International Journal of Contemporary Research in Computer Science and Technology (IJCRCST) e-ISSN: 2395-5325
Volume 2, Issue 3 (March '2016)

THREE LEVEL ENCRYPTION SCHEME FOR CLOUD IN HEALTH CARE SYSTEMS

Abinaya.J,
Student,
Computer Science of Engineering,
Velammal Institute Of Technology,
Chennai, India.

Chitra.R,
Assistant Professor,
Computer Science of Engineering,
Velammal Institute Of Technology,
Chennai, India.

Sangeetha.B ,
Student,
Computer Science of Engineering,
Velammal Institute Of Technology,
Chennai, India.

Sree Keerthi.V.K,
Student,
Computer Science of Engineering,
Velammal Institute Of Technology,
Chennai, India .

Abstract: Big data, can mine new knowledge for economic growth and technical innovation. Big data processing is emerging faster due to its high volume, velocity, and variety (referred to as "3V") challenges. The flourishing of big data also hinges on fully understanding and managing newly arising security and privacy challenges. If the data is not authentic, new mined knowledge will be unconvincing. If privacy is not well addressed, people may be reluctant to share their data. Because security has been investigated as a new dimension, "veracity," in big data. In this project we formalize the general architecture of big data analytics, identify the corresponding privacy requirements, and introduce an efficient and privacy Triple DES as an example in response to the security and privacy requirements in the data stored in cloud by health care systems.

Keywords: Triple DES, De –identification, Server authentication, Database Encryption.

I. INTRODUCTION

Cloud Computing has become popular recently due to several advantages over traditional computing models. Typical advantages include flexibility, scalability, agility, energy efficiency and cost saving [1]. For this reason, it has been expected to be a dominant computing model in the future. Many efforts on big data are focused on the 3V challenges today. However, the flourishing of big data relies not only on the promised solutions for 3V challenges, but also on the security and privacy challenges in big data analytics. It is likely that if the security and privacy challenges are not well addressed, the concept of big data cannot be widely accepted. For example, when big data is exploited in the healthcare con-text, it could save the health care industry up to US$450 bill-lion. Nevertheless, as patients' data are very sensitive, privacy issues become a major concern when exploiting big data in healthcare. Recently a new dimension,"Veracity",has been investigated in big data security to ensure the trustworthiness and accuracy of big data, while the research of big data privacy is still at an early stage.Hence,in this article we exploit new privacy challenges of big data, especially devoting attention towards efficient and privacy preserving computing in the big data era.First,we give a general architecture of big data analytics and its privacy requirements .Then re view some existing privacy preserving big data analytics.To identify big data privacy research more clearly,we also present an efficient and privacy preserving **Triple DES** algorithm as an example. In the end, we draw our conclusions.

II. RELATED WORK

In this section, we overview some existing privacy-preserving techniques , including privacy-preserving aggregation, operations over encrypted data, and de- identification techniques.

Cosine similarity protocol:
In the existing System using Cosine Similarity check Protocol is used for search over the Encrypted data ,Using this protocol ,it can take long time to search over encrypted data, because All the Stored document must be decrypted before starting the comparison process ,All Stored patient information must be encrypted using the same public key So, this methodology is less secure.

Operations over encrypted data*:*
Currently, searching over encrypted data has been widely studied in cloud computing [5]. To keep sensitive documents private, documents and their as sociated keywords are encrypted and stored in a cloud server. When a user submits a "capability" encoding some query conditions, the server can return a set of encrypted documents that meet the underlying

query conditions without knowing other details of the query. In such a way, the user can retrieve the desired data in a privacy-preserving way. Motivated by searching over encrypted data, our first feeling is that we can also run operations over encrypted data to protect individual privacy in big data analytics. However, as operations over encrypted data are usually complex and time-consuming, while big data is high-volume and needs us to mine new knowledge in a reasonable timeframe, running operations over encrypted data is inefficient in big data analytics.

De-identification:

De-identification is a traditional technique for privacy-preserving data mining, where in order to protect individual privacy, data should be first sanitized with generalization (replacing quasi- identifiers with less specific but semantically consistent values) and suppression (not releasing some values at all) before the release for data mining. Compared to privacy-preserving aggregation and operations over encrypted data, de-identification can make data analytics and mining more effective and flexible [7]. However, many real examples indicate that data which may look anonymous is actually not after de-identification; for example, only (5-digit zip code, birth date, gender) can uniquely identify 80 percent of the population in the United States. Therefore, to mitigate the threats from re-identification, the concepts of *k*-anonymity, *l*-diversity, and *t*-closeness have been introduced to enhance traditional privacy-preserving data mining. Obviously, de-identification is a crucial tool in privacy protection, and can be migrated to privacy preserving big data analytics. However, as an attacker can possibly get more external information assistance for de-identification in the big data era[3], we have to be aware that big data can also increase the risk of re-identification. As a result, de-identification is not sufficient for protecting big data privacy.

From the above discussion, we can see that:

- Privacy-preserving big data analytics is still challenging due to either the issues of flexibility and efficiency or de -identification risks.

- However, compared with privacy-preserving aggregation and operations over encrypted data, de-identification is more feasible for privacy-preserving big data analytics if we can develop efficient and privacy-preserving algorithms to help mitigate the risk of re-identification

III. PROPOSED METHODOLOGY

In the proposed methodology we can implement different encryption key for each stored document, Further for fast way to perform search process ,all the document's searchable keyword are Encrypted using same key , so the secrecy of the native data is maintained. Research work on big data privacy should be directed towards efficient and privacy preserving computing algorithms in the big data era, and these algorithms should be efficiently implemented and output correct results while hiding raw individual data. In such a way, we can reduce the re-identification risk in big data analytic and mining by using privacy preserving **Triple DES and LCG** algorithm. Triple DES (aka 3DES, 3-DES, TDES) is based on the DES (Data Encryption Standard) algorithm, therefore it is very easy to modify existing software to use Triple DES. It also has the advantage of proven reliability and a longer key length that eliminates many of the attacks that can be used to reduce the amount of time it takes to break DES. A **linear congruential generator** (**LCG**) is an algorithm that yields a sequence of pseudo-randomized numbers calculated with a discontinuous piecewise linear equation. The method represents one of the oldest and best-known pseudorandom number generator algorithms.

IV. ALGORITHM:

TRIPLE DES ALGORITHM:

In cryptography, **Triple DES** (**3DES**) is the common name for the **Triple Data Encryption Algorithm** (**TDEA** or **Triple DEA**) symmetric-key block cipher, which applies the Data Encryption Standard (DES) cipher algorithm three times to each data block. The original DES cipher's key size of 56 bits was generally sufficient when that algorithm was designed, but the availability of increasing computational power made brute-force attacks feasible. Triple DES provides a relatively simple method of increasing the key size of DES to protect against such attacks, without the need to design a completely new block cipher algorithm.Triple DES uses a "key bundle" that comprises three DES keys, K_1, K_2 and K_3, each of 56 bits (excluding parity bits). The encryption algorithm is:

$$\textbf{ciphertext} = E_{K3}(D_{K2}(E_{K1}(\textbf{plaintext})))$$

I.e., DES encrypt with K_1, DES *decrypt* with K_2,then DES encrypt with K_3Decryption is the reverse:

$$\textbf{plaintext} = D_{K1}(E_{K2}(D_{K3}(\textbf{ciphertext})))$$

I.e., decrypt with K_3, *encrypt* with K_2, then decrypt with K_1.

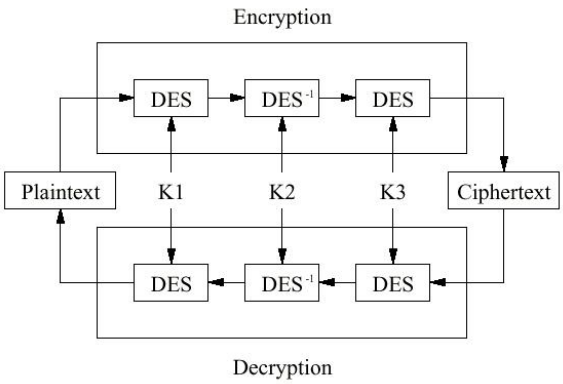

Figure1: Triple DES algorithm

Each triple encryption encrypts one block of 64 bits of data.In each case the middle operation is the reverse of the first and last. This improves the strength of the algorithm when using keying option 2, and provides backward compatibility with DES with keying option 3.

KEYING OPTIONS:

Keying option 1

All three keys are independent.

Keying option 2

K_1 and K_2 are independent, and $K_3 = K_1$.

Keying option 3

All three keys are identical, i.e. $K_1 = K_2 = K_3$.

Keying option 1 is the strongest, with 3 × 56 = 168 independent key bits.

Keying option 2 provides less security, with 2 × 56 = 112 key bits. This option is stronger than simply DES encrypting twice, e.g. with K_1 and K_2, because it protects against meet-in-the-middle attacks.

Keying option 3 is equivalent to DES, with only 56 key bits. It provides backward compatibility with DES, because the first and second DES operations cancel out. It is no longer recommended by the **National Institute of Standards and Technology** (NIST), and is not supported by

Each DES key is nominally stored or transmitted as 8 bytes, each of odd parity,[11] so a key bundle requires 24 bytes for option 1, 16 for option 2, or 8 for option 3.

LCG ALGORITHM:

LCGs are fast and require minimal memory (typically 32 or 64 bits) to retain state. This makes them valuable for simulating multiple independent streams.

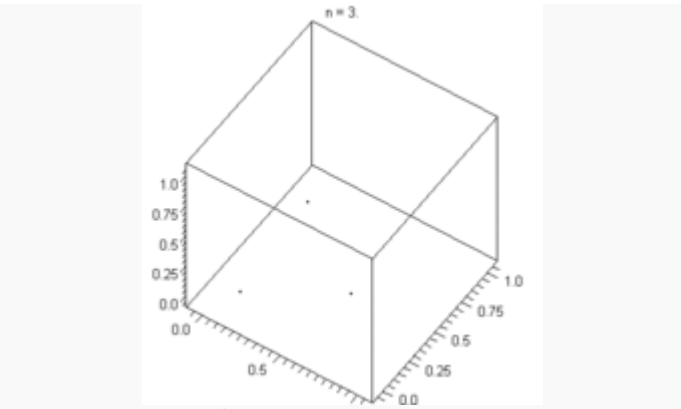

Figure2: Hyperplanes of a linear congruential generator in three dimensions

LCGs should not be used for applications where high-quality randomness is critical. For example, it is not suitable for a Monte Carlo simulation because of the serial correlation(among other things). They also must not be used for cryptographic applications; see cryptographically secure pseudo-random number generator for more suitable generators. If a linear congruential generator is seeded with a character and then iterated once, the result is a simple classical cipher called an affine cipher ; this cipher is easily broken by standard frequency analysis The generator is defined by the recurrence relation:

$$X_{n+1} = (aX_n + c) \mod m$$

Where X is the sequence of pseudorandom values, and

m, 0 < m – the "modulus"
a, 0 < a < m– the "multiplier"
c , 0 ≤ c < m–the "increment"

MODULES:

In this section we give a detailed explanation about the modules involved in the proposed privacy preserving system.

SERVER AUTHENTICATION

In This Module we Design new Client side page to actively interact with server , in order to improve the security purpose we generate unique Authentication scheme for each and every user(patient).Once Server Authentication process is completed , we design patient info web port ,to get all necessary information about a patient's health report. Each patient's uploaded document must be stored in separate encryption Key.This key information is not known to third party(health insurance company).In this module we use LCG algorithm to generate infinite random numbers which in turn creates separate key in the server side for each patient.

DATABASE ENCRYPTION

In this module we use **Triple DES** algorithm to encrypt the database information.It encrypts the patient's information such as name, address and all other information. These information can be accessed by a third party only if it has a key. If a person has a key he can access only a particular patient's information to which the key corresponds.

IMPROVED DE –IDENTIFICATION

Many real examples indicate that data which may look anonymous is actually not after de-identification; for example, only (5-digit zip code, birth date, gender) can uniquely identify 80 percent of the population in the United States. Therefore, to mitigate the threats from re-identification, the concepts of *k*-anonymity, *l*-diversity, and *t*-closeness have been introduced to enhance traditional privacy-preserving data mining. Obviously, de-identification is a crucial tool in privacy protection, and can be migrated to privacy preserving big data ana-

lytics. However, as an attacker can possibly get more external information assistance for de-identification in the big data era, we have to be aware that big data can also increase the risk of re-identification. As a result, de-identification is not sufficient for protecting big data privacy hence we use improved de-identification. Improved de-identification is more feasible for privacy-preserving big data analytics since it uses efficient and privacy-preserving algorithms to help mitigate the risk of re-identification

V. RESULT

Thus the proposed methodology provides privacy preserving and secured system which maintains the patient's information in the cloud with a high level of security. Compared to the previous methods which uses Cosine Similarity protocol the proposed system which uses Triple DES and LCG algorithm is efficient.

VI. CONCLUSION

In this article, we have investigated the privacy challenges in the big data era by first identifying big data privacy requirements and then discussing whether existing privacy-preserving techniques are sufficient for big data processing. We have also introduced an efficient and privacy-preserving **Triple DES and LCG** algorithms in response to the efficiency and privacy requirements of data mining in the big data era. Although we have analyzed the privacy and efficiency challenges in general big data analytics to shed light on the privacy research in big data, significant research efforts should be further put into addressing unique privacy issues in some specific big data analytics in health care systems.

VII. REFERENCES:

[1] IBM, "Big data at the speed of business," http://www-01.ibm.com/software/data/bigdata/, 2012.

[2] S. Liu, "Exploring the future of computing," IT Professional, vol. 15, no.1, 2013, pp. 2–3.

[3] Oracle, "Oracle big data for the enterprise," http://www.oracle.com/caen/technologies/big-data, 2012.

[4] "Big data at CSAIL," http://bigdata.csail.mit.edu/..

[5] P. Paillier, "Public-key cryptosystems based on Composite degree residuosity classes," EUROCRYPT, 1999, pp. 223–38.

[6] A. Cavoukian and J. Jonas, "Privacy by design in the age of big data,"Office of the Information and Privacy Commissioner, 2012.

[7] R. Lu, X. Lin, and X. Shen, "SPOC: A secure and privacy-preserving opportunistic computing framework for mobile-healthcare emergency,"