International Journal of Digital Information and Wireless Communications (IJDIWC) 4(2): 226-235
The Society of Digital Information and Wireless Communications, 2014 (ISSN: 2225-658X)

A New Orthogonal Cryptographic System for Database Security Based on Cellular Automata and Hash Algorithm

Dr. Mohammad V. Malakooti[1], Ebrahim Akhavan Bazofti[2]
[1]Faculty and Head of Department of Computer Engineering, Islamic Azad University (IAU), UAE Branch, Dubai, UAE

[2]Department of Computer Engineering, Islamic Azad University (IAU), UAE Branch, Dubai, UAE

[1] malakooti@iau.ae , [2] e_bazofti@yahoo.com

Abstract- In this paper, we have developed a new orthogonal cryptographic system for database security that has used both Cellular automata and Hash Algorithm. Our Algorithm consists of two different parts; Encryption/Decryption of database tables as well as the generation of the Authentication Tag for the activation of the attack alarm for the database tables while it is unlocked and in protected mode but it has been accessed by the illegal users.

Our proposed orthogonal cryptosystem is considered to be symmetric algorithm and uses a common key for both encryption and decryption processes as oppose to the asymmetric one that requires two keys, private and public keys. Since our transformation matrix is orthogonal, we have used the property of orthogonal matrix to calculate its inverse based on its matrix transpose rather than direct matrix inversion to save the calculation time during the decryption process.

We also have generated secret keys by applying the internal rules of cellular automata on the Malakooti Transform (M-T) to obtain the secret key matrix that can be used to be multiplied with the matrix of ASCII code obtained from the records of the database. To apply another level of security on the resulting encrypted code, the Hash values obtained from each record are multiplied by the elements of the secret key matrix and the XOR operation is performed on the resulting values and the elements of the encrypted codes.

In addition, we also proposed a robust and fast algorithm for the database security and authentication that automatically and accurately will generate the Hash values for the entire rows of the database tables to obtain a unique Hash value for each table. This unique hash value can be used to check the validity of the data inside the database and guarantee the authentication of all information in each database.

Our proposed method is capable of detecting any slight change that might be occurs on the database while it is in the protected mode. The generated Hash value will be calculated from the records elements of the database periodically to be compared with the value of the Hash value stored outside database for the authentication. Should the generated Hash value be different from the stored Hash value, the alarm flag would be activated to inform the administrator about unauthorized change of database while in protected mode via SMS or Email.

Keywords**:** Cryptosystem, Authentication, database, Tag, Malakooti Transform, Hash Value, Cellular Automata, Decryption, Encryption.

I.INTRODUCTION

Database security is the most important issue of cloud computing, information sharing and processing where many users are assigned to share the same information stored on the distributed networks or database servers. Although there are many security checks and authentication techniques to identify the users before they are allowed to access the data centers or database servers but yet the unauthorized users and hackers can always find a way to bypass all these security checks and illegally access the database. The multilayer encryption is one the best ways to secure the contents of database table and prevent the access of the illegal users from the original data even they have reached to the core of data centers.

They are several different encryption techniques that have been used for database security but none of them have used three levels of security as we applied. In addition we have applied another algorithm called

Malakooti-Bazofti Hash algorithm to generate a unique Tag or Hash pattern for the entire database content to be compared with previous stored Tag. If the new Tag is different from the previous Tag, then the system will activate the Attack Alarm and informs the administrator via SMS or Email.

The idea of securing databases using cryptographic system is not new and several researchers have proposed new techniques for the database encryption. Gudes, Koch and Stahl [1] have presented anew method for database encryption based on substitution, transposition,
Reduction and expansion of data items but it preserves data structures. Davida, Wells and Kam [2] have proposed an encryption technique based on the Chinese Remainder Theorem in which some algebraic operation are performed on the content of database and each records are encrypted by applying some mathematical operations. The encrypted data also can be decrypted by the inverse operation using Chinese remainder theorem. Several other researchers including [3-6] have used Data Encryption Standard (DES) and RSA algorithm to perform the encryption and decryption based on the public and private keys as well as performed the authentication based on the checksums of the data elements and the checksum are obtained from the identifiers and the database key.
Our proposed method for encrypting the database contents is based on the Malakooti Orthogonal Transform in which its orthogonal property can be used to invert the Transformation matrix by Matrix Transpose operation instead of direct calculation of the matrix inversion, during the decryption process. To increase the level of security on the database contents we have obtained the matrix of secret byesby applying the Cellular Automata on the elements of the M-T matrix. Once the secrete key matrix is calculated it will be multiplied with the matrix of coded data elements and finally XOR operation is applied on the resulting elements and the Hash values derived from Hash function applied on the corresponding database records.
In addition to applying three levels of security on database based on the M-T transform, Cellular Automata, and Hash Algorithm, we have generated a Unique Hash pattern or Authentication Tag based on Malakooti-Bazofti (MB) Hash Algorithm, in which the generated Tag can be compared with the stored Tag and Any difference between these two Tags will activate the attack alarm to inform the administrator via SMS or Email. The proposed Attack Alarm Tag will make our cryptographic system unique, robust, reliable, and fully alert in which can be applied for the highly secure database used over the distributed system, cloud computing environment, and internet.

II. MALAKOOTI TRANSFORM ALGORITHM

The Malakooti Transform, M-T, is an orthogonal transform similar to the Hadamard Transform that was developed by Mohammad V. Malakooti in 1987 for the data compression, encryption, and watermarking. Once this transform is applied on the data matrix the resulting coefficients contained useful information about the spectral characteristic of the underlying data matrix and can be used for data transmission, encryption and compression. Many of the databases elements are highly redundant and this transform can be applied to reduce the redundancy and increase the data storage capacity. The optimal selection of M-T coefficients can be used to reconstruct or to represent the desired database elements with less coefficients and resulting in a saving of transmission speed and memory [11].

III. GENERATION OF M-T MATRIX

Assume that the initial value of the first order M-T matrix,M_o, is equal to one, thus $M_o = 1,$ (3.1)
and the elements of the second order M-T matrix, M_1 , is formed according to following equation:

$$M_1 = \begin{bmatrix} aM_o & abM_o \\ -abM_o & aM_o \end{bmatrix} = \begin{bmatrix} a & ab \\ -ab & a \end{bmatrix}, \quad (3.2)$$

Where "a" and "b" are two constant parameters to change the content of M-T Matrices.
The matrix M is a 2 x 2 anti-symmetric unitary matrix

$$M_1^T M_1 = M_1 M_1^T = cI, \quad (3.3)$$

Where the matrix I is a 2 x 2 identity matrix and constant parameters c is equal to the determinant of M_1. Thus,

$$c = a^2(1 + b^2) \quad (3.4)$$

Thus,M_1 inverse is given as

$$M_1^{-1} = \frac{M_1^T}{c} \quad (3.5)$$

Similarly, the fourth order M-T matrix,M_2 can be obtained according to (3.6)

$$M_2 = \begin{bmatrix} aM_1 & abM_1 \\ -abM_1 & aM_1 \end{bmatrix} \quad (3.6)$$

The matrix M_2 is a 4 x 4 anti-symmetric unitary matrix

$$M_2^T M_2 = M_2 M_2^T = \mathbf{C^2\, I}, \quad (3.7)$$

where the matrix I is an 4 x 4 identity matrix, C is given in (3-4), and the inverse of M_2 is calculated according to

$$\mathbf{M_2^{-1}} = \frac{\mathbf{M_2^t}}{\mathbf{C^2}} \quad (3.8)$$

Without loss of generality, the 2^k x 2^k M-T matrix, M_k can be obtained from

$$M_k = \begin{bmatrix} aM_{k-1} & abM_{k-1} \\ -abM_{k-1} & aM_{k-1} \end{bmatrix}, \quad (3.9)$$

AndM_k inverse is given according to (3.10)

$$\mathbf{M_k^{-1}} = \frac{\mathbf{M_k^t}}{\mathbf{C^k}} \ . \quad (3.10)$$

Using the Kronecker product notation

$$A \otimes B = \begin{bmatrix} a_{11}B & a_{12}B & \cdots & a_{1n}B \\ a_{21}B & a_{22}B & & a_{2n}B \\ \vdots & & & \vdots \\ a_{n1}B & a_{n2}B & \cdots & a_{nn}B \end{bmatrix}, \quad (3.11)$$

Thus, the M-T matrices can written accordingly,

$$M_1 = M_1 \otimes M_o = \begin{bmatrix} aM_o & abM_o \\ -abM_o & aM_o \end{bmatrix} \quad (3.12)$$

And

$$M_2 = M_1 \otimes M_1 = M_1 \otimes (M_1 \otimes M_o) = (M_1 \otimes M_1) \otimes M_o = M_1^{(2)} \otimes M_o = M_1^{(1)} \otimes M_1, \quad (3.13)$$

Where$M_1^{(2)}$is the Kronecker power 2 of M_1 and the symbol ⊗ denotes the Kronecker product. Similarly,

$$\begin{aligned} M_3 &= M_1 \otimes M_2 \\ &= M_1 \otimes M_1^{(2)} \otimes M_o \\ &= M_1^{(3)} \otimes M_o \\ &= M_1^{(2)} \otimes M_1 \end{aligned} \quad (3.14)$$

. . . .

$$\begin{aligned} M_k &= M_1 \otimes M_{k-1} \\ &= M_1 \otimes M_1^{(k-1)} \otimes M_o \\ &= M_1^{(k)} \otimes M_o \\ &= M_1^{(k-1)} \otimes M_1. \end{aligned} \quad (3.15)$$

We can easy generate the elements of the M-T matrices by assuming that a=1, b=2, and expand the idea to get

M-T matrices of size 2, 4, 8, 16,… recursively as following:

$$\mathbf{M_1} = \begin{bmatrix} \mathbf{1} & \mathbf{2} \\ \mathbf{-2} & \mathbf{1} \end{bmatrix} \quad (3\text{-}16)$$

$$\mathbf{M_2} = \begin{bmatrix} \mathbf{1} & \mathbf{2} & \mathbf{2} & \mathbf{4} \\ \mathbf{-2} & \mathbf{1} & \mathbf{-4} & \mathbf{2} \\ \mathbf{-2} & \mathbf{-4} & \mathbf{1} & \mathbf{2} \\ \mathbf{4} & \mathbf{-2} & \mathbf{-2} & \mathbf{1} \end{bmatrix} \quad (3.17)$$

The generation of any size M-T matrix can be obtained easily by a recursive equation and it can be multiplied by a data matrix of the same size to obtain the encrypted database with a high speed and accuracy due to its orthogonal property. In addition, we can take advantage the orthogonal property of M-T transform matrix and can calculate its inverse by using orthogonal property of the M-T matrix and calculate the inverse by using its matrix transpose rather than direct calculation of the inverse matrix.

One can easily see that M-T matrix has special features that can be used to encrypt the content of database tables with low calculation cost and high accuracy. The process of decryption is also similar to encryption but the inverse of transformation matrix can be obtained via matrix transpose rather than direct inverse calculation. This high speed inverse transformation can decrease the calculation cost ad increase the speed of decryption process[12].

IV. CELLULAR AUTOMATA BASICS

A Cellular Automaton (CA) is a discrete model consists of a regular grid of cells that each cell has finite number of states but usually they holds two states of on and off. The grids are usually considered to be one dimensional or two dimensional but higher order dimension grids also are also possible. The application of CA is in several filed of science and technology including mathematics, physics, biology,

and other branches of sciences. In two dimensional CA grids each cell has a few neighborhood cells. If the neighboring cells are located at right side, left side, top, or bottom of the specified cell, they are called Von Neumann Neighborhood in the honor of Von Neumann who worked with Stanislaw Ulam at Los Alamos National Laboratory, New Mexico, USA, 1940.

An initial state of each cell at time t=0 is givens but the new state of each cell at other times, t>0, is calculated based on current state of the cell and the sates of cells in its neighborhood. The mathematical rules for updating the state of all cells are the same and it will not be changed over the time [7].

Given the rule, anyone can easily calculate future states, but it appears to be very difficult to calculate previous states. However, the designer of the rule can create it in such a way as to be able to easily invert it. Therefore, we can say that it is Cleary a trapdoor function, and can be used as a public-key cryptosystem. The security of such systems is not currently known [8].

The idea of Cellular Automata is intuitive and simple, and it consists of a regular grid of cells. Each of which may be in a predetermined number of states. Cell a_{i+1} with the following rule:

$$\text{Cell } a_{i+1}=\text{Cell } a_{i-1} + \text{Cell } a_i \qquad (4.1)$$

Suppose that we have the string of 11010010 and we want to use the above rule such as our cellular automata, then the Generated string will be 10101011. Table-1 shows the Generated string of internal cellular automata rule [9].

Table 1: An Example of Cellular Automata

Input string	1	1	0	1	0	0	1	0
Internal cellular rule	**Cell a_{i+1}=Cell a_{i-1} + Cell a_i**							
Generated string	1	0	1	0	1	0	1	1

V. PROPOSED METHOD

In addition to our suggested method for encrypting the database using Malakooti Orthogonal Transform that applied on the matrix of ASCII values representing the elements of each record in the database, the idea of Cellular Automata also is used. We have used the mathematical rules of Cellular Automat and applied on the elements of MT matrix to generate the elements of secret key matrix, Kt, that is required to multiply by each row of coded matrix, M_C, to obtain the elements of the encrypted data matrix, Me, that represents the encrypted version of the corresponding data table.

We also have proposed an entirely fast, secure, and irreversible Hash algorithm to obtain the Hash streams of the each database stored on the server. We have called this method as the Log2 Algorithm, because the $\log_2$ of number of records in the database is calculated to divide the entire records in to "N" different groups. Once the number of rows or records in each group is obtained, then our proposed Hash Algorithm based on the consecutive XOR and NOR operations at each stage is calculated.

This approach is repeated so that hashed keys, rows and columns and every database table could be turned into a set of characters so called the Authentication Tag. This Tag can be saved on a safe place that is totally different than places where the databases are stored and it will be used for the database protection and authentication.

VI. HASHING TECHNQUES

The databases are the most valuable resources that are stored on the servers that can be accessed and shared by several clients with different level of privilege, authentication and security. Thus, the confidentiality, protection and maintenance of these valuable resources are highly recommended. Thus, we have proposed and applied several levels of security on the database to obtain the required security on the database servers.

One of the effective technique to protect the database and prevent it from the unauthorized access and illegal manipulation of its contents by the hackers and attackers is to apply the fast, efficient, and robust encryption algorithm to apply several levels of security on the databases and finally use the hash algorithm to obtain the fixed size hash values and store them on severs or send them through distributed network to other servers. Our proposed methods are based on Encryption Algorithm, Cellular Automata,

hash function operation and finally calculation of the Hash Value to generate the Authentication Tag.

Hash Value Generation Method

Due to the significance of data accuracy and originality of the available information in each database table, we have proposed the Hash Value, H_V, generation algorithm to be applied for each table of the database. This hash value algorithm provide the final hash value as the authentication Tag that is able to detect any unauthorized access and database manipulation while database is unlocked and in protected mode. The new generated hash value will be compared with the old hash value that is stored on the safe location. Once any slight different between these two hashes are reported, the attack alarm flag will be activated and the network administration will be informed via SMS or Email.

Hash Algorithms

1- Calculate the ASCII value of each record and save it into data matrix, M_a.
2- Count the number of records in each database tables, N.
3- Compute the logarithm base 2 of N to get M.
4- Divide the records of each table into M sections.
5- Perform the **XOR & NOR** operations on the selected records in each group to get level -1 hash operation.
6- Replace the value of N with M, N=M.
7- Perform the operations of steps 3-6 to generate the level- 2 of hash operation.
8- Repeat the operation of steps 3-6 to change all records in to one hash value, H_v, required for the authentication.

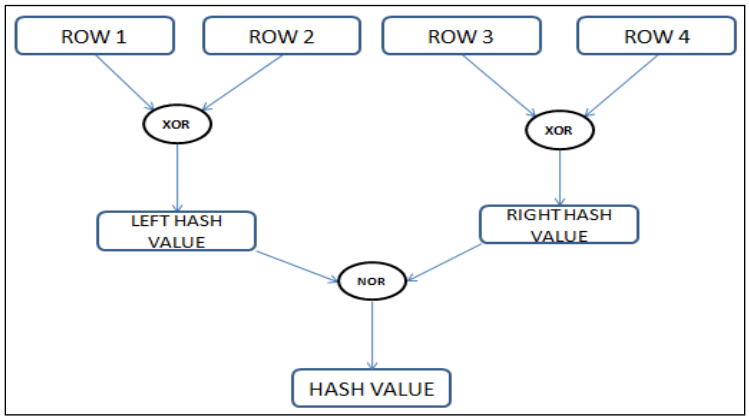

Figure 1: Calculation of the Hash Value for Authentication

The Log2 Algorithm for calculation the Hash Value

Input is $a_{11}, a_{12}, \ldots\ldots, a_n$

For **I= 2: $\log_2^n$+1**

T= **$n/2^i$-2**

S= **$n/2^i$-1**

IF T is even Then

For j=1

IF I is even Then

$a_{ij} = a_{i-1,j}$ XOR $a_{i-1,(n-(j-1))}$

Else

$a_{ij} = a_{i-1,j}$NOR$a_{i-1,(n-(j-1))}$

End

Else T is odd

For j=1

IF I is even

$a_{ij} = a_{i-1,j}$XOR $a_{i-1,(n-(j-1))}$

$a_{is} = a_{i-1,s}$

Else

$a_{ij} = a_{i-1,j}$ NOR $a_{i-1,(n-(j-1))}$

End

End

End.

VII. KEY GENERATION ALGORITHM

The Generation of secret key matrix is based on the Malakooti-Bazofti(M-B) Algorithm as following:

1- Set the size of M-T Input matrix equal to the number of records in the database, i.e, N = 4, and generate the M-T matrix as following:

$$M_2 = \begin{bmatrix} 1 & 2 & 2 & 4 \\ -2 & 1 & -4 & 2 \\ -2 & -4 & 1 & 2 \\ 4 & -2 & -2 & 1 \end{bmatrix}$$

2- Specify the Rule of Cellular Automata for Key generation algorithm, For example , we used this rules for cellular automata:

Cell a_{i+1}=Cell a_{i-1} + Cell a_i (7-1)

3- Apply the internal rule of cellular automata on all elements of M-T matrix to get secret key matrix.

$$Kt = \begin{bmatrix} 1 & 3 & 4 & 6 \\ -2 & -1 & -3 & -2 \\ -2 & -6 & -3 & 3 \\ 4 & 2 & -4 & -1 \end{bmatrix}$$

4- Compute the determinant of the key matrix to make sure that key matrix is invertible. Its inversion is required for the decryption process.

5- Generate a new key matrix if the determinant of the key matrix is equal to zero.

The inverse of matrix K_t must exist otherwise the decryption process cannot be performed. One can easily show that possibility of the determinant of Kt to be zero is very low because the rows of database table have different value and chance of the determinant to be zero is very low but it is not impossible.

The Source Code of key Generation Algorithm

```
int u = 0,g;
publicvoidKey_Gen_Cellula()
    {
     if (u == 0)
                {
    for (inti = 0; i<= 7; i++)
              {
    for (int j = 0; j <= 7; j++)
                 {
                    g = j;
    for (int k = j + 1; k <= j + 1; k++)
                 {
    if (k == 8) { }

    if (j == 0)
                        {
    Matrix_KeyGen_Cellula_Data[i, j] = M_Transfer[i, j] + 1;
                        }
    elseif (j == 7) Matrix_KeyGen_Cellula_Data[i, j] = 1;
else
Matrix_KeyGen_Cellula_Data[i, j] = M_Transfer[i, g - 1] +
M_Transfer[i, k];
              }

            }
        }
for (inti = 0; i<= 7; i++)
        {
for (int j = 0; j <= 7; j++)
            {

Matrix_KeyGen_Cellula_Copy_Data[I,,j]                    =
Matrix_KeyGen_Cellula_Data[i, j];

            }
        }
    }
else
    {
for (inti = 0; i<= 7; i++)
        {
for (int j = 0; j <= 7; j++)
            {

Matrix_KeyGen_Cellula_Data[i,j]                    =
Matrix_KeyGen_Cellula_Copy_Data[i, j];

            }

        }

for (inti = 0; i<= 7; i++)
        {
for (int j = 0; j <= 7; j++)
            {

                g = j;
for (int k = j + 1; k <= j + 1; k++)
            {
if (k == 8) { }
```

```
if (j == 0)
                {
Matrix_KeyGen_Cellula_Data[i,j]                    =
Matrix_KeyGen_Cellula_Copy_Data[i, j] + 1;
                }
elseif (j == 7) Matrix_KeyGen_Cellula_Data[i, j] = 1;
else
Matrix_KeyGen_Cellula_Data[i,j]                    =
Matrix_KeyGen_Cellula_Copy_Data[i,g-1]             +
Matrix_KeyGen_Cellula_Copy_Data[i, k];
                }
              }
            }
for (inti = 0; i<= 7; i++)
            {
for (int j = 0; j <= 7; j++)
              {
Matrix_KeyGen_Cellula_Copy_Data[i,j]               =
Matrix_KeyGen_Cellula_Data[i, j];

              }
            }
          }
u++;
      }
```

VIII. DATABASE ENCRYPTION ALGORITHM

The encryption algorithm for the records of the database tables is given as following:

1- Read the entire table of database or just read some important column of database tables (fields of database table).
2- Calculate the ASCII code of the records and save them into one matrix.
3- Insert the ASCII code of all records into a matrix called ASCII Code matrix, Ma.

Table 2: The Content of original database table

stunum	name	lname	score
8800155	Alis	ayne	18
90115302	Mohsen	dehghani	10
90115807	BOB	Anderson	20

$$M_a = \begin{bmatrix} 57 & 0 & 48 & 0 & 49 & 0 & 49 & 0 \\ 53 & 0 & 56 & 0 & 48 & 0 & 55 & 0 \\ 32 & 0 & 66 & 0 & 79 & 0 & 66 & 0 \\ 32 & 0 & 65 & 0 & 110 & 0 & 100 & 0 \\ 101 & 0 & 114 & 0 & 115 & 0 & 111 & 0 \\ 110 & 0 & 32 & 0 & 50 & 0 & 48 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

We have obtained an 8*8 Matrix for three rows of the database table.

4- Generate the elements of M-T matrix according to the size of the ASCII matrix.

$$M3 = \begin{bmatrix} 1 & 2 & 2 & 4 & 2 & 4 & 4 & 8 \\ -2 & 1 & -4 & 2 & -4 & 2 & -8 & 4 \\ -2 & -4 & 1 & 2 & -4 & -8 & 2 & 4 \\ 4 & -2 & -2 & 1 & 8 & -4 & -4 & 2 \\ -2 & -4 & -4 & -8 & 1 & 2 & 2 & 4 \\ 4 & -2 & 8 & -4 & -2 & 1 & -4 & 2 \\ 4 & 8 & -2 & -4 & -2 & -4 & 1 & 2 \\ -8 & 4 & 4 & -2 & 4 & -2 & -2 & 1 \end{bmatrix}$$

5- Multiply the M-T matrix with ASCII code matrix.

$$C_d = M_T * M_a \qquad (8\text{-}1)$$

6- Set the size of M-T Input matrix equal to the number of records in the database.
7- Generate the secret key matrix by applying the rule of Cellular Automata.

$$Kt = \begin{bmatrix} 4 & 10 & 22 & 17 & 34 & 17 & 21 & 1 \\ 1 & -14 & 9 & -34 & 17 & -31 & 11 & 1 \\ 1 & -4 & -11 & -9 & -18 & -6 & -9 & 1 \\ 7 & 14 & 0 & 18 & -9 & 15 & -4 & 1 \\ 1 & -9 & -31 & -9 & -18 & 4 & 1 & 1 \\ 7 & 24 & -10 & 18 & -9 & -5 & 1 & 1 \\ 7 & 4 & 5 & -7 & -14 & -5 & -9 & 1 \\ -5 & -2 & -7 & 14 & -7 & 13 & -4 & 1 \end{bmatrix}$$

$$M_C = K_{t*} C_d \qquad (8\text{-}2)$$

8- Multiply the Hash Value of each row obtained from Hash function, H_f, into the corresponding row of the secret key matrix, K_t, to get complex secret key values. Apply these keys on each row of M_C Matrix to obtain the Encrypted data matrix, Me, that represents the encrypted version of the corresponding data table.

$$Me = M_C XOR(H_f * Kt) \quad (8\text{-}3)$$

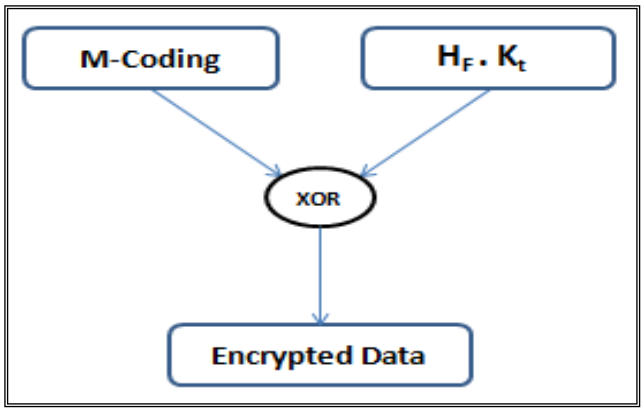

Figure 2:Final Encryption process using the XOR

Table 3- The Hash Table

	id	hash
	37	KG01S9zUhXiCw…
	38	t8eIMnon5bu9O…
▶	39	mCnBdN2H7O/a…

IX. DATABASE DECRYPTION ALGORITHM

The decryption algorithm for the records of the database tables is given as following:

1- Read the content of the encrypted data matrix, Me, that was transformed, encrypted, and Hashed during the encryption process.

2- Apply the XOR operation on Me and (H_f*Kt) to get the matrix of M-coding, M_C.

$$M_C = Me \text{ XOR } (H_f * Kt)$$
$$= M_C \text{ XOR}(H_f * Kt) \text{ XOR } (H_f * Kt) \quad (9\text{-}1)$$

3- The result of above process will transform the encrypted, hashed matrix into the M-coding.

4- Apply the inverse of K_t on the M_C matrix to obtain the matrix of coded data.

$$C_d = (K_t)^{-1} * M_C =$$
$$= (K_t)^{-1} * K_t * C_d \quad (9\text{-}2)$$

$$Mc = \begin{bmatrix} -4944 & -9183 & 6050 & -111 & 148 & 4830 & -3607 & 813 \\ -5835 & -11550 & 9045 & -1980 & 2640 & 4770 & -3440 & 660 \\ -9981 & -17607 & 12285 & -5559 & 7412 & 3780 & -3323 & -243 \\ -13197 & -24894 & 18285 & -13563 & 18084 & -2100 & 1529 & -411 \\ -13302 & -24429 & 17160 & -3633 & 4844 & 9900 & -7661 & 999 \\ -1098 & -1746 & -2210 & 6708 & -8944 & 6150 & -4264 & 2376 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

5- Apply the inverse of the M_T on the C_d matrix to obtain the matrix of ASCII code.

$$M_a = (M_T)^{-1} * C_d$$
$$= (M_T)^{-1} * M_T * M_a \quad (9\text{-}3)$$

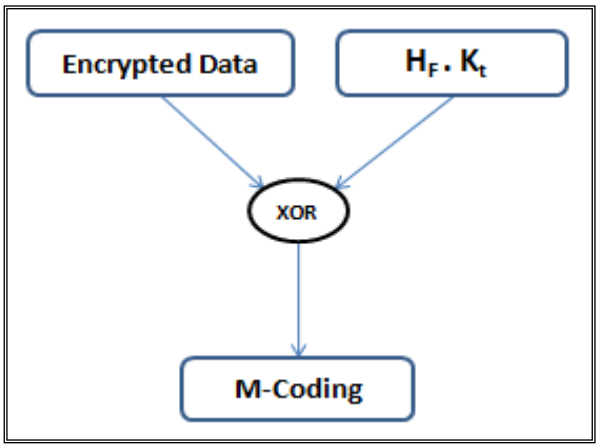

Figure 3: Final Calculation of M-coding

Table 4- Decrypted Table of the database

id	stunum	name	lname	score
32	8800155	Alis	ayne	18
33	90115302	Mohsen	dehghani	10
34	90115807	BOB	Anderson	20

X.GENERATION OF AUTHENTICATION TAG

In this section we have generated an Authentication Tag Value based on the Malakooti- Bazofti (M-B) Algorithm, in which multiple level of XOR and NOR operations are applied on each group of hash values obtained from the database records. Thus, all hash values that obtained from the records are combined together to obtain a unique hash value require for the database security and authentication.
We have proposed a robust and fast algorithm for the database security and authentication that automatically and accurately will generate the Hash values for the entire rows of the database tables to obtain a unique Hash value for each table. This unique hash value can be used to check the validity of the data inside the database and guarantee the authentication of all information in each database. Should any slight change is made on the database while the database is in the protected mode, the generated hash value would be totally different from the stored hash value and the software system automatically will activate the alarm flag to inform the administrator about unauthorized change of database via SMS or Email.
In our proposed method, we have divided each database table into "N" different records and used the concept of parallel algorithm to calculate the Hash values of all records of each table in database as well as to calculate the Hash values of all columns, accurately and efficiently. Once the hash values are calculated, the fast XOR and NOR operations are applied on the generated Hash values to obtain a unique hash values for the Authentication Tag.

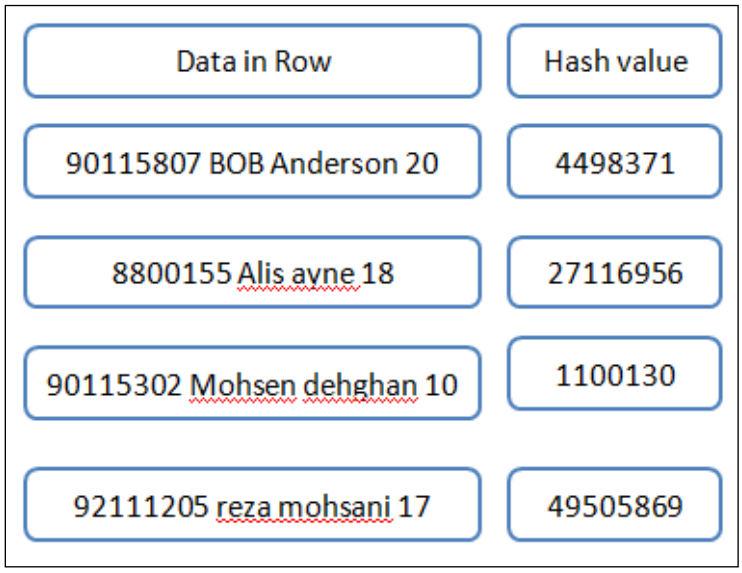

Figure 4:Calculated HashValue for each Row

XI. CONCLUSION AND FUTURE WORK

The objective of this research is to apply multilevel of security on database and protect the contents of the database tables from the unauthorized users and hackers who tried to access our database illegally and perform the read, write, change operations on the database tables. To obtain the above objectives we have transformed the contents of the database records into to ASCII code and then applied the M-T transform on the ASCII code matrix to calculate the matrix of coded data. The matrix of coded data also will be multiplied by the matrix of secret keys that obtained by applying the rule of cellular automata on the elements of the M-T matrix to calculate the matrix of M-coding. To increase additional level of security on the encrypted data, the hash value of each record of the database is calculate and then multiplied by matrix of secret key to obtain the bit patterns that can be used in XOR operation with the matrix of M-coding to obtain the highly secure encrypted values for the records of the database.
We have proposed a robust and fast algorithm for the database security and authentication that automatically and accurately will generate the Hash values for the entire rows of the database tables to obtain a unique Hash value for each table. This unique hash value can be used to check the validity of the data inside the database and guarantee the authentication of all information in each database.

Should any slight change is made on the database while the database is in the protected mode, the generated Hash value would be totally different from the stored hash value and the software system automatically will activate the alarm flag to inform the administrator about unauthorized change of database via SMS or Email.

Our proposed algorithm applied three levels of security on the database contents and guarantees the security of database tables. It also use the orthogonal property of M-T matrix to obtain its inverse using its transpose rather than the direct inverse calculation, required for the decryption process More work need to done to obtain a fast algorithm to perform the cellular automata for generating the matrix of secret keys.

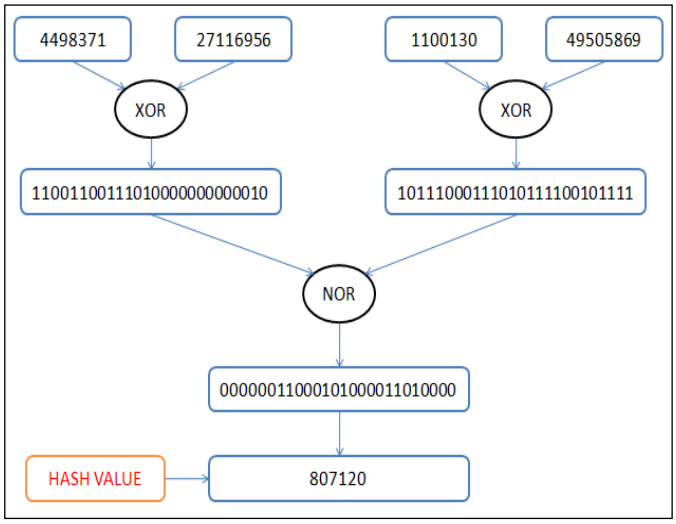

Figure 5: Calculated Hash Value for a Table

XII. REFERENCES

[1] E. Gudes, H.S. Koch and F.A. Stahl "The application of cryptography for data base security". Proceedings of AFIPS National Computer Conference, 1976, pp. 97-107.

[2] G.1. Davida, D.L. Wells and J.B. Kam, "A database encryption system with subkeys", ACM Trans. on Database System, Vol. 6, No.2, June 1981, pp. 312-328.

[3] A.Afyoni "Database Security and Auditing", 2005, Amazon.com.

[4] L.Bouganim, Y.Guo," Database Encryption", Le Chesnay, France, 2009.

[5] L. M.Batten"public key cryptography` application and attacks", 2013, john wily & Sons.

[6] C. Peikariand S.Fogie, Sams, "Wireless maximum security", 2002, Sams.ISBN 0-6723-2488-1.

[7] M. Thomas "Cellular Automata", Nova Science, 2010, ISBN 978-1-62100-148-5(eBook).

[8] T.Ceccherini- Silberstein, Michel Coornaert, "Cellular Automata and Group", Springer, 2010, ISBN 978-3-642-14033-4.

[9] J. L. Schiff, "Cellular Automata", John Wiley & Sons, 2008, ISBN 978-0-470-10879-0.

[10] A.Mousa, O. S.Faragallah, S. El-rabaie and E M Nigm "Security Analysis of Reverse Encryption Algorithm for Databases", IJCA Journal, 66(14):19-27, March 2013. New York, USA.

[11] S.Kulkarni, S. Urolagin, "Databases and Database Security Techniques", IJETAE,ISSN 22502459,Vol 2,Issue 11, November 2012.

[12] M. V. Malakooti, M. Raeisi Nejad Dobuneh "Developing a Lossless Digital Encryption System for Multimedia Using Orthogonal Transforms", Malaysia, 2011.