УДК 004.75 doi: 10.26906/SUNZ.2020.3.104

Y. M. Shcherbinina, B. V. Martseniuk. A. M. Filonenko
National Technical University "Kharkiv Polytechnic Institute", Kharkiv, Ukraine

DATABASE SECURITY AND STUDY OF DATA ENCRYPTION METHODS IN CLOUD STORAGE.

Abstract. Security of Data is the most important task in today’s world. Governments, companies and other organizations have lost a lot of money and many others have closed down due to the activities of dubious hackers and attackers. Over the years various encryption schemes have been developed in order to protect the database from various attacks by the intruders. As data is the life wire of every organization, there is the need to remotely and securely store the data generated daily by these organizations in order to enable them recover quickly in the event of attach and hack. Cloud storage is needed here for the remote data storage.

For many establishments, data security is one of their major concern when sending their files into the cloud. They worry about their files being seen or even compromised by malicious and dubious people because that’s what happened in the past.

Data encryption techniques are required to protect the integrity of the stored data. In the past, many businesses felt comfortable allowing the cloud providers to manage all their data, believing that security risks could be managed through contracts, controls and audits. Over time it has become apparent, however, that cloud providers cannot honor such commitments when responding to government requests for information.

This paper discuses, the importance of database encryption and makes an in depth review of various encryption techniques.

Keywords: Encryption, Cryptography, Hashing, Security, Database, Cloud Storage, Cipher text.

Introduction

In this age of technology, all our work is being done by the computers. From chatting with friends on social networking websites, to making online payments through Net Banking, everything is being done online through computers. Since these facilities are efficient and make our work easy we use them in one way or the other. This means to use these online services we are storing all our personal and sensitive data in the databases of these websites and applications, which indeed make this data prone to various security threats. So protection of this important user data is one of the major priority, in order to avoid any misuse of data [1].

Cloud Storage is a system whereby data is remotely stored, maintained, managed, and backed up. The service is available to users over a network, which is usually the internet.

Important way of protecting this data is by encrypting the data being saved in the databases of these websites.

What is the need of encrypting the data

The need of encrypting the data before saving it in a database is that by restricting the access through authorization and authentication of data can help to a certain limit, but what if the intruder somehow gets to the database. He has all the data of database and can misuse it as he likes, here encryption of data before saving it in database comes into play. If the data is encrypted before saving it in the database, even with access to the database the intruder cannot misuse this data.

So the **purpose** of the article is, discusses the importance of database encryption and makes an in depth review of various encryption techniques.

Database encryption

Database Encryption is a process of encrypting the data in the database [2]. It is a key strategy to protect the contents of data within the database. The main idea behind this is that incase the intruder somehow is able to get to the database of the system; due to encryption he should not be able to misuse the data in the database.

Figure 1, shows basic working of the database encryption and decryption process. The plain text/data to be saved in the database is first converted into cipher text using an appropriate algorithm and a specific key. Then this cipher text is saved into the database. When the user wants to extract the data from the database, the cipher text is converted back to plain text using the decryption algorithm and the same key used in encryption. This will return the plain text to the user, when requested.

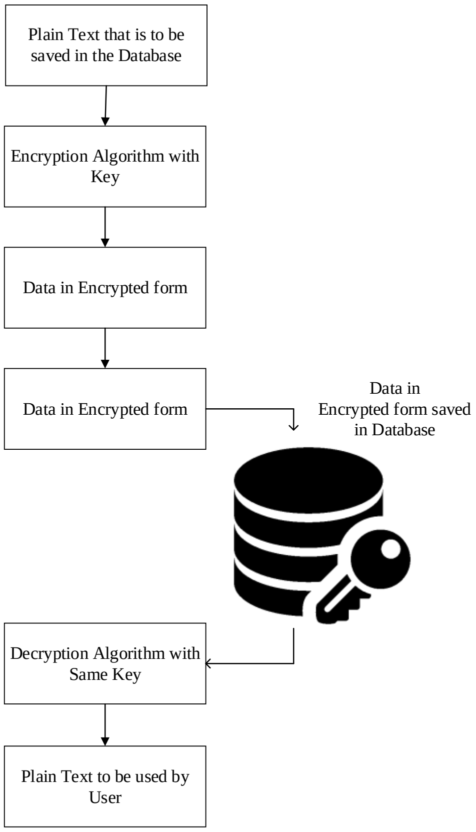
Plain Text that is to be saved in the Database
Encryption Algorithm with Key
Data in Encrypted form
Data in Encrypted form
Data in Encrypted form saved in Database
Decryption Algorithm with Same Key
Plain Text to be used by User

Figure 1. Database Encryption and Decryption Process

Database Encryption can be done in two possible ways:

Encryption: It is a process in which plain text is converted to cipher text with help of key, and then using the same key we can decrypt the cipher text back to plain text [3]. Encryption is performed using various algorithms, with each algorithm having his own advantages and disadvantages. Most commonly used encryption algorithms are DES, RC2, AES128, AES 256 etc. Figure 2, shows working of simple encryption process.

Hashing: It is a one way process, in which plain text is convened into hashed value(encrypted form). Once the data is hashed using a Hash Function it cannot be changed back to plain Text [3]. Generally this approach is used for password encryption, whenever we need to login the password entered is encrypted using hash function and then matched with the password stored in the database which is already in encrypted form, if both matches the user get access else it gets the message of invalid username/password. Most commonly used Hash Functions are MD4, MD5, SHA, SHA-1 etc. Figure 3, shows working of hashing.

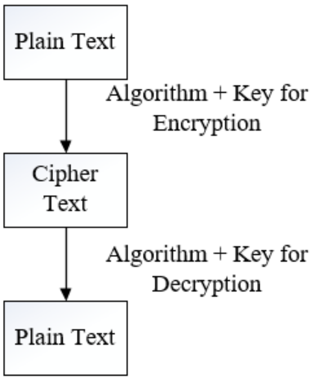
Plain Text
Algorithm + Key for Encryption
Cipher Text
Algorithm + Key for Decryption
Plain Text

Figure 2. Working of Encryption Process

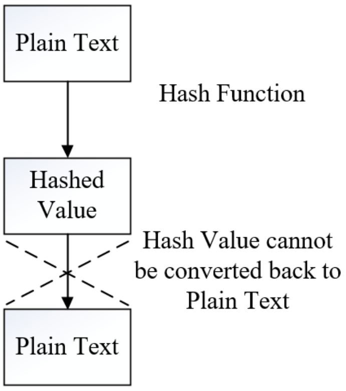

Figure 3. Working of Hashing Process

Cloud data encryption

Cloud encryption is a service offered by cloud storage providers whereby data, or text, is transformed using encryption algorithms and is then placed on a storage cloud. It is the transformation of a cloud service customer's data into cipher text. Cloud encryption is almost identical to in-house encryption with one important difference -- the cloud customer must take time to learn about the provider's policies and procedures for encryption and encryption key management. The cloud encryption capabilities of the service provider need to match the level of sensitivity of the data being hosted. Because encryption consumes more processor overhead, many cloud providers will only offer basic encryption on a few database fields, such as passwords and account numbers [4]. At this point in time, having the provider encrypt a customer's entire database can become so expensive that it may make more sense to store the data in-house or encrypt the data before sending it to the cloud. To keep costs low, some cloud providers have been offering alternatives to encryption that don’t require as much processing power. These techniques include redacting or obfuscating data that needs to remain confidential or the use of proprietary encryption algorithms created by the vendor [5].

Cloud data encryption methods

- Block Ciphers and Stream Ciphers

One of the main categorization methods for encryption techniques commonly used is based on the form of the input data they operate on. The two types are Block Cipher and Stream Cipher. This section discusses the main features in the two types, operation mode, and compares between them in terms of security and performance.

Block Cipher

In this method ciphering, data is encrypted and decrypted if data is in from of blocks. In its simplest mode, you divide the plain text into blocks which arc then fed into the cipher system to produce blocks of cipher text. ECB(Electronic Codebook Mode) is the basic form of block cipher where data blocks arc encrypted directly to generate its correspondent ciphered blocks.

Stream Ciphers

Stream cipher functions on a stream of data by operating on it bit by bit. Stream cipher consists of two major components: a key stream generator, and a mixing function. Mixing function is usually just an XOR function, while key stream generator is the main unit in stream cipher encryption technique. For example, if the key stream generator produces a series of zeros, the outputted ciphered stream will be identical to the original plain text.

- Symmetric and Asymmetric encryptions

Data encryption procedures are mainly categorized into two categories depending on the type of security keys used to encrypt/decrypt the secured data. These two categories are: Asymmetric and Symmetric encryption techniques

Symmetric Encryption

In this type of encryption, the sender and the receiver agree on a secret (shared) key. Then they use this secret key to encrypt and decrypt their sent messages. Fig. 4 shows the process of symmetric cryptography. Node A and B first agree on the encryption technique to be used in encryption and decryption of communicated data. Then they agree on the secret key that both of them will use in this connection. After the encryption setup finishes, node A starts sending its data encrypted with the shared key, on the other side node B uses the same key to decrypt the encrypted messages.

The main concern behind symmetric encryption is how to share the secret key securely between the two peers. If the key gets known for any reason, the whole system collapses. The key management for this type of encryption is troublesome, especially if a unique secret key is used for each peer-to-peer connection, then the total number of secret keys to be saved and managed for n-nodes will be n(n-1)/2.

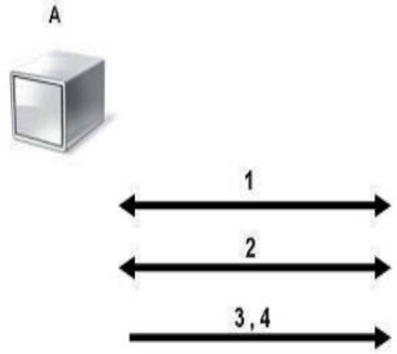

Figure 4. Symmetric Encryption

Asymmetric Encryption

It is also known as Public Key Cryptography (PKC), because users tend to use two keys: public key, which is known to the public, and private key which is known only to the user. Fig. 5 below illustrates the use of the two keys between node A and node B. After agreeing on the type of encryption to be used in the connection, node B sends its public key to node A. Node A uses the received public key to encrypt its messages. Then when the encrypted messages arrive, node B uses its private key to decrypt them.

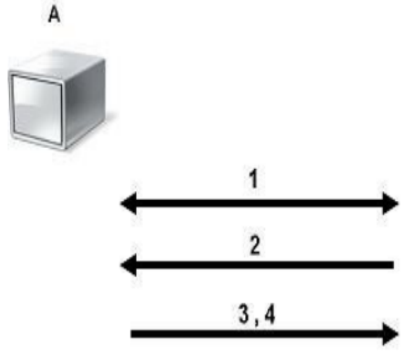

Figure 5. Asymmetric Encryption

This capability surmounts the symmetric encryption problem of managing secret keys. But on the other hand, this unique feature of public key encryption makes it mathematically more prone to attacks. Moreover, asymmetric encryption techniques are almost 1000 times slower than symmetric techniques, because they require more computational processing power. To get the benefits of both methods, a hybrid technique is usually used. In this technique, asymmetric encryption is used to exchange the secret key, symmetric encryption is then used to transfer data between sender and receiver [6].

For example, in an open system, given any two principals X and Y, X should be able to encrypt a message that can only be decrypted by Y. If there is some binding established between principal identities and public keys, then these operations can easily be performed. A naive scheme might function as follows:

1. principal X looks up public key K_y for principal Y and uses it to compute an encryption for Y using some trapdoor function $c = f_{KY}(m)$;

2. then Y, on receipt of this message computes. $f^1 k_y(c) = m$.

But there's a significant problem with this scheme given our definitions of security for shared-key encryption: it doesn't satisfy Semantic Security, since it's trivial for an adversary to compute $fK_y(m)$ and $fK_y(m')$ and compare them against given ciphertexts in the different attack models. Once again we see that there is no Semantic Security without probabilistic encryption. This is especially true in the public-key setting, since every principal has access to an encryption function for every other principal, by definition. Especially when the space of possible messages is small, it is easy to simply check all messages under the encryption function to figure out what has been encrypted.

Conclusions

With advancement in Technology, nowadays everything is being done with computers, so security of these data in the database becomes an important issue. Many researchers have worked on this thing and proposed various algorithms and architectures. Each scheme has its own advantages and disadvantages. But none of them is fully secure, and contain certain loopholes or demerits with can be used by the attackers and the intruders to get access of the database. So there is a scope of improvement in this area.

Many research problems are yet to be identified. Cryptographic techniques are used to provide secure communication between the user and the cloud. Symmetric encryption has the speed and computational efficiency to handle encryption of large volumes of data in cloud storage. This paper proposed a symmetric encryption algorithm for secure storage of cloud user data in cloud storage. The proposed encryption algorithm is described in detail and the decryption process is reverse of the encryption. This algorithm is used in order to encrypt the data of the user in the cloud. Since the user has no control over the data once their session is logged out, the encryption key acts as the primary authentication for the user.

REFERENCES

1. Baraani-Dastjerdi, Ahmad, Josef Pieprzyk, and Reihaneh Safavi-Naini. "Security in databases: A survey study." Department of Computer Science, The University of Wollongong (1996).
2. Denny Cherry and Thomas Larock, "2 - Database Encryption, In Securing SQL Server", edited by Denny Cherry, Thomas Larock, Syngress, Boston, 2011, Pages 27-71, ISBN : 9781597496254.
3. Kessler, Gary C. "An overview of Cryptography." (2003). (http://www.sciencedirect.com/science/article/pii/B97815974962541000 22).
4. Vamsee Krishna, Yarlagadda And Sriram Ramanujam, ―Data Security in Cloud Computing‖, Journal of Computer and Mathematical Sciences, Vol.2 (1), pp 15-23, 2011.

5. Peter Mell, Tim Grance, ―Effectively and Securely Using the Cloud Computing Paradigm‖, NIST, Information Technology Laboratory, http://www.csrc.nist.gov/groups/SNS/cloud-comput ing/cloudcomputingv26.ppt. 2009.
6. Eman M.Mohamed, Hatem S.Abdelkader and Sherif El-Etriby, ―Data Security Model for Cloud Computing‖, The Twelfth International Conference on Networks, ISBN: 978-1-61208-245-5, pp 66-74, 2013.

БЕЗПЕКА БАЗИ ДАНИХ І ВИВЧЕННЯ МЕТОДІВ ШИФРУВАННЯ ДАНИХ В ХМАРНОМУ СХОВИЩІ.

Є.М. Щербініна, Б.В. Марценюк, А.М. Філоненко

Анотація. Безпека даних - найважливіше завдання в сучасному світі. Влада, компанії та інші організації втратили багато грошей, а багато інших закрилися, через діяльність хакерів і зловмисників. За минулі роки були розроблені різні схеми шифрування для захисту бази даних від атак зловмисників. Оскільки дані - це життєвий канал кожної організації, існує потреба в безпечному зберіганні даних, щодня генерованих цими організаціями. Хмарне сховище необхідно для віддаленого зберігання даних.

Для багатьох підприємств безпека даних є однією з основних проблем при відправці файлів в хмару. Вони турбуються про те, що їх файли будуть переглянуті або навіть скомпрометовані.

Для захисту цілісності даних, що зберігаються, необхідні методи шифрування даних. У минулому багато компаній відчували себе комфортно, дозволяючи постачальникам хмарних послуг управляти всіма своїми даними, вважаючи, що ризиками безпеки можна керувати за допомогою контрактів, засобів контролю і аудиту. Однак з часом стало очевидно, що постачальники хмарних послуг не можуть виконувати такі зобов'язання.

У цьому документі обговорюється важливість шифрування бази даних і дається детальний огляд різних методів шифрування.

Ключові слова: шифрування, криптографія, хешування, безпека, база даних, хмарне сховище, зашифрований текст.