National Conference on "Advanced Technologies in Computing and Networking"-ATCON-2015
Special Issue of International Journal of Electronics, Communication & Soft Computing Science and Engineering, ISSN: 2277-9477

Understanding the Importance of Data Leakage Detection with Distribute techniques

Shraddha R. Shriwastav **Prof. Parag R. Kaveri** **Prof. Umesh Gandhi**

Abstract- **Early, A distributor when pass their data to the client by agent. In a way some of these data is leaked and found in an authorized place. (eg: on the assess the probably cover that the leaked data came from one or more agents as web, or somebody laptop) The distributor must likely hood that the leaked data came from one or more agents as opposed to having been independently gathered by other means. conventionally, for the protection of data presented model used 'Watermarking' techniques but this technique doesn't provide complete security against the data leakage. So, for properly detection the leakage of setoff objects or record. Model used a 'unobtrusive' technique.**

In this presented model, different cases is been consider for finding the importance of Data Leakage Detection term.

I. INTRODUCTION

Early, Data leakage is the big dare in front of the industries & different institutes. Though there are number of systems designed for the data security by using different encryption algorithms, there is a big issue of the honor of the users of those systems. It is very hard for any system administrator to trace out the data leaker among the system users. It creates a lot many moral issues in the working environment of the office.

A 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 places(web or somebody's laptop). The distributor must assess the likely hood that the leaked data came from one or more agents, as opposed to having been independently gathered by other means Model propose data allocation strategies (across the agents) that improve the probability of identifying leakages. These methods do not rely on alterations of the released data (e.g., watermarks). In some cases presented model can also inject "realistic but fake" data records to further improve our chances of detecting leakage and identifying the guilty party.

II. A REVIEW ON DATA LEAKAGE DETECTION

The main focus of open paper is the data allocation problem we address, depending on the type of data request made by agents and whether "fake objects" are allowed. The idea of disturbing data to detect leakage is not new. However, in most cases, individual objects are troubled, e.g., by adding random noise how can the distributor "brightly" give data to agents in order to improve the chances of detecting a guilty agent? there are four instances of this to sensitive salaries, or adding a watermark to an image. In case, presented model are disturbing the set of distributor objects by adding fake elements. In some applications, fake objects may cause fewer problems that disturbing real objects. For example, say the distributed data objects are medical records and the agents are hospitals. In this case, even small modifications to the records of actual patients may be unwanted. However, the addition of some fake medical records may be acceptable, since no patient matches these records, and hence no one will ever be treated based on fake records. The use of fake objects is inspired by the use of "trace" records in mailing lists. In this case, company A sells to company B a mailing list to be used once (e.g., to send advertisements). Company A adds trace records that contain addresses owned by company A. Thus, each time company B uses the purchased mailing list, A receives copies of the mailing. These records are a type of fake objects that help identify improper use of data. conventionally, leakage detection is handled by watermarking, e.g., a unique code is set in each distributed copy. If that copy is later exposed in the hands of an not permitted party, the leaker can be identified. Watermarks can be very useful in some cases, but again, engage some modification of the original data. Also, watermarks can sometimes be destroyed if the data recipient is hateful.

At this point, presented model develop a model for judge the "fault" of agents. It also present algorithms for distributing objects to agents, in a way that improves that chances of identifying a leaker.

III. IMPLEMENTED SYSTEM

The goal is to detect when the distributor's sensitive data has been leaked by agents, and if possible to identify the agent that leaked the data. Presented model consider applications where the original sensitive data cannot be disturbed. Is a very useful technique where the data is modified and made "less responsive" before being handed to agents. For example, one can add random noise

to certain attributes, or one can replace exact values by ranges . However, in some cases it is important not to alter the original distributor's data. For example, if an outsourcer is doing our payroll, he must have the exact salary and customer bank account numbers. If medical researchers will be treating patients they may need accurate data for the patients.

This presented paper develop some new techniques and with the help of this technique model keep on safe his data. With some algorithms it assessing the "fault" of agents. Presented model also present algorithms for distributing objects to agents in a way that improves its chances of identifying a leaker. Finally, model also consider the option of adding "fake" objects to the distributed set.

Such objects do not correspond to real entities but appear realistic 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 confident that agent was in the wrong.

IV. Comparison of Existing System and Implemented System

Existing System:

Conventionally, watermarking is the technique which handles leakage detection .Such as, unique code is set in each distributed copy. If that copy is later open in the hands of an not permitted party, the leaker can be identified. In some cases watermark technique is very useful, but again, some change will be involved in original data. Also, watermarks can sometimes be destroyed if the data recipient is hateful. E.g. 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. They call the owner of the data the distributor and the supposedly trusted third parties the agents.

Implemented System:

Presented model objective is to detect when the distributor's responsive data has been leaked by agents, and if probable to identify the agent that leaked the data. Perturbation is a very useful technique where the data is modified and made "less responsive" before being handed to agents. Model develop modest techniques for detecting leakage of a set of objects or records.

Now, presented model develop a model for assessing the "guilt" of agents. They also present algorithms for distributing objects to agents, in a way that improves his chances of identifying a leaker. In conclusion, They also consider the option of adding "fake" objects to the distributed set. Such objects do not correspond to real entities but appear realistic 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 turn out an agent was given one or more fake objects that were leaked, then the distributor can be more confident that agent was in the wrong.

Problem Setup and Notation:

A distributor owns a set $T=\{t_{1,\ldots,}t_m\}$of valuable data objects. The distributor wants to share some of the objects with a set of agents $U_1,U_2,\ldots U_n$, but does not wish the objects be leaked to other third parties. The objects in T could be of any type and size, e.g., they could be tuples in a relation, or relations in a database. An agent Ui receives a subset of objects, determined either by a sample request or an explicit request:

4.1 Explicit Data Request

In case of explicit data request with fake not allowed, the distributor is not allowed to add fake Data objects to the distributed data. So Data allocation is fully defined by the agent's data request. In case of explicit data request with fake allowed, the distributor cannot remove or alter the requests from the agent. However distributor can add the fake object

4.2 Sample Data Request

With sample data requests, each agent may receive any from a subset out of different ones. Hence, there are different allocations. In every allocation, the distributor can permute objects and keep the same chances of guilty agent detection. The reason is that the guilt probability depends only on which agents have received the leaked objects and not on the identity of the leaked objects. Therefore, from the distributor' perspective there are different allocations.

V. Guilt Model Analysis

In presented model parameters interact and to check if the interactions match our intuition, in this presented model study two simple scenarios as Impact of Probability p and Impact of Overlap between Ri and S. In each scenario we have a target that has obtained all the distributor's objects, i.e., T = S.

In each scenario model have a target that has obtained all the distributor's objects, i.e., T To compute this P r{Gi |S}, model need an estimate for the probability that values in S can be "guessed" by the target. For instance, say

some of the objects in S are emails of individuals. Model can conduct an experiment and ask a person with distributor may be able to add fake objects to the distributed data in order to improve his effectiveness in S. detecting guilty agents. However, fake objects may impact the correctness of what agents do, so they may not always be allowable approximately the expertise and resources of the target to find the email of say 100 individuals. If this person can find say 90 emails, then model can reasonably guess that the probability of finding one email is 0.9. On the other hand, if the objects in question are bank account numbers, the person may only discover say 20, leading to an estimate of 0.2. We call this estimate pt, the probability that object t can be guessed by the target. Probability pt is analogous to the probabilities used in designing fault-tolerant systems. That is, to estimate how likely it is that a system will be operational throughout a given period, we need the probabilities that individual components will or will not fail. A component failure in our case is the event that the target guesses an object of S. The component failure is used to compute the overall system reliability, while we use the probability of guessing to identify agents that have leaked information. The component failure probabilities are estimated based on experiments, just as Presented model propose to estimate the pt' s. Similarly, the component probabilities are usually conservative estimates, rather than exact numbers. For example, say model use a component failure probability that is higher than the actual probability, and it design his system to provide a desired high level of reliability. Then it will know that the actual system will have at least that level of reliability, but possibly higher. In the same way, model use pts that are higher than the true values, it will know that the agents will be guilty with at least the computed probabilities.

VI. ALGORITHM ANALYSIS

In order to see how presented model parameters interact and to check if the interactions match our intuition, and study two simple scenarios. In each scenario model have a target that has obtained all the distributor' s objects, i.e., T =S.

Algorithms:

1. Evaluation of Explicit Data Request Algorithms

In the first place, the goal of these experiments was to see whether fake objects in the distributed data sets yield significant improvement the chances of detecting a guilty agent. In the second place, it wanted to evaluate his e-optimal algorithm relative to a random allocation.

2. Evaluation of Sample Data Request Algorithms

With sample data requests agents are not interested in particular objects. Hence, object sharing is not explicitly defined by their requests. The distributor is "forced" to allocate certain objects to multiple agents only if the number of requested objects exceeds the number of objects in set T. The more data objects the agents request in total, the more recipients on average an object has; and the more objects are shared among different agents, the more difficult it is to detect a guilty agent.

Modules:

1. Data Allocation Module:

The main focus of presented model 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.

2. 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. Presented model use of fake objects is inspired by the use of "trace" records in mailing lists. In case model 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.

3. 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.

4. Data Distributor:

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. CASE STUDIES

In this related to presented model, here we take some cases:

Case1:

Whenever, A student pass their data to the B student. And expect that his data will be secure on hand to the B student. But, when A students data will leaked and found in an illegal or not permitted place. At that time "Watermarking" technique somewhere protected the leaked data. But this technique doesn't provide full security of that data. For securing the data. Here we presented implemented model will help us with the use some algorithm And with help of this algorithm A student will securely and faithfully send this data to B student.

Case2:

Another case is Telecom company, whenever A client fill up the all information into the online form and submit this form to the telecom company. But, if in case office employee will be your form submit head department with doing some mistakes. At that time you don't have any problem. But, if in case your telephone will blocked and you go to the telecom office and give complaint about your telephone and then telecom company do not give any response to you because of the interchanging and leaked your data with the other person. And here not only you have in problem but also that other person also in problem. So, here also we used implemented model.

Case3:

Also another case is of Cyber Café, is the very general place in our today's life. Because it is very kind of help us. Whenever, we download any documents through the internet and copy into our pen drive or memory card etc. but if we doesn't deleted that downloaded documents into the PC's desktop. Then that document will be leaked and some unknown person will takes it benefit. Here also we used implemented model to prevent our document.

VIII. CONCLUSION

From this study we conclude that the data leakage detection system model is very useful as compare to the existing watermarking model. Presented model can provide security to our data during its distribution or transmission and even model can detect if that gets leaked. Thus, using this model security as well as tracking system is developed. Watermarking can just 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 shred with third party. Now, industry & various offices can rely on this security & detection model.

XI. ACKNOWLEDGEMENT

For all the efforts behind the paper work, I first & foremost would like to express my sincere appreciation to the staff of Dept. of Computer Sci.& Tech . for their extended help & suggestions at every stage of this paper. It is with a great sense of gratitude that I acknowledge the support, time to time suggestions and highly indebted to my guide Prof. P.R. Kaveri (my project guide), and Dr. Deshpande (HOD). Finally, I pay sincere thanks to my parents and all those who indirectly and directly helped me towards the successful completion of the paper.

REFERENCES

1.Miss Shraddha R. Shriwastav (Dept of Comp.sci and Tech,.H.V.P.Mcollege,S.G.B. University, Amravati(M.S) India."Understanding the Importance of Data Leakage Detection Method"2015

2. Sandip A.Kale, Prof. Kulkarni S.V. (Department Of Computer Sci.&Engg,MIT College of Engg, Dr.B.A.M.University, Aurangabad(M.S),India, Data Leakage Detection: A Survey, (IOSR Journal of ComputerEngineering (IOSRJCE)ISSN : 2278-0661 Volume 1, Issue 6 (July-Aug2012), PP 32-35 www.iosrjournals.org

3. Sruthi Patil1International Journal Of Engineering And Computer Science 2:2 Feb 2013(395-399)

4. Naresh Bollam,Mr.V.Malsoru/ International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com Vol. 1, Issue 3, pp.1088-1091

5. Chandni Bhatt et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (2) , 2014, 2556-2558