Implementing the Triple-Data Encryption Standard for Secure and Efficient Healthcare Data Storage in Cloud Computing Environments

Wid Akeel Awadh[1*], Mohammed S. Hashim[2], Ali Salah Alasady[3]
[1] Department of Computer Information Systems, University of Basrah, Basrah, Iraq
[2] Department of Computer science, Education College for Pure Sciences, University of Basrah, Basrah, 61004, Iraq
[3]Department of Computer Science, University of Basrah, Basrah, Iraq
E-mail: wid.jawad@uobasrah.edu.iq[1], moh.salah@uobasrah.edu.iq[2], Ali_s.hashim@uobasrah.edu.iq[3]
[*]Corresponding author

Keywords: healthcare data, cloud computing, triple-data encryption standard, data encryption standard, security

Received: Januar 6, 2024

Recently, big data analysis has been a very active research area with a significant impact on industrial and scientific domains. Thus, the security of big data provides the conditions for securing and monitoring cloud applications that need protect highly sensitive data hosted in cloud platforms. Nevertheless, the big data security issues have become increasingly problematic, leading organizations to restrict the utilization of cloud services. The existing security methods have revealed several issues, including a weakness of data security and Inaccuracy in data analysis, inefficiencies in performance and dependence on a third party. To address these issues, this study proposed a simpler technique as known a triple-data encryption standard (3DES) through providing the long keys sizes in Data Encryption Standard (DES) to protect the privacy of data against the potential attack in cloud computing environments. The experimental results confirm the efficiency of the proposed method in enhancing the secure of big healthcare data storage in cloud computing environment with less computational time compared to the existing method of Intelligent Framework for Healthcare Data Security. In conclusion, the proposed 3DES is recommended as a candidate encryption and decryption method in healthcare applications in cloud computing environments.

Povzetek: Raziskava predlaga uporabo metode trojnega šifriranja podatkov (3DES) za zaščito zasebnosti podatkov v okoljih računalništva v oblaku, posebej uporabno za varnost velikih zdravstvenih podatkov.

1 Introduction

Rapid advancements in information technology have led enterprises to produce vast volumes of big data, which require efficient security, storage and processing [1], [2]. Cloud computing is a dynamic storage computing platform that offers numerous benefits, including cost effectiveness, flexibility, accessibility, reliability and large storage [3]. Many enterprise users outsource the storage and management of their big data into cloud servers. Cloud computing and big data were introduced in the business technology scene in the 1990s [4]. The growth of big data highlights the expansion in data volume, velocity and variety [5]. The development of cloud computing analytics and big data has coincided with information management development starting with data storage and retrieval, advancing to extraction, transformation and loading processes, business intelligence platforms, advanced analytics libraries and, more recently, comprehensive data governance solutions [6].

Cloud computing applications and big data rely on resource management, real-time demand services and service integration requirements [7]. Cloud computing can rapidly search for dynamic data, enabling the rapid transmission of updated data, storage statuses and dynamic information to cloud users, consequently improving the efficiency of economic gains and distribution by obtaining computational results in a reduced timeframe [8].

Cloud computing, a rising technology in data analytics, is utilized for retrieving, storing and disseminating big data across a distributed network. Daily, enterprise users sort their data onto cloud servers. Consequently, enterprise users are becoming increasingly concerned about the security of data stored within the cloud environment [9]. Cloud computing offers different types of services, namely, software as a service (SaaS), infrastructure as a service (IaaS) and platform as a service (PaaS); however, ensuring big data security in the cloud poses a significant and difficult challenge. Data from sectors such as government, medical, and military often comprises sensitive information that necessitates storage in the cloud [10]. However, users frequently harbor concerns about the security assurances offered by service providers [11].

Whilst the cloud offers numerous benefits, security measures for data storage are sometimes lacking. Opting to store big data in a singular cloud is less favored due to issues such as unavailable resources and data pilferage by internal attackers [12]. The existing security methods have

several issues, including a lack of data security and inaccuracies in data analysis, inefficiencies in performance and complete dependence on a third party [13]. Consequently, assurance of data security has emerged as a significant concern in the field of big data, with widespread apprehension on the misuse of important information, especially when collecting information from diverse sources [14]. Addressing security issues is a technical challenge and should be acknowledged prior to fully leveraging big data opportunities [15]. This issue is addressed in this study by utilizing the newly proposed method called the triple-data encryption standard (3DES). The aim is to improve large-scale data security within the realm of cloud computing, with a specific focus on healthcare datasets.

The main contributions of this research are as follows:

- An effective 3DES method is implemented to enhance the security of big healthcare data. The proposed 3DES aims to partition the input data into three groups depending on their significance to balance the length of the key and its strength. Then, an encryption technique, which includes a triple-encryption technique for highly sensitive data, a double-encryption technique for medium-sensitivity data and a single-encryption technique for low-sensitivity data, is applied.
- Symmetric key handling and data partitioning facilitate the improvement of data structuring, thus enhancing network efficiency. Techniques involving the aforementioned three sub keys and key padding further reinforce data structure management.
- The suggested 3DES approach takes 275 ms to complete the encryption process. By contrast, Tabu requires 406 ms, the MA-ABE technique consumes 494 ms and the hybrid encryption method necessitates 435 ms for encryption. Additionally, the 3DES method boasts a packet delivery ratio of 99%, whereas existing methods, namely, Tabu, MA-ABE and hybrid encryption, yield packet delivery ratios of 93%, 95% and 91%, respectively. The evaluation metrics for encryption time and packet delivery ratio demonstrate the effectiveness of the proposed 3DES method.

This study reviews the previous works on big data within cloud environment and presents them in Section 2. In Section 3, a detailed description of the proposed 3DES is provided. In Section 4, the experimental results are explained. The last section concludes the study and outlines future work.

2 Related works

Cloud computing plays a pivotal role in information and knowledge extraction. However, the effective management of big data frequently leads to numerous security concerns, particularly problems associated with conventional encryption methods. Previous encryption studies focused on datasets of modest proportions, a method ill-suited for complex big data, in which issues of performance and scalability are exacerbated. Consequently, exigency emerges when formulating effective policies to govern data access and safety management, which should be tailored to the idiosyncrasies of big data. Innovative data management systems should be integrated to contend with the distinctive challenges posed by this voluminous and intricate data landscape. Furthermore, the unprecedented interconnectivity observed amongst intelligent devices and computational platforms has ushered in the era of big data. However, this phenomenon also entails privacy issues given the inherent nature of digitally capturing an individual's whereabouts, activities and transactions. This section reviews the existing methods related to big data within the context of cloud computing as show in Table 1.

In [16], a cloud-based method utilizing MapReduce and Hadoop was designed for big data processing to enhance multimedia communication. A genetic algorithm optimized key performance parameters. However, its application to large multimedia data distribution in cloud settings was deemed unsuitable due to limitations in parallelization and algorithmic efficiency.

In [17], a cloud framework employing Dempster–Shafer theory analyzed agricultural data to evaluate land suitability for orange cultivation, incorporating parameters like temperature and rainfall. This approach, integrating cloud computing and the inverse distance weighting model, produced accurate land suitability maps. However, areas with low humidity precision showed reduced suitability values.

In [18], a blockchain-based public auditing system was developed for cloud storage big data to enhance security and reduce overhead. This system involves dividing files into blocks, encrypting them, and using tags in a Merkle hash tree for data integrity verification. Despite its effectiveness against attacks, including the 51% attack, limitations in security and service efficiency were observed.

In [19], the Intelligent Framework for Healthcare Data Security (IFHDS) was introduced, employing a column-based method to secure big healthcare data with minimal impact on processing. It selectively encrypts sensitive data, which is segmented and stored across distributed cloud storage. While enhancing data security, it requires increased computational time.

In [20], a secure data storage scheme for blockchain-enabled edge computing was proposed, integrating bilinear pairing and BLS–homomorphic linear authenticator technologies for data integrity verification. This approach supports error detection and dynamic data management with a counting Bloom filter, ensuring strong data security and reduced computational costs. However, the use of encryption algorithms may impact healthcare application performance.

In [21], a secure multi-cloud framework was developed to mitigate insider attacks through processes like data partitioning and encryption, outperforming AES

and 3DES but increasing computational time. [22] introduced a model combining two-level discrete wavelet transform with AES and RSA for securing medical data, embedded in image RGB channels, requiring improved data hiding capabilities. [23] proposed using Tabu search to select encryption algorithms, optimizing multimedia data security and reducing execution time, yet necessitating local memory for storage.

In [24], an attribute-order-preserving-free-SFS algorithm was introduced for enhancing cloud data security, using encryption to manage dominance relationships but increasing computational time for large databases. [25] presented the Improved Chaos Encryption (ICE) technique, combining chaos encryption with the Lorentz 96 model to boost security through increased randomness, becoming complex with extensive databases.

The review literature indicates that most encryption methods drastically increase computational time, as they necessitate the encryption of big data. Techniques such as intelligent framework for healthcare data security (IFHDS), Advanced Encryption Standard (AES), genetic algorithm, Improved Chaos Encryption (ICE) and Tabu search have large key lengths, that complicate the improvement process and increase the risk of being stuck in suboptimal solutions. Furthermore, existing methods do not maintain an appropriate structure of data, leading to increased computational time. Therefore, the newly proposed 3DES method was utilized to address these issues. The main motivation for this study lies in the critical need for developing a solution that can ensure both the efficiency and security of big data storage in cloud environments, with a concurrent focus on minimizing computational time.

Table 1: Summary of the related works on big data encryption methods in cloud environment

Ref.	Method	Dataset Used	Finding	Limitations
[16]	• MapReduce • Hadoop • genetic algorithm	Large-scale multimedia data	Optimized performance parameters for cloud model	Unsuitable for large multimedia data distribution in cloud
[17]	• Dempster–Shafer theory	Agricultural data for orange cultivation	Accurate land suitability maps for orange cultivation	Decrease in suitability values in low humidity precision areas
[18]	• Blockchain	Big data in cloud storage	Improved data security with reduced overhead	Limited security and efficiency in services
[19]	• Intelligent Framework for Healthcare Data Security (IFHDS)	Big healthcare data	Enhanced security of sensitive patient data	Extended computational time required
[20]	• Bilinear pairing • BLS-Homomorphic Linear Authenticator (BLS-HLA) • Counting Bloom Filter (CBF)	Healthcare applications data	Strong data integrity and security with reduced computational costs	Performance impact on healthcare applications
[21]	• Feistel network • AES • S-box	Big data in multi cloud environment	Improved performance over AES and 3DES	Increased computational time due to encryption
[22]	• AES • RSA	Medical diagnosis data embedded in RGB channels of images	Secured medical diagnosis data	Enhanced capability required for optimal pixel adjustment
[23]	• Tabu	Multimedia data	Highest data security with reduced execution time	Necessity for local memory table for data storage
[24]	• Attribute-order-preserving-free-SFS algorithm	Big data in multi cloud environment	Improved security of data in cloud	Increased computational time with larger databases
[25]	• Improved Chaos Encryption (ICE) approach	Extensive databases	Elevated randomness and enhanced overall security	Complexity with extensive databases

3 Methodology

The 3DES method is proposed in this study to improve the security of big healthcare data in a cloud computing environment. The first phase of the proposed 3DES is input selection, in which the big healthcare data are selected as input data. In the second phase, the big healthcare data are processed, and the 3DES method is utilized for data encryption. The 3DES method is a widely adopted open encryption method that offers strong security with key lengths of either 112 or 168 bits. The encrypted data is then stored in cloud environments. Subsequently, the decryption process is applied to retrieve the big data from the cloud computing environment. The decryption is accomplished by again employing the 3DES method. Figure 1 presents a diagram of the proposed DES.

Figure 2 illustrates the proposed 3DES for big data in a cloud environment.

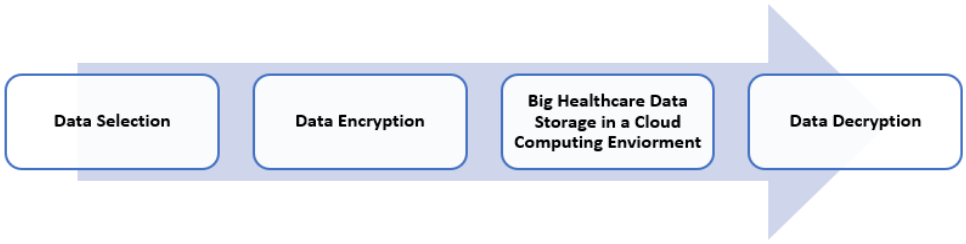

Figure 1: Proposed 3DES method.

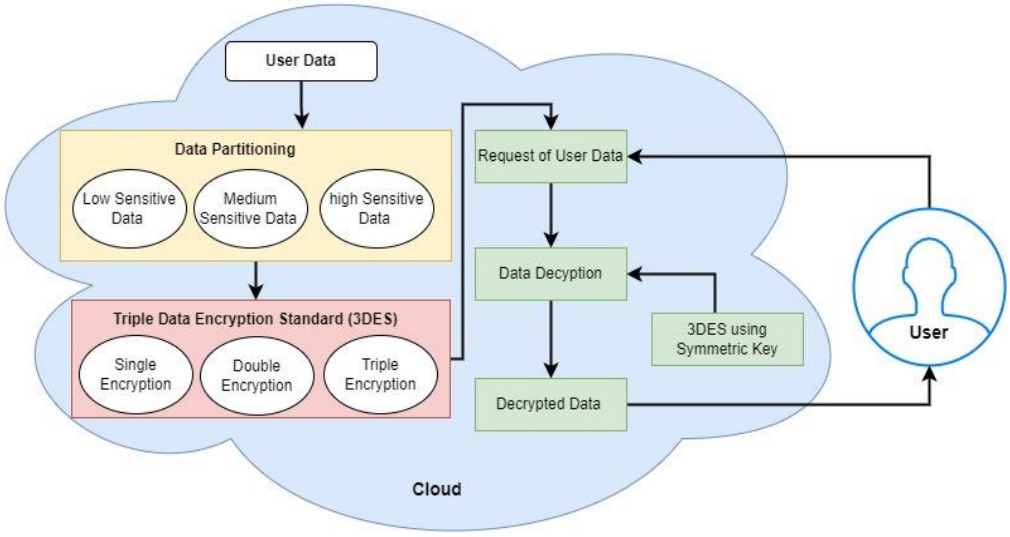

Figure 2: Proposed 3DES method for big data in a cloud environment.

3.1 Data selection

The first phase in the proposed 3DES involves selecting a healthcare dataset as the primary input. In this study, the experimental investigation was based on a real-time healthcare dataset. The healthcare dataset involves various attributes, such as personal information of patients (name, gender, age, address, medical history, job and weight) and their disease information (disease name, symptoms, blood pressure, electrocardiographic, cholesterol and heart rate), comprising a total of 2780 records. From these patient attributes, necessary data were further selected for encryption. Before encrypting the data, an administrator applied data masking techniques to hide attributes of patient information. Table 2 shows a sample dataset involving personal and disease information related to patients.

Table 2: Data sample with the personal and disease information of patients.

ID	Name	Gender	Age	Address	History	Job	Weight	Disease	Symptoms	Blood pressure	Electro_ cardiographic	Cholesterol	Heart rate
ID001	A.M	Male	35	Basrah	Yes	Yes	75	Cardiopathy	Fatigue	80	80	190	140
ID002	M.F	Female	40	Basrah	Yes	Yes	60	Hepatitis	Fever	110	77	179	120
ID003	M.A	Female	20	Basrah	Yes	No	55	Diabetes	Urinate	92	88	181	150
ID004	Y.K	Male	33	Nasria	No	Yes	90	Cardiopathy	Tiredness	88	70	188	138
ID005	N.L	Male	65	Baghdad	No	Yes	63	Diabetes	dry skin	98	85	200	149
ID006	K.T	Male	38	Baghdad	Yes	No	75	Asthma	Wheezing	125	78	195	115
ID007	L.C	Female	45	Baghdad	Yes	yes	65	Hypertension	Headache	140	85	200	130

Table 2 shows the data treated as a collection of characters. These data were collected and interpreted for analysis. Input data representation is described in Equation (1):

$$S_i = S_1, S_2, S_3, \ldots, S_n \quad (1)$$

Where S_i is a number of input data that have been selected, and S_n represents the total number of input data available for selection.

3.2 Data encryption

In the second phase, the 3DES method is used to encrypt the data that had been selected in the first phase. The 3DES method is a symmetric-key block cipher encryption algorithm that applies the data encryption standard (DES) algorithm three times to each data block. Its design focuses on providing a higher level of security compared with the original DES algorithm, which is vulnerable to brute force attacks due to its relatively short key length through increases the effective key length. For encryption, the 3DES method typically employs a 112 or 168 bit key. A single key is used to encrypt the data, which is then decrypted using a second key and encrypted once more using a third key. Equation (2) is typically used to define the proposed 3DES [26].

$$E^1 = E^3 = E, E^2 = D \quad (2)$$

Where E stands for single encryption, E^2 for double encryption, E^3 for triple encryption and D for decryption.

Because the 3DES method has larger key lengths than other encryption techniques, it is beneficial for both encryption and decryption procedures. In addition, the 3DES technique inherits the established encryption tradition of DES and is engineered to interact smoothly with legacy systems and applications that rely on DES encryption [27]. To do this, the single DES algorithm is used three times with three subkeys, and key padding is then added. To guarantee compatibility and flexibility, the keys are extended to 64 bits before being modified for use in 3DES. Depending on how it is used for data encryption, the suggested 3DES can be divided into different categories [27]. The following types are typically recognized in three distinct ways:

- Encrypt-decrypt-encrypt (EDE): This is the most popular way to use 3DES. It entails using the first key to encrypt data, the second key to decrypt it, and the third key to encrypt it once again.
- Encrypt–encrypt–encrypt (EEE): In EEE, data are encrypted three times in succession with three different keys.
- Decrypt–decrypt–decrypt (DDD): DDD involves decrypting data three times in succession with three different keys.

Where the encryption is denoted by $E_K(P)$ and the decryption of the plaintext (P) using the K key is represented by $D_K(P)$.

In 3DES, two basic types of operations are used:

- First, 3DES's encryption process is in responsible for converting plaintext data into ciphertext, which makes the data safe and unreadable by unauthorized parties. The general formula is represented by Equation (3).

$$Ciphertext\ (C) = E(K_3, D(K_2, E(K_1, (P)))) \quad (3)$$

- Secondly, the 3DES decryption method is the opposite of the encryption procedure. By performing this process, authorized individuals can retrieve the original data by converting ciphertext back into plaintext. The general formula is represented by Equation (4).

$$Plaintext\ (P) = D(K_1, E(K_2, D(K_3, (C)))) \quad (4)$$

In 3DES, the set of sub keys is adopted in both processes of encryption and decryption by using two encryption algorithms:

- 2-key 3DES encryption algorithm: The encryption and decryption processes use two separate 56-bit keys. The first key is used to encrypt the data, which are then decrypted using the second key and encrypted once more using the first key.
- 3-key 3DES encryption algorithm: This algorithm, which uses three separate 56-bit keys, is the safest and most advised kind. First, the data are encrypted; second, they are decrypted; and third, they are encrypted once more. Compared to the 2-key 3DES, the 3-key 3DES offers a far better level of protection.

Out of the three subkey utilization methods, the first method is the most resilient since it uses several combinations and has an effective key length of 168 bits. With an effective key length of 112 bits, the second method, which is similar to the first and third subkeys, offers more security than double DES encryption. The third method is weaker than the first and second methods, as the first and second subkeys are excluded from the encryption and decryption process, resulting in an effective key length of 56 bits, which aligns with the key length used in DES methods.

3DES and DES can be used jointly to benefit from each other's strengths and maintain compatibility. For instance:

- Although the encrypted data are calculated using a single DES operation, they are integrated into the 3DES process.
- Although the encrypted data are calculated using a single 3DES operation, they are integrated into the DES process.

3.3 Big healthcare data storage in a cloud computing environment

After encrypting the data, the next phase is securing the storage of large healthcare datasets in the cloud computing environment, where the storage effectively

supports information retrieval and updating [27], [28]. Big data infrastructures typically consider key components such as redundancy, scalability and the choice between direct-attached storage pools, clustered network-attached storage or object-based storage [29], [30], [31]. The integration of big data storage with cloud computing server nodes accelerates the processing and retrieval of much larger datasets. Within this architecture, Spark technology operates within the Hadoop distributed environment to provide memory resources within the cloud, enabling the seamless distribution of encryption keys to worker machines. An enhanced key management system focuses on encrypting data at the column level rather than the row level. The system efficiently manages encryption keys within the Hadoop environment, generating master nodes for various security levels and securely storing these keys for future use during end-user queries. In the 3DES method, client queries are processed by distributing the necessary keys to cloud workers, aligning closely with the queries initiated by workers in cloud computing environments. All data are managed within 'Yet Another Resource Negotiator' (YARN) federations, with each cloud utilizing its own resource manager [32]. This strategy guarantees efficient data processing, similar to the cloud model based on Spark. The YARN data scaling method is used to combine multiple nodes of sub-data YARN, merging them into a cohesive and larger YARN cluster for diverse application needs.

3.4 Data decryption

In the decryption phase, the proposed 3DES is employed to retrieve user data. Data owners can securely retrieve their data from the cloud storage of healthcare organizations. Access to these data is facilitated by the resource manager of the Hadoop cluster [33]. The resource manager is responsible for mapping tables of extensive data across different data nodes or providers, ensuring efficient retrieval. In the 3DES method, data decryption is performed according to security levels, with the appropriate decryption keys used for each attribute. Workers within the YARN framework leverage the Hadoop decryption capabilities provided by the key management system. When a client requests data from cloud computing providers, a resource manager allocates data locations across various data nodes [34]. Subsequently, the nodes of data transmit the data to the proposed 3DES, which assesses the security level of each attribute using a similar approach. Once the 3DES process is completed, Hadoop decrypts the data in the user-configured output directory.

4 Results and discussion

Whilst cloud computing offers numerous advantages, it still poses a significant challenge in regard to ensuring the security of big data storage. Numerous methods have been devised to enhance the security of storing large datasets in cloud environments. Previously employed security-preserving methods exhibited notable shortcomings, including deficiencies in data security and lack of data analysis, suboptimal performance efficiency and dependence on third-party entities, which raises concerns about security vulnerability and potential exposure of individuals' private data. This study introduced the use of the 3DES encryption method to secure large healthcare datasets in cloud environments. The experimental setup involved Python on a system equipped with an i5 processor, 8 GB of RAM, a 6 GB GPU, operating under Windows 10 OS. As discussed in this section, the proposed method utilizes performance metrics as the basis for assessment. The proposed method is also compared with existing methods in this section.

4.1 Performance metrics

The 3DES method for storing big healthcare data in a cloud computing environment is assessed by utilizing certain metrics. This section discusses security measures that can be employed to enhance the cloud computing environment.

- Execution time is known as a period of time taken to complete a specific task of the system. It is defined in Equation (5).

$$Execution\ Time\ = N * R * W \tag{5}$$

Where N is the number of instructions executed, R is the clock rate of instructions per time and W is the waiting time.

- Network usage is a measure of how much of the available network bandwidth or capacity is being used to transmit data. It is usually expressed as a percentage, ranging from 0% (no network activity) to 100% (the network is fully saturated with data). Network usage is calculated using Equation (6).

$$network\ usage = \frac{\text{Netwok bandwith}}{\text{Network traffic}} * 100 \tag{6}$$

- CPU utilization is a metric that quantifies the percentage of time the CPU can actively execute instructions or perform computations with respect to the total available CPU processing time within a specific time interval. CPU utilization is calculated using Equation (7).

$$\text{CPU utilization} = (100\% - \text{Time spent in an idle task}) \tag{7}$$

4.2 Comparative analysis

This section compares the 3DES method with SA-EDS-AES [35] and IFHDS [19] methods, in terms of the performance metrics (Table 3).

Table 3: Comparison table of the proposed 3DES with existing methods.

Method	Execution Time (Min)	Network Usage (GB)	CPU Utilization (%)
SA-EDS-AES	54	83	72
IFHDS	52	81	89
3DES	46	85	94

The comparative performances of the selected methods were evaluated on the basis of 'execution time', 'network usage' and 'CPU utilization'. In the proposed method, the execution time encompasses the duration needed for each process, including the encryption, storage and decryption of big healthcare data. Network usage estimates the encryption, storage and decryption of big healthcare data. Here, CPU utilization is used to ensure better performance and compliance with stringent security requirements. On the basis of the execution time metric, the 3DES method is the fastest, as shown by its lowest execution time of 46 min. SA-EDS-AES [35] has an execution time of 54 min; although slower than 3DES, it is faster than IFHDS [19], which has an execution time of 52 min. In terms of the network usage metric, the proposed method consumes the highest amount of network resources (85 GB), whereas IFHDS [19] uses slightly fewer network resources (81 GB), and SA-EDS-AES [35] consumes 83 GB. In terms of the CPU utilization metric, the proposed 3DES has the highest CPU utilization (94%), indicating that it places a higher demand on the CPU compared with IFHDS (89%) and SA-EDS-AES (72%). The 3DES method offers a more streamlined approach for data security enhancement. In this proposed method, the key sizes in the original DES are augmented to fortify against potential attacks, thereby safeguarding the privacy of data. Figure 3 graphically compares the proposed 3DES with the existing methods.

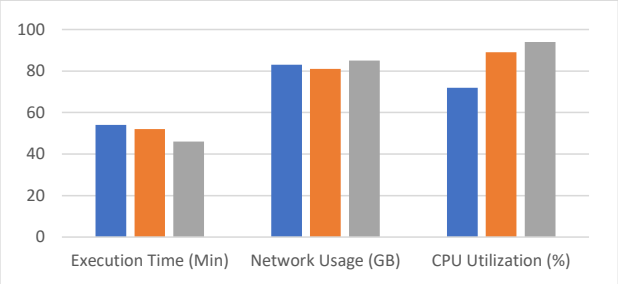

Figure 3: Comparison between the proposed 3DES and the existing methods.

Table 4 compares the proposed 3DES and the existing methods in terms of execution time across various data sizes (varied from 100 to 500). The 3DES consistently outperforms SA-EDS-AES [35] and IFHDS [19] in terms of encryption, storage and decryption for all data sizes. Furthermore, the 3DES method exhibits significantly shorter encryption times, offering faster data protection than the other methods. For instance, at a data size of 500, the 3DES method encrypts in 55 min, whereas SA-EDS-AES [35] and IFHDS [19] take 60 and 63 min, respectively. Figure 4 graphically compares the proposed 3DES and the existing methods in terms of execution time.

Table 4: Comparison between the proposed 3DES and the existing methods in terms of execution time.

Data Size	SA-EDS-AES (min)	IFHDS (min)	3DES (min)
100	25	28	20
200	30	30	25
300	35	35	30
400	43	48	40
500	60	68	55

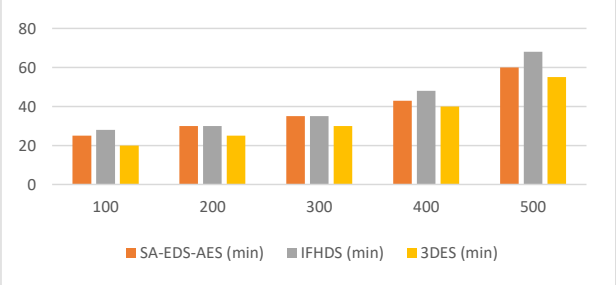

Figure 4: Comparison between the proposed 3DES and the existing methods in terms of execution time.

Table 5 compares the proposed 3DES with the existing methods in terms of network usage across various data sizes (varied from 100 to 500). Regarding network usage, the 3DES method outperforms SA-EDS-AES [35] and IFHDS [19] in terms of network utilization efficiency for all data sizes, requiring less storage space in gigabytes (GB) to secure the same amount of data. For instance, at a data size of 500, the 3DES method utilizes only 0.54 GB for encryption, storage and decryption, whereas SA-EDS-AES and IFHDS consume 0.48 and 0.50 GB, respectively. Figure 5 graphically compares the proposed 3DES and the existing methods in terms of network usage.

Table 5: Comparison between the proposed 3DES and the existing methods in terms of network usage.

Data Size	SA-EDS-AES (GB)	IFHDS (GB)	3DES (GB)
100	0.18	0.19	0.22
200	0.24	0.26	0.28
300	0.32	0.33	0.37
400	0.35	0.38	0.41
500	0.48	0.50	0.54

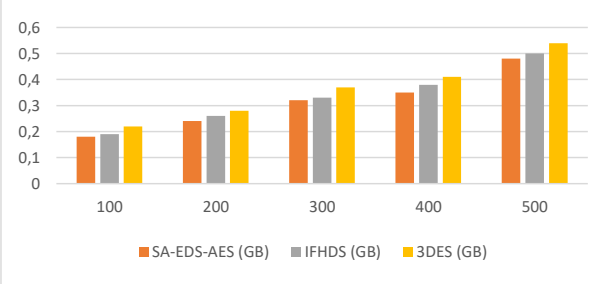

Figure 5: Comparison between the proposed 3DES and the existing method in terms of network usage.

Table 6 compares the proposed 3DES with the existing methods in terms of CPU utilisation across various data sizes (varied from 100 to 500). The proposed 3DES outperforms SA-EDS-AES [35] and IFHDS [19] in terms of CPU utilisation efficiency. For instance, at a data size of 500, the 3DES method utilises only 32% of the CPU, whereas SA-EDS-AES and IFHDS consume 34% and 36%, respectively. Figure 6 graphically compares the proposed 3DES and the existing methods in terms of CPU utilisation.

Table 6: Comparison between the proposed 3DES and the existing methods in terms of CPU utilization.

Data Size	SA-EDS-AES (%)	IFHDS (%)	3DES (%)
100	24	24	23
200	26	26	25
300	26	28	27
400	31	32	29

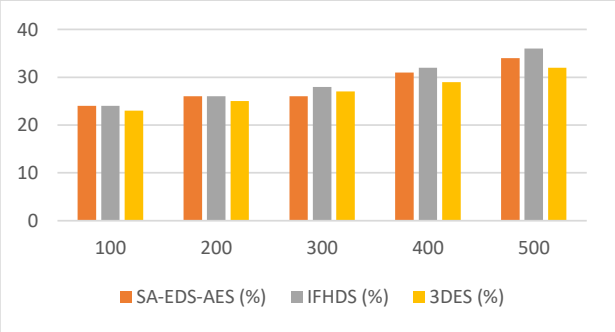

Figure 6: Comparison between the proposed 3DES and the existing methods in terms of CPU utilization.

Then, the proposed 3DES method in a cloud computing environment is compared with existing methods such as Tabu [23], hybrid encryption [21] and MA-ABE [36].

Table 7 shows the comparative performances of the proposed 3DES and the three other existing methods (Tabu [23], hybrid encryption [21] and MA-ABE [36]) in terms of encryption time across varying numbers of blocks (varied from 100 to 500). The 3DES provides a simpler method for data security by balancing the symmetric key length and strength according to the importance of the data. Compared with the existing methods, a significant reduction in encryption time is achieved by the proposed 3DES, as it applies the symmetric key to deal with the data structure, which supports reading and writing. The 3DES method requires the shortest encryption times, demonstrating its superior performance. For instance, at 500 blocks, the 3DES method requires only 275 ms for encryption, whereas Tabu [23], hybrid encryption [21] and MA-ABE [36] consume 406, 494 and 435 ms, respectively. Evidently, the proposed 3DES can enhance the encryption speed, facilitating quick data security whilst maintaining robust protection potential attacks. Figure 7 graphically compares the encryption time between the proposed 3DES and the existing methods.

Table 7: Comparison between the proposed 3DES and the existing methods in terms of encryption time.

Block Number	Tabu (ms)	Hybrid Encryption (ms)	MA-ABE (ms)	3DES (ms)
100	0	0	0	0
200	395	485	401	205
300	399	487	406	259
400	402	489	419	265
500	406	494	435	275

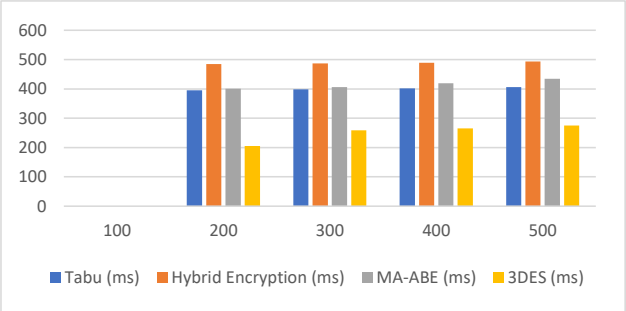

Figure 7: Comparison between the proposed 3DES and the existing methods in terms of encryption time.

Table 8 presents the comparative performances of the proposed 3DES and the existing methods (Tabu [23], hybrid encryption [21] and MA-ABE [36]) in terms of decryption time across varying numbers of blocks (varied from 100 to 500). The 3DES method provides a simpler method for data security by balancing the symmetric key length and strength according to the data's importance, presenting a significant decrease in decryption time, compared with the existing methods in terms of securing big data in cloud environments. The utilization of long key lengths in Tabu [23] results in local optima and increases the time required for model encryption. Meanwhile, hybrid encryption [21] and MA-ABE [36] increase the encryption time due to the need to encrypt data. Amongst them, the 3DES method requires the shortest decryption times. For instance, at 500 blocks, the 3DES method requires only 559 ms for decryption, whereas Tabu, hybrid encryption and MA-ABE consume 657, 783 and 601 ms, respectively. Figure 8 graphically compares the decryption times between the proposed 3DES and the existing methods.

Table 8: Comparison between the proposed 3DES and the existing methods in terms of decryption time.

Block Number	Tabu (ms)	Hybrid Encryption (ms)	MA-ABE (ms)	3DES (ms)
100	0	0	0	0
200	634	779	675	425
300	675	768	677	473

400	602	773	604	410
500	657	783	601	559

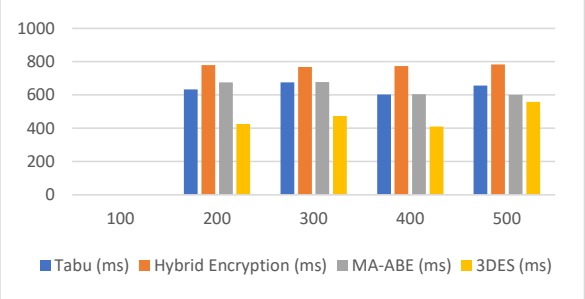

Figure 8: Comparison between the proposed 3DES and the existing methods in terms of decryption time.

Table 9 shows the comparative execution times of the 3DES and existing methods across varying numbers of data blocks. Compared with the existing methods, the 3DES method has a shorter execution time owing to its efficiency in managing data structures and achieving a balanced relationship between key length and strength. The 3DES employs a categorization approach for data based on sensitivity, applying triple encryption to highly sensitive data, double encryption to moderately sensitive data and single encryption to less-sensitive data. The 3DES method aids in decreasing the overall execution duration compared with the existing methods. Tabu [23] requires long execution times and necessitates high key strength, which can be primarily attributed to challenges related to local optima during the search. Meanwhile, hybrid encryption [21] and MA-ABE [36] involve longer execution times due to their processes for data encryption and decryption. Figure 8 graphically compares the execution times between the proposed 3DES and the existing methods.

Table 9: Comparison between the proposed 3DES and the existing methods in terms of execution time.

Block Number	Tabu (ms)	Hybrid Encryption (ms)	MA-ABE (ms)	3DES (ms)
100	0	0	0	0
200	970	1390	1225	830
300	1016	1345	1280	834
400	1020	1341	1111	815
500	979	1272	1119	983

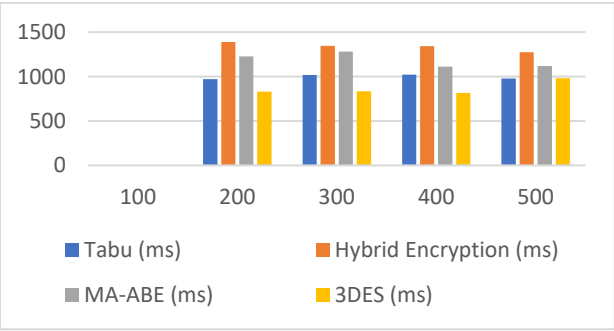

Figure 9: Comparison between the proposed 3DES and the existing methods in terms of execution time.

Table 10 presents the latency time measurements and their comparison for the proposed 3DES and the existing method across different numbers of data blocks. The 3DES has a significantly lower latency time compared with the existing methods, which can be attributed to the proposed method's effective utilization of the data structure when using a symmetric key. By contrast, Tabu [23], MA-ABE [36] and hybrid encryption [21] fail to maintain an appropriate data structure, resulting in increased latency time within the model. For instance, when dealing with 500 data blocks, the proposed 3DES demonstrates a latency of 61 ms, whereas Tabu, MA-ABE and hybrid encryption exhibit 92, 117 and 105 ms of latency time, respectively. Figure 9 graphically compares the latency times between the proposed 3DES and the existing methods.

Table 10: Comparison between the proposed 3DES and the existing methods in terms of latency time.

Block Number	Tabu (ms)	Hybrid Encryption (ms)	MA-ABE (ms)	3DES (ms)
100	0	0	0	0
200	55	82	70	13
300	80	83	77	46
400	88	102	105	59
500	92	105	117	61

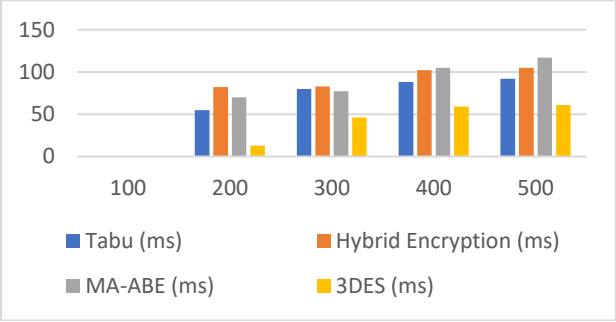

Figure 10: Comparison between the proposed 3DES and the existing methods in terms of latency time.

Table 11 shows the comparative throughputs between 3DES and the existing methods. The 3DES method, mainly because of its efficient handling of structured data when using symmetric key encryption, is superior to the existing methods in terms of throughput. The proposed method adopts a categorization strategy for data based on its sensitivity. As a result, the 3DES method can transfer low-sensitivity data at high speed by employing a single encryption process. By contrast, Tabu [23], MA-ABE [36] and hybrid encryption [21] fail to maintain appropriate data structures, leading to reduced throughput in the cloud environment. The 3DES has a throughput of 105 Mbps, whereas Tabu, MA-ABE and hybrid encryption exhibit throughputs of 82, 47 and 30 Mbps. Figure 11 graphically

compares the throughputs between the proposed 3DES and the existing methods.

Table 11: Comparison between the proposed 3DES and the existing methods in terms of throughput.

Block Number	Tabu (Mbps)	Hybrid Encryption (Mbps)	Mbps	3DES (Mbps)
100	0	0	0	0
200	70	14	40	98
300	72	16	43	99
400	81	23	45	102
500	82	30	47	105

The proposed 3DES and the existing methods are compared in Table 12 in terms of packet delivery ratio for varying block numbers. The 3DES method's efficient data classification and organized processing are responsible for its higher packet delivery rates. In contrast, because Tabu [23], MA-ABE [36], and hybrid encryption [21] don't retain the proper data structures, they show low packet delivery efficiencies. While Tabu, MA-ABE, and hybrid encryption record 93%, 95%, and 91% packet delivery ratios, respectively, 3DES obtains 99%. A graphic comparison of the packet delivery ratios between the proposed 3DES and the existing methods is shown in Figure 12.

Table 12: Comparison between the proposed 3DES and the existing methods in terms of packet delivery ratio.

Block Number	Tabu (MBPS)	Hybrid Encryption (MBPS)	MA-ABE (MBPS)	3DES (MBPS)
100	0	0	0	0
200	88	88	86	91
300	89	88	87	93
400	90	89	88	97
500	93	91	95	99

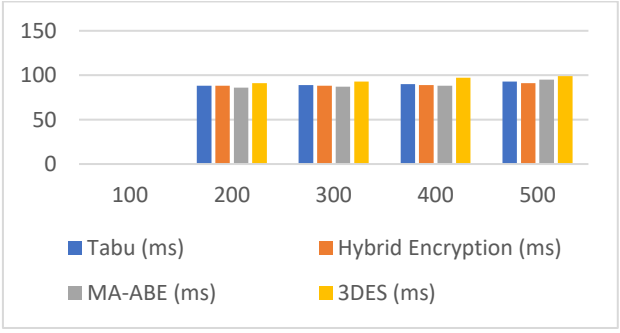

Figure 12: Comparison between the proposed 3DES and the existing methods in terms of packet delivery ratio.

4.3 Enhancing healthcare data security with 3DES

The adoption of the proposed 3DES method in healthcare applications profoundly enhances data security by encrypting patient data during transmission and storage in cloud environments. This prevents unauthorized access and ensures data integrity. Patient confidentiality is safeguarded by implementing strict access controls that restrict access to patient information to only authorized persons. Additionally, healthcare practitioners may make quick, well-informed decisions thanks to 3DES's effectiveness in ensuring low data access latency. 3DES is a great way to manage sensitive healthcare data in cloud environments because it strikes a compromise between strict security measures and operational efficiency, guaranteeing compliance and accessibility.

5 Conclusion and future works

A lot of enterprise users outsource with cloud servers to management and storage their big data. However, ensuring the security of big data continues to be a major concern, which restricts the use of cloud computing services by enterprise users. This study addresses this challenge by using the proposed 3DES for cloud-based big healthcare data storage security.

The proposed 3DES method started with the choosing of input data. In particular, a healthcare dataset containing attributes, such as the personal information of patients (name, gender, age, address, medical history, job and weight) and their disease information (disease name, symptoms, blood pressure, electrocardiographic, cholesterol and heart rate), was selected. Subsequently, the data were encrypted using the 3DES method. Then, the resulting encrypted healthcare data were stored in a cloud computing environment from which efficient read–write data operations could be facilitated. Regarding data retrieval from the cloud environment, the proposed 3DES was utilized for the decryption process. The 3DES method offers a simplified approach by increasing the key sizes used in the DES method to improve protection against potential attacks whilst ensuring the security of data. Experimental results confirmed the effectiveness of the proposed 3DES in enhancing security for large healthcare datasets in the cloud. However, this model demands high network usage and CPU utilization.

In future work, we plan to integrate a modified cryptographic method based on elliptic curves with blockchain technology to address the significant issues, particularly those related to CPU utilization, network usage and computational time in cloud computing environments. Additionally, we intend to expand our research by testing the proposed method on audio and image files. The final goal is to enhance and fortify the security of cloud services, leveraging the effectiveness of multimodal data analysis for accurate treatment and early disease diagnosis.

References

[1] Y. Hajjaji, W. Boulila, I. R. Farah, I. Romdhani, and A. Hussain, “Big data and IoT-based applications in smart environments: A systematic review,” *Comput. Sci. Rev.*, vol. 39, p. 100318, 2021, doi: 10.1016/j.cosrev.2020.100318.

[2] N. Deepa *et al.*, “A survey on blockchain for big

data: Approaches, opportunities, and future directions," *Futur. Gener. Comput. Syst.*, vol. 131, pp. 209–226, 2022, doi: 10.1016/j.future.2022.01.017.

[3] D. ThiBac and N. H. Minh, "Design of network security storage system based on under cloud computing technology," *Comput. Electr. Eng.*, vol. 103, no. August, p. 108334, 2022, doi: 10.1016/j.compeleceng.2022.108334.

[4] W. A. Awadh, A. S. Alasady, and A. K. Hamoud, "Hybrid information security system via combination of compression, cryptography, and image steganography," *Int. J. Electr. Comput. Eng.*, vol. 12, no. 6, pp. 6574–6584, 2022, doi: 10.11591/ijece.v12i6.pp6574-6584.

[5] M. N. Ramachandra, M. Srinivasa Rao, W. C. Lai, B. D. Parameshachari, J. Ananda Babu, and K. L. Hemalatha, "An Efficient and Secure Big Data Storage in Cloud Environment by Using Triple Data Encryption Standard," *Big Data Cogn. Comput.*, vol. 6, no. 4, 2022, doi: 10.3390/bdcc6040101.

[6] B. Berisha, E. Mëziu, and I. Shabani, "Big data analytics in Cloud computing: an overview," *J. Cloud Comput.*, vol. 11, no. 1, 2022, doi: 10.1186/s13677-022-00301-w.

[7] S. Arisdakessian, O. A. Wahab, A. Mourad, H. Otrok, and M. Guizani, "A Survey on IoT Intrusion Detection: Federated Learning, Game Theory, Social Psychology, and Explainable AI as Future Directions," *IEEE Internet Things J.*, vol. 10, no. 5, pp. 4059–4092, 2023, doi: 10.1109/JIOT.2022.3203249.

[8] S. A. Bello *et al.*, "Cloud computing in construction industry: Use cases, benefits and challenges," *Autom. Constr.*, vol. 122, no. xxxx, p. 103441, 2021, doi: 10.1016/j.autcon.2020.103441.

[9] A. Nikiforova, A. Daskevics, and O. Azeroual, "NoSQL security: Can my data-driven decision-making be influenced from outside?," *Big Data Decis. Appl. Uses Public Priv. Sect.*, pp. 59–73, 2023, doi: 10.1108/978-1-80382-551-920231005.

[10] M. S. Hashim and A. Yassin, "Breast Cancer Prediction Using Soft Voting Classifier Based on Machine Learning Models," *Iraqi J. Electr. Electron. Eng.*, vol. 19, no. 2, pp. 42–51, 2023, doi: 10.37917/ijeee.19.2.6.

[11] D. abo aly, W. Atwa, and H. Mousa, "Survey of Computation Integrity Methods For Big Data," *IJCI. Int. J. Comput. Inf.*, vol. 8, no. 2, pp. 77–81, 2021, doi: 10.21608/ijci.2021.207757.

[12] A. Daskevics and A. Nikiforova, "IoTSE-based open database vulnerability inspection in three Baltic countries: ShoBEVODSDT sees you," *2021 8th Int. Conf. Internet Things Syst. Manag. Secur. IOTSMS 2021*, pp. 1–8, 2021, doi: 10.1109/IOTSMS53705.2021.9704952.

[13] M. Ghiasi, T. Niknam, Z. Wang, M. Mehrandezh, M. Dehghani, and N. Ghadimi, "A comprehensive review of cyber-attacks and defense mechanisms for improving security in smart grid energy systems: Past, present and future," *Electr. Power Syst. Res.*, vol. 215, no. PA, p. 108975, 2023, doi: 10.1016/j.epsr.2022.108975.

[14] W. A. Awadh, A. S. Alasady, and M. S. Hashim, "A multilayer model to enhance data security in cloud computing," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 32, no. 2, pp. 1105–1114, 2023, doi: 10.11591/ijeecs.v32.i2.pp1105-1114.

[15] I. Yaqoob *et al.*, "A REVIEW ON INTERNET OF THINGS ARCHITECTURE FOR BIG DATA PROCESSING," *2017 Int. Conf. Energy, Commun. Data Anal. Soft Comput.*, no. December 2019, pp. 3507–3508, 2017.

[16] Z. Zhou and L. Zhao, "Cloud computing model for big data processing and performance optimization of multimedia communication," *Comput. Commun.*, vol. 160, pp. 326–332, 2020, doi: 10.1016/j.comcom.2020.06.015.

[17] M. Mokarram and M. R. Khosravi, "A cloud computing framework for analysis of agricultural big data based on Dempster–Shafer theory," *J. Supercomput.*, vol. 77, no. 3, pp. 2545–2565, 2021, doi: 10.1007/s11227-020-03366-z.

[18] J. Li, J. Wu, G. Jiang, and T. Srikanthan, "Blockchain-based public auditing for big data in cloud storage," *Inf. Process. Manag.*, vol. 57, no. 6, p. 102382, 2020, doi: 10.1016/j.ipm.2020.102382.

[19] Y. M. E. & E. E.-D. H. & A. E.-M. & G. A. & Ayman, "TRANSACTIONAL PROCESSING SYSTEMS IFHDS : Intelligent Framework for Securing Healthcare BigData," *J. Med. Syst.*, vol. 43, no. 124, 2019, [Online]. Available: https://doi.org/10.1007/s10916-019-1250-4

[20] D. Liu, Y. Zhang, D. Jia, Q. Zhang, X. Zhao, and H. Rong, "Toward secure distributed data storage with error locating in blockchain enabled edge computing," *Comput. Stand. Interfaces*, vol. 79, no. July 2021, p. 103560, 2022, doi: 10.1016/j.csi.2021.103560.

[21] G. Viswanath and P. V. Krishna, "Hybrid encryption framework for securing big data storage in multi-cloud environment," *Evol. Intell.*, vol. 14, no. 2, pp. 691–698, 2021, doi: 10.1007/s12065-020-00404-w.

[22] R. Denis and P. Madhubala, *Hybrid data encryption model integrating multi-objective adaptive genetic algorithm for secure medical data communication over cloud-based healthcare systems*, vol. 80, no. 14. Multimedia Tools and Applications, 2021. doi: 10.1007/s11042-021-10723-4.

[23] N. Jayapandian, "Cloud Dynamic Scheduling for Multimedia Data Encryption Using Tabu Search Algorithm," *Wirel. Pers. Commun.*, vol. 120, no. 3, pp. 2427–2447, 2021, doi: 10.1007/s11277-021-08562-5.

[24] A. Cuzzocrea, P. Karras, and A. Vlachou, "Effective and efficient skyline query processing over attribute-order-preserving-free encrypted

data in cloud-enabled databases," *Futur. Gener. Comput. Syst.*, vol. 126, pp. 237–251, 2022, doi: 10.1016/j.future.2021.08.008.

[25] P. Rashmi, M. C. Supriya, and Q. Hua, "Enhanced Lorenz-Chaotic Encryption Method for Partial Medical Image Encryption and Data Hiding in Big Data Healthcare," *Secur. Commun. Networks*, vol. 2022, 2022, doi: 10.1155/2022/9363377.

[26] I. A. Najm *et al.*, "OLAP Mining with Educational Data Mart to Predict Students' Performance," *Inform.*, vol. 46, no. 5, pp. 11–19, 2022, doi: 10.31449/inf.v46i5.3853.

[27] L. Abualigah, A. Diabat, and M. A. Elaziz, "Intelligent workflow scheduling for Big Data applications in IoT cloud computing environments," *Cluster Comput.*, vol. 24, no. 4, pp. 2957–2976, 2021, doi: 10.1007/s10586-021-03291-7.

[28] F. M. Awaysheh *et al.*, "Security by Design for Big Data Frameworks Over Cloud Computing," *Secur. by Des. Big Data Fram. Over Cloud Comput.*, pp. 1–18, 2020, doi: doi.org/10.1109/TEM.2020.3045661.

[29] A. W. Khan *et al.*, "Analyzing and Evaluating Critical Challenges and Practices for Software Vendor Organizations to Secure Big Data on Cloud Computing: An AHP-Based Systematic Approach," *IEEE Access*, vol. 9, pp. 107309–107332, 2021, doi: 10.1109/ACCESS.2021.3100287.

[30] C. Niu and L. Wang, "Big data-driven scheduling optimization algorithm for Cyber–Physical Systems based on a cloud platform," *Comput. Commun.*, vol. 181, no. October 2021, pp. 173–181, 2022, doi: 10.1016/j.comcom.2021.10.020.

[31] S. K. S. Tyagi, A. Mukherjee, Q. Boyang, and D. K. Jain, "Computing Resource Optimization of Big Data in Optical Cloud Radio Access Networked Industrial Internet of Things," *IEEE Trans. Ind. Informatics*, vol. 17, no. 11, pp. 7734–7742, 2021, doi: 10.1109/TII.2021.3055818.

[32] M. Shabbir *et al.*, "Enhancing Security of Health Information Using Modular Encryption Standard in Mobile Cloud Computing," *IEEE Access*, vol. 9, pp. 8820–8834, 2021, doi: 10.1109/ACCESS.2021.3049564.

[33] R. Venkatesan *et al.*, "Secure online payment through facial recognition and proxy detection with the help of TripleDES encryption," *J. Discret. Math. Sci. Cryptogr.*, vol. 24, no. 8, pp. 2195–2205, 2021, doi: 10.1080/09720529.2021.2011096.

[34] C. Kavitha, S. R. Srividhya, W. C. Lai, and V. Mani, "IMapC: Inner MAPping Combiner to Enhance the Performance of MapReduce in Hadoop," *Electron.*, vol. 11, no. 10, pp. 1–16, 2022, doi: 10.3390/electronics11101599.

[35] Y. Li, K. Gai, L. Qiu, M. Qiu, and H. Zhao, "Intelligent cryptography approach for secure distributed big data storage in cloud computing," *Inf. Sci. (Ny).*, vol. 387, pp. 103–115, 2017, doi: 10.1016/j.ins.2016.09.005.

[36] Y. Ming, B. He, and C. Wang, "Efficient Revocable Multi-Authority Attribute-Based Encryption for Cloud Storage," *IEEE Access*, vol. 9, pp. 42593–42603, 2021, doi: 10.1109/ACCESS.2021.3066212.