



Identification of Data Leakage and Detecting Guilty Agents Using Data Watcher

Priya Walunj, Priya Tadge, Navnath Kondalkar, Satish Mahamare
MMIT, Pune University, Pune,
Maharashtra, India

Abstract— Researchers have proposed several mechanisms to secure data from unauthorized use but there is very less work in the field of detecting and managing an authorized or trustworthy agent that has caused a data leak to some third party advertently or unknowingly. In this project, we implement methods aimed at improving the odds of detecting such leakages when a distributor's sensitive data has been leaked by trustworthy agents and also to possibly identify the agent that leaked the data. We also implement some data allocation strategies that can improve the probability of identifying leakages and can also be used to assess the likelihood of a leak at a particular agent assuming the fact that the data was not simply guessed by the third party where the leaked data set has been found. We also propose new allocation strategies that work on the basis of SRF (Shortest Request First) algorithm. In this paper we implement a system called the Data Watcher for find out guilty agent [3].

Keywords— Allocation Strategies, Data privacy, Data Leakage, Detection, Data watcher.

I. INTRODUCTION

Now days, every organization is facing data leakage. That is very serious problem faced by organization. Data Leakage is the unauthorized transmission of private or sensitive data or information from within an organization to a third party [1]. In the real world scenario, a distributor needs to share sensitive data among various stakeholders such as employees, business partners and customers. This increases the risk that confidential information will fall into unauthorized hands.

For example, a hospital may give patient records to researchers who will devise new treatments. Similarly, a company may have partnerships with other companies that require sharing customer data. Another enterprise may outsource its data processing, so data must be given to various other companies. We call the owner of the data the distributor and the supposedly trusted third parties the agents. Our goal is to detect when the distributor's sensitive data have been leaked by agents, and if possible to identify the agent that leaked the data [2]. Traditionally, leakage detection is handled by watermarking, e.g., a unique code is embedded in each distributed copy. Water-marks can be very useful in some cases, but again, involve some modification of the original data.

We refer the following scenario: After giving a set of objects to agents, the distributor discovers some of those same objects in an unauthorized place. (For example, the data may be found on a website, or may be obtained through a legal discovery process.) At this point, the distributor can assess the likelihood that the leaked data came from one or more agents, as opposed to having been independently gathered by other means. In our system, we develop a model for assessing the guilt of agents. We also present algorithms for distributing objects to agents, in a way that improves our chances of identifying a leaker. Finally, we also consider the option of adding fake objects to the distributed set [5]. Such objects do not correspond to real entities but appear realistic to the agents. In a sense, the fake objects act as a type of watermark for the entire set, without modifying any individual members. If it turns out that an agent was given one or more fake objects that were leaked, then the distributor can be more confident that agent was guilty.

II. OBJECTIVE

1. The objective of the system is to detect when the distributor's sensitive data has been leaked by hackers, and if possible to identify the agent that leaked the data.
2. A data infringe is the inadvertent release of secure information to an un-trusted environment.
3. The goal is to estimate the likelihood that the leaked data came from the agents as opposed to other sources.
4. Not only do we want to estimate the likelihood the agents leaked data, but we would also like to find out if one of them in particular was more likely to be the leaker with large number of over-lapping.
5. The data allocation strategies help the distributor cleverly give data to agents.
6. Fake objects are added to identify the guilty part, to address this problem four instances are specified.
7. Depending on which the data request is provided.
8. Depending upon the type of data request, the fake objects are allowed.

III. EXISTING SYSTEM

In existing system data leakage detection is handled by watermarking, e.g., a unique code is embedded in each distributed copy. If that copy is later discovered in the hands of an unauthorized party, the leaker can be identified. Watermarks can be very useful in some cases, but again, involve some modification of the original data. Furthermore, watermarks can sometimes be destroyed if the data recipient is malicious. System is not online capture of leak scenario also in existing system more focus on data allocation problem. All paragraphs must be indented. All paragraphs must be justified, i.e. both left-justified and right-justified.

IV. PROPOSED SYSTEM

To find the solution on this problem we develop two models. First, when any employee of enterprise access sensitive data without the consent of owner in that case, we developed data watcher model to identifying data leaker in this point suppose data leaker will identify then no need to calculating the probability of agents that method gives near about 90 % of result. But suppose employee given data outside the enterprise for that we devolved second model for assessing the “guilt” of agents. Guilt model are used to improve the probability of identifying guilty third parties.

In this approach, the model for assessing the “guilt” of agents is developed. The option of adding “fake” objects to the distributed set are considered. Such objects do not correspond to real entities but appear practical to the agents. In a sense, the fake objects acts as a type of watermark for the entire set, without modifying any individual members. If it turns out an agent was given one or more fake objects that were leaked, then the distributor can be more con dent that agent was guilty.

Module

1) Database Maintenance:

Here the agent registration details are maintained and the sensitive data which are provided to agents are specified. The designing of the whole database is done.

2) Agent Maintenance:

Registration: Here details of agents are registered and it collects the information about them like what are the sensitive data they want.

History: Here the agent history is maintained like what all the details are given by distributor previously. It maintains entire details of the agent. To detect the guilty agents it checks the history and detects those agents who have fake details from the third party.

3) Detecting Guilty Agent:

Suppose that after giving objects to agents, the distributor discovers that a set S has leaked. Since the agents U1....Un have some of the data, it is reasonable to suspect them leaking the data. For example, say one of the objects in S represents a customer X. Perhaps X is also a customer of some other company, and that company provided the data to the target. The goal is to estimate the likelihood that the leaked data came from the agents as opposed to other sources. If one of the S objects was only given to agent U1, while the other objects were given to all agents, suspect U1 more. It says an agent Ui is guilty and if it contributes one or more objects to the target.

4) Data Allocation:

The two types of requests we handle: sample and explicit. Fake objects are objects generated by the distributor that are not in set T. The objects are designed to look like real objects, and are distributed to agents together with the T objects, in order to increase the chances of detecting agents that leak data. Fake objects are rep-resented using four problem instances with the names EF, EF, SF and SF. Where E stands for explicit requests, S for sample requests, F for the use of fake objects, and F for the case where fake objects are not allowed. Sample request $R_i = \text{SAMPLE}(T, m_i)$; Any subset of m_i records from T can be given to U_i . Explicit request $R_i = \text{EXPLICIT}(T, \text{condi})$; Agent U_i receives all the T objects that satisfy condition.

V. SYSTEM ARCHITECTURE

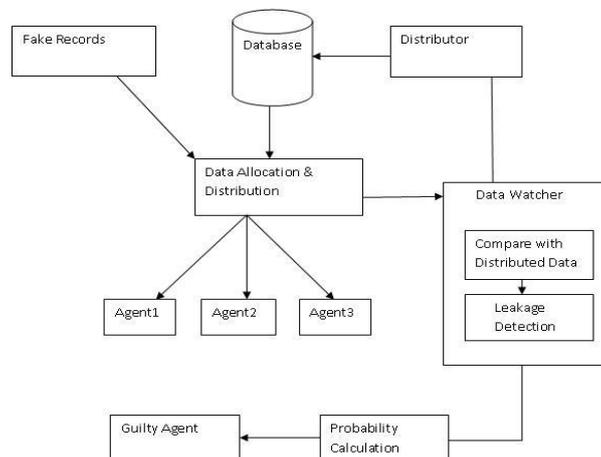


Fig.1 System Architecture

In this application, we try to implement a model application to detect the data leakages between distributor and agents. When the distributor sends a le to agent, it adds some fake object in record set which is given to that particular agent. In this system database maintained by Distributor according to the request by agents data passing to agents with fake or without fake object. When agent doing the business with target without the consent of distributor and leak data. In this sequence, le name and le path is stored in the database for future reference. Similarly when the agent sends a le to the unauthorized agent the action is watch by watcher system [3]. Then distributor match leak data with his data. Distributor also checks overlapping of data among agents and then he will calculate probability of agents [2]. The probability function is calculated based on the number of guilt agents by the number of transfers between the agent and unauthorized person. Thus we can find the guilt agent.

VI. ADVANTAGES

1. Using the technique of Perturbation data is made less sensitive for the agents to handle.
2. Realistic but fake objects are injected to the distributed data set to identify the guilt agent.
3. If two agents have same probability then the FIFO order is maintained to show the guilty agent.
4. Possibility of full service with maintenance and SLA in overall service.
5. Easier access to new software versions.

VII. APPLICATIONS

1. It provides data secrecy and identifies the guilt agent in case of data leakage.
2. Our work is mostly relevant to watermarking that is used as a means of establishing original ownership of distributed objects.
3. Our approach and watermarking are similar in the sense of providing agents with some kind of receiver identifying information.
4. The goal of these experiments was to see whether fake objects in the distributed data sets yield significant improvement in our chances of detecting a guilty agent.

VIII. CONCLUSION

In existing system watermarking technique was used but it requires some modification in original data. In some cases watermarks could be tempered or totally destroyed if recipient is malicious. Our model is relatively simple, but we believe it captures the essential trade-offs. The algorithms we have presented implement data allocation strategy that can improve the distributor's chances of identifying a leaker.

ACKNOWLEDGMENT

We would like to express our gratitude to all those who helped us to complete this work. We want to thank our guide Prof. Mane G.V. for her continuous help and generous assistance. She helped in a broad range of issues from giving us direction, helping to find the solutions, outlining the requirements and always having the time to see us. We would like to thank our colleagues who helped us time to time from preparing report and giving good suggestions. We also extend sincere thanks to all the staff members of Department of Computer Engineering and Information Technology for helping us in various aspects.

REFERENCES

- [1] Panagiotis Papadimitriou, Member, IEEE, Hector Garcia-Molina, Member, IEEE., *Data Leakage Detection*, IEEE Transactions On Knowledge And Data Engineering, Vol. 23, No. 1, January 2011
- [2] P. Papadimitriou and H. Garcia-Molina, *Data Leakage Detection*, technical report, Stanford Univ., 2008
- [3] N. P. Jagtap, S. J. Patil, A. K. Bhavsar, *Implementation of data watcher in data leakage detection system*, International Journal of Computer Technology Volume 3, No. 1, Aug, 2012
- [4] Ankit Agarwal, Mayur Gaikwad, Kapil Garg, Vahid Inamdar, *Robust Data leakage and Email Filtering System*, International Conference on Computing, Electronics and Electrical Technologies, 2012
- [5] Keerthi.P, M. Sheshikala, D. Rajeswara Rao, *Guilty Agent Detection by Using Fake Object Allocation*, International Journal of Computer Technology Volume -1, 2013
- [6] Rudragouda G Patil, *Development of Data Leakage Detection Using Data Allocation Strategy* International Journal of Computer Applications in Engineering Sciences, VOL I, Issue II, June 2011
- [7] Ajay Kumar, Ankit Goyal, Ashwani Kumar, Navneet Kumar Chaudhary, Sowmya Kamath S, "Comparative Evaluation of Algorithms for Effective Data Leakage Detection", Proceedings of 2013 IEEE Conference on Information and Communication Technologies (ICT), 2013.
- [8] Ahirrao P. P., Rai S. S., Pathania B. R., "Data Leakage Detection", International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-3, Issue-1, March 2014
- [9] B. Sruthi Patil, Mrs. M. L. Prasanthi, "Modern Approaches for Detecting Data leakage Problems", International Journal Of Engineering And Computer Science ISSN:2319-7242, Volume 2, Is-sue 2, Feb 2013
- [10] Ajay Kumar, Ankit Goyal, Ashwani Kumar, Navneet Kumar, Chaudhary, Sowmya Kamath S, Dept of Information Technology NITK Surathkal, India, *Comparative Evaluation of Algorithms for active Data Leakage Detection*, Proceedings of 2013 IEEE Conference on Information and Communication Technologies (ICT), 2013