IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

DUAL ENCRYPTION MECHANISM IN OUTSOURCED TRANSACTION DATABASE

DHAVAL R. PATEL
HASMUKH GOSWAMI COLLAGE OF ENGINEERING
NAVSARI, GUJARAT

Abstract: In my research paper there are used a dual security in many other research I see that there are security at client side server side some other but I observe there is some problem I there so I think that security provide at all side because some time provider or database administrator also see our data. So here I implement such security which is protect our database from client side as well as service provider side. In our system actual data can change using fact entry and using key and hashing function we can get actual data.

Index Terms **- Encryption, fake Transactions, Hash Table, RSA Algorithm, Decryption**

I. INTRODUCTION

Data-mining is the computational process of finding the large pattern from the database and this information may be important for business point of view or industrial point of view so it must be require to keep private from the others so privacy preservation in the data-mining is nothing but help to achieve data-mining goals without affecting the privacy of the data. Database outsourcing is the one disciplinary in the field of the data-mining in which more than one different data-owners sends their valuable or precious data at the third party service provider's sites by paying some predefined cost where service provider provides the different services related to the database management system like data-owner can create, delete, update or manage the database at the service-provider's site so data-owner has no burden of the data at their sites.

In the database outsourcing process data-owners or client can easily fire the query and get the output from the original database. But this information are sometimes important in the market value analysis or use for the predict something about the products so this database is very important part of the data-owners. The service-provider who provides the service is not always the trusted person so privacy preservation of the data-owner's data is required or it may be possible that any adversary brake the security and hake the original database. So privacy preservation in the database outsourcing is nothing but hide the original database from the service-provider who may be an adversary. So privacy preservation in the database outsourcing is current research topic and in this work research is done in to preserve the privacy of the data-owner's data from the any attacker or service-provider and the term is called as the "Corporate Privacy". Corporate privacy define as not only the private information of the particular one person but it is a whole organization's data which is very valuable or important data which must be require to keep private from the unauthorized person.

As the example of getting corporate or organization privacy is Super-market chain data management in which the operational transactional database from different shops of a super-market gets it services by the agent providing services on help regarding the data mining services. Considering here the case of super-market, the consumer here is the one owning the data whereas the agent providing the services is termed as the servers. Major problem with this the agent providing the services can get the information all confidential data of the agent who owns the data and if it is not properly secure then the server can access the information considering the yahoo.com store the password of the user registered to it but it uses the hash indexing to store to them if they are straight away stored into database then the server can and may see any other users account hence the privacy is in the picture and the main concern within it keeping the track.

II. METHODOLOGY

Encryption of Original TDB:-

Input: - Original TDB D *Output:* - Encrypted TDB D*

Steps:-

1. First apply 1-1 substitution method to hide original item's name, so convert original TDB D as cipher transaction database D^*.
2. After applying substitution method arrange all the cipher items in tabular form with respect to their support values (number of occurrences of item in original TDB).
3. Arrange all the items in the decreasing order of support and apply rob frugal k-grouping method to divide items in group. The grouping algorithm given below:-
   Gfrug definition:- Assume $e1, e2, \ldots, en$ is the list of cipher items in descending order of support (with respect to D), the groups created by Frugal are $\{e1, \ldots, ek\}$, $\{ek+1, \ldots, e2k\}$, and so on. The last group, if less than k in size, is merged with its previous group. They denote the grouping obtained using the above definition as Gfrug. Given a TDB D and its Frugal grouping Gfrug = ($G1$, ...,Gm), the grouping method *Rob Frugal* consists in modifying the groups of Gfrug by repeating the following operations, until no group of items is supported in D:
   1) Select the smallest $j \geq 1$ such that supp$D$(Gj) >0;
   2) find the most frequent item $i \in Gj$such that, for the least frequent item I of GjThey have: supp$D$($Gj \setminus \{i\} \cup \{i\}$) = 0; and
   3) swap I with i' in the grouping.
4. Adding fake transaction in following way
   a. Put "0" value of the noise column in which item has maximum support in the group.
   b. Find noise value corresponding to item with maximum support in group in table.
   c. Count noise value for every items using equation N(ei) =Max support of Item – Support of (ei).
   d. Discard all rows whose noise value are "0" and arrange all rows in decreasing order of their noise values.
   e. Create hash table to store value of noise or frequency of occurrence of fakely occurred in TDB with <ei, Timesi, occursi> where, ei = Num of item in TDB, times$i$represents the number of times that the fake transaction $\{e1, e2, \ldots, ei\}$ occurs in the set of fake transactions, and occ$i$is the number of times that eioccurs altogether in the future fake transactions after the transaction $\{e1, e2, \ldots, ei\}$, the ith entry of a hash table HT containing the item eihas times$i$= $N(ei) - N(ei+1)$occurs$i$= $\sum_{j=i+1}^{g} N(ej)$ where g is the number of items in the current group.
   f. Do these all steps till added all fake transaction in all group.
5. Then finally add these fake-transactions in the original database and sends to the third party service-provider.
   END

Decryption (True Pattern-Mining Task):-

Input: - Query *Output:* -True Pattern Mining Result

Steps:-

1. Data-owner fire query or give minimum threshold value of support for mining particular pattern.
2. Servers mining result from the encrypted pattern and send mined result to the data-owner.
3. Then after data-owner removes fake transaction with the help of below equation

Support(S) = Supp D*(E) – (Supp D*(E) - Supp D (E))

Where, for every item set S and its corresponding cipher item set E, we have that supp$D$(S) ≤ supp$D*$ (E).

S = support of item set in TDB

D*(E) = Encrypted TDB with fake support

D (E) = Encrypted pattern with original support

4. Finally Data-owner get true pattern from fake-transaction.

END

III. MODELING AND ANALYSIS

Example of TDB and its support table.1 (a) TDB. (b) Item support table.

TDB
Candy
Pen Candy
Candy Pen
Cap Candy
Candy Pencil
Candy Bubble
Pencil Candy
Pen

(a)

ITEM	SUPPORT
Candy	6
Pen	3
Cap	1
Bubble	1
Pencil	2

(b) (b)

Step 1 Apply substitution method in order of alphabetically of every items.

Table 2 Encrypted TDB

Items	Support
e1	1
e2	6
e3	1
e4	3
e5	2

Step 2 Arrange tables of items in decreasing order of support.

Table 3 Encrypted TDB in decreasing order of support

Items	Support
e2	6
e4	3
e5	2
e1	1
e3	1

Step 3 Do grouping using Rob frugal grouping method (Grouping with k=2).

Table 4 Grouping of encrypted TDB D*

Items	Support
e2	6
e5	2
e4	3
e1	1
e3	1

Here k=2 means in one group minimum items elements are 2 so G1= {e2, e5} and G2= {e4, e1, e3}

Step 4 Adding fake transactions

A. Find noise value corresponding maximum support of an item in particular group

Table 5 Noise table of TDB

Items	Support	Noise
e2	6	0
e5	2	3
e4	3	0
e1	1	2
e3	1	2

B. Discard the row which has noise value is "0"

Table 6 Noise table after discarded rows of "0" value

Items	Support	Noise
e5	2	3
e1	1	2
e3	1	2

C. Arrange the rows in decreasing order of noise

Table 7 Noise table of decreasing order of noise

Items	Support	Noise
e5	2	3
e1	1	2
e3	1	2

D. Create Hash table to store the value of noise or frequency of occurrence of fakely in original TDB using <Ei, Timei, Occursi> In general, the *i*th entry of a hash table HT containing the item *ei*has times*i*= *N*(*ei*) − *N*(*ei*+1)

occur*i*= $\sum_{j=i+1}^{g} N(ej)$ where g is the number of items in the current group.

Here, i=5 N(e5)=3 times3=N(e5) – N(e1)

= 3 – 2=1

Occurs of e5 = 2

Table 8 Hash Table

Hash Table
< e5,1, 2 >
< e1, 0, 2 >
< e3, 2, 0 >

IV. RESULTS AND DISCUSSION

Comparison of Grouping Execution Time for Both works:-

Here we have been done experiment on chess-dataset and retail-dataset to find the total execution time for grouping and we proved that with the help of our grouping method we get less execution time as compare to rob frugal encryption scheme which are shown in below figures:-

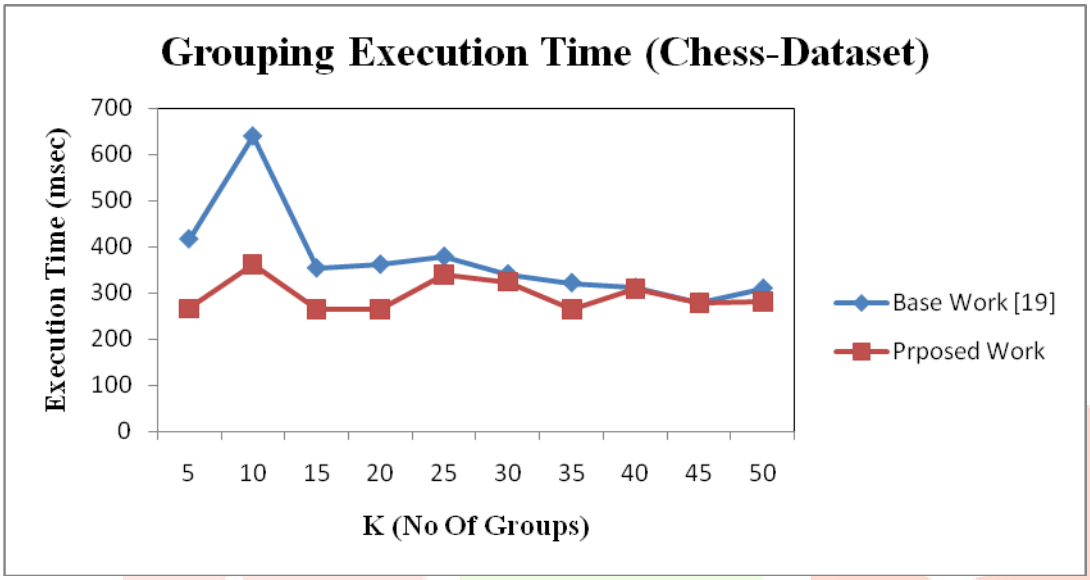

Figure 1 Comparison of grouping execution time for chess-dataset

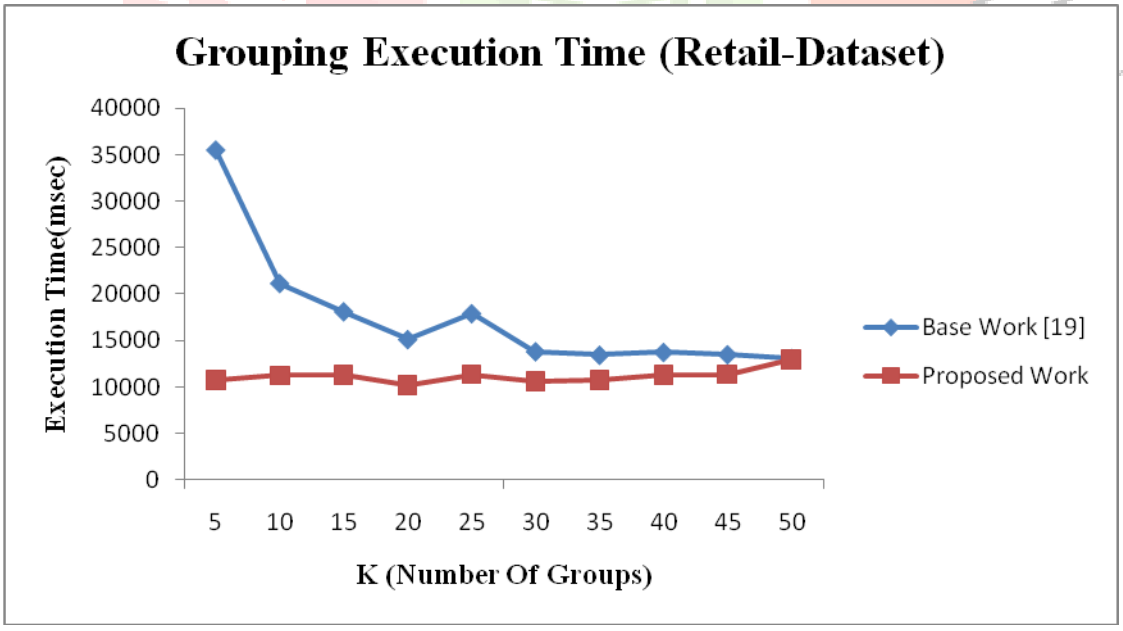

Figure 2 Comparison of grouping execution time for retail-dataset

Comparison of Fake-Transactions:-

In our proposed algorithm fake-transactions are increasing as compare to rob frugal scheme so we can say that by using our approach complexity is increases in TDBso our approach provides more security. The comparison of adding fake transactions is given below graph:-

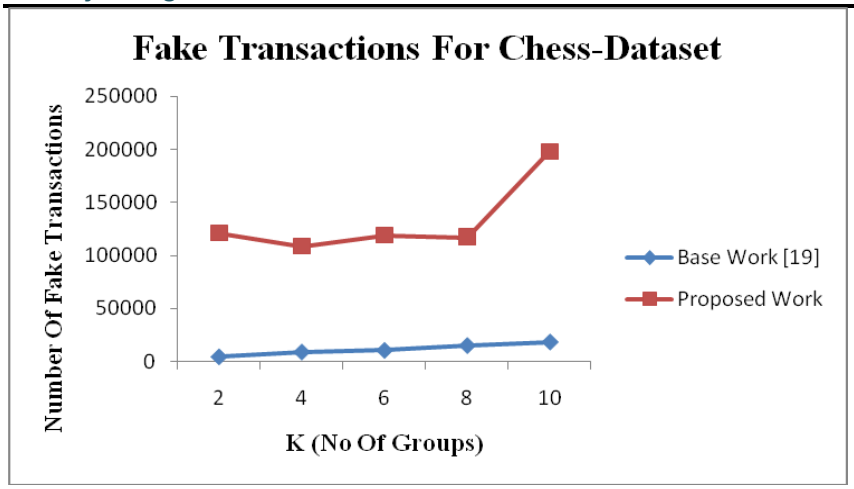

Figure 3 Comparison of adding fake-transactions for Chess-dataset

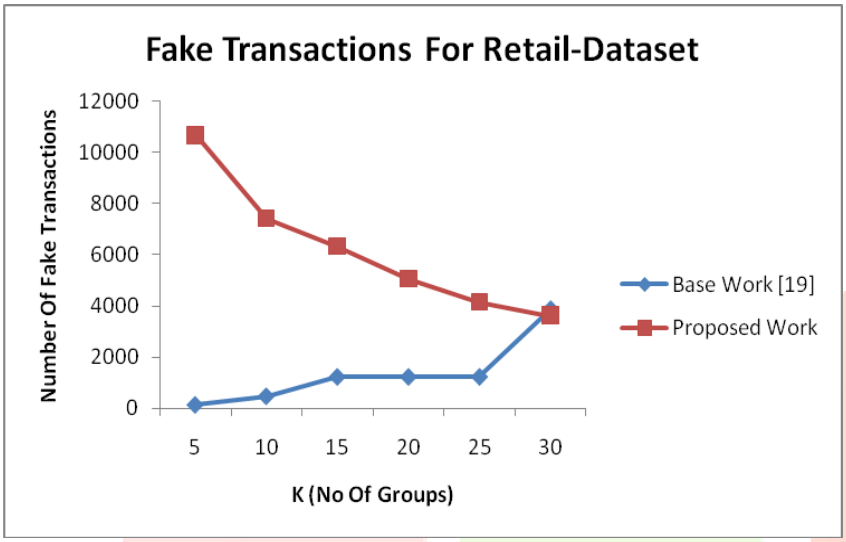

Figure 4 Comparison of adding fake-transactions for Retail-dataset

Comparison of Decryption Time:-

In our proposed approach decryption time is increase because in our approach we provide two layer securities so first time taken for decrypt the result and secondly time taken for remove fake transaction from original TDB. So double time is required for our approach but it is a trade-off of our approach where securities increases decryption time also increases. We have done experiment for both dataset for both work proposed work and base work.

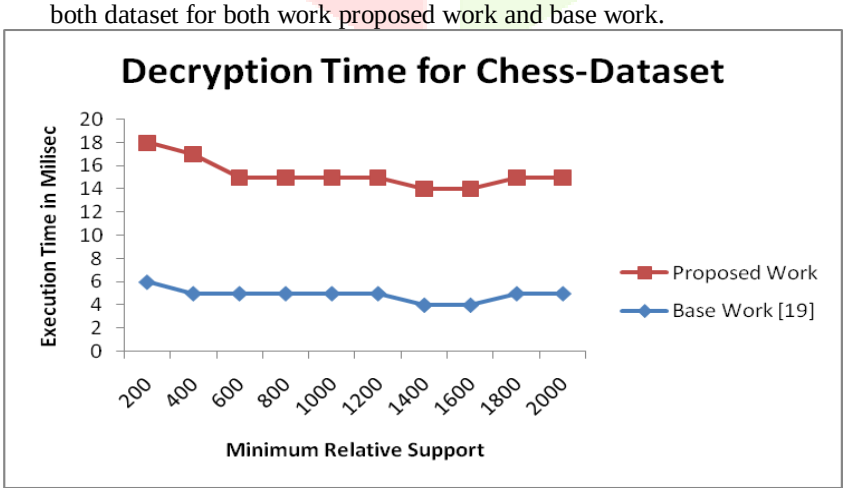

Figure 5 Comparison of Decryption-time for Chess-dataset

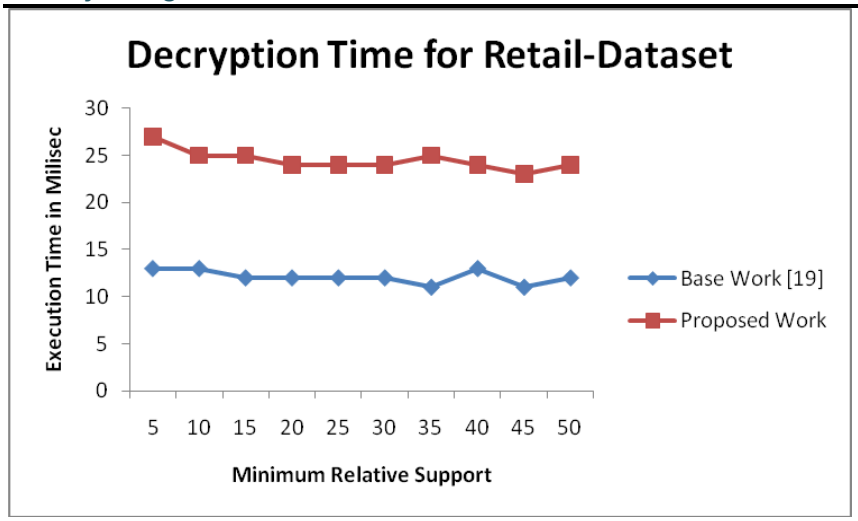

Figure 6 Comparison of Decryption-time for Retail-dataset

V. CONCLUSION

We have mainly focused on how the security applied in outsourced databases and analyzed the techniques with their usefulness for the same. In this work, we studied the problem of (corporate) privacy preservation in the database outsourcing.

We conclude that our framework is better in allows to a data owner, like a supermarket, to give its data in outsourcing to a service provider and to obtain an association rule mining service from it, without disclosing important information, deriving from the mining analysis, describing for example the customers' behaviour. So in this way we can say that by our proposed approach security is enhancing and satisfies the title of dissertation. Additionally we have proposed bidirectional encryption scheme to achieve integrity and gives perfect privacy of TDB. Moreover, our work gives extra privacy in the TDB as compare to previous approach by using two layer security.

VI. REFERENCES

[1] [1]. www.oracle.com/technetwork/topics/.../oes-refarch-dbaas 508111.pdf
[2] [2]. E. Mykletun, M. Narasimha, and G. Tsudik, Authentication and integrity in outsourced databases, In Proc. of ACM Trans. On Storage, vol. 2, 2006, pp. 107-138.
[3] [3]. M. Xie, H. Wang, J. Yin, and X. Meng, Integrity auditing of outsourced data,"VLDB 2007, pp. 782-793.
[4] [4]. Zheng-Fei Wang, Ai-Guo Tang, Implementation of Encrypted Data for Outsourced Database, In Proc. of Second International Conference on Computational Intelligence and Natural Computing (CINC), IEEE, 2010, pp. 150-153.
[5] [5]. Feifei, Marios H, George K, Dynamic Authenticated Index Structures for Outsourced Database, In Proc. of ACM SIGMOD'06. Chicago, Illinois, 2006, pp. 121-132
[6] [6]. Al Amin Hossain, Seung-Jin Lee, Eui-Nam Huh, "Shear-based Spatial Transformation to Protect Proximity Attack in Outsourced Database", ACM 2013.
[7] [7]. Mohammad Etemad and Alptekin K¨up¸c¨u, "Database Outsourcing with Hierarchical Authenticated Data Structures", SPRINGER 2014