Privacy-preserving Association Rule Mining Algorithm for Encrypted Data in Cloud Computing

Hyeong-Jin Kim, Jae-Hwan Shin, Young-ho Song
Dept. of Computer Science and Engineering
Chonbuk National University
Jeonju-si, Republic of Korea
{yeon_hui4, djtm99, songyoungho}@jbnu.ac.kr

Jae-Woo Chang*
Dept. of Information Technology and Engineering
Chonbuk National University, Jeonju-si, Korea
jwchang@jbnu.ac.kr
*Corresponding Author

***Abstract*—Recently, privacy-preserving association rules mining algorithms have been proposed to support data privacy. However, the algorithms have an additional overhead to insert fake items (or fake transactions) and cannot hide data frequency. In this paper, we propose a privacy-preserving association rule mining algorithm for encrypted data in cloud computing. For association rule mining, we utilize Apriori algorithm by using the Elgamal cryptosystem, without additional fake transactions. Thus the proposed algorithm can guarantee both data privacy and query privacy, while concealing data frequency. We show that the proposed algorithm achieves about 3-5 times better performance than the existing algorithm, in terms of association rule mining time.**

Keywords- association rule mining; Apriori algorithm; encrypted data; cloud computing; Elgamal cryptosystem;

I. INTRODUCTION

Research on preserving data privacy in outsourced databases has been spotlighted with the development of cloud computing. Because the outsourced database may include sensitive information, it should be protected against adversaries including a cloud server. Therefore, the database should be encrypted before being outsourced to the cloud. As one of the widely used data mining in the cloud, the association rule mining analyzes the specific data of a company and the association of sales information. Recently, privacy-preserving association rules mining algorithms have been proposed to support data security [1, 2, 3]. However, these algorithms have an additional overhead by inserting fake items and cannot hide the data frequency. During query processing, the cloud can derive sensitive information from the original data by observing data frequency even if both the data and the query are encrypted.

In this paper, we propose a privacy-preserving association rule mining algorithm for encrypted data in cloud computing. For association rule mining, we select the Apriori algorithm because it is widely used for frequent item set mining and association rule learning over transaction databases [4]. To verify that two ciphertexts have the same plaintext, we also propose a secure plaintext equality test protocol. As a result, the proposed algorithm can guarantee both data privacy and query privacy, while concealing data frequency.

II. RELATED WORK

To support data security, privacy-preserving association rule mining algorithms have been proposed. First, Wong et al. [1] proposed a one-to-many item mapping that transform transactions non-deterministically. However, there is a disadvantage that fake items are easily distinguished from the original data because the probability of fake items in the transaction database is the same. Second, Giannotti et al. [2] proposed an association rule mining algorithm using k-anonymity. This algorithm adds fake transactions to the transaction database so that each item can have k-1 frequency. However, the original data can be exposed if fake transaction is known. Also, additional operations are needed to remove the frequency of fake transactions. Finally, Xun et al. [3] proposed an association rule mining algorithm that supports k-anonymity on an encrypted database. This algorithm supports data protection and query protection by using Elgamal encryption system. However, it has an additional overhead for adding encrypted fake transactions. To compute the frequency of candidate set, it uses a conditional gate based on the binary array of ciphertext. However, the original data can be inferred if an attacker has some knowledge about data frequency because it does not encrypt the data frequency in query processing.

III. SYSTEM ARCHITECTURE AND SECURE PROTOCOL

A. System architecture

The typical types of adversaries are semi-honest and malicious [5]. We consider the clouds as insider adversaries who have more authorities than outsider attackers. In the semi-honest adversarial model, the cloud correctly follows the given protocol, but may try to obtain the additional information being not allowed to it. In the malicious adversarial model, the cloud can deviate from the protocol. We adopt a semi-honest adversarial model by following the earlier work [3]. The system architecture of the proposed algorithm is shown in Figure 1.

The system consists of Data Owner (*DO*), Cloud A(C_A), Cloud B(C_B), and Authorized User(*AU*). *DO* owns the original database, and *AU* is the service recipient who gains accesses to the cloud. The proposed algorithm uses secure two-party computation protocols, where two cloud servers, called C_A and C_B, perform computations securely. The procedure of building the system is as follows. First, *DO* generates an Elgamal encryption key pair and encrypts the original database. Second, *DO* sends both the encrypted database and the public key to C_A. Third, *DO* sends the Elgamal encryption key pair to C_B and sends the public key to *AU*. Finally, *AU* encrypts the query and sends it to C_A. Because the original data can be exposed using the plaintext equality test protocol [6], we propose a secure plaintext

2159-6190/19/$31.00 ©2019 IEEE
DOI 10.1109/CLOUD.2019.00086

equality test protocol(SPET) which checks whether two encrypted data are the same, without decrypting the original data. By using the SPET protocol, C_A perform the Apriori algorithm in cooperation with C_B.

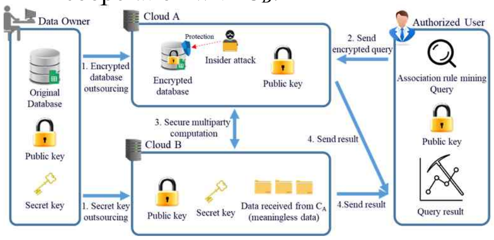

Figure 1. System architecure

B. *Secure protocol*

The proposed SPET protocol returns 1 if the plaintexts of two ciphers are equal and returns 0 otherwise. The SPET protocol is shown in Algorithm 1. First, C_B generates a composite number t and send E(t) to C_A(line 1~2). Second, C_A multiplies E(t) by E($cipher_1$) and E($cipher_2$), respectively. C_A sends g^{r1} and g^{r2} to C_B, where g^{r1} and g^{r2} represent the front of E($t \times cipher_1$) and that of E($t \times cipher_2$), respectively(line 3~6). Third, C_B returns $g^{r1} \times g^{x}$ and $g^{r2} \times g^{x}$, where x is the secret key(line 7~8). Fourth, C_A computes $\alpha = \frac{t \times m_1 g^{r1x}}{t \times m_2 g^{r2x}} \times \frac{g^{r2x}}{g^{r1x}} = \frac{m_1}{m_2}$(line 10). Finally C_A returns 1 if α is 1 and returns 0 otherwise(line 11~12).

Algorithm 1. Secure plaintext equality test protocol

Input: E($cipher_1$), E($cipher_2$)
Output: if $cipher_1$= $cipher_2$ return $\alpha = 1$ else $\alpha = 0$

C_B
01: generate t(t is composite number)
02: send E(t) to C_A
C_A
03: receive E(t) from C_B
04: E($cipher_1$)*E(t) = $(g^{r1}, t \times m_1 g^{r1x})$
05: E($cipher_2$)*E(t) = $(g^{r2}, t \times m_2 g^{r2x})$
06: send to g^{r1}, g^{r2} to C_B
C_B
07: calculate g^{r1x}, g^{r2x} $(x = secret\ \ key)$
08: send to g^{r1x}, g^{r2x} to C_A
C_A
09: receive g^{r1x}, g^{r2x} from C_B
10: calculate $\alpha = \frac{t \times m_1 g^{r1x}}{t \times m_2 g^{r2x}} \times \frac{g^{r2x}}{g^{r1x}}$
11: if α ==1, then return result = 1
12: else return result = 0

IV. PROPOSED ASSOCIATION RULE MINING ALGORITHM

For association rule mining, we propose a privacy-preserving Apriori algorithm by using SPET protocol in cloud computing. The proposed algorithm consists of candidate set generation and frequency set calculation.

A. *Candidate set generation*

The candidate set generation step generates a candidate set containing many patterns, each of which has multiple items. The procedure of the candidate set generation step is as follows. First, one pattern pair <p_1, p_2> is selected in the k-1 frequent set, where p_1 and p_2 are different patterns. Second, we perform a join operation between p_1's items and p_2's items, and insert the joined result into the candidate set, i.e., S_k, if the result consists of k items. Finally, we perform a join operation for all pairs except <p_1, p_2> and return S_k to C_A.

B. *Frequent set calculation*

The frequent set calculation step calculates the frequency of S_k, as shown in Algorithm 2. First, one pattern of S_k is selected (line 1~2). Second, the SPET protocol is performed between the items of the selected pattern and the items of the transaction. If the result of SPET protocol is 1, the number of the matched items($match$) is incremented by 1 (line 3~8). Third, when $match$ is equal to k, E($x.sup$) is multiplied by g, where g is an arbitrary integer that is not included in a cyclic group of the encryption key (line 9~10). Fourth, the SPET protocol is performed between E($x.sup$) and E(g^{minsup}). If the result of SPET is 1, the frequent attribute of x is included in the frequent set (line 11~14). Finally, the frequent set calculation for the remaining patterns of S_k is performed in the same way (line 15~16).

Algorithm 2. Frequent set calculation

Input: Candidate k-item set S_k
Output: Frequent set L_k

01: for(all $x \in S_k$)
02: for(all $y \in$E(T))
03: $match$=0
04: for(i = 0 to k)
05: for(j = 0 to $y.NumItem$)
06: if(SPET(x_i, y_j)) $match$++
07: end for
08: end for
09: if($match$==k){
10: enc_mul($x.sup$, g)
11: if(SPET($x.sup$, E(g^{minsup})) $x.freq$ = true }
12: end for
13: if($x.freq$ == true) $L_k \cup x$
14: end for
15: return L_k

C. *The proposed Apriori algorithm*

The proposed Apriori algorithm is shown in Algorithm 3. First, we set L_1 to 1-item sets which are received from the data owner (line 1). Second, we perform the candidate set generation algorithm of 4.1, called Candidate_set_generation(E(L_{k-1})), where E(L_{k-1}) represents the k-1 frequent set (line 4). Third, the frequency of $\underline{S_k}$ is calculated (line 6). Finally, if the k frequent set is no longer generated, the k-1 frequent set is returned (line 5).

Algorithm 3. Proposed Apriori Algorithm

Input: Encrypted transaction database E(T)
Item set length k
Candidate pattern set S_k
Output: Frequency pattern set L_{k-1}

01: $L_1 = \{l_1, \ldots, \underline{l_n} \mid \forall\ l \in$E($T$)$\}$
02: k = 2
03: while(TRUE)
04: E(S_k) = Candidate_set_generation(E(L_{k-1}))
=$\{c_1, \ldots, c_p | c \in k$ candidate set$\}$
05: if(E(S_k)=$\emptyset$) return E(L_{k-1}) to AU
06: E(L_k)= Frequent_set_calculation(E(T), E(S_k))

```
    k++
end while
```

V. SECURITY PROOF

In this section, we perform the security proof of the proposed Apriori algorithm. In the viewpoint of C_A, the proposed algorithm encrypts the data frequency and the encrypted database consists of unidentifiable encrypted transactions. Because the Elgamal cryptosystem returns different ciphertexts for the same plaintext, there is no leakage of the original data. In the viewpoint of C_B, the data cannot be exposed because the front of the ciphertext is not contained in the original data. Therefore, the proposed Apriori algorithm proves to be safe in the semi-honest model.

VI. PERFORMANCE ANALYSIS

We evaluate the performance of the proposed Apriori algorithm, called S-ARM (Secure Association Rule Mining). The performance analysis was done under Intel Xeon E3-1220v3 3.10GHz, 32GB RAM. The proposed algorithm uses GMP library to represent a big integer in an Elgamal cryptosystem. The proposed algorithm is compared with the DP-ARM (Data Privacy Association Rule Mining) algorithm proposed by Xun et al.[3] because DP-ARM is the only existing algorithm to support both data privacy and query privacy. For performance analysis, we use the retail dataset collected from the Belgian market [7], and measure the performance of S-ARM and DP-ARM by varying the number of data. We also measure their performances by varying support changes (*minsup*) from 5% to 30% of data. Table 1 shows parameters for our performance analysis.

TABLE I. PARAMETERS FOR PERFORMANCE ANALYSIS

The number of data	2k, 4k, 6k, 8k, 10k
Fake transaction ratio(φ)	50%, 100%
Minimum support	5%, 10%, 15%, 20%, 25%, 30%
Key Size	1024

A. Performance analysis varying the number of data

The performance result of S-ARM and DP-ARM by varying the number of data is shown in Figure 2. When *minsup* is 10% and φ is 50%, S-ARM shows 205% performance improvement on the average, compared with DP-ARM, and when φ is 100%., S-ARM shows 405% performance improvement. The reason is why S-ARM requires no additional operation for fake transactions unlike DP-ARM. In addition, S-ARM requires no binary operation by using Elgamal cryptosystem through SPET protocol.

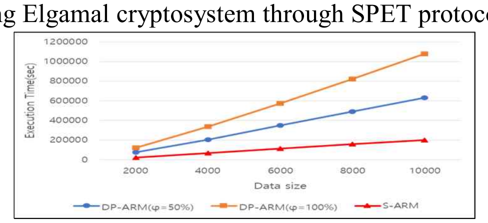

Figure 2. Performance result by varying the number of data

B. Performance analysis varying minsup

The performance result of S-ARM and DP-ARM according to *minsup* is shown in Figure 3. When the number of data is 10,000 and φ is 50%, S-ARM shows 216% performance improvement on the average, compared with DP-ARM, and when φ is 100%, S-ARM shows 429% performance improvement on the average. The reason is why S-ARM does not require no additional operation for the fake transactions unlike DP-ARM.

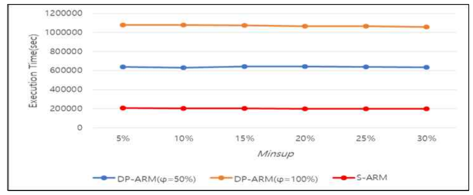

Figure 3. Performance result by varying minsup

VII. CONCLUSIONS AND FUTURE WORK

For association rule mining, we proposed a privacy-preserving Apriori algorithm using the Elgamal cryptosystem, without additional fake transactions for encrypted data. The proposed algorithm supports both data privacy and query privacy, while hiding data frequency in a cloud. We showed that the proposed algorithm achieves about 3~5 times better performance than the existing algorithm, in terms of association rule mining time. As a future work, we plan to study on the parallel execution of the proposed algorithm for fast processing.

ACKNOWLEDGMENT

This work was partly supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIP)(No. R0113-16-0005, Development of an Unified Data Engineering Technology for Large-scale Transaction Processing and Real-time Complex Analytics), This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(grant number 2016R1D1A3B03935298)

REFERENCES

[1] Wong, Wai Kit, et al. "Security in outsourcing of association rule mining." Proceedings of the 33rd international conference on Very large data bases. VLDB Endowment, 2007.

[2] Giannotti, Fosca, et al. "Privacy-preserving mining of association rules from outsourced transaction databases." IEEE Systems Journal 7.3 (2013): 385-395.

[3] Yi, Xun, et al. "Privacy-preserving association rule mining in cloud computing." Proceedings of the 10th ACM symposium on information, computer and communications security. ACM, 2015.

[4] Agrawal, Rakesh, and Ramakrishnan Srikant. "Fast algorithms for mining association rules." Proc. 20th int. conf. very large data bases, VLDB. Vol. 1215. 1994.

[5] Kim, Hyeong-Jin, Hyeong-Il Kim, and Jae-Woo Chang. "A Privacy-Preserving kNN Classification Algorithm Using Yao's Garbled Circuit on Cloud Computing." Cloud Computing (CLOUD), 2017 IEEE 10th International Conference on. IEEE, 2017.

[6] Jakobsson, Markus, and Ari Juels. "Mix and match: Secure function evaluation via ciphertexts." International Conference on the Theory and Application of Cryptology and Information Security. Springer, Berlin, Heidelberg, 2000.

[7] Brijs, Tom. "Retail market basket data set." Workshop on Frequent Itemset Mining Implementations (FIMI'03). 2003.