IJISET - International Journal of Innovative Science, Engineering & Technology, Vol.2 Issue 3, March 2015.
www.ijiset.com
ISSN 2348 – 7968

Detection of Guilt Model for Data Leakage

Kavitha S[1], Jackulin DuraiRani A[2], Sowmiyaa P[3], Gayathri R Krishna[4]
[1,2,3,4] Department of Computer Science and Engineering,
Dr. N. G. P. Institute of Technology, Coimbatore, India

Abstract
Nowadays the data are distributed for the business purposes through trusted agents. There is a chance for leakage of data and can be identified in unauthorized places. The probability of the leaked information came from one or additional agents should be assessed by the distributor, as hostile having been severally gathered by different means. In this paper we propose information allocation ways (across the agents) that improve the likelihood of distinguishing leakages. These ways do not think about alterations of the free information (e.g., watermarks). In some cases, we will additionally inject "realistic but fake" information records to further improve our possibilities of detection escape and distinguishing the problem.

Keywords: *data sharing, data leakage, guilty agent, fake objects, data allocation.*

1. Introduction

In this business world, the data sharing and publishing becomes the most important thing to develop the relationship between the business communities. The data sharing is done when the organizations want to create partnerships or for the data analysis purpose. For these purposes the customer's data of the organizations are distributed by the third party agents as data outsourcing. The terms distributor means the owner of the record, if the data is given to the unauthorized person then it is said to be the leakage and the person from whom the data is distributed to other persons or organizations are said to be the agents mostly third parties. Let us consider the data that cannot be perturbed because perturbation data are conversion of original data by adding random noise which can be used in data publishing that uses less sensitive data for data analysis. On data sharing the data should be confidential which should provide exact data values of the customers to improve their business services. The best example for data sharing is the data outsourcing by the companies where the work is completed by some other persons instead of the data owner's organization.

Traditionally watermarking embeds the data in all the copies that are distributed. If the copy is identified in unauthorized person's place then the leaker is identified. But the disadvantage of using watermark is it can be destroyed. Hence in this paper, we develop a model for identifying the agent's guilt using algorithms to distribute the data to agents that helps to improve the identification of leaker.

2. Related Work

This detection of guilty agents is related to the data provenance problem where the identification of guilty agents is done by tracing the lineage of objects [1] and the tracing of lineage objects from data warehouses is defined in [2], [3]. Here we formulated the objects and sets generally to simplify lineage tracing and to avoid data transformation. The data allocation strategies are relevant to the usage of watermarks. Watermarks hide the data inside images [4], video [5] and audio [6]. The insertion of watermarks to the relational data also described in [7]-[10]. The proposed approach is similar to watermarks but the watermark can be modified and if not, it cannot be inserted. So it is not applicable to distributed data. The other methods to avoid unauthorized access are proposed in [11], [12] but these methods are restrictive and also cannot be used to identify the guilty agents.

3. Problem Setup

Let T= $\{t_1, t_2 \ldots t_n\}$ be the objects that are owned by the distributor which can be shared to the agents A_1, $A_2, \ldots, A_n$. but it should not be leaked to others by the third parties. The agent can receive the object either by sample request or explicit request. Sample request provides a subset of objects to the agent and upon explicit request agents receive all the objects. If the distributor identifies the object T in an unauthorized person's laptop or website then the object is leaked by the third parties called target. We have to identify the agents A_i who have leaked the data.

4. Data Allocation

The data allocation problem is focused in order to give the data intelligently to the agents by the distributor to identify the guilty agents. Here we address this based on the requests and the "fake objects" insertion. The sample and explicit requests are addressed by inserting the fake objects to T which are look like real objects and distributed to the agents. Let F be the fake objects, S be the sample requests and E be the explicit requests.

ISSN 2348 – 7968

Let us assume that in the instances of E problem, explicit requests are made by all the agents whereas in the instances of S, sample requests are made by all the agents represented in fig. 1. We provide an idea to handle the mixed requests which is not elaborated. Let us assume that two agents with their requests respectively R_1=E (T,cond$_1$) and R_2=S (T`,1), where T`=E (T,cond$_2$). Where cond$_1$ is "state=CA" (state field for object). Suppose agent A_2 has the same condition cond$_2$=cond$_1$, an equivalent problem is created with sample data requests on set T`. The problem is with the distribution of the objects to two agents $R_1$=S (T`,|T`|) and $R_2$=S (T`,1). Suppose A_2 uses condition "state=NY," and two different problems are solved for sets T` and T-T`. For each problem, only one agent is there. At last when conditions are partially overlapped, $R_1 \cap T` \neq \varphi$, but $R_1 \neq T`$, so that three different problems are solved for sets R_1- T`,$R_1 \cap$ T`, and T`- R_1.

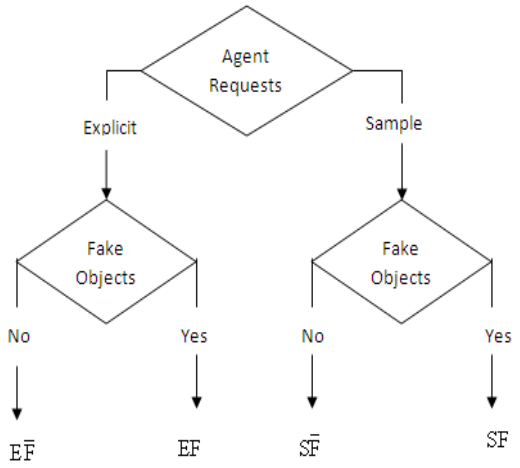

Fig. 1. Data Allocation instances

5. Optimization

There is one constraint and objective for the data allocation to the agents. The constraint is that the distributor has to provide the requested objects by the agents that satisfy the necessary conditions in order to satisfy the requests of all agents. The objective of the distributor is to identify the agent who has leaked the portion or complete data. The distributor will not deny the request of the agents [13] and also does not provide perturbed data [7]. The relaxation of the constraint is only adding the fake objects. Our proposed ideal objective is the detection can be done only if no object is distributed to the agents by the distributor [14].Instead of that our objective is to increase the probability of identifying the guilty agent. The probability of identifying the guilt agent is calculated with the differences in the probability of the requests for objects of S defined in eq. 1.

$$\Delta(i, j) = p_r\{G_i| R_i\} - p_r\{G_i| R_i\} \quad i, j = 1,\ldots\ldots,n. \quad (1)$$

The value of Δ has positive values where R_i consists of all leaked objects and agent A_i can be the leaked agent. For any agent A_j, if $\Delta(i, j)$ is positive, then R_j does not consists all data S. if $R_i \subseteq R_j$ then both agents A_i and A_j are suspected equally guilty by the agents because both of them should received all the leaked objects. When Δ (i, j) is larger, then agent A_i can be easily identified as the leaking agent. So the distributed data should have large Δ value.

6. Allocation Strategies

The allocation strategies are described to solve the different instances represented in fig. 1. We provide the solutions for optimization rather when the case is inefficient. The allocation strategies are based on the requests of agents either explicit or sample requests.

6.1 Explicit Requests

The explicit data requests are used for both object selection and agent selection. In case of object selection, if there is a problem of class EF, the fake objects are not allowed to add by the distributor to the distributed data. Hence only the agent's data requests define the allocation of data and optimization is not necessary. The agent selection is done by e-optimal. This allocates one fake object per agent. This approach makes optimal distribution of data to the agent based on the priority of their request in order to satisfy the objective.

6.2 Sample Requests

The sample data requests are used only for object selection. Here the data request does not define the data sharing explicitly. The data allocation is done to multiple agents by the distributor. So the data is allocated using s-max by the distributor where the object is allocated in order to minimize the overlapping of objects so that it increase the probability chances of finding the guilty agent.

ISSN 2348 – 7968

7. Guilt Analysis

The interaction of model parameters is evaluated by considering that the target contains all the objects of the distributor. i.e., T⊆S. Let T consists of 16 objects: Let 16 objects are given to A_1 and only 8 to A_2. The probabilities are calculated as $p_r\{G_1| S\}$ and $p_r\{G_2| S\}$ if probability in the range [0, 1]. When probability approaches to 0, then all 16 values are guessed by the target. When it approaches to 1 it is determined as individual guilt. If the probability value increases the probability of A_2 is decreased significantly and eight objects of A2 are given to A_1. So it is harder to say only A_2 has leaked the data.

Let us consider two agents again, where all the objects are received by one agent. i.e., T ⊆ S data and the agent receive some fraction of data. A function $|R_2 \cap S|/|S|$ is used to calculate the probability value. If probability value is less than 0.2 then A_1 has all 16. Here also the probability of guilt decreases when the probability value increases but it is easier to identify the leakage of data and the guilty agent.

8. Experimental Settings

This section explains the experimental evaluation and its results.

8.1 Experimental Setup

The client server GUI application represented in fig.2 is developed using .Net Framework where the distributor can manage the agents list and for the data distribution to the agents. The probability of data leakage and guilty agents can be identified by the distributor when it is leaked. The agents once received the objects they distribute only to the authorized persons those are registered with distributors. If not the data should be destroyed to the agent upon data transformation to the others for security purposes.

Fig. 2. Distributor login for data distribution (GUI)

8.2 Experimental Results

The data allocation is done with allocation strategies based on the agent's request to the distributor. The distributor sends the object with fake objects which exactly looks like real objects. Once the data is distributed to unauthorized persons by the agents then the count will be increased in the distributor application. From that the guilt agent probability is calculated and identified in fig 3.

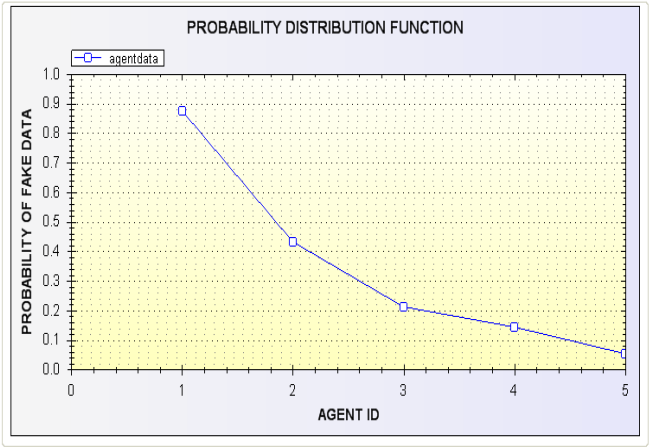

Fig 3. Probability Distribution for Data Leakage by Agents

9. Conclusion and Future work

In this business world, data sharing is an important thing but the security is an issue to maintain the confidential data when it is distributed through agents. We cannot trust the agents completely and formally watermark is used. The sensitive data is generally transferred using watermark which can be destroyed. We proposed the method by inserting the fake objects that can be distributed to identify the guilty agents based on the overlapping of data. The probability of identification can be improved with our proposed method. In future the leakage scenarios of guilty agents can be studied.

References

[1] P. Buneman, S. Khanna, and W.C. Tan, "Why and Where: A Characterization of Data Provenance," Proc. Eighth Int'l Conf. Database Theory (ICDT '01), J.V. den Bussche and V. Vianu, eds., pp. 316-330, Jan. 2001.

[2] P. Buneman and W.-C. Tan, "Provenance in Databases," Proc. ACM SIGMOD, pp. 1171-1173, 2007.

ISSN 2348 – 7968

[3] Y. Cui and J. Widom, "Lineage Tracing for General Data Warehouse Transformations," The VLDB J., vol. 12, pp. 41-58, 2003.
[4] J.J.K.O. Ruanaidh, W.J. Dowling, and F.M. Boland, "Watermarking Digital Images for Copyright Protection," IEE Proc. Vision, Signal and Image Processing, vol. 143, No. 4, pp. 250-256, 1996.
[5] F. Hartung and B. Girod, "Watermarking of Uncompressed and Compressed Video," Signal Processing, vol. 66, no. 3, pp. 283-301, 1998.
[6] S. Czerwinski, R. Fromm, and T. Hodes, "Digital Music Distribution and Audio Watermarking,"http://www.scientificcommons.org/43025658, 2007.
[7] R. Agrawal and J. Kiernan, "Watermarking Relational Databases," Proc. 28th Int'l Conf. Very Large Data Bases (VLDB '02), VLDB Endowment, pp. 155-166, 2002.
[8] R. Sion, M. Atallah, and S. Prabhakar, "Rights Protection for Relational Data," Proc. ACM SIGMOD, pp. 98-109, 2003.
[9] Y. Li, V. Swarup, and S. Jajodia, "Fingerprinting Relational Databases: Schemes and Specialties," IEEE Trans. Dependable and Secure Computing, Vol. 2, No. 1, pp. 34-45, Jan.-Mar. 2005.
[10] F. Guo, J. Wang, Z. Zhang, X. Ye, and D. Li, "An Improved Algorithm to Watermark Numeric Relational Data," Information Security Applications, pp. 138-149, Springer, 2006.
[11] S. Jajodia, P. Samarati, M.L. Sapino, and V.S. Subrahmanian, "Flexible Support for Multiple Access Control Policies," ACM Trans. Database Systems, vol. 26, no. 2, pp. 214-260, 2001.
[12] P. Bonatti, S.D.C. di Vimercati, and P. Samarati, "An Algebra for Composing Access Control Policies," ACM Trans. Information and System Security, vol. 5, no. 1, pp. 1-35, 2002.
[13] S.U. Nabar, B. Marthi, K. Kenthapadi, N. Mishra, and R. Motwani, "Towards Robustness in Query Auditing," Proc. 32nd Int'l Conf. Very Large Data Bases (VLDB '06), VLDB Endowment, pp. 151-162, 2006.
[14] B. Mungamuru and H. Garcia-Molina, "Privacy, Preservation and Performance: The 3 P's of Distributed Data Management," technical report, Stanford Univ., 2008.