International Journal of Electronics Communication and Computer Engineering
Volume 6, Issue 5, ISSN (Online): 2249–071X, ISSN (Print): 2278–4209

A Survey on Some of the Symmetric Key Encryption Algorithms used for Database Security

M. Pushpa, M. Sujitha

***Abstract* – Data's are the most valuable assets in today's world. They are stored in Database's which helps organizations as well as individuals to extract information as well as to make various decisions. Even though there are various security measures provided to the Databases, there are various threats for them. To protect database from the various threats, it is essential to protect the data present in the database through Database Security approaches. Database Security is the important challenge to secure Database from unauthorized persons. This paper discusses about one of the highly secured Database security at the database level algorithm known as Encryption.**

***Keyword* – Encryption, Symmetric Key, DES, AES, REA.**

I. INTRODUCTION

Information or data[1] is a valuable asset in any organization. Almost all organization whether social, governmental, educational etc., have now automated their information systems and other operational functions. Organizations that are running successfully demand the confidentiality of their database as they have maintained the databases that contain the crucial information. They do not allow the unauthorized access to their data/information and, they also demand the assurance that their data is protected against any malicious or accidental modification. Data protection and confidentiality are the security concerns. Database security demands permitting or prohibiting user actions on the database and the objects inside it. Protecting the confidential/sensitive data stored in such a repository is actually known as the database security. So database security is a serious concern. It deals with making database secure from any form of illegal access or threat at any level. Database security demands permitting or prohibiting user actions on the database and the objects inside it.

II. DATABASE ENCRYPTION

Database security encompasses three main properties: confidentiality, integrity, and availability. The confidentiality property enforces predefined restrictions while accessing the protected data, thus preventing disclosure to unauthorized persons. The integrity property guarantees that the data cannot be corrupted in an invisible way. Finally, the availability property ensures timely and reliable access to the database [3].

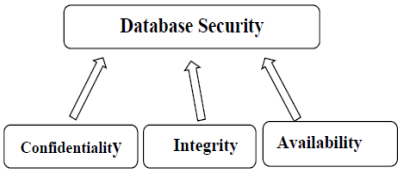

Fig.1. Properties of DB Security

Database encryption refers to the use of encryption techniques to transform a plain text into an encrypted text, thus making it unreadable to anyone except those who possess the knowledge of the encryption key(s). Encryption is a key control that receives a lot of attention in organizations today. Encryption can help prevent data loss or theft, as well as prevent fraud within an organization. There have been many situations over the years in which backup tapes have gone missing, either lost or stolen, and their sensitive data was not encrypted. Organizations know they need to encrypt sensitive and regulated data [2]. Encryption can provide strong security for data at rest, but developing a database encryption strategy must take many factors into consideration [3]. Some of the factors to be considered are the level of the database, on which data and performance issues on database encryption etc.

III. ENCRYPTION LEVEL

A. Database-level encryption

Database-level encryption allows securing the data as it is inserted to, or retrieved from the database. The encryption strategy can thus be part of the database design and can be related with data sensitivity and/or user privileges. Selective encryption is possible and can be done at various granularities, such as tables, columns, rows[5].

B. Application-level encryption

Application-level encryption moves the encryption/ decryption process to the applications that generate the data. Encryption is thus performed within the application that introduces the data into the system, the data is sent encrypted, thus naturally stored and retrieved encrypted, to be finally decrypted within the application [5].

C. Storage-level encryption

Storage-level encryption enables enterprises to encrypt data at the storage subsystem, either at the file level or at the block level. This type of encryption is well suited for encrypting files, directories, storage blocks, and tape media [6].

IV. ENCRYPTION TECHNIQUES

Depending on the type of security keys used to encrypt/decrypt the secured data, encryption procedures are mainly categorized into two categories namely Asymmetric and Symmetric encryption techniques

- ***Symmetric key algorithm***- uses the same key to both encrypt and decrypt the database
- ***Asymmetric key algorithm*** - (Public key encryption)- uses two different keys instead of a single key — one key

to encrypt the database and another to decrypt the database.

Each type of algorithm has its own advantages and limitation. The advantage of symmetric algorithms is that they tend to be faster than asymmetric algorithms. However, the disadvantage is that key management can be more difficult. Because the same key is used to encrypt and decrypt the data, anyone who has the key for encryption can use the same key to decrypt any of the data that has been encrypted.

V. Symmetric Key Encryption Algorithms

Data Encryption algorithm is one of the way to make sure of the privacy, reliability, validation, accessibility and documentation of the user's data. They also helps to maintain the user data as well as to provide security and privacy against the threats. This paper analyzes three different algorithms namely DES, AES and REA with their advantages and disadvantages and which is best for database security.

A. Data Encryption Standard

Data Encryption Standard or DES is Symmetric encryption algorithm developed by Ron IBM cryptography researchers Horst Feistel in the early 70's. It is the block cipher which takes a fixed-length string of plaintext bits and transforms it through a series of complicated operations into another cipher text bit string of the same length [7]. As it is a symmetric key algorithm both sender and receiver use a shared key to encrypt and/or decrypt the data.

In this algorithm, the block size is 64 bits it also uses a key to customize the transformation, so that decryption can supposedly only be performed by those who know the particular key used to encrypt. The key basically consists of 64 bits however, only 56-bits of these are actually used by the algorithm. Eight bits are used solely for checking parity, and are thereafter discarded. Hence the effective key length is 56-bits, and it is always quoted as such. Every 8th bit of the selected key is discarded i.e., positions 8, 16, 24, 32,40, 48, 56, 64 are removed from the 64-bit key leaving behind only the 56-bit key[3].

DES is based on two fundamental attributes of cryptography Diffusion (Substitution) and Confusion (Permutation) consisting of 16 rounds. In each round key and data bits are shifted, permuted, XORed and sent through, 8 s-box. In the first round 64 bit plaintext is handed to initial permutation (IP). Then IP generates two halves left plaintext(LPT)and right plaintext(RPT).Each LPT and RPT goes through 16 rounds. At the last LPT and RPT are rejoined. Decryption is same process perform rounds in reverse order.

Algorithm

1. DES proceeds by taking an input of 64-bit long plaintext and 56-bit key (8 bits of parity) and produce the output of 64 bit block.
2. The plaintext block shifts the bits around.
3. The 8 parity bits are removed from the key by subjecting the key to its Key Permutation.
4. The plaintext and key are processed in 16 rounds consisting of:

a. The key is split into two 28 bit halves
b. Each half of the key is shifted (rotated) by one or two bits, depending on the round.
c. The halves are recombined and subject to a compression permutation to reduce the key from 56 bits to 48 bits. This compressed key is used to encrypt this round's plaintext block.
d. The rotated key halves from step 2 are used in next round.
e. The data block is split into two 32-bit halves.
f. One half is subject to an Expansion Permutation to increase its size to 48 bits.
g. Output of step 6 is exclusive-OR'ed with the 48-it compressed key from step 3.
h. Output of step 7 is fed into an S-box, which substitutes key bits and reduces the 48-bit block back down to 32-bits.
i. Output of step 8 is subject to a P-box to permute the bits.
j. The output from the P-box is exclusive- OR'ed with other half of the data block.
k. The two data halves are swapped and become the next round's input.

DES is a reversible process, i.e., same algorithm which is used for encryption in DES also works for decryption, the only difference between the encryption and decryption process is the reversal of key portions. If the original key K was divided into K1, K2, K3……K16 for the 16 encryption rounds then for decryption the key should be used as K16, K15, K14……K1.

B. Advanced Encryption Standard

The Advanced Encryption Standard or AES was published by the National Institute of Standards and Technology (NIST) in 2001. AES is a symmetric encryption algorithm used by the government to protect classified information and is implemented in software and hardware throughout the world to encrypt sensitive data [9]. The AES Encryption algorithm would be unclassified and had to be capable of protecting sensitive information.

Advanced Encryption Standard algorithm works on the principle of Substitution Permutation network. AES doesn't use a Feistel network and is fast in both software and hardware. AES operates on a 4×4 matrix of bytes termed as a state. The AES cipher is specified as a number of repetitions of transformation sounds that convert the input plaintext into the final output of cipher text. Each round consists of several processing steps, including one that depends on the Encryption key.

The key length and length of plain text blocks need to be selected independently. AES mandates that the plain text block size must be 182 bits and key size should be 128,192 or 256 bits. In general two versions of AES are used: 128 bit plain text block combined with 128 bit key block and 128 bit plain text block with 256 bit key block. The first one becomes a commercial standard. AES-128 bit algorithm uses the basic techniques of substitution and transposition. The key size and the plain text block decide how many rounds need to be executed. The minimum number of rounds is 10 and maximum number of round is 14.

Algorithm: (for 128 bits)

1. Derive the set of round keys form the ciper key(The cipher takes a plaintext bolock size of 128 bits, or 16 bytes);
2. Initialize the state array with the block data(plain text).(The state block is depicted as a 4*4 square matrix of bytes.
3. Add the initial round key to the starting state array.
4. The number of rounds is 10, for the case when the encryption key is 128 bit long.(the number of rounds is 12 when the key is 192 bits and 14 when the key is 256).
5. Perform nine rounds of state manipulation.
6. Perform the tenth and final round of state manipulation.
7. Copy the final state array output as the encrypted data (cipher text).
8. Each round of the encryption process requires a series of steps to alter the state of array. These steps involve four steps of operations. They are

a) Sub Bytes: This operation is a simple substitution that converts every bits into a different value.

b) Shift Rows: Each row is rotate d to the right by a certain number of bytes.

c) Mix Columns: Each column of the state array is processed separately to produce a new column. The new column replaces the old one.

d) Add Round key: This operation simply takes the existing state array.

9. Decryption involves reversing all the steps in encryption using inverse functions like InvSubBytes, InvShiftRows, InvMixColumns.

A set of reverse rounds are applied to transform cipher text back into the original plaintext using the same encryption key.

C. Reverse Encryption Algorithm

Reverse Encryption algorithm(REA) is a symmetric encryption algorithm that is used to protect sensitive data in database. REA is simple, secure and efficient, and takes a variable-length key, making it flawless for data security[6].

The REA algorithm encipherment and decipherment consists of the same operations, only the two operations are different: 1) added the keys to the text in the encipherment and removed the keys from the text in the decipherment. 2) Executed divide operation on the text by 4 in the encipherment and executed multiple operation on the text by 4 in the decipherment. We execute divide operation by 4 on the text to narrow the range domain of the ASCII code table at converting the text.

The keys are concatenated to the text in the encryption process and removed from the text in the decryption process. Mathematical Divide operation is performed on the text data by 4 in the encryption process and multiple operations on the text by 4 has been done in the decryption process. Divide by 4operation is performed on the text to narrow the range domain of the ASCII code. [8].

The cost time of the encryption and decryption operations can be reduced and the performance is also improved by REA [9].

REA Encryption Algorithm:

1. Add the key before the data to be encrypted.
2. Replace the data to ASCII code and change that ASCII to binary data.
3. Reverse the binary data and convert 8 bits binary data in the form of ASCII code
4. Divide the converted ASCII code by 4 from Divide operation put the Quotient as the 1st character and Remainder as the 2nd character
5. Return encrypted data.

Table 1 gives a comparative study between AES, DES, REA is presented in to fourteen factors [7][10],[11],

Table 1: Comparison between some Symmetric Key Encryption Algorithms

Factors	**AES**	**DES**	**REA**
Developed	2000	1977	2012
Abbreviation	Advanced Encryption Standard	Data Encryption Standard	Reveres Encryption Algorithm
Algorithm	Symmetric Algorithm	Symmetric Algorithm	Symmetric Algorithm
Security	Secured	Not Secure Enough	More Secured
Key used	Same Key used for both Encryption and Decryption	Same Key used for both Encryption and Decryption	Same Key used for both Encryption and Decryption
Rounds	10/12/14	16	1
Encryption and Decryption algorithm	Different	Different	Same but two Processes are different
Key size	128,192 or 256 bits	56 bit	Variable size
Block size	128,192,224,256-bit	64 bit	Variable size
Scalability	Not Scalable	It is Scalable algorithm due to varying the key size and Block size	It is Scalable algorithm due to varying the key size and Block size
Encryption/ Decryption Speed	Faster	Moderate	More Faster
Hardware & Software Implementation	Faster	Better in hardware than in Software	
Speed depends on keys?	‘Yes	Yes	Yes
Power Consumption	High	Low	Low

VI. CONCLUSION

As data stored in the database is the backbone of the many organization. It is important to secure them. Thus, this paper discusses about a brief description about some of the asymmetric key Encryption algorithm to protect them. This lead to more concrete solution for database security issues and to improve the techniques used for database security.

REFERENCES

[1] IqraBasharat, FarooqueAzam, Abdul WahabMuzaffar(june 2012),” Database Security and Encryption: A Survey Study”, International Journal of Computer Applications(0975-888) volume 47-No.1, pp-28-34.

[2] Tanya Baccam, "Transparent Data Encryption: New Technologies and Best Practices for Database Encryption",SANS Analyst Program, (A SANS Whitepaper-April 2010). http://www.sans.org/reading-room/whitepapers/analyst/transparent-data-encryption-technologies-practices-database-encryption-34915

[3] Luc Bouganim,LucBouganim,"Database Encryption", http://www.smis.inria.fr/~bouganim/Publis/BOUGA_B6_ENC_CRYPT_2009.pdf

[4] SayedTathir Abbas, RavinderaKumar(Aug 2013),” Analytical Study of AES and Proposed Variant with Enhance Block Length and Key length”, International Journal of Science and Research (IJSR), India Online ISSN: 2319-7064 Volume:2 Issue 8, pp.176-178.

[5] Hacigümüs H., Iyer B., Li C., Mehrotra S., Providing Database as a Service, International Conference on Data Engineering (ICDE), 2002, pp. 29-39.

[6] Ulf T. Mattsson, "Database Encryption - How to Balance Security with Performance", http://hosteddocs.ittoolbox.com/UM070805.pdf.

[7] MohitMarwahha, Rajeev bedi, AmritpaulsinghTejindersingh (July-Sept 2013),”Comparative Analysis of Cryptographic algorithms”, International journal of advanced engineering technology, E-ISSN 0976-3945, pp.16-18.

[8] AymanMousa, Osama S.Faragallah, S.EL-Rabaie, E.M.Nigm(March 2013),"Security Analysis of Reverse Encryption Alogrithm for Databases", International Journal of Computer Applications(0975-8887),volume 66-No.14,pp.19-26.

[9] RobiniA.Chirde, S.S. Kulkarni(January 2014),”Assessing Performance of Encrypted Databases under query processing with the REA Algorithm”, International Journal of Advance Research in Computer Science and Management Studies, ISSN:2321-7782,Volume 2,Issue 1,pp.424-430.

[10] Prerna Mahajan &AbhishekSachdeva(2013), “A Study of Encryption Algorithms AES, DES and RSA for Security”, Global Journal of Computer Science and Technology Network, Web & Security, Volume 13 Issue 15 Version 1.0, Online ISSN: 0975-4172 & Print ISSN: 0975-4350, pp.15-22.

[11] Vekariya Meghna (Aug 2014), ”Comparative Analysis of Cryptographic Algorithms and Advanced Cryptographic Algorithms”, International Journal of Computer Engineering and Science, pp.1-8.

AUTHOR’S PROFILE

M. Pushpa
received her Post Graduation in Computer Applications from the University of Madras, Chennai and M.Phil. Degree from Mother Theresa’s Women’s University, Kodaikannal, Tamilnadu, and she has more than fifteen years of Teaching experience with both Undergraduate and Postgraduate Degrees Courses in Computer Science and Computer Application, with University of Madras. She had presented more than fifteen papers in national and international conferences and also published around ten papers in National and International level Journal. Her area of interest is Data Mining and Artificial Intelligence.

M. Sujitha
received her MSc. Information Technology in 2014 from University of Madras. She is pursuing her MPhil Computer Science under the supervision of Ms. M.Pushpa in Quaid-E-Millath Government College for Women, Affiliated to University of Madras. She has presented papers in national and international conferences and published a paper in national level Journal. Her area of interest is Database Management System.