Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 2, Issue. 4, April 2013, pg.577 – 581

REVIEW ARTICLE

A Review on Data Leakage Detection

SHAJ.V[1], K.P. KALIYAMURTHIE[2]
[1]Department of Information Technology, Bharath University, India
[2]Department of Information Technology, Bharath University, India

Abstract— This paper contains concept of data leakage, its causes of leakage and different techniques to protect and detect the data leakage. The value of the data is incredible, so it should not be leaked or altered. In the field of IT, huge database is being used. This database is shared with multiple people at a time. But during this sharing of the data, there are huge chances of data vulnerability, leakage or alteration. So, to prevent these problems, a data leakage detection system has been proposed. This paper includes brief idea about data leakage detection and a methodology to detect the data leakage persons.

Key Terms: - IT; watermarking guilty agent; explicit data; DLP (data leakage prevention)

I. INTRODUCTION

Data leakage is defined as the accidental or unintentional distribution of private or sensitive data to unauthorized entity. Sensitive data of companies and organizations includes intellectual property (IP), financial information, patient information, personal credit-card data, and other information depending on the business and the industry. Furthermore, in many cases, sensitive data is shared among various stakeholders such as employees working from outside the organizational premises (e.g., on laptops), business partners and customers.

This increases the risk of confidential information falling into unauthorized hands. Whether caused by malicious intent, or an inadvertent mistake, by an insider or outsider, exposed sensitive information can seriously hurt an organization. The potential damage and adverse consequences of a data leak incident can be classified into the following two categories: direct and indirect loss. Direct loss refers to tangible damage that is easy to measure and estimate quantitatively. Indirect loss, on the other hand, is much harder to quantify and has a much broader impact in terms of cost, place and time [Bunker, 2009]. Direct loss includes violating regulations (such as those protecting customer privacy) resulting in fine/settlement/customer compensation fees; litigation of lawsuits; loss of future sales; costs of investigation and remedial/restoration fees. Indirect loss includes reduced share-price as a result of the negative publicity; damage to company's goodwill and reputation; customer abandonment; and exposure of Intellectual Property (business plans, code, financial reports, and meeting agendas) to competitors.

II. HOW WAS ACCESS TO THE DATA GAINED?

The "How was access to the data gained?" attribute extends the “Who caused the leak?” attribute.

These attributes are not interchangeable, but rather complementary and the various ways to gain access to sensitive data can be clustered into the following groups.

The classification by leakage channel is important in order to know how the incidents may be prevented in the future and can be classified as physical or logical.

Physical leakage channel means that physical media (e.g., HDD, laptops, workstations, CD/DVD, USB devices) containing sensitive information or the document itself was moved outside the organization. This more often means that the control over data was lost even before it leaved the organization.

III. Detecting Challenges

A. Encryption: and preventing data leaks in transit are hampered due to encryption and the high volume of electronic communications. While encryption provides means to ensure the confidentiality, authenticity and integrity of the data, it also makes it difficult to identify the data leaks occurring over encrypted channels. Encrypted emails and file transfer protocols such as SFTP imply that complementary DLP mechanisms should be employed for greater coverage of leak channels. Employing data leak prevention at the endpoint – outside the encrypted channel – has the potential to detect the leaks before the communication is encrypted.

B. **Access Control:** Access control provides the first line of defense in DLP. However, it does not have the proper level of granularity and may be outdated. While access control is suitable for data at rest, it is difficult to implement for data in transit and in use. In other words, once the data is retrieved from the repository, it is difficult to enforce access control. Furthermore, access control systems are not always configured with the least privilege principle in mind. For example, if an access control system grants full access to all code repositories for all programmers, it will not effectively detect data leaks where a programmer accesses a project that he/she is not involved in.

C. **Semantic Gap in DLP:** DLP is a multifaceted problem. The definition of a data leak is likely to vary between organizations depending on the sensitive data to be protected, the degree of interaction between the users and the available communication channels. The current state-of-the-art, which is reviewed in Section III, mainly focuses on the use of misuse detection (signatures) and post-mortem analysis (forensics). The common shortcoming of such approaches is that they lack the semantics of the events being monitored. When a data leak is defined by the communicating parties as well as the data exchanged during the communication, a simple pattern matching or access control scheme cannot infer the nature of the communication. Therefore, data leak prevention mechanisms need to keep track of who, what and where to be able to defend against complex data leak scenarios.

The classification by leakage channel is important in order to know how the incidents may be prevented in the future and can be classified as physical or logical.

Physical leakage channel means that physical media (e.g., HDD, laptops, workstations, CD/DVD, USB devices) containing sensitive information or the document itself was moved outside the organization. This more often means that the control over data was lost even before it leaved the organization.

IV. 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 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.

V. 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 propose to develop *unobtrusive* techniques for detecting leakage of a set of objects or records.

In this section, we propose to 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. 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 guilty.

VI. ALGORITHMS

A. 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 in our chances of detecting a guilty agent. In the second place, we wanted to evaluate our e-optimal algorithm relative to a random allocation.

B. 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.

VII. MODULES

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.

E. Agent Guilt Module:

To compute this PrfGijSg, we need an estimate for the probability that values in S can be "guessed" by the target. For instance, say some of the objects in T are emails of individuals. We can conduct an experiment and ask a person with 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 we 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. To simplify the formulas that we present in the rest of the paper, we assume that all T objects have the same pt, which we call p. Our equations can be easily generalized to diverse pt's though they become cumbersome to display. Next, we make two assumptions regarding the relationship among the various leakage events. The first assumption simply states that an agent's decision to leak an object is not related to other objects.

VIII. CONCLUSION

From this study, I conclude that 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, data states, users, and IT platforms.

I 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 Ire leaked, then the distributor can be more confident that agent was guilty.

REFERENCES

[1] Technical Report TR-BGU-2409-2010 24 Sept. 2010 1 A Survey of Data Leakage Detection and Prevention Solutions P.P (1 -5, 24-25) A. Shabtai, a. Gershman, M. Kopeetsky, y. Elovici Deutsche Telekom Laboratories at Ben-Gurion University, Israel.

[2] IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 22, NO. 3, MARCH 2011 Data Leakage Detection Panagiotis Papadimitriou, Member, IEEE, Hector Garcia-Molina, Member, IEEE P.P (2,4-5)

[3] Data Leakage: What You Need to Know by Faith M. Heikkila, Pivot Group Information Security Consultant . P.P (1-3)

[4] International Journal of Computer Applications in Engineering Sciences [VOL I, ISSUE II, JUNE 2011] [ISSN: 2231-4946] P.P (1, 4) Development of Data leakage Detection Using Data Allocation Strategies Rudragouda G Patil Dept of CSE, The Oxford College of Engg, Bangalore.

[5] Mr.V.Malsoru, Naresh Bollam/ International Journal of Engineering Research and Applications (IJERA) ISSN:2248 -9622 www.ijera.com Vol. 1, Issue 3, pp.1088-1091 1088 | P a g e REVIEW ON DATA LEAKAGE DETECTION.

[6] Mr.V.Malsoru, Naresh Bollam/ International Journal of Engineering Research and Applications (IJERA) ISSN: 2248 -9622 www.ijera.com Vol. 1, Issue 3, pp.1088-1091 1088 | P a g e REVIEW ON DATA LEAKAGE DETECTION.

[7] International Journal of Computer Applications in Engineering Sciences[VOL I, ISSUE II, JUNE 2011] [ISSN: 2231-4946] P.P (1, 4)Development of Data leakage Detection Using Data Allocation StrategiesRudragouda G Patil Dept of CSE, The Oxford College of Engg, Bangalore. patilrudrag@gmail.com

[8] A Model for Data Leakage Detection Panagiotis Papadimitriou 1, Hector Garcia-Molina 2 Stanford University 353 Serra Street, Stanford, CA 94305, USA P.P (1, 4-5) 1papadimitriou@stanford.edu

[9] Web-based Data Leakage Prevention Sachiko Yoshihama1, Takuya Mishina1, and Tsutomu Matsumoto2 1 IBM Research - Tokyo, Yamato, Kanagawa, Japan fsachikoy, tmishinag@jp.ibm.com, P.P (2,14) 2 Graduate School of Environment and Information Sciences, Yokohama National University, Yokohama, Kanagawa, Japan tsutomu@ynu.ac.jp

[10] Data Leakage: Affordable Data Leakage Risk Management by Joseph A. Rivela Senior Security Consultant P.P (4-6)

[11] Data Leakage Prevention: A news letter for IT Professionals Issue 5 P.P (1-3)

[12] Data Leakage Detection Panagiotis Papadimitriou, Student Member, IEEE, and Hector Garcia -Molina, Member, IEEE P.P (2-6)IEEE transactions on knowledge and data engineering, vol. 23, no. 1,

JANUARY 2011

[13] The Who, What, When & Why of Data Leakage Prevention/Protection Presented by: Archie Alimagno California Department of Insurance P.P (2-7)

[14] An ISACA White Paper Data Leak Prevention P.P (3-7)

[15] Mr.V.Malsoru, Naresh Bollam/ International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com Vol. 1, Issue 3, pp.1088-1091 1088 | Page REVIEW ON DATA LEAKAGE DETECTION.