Volume 6, Issue 3, March 2016 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Detecting Leakage of Sensitive Data Using Water Marking Technique

V. Vijayalakshmi*, T. Rohini, S. Sujatha, A. Vishali
CSE & Pondicherry University, Puducherry,
India

Abstract - In recent years, Data leakage is the big challenge in front of the research institutions and government organizations. Though there are number of systems designed for the data security by using different encryption algorithms, there is a big issue of the integrity of the users of those systems. It is very hard for any system administrator to trace out the data leaker and data leakage among the system users and data. In this paper, we propose a privacy preserving data-leak detection solution and watermarking techniques which can be outsourced and be deployed in a semi honest detection environment. Using watermarking model, we can provide security to our data during its distribution or transmission and even we can detect if that gets leaked.

Keyword: DLD, Data leak, watermarking techniques, fake object, clustering algorithms

I. INTRODUCTION

According to a report from Risk Based Security (RBS), the number of leaked sensitive data records has increased dramatically during the last few years. Deliberately planned attacks, inadvertent leaks (e.g., forwarding confidential emails to unclassified email accounts), and human mistakes (e.g., assigning the wrong privilege) lead to most of the data-leak incidents. Detecting and preventing data leaks requires a set of complementary solutions, which may include data-leak detection, data confinement, stealthy malware detection and policy enforcement.

Network data-leak detection (DLD) typically performs deep packet inspection (DPI) and searches for any occurrences of sensitive data patterns. DPI is a technique to analyze payloads of IP/TCP packets for inspecting application layer data, e.g., HTTP header/content. Alerts are triggered when the amount of sensitive data found in traffic passes a threshold. The detection system can be deployed on a router or integrated into existing network intrusion detection systems.

Straightforward realizations of data-leak detection require the plaintext sensitive data. However, this requirement is undesirable, as it may threaten the confidentiality of the sensitive information. If a detection system is compromised, then it may expose the plaintext sensitive data. In addition, the data owner may need to outsource the data-leak detection to providers, but may be unwilling to reveal the plaintext sensitive data to them. Therefore, one needs new data-leak detection solutions that allow the providers to scan content for leaks without learning the sensitive information.

We propose a data-leak detection solution which can be outsourced and be deployed in a semi honest detection environment. We design, implement, and evaluate our watermarking technique that enhances data privacy during data-leak detection operations. Statistics from security firms, research institutions and government organizations show that the number of data-leak instances has grown rapidly in recent years. Among various data-leak cases, human mistakes are one of the main causes of data loss. There exist solutions detecting inadvertent sensitive data leaks caused by human mistakes and to provide alerts for organizations.

A common approach is to screen content in storage and transmission for exposed sensitive information. Such an approach usually requires the detection operation to be conducted in secrecy. However, this secrecy requirement is challenging to satisfy in practice, as detection servers may be compromised or outsourced. In this project, we identify the privacy needs in an outsourced data-leak detection service and provide a systematic solution to enable privacy-preserving DLD services with watermarking technique. The advantage of our method is that it enables the data owner to safely delegate the detection operation to a semi-honest provider without revealing the sensitive data to the provider.

II. SCOPE OF THE PROJECT

In our project, we identify the privacy needs in an outsourced data-leak detection service and provide a systematic solution to enable privacy-preserving DLD services with watermarking technique. Data leakers are detected using watermarking technique, 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, consider the option of adding "fake" objects to the distributed set. Such objects do not correspond to real entities but appear realistic to the leakers. 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 a leaker was given one or more fake objects that were leaked, then the data owner can be more confident about the leaker.

III. RELATED WORK

We proceed into taxonomy of previous solutions in the area of privacy preserving data analysis. We start with more generic solutions and we further describe previous work in the context of specific privacy preserving similarity detection algorithms for clustering.

Data perturbation several techniques have been proposed in order to obfuscate data such that when users submit their data to the data analyzer individual data privacy is being protected but specific data mining algorithms can be applied on it. Privacy preserving data mining by adding noise on data has been first proposed in ([5], [6]). The solution has been proposed for privacy preserving decision trees as a solution to derive association rules from databases. In [7] the authors proposed geometrical transformation for data clustering. Transformation though are data dependent and do not scale for multidimensional data.

Anonymization Data anonymization asks for unlink ability on data records and users. K-anonymity [8] has been proposed as a solution to protect the release of data to an untrusted party such that the personal private information for each data record cannot be distinguished from k−1 other users. Suppression and generalization are two techniques to achieve k-anonymity. By generalization [9] specific attributes are generalized in order to protect user anonymity. For instance instead of releasing the exact date of birth only the month and the year is released. With suppression [10] no data is released. Solutions for data anonymity imply an information loss throughout the described techniques and operation after the release of the data is inconsistent.

Data separation in [11] cryptographic tools are being used to protect user data privacy when the id3 tree is constructed for association rules. The id3 tree is a widely known technique for data classification. The categorical data of a set of records is being constructed by choosing the attributes than containing the higher information gain. Information gain is expressed as conditional entropy and the problem of id3 construction is approximated by finding the attributes that information gain is maximized. The authors assume that data are split horizontally, thus the data analyzer in order to compute the conditional entropy of two users should separately and privately obtain the data from both. It turns out that information gain for an attribute between two users is expressed as $(u1+u2)\cdot\log(u1+ u2)$. The problem has been addressed as a secure multi-party computation of this expression for two users.

IV. SYSTEM ANALYSIS

In a broad sense, a general methodology (not a fixed set of techniques) that applies a'systems'or 'holistic' perspective by taking all aspects of the situation into account, and by concentrating on the interactions between its different elements. It provides a framework in which judgments of the experts in different fields can be combined to determine what must be done, and what is the best way to accomplish it in light of current and future needs. Although closely associated with data or information processing, the practice of SA has been in existence since long before computers were invented.

A. Exisiting Work

Deliberately planned attacks, inadvertent leaks, and human mistakes lead to most of the data-leak incidents. Detecting and preventing data leaks requires a set of complementary solutions. A privacy preserving data-leak detection (DLD) solution introduced to solve the data leakage problem. But the leaker can't be identified. It is very hard for any system administrator to trace out the data leaker among the system users. It creates a lot many ethical issues in the working environment of the office. The evaluation results show that our method can support accurate detection with very small number of false alarms under various data-leak scenarios.

B. Drawbacks of Existing System

- Small number of false alarms under various data-leak scenarios.
- Data leaker can't be identified.
- Data leakage
- Increases collisions for fuzzification purpose

C. Proposed Work

In our proposed work, data leakers are detected using watermarking technique, 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, consider the option of adding “fake” objects to the distributed set. Such objects do not correspond to real entities but appear realistic to the leakers. 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 a leaker was given one or more fake objects that were leaked, then the data owner can be more confident about the leaker.

D. Advantages of Proposed System

- Data leaker can be identified.
- Achieve Security and Privacy of data
- Achieve user privacy
- Reduce Data leakage

V. SYSTEM ARCHITECTURE

System architecture is a conceptual model that defines the structure, behaviour, and more views of a system. An architecture description is a formal description and representation of a system, organized in a way that supports reasoning about the structures and behaviours of the system.

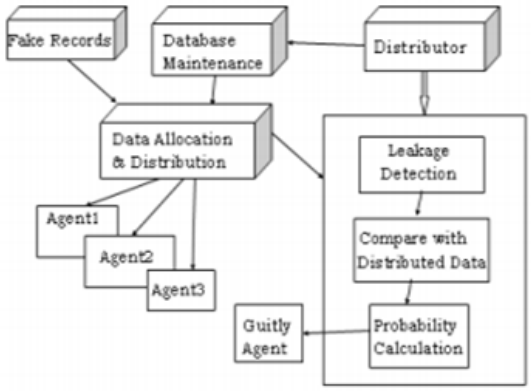

Fig. 1 System Architecture

VI. MODULES DESCRIPTION

A module is a part of a program. Programs are composed of one or more independently developed modules that are not combined until the program is linked. A single module can contain one or several routines. Our project modules are given below:

- Data Allocation Module
- Fake Object Module
- Optimization Module
- Data Distributor Module

A. Data Allocation Module

The main focus of our project is the data allocation problem as how can the distributor "intelligently" give data to agents in order to improve the chances of detecting a guilty agent, Admin can send the files to the authenticated user, users can edit their account details etc. Agent views the secret key details through mail. In order to increase the chances of detecting agents that leak data.

B. Fake Object Module

The distributor creates and adds fake objects to the data that he distributes to agents. Fake objects are objects generated by the distributor in order to increase the chances of detecting agents that leak data. The distributor may be able to add fake objects to the distributed data in order to improve his effectiveness in detecting guilty agents. Our use of fake objects is inspired by the use of "trace" records in mailing lists. In case we give the wrong secret key to download the file, the duplicate file is opened, and that fake details also send the mail. Ex: The fake object details will display.

C. Optimization Module

The Optimization Module is the distributor's data allocation to agents has one constraint and one objective. The agent's constraint is to satisfy distributor's requests, by providing them with the number of objects they request or with all available objects that satisfy their conditions. His objective is to be able to detect an agent who leaks any portion of his data. User can able to lock and unlock the files for secure.

D. Data Distributor Module

A data distributor has given sensitive data to a set of supposedly trusted agents (third parties). Some of the data is leaked and found in an unauthorized place (e.g., on the web or somebody's laptop). 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 Admin can able to view the which file is leaking and fake user's details also.

VII. EVALUATION

We demonstrate the correctness of our protocol through an experimental evaluation setup. We obtain data originating from a personality experiment. We first cluster the data based on cosine similarity using a well known clustering algorithm. The same clustering algorithm is further applied over the encryption of the same data using φ which has already described combines rotation and random scaling.

A. Data Set The dataset contains an extract of the results from the Foursquare Personality Experiment1 which uses the mobile social network Foursquare2 combined with a standard personality test to allow the link between personality (as defined by the five-factor model and the places that people visit to be examined. To the best of our knowledge, this is the first time that it has been possible to correlate personality with place on such a granular level.

When accessing the experiment, users sign in using their Foursquare account, allowing us to access the list of venues which they have 'checked in' to on the Foursquare service. We access only this list, storing the venues that the user has been to and the number of times they have visited each venue, but without accessing or storing the information about the individual checking - we do not store when each visit to the venue occurred, nor the order in which venues were visited. Once users have accessed the system they then take a 44-item personality test revealing their five-factor personality scores. The five-factor model gives each person a score between 1 and 5 for each of the five personality traits: Openness, Conscientiousness, Extraversion, Agreeableness and Neuroticism. The users participating in the study are a self-selecting group comprised of 173 people who both use foursquare online location based tagging system and are willing to take part in a personality-based experiment.

C. Simulation Setup and Results We apply the hierarchical algorithm over the personality dataset with the complete linkage metric and based on cosine similarity. The data consists of 173 different 5 dimensional vectors describing users' personality with respect to the 5 personality traits as previously described. We did not include venue visits frequency since we believe that personality traits are considered much more sensitive data compared to location information and that users would be more interested in hiding such information. We consider similarity on 3 subvectors per user data: the subvectors are constructed with the (1st , 2 nd),(3rd , 4 th) and (1st , 5 th) coordinates of the original vector respectively. Any pair wise subvector could be having chosen such that the union of the set of subvectors entails all the coefficients. The main similarity metric is computed as the average of the similarities between subvectors. In order to protect their privacy, every user chooses a random scaling factor per two dimensions. After the random scaling process users apply the rotation operation to their partially obfuscated subvectors.

VIII. CONCLUSION

From this study we conclude that the data leakage detection system model is very useful as compare to the existing model. We can provide security to our data during its distribution or transmission and even we can detect if that gets leaked and who is the leaker. Thus, using this model security as well as tracking system is developed. Watermarking provide security using various algorithms through encryption, whereas this model provides security plus detection technique. This model is very helpful in various industries, where data is distribute through any public or private channel and shared with third party. Now, industry & various offices can rely on this security & detection model.

ACKNOWLEDGMENT

I would like to express my thanks to the people who helped me most throughout my project. I am grateful to my guide for nonstop for the project. Special thanks of mine goes to my project members who helped me out in completing my project, where they all exchanged their own ideas, thoughts and made this possible to complete my project with all accurate information. I wish to thank my parents of their personal support or attention who inspired me to go my own way. At last but not the least I want to thank my friends who treasured me for my hard work and encouraged me and finally to God who made all the things possible for me till the end.

REFERENCES

[1] X. Shu and D. Yao, "*Data leak detection as a service*," in Proc. 8th Int.Conf. Secur. Privacy Commun. Netw., 2012, pp. 222–240.

[2] Risk Based Security. (Feb. 2014). Data Breach Quick-View: An Executive's Guide to 2013 Data Breach Trends. [Online]. Available: https://www.riskbasedsecurity.com/reports/2013-DataBreachQuickView.pdf,accessed Oct. 2014.

[3] Ponemon Institute. (May 2013). 2013 Cost of Data Breach Study: Global Analysis. [Online]. Available: https://www4.symantec.com/mktginfo/whitepaper/053013_GL_NA_WP_Ponemon-2013-Cost-of-a-Data-Breach-Report_daiNA_cta72382.pdf, accessed Oct. 2014.

[4] Identity Finder. Discover Sensitive Data Prevent Breaches DLP Data Loss Prevention. [Online]. Available: http://www.identityfinder.com/,accessed Oct. 2014.

[5] R. Agrawal and R. Srikant, "*Privacy-preserving data mining*," 2000.

[6] D. Agrawal and C. C. Aggarwal, "*On the design and quantification of privacy preserving data mining algorithms,*" in Proceedings of the twentieth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems, ser. PODS '01. New York, NY, USA: ACM, 2001, pp. 247–255.

[7] S. R. M. Oliveira and O. R. Za¨ıane, "*Privacy preserving clustering by data transformation*," JIDM, vol. 1, no. 1, pp. 37–52, 2010.

[8] L. Sweeney, "*k-anonymity: a model for protecting privacy*," Int. J. Uncertain. Fuzziness Knowl.-Based Syst., vol. 10, no. 5, pp. 557–570, Oct. 2002.

[9] V. S. Iyengar, "*Transforming data to satisfy privacy constraints*," in Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, ser. KDD '02. New York, NY, USA: ACM, 2002, pp. 279–288.

[10] P. Samarati and L. Sweeney, "*Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression,*" Tech. Rep., 1998.

[11] Y. Lindell and B. Pinkas, "*Privacy preserving data mining*," in CRYPTO, 2000, pp. 36–54.

[12] M. Kantarcioglu and C. Clifton, "*Privacy-preserving distributed mining of association rules on horizontally partitioned data*," IEEE Trans. on Knowl. And Data Eng., vol. 16, no. 9, pp. 1026–1037, Sep. 2004. [Online]. Available: http://dx.doi.org/10.1109/TKDE.2004.4

[13] C. Clifton, M. Kantarcioglu, J. Vaidya, X. Lin, and M. Y. Zhu, "*Tools for privacy preserving distributed data mining,"* ACM SIGKDD Explorations, vol. 4, p. 2003, 2003.

[14] S. R. M. Oliveira and et al., "*Privacy-preserving clustering by object similarity-based representation and dimensionality reduction transformation*," in in proc. of the workshop on privacy and security aspects of data mining (psadm04) in conjunction with the fourth ieee international conference on data mining (icdm04, 2004, pp. 21–30.

[15] B. Goethals, S. Laur, H. Lipmaa, and T. Mielikainen, "*On private ¨ scalar product computation for privacy-preserving data mining,"* in Proceedings of the 7th international conference on Information Security and Cryptology, ser. ICISC'04. Berlin, Heidelberg: Springer-Verlag, 2005, pp. 104–120.

