ISSN 2348 – 9928
IJAICT Volume 3, Issue 3, March 2016 Doi:01.0401/ijaict.2014.05.09 Published on 05 (3) 2016

DATA SECURITY DESCRIPTION OF ENHANCED DATA MINING ANALYSIS USING SYMMETRIC INFERENCE MODEL

Ms. A. Kannagi
Associate Professor,
Department of Computer Science and Engineering,
Pavai College of Technology,
Namakkal, Tamilnadu, India

Mr. M. Muthuraja
Assistant Professor,
Department of Computer Science and Engineering,
Pavai College of Technology,
Namakkal, Tamilnadu, India

Abstract— In a data distribution scenario the sensitive data given to agents can be leaked in some cases and can be found in unauthorized places. Our aim is to detect when the distributor's sensitive data have been leaked by agents and to identify the agent who leaked the data. We consider the addition of fake objects to the distributed set which do not correspond to real entities but appear realistic to the agents. The distributor must assess the likelihood that the leaked data came from one or more agents, as opposed to having been independently gathered by other means. We also present data allocation strategies and algorithms for distributing objects to agents, in a way that improves our chances of identifying a leaker. Our main idea is to prevent the agents from comparing their data with one another to identify fake objects. A Symmetric Inference Model (SIM) is used here to find out the probability of identifying dependency among the data distributed to various agents. Using this technique a symmetric inference graph (SIG) is drawn denoting the links among data sets.

Keywords— Symmetric Inference Model, Symmetric Inference Graph, Sensitive Data.

I. INTRODUCTION

Secure data allocation a model is developed for evaluating inference based on the past fake object allocation sequences. Semantic Inference Model (SIM) consists of data dependency, relational database schema and domain-specific semantic knowledge [1]. Following example explain how inference affects data leakage. In this database, the attribute City does not functionally determine attribute Salary, as both Azar and Benjamin live in Pune they earn different salaries. As a result, schema based inference detection systems do not report any inference threat in this database [2].

Table 1: Representation of Database 1

Name	Salary	City
Benjamin	45 K	Pune
Azar	50 K	Pune
Jackson	60 K	Chennai

Table 2: Representation of Database2

City	Salary
Coimbatore	45K
Coimbatore	50K
Chennai	60K

In this database, the attribute City does not functionally determine attribute Salary, as both Azar and Benjamin live in Pune they earn different salaries [3]. As a result, schema based inference detection systems do not report any inference threat in this database. However, if a user knows that Jackson is the only employee who lives in Chennai, the user can infer the salary of Jackson by querying the database to find the salary of the employee who lives in Chennai in the second table. This example illustrates that simply examining the database schema to detect inference is not sufficient, and taking the data in the database into consideration can lead to the detection of more inferences [4][5].

We accessing them when fake objects created to any agent, probability will be calculated and on each fake object that probability goes on increasing. If probability is below threshold then fake object is allocated to that agent but if probability exceeds specified threshold, then that agent is not getting fake objects. This is the case for single agent [6].

In the same way for multi agent environment, when different agent tries to collaborate to increase probability of accessing information, then probability of the agent will goes on increasing whose information other agents are accessing. Here basically we have tried to implement inference controlling mechanism for creating fake object for all agents and their probability will be calculated [7][8].

So, a Semantic Inference Model (SIM) representing them as probabilistic inference channels to access any data from the system. Probability is calculated as conditional probability,

ISSN 2348 – 9928
Doi:01.0401/ijaict.2014.05.09 Published on 05 (3) 2016

given as Pi|j = Pr (B=bi|A=aj). It represents the occurrence of A and b and Co occurrences of A and B. Also it represents the dependency from B to A. Initially probability and data probability is set to 0.0. When data is allocated to first agent, probability is calculated as number of fake objects is allocated to specific agent within number of data is divided by total number of times agent has been allocated data within number of fake objects [9][10].

This probability will be stored in log. Next time when same agent is allocated for objects, probability will be checked from log. If it is below threshold objects can be allocated to the same agent, otherwise the other type of fake object is created and allocated to that agent [11][12].

Agent: string
Access Data from Agent: integer
Probability: integer,
Total count: integer
Count: integer

Probability= (count/TotalCount)+previous(Probability)
TotalCount = number of times objects allocated
Count = number of time fake objects allocated

Table 3, gives an idea about probability calculation. Two variables are maintained for it Datacount1 and Datacount2. Initially these two variables are set to 0. First time probability is calculated as 0.1. Difference between above two examples is that, in first case agent is accessing data from same table and in second case agent is accessing data from two different tables. Depending on that count will be calculated differently.

Table 3: Probability calculation (Multiple agents)

Agent	Accessing data from other agent	Data probability	Data count1	Data count2
	D	0.0	0	0
A	D	0.1	0	0
B	D	0.2000	0	1
C	D	0.3000	0	2
B	D	0.6333	1	3
A	D	0.8333	1	4

Data probability: continuous
Data count1: integer
Data count2: integer

Data Probability = (datacount1 / datacount2) + Previous (Data Probability)
Datacount1 = Keeping record of fake objects allocated
Datacount2 = Total number of count incrementing depending on each access.

When Agent A is accessing data of Agent D, probability of A will be increased and data probability of Agent D will be increased. Same goes on continuing, if probability or data probability which one is reaching to threshold earlier, allocation is denied or new allocation done. Probability calculation for multi user environment is same like single agent. Difference here is, the data probability is calculated for agent who accessing data from other agent [13][14].

II. ALGORITHM FOR PROPOSED SYSTEM

Consider two dependent objects A and B. Let A be the parent object and B be the child object. The degree of dependency from B to A can be represented by the conditional probabilities pi|j =Pr(B=bi|A=aj) [15].

The conditional probabilities of the child object given all of its parents are summarized into a Conditional Probability Table (CPT) that is attached to the child object [16][17].

If the semantic relation between the source and the target object is unknown or if the values of the source object is unknown, then the source and target object are independent. Thus, there is no semantic link between them [18].

To represent the case of the unknown semantic relationship, we need to introduce the attribute value "Unknown" to the source object and set the value of the source object to "unknown." In this case, the source and target object are independent, i.e., ***Pr(T=ti|P1=v1,…Pn=vn, PS=unknown)=Pr(T=ti|P1=v1, … Pn=vn).***

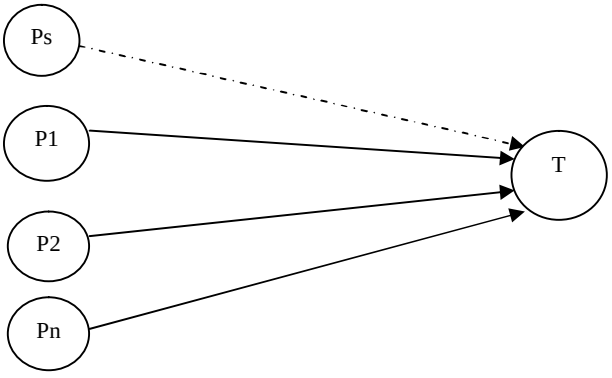


Fig 1. Semantic Relationships between source and Target Objects

When the semantic relationship is known, the conditional probability table of the target object is updated with the

known semantic relationship. If the value of the source object and the semantic relation are known, then Pr (T=ti| P1= v1, … Pn= vn, PS=sj) can be derived from the specific semantic relationship [19].

In Fig 1, the semantic relationship decides that Pr (T=t1| P1, … Pn, PS=s1)=0.6 and Pr(T=t1| P1, … Pn, PS=s2)=0.8.

III. EXISTING SYSTEM

There have been several approaches for the data leakage detection problem. An existing tutorial provides a good overview on the research conducted in this field. Suggested solutions are domain specific, such as lineage tracing for data warehouses, and assume some prior knowledge on the way a data view is created out of data sources. Watermarks were initially used in images, video, and audio data whose digital representation includes considerable redundancy. However in this approach, a watermark modifies the item being watermarked. If the object to be watermarked cannot be modified, then a watermark cannot be inserted. In such cases, methods that attach watermarks to the distributed data are not applicable. Finally, there are also lots of other works on mechanisms that allow only authorized users to access sensitive data through access control policies. Such approaches prevent in some sense data leakage by sharing information only with trusted parties [20].

Disadvantages of Existing System

Even though the existing approaches are able to detect the data leakage, they were all restricted or made impossible to satisfy agents' request.

IV. PROPOSED SYSTEM

The lists of modules developed for the proposed system are:

- ***Explicit Data Request with e-random***
- ***Explicit Data Request with e-optimal***
- ***Sample Data Request with s-random***
- ***Sample Data Request with s-overlap***
- ***Sample Data Request with s-max***
- ***Semantic interface model technique***
- ***Performance Evaluation***

- ***Explicit Data Request with e-random***

In this model we present an approach of explicit data request based on e-random. Here we combine the allocation of the explicit data request with the agent selection of e-random. We use e-random as our baseline in our comparisons with other algorithms for explicit data requests [21]. Initially we the finds agents that are eligible to receiving fake objects in O(n) time. Then, the algorithm creates one fake object in every iteration and allocates it to random agent. The main loop takes O(B) time. Hence, the running time of the algorithm is O (n + B).

- ***Explicit Data Request with e-optimal***

To improve the algorithm for allocation explicit data request we are combining this algorithm with the agent selection for e-optimal method. This is based on e-optimal makes a greedy choice by selecting the agent that will yield the greatest improvement in the sum-objective [22].
The cost of this greedy choice is $O(n^{2)}$ in every iteration. The overall running time of e-optimal is $O(n + n^2B) = O(n^2B)$.

- ***Sample Data Request with s-random***

Here we present the sample data request with s-random. Here in this method we present the object selection for s-random. In s-random, we introduce vector a O N|T| that shows the object sharing distribution. In particular, element a[k] shows the number of agents who receive object tk [23]. Algorithm s-random allocates objects to agents in a round-robin fashion. After the initialization of vectors d and a, the main loop is executed while there are still data objects (remaining > 0) to be allocated to agents. In each iteration of this loop, the algorithm uses function SELECTOBJECT () to find a random object to allocate to agent Ui. This loop iterates over all agents who have not received the number of data objects they have requested [24][25].

- ***Sample Data Request with s-overlap***

In the previous section the distributor can minimize both objectives by allocating distinct sets to all three agents. Such an optimal allocation is possible, since agents request in total fewer objects than the distributor has. This is overcome by presenting an object selection approach for s-overlap. Here in each iteration of allocating sample data request algorithm, we provide agent U*i* with an object that has been given to the smallest number of agents. So, if agents ask for fewer objects than jTj, agent selection for s-optimal algorithm will return in every iteration an object that no agent has received so far. Thus, every agent will receive a data set with objects that no other agent has. The running time of this algorithm is O(1).

- ***Sample Data Request with s-max***

In this module we present an improved algorithm than s-overlap and s-random which we used in allocation algorithm. This algorithm we present here is termed as object selection

ISSN 2348 – 9928
Doi:01.0401/ijaict.2014.05.09 Published on 05 (3) 2016

for s-max [26]. If we apply s-max to the example above, after the first five main loop iterations in algorithm of allocating data request, the R_i sets are:

$$R_1 = \{t_1, t_2\};\ R_2 = \{t_2\};\ R_3 = \{t_3\};\ \text{and}\ R_4 = \{t_4\}$$

In the next iteration, function SELECTOBJECT () must decide which object to allocate to agent U_2. We see that only objects t_3 and t_4 are good candidates, since allocating t_1 to U_2 will yield a full overlap of R_1 and R_2. Function SELECTOBJECT () of s-max returns indeed t_3 or t_4. The running time of SELECTOBJECT () is O (|T|n).

- ***Semantic interface model technique***

In this module we are proposing our enhanced approach for detecting the guilty agents. In this technique we use the semantic inference graph approach for the semantic inference model that represents the possible colluding attacks from any agents to the different data allocation strategies [27].

SIM represents dependent and semantic relationships among attributes of all the entities in the information system. The related attributes (nodes) are connected by three types of relation links: dependency link, schema link, and semantic link. The dependency link connects dependent attributes within the same entity or related entities [28][29].

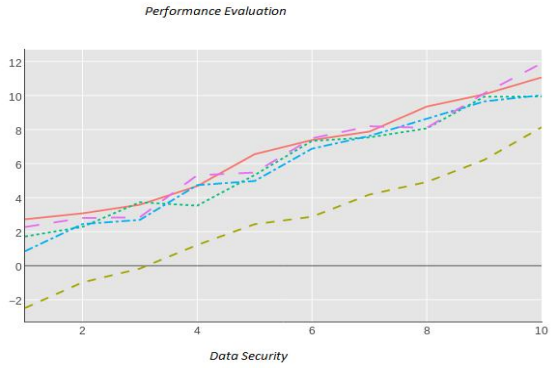
The schema link connects an attribute of the primary key to the corresponding attribute of the foreign key in the related entities. The semantic link connects attributes with a specific semantic relation. To evaluate the inference introduced by semantic links, we need to compute the CPT for nodes connected by semantic links [30][31]. In order to perform inference at the instance level, we instantiate the SIM with specific entity instances and generate a SIG Each node in the SIG represents an attribute for a specific instance.

Related attributes are then connected via instance-level dependency links, instance-level schema links, and instance-level semantic links. The attribute nodes in the SIG have the same CPT as in the SIM because they are just instantiated versions of the attributes in entities [32][33].
As a result, the SIG represents all the instance-level inference channels.

V. PERFORMANCE EVALUATION

In this section, the performance of the proposed approach is evaluated with the existing approaches. In this performance evaluation we are also finding how effective the approximation is. We also present the evaluation for sample requests and explicit data requests. The experimental result shows that our approach of using the semantic inference graph performs better than the existing approaches. The graph shows the performance evaluation of the proposed system.

Fig 2 : Performance Evaluation



VI. CONCLUSION

We analyze and compare the performance offered by Explicit random, Explicit optimal, sample random, sample overlap, sample max and semantic inference model. Here the semantic inference model has high confidence rate when compared with other existing algorithm. Based on the comparison and the results from the experiment show the proposed approach works better than the other existing systems.

References

[1] 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.

[2] 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.

[3] 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.

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

[5] Y. Cui and J. Widom, "Lineage Tracing for General Data Warehouse Transformations," The VLDB J., vol. 12, pp. 41-58, 2003.

[6] S. Czerwinski, R. Fromm, and T. Hodes,"Digital Music Distribution and Audio Watermarking," http://www.scientificcommons.org/43025658, 2007.

ISSN 2348 – 9928
Doi:01.0401/ijaict.2014.05.09 Published on 05 (3) 2016

[7] 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.

[8] F. Hartung and B. Girod, "Watermarking of Uncompressed and Compressed Video," Signal Processing, vol. 66, no. 3, pp. 283-301, 1998.

[9] 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.

[10] 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.

[11] B. Mungamuru and H. Garcia-Molina, "Privacy, Preservation and Performance: The 3 P's of Distributed Data Management, "technical report, Stanford Univ., 2008.

[12] V.N. Murty, "Counting the Integer Solutions of a Linear Equation with Unit Coefficients," Math.Magazine, vol. 54, no. 2, pp. 79-81, 1981.

[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] P. Papadimitriou and H. Garcia-Molina, "Data Leakage Detection, "technical report, Stanford Univ., 2008.

[15] P.M. Pardalos and S.A. Vavasis, "Quadratic Programming with One Negative Eigen value Is NP-Hard," J. Global Optimization, vol. 1, no. 1, pp. 15-22, 1991.

[16] W. Bender,D. Gruhl, and N.Morimoto Techniques for data hiding In Proc of the SPIE 2420 (Storage and Retrieval for Image and Video Databases III), pages 164–173, 1995.

[17] S. Benjamin, B. Schwartz, and R. Cole Accuracy of ACARS wind and temperature observations determined by collocation. Weather and Forecasting, 14:1032–1038, 1999.

[18] L. Boney, A. H. Tewfik, and K. N. Hamdy Digital watermarks for audio signals. In International Conference on Multimedia Computing and Systems, Hiroshima, Japan, June 1996.

[19] C. S. Collberg and C. Thomborson.Watermarking, Tamper- Proofing, and Obfuscation—Tools for Software Protection. TechnicalReport 2000-03, University of Arizona, Feb 2000.

[20] I. J. Cox andM. L.Miller. A review of watermarking and the importance of perceptual modeling.In Proc. of Electronic Imaging, February 1997.

[21] S. Craver, N.Memon, B.-L. Yeo, and M. M. Yeung Resolving rightful ownerships with invisible watermarking techniques: Limitations, attacks, and implications. IEEE Journal of Selected Areas in Communications, 16(4):573–586, 1998.

[22] S. Czerwinski. Digital music distribution and audio watermarking. Available from http://citeseer.nj.nec.com.

[23] J.-L. Dugelay and S. Roche.A survey of current watermarking techniques. In S. Katzenbeisser and F. A. Petitcolas, editors, Information Hiding Techniques for Steganography and Digital Watermarking, chapter 6, pages 121–148. Artech House, 2000.

[24] F. Hartung and B. Girod. Watermarking of uncompressed and compressed video.Signal Processing, 66(3):283–301, 1998.

[25] N. F. Johnson, Z. Duric, and S. Jajodia. Information Hiding: Steganography andWatermarking – Attacks and Countermeasures. Kluwer Academic Publishers, 2000.

[26] Joseph J. K. ´O Ruanaidh, W. J. Dowling, and F. M. Boland Watermarking digital images for copyright protection.IEEE Proceedings on Vision, Signal and Image Processing, 143(4):250–256, 1996.

[27] S. Katzenbeisser and F. A. Petitcolas, editors. Information Hiding Techniques for Steganography and Digital Watermarking. Artech House, 2000.

[28] A. Kerckhoffs. La cryptographiemilitaire. Journal des SciencesMilitaires, 9:5–38, January 1883.

[29] E. Lander. Array of hope.Nature Genetics, 21:3–4, 1999.

[30] M. Maes. Twin peaks: The histogram attack on fixed depth image watermarks. In Proc. of the 2nd International Workshop on Information Hiding, pages 290–305. Springer- Verlag Lecture Notes in Computer Science 1525, 1998.

[31] N. Maxemchuk. Electronic document distribution. Technical Journal, AT&T Labs, September 1994.

[32] B. Schneier. Applied Cryptography. John Wiley, second edition, 1996.

[33] N. R. Wagner. Fingerprinting.In IEEE Symp.on Security and Privacy, pages 18–22,Oakland, California, April 1983