INTERNATIONAL JOURNAL FOR TRENDS IN ENGINEERING & TECHNOLOGY
VOLUME 5 ISSUE 2 – MAY 2015 - ISSN: 2349 - 9303

Optimizing Data Confidentiality using Integrated Multi Query Services

J.Srilatha[1]
[1]Anna University, Department of CSE,
latha2k@yahoo.co.in

S.Kerthy[2]
[2]Anna University, Department of CSE,
skerthy_it@yahoo.com

Abstract — Query services have experienced terribly massive growth within past few years for that reason large usage of services need to balance outsourcing data management to Cloud service providers that provide query services to the client for data owners, therefore data owner needs data confidentiality as well as query privacy to be guaranteed attributable to disloyal behavior of cloud service provider consequently enhancing data confidentiality must not be compromise the query processed performance. It is not significant to provide slow query services as the result of security along with privacy assurance. We propose the random space perturbation data perturbation method to provide secure with kNN(k-nearest-neighbor) range query services for protecting data in the cloud and Frequency Structured R-Tree (FSR-Tree) efficient range query. Our schemes enhance data confidentiality without compromising the FSR-TREE query processing performance that also increases the user experience.

Index Terms — Confidentiality, FSR-Tree, Minimum bounding Region, Range query, Query privacy.

1 INTRODUCTION

Due to the plenty of users query services has shifted to the Cloud for their non-interrupted accessibility to reduce the infrastructure cost. With these cloud infrastructures, the service owners handily scaled up or down the service pay for hours of using the servers. Therefore the service providers lose the control over the data in the cloud, data as in intimacy and query solitude become leading concerns.

Attackers as service providers, possibly make a copy of the DB or eavesdrop user's queries are difficult to detect/ prevent in the infrastructures of cloud. During the time that new approaches are needed to maintain data confidentiality and query solitude the efficiency of query services using the clouds should also be preserved.

The existing approaches such as crypto index approach puts heavy burden on the in-house infrastructure to enhance security and privacy, order preserving encryption are open to the attacks and New Casper approach uses cloaking boxes to protect data along with queries produce efficiency of query processing including in-house workload.

In this project we propose random space perturbation (RASP) perturbation method to provide shield are able to minimize the in-house processing workload , FSR-Tree is used to provide efficient range query are able to kNN query services for protecting data in the cloud.

The basic idea is to randomly transform the multidimensional data sets with a combination of order preserving encryption, dimensionality enlarged, random noise injection, and random project, so that utility for processing range queries is preserved.

The Random Space Perturbation is designed in such some way that the queried ranges are securely transformed into polyhedral in RASP-perturbed data space, efficiently processed with help of indexing structures in the perturbed space.

The Random Space Perturbation approach preserves the topology of multidimensional range in secure transformation that permits indexing together with efficiently queries processing. The proposed service constructions are ready to minimize the in-house processed workload as low perturbation cost and high precision query results. This is imperative enabling practical cloud-based solutions.

2 SYSTEM ARCHITECTURE

The system architecture diagram proposed is shown in figure 1. For both RASP-based range query service, FSR-Tree are separated in to two groups is the loyal parties and the disloyal parties.

The loyal parties include the data owner exports the perturbed data to the cloud including authorized users submit queries. The disloyal parties include the cloud providers who host the query services furthermore protecting database. Thus, transformation range queries the client side that handles data encryption/decryption and query encryption.

The data owner, authorized users submit the original data ,queries to the proxy server; hence proxy server sends the encrypted data/queries to the service provider, therefore service provider is to index the encrypted data efficiently process encrypted queries.

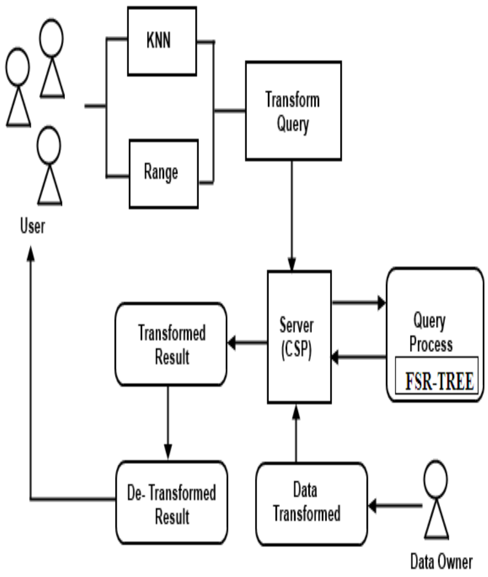


Fig.1. System Architecture

Thus transformation range queries are securely transformed to the encrypted data space efficiently processed with a two-stage processing algorithm.

3 ORDER PRESERVING ENCRYPTION

In this segment, we present the fundamental definitions of OPE and its algorithm.

3.1 Definitions of OPE

OPE transform a hypercubic query range into another hyper cubic query range. An order preserving encryption scheme E, with keys K, are applied to each dimension of x to change the dimensional distributions, to the normal distributions with each dimension's value order still preserved.

3.2 OPE Algorithm

Order Preserving Encryption is shown in figure 2, thus transforms a hypercubic to form a polyhedron. The query service have to be compelled to notice, the record within the polyhedron area that is supported by the two-stage processing algorithm is shown in OPE algorithm is given below.

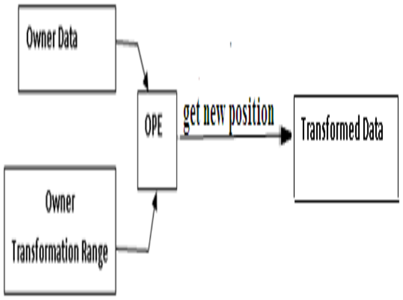


Fig.2. Order Preserving Encryption

```
Get Transformation_ Range// input from server

Get UserData //input from user

For each lettering in UserData

Start For Each

IntnewPoint;

newPoint = getNewPosition (T_Range, T_Data );

assign new position;

Add to Stack

End for Each

Transform_Data (Stack, UserData)

Return Transformed_Data;
```

Proposition 1: OPE functions transform a hypercubic query range to another hypercubic query range.

Proof - The initial range query condition consists of simple conjunction conditions, is $bi \leq Xi \leq a_i$ for each dimension. Where the order is preserved, then each simple conjunction condition is transformed as follows: Tope $(b_i) \leq$Tope $(X_i) \leq$Tope (a_i) that means the transformed range is still a hyper cubic query range.

Let Tope(x) and c_i =Tope (a_i). A simple condition $Y_i \leq c_i$ defines a half-space. With the extended dimensions $e^T = (y^T ,1, v)$, the half-space are represented as h^T $e\leq 0$, where h is a d + 2 dimensional vector with $h_i = 1$; $h_{d+1} = -c_i$, and $h_j = 0$ for $j \neq i$, d + 1. Finally, let u =Ae, according to the random space perturbation transformations. With this illustration, the original condition is equivalent to

$$h^T A^{-1} u \leq 0;$$

in the RASP-perturbed space, is still a half-space condition are transformed conditions together form a polyhedron. The query service needs to find the records in the polyhedron area that is supported by the two-stage query processing algorithm with multidimensional index tree.

4 FREQUENCY STRUCTURED R-TREE

In this segment, we present the basic definitions of FSR-Tree and its algorithm.

4.1 Definitions of FSR-Tree

The Frequency Structured R-Tree is designed in such a way that the queried ranges are securely transformed to polyhedral in the minimum bounding region efficiently processed with support of indexing structures in the perturbed space.

4.2 FSR-Tree Algorithm

Frequency Structured R-Tree algorithm gets the user a document, clean the unwanted space and extra lines, then remove the stop words from documents like verbs, Adjective then only Keyword is extracted from the document and resultant shown in R-Tree in query process is shown in fig. 3.

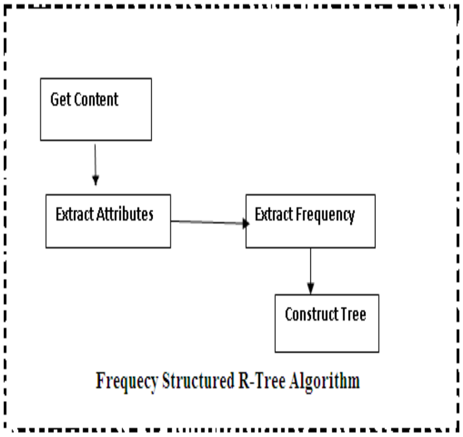


Fig.3. FSR-Tree Algorithm

FSR-Tree algorithm is good performance for queries is shown in below.

```
Get User_Document //input

CleanData=Remove_WhiteSpaces (UserDocucment)

StopwordData=RemoveStopword (CleanData)

Array keyword []=FindFrquency()

Array Attributes=ExtractAttributes (keyword [])

For each attribute in Attributes

Start For

Insert in to Tree

End For Each

Return Tree
```

5 MULTIDIMENSIONAL INDEX TREE

Most multidimensional index tree algorithms are derived from FSR-Tree as algorithm, wherever axis-aligned the minimum bounding region (MBR) is that the construction block for indexing the multidimensional data. An MBR is a rectangle and for higher dimensions, the shape of the minimum bounding region is extended to hypercube. The Frequency Structured R-tree range query algorithm compares the MBR and the queried range to find the answers are shown in fig.4. In the first stages of processing algorithm, the clients finds the MBR of The polyhedron and submit the MBR (rectangle region) and a set of secured query conditions to the server.

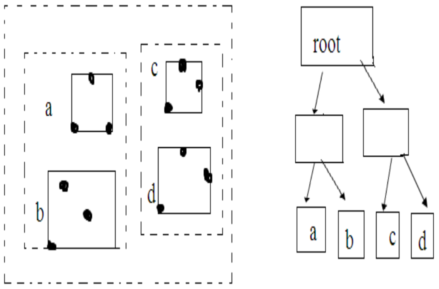


Fig.4. FSR-Tree index

Thus server uses the index tree to find the set of records enclosed by the MBR (rectangular region) and in the second stage of processing algorithm the minimum bounding region of the polyhedron will possibly enclose the entire data set which is extracted from the first stage and the second stage dataset is reduced and the return the exact range query result to the proxy server that significantly reduces the post processing cost that the proxy server needs to take.

6 RANGE QUERY PROCESSING

The original distance-based kNN query processing finds the nearest k points in the spherical range that is centered at the query point. The essential plan of our algorithm is to use square ranges, instead of spherical ranges. A square range is a hypercube that is centered at the query point and with equal-length edges. We design an algorithm similar to a binary search to efficiently find the square range.

The range query used to store data in the database and it will retrieve the records from the database where it can denote some value between upper and lower boundary. Thus range query is an important query for several data analysis tasks from simple aggregation to extra refined machine learning task.

Let Ta be a table and X_i, X_j, and X_k is being the real valued attributes in Ta, and a, b is constants.

For example, take the counting query. A typical range query looks as

Select count (*) from Ta where X_i ε [a_i; b_i] and X_j ε (a_j; b_j)

and $X_k = a_k$;

which calculates the number of records in the range defined by conditions on X_i, X_j, X_k. Range queries applied to arbitrary number of attributes and conditions of this attributes combined with conditional operators "and"/"or." we tend to decide every a part of the query condition that involves just an attribute as a simple condition. A simple condition as X_i ε [a_i; b_i], described with two half-space conditions ($X_i \leq b_i$, - $X_i \leq a)_i$. Without loss of generality, we tend to discuss a way to process half-space conditions as $X_i \leq b_i$ in this process. A slight modification extends the discussed algorithms to handle other conditions as $X_i < b_i$ and $X_i = b_i$

7 RELATED WORK

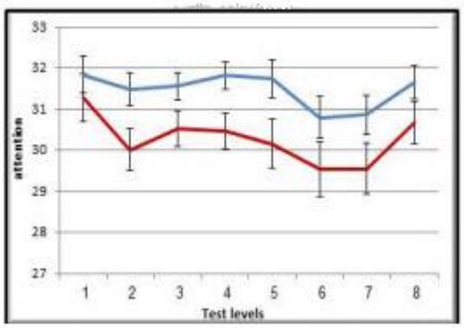
7.1 Transforming Queries in the Database

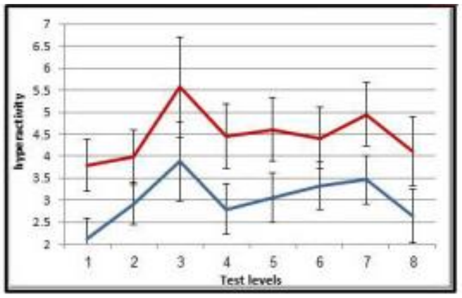
The authorized user to access only authorized portions of data while in the cloud database scenario takes more responsibilities of indexing and query processing so that transforming query in the database to secure the data in the database.

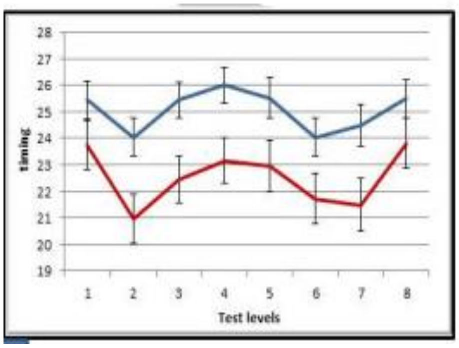
7.2 Unstructured Data Confidentiality

Our scheme enhances data confidentiality without compromising the unstructured data by using FSR-TREE algorithm for query processing performance, which also increases the user experience in the cloud.

SAMPLE RESULTS







8 CONCLUSION

To explaining the working procedure, algorithm and proving its efficiency theoretically compare with the existing system models. At the same time, despite the relatively complex proof, the actual algorithm is quite short evaluation of this concept will be done in proposed project work where we develop the range query service furthermore security of both the perturbed data including protecting queries is carefully scrutinized, for that we also conduct several sets of experiments to show the efficiency of query processed and the low cost of in-house processing. We propose random space perturbation on two aspects: (1) further improve the performance of query processing for both range queries and FSR-Tree. (2) Formally analyze the leaked query in the database and access patterns and the possible effect on both data and query confidentiality by OPE Algorithm.

9 REFERENCES

[1] R. Agrawal, J. Kiernan, R. Srikant, and Y. Xu, (2004),"Order Preserving Encryption for Numeric Data," Proc. ACM SIGMOD Int'l Conf. Management of Data .

[2] Alexandra Boldyreva, Nathan Chenette, Younho Lee and Adam O'Neill,(2004)" R. Agrawal, J. Kiernan, R. Srikant, and Y. Xu, (2004),"Order Preserving Encryption for Numeric Data," Proc. ACM SIGMOD Int'l Conf. Management of Data .

[3] K. Chen, R. Kavuluru, and S. Guo, (2011)"RASP: Efficient Multidimensional Range Query on Attack-Resilient Encrypted Databases," Proc. ACM Conf. Data and Application Security and Privacy, pp. 249-260, 2011.

[4] Elaine Shi John Bethencourt,T-H. Hubert Chan Dawn Song Adrian Perrig (2007) Multi-Dimensional Range Query over Encrypted Data .

[5] H. Hacigumus, B. Iyer, C. Li, and S. Mehrotra,(2002) "Executing SQL over Encrypted Data in the Database-Service-Provider Model," Proc. ACM SIGMOD Int'l Conf. Management of Data, Pages 216-227.

[6] B. Hore, S. Mehrotra, and G. Tsudik, (2004) "A Privacy-Preserving Index for Range Queries".

[7] M.F. Mokbel, C. yin Chow, and W.G. Aref,(2006) “The New Casper: Query Processing for Location Services without Compromising Privacy,” Proc. 32nd Int’l Conf. Very Large Databases Conf, pp. 763-774.

[8] S. Papadopoulos, S. Bakiras, and D. Papadias,(2010) “Nearest Neighbor Search with Strong Location Privacy,” Proc. Very LargeDatabasesConf.

[9] M.L. Liu, G. Ghinita, C.S. Jensen, and P. Kalnis, (2010) “Enabling Search Services on Outsourced Private Spatial Data,” The Int’l J. Very Large Data Base, vol. 19, no. 3, pp. 363-384.

[10] R. Sion, (2005),“Query Execution Assurance for Outsourced Databases,” Proc. Very Large Databases Conf. (VLDB).