A Study of secure data management system using homomorphic encryption in defense environment

Hyun-Jong Cha [1], Jin-Mook Kim [2], Hwang-Bin Ryou [3]

[1.] Department of Defense Acquisition Program, Kwangwoon University
[2.] Division of IT Education, Sunmoon University
[3.] Department of Computer Science, Kwangwoon University
chj826@kw.ac.kr, calf0425@sunmoon.ac.kr, ryou@kw.ac.kr

Abstract: The cloud computing is a system enabling the use of information stored in data center by mobile device without any software. The Ministry of National Defense aims to building of national defense cloud computing environment as a part of Basic Plan for Information in National Defense by 2017 published in 2011. The security for personal information such as information availability should be also considered in cloud computing. The discussion about security measures is essential, especially, because the field of national defense includes military informations influencing directly to the national security. The enhancement of security, however, makes may difficult for information to be managed in real-time, resulting in limiting the use of information in real time, therefore the both security and efficiency should be considered. In this study, a homomorphic crypto system for safe search in network-centric warfare is proposed. The proposed technique allows the searching encrypted information without process of decryption, using a homomorphic cryptographic technique capable of providing service without decryption of encrypted information. The application of proposed technique ensures the safe management of information.
[Hyun-Jong Cha, Jin-Mook Kim, Hwang-Bin Ryou. **Taxonomic Diversity of Understorey Vegetation in Kumaun Himalayan Forests**. *Life Sci J* 2014;11(7s):145-149] (ISSN:1097-8135). http://www.lifesciencesite.com. 27

Keywords: Defense Environment; Homomorphic Encryption; Secure data management

1. Introduction

The cloud clouding is a technique providing users IT resources, using network techniques. The users don't carry IT resources such as network, server, storage, service, application for information processing, but they lend resources as much as need and pay for them. The field of national defense is more active in studying those techniques than civil communities because of necessity of occupying advantage by sharing mass information in war.

The merit of cloud computing that it allows easy using of information even without expertise or IT technique may be vulnerability in view of security. The IT resources providing service should be en- and decrypted because of personal information of users, amount of information, and frequent accesses. The discussion about security measures is essential, especially, because the field of national defense includes military informations influencing directly to the national security, however whiling maintaining the availability of information for real time use [1, 2, 3].

In this study, a homomorphic crypto system for safe search in network-centric warfare is proposed. The proposed technique allows the searching encrypted information without process of decryption. The application of proposed technique ensures the safe management of information.

2. Related research

2.1 Defense cloud computing

The introduction of cloud computing in national defense system has some advantages. First, the space and energy in data centers are saved. Second, the operation and building costs are saved due to integration of systems. The intensive security management is necessary in army because, in army, the services are provided during movement using wire and wireless communication against network war.

The Defense Information Systems Agency (DISA) under U.S. Department of Defense built the Rapid Access Computing Environment (RACE) in 2008. RACE is a service development infra and test environment, providing such required development environments as server and web application platform via internet. This system allowed DISA to achieve saving in cost for purchasing IT infra, and to improve productivity and save operation cost by reduction in resource allocation time per request from each client's [4].

2.2 Homomorphic encryption

The homomorphism refers to a mapping between two sets possessing defined operation, preserving the operations. The homomorphic encryptions is an encryption technique allowing operation of plaintext by applying typical arithmetic operation such as addition and multiply to coded

messages. The fully homomorphic encryption is differentiated from basic one in that it preserves all arbitrary logical operation. The fully homomorphic encryption may be summarized as an encryption technique capable of performing AND and XOR operation in bit unit using coded messages without decryption because this technique organizes any kind of operation by combining given AND and XOR operation [5, 6, 7, 8, 9, 10].

3. Secure data management system

The searching technique for encrypted data proposed in this study is as follows.

3.1 System overview

The layout of system for searching encrypted date is shown in Figure 1.

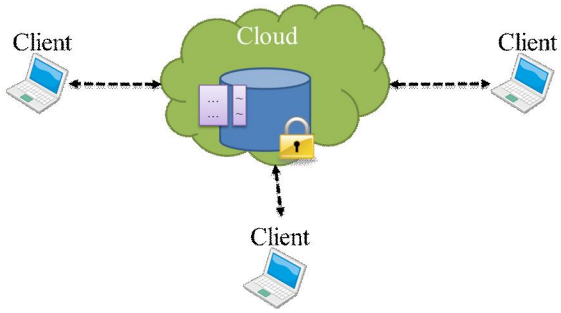

Figure 1. System overview

The encrypted data managed in cloud service can be accessed by an authorized user through authentication. The data managed by server are encrypted and any revisions of data other than storage and deletion are not allowed in server. The keywords used in searches are generated for each data stored in cloud service. The encrypted messages are sent and received in communicating with clients, so the contents of intercepted packet during communication would not be decrypted by enemies because they are encrypted and sent by using keywords for search.

3.2 Data generation

The data is generated and encrypted by using agreed encryption algorithm by authorized clients. The clients generate keywords, which is central concept, for search as well. Those keywords and encrypted data are sent to server.

The user is able to create data only when he has been authorized access privileges, owns the corresponding group key, and has data creation privileges. The process of data creation is shown in Figure 2. The specific steps for data creation are as follows.

Step 1: Client encrypts the data using the owned group key, and creates the keyword hash value for the data.

Step 2: A single message is made from the created ciphertext and hash value, which is then transmitted

Step 3: For the hash value received from Client, a trap door is created by the role of the index linked with data, and stored.

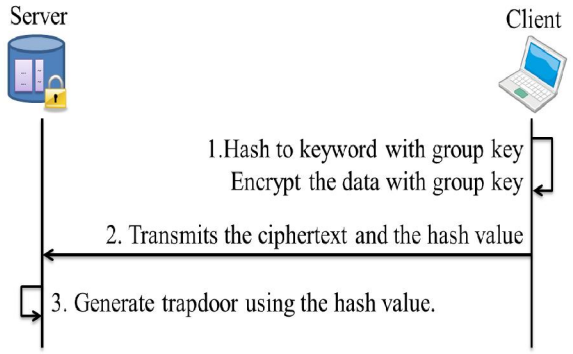

Figure 2. The generation procedure of encrypted data

3.3 Data search

The data created by User A are stored at the server in an encrypted state. In addition, a trap door is created with the corresponding keyword, which is also stored. Figure 3 shows the scenario whereby encrypted state data are searched.

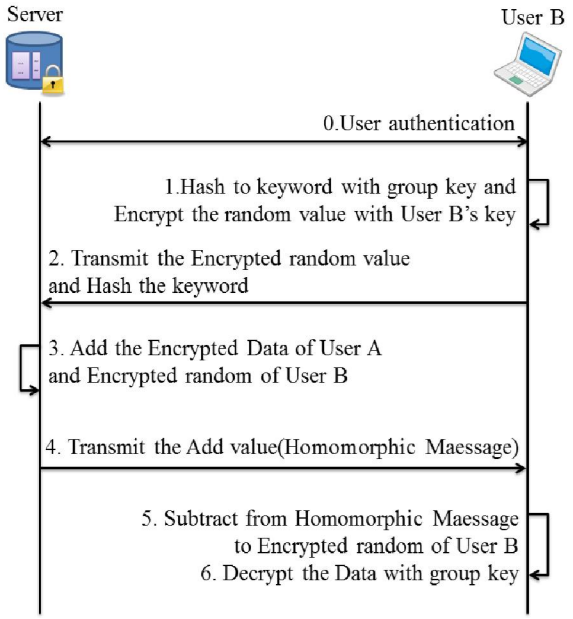

Figure 3. The search procedure of encrypted data

User B wants to search data. Before searching data at the database server, User B has

went through the authentication step for gaining access privileges. Also, as User A belongs to the same group, User B has the same group key as User A. The specific steps for data search are as follows.

Step 1: User B encrypts the random number created from his system with his key, and calculates the hash value with the keyword to search as the group key.

Step 2: The ciphertext with respect to the random number and the hash value of the keyword to search are transmitted to the server.

Step 3-1: The server compares the received hash value with the trap door that it owns and extracts the same data. If there is no same hash value, it means there are no data, so the scenario ends.

Step 3-2: If there is the same value as the hash value received from User B, the corresponding data are prepared to be transmitted to User B. An operation is done on the encrypted random number received from User B and the stored data.

Step 4: The server transmits to User B the ciphertext of the data created by User A, as well as the computed value of the ciphertext of the random number created by User B, which is Homomorphic Message.

Step 5: An inverse operation is done on the ciphertext of the random number created by User B with respect to the computed value of the ciphertext received from the server, Homomorphic Massage.

Step 6: User B can decrypt the ciphertext with the group key that he owns to obtain the plaintext.

3.4 Data share and modify

Figure 4 shows the scenarios of the data modify.

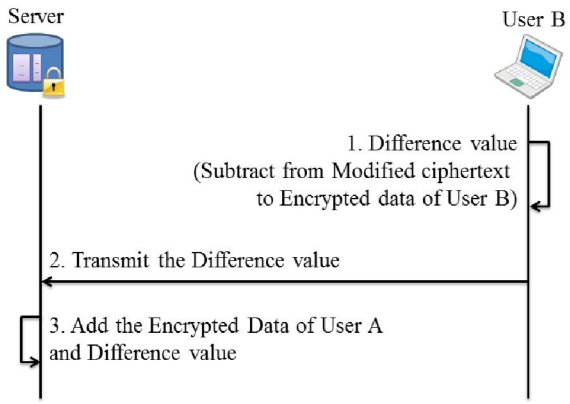

Figure 4. The search procedure of data modify

User A and User B can access data because they've gained access privileges to the server. Because User A and User B belong to the same group, they have the same group key. At the server the ciphertext of the data created by User A is stored.

For these data, User B not only has read privileges but modify privileges as well. The encryption data created by User A, which are stored at the server, have been received by User B after doing a search.

The specific steps for data modification are as follows.

Step 1: User B has modified the data created by User A in order to create modified data of User A, and has encrypted it with a group key. Afterward, an operation was done with the ciphertext of the original data.

Step 2: The computed result value is transmitted to the server.

Step 3: The server receives the computed value and performs an inverse operation with the data being stored. This is the server performing configuration management in order to make records that the data have been modified.

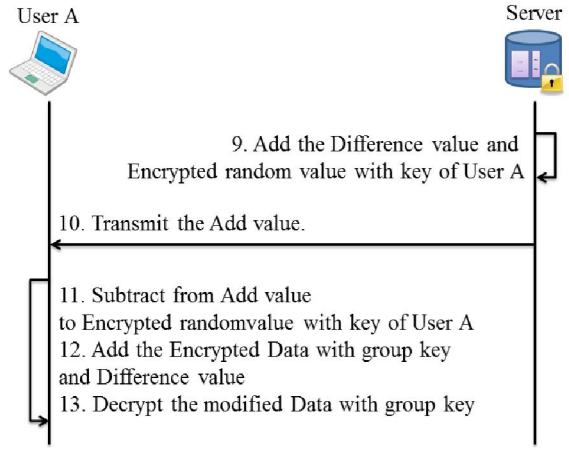

Figure 5. The search procedure of data shared

Figure 5 shows the scenario of the data shared. The modified value was applied at the server. Afterward, User A wants to again search for the data that he created. Steps 4-8 are the data search steps of Section 3.2, so they're omitted. The specific steps for User A receiving the modified data at a later time and decrypting them are follows.

Step 9: The server computes the ciphertext for D which User B has transmitted and the random number which User A has transmitted.

Step 10: The computed results are transmitted to User A.

Step 11: The ciphertext of the random number that one transmitted to the server is decrypted in order to obtain the data that User B transmitted to the server.

Step 12: An inverse operation is done on the original data that one has in possession and the data received from the server in order to obtain the ciphertext of the data that's been modified by User B.

Step 13: Decryption is done using the group key to obtain the modified data.

4. System Implementation and Analysis

The suggested system is able to use an existing encryption algorithm. The ciphertext operations are broadly divided into matrix operations and finding the degree of similarity between two ciphertexts.

4.1 Matrix Operations

To operate on two ciphertexts, matrix operations are performed. When two matrixes have the same size, corresponding elements can be added or subtracted. For multiplication, this is possible only when the number of rows of the first matrix is the same as the number of rows of the second matrix. Matrix operations are typically used to measure the degree of similarity between two images. For a block encryption algorithm, matrix operations are suitable because the number of blocks is fixed.

For the two $m \times n$ matrixes A and B, the addition (A+B) and subtraction (A-B) are defined respectively as the summation and difference of each of the elements as shown in (1) and (2).

$$(A+B)_{ij} = A_{ij} + B_{ij} \quad (1)$$

$$(A-B)_{ij} = A_{ij} - B_{ij} \quad (2)$$

4.2 Difference value operations

This operation is needed to get a value for the difference of two ciphertexts. This is similar to measuring the degree of similarity between two images or videos based on correlations. In the image or video comparison, the correlation-based matching generally consists of calculating the difference between pixels in the area. In the case of the block ciphertext, it is the same principles as comparing the values of pixels.

Therefore, among many different methods available for comparing the degree of similarity for images and videos, SAD (Sum of Absolute Differences) is chosen and modified for use as shown in (3). Note that I_1 refers to the first ciphertext in the comparison, while I_2 refers to the second.

$$I_1(i,j) - I_2(x+i, y+j) \quad (3)$$

4.3 Calculation of the difference between the ciphertext

For ciphertext operations, matrix operations and difference value operations are used together. Furthermore, if addition operation is done, then subtraction operation also has to be included, which is its inverse operation. Figure 6 below shows the part where ciphertext operations are done, in pseudocode.

```
read ciphertext1, ciphertext2
read Size ← length of ciphertext1,2

for i=0 to Size
    if operation == ADD than
        Result[i] ← chipertext1[i]+chipertext2[i]
    else
        Result[i] ← chipertext1[i]-chipertext2[i]
End for

Return Result
```

Figure 6. Difference algorithm

The calculations for getting the difference value between two ciphertexts have to be performed in a byte-by-byte comparison of the two ciphertexts. They consist of additions and subtractions.

4.4 Scenario

(Figure 6) shows a scenario formation for analyzing performance of proposed system. The military operation of future will be mainly a joint operation performed by units selected from each force rather than operation performed by each force separately. In the case of precautions for a boat drift away from North Korea, for instance, are as follows.

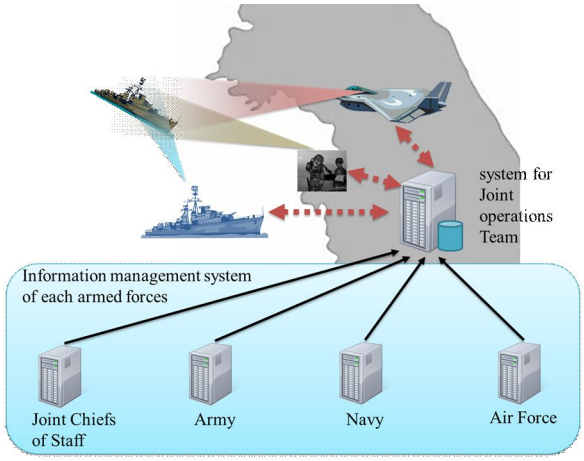

Figure 6. Scenario overview

An unidentified boat was detected by radar watching for maritime. This is reported to the office of joint chiefs staff, thus is propagated to each force, which order a task to nearest coast guard post, military patrol craft and military patrol plane. The ordered teams are grouped into access authority for team server and access group for the purpose of smooth sharing of information, and data are sent. The position of the boat is reported and the operation is performed jointly for the identification. The data about team's position and operation results from each team is encrypted and sent and stored in data in state of encryption as well.

4.5 Analysis for scenario

The stored in server and communicated data are encrypted therefore the confidentiality of data is kept even when the server is exposed outwardly. The data is not accessed even though the encrypted text is revealed and decrypted primarily if the group key is not available, because the operation is performed by using group key.

The existing method in army force was repetition of reports to higher unit and propagation to lower units of information, just as procedures in civil companies. The proposed system, however, is effective in faster sharing of information in virtue of real time sharing for the informations.

5. Discussions

This study proposes a homomorphic crypto system ensuring confidentiality in searching encrypted data in national security cloud computing. The proposed technique uses a homomorphic crypto technique, thus the encrypted data is possible to be searched without decryption. This system ensures confidentiality of data and anonymity of users during data searching. The study on technique of low complexity for moving users is warranted.

Corresponding Author:
Prof. Jin-Mook Kim
Division of IT Education
Sunmoon University
Asan-si, ChungNam, 336-708, Korea
E-mail: calf0425@sunmoon.ac.kr

4/28/2014

References

1. S. T. Kim, The introduction of cloud computing trends and military considerations (1), for Concepts and skills, defense Weekly, Korea Institute for Defense Analyses, 2011:1386:1-8.
2. S. T. Kim, The introduction of cloud computing trends and military considerations (2). For Cases the effect of introducing environmental considerations, defense Weekly, Korea Institute for Defense Analyses, 2012:1411:3-4.
3. N. S. Jho, K. Y. Jang, Trend and Issue on Homomorphic Encryption, Weekly IT Brief, 2011:1522:15-25.
4. R. L. Rivest, L. Adleman, M. L. Dertouzos, On data bank and privacy homomorphisms, Proceedings of the 9th Annual Symposium on Foundations of Secure Computation-FSC 1978, 1978:169-180.
5. Pascal paillier, Public-Key Cryptosystems based on Composite Degree Residue Classes, Proceedings of EuroCrypt 99, 1999:1592:223-238.
6. I. Damgard and M. Jurik, A generalization, a simplification and some applications of Paillier's probabilistic public-Key system, International Conference on Public Key Cryptography 2001, 2001:1992:119-136.
7. N. S. Jho, K. Y. Jang, Trend and Issue on Homomorphic Encryption, Weekly IT Brief, 2011:1522:15-25.
8. Rivest, Adleman, Dertouzos, On data bank and privacy homomorphisms, Proceedings of the 19th Annual Symposium on Foundations of Secure Computation-FSC 1978, 1978:169-180.
9. J.Domingo-Ferrer, A New privacy homo - morphism and applications, Information Processing letters, 1996:60(5):277-282.
10. Craig Gentry, Fully homomorphic encryption using ideal lattices, in Proceedings of the 41st ACM Symposium on Theory of Computing - STOC 2009, 2009:169-178.