Journal of Information Systems Engineering and Management
2025, 10(2)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ **Research Article**

Secure Employee Data Hiding in Relational Databases Using the Elliptic Curve Diffie-Hellman Approach.

TS Bharath[1], Channakrishnaraju[2]

[1,2] Department of Computer Science and Engineering, Sri Siddhartha Institute of Technology, SSAHE, Tumkur, India.

* Corresponding author's email id: bharathts@ssit.edu.in

ARTICLE INFO

Received: 10 Oct 2024

Revised: 12 Dec 2024

Accepted: 27 Dec 2024

ABSTRACT

Data masking is greatly efficient in securing data openings and cannot allow hijackers to easily hack the system. An effective approach aims to minimize data privacy breaches and involves developing techniques that leverage encryption for efficient data security. Nevertheless, it is important to hide data in a way that preserves its actual identity for authorized personnel while masking it from cybercriminals to prevent breaches. The process of data hiding is indicated to obtain a clear layout of the dynamic masking process and acquire an efficient solution for the security of a database. This research aims to propose a cryptographic approach Elliptic Curve Diffie Hellman (ECDH) for securely hiding the employee data in relational databases which provide a strong confidentiality level. Initially, the employee data is collected and stored by an organization regarding its employees and a distributed database is performed on different sites that cannot share physical components. The Secure Hash Algorithm (SHA-256) is utilized to hash a encrypted data. After obtaining encrypted data, data hiding is performed to efficiently embed the extended secret data. Then, data parsing is established to convert data from one format to another for structuring data. Finally, the secret data extraction and retrieval are performed on the receiver side.

Keywords: Data Confidentiality, Data Hiding, Elliptic Curve Diffie Hellman, Relational database, Secure Hash Algorithm-256, Security.

1. INTRODUCTION

Privacy and security of personal data become most significant as mobile internet and cloud storage developments continue to advance. The sensitive data of the user are accessed without their permission through the cloud provider or the criminal attackers [1][2]. Before outsourcing, data encryption was the most general approach for ensuring the confidentiality of the data [3]. In recent decades, data hiding has enhanced as a promising solution in enhancing the security by the protection of the digital data [4]. Reversible Data Hiding (RDH) approaches are introduced, enabling the accurate recovery of an actual cover image after the extraction of the hidden secret data [5]. Data ensures that sensitive data remains protected while handling integrity for authorized access. Mostly, a cover medium obtains the alteration at the time of the data-hiding procedure [6]. Various sensitive applications such as healthcare, satellite, defense, etc constrained recovery of a cover medium lack of corruption in the actual properties. Reversible Data Hiding (RDH) is extremely identified in this context [7-9]. RDH is a procedure of hiding secret data within a cover image to recover it to the form of actual quality as well as retrieve embedded data effectively [10].

RDH involves embedding secret data into a cover image, ensuring that both secret data and actual cover data have been effectively recovered at the receiver's end [11][12]. Encryption is an appropriate

solution to handle the confidentiality as well as privacy of the data. An association of encryption, as well as RDH advancements, becomes a significant part of privacy protection. Thus, the RDH in an Encrypted Image (RDH-EI) has motivated the significant focus from the research area [13-15]. Between the different threats faced by different applications, Structured Query Language (SQL) injection stands out as one of the most important and potentially devasting [16]. SQL injection is a type of cyberattack that exploits vulnerabilities in web applications to obtain unauthorized access to the relational database [17]. Once the attackers effectively implement the SQL injection attack, they employ the database to retrieve, modify or delete sensitive data. This impacts the important risks to the integrity, confidentiality as well as applicability of the data and the trust users place in various applications [18-20]. However, sensitive data accessible to authorized employees takes an inherent risk of intended or unplanned disclosure or misuse, resulting in a serious threat to data privacy, security and overall organizational integrity. Moreover, rapidly changing the permission of data access and dynamic user roles face difficulties as sensitive data is inadvertently vulnerable in real-time which maximizes the unauthorized access risk.

The primary highlights of this research are as follows:

- The SQL query processing approach is created which enables the Database Management System (DBMS) to employ the search operations in encrypted data, in a similar manner as in an unencrypted data store.
- The Elliptic Curve Diffie Hellman (ECDH) approach is developed for both encryption as well as decryption processes. The elliptic curve computations in ECDH are faster and require fewer resources than traditional public-key cryptography, making it appropriate for both encryption and decryption.
- The SHA-256 is a cryptographic hash function, meaning it converts employee data into a fixed-length hash value, ensuring that sensitive information, such as passwords or personal details, cannot be reversed or retrieved from the hash.

Paper is arranged as follows: Section 2 literature survey related to data hiding. Section 3 demonstrates a developed methodology. Section 4 illustrates a results and discussion. Section 5 illustrates a conclusion.

2. LITERATURE SURVEY

In this section, the existing works related to data hiding based on the various approaches are analyzed, along with their advantages and limitations.

Vitalii Yesin et al. [21] developed a SQL query processing approach that enables a server of DBMS to establish search functions by encrypted data like an unencrypted database. This was attained via the automatic decryption organization by establishing protected software of appropriate information for searching without a probability of enabling these data individually. This approach contains separate SQL query processing on an application and DBMS server side. The developed technique generates a practical and reliable confidentiality state without the significant complexity of a system, enhancing the memory amount required with adequate query execution time. However, encrypted data limits the capability to establish effective indexes which affects the searching function speed compared to searching on plaintext data.

Guo-Dong Su & Ching-Chun Chang [22] developed an RDHEI using a Huffman-based lossless image coding approach for achieving a large embedding capacity. The original image was compressed into a bitstream via Joint Photographic Expect Groups (JPEG) compression. Then, a variation between an original image and the reconstructed JPEG image was encoded by employing Huffman coding. This approach not only provides an original image's lossless recovery but also a secret message error-free extraction. However, weak spatial correlation in image areas increases the risk of data extraction failure, compromising the integrity and recovery of embedded data.

Peyman Rahmani et al. [23] implemented a secure data outsourcing plan that combined secret sharing-based data hiding for relational databases. This approach embeds one or various secret features about one or various cover attributes in a similar relation. The group of shared columns was established to be connected by cover features and those shared columns and certain virtual shared columns were utilized to retrieve cover and secret attributes. By integrating both secret sharing and data hiding, this approach effectively protects secrecy and data confidentiality. However, the index columns need to be generated in the implemented approach for each secret attribute which requires share columns' searchability.

Mohammed Abdulridha Hussain et al. [24] developed a prevention approach for SQL injection attacks by cryptography as well as searchable encryption. The developed prevention approach utilized the cryptography approach for the encryption of all database data, where, every user data was encrypted through the individual key. A remainder of the database data was ciphered through the secret keys and the searchable encryption was utilized for the datasource processes to safeguard the privacy. A login procedure compared a ciphered username from a database as well as a user entry for the authentication of a user.

Tumkur Shankaregowda Bharath and Channakrishnaraju et al. [25] introduced the snowflake schema-based hap map through Secure Hash Algorithm-256 (SHA-256) for a data masking process. The hashing approach of SHA-256 integrated the data masking through secret sharing to the relational data sources for secure the privacy and confidentiality of a secret employee data. Data masking algorithm effectively preserved a privacy of sensitive and intricate employee data. Data masking was introduced on the chosen database to cover a sensitive information in a group of query results. The introduced approach embedded more than secret attributes over various cover attributes in a similar relational database. However, a index columns need to be generated in the implemented approach for each secret attribute which requires share columns' searchability.

Guo-Dong Su and Ching-Chun Chang [26] implemented the Reversible Data Hiding Encrypted Image (RDHEI) which utilized a Huffman-based lossless image coding approach to obtain the embedding dimensions. The large embedding was designed due to the maximum compression rate as well as the coding efficiency of an integrated plan. The suggested approach aimed to set up a private as well as secure communication channel between sender and receiver through data embedding into encrypted images. The developed method also ensured the error-free extraction of privacy messages. However, the bit extraction defeated risk and image recovery gained on the architectural extent whose spatial correlation was poor.

Sharmistha Jana et al. [27] introduced the effective reversible data hiding approach after the performance of an image interpolation through the exploitation of the centre aligned through Fuzzy Weight Strategy (FWS). Interpolated pixel values were acquired through the consideration of a fuzzy weight of every pair of pixels in the chosen image chunk. This method identified a membership of every pair of pixels and after utilized the fuzzy-rule assisted method for an identification of the benefit of fuzzy principle. Eventually, estimate the interpolated pixel value based on the benefit of the fuzzy principle. However, the estimation of fuzzy weights, membership functions and utilization of the fuzzy-rule-based system imposed a greater computational burden compared to simpler data-hiding approaches.

Chin-Feng Lee and Kuo-Chung Chan et al. [28] developed a new data-hiding approach that performed the Vector Coordinate with Triangular Order Coding (TOC) approach to obtain the RDH in dual images. By the TOC approach, the coordinate vectors were converted into the decimal value for the extraction of secret data. On the other hand, the decimal value was converted into the vector coordinates through the TOC approach for embedding the secret data. Moreover, this approach enabled the setting of the k value based on the number of secret data, where a greater k value enabled the embedding of a larger amount of data, providing enhanced flexibility for the data hider.

Rupali Bhardwaj and Ashish Niranjan et al. [29] introduced the dual image separable data hiding approach for the encrypted Hierarchical Absolute Moment Block Truncation Coding (HAMBTC) approach for the compression of the cover image rather than the traditional one. Modification, as well as modulo operation, was utilized rather than the traditional data hiding approaches for embedding two secret data to enhance its capacity. Then, the Fragile watermark was utilized for the message authentication at the receiver side. However, the image recovery and risk of defeat of bit extraction increases on the area of texture whose spatial correlation was weak.

Xi Ye et al. [30] implemented the usability Thumbnail-Preserving Encryption (TPE) approach, particularly for JPEG images through the utilization of reversible data hiding as well as image encryption in frequency form. In this approach, the actual image was initially transformed into frequency form and chosen few particular locations for the additional data recording in each block. Then, the actual coefficient values at the chosen locations were gathered and embedded into a remaining block before the encryption. Eventually, an additional data was expected to be recorded into chosen positions of each block for non-visual usability.

3. PROPOSED METHODOLOGY

This research determines the significant threat to database security as an inquisitive Database Administrator (DBA) or third parties through complete estimation of the data stored in the Distributed Database Management System (DBMS) server. In the resolution, this research utilizes the searchable encryption plan which enables a user of a secret key for data reading and writing, generating searchable encrypted data and trapdoors. In that, this research extends this plan referred to as Multiwriter or Multireader (M/M) architecture layer through the distribution of the secret key to enable the various users to employ the write and search within an encrypted database. Figure 1 outlines the comprehensive framework of proposed method.

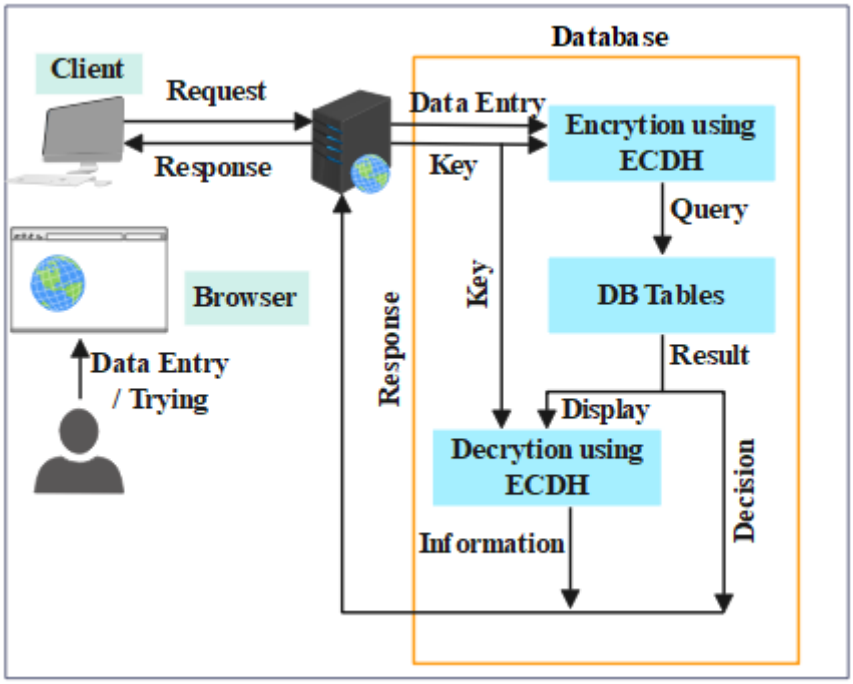

Figure 1. Comprehensive framework of proposed method

3.1 Dataset

This research considers the employee data, which is gathered and stored by an organization regarding its employees. Employee data is information gathered and stored by an organization regarding its employees. It contains personal information like name, contact information, Date of Birth (DoB),

address, social security number, salary, job title, and performance evaluations. The primary goal of employee data is to secure personal information. This research considers the employee database having the individual tables of dept_emp, dept_manager, title, salary, employee, and department. A total of 11,000 employees have been considered for implementation purposes and the details involving employee id, DoB, first name, sur name, gender, hire date. In that, the department table involves the 9 departments with dept_no and dept_name. Dept_manager and dept_employee table have emp_no, dept_no, from_date, to_date. Salary table involves emp_no, amount, from date to date. A distributed database is situated on different sites that cannot share physical components. This is required while a specific database needs to be accessed globally by different users. It requires to be handled such that the users look like one single database. It resolves different problems like fault tolerance, availability, scalability, latency, and throughput that arise from employing a single machine and a single database.

3.2 Encryption using Elliptic Curve Diffie Hellman

ECDH is a key exchange approach enables two parties to firmly create a shared secret across an apprehensive communication network. The computational complexity of addressing a discrete log issue on elliptic curves lies in its difficulty and the corresponding reliance on complex mathematical principles. ECDH is broadly utilized in advanced cryptographic protocols because of its effectiveness and robust security properties. In contrast to private key and public key cryptography, which needed the contribution of specific entities to reveal a secret, these methods are generally slower. The ECC for encryption is represented in equation (1) as follows:

$$y^2 = x^3 + ax + b \tag{1}$$

Where, a and b demonstrates the elements of a finite area involves p elements, which is greater than 3. The acquisition of ordered pairs (x, y) involving the coordinates in an area and such that x and y meets the relation provided through an equation defining a curve, which is a set of points on a curve. Also, the set is created through the set of points on the elliptic curve which has the coordinates in the finite area as well as process is as succeeds: to enhance the curve through two points Q_1 and Q_2 results $(-R_1)$. This task of the group's idea involves different points Q_1, Q_2 and R_1 positioned in straight line and points which add up to zero as an outcome of function meeting a curve is depicted in Figure 2.

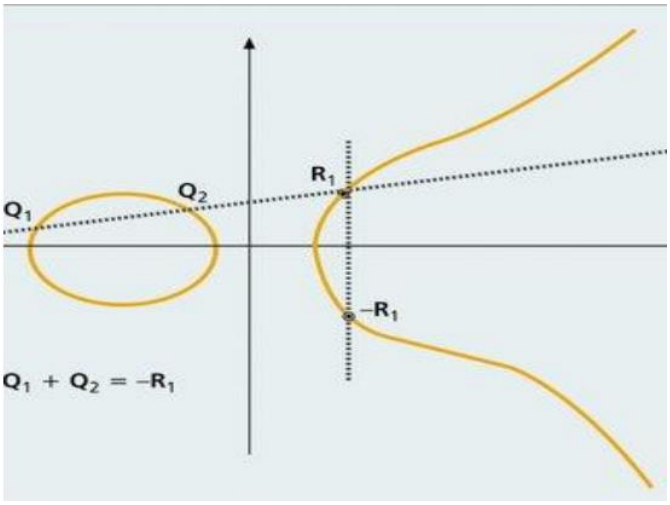

Figure 2. Group law on elliptic curve

Due to popularity of employee-sensitive data being unsecured, ensuring a reliable key exchange becomes a challenging task. The Diffie-Hellman key is a type of elliptic curve that provides a solution to the outlined challenges. When two parties exchange keys, but one of the parties exposes those keys to specific processes after the exchange, and the keys are then encrypted by that party. A strength of Diffie-Hellman key lies in the difficulty of predicting the mathematical operations and the values involved, which form the basis of its security. Hence, it is important to obtain the group operation up

as well as down as efficiently as possible. Some points on the elliptic curve through affinal coordinates, as described earlier, need to be represented. To add two points $Q_1 = (x_1, y_1)$ and $Q_2 = (x_2, y_2)$, where, $x_1 \neq x_2$, it is required to obtain a slope of a line which permits through them, is represented in equation (2) as follows:

$$\lambda = (y_2 - y_1)/(x_2 - x_1) \tag{2}$$

This requires performing division in the finite field. Next, the point where a line meets a curve for the third time is determined, resulting in $(-R_1) = (x_3, y_3)$, where, x_3 is formulated in equation (3) as follows:

$$x_3 = \lambda^2 - x_1 - x_2 \tag{3}$$

Set of three coordinates is utilized in weighted homogenous coordinates (x, y, z), respective to the affine coordinates $\left(\frac{x}{z_2}, \frac{y}{z_3}\right)$ whenever $z \neq 0$. These coordinates involve the benefits of enabling the addition point on an elliptic curve which is performed in 16 area multiplications rather than of all the divisions. The stages of a ECDH approach are provided below:

1. Choose a number (P), that should be significnat and greater than 3.
2. Choose two numbers (a, b). Here $\left((4a^3 + 27b^2) mod\ P \neq 0\right)$.
3. Identify a set of points (G) on an elliptic curve using the formula $y^2 = x^3 + ax + b$ over Z. An addition rule is provided as:
    i. $P + Q = Q + P\ for\ all\ P \in E(Z_p)$
    ii. $if\ P = (x, y) \in E(Z_p), then\ (x, y) + (x_1 - y) = Q$
    iii. Assume $P = (x_1, y_1) \in E(Z_p)\ and\ Q_2 = (x_2, y_2) \in E(Z_p), where\ P \neq -Q$.

Then $P + Q = (x_3, y_3)$, where, x_3, y_3 are formulated in equations (4), (5) and λ is formulated in equations (6) and (7):

$$\mathrm{x_3} = \lambda_2 - \mathrm{x_1} - \mathrm{x_2} \tag{4}$$

$$\mathrm{y_3} = \lambda(\mathrm{x_1} - \mathrm{x_3}) - \mathrm{y_1} \tag{5}$$

$$\lambda = \frac{(\mathrm{y_2} - \mathrm{y_1})}{(\mathrm{x_2} - \mathrm{x_1})} \text{ if } \mathrm{P} \neq \mathrm{Q} \tag{6}$$

$$\lambda = \frac{(3\mathrm{x_1^2} + \mathrm{a_1})}{2\mathrm{y_1}} \text{ if } \mathrm{P} = \mathrm{Q} \tag{7}$$

Then an arbitrary point is selected from the set of points (G) from the set of points:

4. Selection of larger number n.
5. User A key generation:
    i. Choose private n_A through condition $n_A < n$
    ii. Evaluate public p_A, which is formulated in equation (8) as follows:

$$\mathrm{p_A} = \mathrm{n_A} \times \mathrm{G} \tag{8}$$

6. User B key generation:
    i. Choose private n_B through condition $n_B < n$
    ii. Evaluate public p_B, which is formulated in equation (9) as follows:

$$\mathrm{p_B} = \mathrm{n_B} \times \mathrm{G} \tag{9}$$

7. Two sides exchange keys (p_A, p_B).
8. Evaluate secret key K through user A is formulated in equation (10) as follows:

$$\mathrm{k} = \mathrm{n_A} \times \mathrm{p_B} \tag{10}$$

9. Evaluate secret key K through user B is formulated in equation (11) as follows:

$$K = n_B \times p_A \tag{11}$$

10. Convert a packet data to set of points (P_m) and after utilize a subsequent encryption formula is expressed in equation (12):

$$\mathrm{C_m} = \{\mathrm{kG}, \mathrm{P_m} + \mathrm{kp_B}\} \tag{12}$$

11. Decryption for C_m is formulated in equation (13):

$$\mathrm{P_m} + \mathrm{kP_B} - \mathrm{n_B(kG)} = \mathrm{P_m} + \mathrm{k(n_BG)} - \mathrm{n_B(kG)} = \mathrm{P_m} \tag{13}$$

Tag verification ensures that the ciphertext remains complete and unmodified, assuring the data integrity throughout the encryption and decryption process. This supports to prevent the unauthorized modification of the tampering and security.

3.3 Hashing using Secure Hash Algorithm-256

Hashing is the process of transforming string of characters to a fixed-length key, which represents original string in a compact and consistent format. This procedure is used in the database to store as well as preserve the data, enabling rapid and efficient access through the use of digests. [31]. The hash values are preimage-resistant and are not decrypted through any third-party users without a confidential user. In this research, the modified cryptographic algorithm of SHA-256 is used for privacy of confidential employee's salary details which are stored in a database [32]. SHA256 is a family of SHA-2-rnabled hashing where the character string is transformed into 256-bit digest. In that, two modifications have been employed in this research: Initially, modification of compression function in SHA-256 approach. Next, data appending of some textboxes into an individual input text and hashing it to acquire the protected result.

In Figure 3, assuming an initial textbox function is T1 and the values from the textbox are initially transformed to the binary form and after concatenated through power (2,6) value's generated constants, these are the keys to a hashing approach. Thus, by concatenating the values from Textbox 1 and ensuring persistence, the primary symbol value is obtained in equation (14) as:

$$C_1 = concat(T_1, K) \tag{14}$$

Likewise, assuming the nth textbox is formulated in equation (15) as follows:

$$C_n = concat(T_n, K) \tag{15}$$

Then, an input mixture M of hex constants and binary format of a character is considered. It is shown in equation (16) as follows:

$$M = \sum_{i=0}^{n} concat(T_i, K) \tag{16}$$

Here, there are the possibilities of lack of acquiring the 512-bit as a result. Rather than, padding operation's result is employed through providing 0s and 1s to acquire a 512-bit output. In this research, an initial hashing input obtains concatenated through compression function. The result is partitioned into 8-bit blocks, which undergo repetitive padding and compression until a message is separated into 4-bit blocks. Here, a verified Merkle-Damgard (MD) hash function is utilized as the compression function. An MD hashing function is formulated in equation (17) as:

$$\mathrm{MeDa}: \{0,1\} * \rightarrow \{0,1\}\mathrm{n} \tag{17}$$

Hence, obtaining a result of the compression function is formulated in equation (18) as follows:

$$MeDa(M) = f * (padding(M), public\ value) \tag{18}$$

A public value is demonstrated to the character string. Eventually, acquired a 256-bit digest and this approach creates an operation as irreversible. A compression function meets the two significant aims of discouraging a message digest size and preventing crashes in a hash.

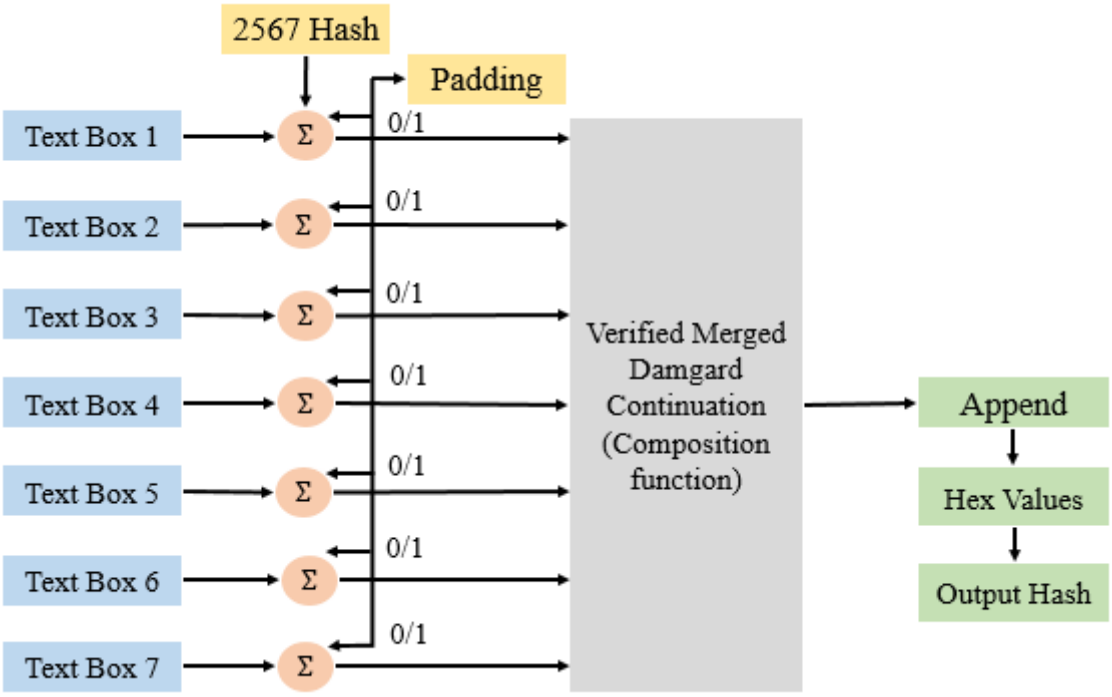

Figure 3. Framework of the proposed encryption module

3.4 Proposed approach for needed data searching

Formulating to implement the search query.

The main operations involve generating and executing search queries to retrieve encrypted data stored in a database on third-party isolated DBMS server. These queries are necessary for managing and operating on the encrypted data.:

A. Operations based on creating security tokens for genuine users.
B. Operations are based on the creation of secret keys, then utilized in a significant manner through the proxy server as well as the database server.
C. Operations are based on formulating for a secure operation of an isolated database.

Then, consider an original approach to generate the search query which is executed on reliable application (database proxy server). The significant phases of a developed approach to generate an SQL search query on encrypted databases are provided in the following.

1. The ith genuine user at an active session receives.
2. Decrypting the file with Steg containers, formulating them for transmission and transmission of Steg containers to DBMS server.
3. Parsing an original SQL query Q.
4. Generating a random number x_k.
5. Creation of trapdoor T_k, by which search will be taken out within an encrypted database.
6. Phases 4 and 5 are continued for all $w_k (w_k \in W)$. Nevertheless, rather than the creation of random number x_k every time, an increase of their primarily produced value is enabled: $x_{k+1} = x_k + 1. k = 1, \ldots |W|$, here, $|W|$ denotes a cardinality of set of keywords w_k.
7. Design of a last query (Q_M) to a database which lack of disclosing the penetrating information.

8. Introducing a few processes to store on the DBMS server (PDS), which identifies interaction variables according to data extracted from transformed Steg containers. These parameters are legal for genuine users alone in an active interaction. These are utilized in respective modules of specific software which helps the characteristics of a developed method.
9. Transforming converted user query to DBMS server for implementation.

Algorithm 1 represents a basic system of search query creation approach.

Algorithm 1: Query Formation Approach (QFA)

Input: K_1^R, $K_2^j Rj$, A_{enc}, M_{enc}, P_i

Output: Modified user query Q_M and its transformed for implementation

1. Read $file_{i\_zip}^{enc} = Enc_{P_i}\left(file_i^1(K_{sec}), file_i^2(K_{ss})\right)$
2. Decrypt file containing the Stego containers $Enc_{P_i}\left(file_{i\_zip}^{enc}\right)$
3. Extract and rename the Stego conatiners; $file_i^1(K_{sec})$, $file_i^2(K_{ss})$
4. Read actual SQL -Q
5. Explaining an actual query Q: extract $w_k \in W$
6. Random number creation: $x_1 = random\_PRNG$
7. $T_1 = Enc_{f\left(K_1^R, K_2^j, x_1\right)}\ Enc_{f\left(K_1^R, K_2^j, x_1\right)}$
8. For $k = 2$ to $|W|$
9. $x_k = \begin{cases} random\_PRNG \\ x_{k-1} + 1 \end{cases}$ $/*\ implementation\ dependent\ */$
10. $T_k = Enc_{f\left(K_1^R, K_2^j, x_k\right)}\ Enc_{f\left(K_1^R, K_2^j, x_k\right)}$
11. End for
12. Design of final query to a database Q_M.
13. Executing the process that describes the session parameters.
14. Transforming a converted user query to DBMS for implementation.

Algorithm 2 demonstrates a Query Execution Approach (QEA).

Algorithm 2: Query Execution Approach

Input: Q_M, $file_i^1(K_{sec})$, $file_i^2(K_{ss})$, K_{pd}

Output: Q

1. Execute a stored process P_{DS}.
   1.1 Decrypting the particular software modules.
   1.2 Compilation as well as execution of decrypted modules of particular software.
   1.3 Extracting a key K_{ss} from Stego container $file_i^2(K_{ss})$.
   1.4 An application context feature ($atrc$) is fixed to the value of K_{ss}.
   1.5 Utilizing a value of $atrc1$ application context feature, then decrypted a further module of special software.
   1.6 Compilation as well as execution of some decrypted modules for the extraction of key K_{sec}.
   1.7 Extracting the key K_{sec} from Stego container $file_i^1(K_{sec})$.
   1.8 An $atrc$ application context feature is fixed to K_{sec}.
2. Implementing search query Q_M within encrypted data.
   2.1 Decrypting the rest of stored modules of distinct software, its compiling and implementation.
   2.2 A pre-decrypted software offers secure, hidden as well as automatic encryption as well as decryption of encrypted data needed for searches.
   2.3 Execution of search query Q_M.

2.4 Eliminating decrypted stored modules of special software.

3. DBMS server receives an outcome of query Q_M from a database proxy server in an encrypted way.
4. A database proxy decrypts obtained encrypted data and receives an application of an actual user query Q.

Illustrations of data search queries (SELECT statement)

Equality (=): e.g., SELECT * FROM employee_plaintext WHERE department = 'HR';

Range (BETWEEN): e.g., SELECT * FROM employee_plaintext WHERE salary BETWEEN 50000 AND 100000;

Set Membership (IN): e.g., SELECT * FROM employee_plaintext WHERE department IN ('HR', 'Finance');

Pattern Matching (LIKE): e.g., SELECT * FROM employee_plaintext WHERE name LIKE 'A%';

4. EXPERIMENTAL RESULTS

An importance of the proposed method is implemented on Python 3.10.11 with the system configurations of intel i5 processor, windows 10 OS and 16GB RAM.

4.1 Performance Analysis

Table 1 demonstrates the average search times for the different conditions. The different search times of both plaintext and encrypted are estimated with different conditions such as equality, range, pattern matching and membership. By the utilization of the proposed ECDH approach, the equality, range, pattern matching and membership conditions attain the minimum plaintext search time of 0.05s, 0.1s, 0.3s and 0.2s as well as minimum encrypted search time of 0.1s, 0.25s, 0.6s and 0.45s respectively.

Table 1. Average search times for different conditions

Condition	Plaintext Search Time (s)	Encrypted Search Time (s)
Equality	0.05	0.1
Range	0.1	0.25
Pattern Matching	0.3	0.6
Membership	0.2	0.45

Table 2 demonstrates the average data decryption times for encrypted data. The different conditions such as equality, range, pattern matching and membership attain the minimum decryption time of 0.3s, 0.4s. 0.7s and 0.6s respectively.

Table 2. Average data decryption times for encrypted data

Condition	Decryption Time (s)
Equality	0.3
Range	0.4
Pattern Matching	0.7
Membership	0.6

Tables 3 to 6 and Figures 4 to 8 demonstrate an outcome of comparative analysis of average search times (in seconds) for both unencrypted (plaintext) and encrypted data across different states. The data is estimated with their entire output for the representation based on the number of acquired values (lines-N) during an execution of the respective SELECT query. Such as queries that please significant kinds of search environments: comparison (equality, inequality), range, set membership

and pattern matching. Two identical tables are utilized for the estimation of a respective time values: One table stores unencrypted data and another one stores similar data in an encrypted way

Table 3. Performance analysis of average values for equality state

Number of Rows (N)	Plaintext Search Time (s)	Encrypted Search Time (s)	Plaintext Data Output Time (s)	Encrypted Data Output Time (s)
10	0.01	0.02	0.05	0.1
100	0.05	0.1	0.2	0.4
1000	0.2	0.5	1	2
10000	1	2	5	10

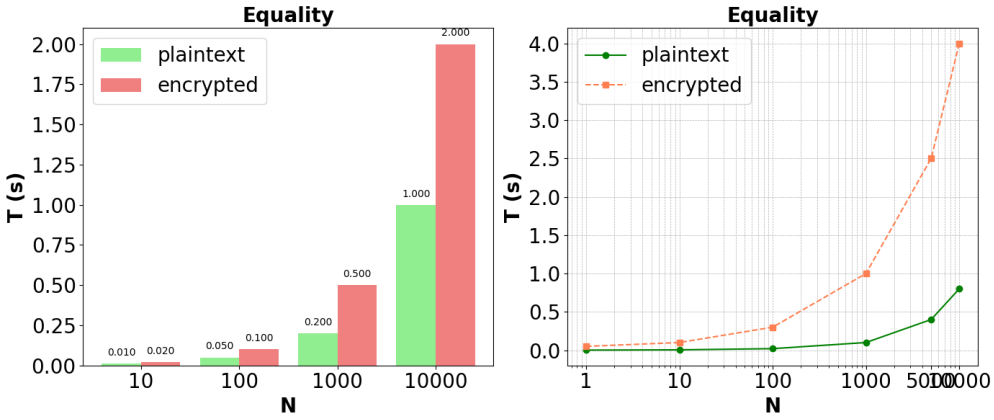

Figure 4. Graphical representation of Average values for Equality state

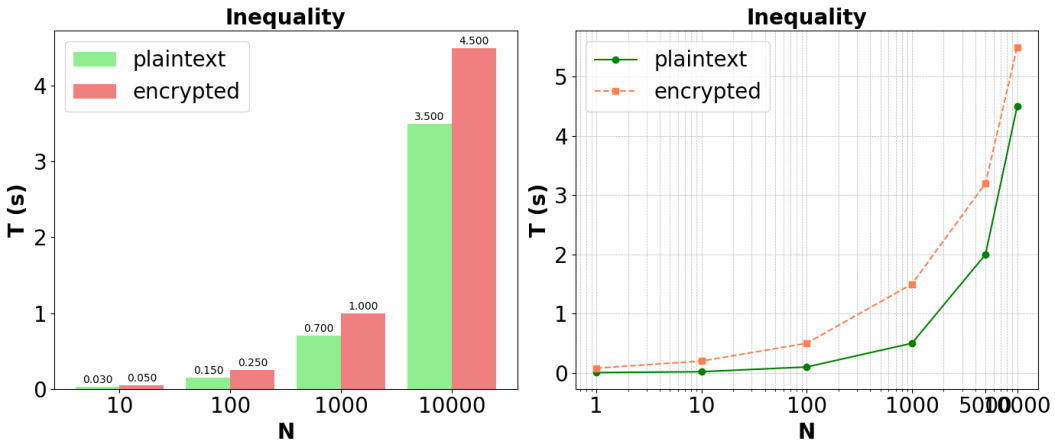

Figure 5. Graphical representation of Average values for Inequality state

Table 4. Performance analysis of average values for Range condition

Number of Rows (N)	Plaintext Search Time (s)	Encrypted Search Time (s)	Plaintext Data Output Time (s)	Encrypted Data Output Time (s)
10	0.05	0.1	0.1	0.15
100	0.1	0.25	0.3	0.6
1000	0.4	0.8	1.5	3
10000	2	4	8	16

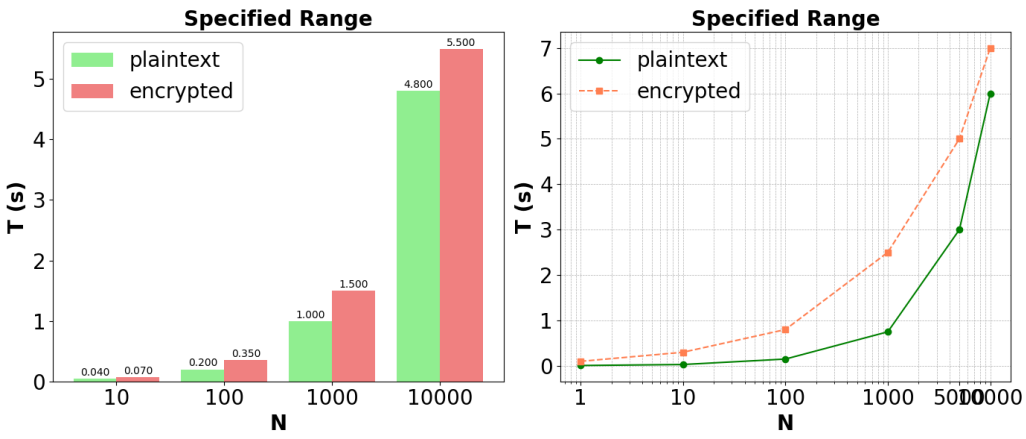

Figure 6. Graphical representation of Average values for Range state

Table 5. Performance analysis of average values for Pattern Matching condition

Number of Rows (N)	Plaintext Search Time (s)	Encrypted Search Time (s)	Plaintext Data Output Time (s)	Encrypted Data Output Time (s)
10	0.1	0.2	0.15	0.25
100	0.3	0.6	0.5	1
1000	1	2	4	8
10000	5	10	20	40

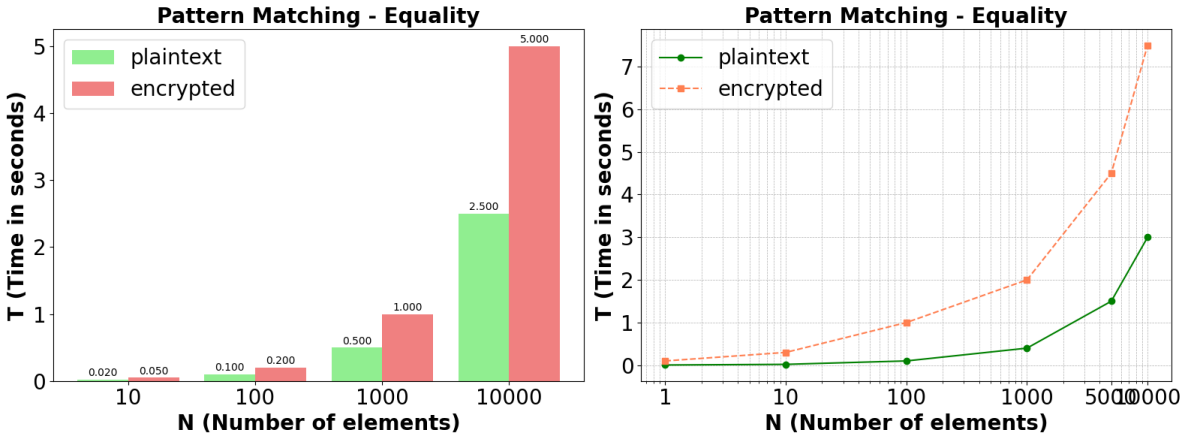

Figure 7. Graphical representation of Average values for Pattern Matching state

Table 6. Performance analysis of average values for Membership condition

Number of Rows (N)	Plaintext Search Time (s)	Encrypted Search Time (s)	Plaintext Data Output Time (s)	Encrypted Data Output Time (s)
10	0.05	0.1	0.1	0.15
100	0.15	0.3	0.25	0.5
1000	0.5	1	2	4
10000	2.5	5	10	20

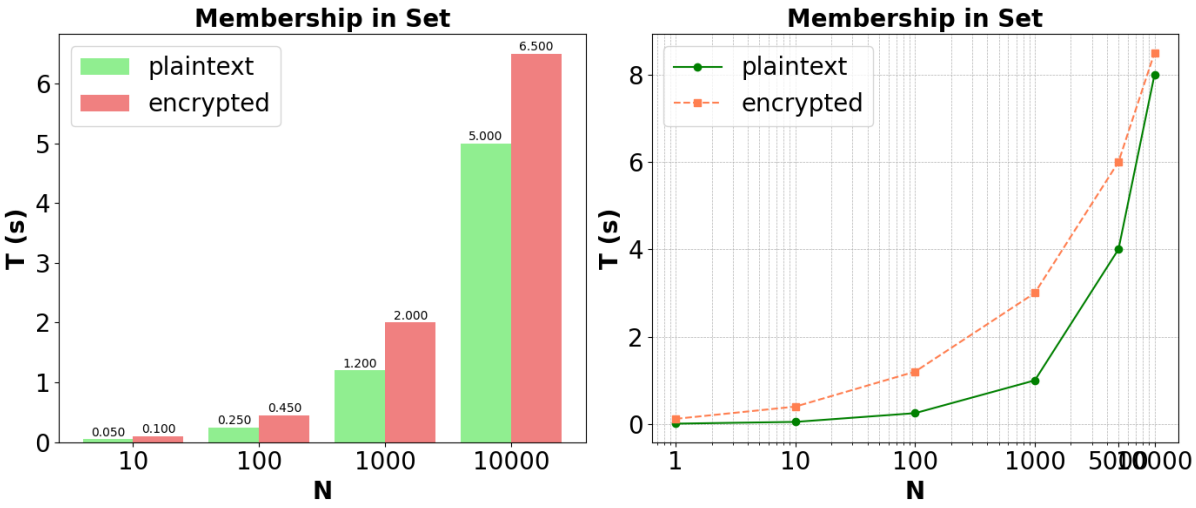

Figure 8. Graphical representation of Average values for Membership state

The analysis results demonstrate a significant overhead in a developed solution. Increase in a respective search and decryption times seems important to a search time for the data in unencrypted databases and a search and decryption time in a few previous searchable encryption databases. This ensures data security through the prevention of anything beyond a access pattern from being leaked, provides query expressiveness with the help of a broad range of search queries and controls the usability of the normal SQL, similar to that of unencrypted databases.

4.2 Comparative Analysis

Table 7 portrays the comparison of developed method with previous methods. Developed method is assessed by different performance metrics like encryption time and decryption time. The existing methods such as SHA-256 [25] are compared with the proposed ECDH approach to validate the significance. The existing approach's values are simulated according to the proposed method values.

Table 7. Comparative Analysis of the proposed method

Method	Encryption time (s)	Decryption time (s)
SHA-256 [25]	0.99	0.84
Proposed ECDH	0.87	0.65

5. CONCLUSION

This research proposes a solution that provides significant level of confidentiality at the time of search, insert, modification and deletion of penetrating information in an isolated database whose data are encrypted. This research proposes an SQL query processing approach that enables DBMS server to process search operations on encrypted data in a same way as it does with unencrypted data. This is obtained by automatic decryption using designed secure software to retrieve a necessary data for the search, without exposing actual data. Furthermore, the ECDH approach is proposed for performing the encryption and decryption of the employee's sensitive data. Then, the SHA-256 approach is utilized for the hashing of the input data. The proposed ECDH method demonstrates significant improvements in confidentiality and privacy based on an evaluation of its effectiveness. The proposed ECDH method attains a better encryption time of 0.87s and decryption time of 0.65s respectively as compared to SHA-256. Future work will involve the hybrid cryptographic algorithm to enhance the overall performance of the model.

REFERENCES

[1] Malik, A., Ashraf, A., Wu, H. and Kuribayashi, M., Reversible Data Hiding in Encrypted Text Using Paillier Cryptosystem. In 2022 Asia-Pacific Signal and Information Processing Association

Annual Summit and Conference (APSIPA ASC) (pp. 1495-1499). IEEE, Chiang Mai, Thailand (2022). https://doi.org/10.23919/APSIPAASC55919.2022.9979998

[2] Vidhya, A. and Kumar, P.M., An Enhanced Reversible Data Hiding Algorithm to Maintain Privacy of Multimedia in Cloud. In 2022 International Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems (ICSES) (pp. 1-7). IEEE, Chennai, India (2022). https://doi.org/10.1109/ICSES55317.2022.9914297

[3] Abduljaleel, I.Q., Abduljabbar, Z.A., Al Sibahee, M.A., Ghrabat, M.J.J., Ma, J. and Nyangaresi, V.O., A lightweight hybrid scheme for hiding text messages in colour images using LSB, Lah transform and Chaotic techniques. Journal of Sensor and Actuator Networks, Vol. 11, no. 4, p.66, (2022). https://doi.org/10.3390/jsan11040066

[4] Huang, Z., Lin, Y. and Chen, X., A block-based adaptive high fidelity reversible data hiding scheme in interpolation domain. Multimedia Tools and Applications, Vol. 83, pp.61715-61736, (2023). https://doi.org/10.1007/s11042-023-14389-y

[5] Upendra Raju, K. and Amutha Prabha, N., Dual images in reversible data hiding with adaptive color space variation using wavelet transforms. International journal of intelligent unmanned systems, Vol. 11, no. 1, pp. 96-108, (2021). https://doi.org/10.1108/IJIUS-08-2021-0095

[6] Panchikkil, S., Manikandan, V.M. and Zhang, Y.D., An efficient spatial transformation-based entropy retained reversible data hiding scheme in encrypted images. Optik, Vol. 261, p.169211, (2022). https://doi.org/10.1016/j.ijleo.2022.169211

[7] Anand, A. and Singh, A.K., A hybrid optimization-based medical data hiding scheme for industrial internet of things security. IEEE Transactions on Industrial Informatics, Vol. 19, no. 1, pp. 1051-1058, (2022). https://doi.org/10.1109/TII.2022.3164732

[8] Bhardwaj, R., Hiding patient information in medical images: an encrypted dual image reversible and secure patient data hiding algorithm for E-healthcare. Multimedia Tools and Applications, Vol. 81, pp.1125-1152, (2022). https://doi.org/10.1007/s11042-021-11445-3

[9] Anand, A., Singh, A.K. and Zhou, H., ViMDH: visible-imperceptible medical data hiding for internet of medical things. IEEE Transactions on Industrial Informatics, Vol. 19, no. 1, pp. 849-856, (2021). https://doi.org/10.1109/TII.2022.3172622

[10] Sadhu, P.K., Yanambaka, V.P. and Abdelgawad, A., Physical unclonable function and machine learning based group authentication and data masking for in-hospital segments. Electronics, Vol. 11, no. 24, p. 4155, (2022). https://doi.org/10.3390/electronics11244155

[11] Bhardwaj, R., An improved reversible data hiding method in encrypted domain for E-healthcare. Multimedia Tools and Applications, Vol. 82, no. 11, pp.16151-16171, (2022). https://doi.org/10.1007/s11042-022-13905-w

[12] Melman, A. and Evsutin, O., Image data hiding schemes based on metaheuristic optimization: a review. Artificial Intelligence Review, Vol. 56, no. 12, pp. 15375-15447 (2023). https://doi.org/10.1007/s10462-023-10537-w

[13] Benseddik, M.L., Zebbiche, K., Azzaz, M.S. and Sadoudi, S., Interpolation-based reversible data hiding in the transform domain for fingerprint images. Multimedia Tools and Applications, Vol. 81, no. 14, pp. 20329-20356 (2023). https://doi.org/10.1007/s11042-022-12288-2

[14] Gowda, B.T.S. and Raju, C.K., A Key-Value pair Schema based Message Digest-5 Hash Algorithm for the Dynamic Data Masking. In 2024 International Conference on Data Science and Network Security (ICDSNS) (pp. 1-7). IEEE. Tiptur, India (2024). https://doi.org/10.1109/ICDSNS62112.2024.10691251

[15] Ming, Y., Zhang, W., Liu, H. and Wang, C., Certificateless public auditing scheme with sensitive information hiding for data sharing in cloud storage. Journal of Systems Architecture, Vol. 143, p. 102965, (2023). https://doi.org/10.1016/j.sysarc.2023.102965

[16] Mohanraj, A., Kumar, N.M., Nikhilesh, S., Sanjay, G.S. and Sanjay, N., Shielding Web Data: Mitigating SQL Injection Threats via Hashing Strategies. In 2024 10th International Conference on Advanced Computing and Communication Systems (ICACCS) (Vol. 1, pp. 2602-2607). IEEE. Coimbatore, India (2024). https://doi.org/10.1109/ICACCS60874.2024.10717176

[17] Mallikarachchi, D., Wong, K. and Lim, J.M.Y., An authentication scheme for FANET packet payload using data hiding. Journal of Information Security and Applications, vol. 77, p.103559 (2023). https://doi.org/10.1016/j.jisa.2023.103559

[18] Fotache, M., Munteanu, A., Strîmbei, C. and Hrubaru, I., Framework for the Assessment of Data Masking Performance Penalties in SQL Database Servers. Case Study: Oracle. IEEE Access, Vol. 11, pp.18520-18541 (2023). https://doi.org/10.1109/ACCESS.2023.3247486

[19] Sundaram, A., Abdel-Khalik, H. and Al Rashdan, A., Deceptive Infusion of Data: A Novel Data Masking Paradigm for High-Valued Systems. Nuclear Science and Engineering, Vol. 196, no. 8, pp. 911-926, (2022). https://doi.org/10.1080/00295639.2022.2043542

[20] Bhardwaj, R., Hiding patient information in medical images: an enhanced dual image separable reversible data hiding algorithm for E-healthcare. Journal of Ambient Intelligence and Humanized Computing, Vol. 14, no. 1, pp.321-337, (2023). https://doi.org/10.1007/s12652-021-03299-2

[21] Yesin, V., Karpinski, M., Yesina, M., Vilihura, V., Kozak, R. and Shevchuk, R., Technique for Searching Data in a Cryptographically Protected SQL Database. Applied Sciences, Vol. 13, no. 20, p.11525, (2023). https://doi.org/10.3390/app132011525

[22] Rafique, A., Van Landuyt, D., Beni, E.H., Lagaisse, B. and Joosen, W., CryptDICE: Distributed data protection system for secure cloud data storage and computation. Information Systems, Vol. 96, p.101671, (2021). https://doi.org/10.1016/j.is.2020.101671

[23] Rahmani, P., Taheri, M. and Fakhrahmad, S.M., A novel secure data outsourcing scheme based on data hiding and secret sharing for relational databases. IET Communications, Vol. 17, no. 7, pp.775-789, (2023). https://doi.org/10.1049/cmu2.12581

[24] Hussain, M.A., Hussien, Z.A., Abduljabbar, Z.A., Ma, J., Al Sibahee, M.A., Hussain, S.A., Nyangaresi, V.O. and Jiao, X., Provably throttling SQLI using an enciphering query and secure matching. Egyptian Informatics Journal, Vol. 23, no. 4, pp.145-162 (2022). https://doi.org/10.1016/j.eij.2022.10.001

[25] Bharath, T.S. and Raju, C.K., An efficient snow flake schema with hash map using SHA-256 based on data masking for securing employee data. Bulletin of Electrical Engineering and Informatics, Vol. 14, no. 1, pp.790-799, 2025, https://doi.org/10.11591/eei.v14i1.8767

[26] Su, G.D. and Chang, C.C., Toward high-capacity crypto-domain reversible data hiding with huffman-based lossless image coding. The Visual Computer, Vol. 39, no. 10, pp.4623-4638, (2022). https://doi.org/10.1007/s00371-022-02613-z.

[27] Jana, S., Jana, B., Lu, T.C. and Vo, T.N., Reversible data hiding scheme exploiting center folding with fuzzy weight strategy. Journal of Information Security and Applications, Vol. 69, p. 103276 (2022). https://doi.org/10.1016/j.jisa.2022.103276

[28] Lee, C.F. and Chan, K.C., A novel dual image reversible data hiding scheme based on vector coordinate with triangular order coding. IEEE Access. Vol. 12, pp. 90784-90814, (2024). https://doi.org/10.1109/ACCESS.2024.3421545

[29] Bhardwaj, R. and Niranjan, A., An improved dual image separable reversible data hiding algorithm for encrypted HAMBTC compressed images. Multimedia tools and applications, Vol. 82, no. 3, pp. 3335-3362 (2023). https://doi.org/10.1007/s11042-022-13209-z

[30] Ye, X., Zhang, Y., Xiao, X., Yi, S. and Lan, R., 2023. Usability enhanced thumbnail-preserving encryption based on data hiding for jpeg images. IEEE Signal Processing Letters, Vol. 30, pp. 793-797, (2023). https://doi.org/10.1109/LSP.2023.3290836

[31] Suman, R.R., Mondal, B. and Mandal, T., A secure encryption scheme using a Composite Logistic Sine Map (CLSM) and SHA-256. Multimedia Tools and Applications, Vol. 81, no. 19, pp. 27089-27110, (2022). https://doi.org/10.1007/s11042-021-11460-4

[32] Mohanty, M.D., Das, A., Mohanty, M.N., Altameem, A., Nayak, S.R., Saudagar, A.K.J. and Poonia, R.C., Design of smart and secured healthcare service using deep learning with modified SHA-256 algorithm. In Healthcare (MDPI), Vol. 10, No. 7, p. 1275, (2022). https://doi.org/10.3390/healthcare10071275