Message Sharing Database Security Protocol Designing (MSDSP)

Somchai Prakancharoen
Department of Computer and Information Science,
Faculty of Applied Science,
King Mongkut's University of Technology North Bangkok, Thailand
spk@kmutnb.ac.th

Abstract - **The objective of this research was to design a security protocol of database manipulation. Stakeholder of this protocol were Security Administrator (SA), Database Client (DC), and Database Administrator (DA). SA was responsible in data encryption and decryption. DA acted just as a data keeper. DC was one who process data manipulation on plain data. Under SA responsibility, cipher text was transformed to secret shares. After that, this secret shares were sent to be kept in many database engines under DA's management. When DC requested specific record for data processing task, The ID of this record was sent to SA. SA requested this defined record ID to all DA. All DA retrieved ordered record ID from their database engine then sent it back to SA. SA used these secret shares to reconstruct the plain text then sent it to DC in order to perform any processing on plain data. When the processing was ended, SA compiled encrypt the new plain text and do transform this one to a new secret shares and repeat sending them to DA to look after the secret shares again as before. These designed security protocol, especially in data processing, were written as function program prototype. The function program prototype and detail of designed security protocol were sent to five security experts in order to evaluate of its performance and data processing correction. Evaluation result indicates that average appreciation on all evaluation topics were in good level (3.60). Expert's opinion on possibility of practicable application was about 70.20%.**

Keywords - **Security Protocol, Database, Secret Sharing**

I. INTRODUCTION

In database manipulation, there are many problems about secrecy or privacy of kept data in database engine. Many intruders try to attack database data in order to gain some sensitive organization's information.

This research suggests a "Message sharing database security protocol". This protocol should protect data that kept in database engine from intruder or even untrustworthy DA.

There were three actors work in this protocol. Three actors have different responsibilities. SA took care of security activities. A group of DA are data record (secret shares) keepers and retrievers. In further more DA may be applied as anonymous cloud databases. DC is one who under take data processing on plain text. For message sharing task in this research, Shamir's secret sharing scheme was used to create pieces of secrets.

In addition, this designed MSDSP protocol could increase more secrecy on DC, who has charge of data processing, by checking authentication login and frequently change public key.

II. RELATED TECHNIQUES AND RESEARCH

A. Shamir's Secret Sharing Scheme [1]

Secret sharing is a technique that could construct new secret shares from a given plain text (in integer number form: Z), called as "a_0". Univariate function has to be defined, $f(x)$, in order to construct a number of difference function values, $D(x_i, f(x_i))$. The a_0 value should be reconstructed by some mathematic techniques.

There are many secret sharing schemes such as Sharmir's secret sharing scheme, Chiness remainder. Sharmir's secret sharing was used in this protocol designing.

According to Sharmir's secret sharing, the actor has to arbitrarily defined n – order univariate polynomial equation, $f(x) = a_0 + a_1x^1 + a_2x^2 + a_3x^3 + ...a_nx^n$. Actor has to define the value of $a_1, a_2, a_3, ...a_n$ while a_0 is a plain text that aim used it to construct secret shares.

For example 3 order univariate polynomial equation as equation (1)

$$f(x) = a_0 + 2x^1 + 4x^2 + 2x^3 \quad (1)$$

The values of function of equation (1), $D(x_i, f(x_i)), i = 0,1,2,3$, when x_i are assigned, $x_i = 1,2,3$, are D_0 = 3, D_1 = 11, D_2 = 39, and D_3 = 99.

This function value could not imply any meaning to plain text, D_0, even though DA can reveal all function values.

These secret shares, D_1, D_2, D_3, are sent and kept in a set of database engines. In case of D_0 reconstruction, D_1, D_2, D_3 are retrieved back from allocated databases then (1) should be reconstruct by Lagrange basis polynomial (2, 3).

$$l_j(x) = \frac{x - x_0}{x_j = x_0} . \frac{x - x_{j+1}}{x_j - x_{j+1}} \quad (2)$$

$$f(x) = \sum_{j=0}^{n} D_{x_j} . l_j(x) \quad (3)$$

After (3) is found out, the value of a_0 or D_0 is turn up. Note that, the number of called back D_{x_j} must larger or equal to $n+1$ in order to successfully compile equation (2) and (3).

B. Public Encryption System [2]

Public key system is a cryptology technique. There are two kinds of key. The first is public key, K_{pub}. Normally, public key is used to encrypt a plain text to be a cipher text. The second is private key. K_{priv}. This key is used to decrypt cipher text to become an original plain text. In general, private key is kept secret and only known by one who wants to get a plain text.

This research chose "RSA public key system" because of simple implementation while still preserved high level of secrecy.

C. Related Research

1) **Database Firewall [3]**

Database management system is the most important mechanism that provide protect of unauthorized access and data modification. Access control on logical and physical access is configured as rule base in order to control and protect database security intrusion.

This technique can protect data content that are stored in database engine from outside intruders. However, this technique is unsecure if actor organization's database administrator is dishonest.

2) **Message or Secret Sharing [4]**

This technique tries to separately manage on each column (attribute). For example, attributes "name", "position" are separately kept, accessed or updated on vary database engines.

Data storing in this technique are still in database engines therefore stored data should be encrypt in order to increase more security protection. Nevertheless, DA and DC are still could observe all plain text.

III. MSDSP PROTOCOL DESIGN

A. Overall Protocol Design

There are three actors (SA, DA, and DC) that cooperate their tasks in order to fulfilled protocol functions. Security administrator (SA) take care of security activities. Database administrators (DA) act as data record (secret shares) keepers and retrievers. Database client (DC) is one who perform data access and update.

The design of "message sharing database security protocol designing (MSDSP)" as shown in fig. 1, Flowchart message sharing database security protocol designing (MSDSP).

There are some tasks are not shown in the figure since they are performed that task by them self.

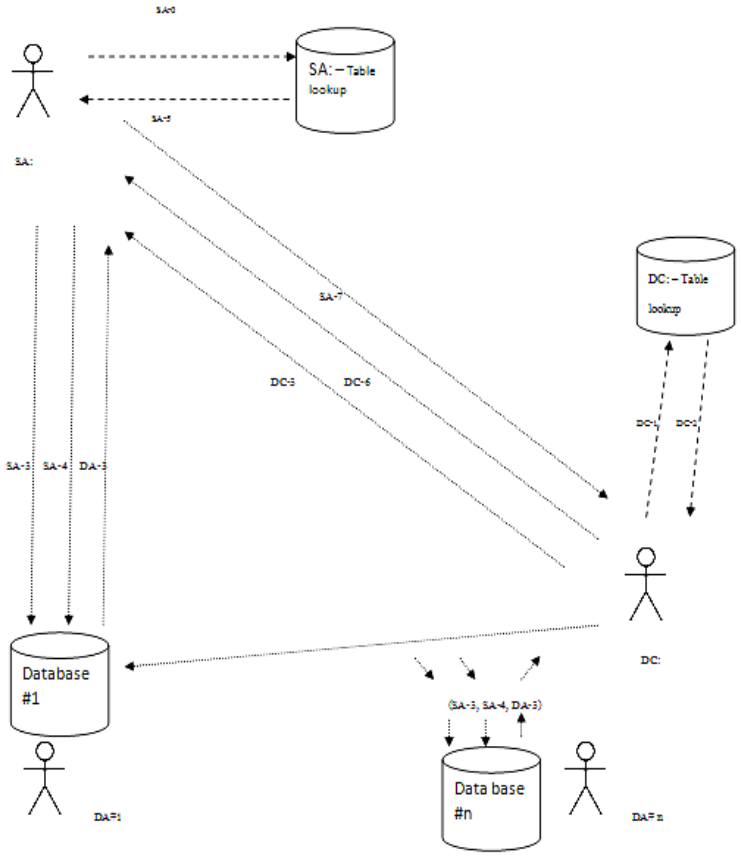

Fig. 1 Flowchart Message Sharing Database Security Protocol Designing (MSDSP)

B. Protocol Actor and Actor's Task

The details of every tasks of each actor are explained in table I. There are seventeen tasks that must perform in order to complete data access and update.

In addition, MSDSP require two table lookups for keeping public key or other parameters which need used in their task processing.

TABLE I
ACTOR'S TASK OF MSDSP (DA)

		Task	Description	Related actors or System
Actor	DA:	DA-1	store retrieved data record (secret share) to database engine#1	Database#1
		DA-2	retrieve data record	Database#1
		DA-3	send retrieved kept secret share to SA:	SA:

TABLE II
ACTOR'S TASK OF MSDSP (SA)

		Task	Description	Related actors or System
Actor	SA:	SA-0	public & private key preparation the keep then in SA:'s table lookup	SA:'s table lookup
		SA-1	encrypt initial plain text "m" with public key	-
		SA-2	construct secret shares of this cipher text to n piece secrets	-
		SA-3	store these n pieces into separate database engine #1…#n	Database engines#1…#n
		SA-4	request DA:s for data record (secret shares) with specific ID	DA:
		SA-5	re construct secret (a_0 or private key)	-
		SA-6	decrypt cipher text	-
		SA-7	sent plain text to DC:	DC:
		SA-8	Perform repetitively SA-2, SA-3	Database engines#1…#n

TABLE III
ACTOR'S TASK OF MSDSP (DC)

		Task	Description	Related actors or System
Actor	DC:	DC-1	public key receiving and keeping	SA:,DC:- table lookup
		DC-2	public key retrieving	DC:- table lookup
		DA-3	request SA: for data record with assigned record ID	SA:
		DA-4	perform processing on received data record	-
		DA-5	encrypt processed data with public key (from DC-2)	-
		DA-6	sent cipher text to SA:	SA:

C. Protocol Evaluation Topics

The evaluation topics of protocol evaluation were coverage in performance and data processing correction. Likert's five scale of preference as shown in table II.

TABLE IV
PREFERENCE LEVEL
IN PROTOCOL EVALUATION

Level	score
Not desire	0.00-1.00
Low desire	1.01-2.00
Moderate desire	2.01-3.00
More desire	3.01-4.00
Most desire	4.01-5.00

IV. PROTOCOL EVALUATION

In case of performance of database manipulation, reference on table I, the MSDSP takes at least 17 tasks while ordinary database manipulation require only 4 tasks, consider on data access and data update activities.

Step of tasks in ordinary database manipulation are 4 steps, as shown here.

Start:

DC- -1st request for record ID - ->
- ->DA - -2nd retrieve and sent data record back to DC- - >
- ->DC- -3rd data update and sent to DA- - >
- ->DA - -4th store data record in database- - > Database

Stop:

V. SUMMARY AND SUGGESTION

A. Summary

Unfortunately, MSDSP require four times of activities more than ordinary database manipulation in the same job. Organization has to trade off between high security level and low availability.

B. Suggestion

Although MSDSP can enhance privacy from untrustworthy DA (may be untrustworthy cloud database) but there are some vulnerability such as untrustworthy DC and unauthentic actors. Therefore, the next adaptation of this protocol are to, the first, detect authentication of every actors before login to system and, the second, to prevent DC from access directly to disclosed data. Homomorphic encryption schemes [5] should be applied in further research.

VI. ACKNOWLEDGMENT

Message Sharing Database Security Protocol Designing has been support from annual research funding, year 2015, of "Science and Technology Institute: STRI" King Mongkut's University of Technology North Bangkok Thailand.

REFERENCES

(Arranged in the order of citation in the same fashion as the case of Footnotes.)

[1] Shamir, A. (1979). "How to share a secret". Communications of the ACM, 22(11), pp. 612-613, doi: 10.1145/359168.359176.

[2] Stalling, W. (2010). "Fire wall-cryptography and network security (4th Ed)". USA.

[3] Kedar, S. (2009). "Database Management Systems". Technical Publications, pp. 15, ISBN: 978-81-8431-584-4.

[4] Wagner, D. (2004). "Secret sharing". Berkeley University, USA.

[5] Halevi, S. (2013). "Homomorphic Encryption and Lattices". Columbia University, USA.