Scientific Journal Impact Factor (SJIF) : 1.711

IJMTER

International Journal of Modern Trends in Engineering and Research

ISSN (ONLINE) : 2349 - 9745

ISSN (PRINT) : 2393 - 8161

PRIVACY PRESERVED ACCESS CONTROL FOR RDBMS

Ms. M. Kanchana[1],Mrs. R. Sasiregha[2]

[1]MPhil., Research Scholar, [2]M.Sc., MPhil., Ph.D., Assistant Professor,
Department of Computer Science
SSM College of Arts and Science, Komarapalayam, Tamilnadu, India

Abstract -Privacy preserved data mining methods are used to protect the sensitive attributes in knowledge discovery process. Privacy preservation is used to protect private data values. Anonymity is considered in the privacy preservation process. Clustering method is used to group up the records based on the relevancy. Distance or similarity measures are used to estimate the transaction relationship. Census data and medical data are referred as micro data.

User permissions are managed with dynamic data and policy management mechanism with privacy. Privacy Protection Mechanism (PPM) uses suppression and generalization of relational data to anonymize and satisfy privacy needs. Accuracy-constrained privacy-preserving access control framework is used to manage access control in relational database. The access control policies define selection predicates available to roles while the privacy requirement is to satisfy the k-anonymity or l-diversity. Imprecision bound constraint is assigned for each selection predicate. Role-based Access Control (RBAC) allows defining permissions on objects based on roles in an organization. Top Down Selection Mondrian (TDSM) algorithm is used for query workload-based anonymization. Query cuts are selected with minimum bounds in Top-Down Heuristic 1 algorithm (TDH1). The query bounds are updated as the partitions are added to the output in Top-Down Heuristic 2 algorithm (TDH2). The cost of reduced precision in the query results is used in Top-Down Heuristic 3 algorithm (TDH3). Repartitioning algorithm is used to reduce the total imprecision for the queries.

The policy based access control mechanism is enhanced to support dynamic data management technique. Data insert, delete and update operations are connected with the partition management mechanism. Cell level access control is provided with differential privacy method. Dynamic role management model is integrated with the access control policy mechanism for query predicates.

I. INTRODUCTION

The last few years have seen a remarkable increase in the construction, transfer and sharing of databases. This is mainly due to the reinforcement of their economical value and decisional interest, the latter being related in part to the progress of data mining and analysis tools. These new access capabilities induce at the same time security risks, as data records may be redistributed or modified without permission. Several examples of information leaks appear each year, even in sensitive areas like defense [1] or health care [2]. Secure access and confidentiality of data are usually achieved by means of cryptographic mechanisms. Nevertheless, once these mechanisms bypassed or more simply when the access is granted, data are no longer protected. Here comes the interest for watermarking, an a posteriori protection, that leaves access to data while maintaining them protected in terms of integrity or traceability as example. Watermarking lies in the insertion of a message or a watermark into a host document slightly perturbing host data. More precisely, the insertion process is based on the principle of controlled distortion of host data. Watermarking has been successfully applied in multimedia protection [4], but database watermarking was only introduced in 2002 by Agrawal and Kiernan. Since then, several methods have been proposed.

Depending on the embedding modulation, we can distinguish "attribute-distortion-free" methods that do not modify attributes values from "attribute-distortion-based" methods. The former are usually based on the modulation of the order of tuples within a relation. If one may consider that no data perturbation has been introduced, such a technique makes the watermark dependent on the way the database is stored, inducing constraints on the database management system. As a consequence, the application range this family of methods can be used for is limited. These methods are fragile as any reordering of tuples will eliminate the watermark.

The normal interpretation of data will not be perturbed if some alteration is carried out in the database for message insertion. Nevertheless, in order to take into account watermark imperceptibility, most recent "distortion based" schemes consider distortion constraints. For instance, the embedding process does not modify numerical attributes if some "data usability conditions", measured in terms of the mean squared error, are not respected. Shehab et al. consider additional attribute statistics constraints on attribute values and adapt the watermark amplitude by means of optimization techniques. In a recent work [11], Kamran and Farooq go one step further. Their watermarking scheme preserves classification results of a prior data-mining process. To do so, attributes are first grouped according to their importance in the mining process. Some local and global constraints are then defined in the statistical relations between attributes. The allowed perturbation of tuples for a set of attributes is obtained by means of optimization techniques. In [3], the same authors introduce the concept of "once for all" usability constraints considering the application framework where a database is sent to several recipients for different purposes. In their approach, the constraints are established in terms of numerical attributes' mean and standard deviation variations defined by the data owner and recipients. The more restrictive set of variations constitute the "once for all" constraints. They then optimize their detection based on these constraints. If a recipient has lower distortion constraints, he will receive a more distorted database leading to a more robust watermark. Lafaye et al. consider a query result approach and look at preserving the response to a priori known queries of aggregation and modulate pairs of tuples in consequence.

As exposed, the above methods focus on preserving the database statistics and do not take into account the full database semantics that should also be preserved. Semantics refer to the meaning of a piece of information. For instance, let us consider a medical database having two attributes "gender" and "diagnosis". There exist a strong semantic relation between the "gender" value "female" and the value "pregnancy" of "diagnosis". It would be incoherent to have "gender"="male". Although statistics may provide hints about the existence of such semantic links, as they evaluate the dependencies or the co-occurrences of values in the database, they do not allow directly identifying such a situation. In general, watermarked tuples must remain semantically coherent in order to: i) ensure the correct interpretation of the information without introducing impossible or unlikely records; ii) keep the introduced perturbations invisible to attackers. Indeed, an "impossible" tuple can be statistically insignificant but highly semantically detectable [8].

II. RELATED WORK

The research area on purpose based access control includes only few proposals targeting relational DBMSs. Agrawal et al. discussed high level development strategies of DBMS components charged to monitor DBMS activities based on privacy policies. Our work is aligned with the key role of purposes, and some enforcement strategies. Byun and Li propose a purpose and role based access control model for relational DBMS, where policies are enforced by means of query rewriting. Kabir and Wang propose a conditional purpose based access control model (CPBAC) which extends with conditional purposes. In [9], Kabir et al. propose the role-involved purpose-based access control (RPAC), an extended version of CPBAC that integrates concepts from RBAC. Ni et al. [10] propose a family of models called Conditional privacy-aware role based access control (P-RBAC), which extend

RBAC with concepts like purposes and obligations. Peng et al. propose a purpose based access control model that extends RBAC. The characterizing feature is the dynamic association of access purposes to user queries based on system and user attributes. Our work differs for the access control model, as none of them is action aware. In [5], we proposed a framework for the automatic generation of enforcement monitors for purpose and role based privacy policies and their integration into DBMSs. In [6] we have extended the framework to support policies also including obligations. The access control models and do not support action aware policies.

Although not directly related, other work in the area of privacy-aware access control propose logic-based approaches to the specification and enforcement of privacy policies which also consider purposes. The logic framework proposed by Datta et al. also allows checking audit logs for compliance with privacy policies. DeYoung et al. use the framework in [12] to formalize some US privacy laws. Jafari et al. [7] formalize the concept of purpose and its relation to system actions, and use model checking to evaluate system compliance wrt the policies. Other formal frameworks support privacy requirement specification and provide mechanisms to check system correctness these requirements. Different from our framework, these formalisms are privacy oriented and do not enforce policies at SQL query execution time. Finally, some work in the literature proposes language based policy specification and enforcement frameworks. These approaches target applications under development, separating the programming of functional aspects from privacy concerns. In contrast, our framework aims at complementing existing relational DBMSs with data protection capabilities.

III. ROLE-BASED ACCESS CONTROL (RBAC) TECHNIQUES

Role-Based Access Control (RBAC) is a promising access control technology for the modern computing environment. In RBAC permissions are associated with roles and users are assigned to appropriate roles thereby acquiring the roles' permissions. This greatly simplifies management. Roles are created for various job functions in an organization and users are assigned roles based on responsibilities and qualifications. Users can be easily reassigned from one role to another. Roles can be granted new permissions as new applications come on line and permissions can be revoked from roles as needed. Role-role relationships can be established to lay out broad policy objectives.

RBAC is policy neutral and flexible. The policy en-forced is a consequence of the detailed configuration of various RBAC components. RBAC allows a wide range of policies to be implemented. Administration of RBAC must be carefully controlled to ensure the policy does not drift away from its original objectives. In large systems the number of roles can be in the hundreds or thousands, users can be in the tens or hundreds of thousands and permissions in the millions. Managing these roles and users and their interrelationships is a formidable task that cannot realistically be centralized in a small team of security administrators. Decentralizing the details of RBAC administration without loosing central control over broad policy is a challenging goal for system designers and architects [11]. There is tension here between the desire for scalability through decentralization and maintenance of tight control.

Since the main advantage of RBAC is to facilitate administration, it is natural to ask how RBAC itself can be used to manage RBAC. The use of RBAC for managing RBAC will be an important factor in its long-term success. There are many components to RBAC. RBAC administration is therefore multi-faceted. In particular we can separate the issues of as- signing users to roles, assigning permissions to roles and assigning roles to roles to define a role hierarchy. These activities are all required to bring users and permissions together. In many cases, they are best done by different administrators or administrative roles. Assigning permissions to roles is typically the province of application administrators. Thus a banking application can be implemented so credit and debit operations are assigned to a teller role, whereas approval of a loan is assigned to a managerial role. Assignment of actual individuals to the teller and managerial roles is a personnel management function. Assigning roles to roles has aspects of user-role and permission-role administration. More generally, role-role relationships establish broad policy. An administrative model called ARBAC97 was recently introduced by Sandhu et al. ARBAC97 has three components: URA97 is concerned with user-role administration, PRA97 is concerned with permission-role administration and is a dual of URA97 and RRA97 deals with role-role administration.

IV. PRIVACY PRESERVED ACCESS CONTROL MODEL

In this section, three algorithms based on greedy heuristics are proposed. All three algorithms are based on kd-tree construction. Starting with the whole tuple space the nodes in the kd-tree are recursively divided till the partition size is between k and 2k. The leaf nodes of the kd-tree are the output partitions that are mapped to equivalence classes. Heuristic 1 and 2 have time complexity of $\mathrm{O}(\mathrm{d}\left|Q\right|^{2}\mathrm{n}^{2})$. Heuristic 3 is a modification over Heuristic 2 to have O(d|Q|nl gn) complexity, which is same as that of TDSM. The proposed query cut can also be used to split partitions using bottom- up (Rþ-tree) techniques.

4.1. Top-Down Heuristic 1 (TDH1)

In TDSM, the partitions are split along the median. Consider a partition that overlaps a query. If the median also falls inside the query then even after splitting the partition, the imprecision for that query will not change as both the new partitions still overlap the query as illustrated. In this heuristic, we propose to split the partition along the query cut and then choose the dimension along which the imprecision is minimum for all queries. If multiple queries overlap a partition, then the query to be used for the cut needs to be selected. The queries having imprecision greater than zero for the partition are sorted based on the imprecision bound and the query with minimum imprecision bound is selected. The intuition behind this decision is that the queries with smaller bounds have lower tolerance for error and such a partition split ensures the decrease in imprecision for the query with the smallest imprecision bound. If no feasible cut satisfying the privacy requirement is found, then the next query in the sorted list is used to check for partition split. If none of the queries allow partition split, then that partition is split along the median and the resulting partitions are added to the output after compaction.

4.2. Top-Down Heuristic 2 (TDH2)

In the Top-Down Heuristic 2 algorithm, the query bounds are updated as the partitions are added to the output. This update is carried out by subtracting the ic Q_j P_i value from the imprecision bound BQ_j of each query, for a Partition, say P_i, that is being added to the output. For example, if a partition of size k has imprecision 5 and 10 for Queries Q_1 and Q_2 with imprecision bound 100 and 200, then the bounds are changed to 95 and 190, respectively. The best results are achieved if the kd-tree traversal is depth-first. Preorder traversal for the kd-tree ensures that a given partition is recursively split till the leaf node is reached. Then, the query bounds are updated. Initially, this approach favors queries with smaller bounds. As more partitions are added to the output, all the queries are treated fairly. During the query bound update, if the imprecision bound for any query gets violated, then that query is put on low priority by replacing the query bound by the query size. The intuition behind this decision is that whatever future partition splits TDH2 makes, the query bound for this query cannot be satisfied. Hence, the focus should be on the remaining queries.

4.3. Top-Down Heuristic 3 (TDH3)

The time complexity of the TDH2 algorithm is $\mathrm{O}(\mathrm{d}\left|Q\right|^{2}\mathrm{n}^{2})$, which is not scalable for large data sets. In the Top-Down Heuristic 3 algorithm (TDH3), we modify TDH2 so that the time complexity of $\mathrm{O}(\mathrm{d}\left|Q\right|n\lg n)$ can be achieved at the cost of reduced precision in the query results. Given a partition, TDH3 checks the query cuts only for the query having the lowest imprecision bound. Also, the second constraint is that the query cuts are feasible only in the case when the size ratio of the resulting partitions is not highly skewed. We use a skew ratio of 1:99 for TDH3 as a threshold. If a query cut results in one partition having a size greater than hundred times the other, then that cut is ignored.

V. PRIVACY PRESERVED ACCESS CONTROL SCHEME

The privacy preserved access control framework is enhanced to provide incremental mining features. Data insert, delete and update operations are connected with the partition management mechanism. Cell level access control is provided with differential privacy method. Dynamic role management model is integrated with the access control policy mechanism for query predicates. The cluster based access control system is designed with incremental mining mechanism. The system also provides cell level access control mechanism. The system uses the differential privacy to protect cell level access. The system is divided into six major modules. They are data preprocess, role management, query level analysis, clustering process, incremental mining and data retrieval process. Data preprocess module is designed to perform noise elimination process. User level access permissions are assigned role management process. Query and associated data ranges are analyzed in query level analysis module. Data partitioning is performed in clustering process module. Incremental mining module is designed to modify the database transactions. Data retrieval module is designed to fetch data using query values.

Data populate process is performed to transfer textual data into relational database. Meta data provides the information about the database transactions. Data cleaning process is initiated to correct noisy transactions. Missing values are updated using aggregation based data substitution mechanism. User details and their access permissions are maintained in the role management process. Sensitive attributes selection is carried out to perform data anonymization process. Each user is assigned with different query values. The query values are used to manage the access permissions to the users. User query values are analyzed to estimate the data ranges. Data boundary for each query is estimated using Top-Down Heuristic 1 algorithm (TDH1). TDH2 algorithm is used to update the query bounds as initial partitions. Query results are verified with precision reduction level using TDH3 algorithm. Clustering process is applied to partition the transaction table with query results. TDH based partitioning algorithm is used to cluster the transaction data values. Data partitioning is performed on Anonymized data values. Data partitions are updated into the database.

Data insert, update and delete operations can be performed on the database tables. Tables are associated with the partitioned data values. Reclustering process is performed for the entire database transactions. Cluster refresh process is used to adjust the partitioned data values in incremental mining process. Data retrieval process is carried out using user query values. User query and data retrieval rate are updated into the access logs. User data access is verified with imprecision bound levels. Cell level access control is provided in the query execution process.

VI. EXPERIMENTAL ANALYSIS

The relational database access control process is carried out using the query based access control mechanism. The user privileges are managed using the Privacy Preserved Access Control (PPAC) scheme and Enhanced Privacy Preserved Access Control (EPPAC) schemes. The Privacy Preserved Access Control (PPAC) scheme is designed with the top down heuristics mechanism and the partitioning mechanism. The K-Anonymity model is used for the data anonymization process. The Enhanced Privacy Preserved Access Control (EPPAC) scheme is constructed with the incremental mining mechanism and differential privacy features. The differential privacy scheme is used to protect the cell level privacy. The relation database privacy preserved access control scheme is tested under the Census data environment. The data values are downloaded from the University of California, Irwin (UCI) machine learning repository. The system is tested with three performance matrices. They are imprecision bound, utility rate and computational complexity. The system is tested with different data intervals.

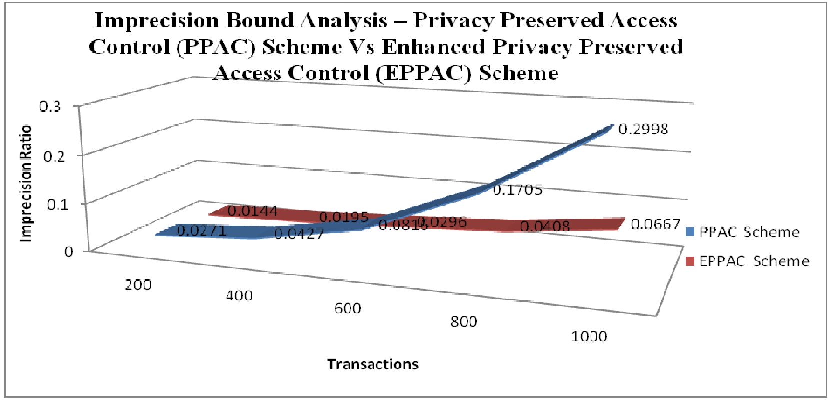

Figure No: 6.1. Imprecision Bound Analysis – Privacy Preserved Access Control (PPAC) Scheme Vs Enhanced Privacy Preserved Access Control Scheme (EPPAC)

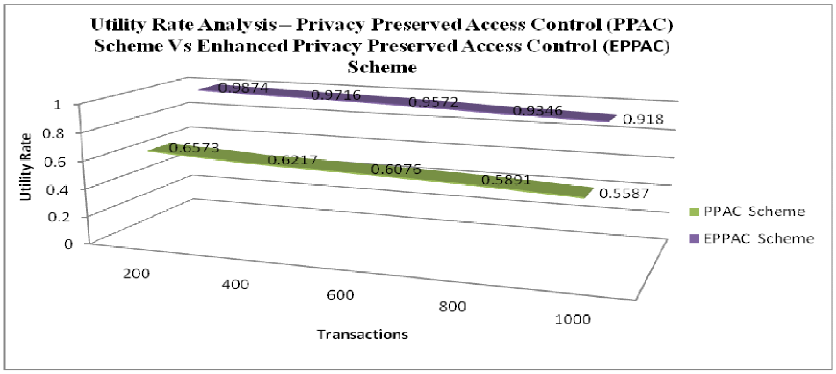

Figure No: 6.2. Utility Rate Analysis – Privacy Preserved Access Control (PPAC) Scheme Vs Enhanced Privacy Preserved Access Control (EPPAC) Scheme

The imprecision bound analysis mechanism verifies the query retrieval accuracy levels. Figure 6.1. shows the imprecision analysis between the Privacy Preserved Access Control (PPAC) scheme and Enhanced Privacy Preserved Access Control (EPPAC) scheme. The analysis results show that the EPPAC scheme reduces the imprecision level 45% than the PPAC scheme. The utility rate analysis shows the data usage ratio under the databases. The data utility rate analysis shown in figure 6.2. The analysis results show that the EPPAC scheme increases the data usage rate 35% than the PPAC scheme. The computational complexity analysis verifies the process time required for the relation database query and partitioning tasks. Figure 6.3 shows the computational complexity analysis between the PPAC and EPPAC models. The EPPAC scheme reduces the process time 30% than the PPAC scheme.

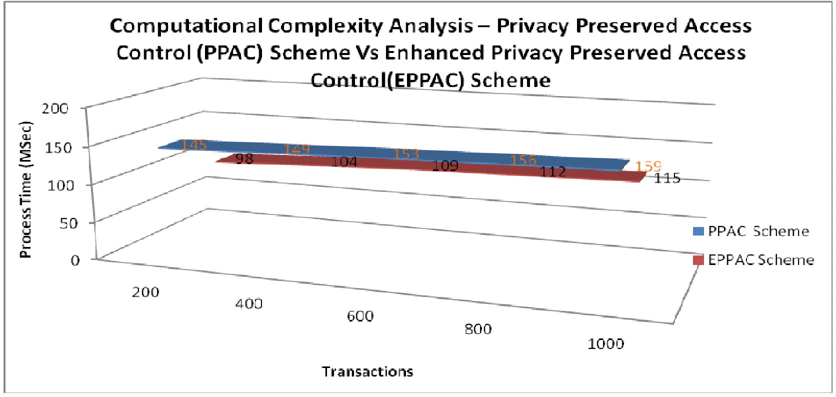

Figure No: 6.3. Computational Complexity Analysis – Privacy Preserved Access Control (PPAC) Scheme Vs Enhanced Privacy Preserved Access Control (EPPAC) Scheme

VII. CONCLUSION AND FUTURE WORK

The Access Control Model (ACM) is used to manage the database access privileges. Role Based Access Control (RBAC) scheme is used to control the user access with query based permissions. The K-Anonymity mechanism is used for the privacy preservation tasks. The top down heuristics and partitioning methods are used to provide the privacy preserved data access in relational database environment. The differential privacy mechanism is used to protect the cell privacy. The system supports incremental mining on privacy preserved access control environment. The system can be enhanced with the following features.

- The query based access control mechanism can be improved to perform privacy preserved rule mining and classification tasks.
- The system can be adapted to handle malicious query requests.
- The relational database access control model can be enhanced to access data through Internet.
- The query model can be improved to carry out encrypted storage management and query requests.

REFERENCES

[1] S. Rogers. WikiLeaks Embassy Cables: Download the Key Data and See How it Breaks Down. [Online]. Available: http://www.theguardian.com/news/datablog/2010/nov/29/wikileaks-cables-data, accessed Nov. 21, 2013.

[2] M. McNickle. Top 10 Data Security Breaches in 2012. Healthcare Finance News. [Online]. Available: http://www.healthcarefinancenews.com/news/top-10-data-security-breaches-2012, accessed Nov. 21, 2013.

[3] M. Kamran, S. Suhail and M. Farooq, "A robust, distortion minimizing technique for watermarking relational databases using once-for-all usability constraints,", IEEE Trans. Knowl. Data Eng., vol. 25, no. 12, pp. 2694–2707, Dec. 2013.

[4] D. Rosiyadi, S.-J. Horng, P. Fan, X. Wang, M. K. Khan and Y. Pan, "Copyright protection for e-government document images," IEEE Multimedia, vol. 19, no. 3, pp. 62–73, Jul./Sep. 2012.

[5] P. Colombo and E. Ferrari, "Enforcement of purpose based access control within relational database management systems," IEEE Trans. Knowl. Data Eng., vol. 26, no. 11, pp. 2703–2716, Nov. 2014.

[6] P. Colombo and E. Ferrari, "Enforcing obligations within relational database management systems," IEEE Trans. Dependable Secure Comput., vol. 11, no. 4, pp. 318–331, Jul./Aug. 2014.

[7] M. Jafari, P. W. Fong, R. Safavi-Naini, K. Barker and N. P. Sheppard, "Towards defining semantic foundations for purpose-based privacy policies," in Proc. 1st ACM Conf. Data Appl. Security Privacy, 2011, pp. 213–224.

[8] J. Franco-Contreras et al., "Data quality evaluation in medical database watermarking," Studies Health Technol. Inform., vol. 210, pp. 276–280, May 2015.

[9] M. Kabir, H. Wang and E. Bertino, "A role-involved conditional purpose based access control model," in E-Government, E-Services and Global Processes, series IFIP Advances in Information and Communication Technology, vol. 334, Springer, 2010.

[10] Q. Ni, E. Bertino, J. Lobo, C. Brodie, C.-M. Karat, J. Karat and A. Trombeta, "Privacy-aware role-based access control," ACM Trans. Inform. Syst. Security, vol. 13, no. 3, p. 24, 2010.

[11] M. Kamran and M. Farooq, "A formal usability constraints model for watermarking of outsourced datasets," IEEE Trans. Inf. Forensics Security, vol. 8, no. 6, pp. 1061–1072, Jun. 2013.

[12] A. Datta, J. Blocki, N. Christin, D. Kaynar and A. Sinha, "Understanding and protecting privacy: Formal semantics and principled audit mechanisms," in Proc. 7th Int. Conf. Inform. Syst. Security, 2011, pp. 1–27.