International Journal of Smart Home
Vol. 9, No. 4 (2015), pp. 225-238
http://dx.doi.org/10.14257/ijsh.2015.9.4.23

A New Query Integrity Verification Method with Cluster-based Data Transformation in Cloud Computing Environment

Miyoung Jang, Min Yoon and Jae-Woo Chang*

Dept. of Computer Engineering
Chonbuk National University
Jeonju, Republic of Korea
{brilliant, myoon, jwchang}@jbnu.ac.kr
**Corresponding Author*

Abstract

Due to advancement in cloud computing technology, the research on the outsourced database has been spotlighted. In database outsourcing, because the service provider might be untrusted or compromised, two issues of data security emerge: data confidentiality and data integrity. Many data transformation schemes were widely studied for preserving data confidentiality, but they are vulnerable to data leakage problem because they do not consider data distribution when encrypting original data. Meanwhile, several query authentication schemes were proposed to verity data integrity, but they suffer from transmission overhead of verification data. Motivated by these problems, we propose a privacy-aware query authentication scheme which guarantees the data confidentiality and the query result integrity of sensitive data. To solve the original data leakage problem, our clustering-based data transformation scheme is designed to select anchors based on data distribution. To verify the query result, we propose a query result authentication index that stores an encrypted signature for each anchor, which is a concatenated hash digest of cluster data. A user compares the verification information with the cluster signatures stored in the verification index. Through performance evaluation, we show that our method outperforms the existing method in terms of query processing time and verification data size.

Keywords: *Database outsourcing; database transformation technique; query result verification method; hash-based signature index*

1. Introduction

Due to the advancement in cloud computing technologies, the research on the outsourced database has been spotlighted as a new paradigm of database management system. Small-size businesses outsource their database to a service provider (SP) in order to reduce costs for managing data. The services provider maintains the outsourced database and provides query results to authorized users. However, the SP is not fully trusted since he/she may sell the data to a competitor. Furthermore, even if the SP is trusted, a malicious attacker can compromise the SP and gain unauthorized access to the data. Because the outsourced database includes sensitive information, such as personal location data, financial and medical record, a large number of database protection techniques have widely been studied.

Especially, the spatial transformation techniques have been proposed for protecting sensitive data[1-2]. Spatial transformation techniques divide the whole space into partitions and encrypt each partition by using space perturbation. M. L. Yiu *et al.*, proposed a Flexible Distance-Based Hashing (FDH) [2], which divides spatial domain into groups with random anchor objects and encrypts each group by using bitmap representation. At query time, a user encrypts a query object q to acquire relative anchor

ISSN: 1975-4094 IJSH

information. The data belonging to the nearest anchor is returned as the nearest neighbor objects.

However, the existing spatial transformation schemes have two main problems. First, the random selection of anchor objects can cause the skewed distribution of clusters. If an adversary has the background knowledge of the data distribution, the cluster information can be easily revealed. If queries are converged on the densely populated cluster, computation cost is highly increased. Second, because some of the existing techniques employ a tree-based index scheme, they can be only used for ordered plaintext. In addition, the query processing cost highly depends on the tree's depth.

On the other hand, in order to verify data correctness and completeness, query authentication should be provided in database outsourcing. For this, verification information is sent to users with query result so that the result can be verified by using data owner's signature. Previous data authentication researches [3-18] can be categorized into three classes: signature-based approaches, authenticated data structures and bucket-based authentication. First, signature-based approaches assign one signature to each data tuple and verify data integrity by comparing all the signatures of data within the results. This leads significant overheads for users both in time and space. For example, well-kwon signature scheme RSA takes 10ms in signing and 123bytes in space [19]. Second, authenticated data structure based approaches generate a tree-based data index. Each leaf node represents a data tuple, and intermediate nodes indicate a data group. The root is signed with data owner's signature as a concatenation of all children nodes' information. However, authentication data structure does not guarantee the data confidentiality, since it cannot be built on the encrypted data. In addition, tree-based index suffer from data update overhead and verification object transmission costs. Finally, J. Wang et al. [4] proposed a bucket-based authentication scheme where a bucket contains a bucket id, data range (upper-lower bound), a checksum and the number of tuples in a bucket. A checksum is similar concept to the data signature and generated by using a Hash function. The limitations of the existing bucket-based authentication scheme are as follows. First, because it generates a bucket with equal width of the data range, the original data distribution can be disclosed. Moreover bucket id is assigned as the ascending order of the data range. The existing bucket-based authentication cannot fully provide the data security. Secondly, because the existing bucket-based authentication methods only consider relational database contents, the distribution of database may not be protected. However, in case of sensitive databases, e.g. physical or mental health details, purchase records and political issues, their distribution can involve meaningful knowledge. To provide thorough protection for the sensitive database, any meaningful information including data distribution should not be revealed.

Motivated by these problems, in this paper, we propose a privacy-aware query authentication scheme which guarantees the data confidentiality and the query result integrity of sensitive data. First, to solve the original data leakage problem, we devise a bitmap-based encryption scheme by selecting anchors based on data distribution. Uniformly distributed partitions prevent the attackers from inferring the original data distribution. We also design an algebraic coding-based hash index that transforms a query to bitmap data and retrieves the anchor information efficiently. Second, to reduce the transmission overhead of verification data, we devise a query result authentication index that stores an encrypted signature for each anchor and compares the anchor signature with the verification data from the data owner. Hence, we can reduce data transfer overhead for query integrity checking while enhancing data privacy.

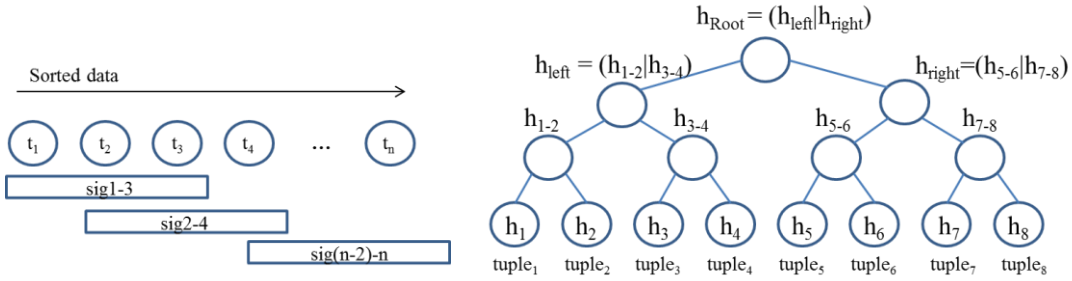
The rest of this paper is organized as follows. Section 2 presents the related work. In Section 3, our bitmap encryption based data integrity scheme is proposed. Section 4 provides an experimental evaluation on the existing and the proposed methods. Section 5 concludes this paper with brief summary and further research directions.

2. Related Work

In this section, we introduce the existing database encryption methods and query result authentication schemes, and analyze them briefly.

2.1. Data Transformation Methods

Data transformation methods are proposed in order to guarantee data confidentiality by transforming the original data domain into another one. First, A. Gutscher *et al.*, [13] proposed a parallel data transformation technique in which data points are transformed based on an axis. However, if an attacker has some known points with their transformed points, he/she can easily predict the transformation function. Secondly, M. L. Yiu. *et al.*, [11] proposed a spatial transformation technique that re-distributes the location points into the transformed space. They introduce two preliminary transformation techniques: i) Hierarchical Space Division (HSD) and ii) Error-based Transformation (ERD). Basically, HSD uses a spatial partitioning technique to redistribute the transformed data. Although this method requires a low cost for data transformation, it is vulnerable against some attack models, *e.g.*, tailored attack. For protecting against a tailored attack, they proposed ERB that utilizes a SHA-512 secure hash function for injecting noise into data. Even if ERB can protect the data against the tailored attack, it requires a high cost for data transformation. Therefore, a new enhanced HSD method (HSD*) was proposed which integrates the merits of HSD and ERB. HSD* efficiently performs a range query processing. On the other hand, they introduce a cryptographic transformation which is used for protecting data confidentiality. However, none of the existing algorithms prevents a proximity attack model that is crucial in database outsourcing. The proximity attack model means that if the approximate location of adjacent data is easy to find, an attacker can infer the location of the transformed data. In this case, the proximity attack model may cause the invasion of data privacy.



(a) Signature Chaining (b) Merkle Hash Tree

Figure 1. Signature Chaining and Authenticated Data Structures

2.2. Result Authentication Schemes

Existing data authentication researches can be categorized into signature-based approaches, authenticated data structures and bucket-based authentication. In this section, we introduce them briefly.

Signature-based approaches

Three signature-based approaches were proposed: tuple level signature [5], aggregated signature [5,6] and signature chaining [6]. First, in tuple level signature, each tuple is signed by a data owner and the integrity of tuples in query result is verified based on the signature. Secondly, in terms of aggregated signature, a signature is calculated by a concatenation of individual signatures. Compared to the

tuple level signature, aggregated signature has faster verification time than that of the tuple level signature since the signature condensing scheme is faster than signature verification of multiple tuples. Finally, a signature chain is introduced in [6] where a signature is signed on three neighboring tuples, i.e. s_i=sign(t_(i-1) |t_i | t_(i+1)). Note that t_1,…,t_N are sorted tuples in the database based on the data owner's preference or frequently performed query as shown in figure 1(a)). The tuple lever signature and aggregated signature only support data correctness. The signature chain approach guarantee both data correctness and completeness but the query processing cost is drastically increased with the large dataset.

Authenticated data structure based approaches

In authenticated data structures, Merkle Hash Tree (MH-Tree) is first studied in the field of Cyptography[7]. Tuples are organized into a tree so that one signature to the root node can guarantee the data integrity of other nodes in the tree. Merkel hash tree (MHT), illustrated in Figure 1(b), is a main-memory binary tree, where each leaf node contains the hash of a tuple, and each internal node contains the hash of the concatenation of its child nodes. To authenticate range queries, the records are sorted on the query attribute and indexed by a MH-tree. The verification process contains following steps. First, the service provider first determines the boundary records whose are neighboring the query range. Then, the paths from the root to the boundary nodes are stored. The verification object contains the paths of the boundary nodes and all visiting nodes' hash information for searching the query result nodes. This verification object is sent to the query user so that the user can rebuild the root's signature. It the reconstructed signature matches the original signature, the result is sound. Since one MHT is built on one attribute, to support authentication of multi-dimensional range queries, multiple MHTs are required to be constructed. The tree-based authentication methods suffer from data update cost, since all the hash values are needed to be updated even for a single tuple insertion. Thus, high update cost for the data owner.

Bucket-based authentication approaches

Bucket-based index was proposed by Hacigumus [16] and extensively studied in [4,17-18]. A bucket is d-dimensional rectangle and bucket-based index contains a bucket id, data range (upper-lower bound), number of tuples in the bucket and a checksum. A bucket is generated by partitioning databases with the equivalent data range (values) or with the equal number of data (count) in a bucket. A bucket checksum is a hash digest such that unique and efficient in calculation. An example of generated bucket-based index is given in Figure 2. The data range is 2000 to 6000 and each bucket is generated with the same number of data. In the example, if a query ranges from 2500 to 3500, a service provider retrieves overlapping bucket range from the index. In Table 1, bucket 1 and 2 are added to the query result. In addition, the checksum of them are also sent to the query result. Upon receiving the result with checksums, the user re-generates the checksum with results and compares them with received ones. If the reconstructed checksum matches the original, it guarantees the query result integrity.

In bucket-based index, security lies in the anonymity of tuples within the same bucket. The larger the bucket size, the less the information disclosure and more secure the bucket index is. In bucket-based authentication, the authentication index contains lower and upper range of each bucket. Thus, applying existing methods to the database is valuable to the data distribution disclosure attack.

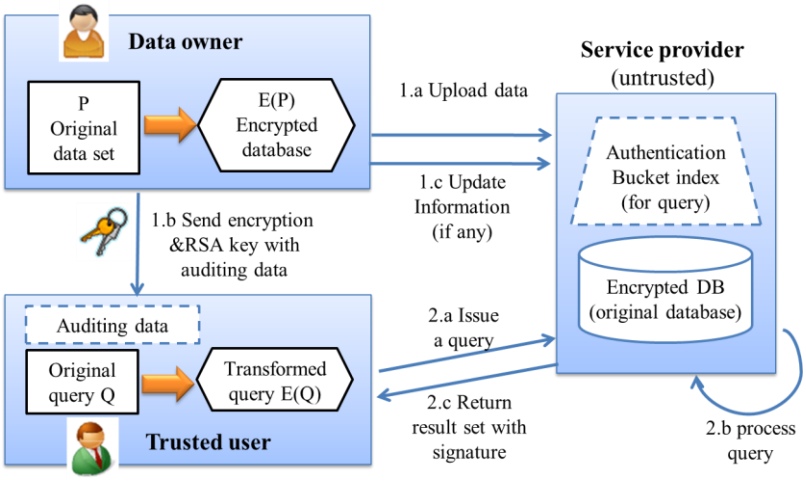


Figure 2. System Architecture and Data Flow

Table 1. Cipertext Table with Checksum

BID	*[LowerBound, UpperBound)*	*Checksum*
1	[2000,3000)	ry*nvhk&*!yie7gtkfd6
2	[3000,4000)	krrh*!ehbrwj*jyu*!yyr
3	[4000,5000)	y*bg&*!ecbrjydgtr*th
4	[5000,6000)	Jk&*!@#k4JKs*w%gh

3. Cluster-based Data Transformation and Query Result Authentication

In this section, we introduce system models and assumptions as a background. After that, we propose a bitmap-based data transformation technique and a query processing algorithm with query result authentication.

3.1. Models and Assumptions

Figure 2 depicts the overall system architecture and the data flow of the proposed algorithm. Our system model assumes that there are three main components in this architecture: a data owner, a service provider and trusted users. In database outsourcing, a data owner and a service provider play different roles. Because our security model assumes that the service provider is honest but curious, the data owner encrypts his/her databases before delegating their management to the service provider. The service provider stores the encrypted data with its index and performs queries issued from the users. The data flow for a query processing can be explained as follows: The data owner performs data encryption in the pre-processing phase for outsourcing data. Hence, an original database (P) is encrypted into a transformed datasets (E(P)). In addition, the data owner generates an authentication bucket index for data integrity-guaranteed query processing. For this, a database is grouped and signed with the data owner's private key in the authenticated data index. After the data owner encrypts his database and constructs an authenticated data index, the data owner outsources the E(P) and the index to the service provider (1-a). At the same time, the owner forwards the database encryption information and the auditing data information to trusted users so that they can utilize the query processing with the service provider as well as the query result verification (1-b).

At query processing time, a trusted user transforms its query (Q) to a transformed query (E(Q)) by applying the same encryption of outsourced database. Then, the

trusted user issues the query to the service provider (2-a). When answering a query, the service provider performs the query by retrieving the authenticated data index (2-b) and forwards the encrypted query result data with the verification data (*i.e.*, signature) (2-c). Upon receiving query results, the authenticated user generates group signature for query results by using the key from the data owner. If generated signature is identical to the signature that is sent with the query results, the user can verify the correctness and completeness of query results.

3.2. Cluster-based Data Transformation

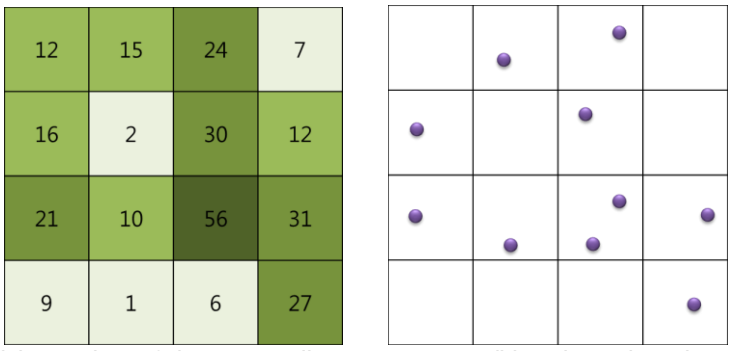
In this section, we propose a clustering-based data transformation technique for processing a range search query on encrypted data. Data encryption is performed in the pre-processing phase, when the data owner outsources his databases to the service provider. Let P be the original dataset of objects. Our clustering -based data transformation scheme is done in three detailed steps: anchor selection, Voronoi-based data clustering, and bitmap data generation. First, in anchor selection step, we generate a grid index for choosing a set of objects from the set P as anchor objects. By using a grid index, the data owner selects anchors by considering the distribution of data. Secondly, in the Voronoi-based data clustering step, the data owner generates a Voronoi diagram by using the selected anchors. Then, the original data finds its nearest anchor. After all data assigned to its nearest anchor, each anchor forms a data cluster with the data. Finally, in the data transformation step, each cluster generates a signature

Step 1: anchor selection with histogram

In this step, as shown in Figure 1, anchor selection is performed as follows. First, the original dataset P is inserted in the n*n grid index. Secondly, the algorithm counts the number of data within each grid cell, and sorts the cells in the descending order of data count. Thirdly, from the most dense grid cell, the data owner determines the number of anchors based on equation 1, and then randomly selects anchors from the cell.

$$\mathrm{A} = \frac{(\#\ of\ Anchor) \times (\#\ fo\ Data\ in\ the\ cell)}{\#\ of\ data\ in\ P} \quad (1)$$

For example, in figure 3(a), the grid cell that encloses 56 data is the largest data group so that the anchor selection is performed from the cell. When we want to select 10 anchors, the number of anchor for the cell is calculated as 2 by using the equation (1). Hence, two anchor nodes are randomly selected in grid cell 12 (Figure 3(b)).



(a) Number of data per cell (b) Selected anchors

Figure 3. Histogram-based Data Partition

Step 2: Voronoi-based data clustering

Given a dataset P, the main idea of Voronoi diagram-based data clustering is to select M objects as pivots, and then split objects of P into M clusters where each object is assigned to the closest pivot. Let the selected anchor objects from step 1 be $a_1, a_2, \ldots, a_A$. To form a data cluster, each anchor is going to be the pivot of the Voronoi cell. For all dataset, each object computes the distance to all anchors, and the object is assigned to the closest anchor a_i. Figure 4 shows an example of splitting objects into 10 partitions by employing the Voronoi diagram-based partitioning.

Step 3: data transformation

Given an object $p \in P$ and the selected anchor objects $a_1, a_2, \ldots, a_A$, we convert the original data information, e.g., a set of coordinates, into an A-length bitmap where the i-th bit of the bitmap is defined as:

$$\mathrm{BM(p)[i]} = \begin{cases} i = 0 \; if \; p \; belongs \; to \; a_i \\ \mathrm{i} = 1 \; \text{otherwise} \end{cases} \quad (2)$$

The process of generating bitmap is simple and intuitive, so we omit the detailed explanation of this step.

3.3. Query Processing with Result Authentication Index

After generating data group, we generate a private data authentication index in order to provide privacy-preserving range query processing. Each data group is signed by the data owner using Condensed-RSA [19] with data ids within the group. By this means, the private authentication index is generated without revealing the

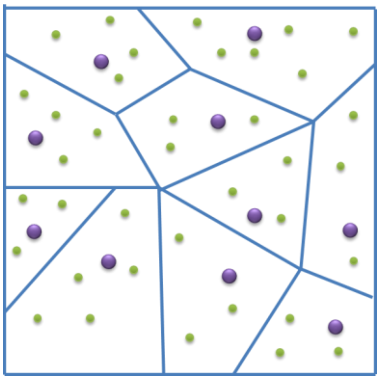


Figure 4. Voronoi-based Data Partition

partitioning information to unauthorized accesses. At query processing time, our method employs a novel technique for searching nearest data groups from a query point, in order to maximize the utility of the transformed data. In the literature, the Hamming distance measure has been employed for approximate NN search. Once the nearest anchor group is retrieved, the service provider sends the encrypted data within the group and its signature. Since the query user was given the transformation key and signature function from the data owner, he/she can decrypt the result data and generates signatures of them. If the generated signature is identical to the signature from the service provider, the client confirms that the query result is correct and genuine.

Private authentication index generation

After constructing a data partition, a signature for each small-group is generated by using Condensed-RSA. An RSA signature is computed on the hash of the input message. Let h() denote a cryptographically strong hash function (such as SHA-1) which takes a variable-length input and produces a fixed-length output. For an input message m, h(m) denotes the hash of m and a standard RSA signature on message m, and computed by using a given formulation from [19]. In this scheme, each tuple signature is generated by using RSA with its id (e.g., POI id). Condensed-RSA signature is computed as a product of individual signatures. Mykletun et al.[6] defined the Condensed-RSA as follows.

Definition 1. Condensed-RSA Signature: Given t different messages {m1, …, mt} and their corresponding signatures {σ1, … , σt } (generated by the same signer), a condensed-RSA signature is given by the product of individual signatures:

$$\sigma = h(m)^d (mod\, n)$$

$$\sigma_{(1,t)} = \prod_{i=1}^{t} \sigma_i \, (mod\, n)$$

Figure 5 shows an example of data authentication index. Each anchor has data that is represented by the same bitmap array. The original data is encrypted using AES algorithm. The group signature is generated by concatenation of each data signatures.

Anchor_id	Data_id	A	Data	Signature	Group-sig
1	1	0111111111	E(0)	RSA(1)	RSA(1|58|...|9)
	58	0111111111	E(0.1)	RSA(58)	
	6	0111111111	E(0.2)	RSA(6)	
	4	0111111111	E(0.3)	RSA(4)	
	9	0111111111	E(0.4)	RSA(9)	
2	5	1111101111	E(0.5)	RSA(5)	RSA(5|71|...|2)
	71	1111101111	E(0.6)	RSA(71)	
	11	1111101111	E(0.7)	RSA(13)	
	2	1111101111	E(0.8)	RSA(2)	
3	60	1111111001	E(0.9)	RSA(60)	RSA(60|3|14)
	3	1111111001	E(1)	RSA(3)	
	14	1111111001	E(1.1)	RSA(14)	
4	26	1110111111	E(1.2)	RSA(26)	RSA(26|8|...|43)
	8	1110111111	E(1.3)	RSA(8)	
	7	1110111111	E(1.4)	RSA(10)	
	43	1110111111	E(1.5)	RSA(43)	

Figure 5. Signature Information and Query Result Integrity Auditing Index

Query processing algorithm

At query time, the user transforms a query into bit-array BM(q), and requests the server to return an encrypted object such that its bitmap is identical to BM(q). However, there is a possibility that the query resulting area does not include any data point. In this case, we need to expand the query region and search the nearest

neighboring anchor for query processing. For this, we use the Hamming distance measure to compare the distances among bitmaps. That is to say, two regions located close together are expected to have a low Hamming distance between their bitmaps.

By using the proposed hash index and authentication index, a range query is performed as illustrated in Algorithm 1. The query processing with query result integrity auditing is performed between two parties: a query user and a service provider. First, the query user encrypts a query range with the bitmap representation so that the query range is converted to a bit array (line 1). Then, the user issues a query to the server (line 2). Upon receiving a query, the service provider retrieves the hash index and the private authentication index to return an encrypted original data set for the query (line 3-4) and their signatures (line 5-8). Finally, the user generates signatures of query results by using the RSA key sent from the data owner, and compares the signature with result signature (line 9-13). If generated signature is identical to the signature with the query results, the user can verify the correctness and completeness of query results.

Algorithm 1. Range query processing algorithm
Input : Query range [low, upper], Encryption Key CK, RSA **Output** : Query result set R, signature set S
//query user
1: encrypt query range in bitmap-array;
2: request the server for all data whose anchor_group overlaps the query region;
//service provider
3: **for** each anchor whose bitmap overlaps the query **do**
4: query result += all data in the anchor_group
5: **for** each requested group id **do**
6: search signature index
7: S += signature
8: **return** R and S to the user
//query user
9: generate signature by Condensed-RSA(R)
10: compare RSA(R) and S
11: **if** matches,
12: select the result R
13: **else** delete R
14: End

4. Experimental Evaluation

In this section, we present the extensive experimental evaluation of our scheme. For the performance analysis, we compare the performance of the proposed scheme with the existing work, proposed by S. Balpande[7]. Table 1 represents our experimental environment. We evaluate our scheme by using four different spatial datasets[20]: Uniform(100,00 points), Gaussian (100,000 points), Skewed (100,000 points), and the real dataset of Northern East America (NE) containing 119,898 point of interests (POIs), as shown in Figure 6. We compare our scheme with the existing work in terms of data insertion and bitmap generation time, range query processing time, false positive results for query processing, and query auditing time. The parameter settings are summarized in Table 2.

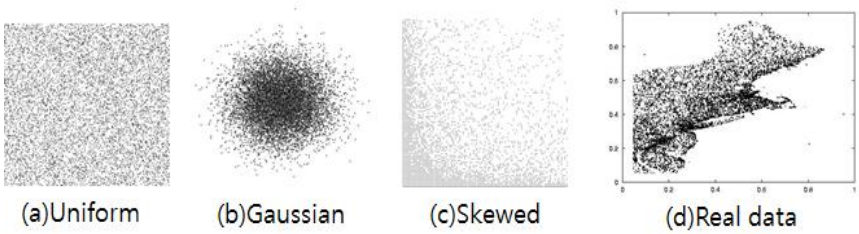


Figure 6. Data Sets

Table 2. Experimental Setup

	Specification
CPU	Intel® Core™ i3-2100 @ 3.10 GH
Memory	4GB
O/S	Windows 7 Enterprise K
Compiler	Microsoft Visual Studio 2010

Table 3. Parameter Settings

Parameter	Range
Query range (% of the whole area)	0.001, 0.005, **0.01**, 0.05, 0.1
# of anchors	100, 300, **500**, 700, 1000

Figure 7 shows the data transformation and privacy-aware query authentication index generation time with varying number of anchors. The time for data insertion and index generation can be increased as the anchors increase. When the number of anchors is set to 500 and 1000 in real dataset, the insertion time of our scheme requires 22 seconds and 89 seconds for preprocessing, respectively. Because the clusters performed merge and split algorithms to store the data into each cluster uniformly. Figure 8 describes the query processing times with varying query range. When the query range is set to 0.05%, the query processing time of our algorithm is 0.048 seconds whereas the existing scheme requires 0.6 seconds. From the result, it is proven that our algorithm outperforms the existing work up to 15 times, in terms of query processing times. This is mainly because our algorithm reduces the data transmission and verification overheads by using bit operation which easily calculates candidate anchors within query range. Also, the hash-based signature index affects to shorten the query processing time.

Figure 9 shows the number of false positive data in query result with varying query window sizes. In this figure, our algorithm reduces about 20~70% of the number of false positive data overall query ranges. This is because our algorithm generates data groups based on the efficient data partition policy, such as anchor selection with data distribution, whereas the existing scheme clusters them solely based on the data values. Therefore, our scheme is stronger against the general attack models, *e.g.*, data group estimation attacks than the existing scheme.

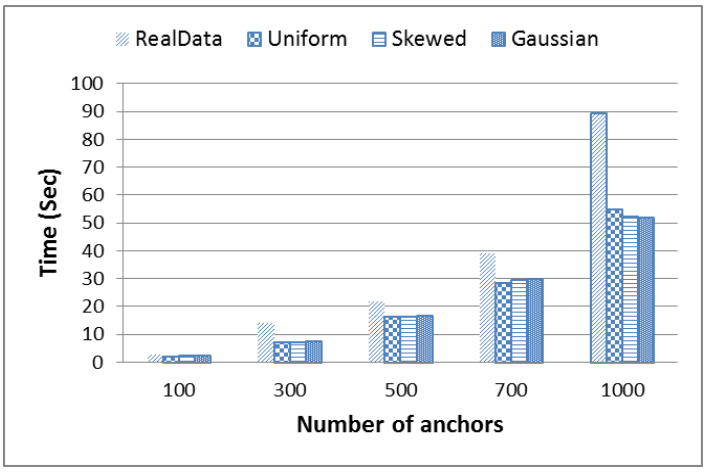


Figure 7. Bitmap-based data transformation and authentication index generation time with varying number of anchors

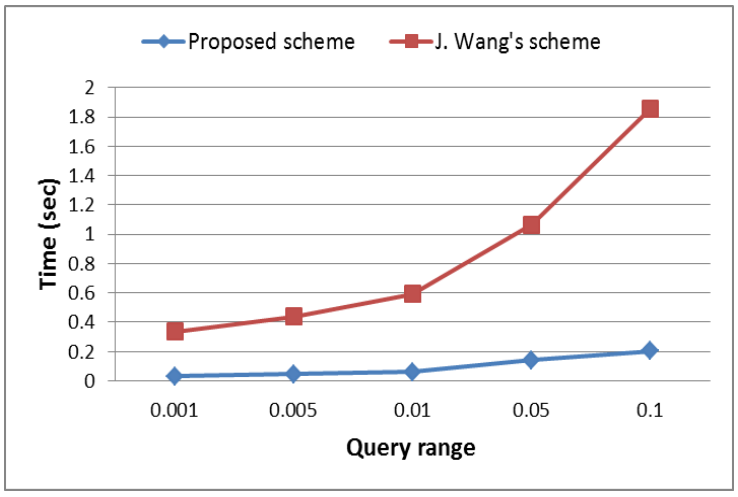


Figure 8. Query processing time with varying query range

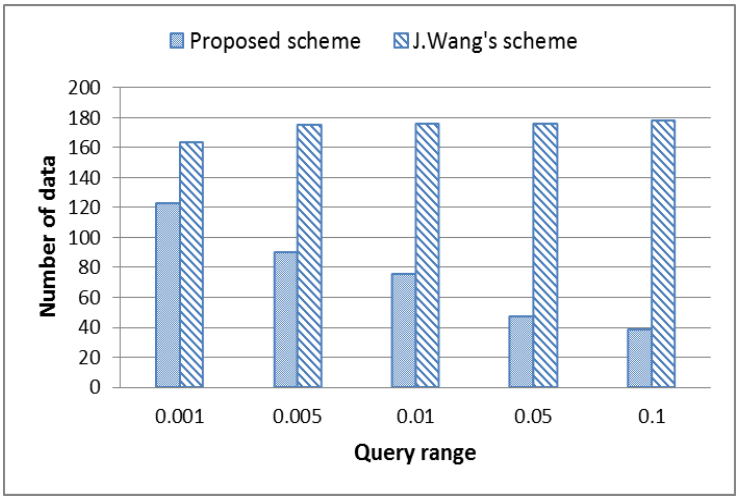


Figure 9. Number of false-positive data in query result with varying query range

5. Conclusion

In this paper, we propose a data transformation scheme that groups data with anchors and transforms them into bitmap information. To provide efficient query processing to users, we also use the Voronoi diagram to find the data overlapping the query region. Moreover, a privacy-aware query authentication that supports data correctness and completeness for users has been proposed for a range query. Through performance evaluation, it is shown that proposed method outperforms the existing method in terms of range query processing time up to 15 times while providing similar performance in returning number of false positive data.

As a future work, we will extend the proposed algorithm to support variety of query types (e.g., k-NN, skyline queries)

Acknowledgements

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2014065816). This research was also supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (grant number 2013R1A1A4A01010099).

.

References

[1] FIP Standard, Advanced Encryption Standard (AES), National Institute of Standards and Technology (NIST) **(2001)**
[2] M. L. Yiu, I. Assent, C. S. Jensen, and P. Kalnis. Outsourced similarity search on metric data assets. IEEE Trans. on Knowledge and Data Engineering. 24, 2, **(2012)** pp.338-352
[3] S. Kerr, M. S. Krkpatrick, E. Bertino. PEAR: a hardware based protocol authentication system. Proceedings of the 3rd ACM SIGSPATIAL International Workshop on Security and Privacy in GIS and LBS. **(2010)** November 3-5; San Jose, CA, USA
[4] J. Wang, X du, J. Lu and W. Lul. Bucket-based authentication for outsourced databases. Concurrency and Computation: Practical and Experience. 22, 9 **(2010)**
[5] E. Mykletun, M. Narasimha, and G. Tsudik. Signature bouquets: Immutability for aggregated /condensed signatures. European Symposium on Research in Computer Security (ESORICS) **(2004)**
[6] E. Mykletun, M. Narasimha, and G. Tsudik. Authentication and integrity in outsourced databases. Journal ACM Transactions on Storage (TOS). 2, 2 **(2006)**
[7] R.C. Merkle. A certified digital signature. Advances in Cryptology—CRYPTO'89 Proceedings. Springer New York, **(1990)** pp.218-238
[8] M. Narasimha, G. Tsudik. DSAC: integrity for outsourced databases with signature aggregation and chaining. Proceedings of the 14th ACM international conference on Information and knowledge management. ACM, **(2005)** October 31 – Nov. 5; Bremen, Germany
[9] D. Sacharidis, K. Mouratidis and D. Papadias. K-Anonymity in the Presence of External Databases. IEEE Transactions on Knowledge and Data Engineering. 22, 3 **(2010)**
[10] Y. Yang, D. Papadias, S. Papadopoulos and P. Kalnis. Authenticated Join Processing in Outsourced Databases. Processing of ACM SIGMOD International Conference on Management of data, **(2009)** June 29- July 2; Rhode Island, USA
[11] D. Liu, and S. Wang. Query encrypted databases practically. Processing of the ACM Conference on Computer and communications security. **(2012)** October 16-18; New York, NY, USA
[12] B. Hore, S. Mehrotra, M. Canim and M. Kantarcioglu. Secure multidimensional range queries over outsourced data. The International Journal on Very Large Data Bases,21, 3 **(2012)**
[13] S. Balpande, R. Shegde and L. Ragha. Data integrity and confidentiality in outsourced database. International Conference & Workshop on Recent Trends in Technology, (TCET), **(2012)**
[14] P. Devanbu, M. Gertz, C. Martel, and S. Stubblebine. Authentic data publication over the internet. Journal of Computer Security. 11, 3 **(2003)**
[15] H. Hacigumus, B. Iyer, C. Li, and S. Mehrotra. Executing SQL over encrypted data in the database service provider model. Proceeding of ACM SIGMOD International conference on Management of data, **(2002)** June 2-6; New York, NY, USA
[16] B. Hore, S. Mehrotra and G. Tsudik. A privacy-preserving index for range queries. Proceedings of the 30th international conference on Very large data bases-Volume 30. VLDB Endowment, **(2004)** August 31-September 3, Toronto, Ontario, Canada

[17] Wang J and Du X. A secure multi-dimensional partition based index in DAS. Proceeding of APWeb, **(2008)** April 26-28; Shenyang, China.
[18] C. Wang, W. Ku, "Efficient evaluation of skyline queries in wireless data broadcast environments," ACM SIGSPATIAL GIS, 2012
[19] R. Rivest, A. Shamir and L. Adleman. A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. Communications of the ACM, 21, 2 **(1978)**
[20] Y, Theodoridis, J. Silva, and M. Nascimento. On the generation of spatiotemporal datasets. Advances in Spatial Databases. Springer Berlin Heidelberg, **(1999)**

Authors

Miyoung Jang, she is a Ph.D candidate in the Chonbuk National University. She received the B.S and M.S degrees in Chonbuk National University in 2010 and 2011, respectively. Her research interests include security and privacy of MapReudce framework and cloud computing.

Min Yoon, he is a Ph.D candidate in the Chonbuk National University. He received the B.S and M.S degrees in Chonbuk National University in 2009 and 2011, respectively. His research interests include security and privacy of sensor network and database outsourcing.

Jae-woo Chang, he is a professor in the Department of Information and Technology, Chonbuk National University, Korea from 1991. He received the B.S. degrees in Computer Engineering from Seoul National University in 1984. He received the M. S. and Ph. D degrees in Computer Engineering from Korea Advanced Institute of Science and Technology (KAIST) in 1986 and 1991, respectively. During 1996–1997, he stayed in University of Minnesota for visiting scholar. And during 2003–2004, he worked for Penn State University (PSU) as a visiting professor. His research interests include sensor networks, spatial network database, context awareness and storage system.