ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 7, July 2014

Secure Database Encryption in Web Applications

Amandeep kaur[1]**, Mrs. Shailja Kumari**[2]

Student of M.Tech , CSA, CDLU, Sirsa, India [1]

Assistant Professor, CSA, CDLU, Sirsa, India [2]

Abstract: In this paper we describe, the Secure databases has become increasingly crucial in all application areas. Database encryption is an important mechanism to secure databases from attacks and unauthorised access. The Transposition-Substitution-Folding-Shifting encryption algorithm (TSFS) is a symmetric database encryption algorithm that is a expansion technique to provide high security: it improves the efficiency of query execution time by encrypting the sensitive data only. However, it applies merely for the alphanumeric characters and numeric keys. This paper extends the data set of the TSFS encryption algorithm to take special characters as well, and corrects substitution and shifting processes by providing more than one modulo factor and four 16-arrays respectively in order to avoid the error that occurs in decryption steps. Experiment results show that enhanced TSFS encryption algorithm outperforms Data Encryption Standard algorithm (DES) and Advanced Encryption Standard algorithm (AES) in terms of query execution time and database added size. This algorithm will be support special characters and provide high security. If data will be secure than nobody can hack data.

Keywords: Security, Transposition, Substitution, Folding, Shifting

I. INTRODUCTION

Data security has consistently been a major issue in web applications. Database security has paramount importance in industrial, civilian and government domains. Organizations are storing big amount of data in database for data mining and other types of analysis. Some of this data is considered sensitive and has to be protected from disclosure. Challenges for security in database are increased due to the enormous popularity of e-business. In recent years, insider attacks gathered more attention than periodic outbreaks of malware. Database systems are usually deployed deep inside the company network and thus insiders has the easiest opportunity to attack and compromise them, and then steal the data. So data must be protected from inside attackers also. Many conventional database security systems are proposed for providing security for database, but still the sensitive data in database are vulnerable to attack because the data are stored in the form of plaintext only.

Fig.1 Database security

In the presence of security threats, database security is becoming one of the most urgent challenges because much damage to data can happen if it suffers from attacks and unauthorized access. With databases in complex, multi-tiered applications, attackers may reach the information inside the database. Damage and misuse of sensitive data that is stored in a database does not only affect a single user; but possibly an entire organization. We can categorize the attackers into three types:
intruder, insider, and administrator. Intruders are external people who infiltrate a database server to steal or tamper with data. Insiders are authorized users in a database system, who conduct some malicious works. Administrators can be database administrators (DBA) or system administrators (SA), and both have absolute rights to database systems. However, if they are malicious, the security of the database may be damaged. Insider and administrator attackers have gathered more attention in recent years because they can access a database without any effort, and they use important data in a wrong way. Database encryption has the potential to secure data at rest by providing data encryption, especially for sensitive data, avoiding the risks such as misuse of the data. In order to achieve a high level of security, the complexity of encryption algorithms should be increased with minimal damage to database efficiency, ensuring performance is not affected.

II. OBJECTIVES

The main objective of this paper is to enhance the TSFS algorithm which will support special characters to provide a high security to the databases at low time cost for encryption and decryption by encrypting sensitive data only.

i. To Design an secure database encryption algorithm which will support special characters also. that algorithm is TSFS algorithm.

ii. To Proposed Algorithm will be Light weight so that I will takle less time to encrypt data.

iii. To Develop a website(ASP.Net, Database SQL 2008) to implement secure database encryption using enhanced TSFS Algorithm.

iv. To Set Sensitive data fields as only Sensitive Data like passwords, Contact Numbers, Address will be Encrypted to save Time and increase Performance of Web Application.

v. To Analysis of security in TSFS algorithm and Enhanced TSFS Algorithm.

Our Objective is Sensitive data Security in web Applications. Database encryption is the only solution for avoid the risk posed by this threat.

1. Web Application Security:

Web application security is a branch of Information Security that deals specifically with security of websites, web applications and web services. At a high level, Web application security draws on the principles of application security but applies them specifically to Internet and Web systems. Typically web applications are developed using programming languages such as PHP, Java EE, Java, Python, Ruby, ASP.NET, C#, VB.NET

2. Need of Information Security:

Information is the most critical resource for many Websites. In many cases, the success of an Website depends on the availability of key information and, therefore, on the systems used to store and manage the data supporting that information. Due to the growth of networked data, security attacks have become a dominant problem in practically all information infrastructures.

Database security is the system, processes, and procedures that protect a database from unintended activity. Unintended activity can be categorized as authenticated misuse, malicious attacks or inadvertent mistakes made by authorized individuals or processes.

In the presence of security threats, database security is becoming one of the most urgent challenges because much damage to data can happen if it suffers from attacks and unauthorized access. With databases in complex, multi-tiered applications, attackers may reach the information inside the database. Damage and misuse of sensitive data that is stored in a database does not only affect a single user; but possibly an entire organization.

We can categorize the attackers into three types: intruder, insider, and administrator. Intruders are external people who infiltrate a database server to steal or tamper with data. Insiders are authorized users in a database system, who conduct some malicious works. Administrators can be database administrators (DBA) or system administrators (SA), and both have absolute rights to database systems. However, if they are malicious, the security of the database may be damaged. Insider and administrator attackers have gathered more attention in recent years because they can access a database without any effort, and they use important data in a wrong way. Database encryption has the potential to secure data at rest by providing data encryption, especially for sensitive data, avoiding the risks such as misuse of the data. In order to achieve a high level of security, the complexity of encryption algorithms should be increased with minimal damage to database efficiency, ensuring performance is not affected.

III.PROPOSED METHODOLOGY

The main objective of this paper is to enhance the TSFS algorithm and accordingly to provide a high security to the databases whilst limiting the added time cost for encryption and decryption by encrypting sensitive data only. The ETSFS algorithm can encrypt the data that consists of alphabetic characters from A to Z, all numbers and the following symbols: (*, -, ., /, :, @ and _). The ETSFS algorithm is a symmetric encryption algorithm, meaning each transformation or process must be invertible and have inverse operation that can cancel its effect. The key also must be used in inverse order.

ETSFS algorithm uses four techniques of transformations, which are transposition, substitution, folding and shifting. Fig. 1 presents the encryption algorithm, where the decryption algorithm reverses the encryption algorithm. The following sections describe the four techniques and contain the algorithms in pseudo-code format to be easy to understand.

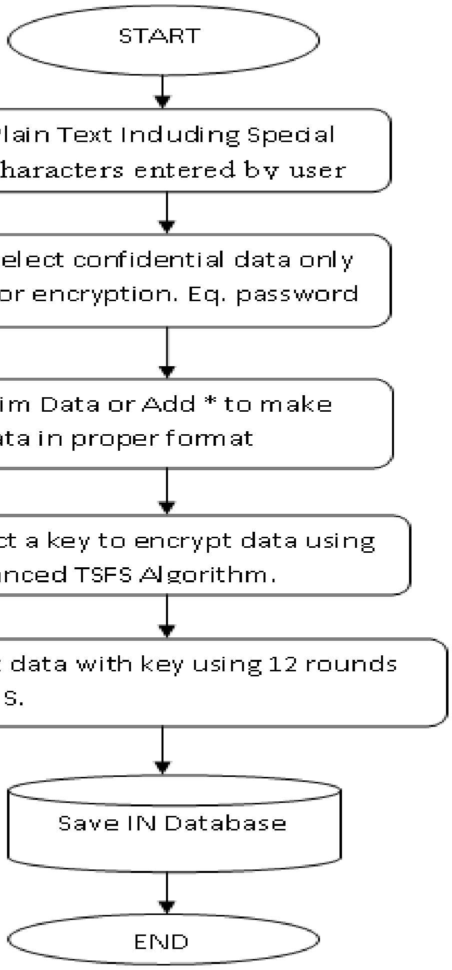
START
Plain Text Including Special characters entered by user
Select confidential data only for encryption. Eg. password
Trim Data or Add * to make data in proper format
Select a key to encrypt data using Enhanced TSFS Algorithm.
Encrypt data with key using 12 rounds of T S F S.
Save IN Database
END

Fig.2 Flow Chart

Encryption algorithm:

```
Algorithm encryption (String data, Array[12] keys )
Pre: data is plain text.
keys is array that contains 12 4x4-key matrices.
Post: encryptedData is data after encrypting.
Matrix[4,4] dataMatrix;
String encryptedData;
if (data length < 16)
padd data by adding *'s;
else if (data length > 16)
cut the data after 16;
end if
dataMatrix = data;
key = expandKeys (keys);
for (int i=0; i<12; i++)
dataMatrix = transposition (dataMatrix);
dataMatrix = substitution (dataMatrix, keys(i),
keys((i+1)mod 12));
dataMatrix = folding (dataMatrix);
dataMatrix = shifting (dataMatrix);
end for
encryptedData = dataMatrix;
return encryptedData
End encryption
```

The decryption algorithm reverses the encryption algorithm.

CONCLUSION

Data-storing and exchanging between computers is growing fast across the world. The security of this data has become an important issue for the world. The best solution centred on securing the data is using cryptography, along with other methods. This paper proposes the enhancement of the TSFS algorithm to support the encryption of special characters, correct substitution process by providing more than one modulo factor to differentiate between data types and prevent increasing the data size, as well as correcting the shifting process for the same reasons by providing four 16-arrays. The experimental results have shown that the ETSFS algorithm successfully encrypted important symbols, as well as alphanumeric data. The improved performance comes without compromising query processing time or database size. Using well-established encryption algorithms as benchmarks, such as DES and AES, the proposed ETSFS algorithm was shown to have consumed the smallest space and encryption time compared to the other algorithms.

REFERENCES

[1] Rakesh Agrawal , Jerry Kiernan , Ramakrishnan Srikant , Yirong Xu,2002, Hippocratic databases, Proceedings of the 28th international conference on Very Large Data Bases.

[2] L. Liu, J. Gai,2008, A new lightweight database encryption scheme transparent to applications, Proceedings of the 6th IEEE International Conference on Industrial Informatics.

[3] K. Kaur, K. Dhindsa, G. Singh, 2009, Numeric to numeric encryption: using 3KDEC algorithm, Proceedings of IEEE International Conference on Advance Computing. [4] D. Manivannan, R .Sujarani, 2010, Light weight and secure database encryption using TSFS algorithm, Proceedings of the International Conference on Computing Communication and Networking Technologies.

[5] Hanan A. Al-Souly, Abeer S. Al-Sheddi, Heba A. Kurdi., 2013 Lightweight Symmetric Encryption Algorithm for Secure Database, (IJACSA) International Journal of Advanced Computer Science and Applications. [6] S. Bhatnagar, Securing Data-At-Rest, Literature by Tata Consultancy Services.

[7] I. Widiasari, Combining advanced encryption standard (AES) and one time pad (OPT) encryption for data security, The International Journal of Computer Applications 57 (2012) 1-8.

[8] Z.Yong-Xia, The Technology of Database Encryption, Proceedings of the 2nd International Conference on MultiMedia and Information Technology, 2010, pp. 268-270.

[9] Mattsson, U.T (2005), “A practical implementation of ransparent encryption and separation of duties in enterprise databases: protection against external and internal attacks on databases”, IEEE, Publication Year: 2005 , Page(s): 559 - 565

[10] Samba Sesay et. al (2005), “A secure database encryption scheme”, Consumer Communications and Networking Conference, 2005. CCNC. 2005 Second IEEE. Page(s): 49 – 53.

[11] Gang Chen, “A Database Encryption Scheme for Enhanced Security and Easy Sharing Computer Supported Cooperative Work in Design”, 2006. CSCWD ‘06. 10th International Conference on Publication Year: 2006, Page(s): 1 - 6

BIOGRAPHIES

Amandeep kaur pursuing her M.Tech in Computer Science from Chaudhary DeviLal University, Sirsa (Hry). She is presently the Student of Computer Science Branch. Her Research interest in Security of database.

Mrs. Shailja kumari is Assistant Professor of Computer Science & Application Department in Chaudhary Devi Lal Vniversity Sirsa (Hry).