ISSN: 2321-7782 (Online)
Volume 2, Issue 3, March 2014
International Journal of Advance Research in Computer Science and Management Studies
Research Article / Paper / Case Study
Available online at: www.ijarcsms.com

Identifying Guilty Agent for Data Leakage Detection System

Bhagwan D. Thorat[1]
Research Scholar
BVDUCOE
Pune, Maharashtra – India

P. R. Devale[2]
Head, Dept. of IT
BVDUCOE
Pune, Maharashtra – India

Abstract: ***We study the concepts of identify guilty agent for data leakage system, factor and methods to identify the leakage of data. The data is used in important and so it should not be altered or leaked. In IT domain now a day's use of huge database is common. Simultaneously this data base is shared among many people. When we interchange the information, the weaknesses, leakage or modification rate increases. The solution for the above stated problem is to prevent data leakage. This paper describes a procedure to prevent and detect the data leakage persons.***

Keywords: data leakage, perturbation, unobtrusive, fake object, guilty agent, watermarking.

I. INTRODUCTION

More times most important data can be sending through the trusted agents (third parties). Significant information of companies and association includes intellectual property, financial information, patient information and different information associated with the domain and industry. Companies distribute consumer's information with other companies who are in partnership with that company. In this scenario data security is important so data leakage detection will play significant role. Our purpose is to identify when the owners significant information have been leaked by third parties and to identify the responsible agents.

II. PREVIOUS WORK

In existing system watermarking technique is used for finding leakage detection, e.g., an exclusive code is fixed in each distributed copy [1]. If that data is shortly exposed in the number of unofficial party, the leaker can be recognized. Watermarks can be most helpful in various cases, but another time, occupy various changes of the unique information. Also, watermarks can sometimes be damaged if the information beneficiary is malicious. A Company may have joint ventures with another company that requires distribution consumer information.

III. PROPOSED WORK

Our purpose is to identify when the owner's essential information have been leaked by agent, and if probable to detect the agent who is responsible to alter or leak the data. *Perturbation* is a very helpful procedure where the information are changed and made "less sensitive" before being hand over to agents. In this sector, we learn the attracting attention method for identifying leakage of the number of entities or documentation.

In this sector, we build up a model to monitor the agent for his implied offence. We also present algorithms to share the objects to number of agents, in a way that develop our possibilities to identifying a leaker. Finally, we also think the alternative of adding "fake" objects to the unique data. Such objects do not match to actual existing entities but look like real to the agent. In intelligence, the fake entity work as a type of watermark for the whole set, without changing any individual members [2]. If an agent who got fake object leaks the information, the distributer can confirm the guilty agents.

IV. DESCRIPTION

A) Entities and Agents

The distributor database owns set of data entity C= {v_1, v_2. . . v_n}. The distributor wants to divide some of the entities with a set of agents U_1, U_2 . . . , U_m but does not wish the entities be leaked to third parties. The entities in C could be of any type and size, e.g., they could be tuples in a relation in a database. The agent U_i receives a subset C_i of object C determined either by an implicit request or an explicit request [4].

Explicit Request

In this explicit demand the agent will send the demand with proper condition. The agents define all set of records the information generated after adding the fake object.

Explicit request T_i = EXPLICIT (C, Condi); Agents U_i receives all C objects that satisfy condition.

Implicit Request

In this request agent's request does not have any condition. The agents fire the need without constraint according to the query and he will get the data with fake object.

Implicit request = IMPLICIT (C, m_i): Any subset of m_i records from C can be given to U_i.

B) Guilty Agents

Guilty agents are responsible to leak the data. Assume that once the objects are given to agents, the distributor find out a set C belongs to T has leaked indicate some trusted agents, called the target, has been caught in control of C. Our purpose is to approximate the disclosed information came from the trusted agents as disparate to other sources. We consider an agent Ui is guilty if it gives single or more objects to the goal. We signify the event that agent Ui is guilty by Gi and the event that agent Ui is guilty for a given leaked set C by Gi|C. Our next step is to estimate Pr (Gi|C), i.e., the probability that agent Ui is guilty given proof C.

C) Related Work

To detect the implied offence the approach we present is related to the information origin problem [1]: tracing the descent of C objects implies fundamentally the identification of the implied offence agents. After the data distribution plan is anxious, our job is mostly related to watermarking that is used as to institute original ownership of distributed objects. Watermarks were used in all types of data whose digital representation includes huge redundancy. Here we projected watermarking algorithms that embed the watermark bits in the least significant bits (LSB) of selected attributes of a selected subset of tuples. This method doesn't grant mechanism for multi bit watermarks; instead only secret key is used. For each tuples a protected message authenticated code (MAC) is computed using the secret key and tuples primary key [3].

V. GUILT AGENT MODEL

We can compute Pr (Gi|C), by possible values in S that can be "assume" by the target. We call this estimate pt, the likelihood that objects t 'can be guessed by the target. Probability pt is related to the probabilities used in designing fault-tolerant systems. To approximating it likely look at the system which are operational over a given period need the probability that individual components will or will not fail. In case the target guesses the object of C it shows the component failure. The component failure is used to calculate the overall system trustworthiness, at the time of possibility of guessing the faulty agent. Possibilities are estimated based on experiments, just as we propose to estimate the points. On the same scale the component possibilities are usually traditional approximation, other than the exact numbers. Now, we will come to know that the level of

trustworthiness of system, but possibly privileged. In the same manner, the pt's those are higher than the true values, will be guilty with at least the computed probabilities.

VI. Data Allocation problem

The core point of discussion in this paper is data allocation problem: how can the distributor "wisely" give data to agents in order to get better likelihood of spot a guilty agent? By identifying misconducts in this problem we referred to the type of data needs by agents and can find whether fake objects are allowed or not.

6.1) Fake Objects:

The objects which are not in set S are fake object. These objects will look like real objects that help us to increase possibility of identify agents that leaks information. In following fig1 we have clarified four problem instances according to the names EF, EF, SF and SF where E refers to explicit request, S refers to sample request, F refers to fake objects, and F for the case where fake objects are not allowed. The plan of disturbing information to detect leakage is not new. In these case the disturbing the set of distributor object by adding fake elements. In a number of application, fake objects may lead to fewer problem that disturbing genuine objects. With the help of example, say the distributor's information objects are industrial records and the agents are industry. Here even little changes to the records of actual consumer may be unwanted. Even though, the addition of some fake industry records may be acceptable, since no consumer equate with these records, and hence we resolve these fake record problems. The distributors generate and put in fake objects to the information that he circulates to the agents. We let Fi belongs Ri be the subset of fake objects that agents Ui receives. We need to create fake objects such as the agents cannot differentiate them from genuine objects.

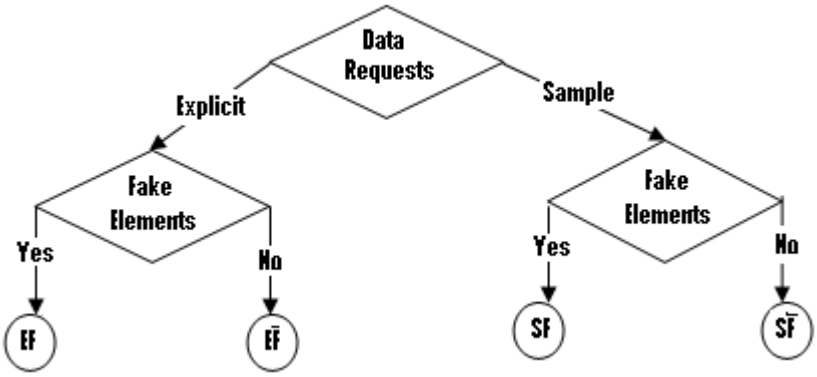

Fig.1. Leakage problem instances.

6.2) Optimization Problem

The optimization model is the distributor's data distribution to agents has one limitation and one goal. The distributor's restriction is to satisfy agents' request, given that with the number of objects they demand all available objects that gratify their circumstances. His aim is to be able to sense an agent who leaks any part of his information. The objective is to make the most of the chances of detecting a responsible agent that leaks all his data objects. The distributors possibly will not provide agents with different worried versions of the same objects an in [1]. We consider fake object distribution as the only possible constraint leisure.

VII. Conclusion

In perfect world there agents may unknowingly or maliciously leak important data. If we want to share important data in the database we should watermark every object then we will be able to identify the decent of the object with complete certainty. However, in many cases, we must certainly take care of the agents that may not be completely trusted, since certain data cannot

acknowledge watermarks. Other than these problems, we have exposed that it is possible to evaluate that an agent is to blame for a leak, based on overlie is information with the leaked information and the information of other agents, and based on the likelihood that objects can be "estimate" by other means. Our model is comparatively simple. The algorithms we have offered implement a variety of data distribution policies that can improve the distributor's chances to find out a leaker. We have exposed that distributing objects can the decent of the object.

References

1. Panagiotis Papadimitriou & Hector Garcia-Molina, Member IEEE. "Data Leakage Detection" IEEE Transaction on Knowledge and data Engineering, Vol. 23, No.1,January 2011.
2. D. Krishna Madhuri, A. Jagadeswara Rao, K. C. Ravi Kumar "Identifying Leakage of Distributor Sensitive Information by Agents". International Journal of Advanced Computer and Mathematical Sciences ISSN 2230-9624. Vol. 3, Issue 3, 2012.
3. Priyanka Barge, Pratibha Dhawale, Namrata Kolashetti "A Novel Data Leakage Detection" International Journal of Modern Engineering Research. Vol.3, Issue 1, Jan-Feb 2013.
4. Jayavarapu Karthik and Dr. P. Harini "Data Leakage Detection" International Conference on Computing and Control Engineering, 12 & 13 April , 2012.
5. B. Srinivas Rao, A.Harshavardhan, N.Chandramouli, V.Kishore "Data Leakage Detection" Research Journal of Computer Systems Engineering Vol.3, Issue 04; August-September 2012.
6. Unnati Kavali, Tejal Abhang, Vaibhav Narawade "Data Allocation Strategies in Data Leakage Detection" International Journal of Engineering Research and Application. Vol. 2, Issue 2, Mar-Apr 2012.
7. Jayavarapu Karthik and Dr. P. Harini. "Data Leakage Detection" International Conference on computing and Control Engineering (ICCCE 2012), 12 & 13 April,2012.
8. Panagiotis Papadimitriou & Hector Garcia-Molina, "A Model for Data Leakage Detection" Stanford University 353 Serra Street, Stanford, CA 94305, USA.
9. R. Agrawal and J. Kieman, "Watermarking Relation Databases" Proc. 28th International Conference Very Large Data Bases (VLDB' 02), VLDB Endownment, pp. 155-166, 2002.

AUTHOR(S) PROFILE

Bhagwan D. Thorat, B.E. Computer Engineering, Pursuing M-Tech Information Technology in Bharati Vidyapeeth Deemed University College of Engineering, Pune, Maharashtra, India.

Prakash R. Devale, M.E. Computer Engg, Pursuing Ph. D in Bharati Vidyapeeth University, Pune, Maharashtra, India.

Currently working as Professor in the Department of Information Technology.

Bharati Vidyapeeth Deemed University College of Engineering, Pune, Maharashtra, India.

