A NEW TECHNIQUE BY USING INVERTED TABLES AND 3D BOX FOR EFFICIENT QUERYING OVER AN ENCRYPTED DATABASE

Atheer Metaab Al Abbassi [1]
[1]*Department of Computer Science*
University of Technology
Baghdad, Iraq
etheer_78@yahoo.com

Abdul Monem S. Rahman[2]
[2]*Department of Computer Science*
University of Technology
Baghdad, Iraq
110003@uotechnology.edu.iq

Nidaa F. Hassan[3]
[3]*Department of Computer Science*
University of Technology
Baghdad, Iraq
110020@uotechnology.edu.iq

Abstract **- The increase in the amount of data in encrypted databases has caused problems in data processing and retrieval time. In traditional query processing methods, there are many difficulties in execute query over an encrypted database because it is time- consuming. In this paper, proposes technique for querying encrypted databases records, allows authorized users to execute queries without decrypting all the records of the encrypted database. In this technique, inverted tables include the numbers of 3D box cover locations that were created to enhance and speed up the retrieval time of query and improve an approach of data embedding according to the random 3D box. The proposed method has been examined on the Iraqi voter encrypted Database. The retrieval time in (second, millisecond) has been computed for the traditional method of query processing and proposed technique that using inverted tables. The retrieval time of query executing of proposed techniques without retrieval of all the records of the encrypted database is 10.870 (seconds, millisecond) where the retrieval time of query executing of conventional method that's retrieval of all the records of the encrypted database is 40.682 (seconds, millisecond).**

Index Terms - 3D Box; Encrypted Database; Inverted table; Query processing; Retrieval Time

I. INTRODUCTION

Data are critical resources that must be securely saved for the efficient transaction of a company. Typically, companies store sensitive information in secure databases. In traditional query processing methods, there are many difficulties in execute query over an encrypted database because requires decrypting all the records of the encrypted database to retrieve specific query processing that is led to large computations and time consuming. Cryptography is an important strategy in database security. Unlike encryption, traditional security techniques cannot provide optimal data security. Data encryption provides an important dimension in security and prevents unauthorized users from gaining illegal access and stealing sensitive contents of database, which are stored in storage media, such as CD-ROM, tapes and disks. DBMS can be defined as data collection in addition to collection of programs for accessing such data. The database includes certain information related to an organization. The main aim of DBMSs is to provide an approach for storing and retrieving database information which is simple and effective [1][2]. Organization databases include sensitive data that can be unprotected from attacks and misuse [3]. Many techniques, such as encryption and other steganography methods, can solve this type of problem [4-6]. Cryptography provides important security for the database. Unlike cryptography, conventional security techniques cannot provide sufficient data security [7]. Data encryption produces a significant dimension of security that prevents users from gaining illegal access and stealing data from the original database when stored in storage mediums (e.g. CD-ROM, tapes and disks) [8]. Nevertheless, encryption safely assists in system execution because querying cannot be directly implemented in the structural query language (SQL) of an encrypted database. SQL query can only work when encrypted data are decrypted. This entire process requires a certain amount of processing time. Although these mechanisms somehow restrict their applicability, several mechanisms have been suggested to solve this performance deterioration problem [9] [10].

II. RELATED WORKS

Providing sufficient security is a big problem for large databases. Thus, the encryption techniques of database management systems (DBMSs) can be used. Anyway, despite the high security provided by encryption, problems, such as system degradation efficiency caused by techniques of encryption, still exist.

Reference [11] suggested a private database query protocol for seeking encrypted records by using an equality test algorithm on the encrypted databases. This suggestion aims to find and execute an effective form of search condition at each fully homomorphic encryption (FHE) cipher-text by using the algorithm of an equality test.

Reference [12] produced a new technique that includes an indexing that used for searching the range queries in the large database. However, the

disadvantage of this technique, is only useful for numerical data and not for character data.

Reference [13] suggested a new method for searching queries on the encryption database by using a homomorphic encryption technique based on the ideas of Gandhi's method. This method has two phases. In the first phase, homomorphic query can be used with a ring-based FHE. In the second phase, we use the homo-morphic query to build a keyword search method in the smart grid. Reference [14] suggested that the range of attributes of user are dividing into set of intervals. On the client's side the compatibility between the original values of DB and intervals are preserved, on the other hand, interval information with encrypted tables are saved in the original database. Data are queried in efficiently manner by matching the original range and equivalent query information with the corresponding interval values.

Reference [15] proposed a new method that includes a designing of a B+-tree with values of plaintext, and then each B+tree node was encrypted and saved at an unauthenticated DBMS. The main B+tree was then executed at the unauthenticated DBMS as a table with (2 attributes), called (node ID) which is assigned by the system upon insertion, and the contents of encrypted node. This technique has pros and cons. the content of B+tree is invisible to an untrusted service provider of database is considered as pros, whereas cons are that it includes a large data processing on client machines.

According to the previous studies, there are time consuming issues when applying query over an encrypted database. Some of researchers, try to overcome this issue however, some problems still exist in query processing. The main contribution of this paper is to suggest inverted table and 3D box to speed the retrieval time of query processing without decrypting all the records of the encrypted database.

III. Database of Iraqi Voter

In order to examine the proposed technique, the Iraqi voter database was used, which is a database that includes information about (first name, second name, third name, ... etc.) as shown in table (1) and it encrypted by using Advanced Encryption standard algorithm with Galois Field (24) to make a balance between time and complexity and enhance query processing in terms of eliminating computational overhead.

overhead.

TABLE 1
IRAQI VOTER DATABASE STRUCTURE

ID	FNAME	SNAME	THNAME	PRE-DOB	GOV-MOT-ID	VRC-NAME-AR	CardserailNumber
1	صدام	فيصل	خليل	1/1/19٧٠	21	الخالص٢	215532700
2	حميد	علي	حسين	11/8/1979	21	الخالص٢	215533646
3	بيداء	هادي	احمد	1/30/1976	21	الخالص	316247142
4	انتصار	خالد	رقيب	2/1/1994	21	الخالص٢	215524933
5	محمود	راقي	كاظم	1/1/1993	21	الخالص٢	215522249
6	جسومة	هديب	ياسر	7/1/1963	21	الخالص٢	215533559
7	عمار	ايمن	سامي	3/26/1952	21	الخالص٢	215521032
8	مجد	تاج	حبيب	7/1/1954	21	الخالص٢	215503967
9	عمار	فاضل	اكرم	1/1/1989	21	الخالص٢	215521020
....	....	....	....	....	....	....	....
1000	عباس	علي	ياسر	12/22/1989	21	الخالص٢	215532824

IV. Nvironment of the Proposed Work

The proposed method consists of two main stages, the first included construction of 3D box to hide the encrypted records of database in randomly manner to increase complexity and the second stage included building of inverted tables for speeding up retrieval time of query executing.

A. Construction of 3D Box

In this stage, a three-dimensional box was built, the dimensions of the box are (x, y, z) and the length of each dimension is (10), meaning that the size of the box is (10, 10, and 10). It is used to hide the encrypted database records as each of the box's locations is a vector that contains all the information of the encrypted row as shown in figure (1).

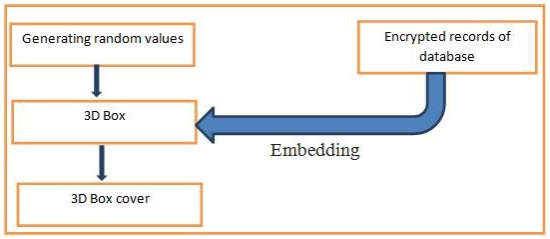

Fig.1 Structure of 3D box components.

Before the process of hiding the encrypted rows inside the three-dimensional box, a random matrix called permutation key was created based on pseudo random generator method for the locations of the box where it works to generate random values for the cells of the box and these values

range between (0,0,0 and 9,9,9). Depending on the permutation key matrix, the encrypted row will be embedded inside one of the cells of the box in random manner in order to increase the randomness and complexity of the encryption process, thus increasing the security level of the database.
For example: the first encrypted row that has zero identification in the original database, will be hide in the location (239) in the 3D box, where 2 is represent the X dimension, 3 is represent the Y dimension and 9 is represent the Z dimension of 3D box . The second encrypted row that has one identification in the original database, will be hide in location (683), where 6 is represent the X dimension, 8 is represent the Y dimension and 3 is represent the Z dimension of 3D box and the process continues for all the encrypted rows of the original database with random manner.
The 3D box cover as (shown in algorithm 1) is a regular 3D box, but the encrypted records are hidden inside it, meaning that each element inside the 3D cover is embed inside it an encrypted record from the database records I.e, all encrypted records inside the box were hidden in a random manner, meaning that (xyz) is a random number and not a sequence. The main goal of this stage is to hide the encrypted records and distribute them randomly to increase diffusion and complexity of the system and thus immune the database system against many types of attacks.

Algorithm 1 : 3D box Construction
Input: x, y, z (the length of dimensions of 3D box are x=100 y =100 z=100)
Output: 3D Box cover
Begin **Step1**: Generating random numbers of values with range (0 to (x*y*z) -1). **Step2**: Fill the 3D box locations with random numbers of values of $step_1$. **Step3**: Embed the encrypted records of the database in 3D box according to random 3D box locations (more diffusion) and save as 3D Box cover **End**

B. Construction of inverted tables

At this stage, the original database has been divided into a set of tables called (Inverted Tables) and the number of them is equal to the number of columns of the original database. Each inverted table contains two columns, where the first column contains the values for the first column that belongs to the original database without repeating, which means, it is similar to the content of the first column in the database with omitting duplicate values. As for the second column, it includes the random numbers for the three-dimensional box locations, which it contains the record numbers that include the first name in the first column in the database. Table (3-9) shows the structure of inverted table that derived from the first column of the original database (database of Iraqi voter). We notice that the first column in the table (3-9) contains all the names of the first column of the database without repetition while the second column includes numbers (038,849,213,237) refer to the locations of 3D box that contain all the names of (صدام). At this stage, the inverted tables are built for each database columns.

Table II
Inverted table structure.

PER_FIRST	ID
صدام	038 849 213 237
حميد	380 748 844 048 306 387
بيداء	314 544 436 443 457
انتصار	254 781 206 643 884 980 691 766
محمود	091 751 116 939 728
جسومة	446 978
عمار	946 379 292 137 815 203 280
محمد	805 813 717 187 912 784 025 114
صبرية	834 808
عباس	163 889 999 976 848 806 945 313
علياء	753 567
عروبة	402 583
فوزية	149 953
ريبوار	812

The main goal of creating these inverted tables is to speed up the search process in the database and minimize computational and time requirements when querying about a specific information. When a specific query is generating by the user of the database, instead of searching in a large database, the search is done in the inverted tables, which represents less space compared to a big database, due to the fact that the values of the inverted tables are non-repeated values and thus increase the speed of response to the queries received into the database, meaning that the proposed system is efficient in terms of responding to individual and complex queries. The mechanism of working the inverted tables depends on the principle of intersection of the information received between more than one inverted table to find the appropriate query. For more details, the following example illustrates the mechanism working of the inverted tables.

V. QUERY IMPLEMENTATION

At this point, specific data contained in the encrypted database is queried by the authorized person. This stage depends on a set of operations as follows:

1. When performing a specific inquiry, the transition will be made to the inverted tables, where the search process will take place within it to obtain the specific elements of the inquiry, i.e. obtaining random numbers that indicate the locations of the 3D box cover.
2. After that, it is moved to the locations specified in the 3D box cover to obtain the numbers of the encrypted records identified in the query which are founded in the encrypted database.
3. Then the encrypted records which were previously identified in the above step will be decrypted using AES with GF(2^4), that means decrypt only the specific encrypted records ,not all the encrypted database.
4. Then the specified items are decoded and then the records become readable by the authorized person.

The following diagram shows the mechanism of Query implementation:

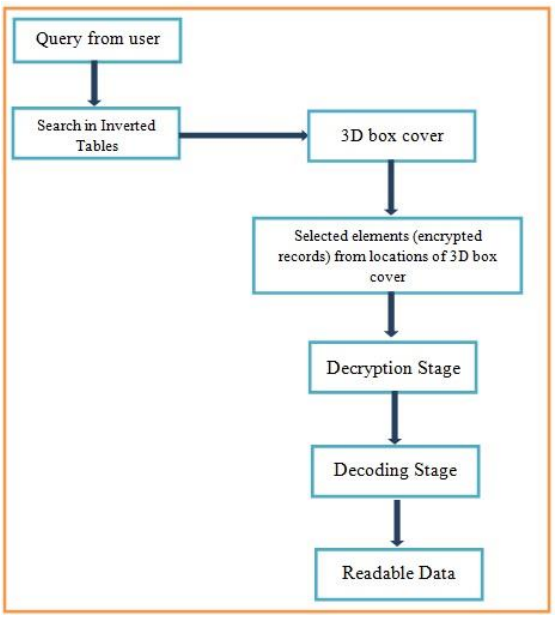

Fig. 2 Diagram of query implementation.

A. Algorithm outline

The proposed algorithm is presented as follows:

1. The user presents a query to the DB system.
2. The query is sent to the inverted tables.
3. From inverted tables we obtain the random numbers of 3D Box cover.
4. The SQL query is performed the searching operation on encrypted records.

If (the search process does match all the values of the query) then

the record is decrypted and decoded and then retrieved with real information to user.

Else

"Information is Not found".

5. Exit.

V I. EXPERIMENTAL RESULTS

Our technique is implemented on Iraqi voter database. The table of encrypted database includes 1,000 records used for testing. In practice, the time of query executing of the proposed method is faster than the traditional method without inverted tables as our technique divided the large database table into smaller tables called inverted tables. Moreover, the proposed technique is more effective in large databases that contain high repetition, especially those that contain Arabic names. Table (3) demonstrate the time consumed by the proposed technique, as compared with the conventional techniques.

Table III
shows Retrieval time of query executing between our algorithm and conventional algorithm

Database size (each record of Database has same size of bits equal to 160 bits)	Time (second, millisecond) Conventional method (Standard AES)	Time (second, millisecond) Proposed method
1000 record	40.682	10.870

According to Table (3) ,The retrieval time of query executing of proposed techniques is faster than the conventional method (Standard AES). The proposed method doesn't need to retrieval all the records of the encrypted database .

V I I. CONCLUSION

This research proposed an effective algorithm for query processing on encrypted database. The main goal of this paper to execute queries without retrieval of all the records of the encrypted database according to suggested inverted tables and 3D box. In traditional methods, the query implementing from a large encrypted database has time consuming as the database needs first to be decrypted entirely or partially before data retrieval to user . This paper addressed the time consuming problem in conventional method that causes system performance degradation and manipulates it by speed up query retrieving time to enhance system performance in terms of using inverted tables. Based on The retrieval time factor, The query processing of proposed techniques requires 10.870

(seconds, millisecond) while it requires 40.682 (seconds, millisecond) in the conventional method

REFERENCES

[1] Sharma, M., Chaudhary, A., and Kumar, S. "Query Processing Perfor-mance and Searching over Encrypted Data by using an Efficient Algorithm" International Journal of Computer Applications (0975 – 8887) Volume 62– No.10, January 2013.

[2] Mahdi, M. S., and Nidaa, F. H. "A SUGGESTED SUPER SALSA STREAM CIPHER." Iraqi Journal for Computers and Informatics ijci 44.2 (2018).

[3] Kaur, G. *"A Review on Database Security."* International Journal of Engineering and Management Research (IJEMR) 7.3 (2017): 269-272. B. Simpson, et al, "Title of paper goes here if known," unpublished.

[4] Mahdi, M. S., and Hassan, N. F. "A Proposed Lossy Image Compression based on Multiplication Table." Kurdistan Journal of Applied Research 2.3 (2017): 98-102.

[5] Morkel, T., Jan, H. E. and Martin, S. O. *"An overview of image steganography."* Information Security South Africa Conference ISSA. 2005.

[6] Tayyeh, H. K., Mahdi, M. S. and AL-Jumaili, A. S. A. "Novel steganography scheme using Arabic text features in Holy Quran." International Journal of Electrical & Computer Engineering (2088-8708) 9.3 (2019).

[7] Kadhim, A. and Mahdi, M. S. "Proposal of New Keys Generator for DES Algorithms Depending on Multi Techniques." Engineering and Technology Journal 32.1 Part (B) Scientific (2014): 94-106.

[8] Mahdi, M. S., and Hassan, N. F. "Design of keystream Generator utilizing Firefly Algorithm." Journal of Al-Qadisiyah for computer science and mathematics 10.3 (2018): Page-91.

[9] Wang, Z., Wang, W. and Shi, B. "Storage and Query over Encrypted Character and Numerical Data in Database" Proceedings of the 2005The Fifth International Conference on Computer and Information Technology, pp.77-81, 2005.

[10] Farhan, A. K. and Mahdi, M. S. "Proposal Dynamic Keys Generator for DES algorithms", islamic college university journal, 29 , 25-48, (2014).

[11] Cheon, J. H., Kim, M. "Optimized search and computer circuits and their application to query evaluation on encrypted data". IEEE Trans. Information Forensics and Security, 11(1):188–199, 2016.

[12] Li , J. and Omiecinski, E.R. "Efficiency and Security Trade-Off in Supporting Range Queries on Encrypted Databases" Technical Re-port, pp. 69-83, 2005.

[13] Sudharaka, P. "Homomorphic encryption and database query privacy" Diss. Memorial University of Newfoundland, 2016.

[14] Hacigümüs, H., Li , C. and Mehrotra, S. "Executing SQL over encrypted data in Database-Service-Provider Model" ACM SIG-MOD Madison, Wisconsin, USA, pp. 216-227, June 2002.

[15] Damiani, E., Vimercati, S. D. C., jodia, S., Paraboschi, S. and Samarati, P. "Balancing confidentiality and efficiency in untrusted relational dbmss", In Proceedings of CCS'03, 2003.