International Journal of Scientific & Engineering Research, Volume 5, Issue 2, February-2014
ISSN 2229-5518

Data Distribution Strategies To Improve The Distributors Chance To Ascertain Leaker

S.V.RAMANAN,MCA,
M.Tech(CSE),
PRIST University.
Puducherry Campus.

MR.V.UDHAYA KUMAR.M.Tech,
Assistant Professor,
Dept. of CSE,
PRIST Universit,y
Puducherry Campus.

ABSTRACT

In course of any domain, data leakage is the main hindrance in data distribution. Distributor provides the required set of records to the agents where they can make use of it. When the agent emerges as a guilty one and had leaked the data to other parties, then data leakage results. Eventually if the data founds to be in some other places other than the agents who received the actual data, then distributor needs to identify the guilty agents. The reorganization of guilty agent can be defined using proposed techniques such as unobtrusive and perturbation analyzes. We implement data allocation strategies ensuring intelligent data distribution. Further data leakage prevention is carried out by DES encryption technique as a significant parameter.

Index Terms— **Allocation strategies, fake records, leakage model, perturbation, unobtrusive.**

IJSER

I. INTRODUCTION

In the real world, considering to any specific domain which consoles any data transfer or data distribution, confidentiality plays a vital role. In major groups of organizations, their main mode of processing will be through data transfer. This data transfer will be carried out through distributor i.e., the owner of the data and the agents who make use of it. In our project we use dynamic creation of database, where the distributor can upload the data corresponding to his domain. Based on the registration we discriminate the agent as normal agent or authorized agent. Further, their data transfer and guessing of guilty agent process in case of data leakage carried out by using the technique named, unobtrusive analysis. And using perturbation technique we discriminate the data into sensitive as well as non sensitive data.

Though the agent is a normal or authorized one the distributor must satisfy their requested constraints. Hence when the agent leaked the set of records or data to any parties, automatically notification is send to the distributor and sequentially the leaked data had made into unreadable format.

II. DESCRIPTION

A. ENTITIES AND AGENTS

Let the distributor database owns a set S={t1….tm} which consists of data objects. Let the no of agents be A1, A2,…..An,. The distributor distributes a set of records S to any agents based on their request such as sample or explicit request.

SAMPLE REQUEST

Sample request is a request send by a normal agent, where additional parameter is generated taking the IP address and port number of the agent system. It is then transmitted along each requested files .
Sample request Ri = SAMPLE(S,ui)

EXPLICIT REQUEST

Explicit request is a request send by an authorized agent defines for all set of records. Here fake objects are generated as a whole for all requested files.

Explicit request Ri = EXPLICIT (S, condi)

B. GUILTY AGENTS

Guilty agents are the agents who had leaked the data. In data distribution, agents send sample request or explicit request to the distributor and can receive the files based on their constraints. Later suppose the agent say A1 had leaked the data knowingly or unknowingly. Then automatically notification will be the send to the distributor defining that agent A1 had leaked the particular set of records which also specifies sensitive or non sensitive records.Our goal is to estimate the likelihood that the leaked data came from the agents as opposed to other sources. For instances, if one of the K objects is transmitted only to an particular agent say A1, where as other agent received other objects. And if found that that particular set of data is leaked it is easily to find out the probability of guessing the guilty agent as A1. Because using perturbation analysis we categorized the data as sensitive and non sensitive data. When we define Ai is guilty and we predict one or more objects had been leaked out as S, then the agent Ai is an guilty one.

III. AGENT GUILT MODEL

In order to measure the probability of guilty agents,we need to describe the sets of values or records or data had been leaked out. Hence the guessing values be K and the probability that agent say A1is computed by

Pr={ GAi | K}.

A component failure in our case is the data leakage (set of records) is leaked out from agent to others (parties) and detecting out from whom (agent) the data had been leaked out. The component failure is considered in order to prove the systems will the high level of reliability. Experimentally, in order to find approximate details of 100 individuals or employees.Taking up of an objects S which contains set of records as addresses of an employee, if this person can find say for 90 addresses of an employees, then probability of identifying a one address is 0.9.Former computing of the formula

Pr={ GAi | K} ,let us assume an simple example where distributor set contains a set of objects named S and agent sets Rs and target set as K are all defined as :

S={ t1….tm} , R1 = {t1 , t2, t3} R2 = { t2, t3}
K= {t1 , t2, t3, t4 }

Here in this type of case, agent may leak any common tuples of record say t2 may leakage had been occur on some specified sets of records. As far from our assumption considering any condition the probability of guessing the guilty agent is identified using an generalized formula as follows

$$\Pr\{ | \} 1 \Pr\{ \} 1 1 \; GA\,K = - GA \quad (1)$$

In the general case, to find the probability that an agent Ax is guilty given a set K, we compute the probability that he leaks a single object t to K. To compute this, we define the set of agents Vt={Ax|tÎRx} that have t in their data sets and it can be given as

$$\Pr\{\textit{some agent leaked t to K}\} = 1 - p \quad (2)$$

Assuming that all agents that belong to Vt can leak t to K with equal probability and we obtain the following

IV GUILT MODEL ANALYSIS

In order to model the parameters we take into consideration of two simple scenarios. In each scenario the target has obtained all objects provided by the distributor i.e., S=K

A. IMPACT OF PROBABILITY P:

In our first scenario, the distributors set S totally contains 16 objects: all of the 16 objects are given to agent A1 and only eight are given to a second agent A2. We calculate the probabilities Pr{GA1|K} and Pr{GA2|K} for p in the range [0,1] and we present the results diagrammatically .The dashed line represents Pr{GA1|K} and the solid line represents Pr{GA2|K}.It is more unlikely the target guessed all 16 values as the value of p approaches 0. Each agent has enough of the leaked data that its individual guilt approaches1. However, the probability that A2 to be guilty decreases significantly as p value increases: since the 8 objects given to A2 is also given to A1. Hence it is harder to blame A2 alone for leakage. Also A2's probability of guilt remains close to 1 as p increases, as A1 has eight objects not given to other agents. As p value approaches 1, the target would have guessed all 16 objects. Hence the agent's probability of guilt goes to 0.

B. IMPACT OF OVERLAP BETWEEN RX AND K:

Here we take into account two agents, one receiving the entire S = K data and the second one receiving a varying fraction of the data. Fig.1b shows the probability of guilt for both agents, as a function of the fraction of the objects owned by A2, i.e., as a function of |R2 Ç S|=|S|. In this case, p has a low value of 0.2, and A1continues to have all 16K objects. We see that when objects are rare (p =0.2), it does not take many leaked objects before we can say that A2 is guilty with high confidence. This result indicates that even an agent holding a small number of incriminating objects is clearly suspicious. Figs.1c and 1d shows scenario, where values of p equal to 0.5 and 0.9. This indicates that the rate of increase of the guilt probability decreases as value of p increases. As the object become easier to guess, it takes more and more evidence of leakage (more leaked objects owned by A2 before we can have high confidence that A2 is guilty. The scenario conclusion shows: If there are more number of agents holding the same replicating data it is harder to blame any of the agent in case of data being leaked.

IV. DATA ALLOCATION PROBLEM

This paper takes intelligent data distribution as a significant parameter. This method of data distribution improves the probability of guessing the guilty agents. We include either randomly generated additional parameters or fake objects for guessing the guilty agents. This inclusion is based on the type of request made by the agents. There are two types of requests made by agents: sample and explicit. We generate random additional parameters incase of serving sample data requests. While handling explicit date requests we include Fake objects generated by the distributor. The fake objects appear realistic and do not belong to the actual real objects. This increases the chances of detecting guilty agents who leaks the data.

A. FAKE OBJECTS:

In order to improve the effectiveness in detecting guilty agents the distributor may be able to add fake objects to the distributed data. Adding fake objects is not allowable always since it has a impact over correctness of what agents do. Perturbing data to detect leakage is not new. In most cases, individual objects are perturbed, e.g., by adding random noise to sensitive salaries, or adding a watermark to an image. In our paper, perturbing distributor data can be carried out by adding fake elements in case of explicit requests. In some sensitive cases such as medical applications, perturbing real objects can cause problems. Here the data objects can be patient records and Hospitals may

be agents. In such sensitive cases fake records are added instead of perturbing real records. The distributor creates and adds fake objects to the data that he distributes to agents. Let Fx ÍRy be the subset of fake objects that agent Ax receives. Fake objects must be created carefully so that agents cannot distinguish them from real objects. In some cases the distributor is limited in creating fake objects. For example, in cases where objects containing e-mail addresses, it is required to create an actual inbox (if not the agent may discover the fake objects). Since such creation and monitoring of inboxes consumes resources and efforts, the distributor is limited in creating fake objects. There are also limitations in the number of fake objects received by each agent so as to not arouse suspicions. Thus the distributor can send up to bi fake objects to agent Ax.

CREATION OF FAKE OBJECTS:

The creation of fake objects looking as that of real is a nontrivial problem. Here, creation of a fake object for agent Ax is modeled as a black box function. CREATEFAKEOBJECT (Rx;Fy;condx) where Rx is the set of all objects that is given as input, Fx the subset of fake objects that Ai has received so far, and condx, and returns a new fake object. The condx is needed to produce a valid object that satisfies Ax's condition. It is necessary that the CREATEFAKEOBJECT() function should be aware of the fake objects added so far, so as to ensure proper statistics The distributor can also use function CREATEFAKEOBJECT() in case of sending the same fake object to a set of agents.

B. OPTIMIZATION PROBLEM:

The distributor's data allocation to agents has one constraint and one objective. The *constraint* deals with serving the agents requests .This can be done by providing the objects satisfying the conditions of agents. The *objective* is to detect the leakage that occurs in any part of the distributed data. The satisfaction of constraint is considered as a strict parameter. The distributor may not deny serving an agent request as in [14] and may not provide agents with different perturbed versions of the same objects as in [2]. The only possible constraint relaxation is fake object distribution. We now introduce some notation to state formally the distributor's objective. Recall that Pr{ GAy | K= Rx} or simply Pr{Gy|Rx} is the probability that agent Uy is guilty if the distributor discovers a leaked table K that contains all Ri objects. The difference function is defined to be D(*x, y*)

Note that differences D have non negative values: provided that set Ri contains all the leaked objects, agent Ax is at least as likely to be guilty as any other agent. Difference D(*x, y*) is positive for any agent Ay, whose set Ry does not contain all data of K. It is zero if Rx ÍRy.

In such cases both the agents Ux and Uy are considered to be guilty by the distributor since both of them have received the leaked objects. The larger the value of D(*x, y*) it is easier to detect Ax as leaking agent. Thus, the data distribution should be in the manner that D values are large.

PROBLEM DEFINITION:

The distributor have data requests for tables t1,t2…tn from various agents A1,A2,…An such that

• He satisfies the agents

• He maximizes the probability of detecting guilty agents Assuming that the Ni sets satisfy the agent's requests, multicriterion optimization problem is expressed as:

maximize(over K1…Kn) (...,D(*x, y*)....) x ¹ y (6)
If the optimization problem has an optimal solution O* ={k1…kn} such that any other feasible location O={k1…kn} yields (*x, y*) (*x, y*) * D ³ D for all a and y.This indicates that the allocation K* allows the distributor to detect the guilty agent with higher confidence.

C. OBJECTIVE APPROXIMATION

The agent's guilt probability does not have any effect on approximation of objective (6) and (7) and therefore on p:

This approximation becomes valid if minimizing the relative overlap maximizes D(*x, y*) The argument shows that

C. APPROXIMATE SUM-OBJECTIVE MINIMIZATION:

The chances of detecting a guilty agent get increased by minimizing the sum-objective, on average, by providing agents who have small requests with the objects shared among the fewest agents. This way, we improve our chances of detecting guilty agents with small data requests, at the expense of reducing our chances of detecting guilty agents with large data requests.

D. APPROXIMATE MAX-OBJECTIVE MINIMIZATION:

Algorithm s-overlap is optimal for the max-objective optimization only if $\Sigma n \; x \; x \; m \leq |T|$. The algorithms s-sum and s-random ignore this objective.

To improve the worst-case behavior, we implement a new algorithm that builds upon algorithm 4 that we used in s-random and s-overlap. We define a new SELECTOBJECT() procedure in algorithm 7. We denote the new algorithm by s-max.

In algorithm 7, we allocate to an agent the object that yields the minimum increase of the maximum relative overlap among any pair of agents. The running time of SELECTOBJECT() is $O(|T|n)$ and its calculation implies that we keep the overlap sizes RxI Ry for all agents in a two-dimensional array that we update after every object allocation. It can be shown that algorithm s-max is optimal for the sum-objective and the max-objective in cases where $M \leq |T|$ and als if $|T| \leq M \leq 2|T|$ or m1=m2=…=mn.

VI. CONCLUSION AND FUTURE ENHANCEMENTS

In a perfect world, there would be certain needs to the data provider to hand over sensitive data to the requested agents that may unknowingly or maliciously leak it. In spite of these difficulties, we have presented that it is possible to assess the likelihood that an agent is responsible for a leak, based on the probability that objects can be identified by other means. Our model is relatively simple where the algorithms we have presented implement a various data distribution strategies that can improve the distributor's chances of identifying a leaker, data leakage detection and also the encryption technique implementation prevents the leaked data from further forwarding by guilty agents to others. Since our implemented algorithms do not include the handling of multiple agent's requests in an online fashion, we can extend the allocation strategies that

support online fashion effectively and leave its implementation to a future paper.

REFERENCES

[1] P. Papadimitriou and H. Garcia-Molina, "Data Leakage Detection," IEEE Transactions on Knowledge and Data Engineering, vol. 23, no. 1, January 2011

[2] 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.

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

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

[5] 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.

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

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

[8] 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), VLDBEndowment, pp. 151-162, 2006.

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

[10] L. Sweeney, "Achieving K-Anonymity Privacy Protection Using Generalization and Suppression," http://en.scientificcommons. org/43196131, 2002

BIOGRAPHY

1.**S.V.RAMANAN** has Passed Master of computer Applications in
Computer Science & Engg, In MCA PASSED 85 Percentage marks
2 **Assist Prof. V. Udhaya kumar**: Has completed Master of Engg in Computer
Science & Engg. Currently working as Associate Professor at PRIST UNIVERSITY, Puducherry, India. Total teaching experience-10
years. .

IJSER