ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Data Leakage Detection Using Cloud Computing

Mohammed Noorullah
M.Tech Student
Dept. of CSE
Avanthi Institute of Engineering and Technology, Vizianagaram.

Mr K.Mehar Prasad
Assistant Professor
Dept. of CSE
Avanthi Institute of Engineering and Technology, Vizianagaram.

Abstract

Data leakage is the big challenge 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 integrity 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 ethical issues in the working environment of the office. The data leakage detection industry is very heterogeneous as it evolved out of ripe product lines of leading IT security vendors. A broad arsenal of enabling technologies such as firewalls, encryption, access control, identity management, machine learning content/context-based detectors and others have already been incorporated to offer protection against various facets of the data leakage threat. The competitive benefits of developing a "one -stop-shop", silver bullet data leakage detection suite is mainly in facilitating effective orchestration of the aforementioned enabling technologies to provide the highest degree of protection by ensuring an optimal fit of specific data leakage detection technologies with the "threat landscape" they operate in. This landscape is characterized by types of leakage channels, datastates, users, and IT platforms.

Introduction

In this paper, we develop a model for finding the guilty 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. 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. We also consider optimization in which leaked data is compared with original data and accordingly the third party who leaked the data is guessed. We will also be using approximation technique to encounter guilty agents. We proposed one model that can handle all the requests from customers and there is no limit on number of customers. The model gives the data allocation strategies to improve the probability of identifying leakages. Also there is application where there is a distributor, distributing and managing the files that contain sensitive information to users when they send request. The log is maintained for every request, which is later used to find overlapping with the leaked file set and the subjective risk and for Assessment of guilt probability.

Data leakage happens every day when confidential business information such as customer or patient data, source code or design specifications, price lists, intellectual property and trade secrets, and forecasts and budgets in spreadsheets are leaked out. When these are leaked out it leaves the company unprotected and goes outside the jurisdiction of the corporation. This uncontrolled data leakage puts business in a vulnerable position. Once this data is no longer within the domain, then the company is at serious risk.

When cybercriminals ―cash out or sell this data for profit it costs our organization money, damages the competitive advantage, brand, and reputation and

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

destroys customer trust. To address this problem, we develop a model for assessing the ―guilt‖ of agents. The distributor will ―intelligentl‖y give data to agents in order to improve the chances of detecting a guilty agent like adding the fake objects to distributed sets. 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. If the distributor sees enough evidence that an agent leaked data then they may stop doing business with him, or may initiate legal proceedings. Mainly it has one constraints and one objective. The Distributor's constraint satisfies the agent, by providing number of object they request that satisfy their conditions.

LITERATURE SURVEY

The guilt detection approach we present is related to the data provenance problem [3]: tracing the lineage of S objects implies essentially the detection of the guilty agents. And assume some prior knowledge on the way a data view is created out of data sources. Objects and sets is more general .As far as the data allocation strategies are concerned; our work is mostly relevant to watermarking that is used as a means of establishing original ownership of distributed objects. [3] Finally, there are also lots of other works on mechanisms that allow only authorized users to access sensitive data through access control policies [9], [2]. Such approaches prevent in some sense data leakage by sharing information only with trusted parties. However, these policies are restrictive and may make it impossible to satisfy agent's requests. Maintaining the Integrity of the Specifications

EXISTING SYSTEM:

Traditionally, 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. E.g . A hospital may give patient records to rese archers 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

PROPOSED SYSTEM:

Our 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. Perturbation is a very useful technique where the data is modified and made "less sensitive" before being handed to agents. we develop unobtrusive techniques for detecting leakage of a set of objects or records.In this section 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 dentifying a leaker. Finally, we 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 atermark 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 guilty.

METHODOLOGY

Problem Definition

The distributor owns the sensitive data set T= {t1, t2… tn}. The agent Ai request the data objects from distributor. 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. The distributor gives the subset of data to each agent, after giving objects to agents; the distributor discovers that a set L of T has leaked. This means some third party has been caught in possession of L. The agent Ai receives a subset Ri of objects T

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

determined either by implicit request or an explicit request. Implicit Request Ri = Implicit (T, mi) : Any subset of mi records from T can be given to agent Ai Explicit Request Ri = Explicit (T, Condi) : Agent Ai receives all T objects that satisfy Condition.

Data Allocation Module

The distributor may be able to add fake objects to the distributed data in order to improve his effectiveness in detecting guilty agents. However, fake objects may impact the correctness of what agents do, so they may not always be allowable. Our use of fake objects is inspired by the use of ―trac‖e re cords 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. The distributor creates and adds fake objects to the data that he distributes to agents. Depending upon the addition of fake tuples into the agent's request, data allocation problem is divided into four cases as:

i. Explicit request with fake tuples (EF)
ii. Explicit request without fake tuples (E~F)
iii. Implicit request with fake tuples (IF)
iv. Implicit request without fake tuples (I~F).

Implicit Request Ri = Implicit (T, mi) : Any subset of mi records from T can be given to agent Ai

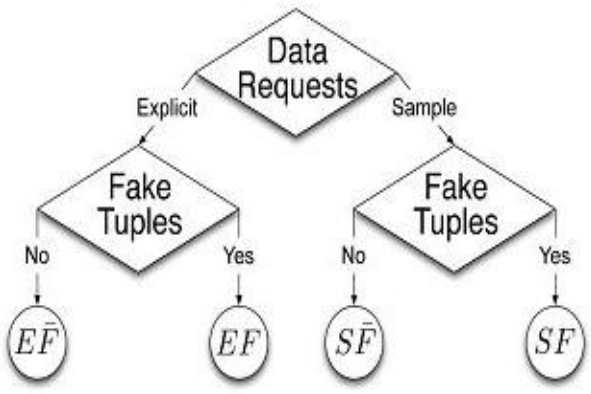

Fig: Leakage problem instances

Optimization Module

The distributor's data allocation to agents has one constraint and one objective. The distributor's constraint is to satisfy agents' 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. The objective is to maximize the chances of detecting a guilty agent that leaks all his data objects. The Pr { Gj|S =Ri } or simply Pr {Gj |Ri } is the probability that agent is guilty if the distributor discovers a leaked table S that contains all objects .

Let the distributor have data request from n agents. The distributor wants to give tables R1 ,R2……..Rn to agents A1 ,A2…… An respectively, so that Distribution satisfies agent's request; and Maximizes the guilt probability differences

$\Delta(i,j)$ for all i, j= 1, 2, ……n and $i \neq j$.

$$\text{maximize}(\text{over} R1\ldots,Rn)\ (\ldots,\Delta(i,j),\ldots)\ i\neq j\ldots\ldots\ldots(A)$$

$$\text{minimize}(\text{over}\ \ R1,\ldots,Rn)\ \ (\ldots,|Ri\cap Rj|\div|Ri|,\ldots)\ i\neq j$$

Guitl Model Assessment

Let L denote the leaked data set that may be leaked intentionally or guessed by the target user. Since agent having some of the leaked data of L, may be susceptible for leaking the data. But he may argue that he is innocent and that the L data were obtained by target through some other means. Our goal is to assess the likelihood that the leaked data came from the agents as opposed to other resources. E.g. if one of the object of L was given to only agent A1, we may suspect A1 more. So probability that agent A1 is guilty for leaking data set L is denoted as Pr{Gi| L} .

Algorithm1:

Allocation of Data Explicitly:

Input: -

i. T= {t1, t2, t3, .tn}-Distributor's Dataset
ii. R- Request of the agent
iii. Cond- Condition given by the agent
iv. m= number of tuples given to an agent m<n, selected randomly

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Output: - D- Data sent to agent
1. D=Φ, T′=Φ
2. For i=1 to n do
3. If(t .fields==cond) then
4. T′=T′U{ t i}
5. For i=0 to i<m do
6. D=DU{ti}
7. T′=T′-{ ti}
8. If T′=Φ then
9. Goto step 2
10. Allocate dataset D to particular agent
11. Repeat the steps for every agent

To improve the chances of finding guilty agent we can also add the fake tuples to their data sets.

Algorithm2:
Addition of fake tuples:
Input:
i. D- Dataset of agent
ii. F- Set of fake tuples
iii. Cond- Condition given by agent
iv. b- number of fake objects to be sent.

Output:- D- Dataset with fake tuples
1. While b>0 do
2. f= select Fake Object at random from set F
3. D= DU {f}
4. F= F-{f}
5. b=b-1

Similarly, we can distribute the dataset for implicit request of agent. For implicit request the subset of distributor's dataset is selected randomly. Thus with the implicit data request we get different subsets. Hence there are different data allocations. An object allocation that satisfies requests and ignores the distributor's objective to give each agent unique subset of T of size m. The s-max algorithm allocates to an agent the data record that yields the minimum increase of the maximum relative overlap among any pair of agents. The s-max algorithm is as follows:
1. Initialize Min_Overlap, the minimum out of the minimum relative overlaps that the allocations of different objects to Ai
2. for k do Initialize max_rel_ov←0, the maximum relative overlap between Ri the allocation of tk to Ai
3. for all j=1,……,n:j=I and tkЄRj do calculate absolute overlap as abs_ov← calculate relative overlap as rel_ov←abs_ov/min(mi, mj)
4. Find maximum relative overlap as Max_rel_ov←MAX(max_rel_ov, rel_ov) If max_rel_ov≤ min_ov then Min_ov←max_rel_ovret_k←k Return ret_k The algorithm presented implements a variety of data distribution strategies that can improve the distributor's chances of identifying a leaker. It is shown that distributing objects judiciously can make a significant difference in identifying guilty agents, especially in cases where there is large overlap in the data that agents must receive.

CONCLUSION

The likelihood that an agent is responsible for a leak is assessed, based on the overlap of his data with the leaked data and the data of other agents, and based on the probability that objects can be "guessed" by other means. The algorithms we have presented implement a variety of data distribution strategies that can improve the distributor's chances of identifying a leaker. We have shown that distributing objects judiciously can make a significant difference in identifying guilty agents, especially in cases where there is large overlap in the data that agents must receive.

REFERENCES

[1] Papadimitriou P, Garcia-Molina H. A Model For Data Leakage Detection// IEEE Transaction On Knowledge And Data EngineeringJan.2011.

[2] International Journal of Computer Trends and Technology- volume3Issue1- 2012 ISSN:2231-2803

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

http://www.internationaljournalssrg.org Data Allocation Strategies for Detecting Data LeakageSrikanthYadav, Dr. Y. Eswararao, V. ShanmukhaRao, R. Vasantha

[3] International Journal of Computer Applications in Engineering Sciences [ISSN: 2231-4946]197 | P a g e Development of Data leakage Detection Using Data Allocation Strategies Rudragouda G PatilDept of CSE,The Oxford College of Engg, Bangalore.

[4] P. Buneman, S. Khanna and W.C. Tan. Why and where: Acharacterization of data provenance. ICDT 2001, 8th International Conference, London, UK, January4-6, 2001,Proceedings, volume 1973 of Lecture Notes in Computer Science, Springer, 2001

[5] S. Jajodia, P. Samarati, M.L. Sapino, and V.S. Subrahmanian, ―Flexible Support for Multiple Access Control Policlies ,A C M Trans. D atabase Systems, vol. 26, no. 2, pp. 214-260, 2001.

[6] P. Bonatti, S.D.C. di Vimercati, and P. Samarati, ― An A lgebra f or Composing Access Control Policies,‖ ACM Trans. Information scientificcommons and System Security, vol. 5, no. 1, pp. 1-35, 2002.

[7] YIN Fan, WANG Yu, WANG Lina, Yu Rongwei A Trustworthiness- Based Distribution Model for Data Leakage Detection: Wuhan University Journal Of Natural Sciences.

[8] RakeshAgrawal, Jerry Kiernan. Watermarking Relational Databases// IBM Almaden Research Center.

[9] L. Sweeney, ―Achieving K-Anonymity Privacy Protection Using Generalization ‖ And Suppression, http://en.scientificcommons.org/43196131,2002

[10] E dward P . H olden, J ai W . K ang, G eoffrey R . A nderson, D ianne P . Bills, Databases in the Cloud: A Work in Progress,2012.

Author Details

Mohammed Noorullah
M.Tech Student
Dept. of CSE
Avanthi Institute of Engineering and Technology,
Vizianagaram.

Mr K.Mehar Prasad
Assistant Professor
Dept. of CSE
Avanthi Institute of Engineering and Technology,
Vizianagaram.