International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056
Volume: 04 Issue: 05 | May -2017 www.irjet.net p-ISSN: 2395-0072

# Implementing ETSFS Algorithm for Database Security

**Nandkishor B. Jawanjal[1], Prof. Pritish A. Tijare[2], Prof. Swapnil N. Sawalkar[3]**

[1]*M.E. Computer Engineering student, Department of CSE Sipna COET, Amravati, Maharashtra, India*
[2] *Associate Professor, Department of Information Technology, Sipna COET, Amravati, Maharashtra, India*
[3] *Assistant Professor, Department of Information Technology, Sipna COET, Amravati, Maharashtra, India*

---

**Abstract -** Database encryption is a se*curity system includes different encryption algorithms for database security. Most of the Organization stores their information in enormous databases that empowers uncomplicated recovery, controls, and furthermore helps in a proficient method for sharing. Database security has now turned into a more dynamic issue as information is the best advantage for any association. Because of the quick increment in the database issue as data is the greatest asset to any organization. To beat these issues, various security strategies have developed to ensure the information in databases. For Example Enhanced Transposition, Substitution, Folding & Shifting (ETSFS). Each of them has its particular benefits and faults. Using the ETSFS algorithm to secure sensitive data (information) in a databases .It has imperative on information size and number of exceptional characters, the proposed technique change concentrates on the encryption of expansive information considering a wide range of unique characters and an arbitrary generator is utilized for producing keys in substitution stage. The proposed strategy of the paper concentrated on the future work of the ETSFS algorithm for secure sensitive Email system.*

***Key Words: Transposition, Substitution, Shifting, Folding, Encryption, Decryption.***

## 1. INTRODUCTION

Information security has reliably been a noteworthy issue in web applications. Database security has vital significance in organization, military, personnel and government areas. Associations are putting away huge measure of information in database for information mining and different sorts of investigation. Some of this information is viewed as delicate and must be shielded from exposure. Challenges for security in database are expanded because of the colossal prevalence of e-business. As of late, insider assaults assembled more consideration than occasional episodes of malware. Database frameworks are normally conveyed somewhere inside the organization system and subsequently insiders has the most effortless chance to assault and trade off them, and after that take the information. So information must be shielded from inside aggressors too. Numerous traditional database security frameworks are proposed for giving security to database, yet at the same time the delicate information in database are helpless against assault on the grounds that the information are put away as plaintext only.[5]

Within the sight of security dangers, database security is getting to be noticeably a standout amongst the most earnest difficulties since much harm to information can happen in the event that it experiences assaults and unapproved get to. With databases in complex, multi-layered applications, assailants may achieve the data inside the database. Harm and abuse of delicate information that is put away in a database does influence a solitary client; as well as conceivably a whole association.

Database encryption would be one of the possible solutions where be one of the access to sensitive information is dependent on the key available, which promises a minimal damage and high performance when it is effective [1].

The ETSFS algorithm provide by avoiding the constraint on the data size and special characters by proposing the usage of all special characters on the data size. This improvement allows handling all special characters and different sizes of input data for processing performed dynamically depending on the input data given by the user; and based on the input size, keys are generated that shows a variation from existing DES algorithm. It showed a successful implementation by accepting almost all special characters.

## 2. LITERATURE REVIEW & RELATED WORK

Data plays a very important role and is stored in database system which should be organized such that it safeguards the data. Most of the organizations sensitive data is housed in database and a backup is maintained for future use. Unauthorized access is one of the serious threats and should be addressed to enhance database security. Encryption, which plays a important role in safeguarding the information, is defined as the process of transforming information into no readable form except by those holding a key to decrypt.[1]
The database security mechanisms, algorithms like TSFS, DES, and AES came into focus, which are different from other and had a few advantages and disadvantages based on their optimization ways.[7] The DES algorithm is one of the well-known symmetric key algorithms considered as insecure for many applications and presents AES as a replacement. D.

Manivannan, R. Sttjarani [3] proposed well-organized encryption technique using the symmetric - key. popularly known as TSFS algorithm, which includes transpositions and substitutions as features in the techniques that limits encryption and decryption operation times. Later an enhanced TSFS is proposed which is an extended work on TSFS which can encrypt the data that contains alphanumeric and few special characters ensuring high level of security to encrypted data but imposes few constraints on the data size and the special characters used [3].

The cloud computing is the top threat identified by the Cloud Security Alliance. Attackers can infiltrate a public cloud.[8]

For providing effective and more security for the database these three keys are expanded in to 12 sub keys by using the key expansion technique.[6]

The principle quality of the calculation is in the substitution change in light of the fact that choosing the key for discovering figure gave greater security to the encode picture. Pictures are in database are encoded and afterward the encoded pictures are taken as information. In this calculation the numeric plaintext have numeric figure , character plaintext have character figure and if the information is alphanumeric sort then the yield figure message additionally alphanumeric, so there is no requirement for change the information field sort when the encoded information are put away in the database[6]

Lightweight cryptography is a relatively new field aimed to develop more efficient cryptographic implementations in response to typical constraints in the hardware used in Internet of Things (IoT). [5]

### 2.1 Motivation

All through organizations, Government division, specialist's and workplaces database frameworks are utilized. This specific framework stores and recovers touchy data, for example, government disability numbers, budgetary articulation, and profoundly grouped information. Associations with delicate information in their grasp should be secured utilizing diverse security strategies and arrangements. Keeping in mind the end goal to secure the information on a PC, they have to actualize strategies like get to control, evaluating, verification, encryption and so forth. However pernicious clients are as yet breaking into organizations information. Obviously, organizations are not executing poor security methods but rather programmers are simply getting more intelligent and more quick witted. This is the place IT workers need to discover new procedures or upgrade past ones. Hence, we actualizing the upgrade ETSFS calculation for touchy database security.

### 2.2 Objective

Important any Information means your organization data, these data an accessible asset to an organization, these data store in database system .Database is used to store the data for quick and secure processing of requests. Data is very important to some boundary and need high level of protection. It is a critical system as almost all the organizations are highly dependent on the data. Databases are designed to be shared among a number of users, where security is of prime concern and if it is compromised then they are vulnerable to malicious attacks that results in great loss. Hence, database system is maintained with many security mechanisms that include prevention of unauthorized access to data from insider or outsider of an organization. Therefore, an effective encryption algorithm should be necessarily applied to secure database.[1]

Database encryption would be one of the possible solutions where be one of the access to sensitive information is dependent on the key available, which promises a minimal damage and high performance when it is effective.

The tremendous development of technology and data storage leads organizations to depend on database systems. Organizations store huge amounts of data in secured databases in order to retrieve them in a fast and secure way. Some of the stored data is considered sensitive and has to be protected. [2]

Our major goal is to an attempt following issues related to:

1. Implement an Enhanced TSFS algorithm which will support special characters also.
2. Proposed Algorithm will be Light weight so that it will take less time to encrypt data.
3. Master key1, key2 are considered randomly used in ETSFS. Key can contain numbers.
4. Develop to a ETSFS Algorithm in PHP & WampServer version 2.5 to evaluate.
5. Analysis of security by Enhanced TSFS Algorithm.

## 3. PROPOSED WORK

### 3.1 Architecture Design

The Primary Purpose of Encryption is to ensure the certainly of advanced information put away on PC frameworks or transmitted through the Internet or other PC systems. Present day encryption calculations assume a fundamental part in the security confirmation of IT framework and correspondences as they can give classification, as well as the accompanying key components of security. Information frequently alluded to as plaintext is encoded utilizing an encryption calculation and an encryption key. This is procedure creates figure message that must be seen in the first frame if unscrambled with the right key. Unscrambling

is essentially the backwards of encryption, taking after similar strides however turning around the request in which the keys are connected.[9]


Figure 3.1 Generalized Procedure of Cryptography

In general the encryption process for securing data that other user can not access or read that information may be images, audio or plain text data by selecting the images, audio, or text file that contents are converted into the binary character or ASCII code

Main objective of this proposed work is to protection of information that is text, Special character or numbers.

Proposed system has two main phases.

A. Encryption Method
B. Decryption Method

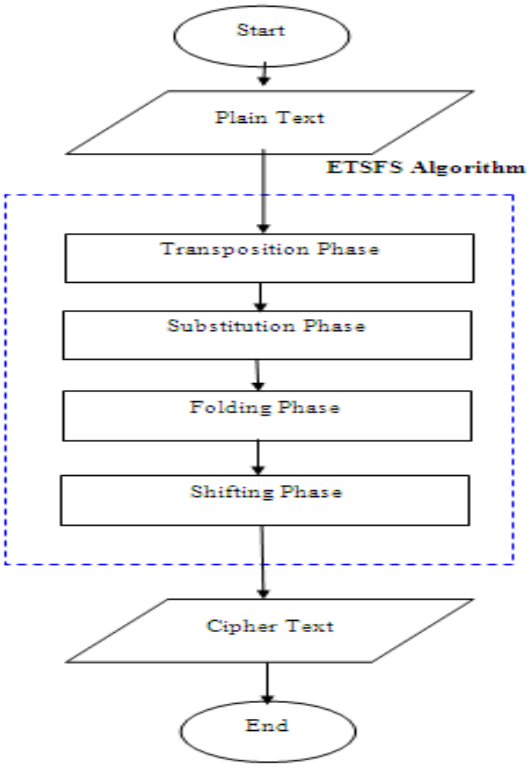


Fig. No. 3.2 Encryption Method

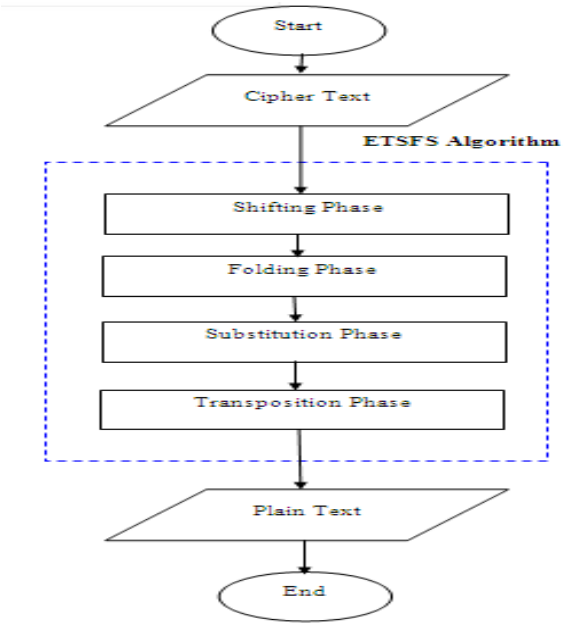


Fig. No. 3.3 Decryption Method

### 3.2 METHODOLOGY

The proposed algorithm involves the collection of new key values to consider randomly. The ETSFS algorithm which have four phases.

1) Transposition Phase
2) Substitution Phase
3) Folding Phase
4) Shifting Phase.

The explanation of ETSFS algorithm is as given below:

The ETSFS algorithm includes the following alphanumeric characters and a few of special symbols ( *, -, /, :, @ and _ ) only used to encrypt the data that is taken as input. But in the proposed methodology it included almost all the special characters. It is a symmetric encryption algorithm which can be inversed that cancels the encryption.

The constrained restricted on the number of characters is successfully imposed by accepting different data sizes dynamically depending on the user input length and if the length of data is less than the near square matrix size then the characters are replaced by *'s. Let say, the input string is 14 in length, then the nearby square matrix is 4x4 and the one character that is left is replaced by *'s. The four techniques of ETSFS are described as:

#### 3.2.1 Transposition Phase :

For a given input data matrix, the diagonal transposition is carried out starting from top leftmost corner till the end of

the matrix by moving laterally in one step followed by diagonal transition continually.
For example. Consider a string **"Infotech@17.com"** to be encrypted.

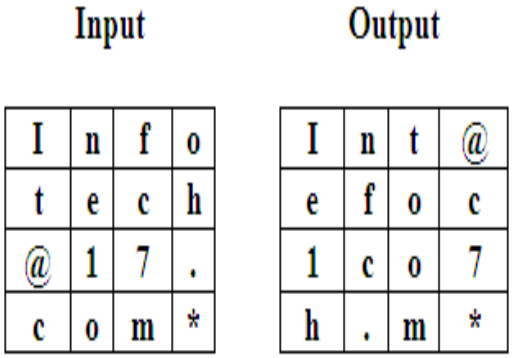


Fig. No: 3.4 Transposition Phase

As shown in the example figure, the output data is obtained by retaining the first element in the matrix.

The transposition in the given input is performed such that zigzag pattern (Int@efoc1co7h ....*) is followed by starting at the initial index of the matrix which is in the data matrix and continued by following a pattern till the end of matrix. The output data matrix will be the insertion of elements followed as it is in zigzag pattern. This is given as input to the substitution phase.

**2) Substitution Phase:**

The transposition is followed by the substitution method where each character is substituted by its encoded character. The encoding involves two keys Kl and K2 and the modulus size M. The Kl and K2 key values are generated using a random generator and the modulus size is dependent on the type of character. If the character is an alphabet letter then M :26. if the character is a digit then M=l0 or if the character is one of the special characters mentioned above. then M is considered to be 7. The encryption of the character is given by:

$$E(x)=(((K1+p) \bmod M + K2) \bmod M \quad ....(1)$$

From Fig.3.4 get the output & these output is input to the substitution phase shown below.
For. Example, In data matrix, the I character is a 0 index then value of I to be substituted of that matrix given eq. (1) & Fig. 3.6 Key1 & key2 are using & get result of encryption below.

0)I, P=8 , k1=1 ,k2=7 , M=26 then
E(x)=(((1+8)mod 26+7)mod 26)
E(x)=((9 mod 26)+7 ) mod 26)
E(x)=((9+7)mod 26)
E(x)=(16 mod 26)=16 =>Q

Then I character is encrypted to Q same process will be continue to 15 index.

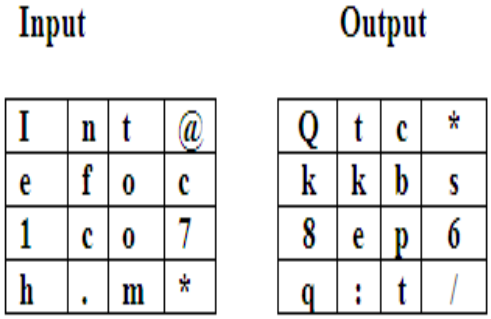


Fig. No: 3.5 Substitutions Phase

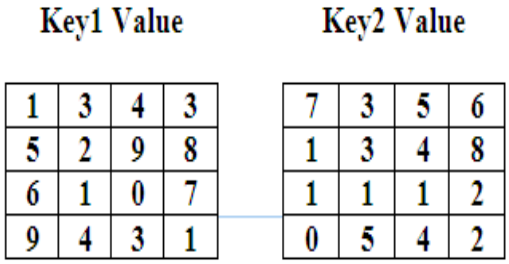


Fig. No: 3.6 Key1& Key2 Value

The encryption is performed on each element of the data matrix by using the function given in (1).
Here the keys Kl and K2 are used from Fig. 3.6 [2].The elements are replaced by its corresponding type of characters respectively.

A reverse operation can also be performed to decrypt the encoded data using the same keys and the decryption of encrypted data is:

$$D(E(x)) = (((E(x) — K2) \bmod M) — Kl) \bmod M \quad ....(2)$$

Reverse process of these above example for decryption of above I character & using eq. (2) & (3) ,E(X)=16 shown below.
D(E(x))=(((16)-7)mod 26-1)mod 26)
D(E(x))=(((9 mod 26)-1)mod 26)
D(E(x)=((9-1)mod 26)=8 mod 26=8 =>I

The above figure shows the example of encryption of data matrix with the given keys. And their decryption is shown as by applying D(E(x)) shown in (2). The output from E(x) is taken as input and processed by using the function in (2) which produces the input of substitution phase which enables that the encryption and decryption is 100 % accurate.[1]

Since most of the programming languages such as Java and C++ deal with the modulus as the remainder of an integer

division, some of the results may have minus sign, and this will create a problem because there is no data that have minus sign representation. So that one step has been included to the ETSFS algorithm implementation to check if the result includes the minus sign, and then apply:

$$D(E(x)) = M - |D(E(x))| \quad (3)$$

The following Fig. 3.5 shows the result of substitution. From the same example in Fig.3.4, if we implemented the decryption operation (2) on the first element, the result would be negative, so the ETSFS algorithm applies function (3) to get the correct result.

**3) Folding Phase :**

The third technique is folding in which the first row is exchanged with last row along with the exchange of first column with last column. And the center elements are exchanged diagonally. The below example illustrates the technique of transformation as shown in the below figure:

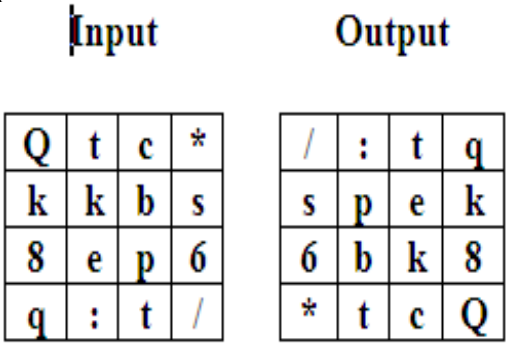


Fig. No: 3.7 Folding phase

**4) Shifting:**

The shitting transformation is the last phase of the methodology where the given array elements of digits exchange with their letter elements respective to the array elements. Alphabetical characters are referenced with an upper and lowercase array element of numbers ranging 0-25 with each number representing an alphabet respectively .Another array element is considered numeric characters ranging from 0-6 (7 characters) is also reflected .Each element in the data matrix is given reference with its location in the array in the array of elements (taken) and its appropriate positioned element is considered from its array elements.

Above example for searching an alphabet "t" which is index position 2 of data matrix then it should verify its position from array elements of alphabetical characters position of "t" is 19th and then start "t" as 'o' index & going toward backward index position '2' of data matrix & get alphabetical character "r" So "t" should be replaced with "r". similarly for the other categories.

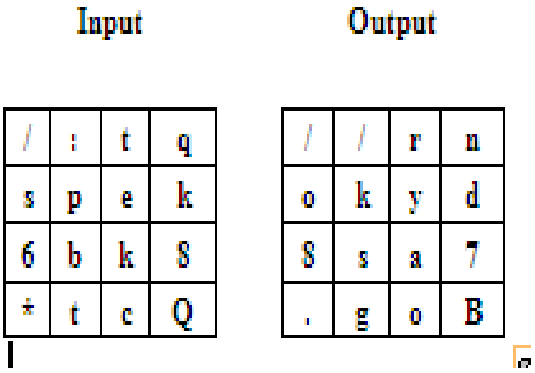


Fig. No: 3.8. Shifting Phase

**4. CONCLUSIONS**

In this paper we have Implemented ETSFS algorithm technique that prevents users from inferring sensitive information from database. By using ETSFS algorithm (Enhanced Transposition , Substitution, Folding, Shifting), we protected the sensitive information into the cipher text. But data will decrypt only the reverse process of (Shifting, Folding, substitution, Transposition) ETSFS algorithm and get the original text or messages. When the both semantic inference of encryption side and decryption side 100 % collaborate or matched with each other.

We concluded that the best way to this approach of securing sensitive data is by using encryption techniques and ensuring database security from attackers.

The Enhanced TSFS algorithm methodology is explained in which security is ensured in databases by simultaneously increasing the performance of encryption and decryption process.

**5. REFERENCES**

[1] Prathyusha Uduthalapally,Bing Zhou "Improvement of ETSFS Algorithm for secure Database" 4th international Symposium on Digital Forensics & Security (ISDFFS-16)25-27 April 2016 Little Rock,AR

[2] Hanan A, Abeer, Heba, "Lightweight Symmetric Encryption Algorithm for Secure Database." IJACSA International Journal of Advanced Computer Science and Applications, Saudi Arabia.

[3] D. Manivannan, R. Sttjarani, Light weight and secure database encryption using TSFS algorithm. Proceedings of the International Conference on Computing Communication and Networking Technologies, 2010, pp. l-7.

[4] L. Liu. J.Gai,A new lightweight database encryption scheme transparent to application, proceeding of the 6th IEEE International Conference on Industrial Informatics,2008,pp 135-140.

[5] Amandeep kaur1, Mrs. Shailja Kumari "Secure Database Encryption in Web Applications" IJARCCE ,vol. 3 issue 7, july 2014

[6] Susithra;M.Lawanya Shri;;K.S.Umadevi, "An Implementation of Database Image Encryption Using

TSFS Techniques",vol 3,Issuse7 July 2013 ISSN:2277 128X,ijarcsse.

[7] H. Alanazi, B. Zaidan, A. Zaidan, H. Jalab, M. Shabbir, Y. Al-Nabhani. New comparative study between , 3DES and AES within nine factors, Journal of Computing 2 (2010) 152-157.

[8] Pooja, Kanchan Narula "Enhancing Data Security in Cloud Computing with WebOS Using TSFS Algorithm "ISSN : 2319-7064.

[9] Vocal,"http://www.vocal.com/cryptography/dsadigital-signature-algorithm," Dated: l3- dec-2012 atl3:l8.