturefor a message or data.
- Unit Y uses its private key to produce a digital signature using a similar process.

2.3 Merging Private Keys:

The private keys generated by Unit X and Unit Y are merged using the XOR algorithm.: XORing the bits of the private keys produces an XOR output,which serves as the merged secret key.

2.4 Transformation to Byte Format:

- The merged secret key, generated from the XOR output, istransformed from bit format to byte format.
- This conversion ensures compatibility with the subsequent steps of the algorithm.

2.5 Hashing the Message:

- Before encryption, the message or data to be transmitted is recommended to undergo a hash function.
- The hash function generates a fixed-size hash value that represents the message.

2.6 Encryption Process:

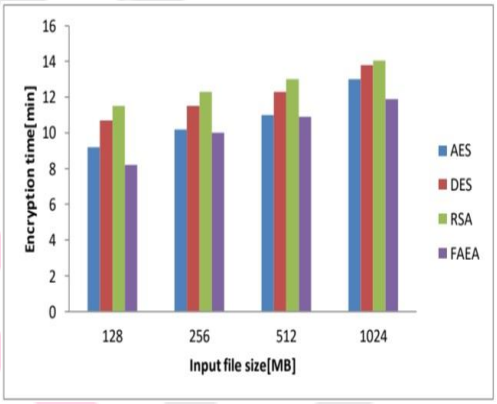
- Both the message and the merged secret key (transformed to byte format) are processed using a hash function forencryption.
- The hash function applies a cryptographic transformation to the message and the secret key, producing encrypted data.

2.7 Verification:

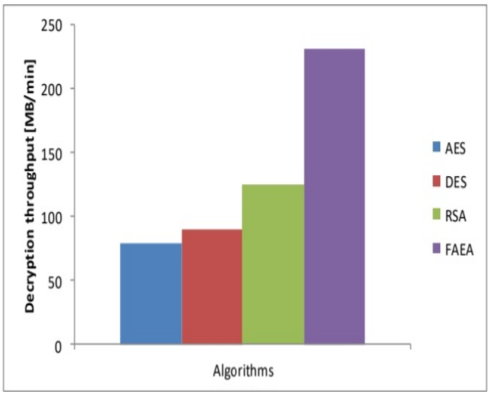
- In the final stage, the verification process takes place.
- The encrypted data is decrypted using the corresponding public key of the recipient (Unit Z).
- The hash value is decoded from the decrypted data.
- The decoded hash value is then compared with the previous hash value generated from the original message.
- If the two hash values match, the verification is successful, indicating the integrity and authenticity of the message.

RESULTS AND ANALYSIS

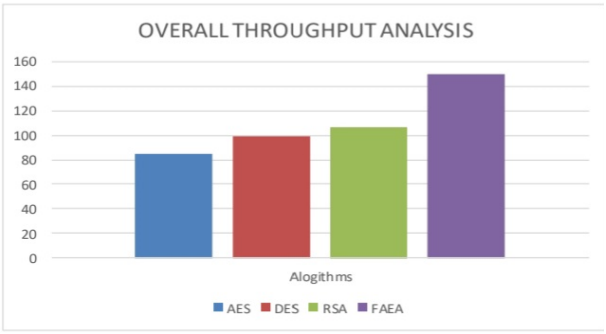
1. **Encryption Time Analysis:** The time of encryption is determined with the observation and calculation of time to generate a from cipher text to plaintext. The simulation metrics represented in the FAEA Algorithm are illustrated in Table 2. In Figure 10 that the time taken for the proposed model is the least for different categories of files. The metric "time" is used to measure the encryption process. There will be increases in the file size slowly from Min(1KB) to Max(1 GB) for the encryption time. From the observation of the performance Analysis, the average time of the Fusion-based enhanced encryption algorithm which ranges the value from Min File Size to Max File Size isrepresented as 8.2 min for FAEA, 9.2 min for AES, 10.7 min for DES, 11.5 min for RSA.



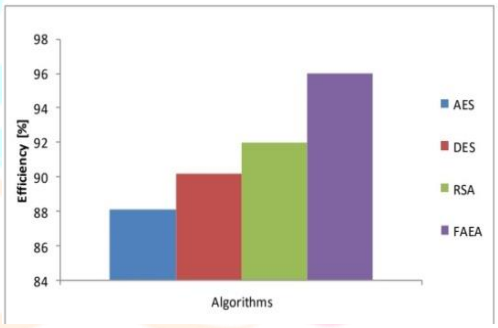
2. **Decryption Time Analysis**: The time of decryption is determined with the observation and calculation of time to generate a from cipher text to plaintext. The simulation metrics represented in the FAEA Algorithm are illustrated in Table 2. In Figure 10, that the time taken for the proposed model is the least for different categories of files. The metric "time" is used to measure the decryptionprocess. There will be increases in the file size slowly from Min(1KB) to Max(1 GB) for the decryption time. From the observation of the performance Analysis, the average time of the proposed algorithm which ranges the value from Min File Size to Max File Size is represented as 5.3 min to



decrypt the 1GB of data, 15.6 min for AES, 14.7 min for DES, 13.8 min for RSA. The average time taken to decrypt the file is computationally high when comparedto the proposed algorithm.



3. **Overall Throughput Analysis:** The overall throughput analysis is measured for the existing and proposed models depicted in Figure and is computed on the performance metrics on the encryption and decryption throughput analysis. Whereas the overall throughput attained by a conventional algorithm such as AES, DES, and RSA is 70 MB/min, 85 MB/min, 99 MB/min, and 107 MB/min which is less than the proposed encryption algorithm FAEA as 150 MB/min



4. **Efficiency Comparison on FAEA:** Based on simulation metrics including throughput, encryption, anddecryption time, the total effectiveness of the proposed system and the existing methods shown in Figure 15 is observed and calculated. The time required by the suggested method to encrypt 1 GB of files is taken into account for all traditional techniques. The suggested algorithm's system efficiency is calculated mathematically by taking into account time consumption, intrusion efficiency, and process throughput.

5. **Performance Evaluation:** The following Table 3 lists the simulation parameters needed to compare the effectiveness of various classical encryption algorithms with the proposed algorithm, the FAEA (Fusion Based Encryption Algorithm). Different simulation measures, such as encryption time (min), decryption time (min), encryption throughput (MB/min), and decryption throughput (MB/min), are used to assess the effectiveness of the proposed technique.The maximum efficiency% of the FAEA is 96.5% as depicted in Figure 16 which is believed to be more efficient than conventional algorithms. It also produces minimal encryption time, decryption time, maximum encryption throughput, and decryption throughput as an average.

Table 2 Efficiency Comparative Analysis of AES, DES, RSA, and the Proposed Algorithm

Parameters	**AES**	**DES**	**RSA**	**FAEA**
Encryption time (min)	25.69	23.48	7.12	6.54
Decryption time (min)	30.18	27.11	6.51	5.59
Throughput for encryption (MB/min)	75.02	90.04	230.25	250.25
Throughput for Decryption (MB/min)	70.64	85.16	220.22	246.50
Efficiency (%)	88.4	90.5	95.5	96.5

CONCLUSION

The integrity and privacy fixes that are necessary to address the issues caused by the development of Cloud Computing technologies for large data were examined in this research. A thorough analysis of the related current research on the popular Fusion-based Encryption (FAEA) method, its development and its use for maintaining the confidentiality and integrity of shared data in storage applications. It contributions to identifying the difficulties and requirements of data integrity and confidentiality is the first stage. A conceptual framework is developed from the viewpoint of various encryption techniques, various data security techniques are tested on the suggested framework. The time taken to encrypt and decrypt is measured with that observations, and the performance of the proposed algorithm is determined It uses the upgraded Fusion-based architecture that is proposed in this study as a benchmark. In short, the research will pave the path for future efforts to address numerous security concerns in cloud computing systems.

REFERENCES

1. Julio Moreno, Manuel A Serrano & Eduardo Fernandez-Medina 2016,'Main Issues in Big Data', Future internet, vol. 8, Issue 3.

2. Dazhi Chong & Hui Shi 2015, 'Big data analytics: a literature review',Journal of Management Analytics, vol. 2, Issue 3, pp. 1-28.

3. Hiba JasimHadi, Ammar HameedShnain, Sarah Hadishaheed &Azizahbthaji Ahmad 2015, 'Big Data and Five Vs Characteristics', International Journal of Advances in Electronics and Computer Science, ISSN: 2393-2835, vol. 2, Issue-1, pp. 1-8.

4. Yahya, S, Khiabani & Shuangqing, Wei 2011, 'Design and analysis of an ARQ based symmetric key generation algorithm', pp. 1273-1278.

5. Prakash, Vishal, Ajay Vikram Singh & Sunil Kumar Khatri 2019, 'A New Model of Light Weight Hybrid cryptography for Internet of Things'. In 2019 3rd International conference on Electronics, Communication and Aerospace Technology (ICECA), pp. 282-285

6. Nasarul Islam, KV & Mohamed Riyas, KV 2017, 'Analysis of Various Encryption Algorithms in Cloud Computing', International Journal of Computer Science and Mobile Computing, vol. 6, Issue 7, pp.1-8.

7. Satyanarayana, Reddy & Beeram 2011, 'Secure Data Transfer Based On Conventional Encryption Technique Including Random Number Key Generation', vol. 2, no. 3.

8. Yibin Li a, Keke Gai b, Longfei Qiuc, Meikang Qiub1 & Hui Zhao 2017, 'Intelligent cryptography approach for secure distributed big data storage in cloud computing', Information Sciences, vol. 389, pp. 1-13.

9. Dhirendra KR Shukla a, Vijay KR Trivedi B & Munesh C Trivedi c 2021, 'Encryption algorithm in cloud computing', Materials Today: Proceedings, vol. 37, pp. 1-7.