Research Article Impact Factor: 0.621 ISSN: 2319-507X
PB Narwade, IJPRET, 2014; Volume 2 (8): 542-554 IJPRET

INTERNATIONAL JOURNAL OF PURE AND APPLIED RESEARCH IN ENGINEERING AND TECHNOLOGY

A PATH FOR HORIZING YOUR INNOVATIVE WORK

DATA LEAKAGE DETECTION USING ASP.NET AND PREVENTION

PROF. P. B. NARWADE[1], PROF. P. J. WANKHADE[2]

1. I/C HOD, Computer Engineering Department, Dr. N. P. Hirani Institute of Polytechnic, Pusad, Maharashtra, India,
2. Lecturer, Computer Engineering Department, Dr. N. P. Hirani Institute of Polytechnic, Pusad, Maharashtra, India,

Accepted Date: 15/02/2014 ; Published Date: 01/04/2014

Abstract: Information is a knowledge of an information to distributor has given sensitive data to a collection of purportedly trusty agents (third parties). a number of the info square measure leaked and located in AN unauthorized place (e.g., on the online or somebody's laptop). The distributor should assess the chance that the leaked knowledge came from one or a lot of agents, as hostile having been severally gathered by different. The security policy contains information on what information is considered sensitive, how information and data transfers should be classified, and how incidents should be handled. Classification should be based on several factors including legal- and regulatory requirements, sensitivity and criticality, impact, and risks and threats. A data leakage incident response can vary in response time and appropriate action depending on the incident that occured. we tend to propose knowledge allocation methods (across the agents) that improve the chance of characteristic leakages. These strategies don't think about alterations of the discharged knowledge (e.g., watermarks). typically water marks will be destroyed if the info recipient is malicious. This paper focuses on sleuthing the distributor's sensitive knowledge that has been leaked by agents, and it's doable to spot the agents that World Health Organization leaks the info. In some cases, we are able to additionally inject "realistic however fake" knowledge records to additional improve our possibilities of sleuthing leak and characteristic the wrongdoer.

Keywords: Data Leakage & Architecture, Agent System, Data Leakage Detection, Using ASP.NET, Prevention, Implementation, DLP Limitation.

PAPER-QR CODE

Corresponding Author: PROF. P. B. NARWADE

Access Online On:

www.ijpret.com

How to Cite This Article:

PB Narwade, IJPRET, 2014; Volume 2 (8): 542-554

INTRODUCTION

In the course of doing business, sometimes sensitive data must be handed over to supposedly trusted third parties. 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 has been leaked by agents, and if possible to identify the agent that leaked the data. 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. Using an analogy with cookies stolen from a cookie jar, if we catch Freddie with a single cookie, he can argue that a friend gave him the cookie. But if we catch Freddie with 5 cookies, it will be much harder for him to argue that his hands were not in the cookie jar. If the distributor sees 'enough evidence´ that an agent leaked data, he may stop doing business with him, or may initiate legal proceedings. In this paper 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 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

authorized parties can. Encrypting data enables unauthorized hands the data is unreadable. The type and length of the keys utilized depend upon the encryption algorithm and the amount of security needed. In conventional symmetric encryption a single key is used

1. Data Leakage:

Data leak, put simply, is that the unauthorized transmission of information (or information) from among a company to AN external destination or recipient. this could be electronic, or is also via a physical methodology. knowledge leak is synonymous with the term info leak.[2] The reader is inspired to be conscious that unauthorized doesn't mechanically mean intentional or malicious. Unintentional or unintended knowledge leak is additionally unauthorized.[1]

1.1 Type of data leakage:

In order to implement the acceptable protecting measures, we tend to should initial perceive what we tend to square measure protective. Supported in public disclosed knowledge leak breaches, the kind of information leaked is attenuated as follows:[2][3]

Table I Type of information leaked:

Type of information leaked	Percentage
Confidential Information	15%
Intellectual property	04%
Customer data	73%
Health records	08%

1.2 Internal threats:

According to knowledge compiled fifty two of information Security breaches square measure from internal sources compared to the remaining forty eighth by external hackers.[5] The noteworthy side of those figures is that, once the inner breaches square measure examined, the proportion attributable to malicious intent is remarkably low, at but a hundred and twenty fifth.[3]The corollary of this is often that the extent of unintended knowledge breach is important (96%). this is often additional deconstructed to forty sixth being attributable to worker oversight, and five hundredth attributable to poor business method.[2]

2. Data Leakage architecture:

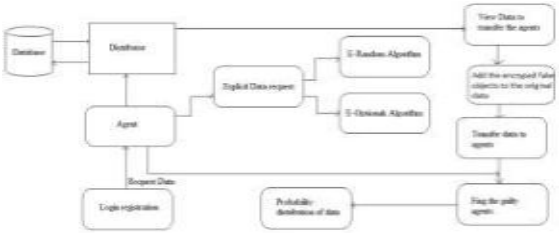

Figure.1: Data Leakage architecture

We study unobtrusive techniques for detecting leakage of a set of objects or records. Specifically, we study the following scenario [10]: After giving a set of objects to agents, the distributor discovers some of those same objects in an illegitimate place. (For example, the data may be found on a web site, 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 [9]

We develop a model as shown in Figure.1 for assessing the "guilt" of agents by considering the option of adding "encrypted fake" objects to the distributed set. Fake objects are encrypted using RSA algorithm. We also present algorithms for distributing objects to agents, in a way that improves our chances of identifying a leaker

3. AGENT SYSTEM:

Separation of powers and responsibilities in an agent community encourages flexibility and encapsulation. As such, our proposed agent system will be heterogeneous with members belonging to one of six principle archetypes, each unique roles and possessing distinct abilities. Figure 1 depicts the classifications of our Information Leakage Detection Agent system and the respective agent ranks. All inter-agent communications will adhere to FIPA Agent[8].

Communication Language (ACL) specifications in order to maintain communication interoperability between different agent platforms. Properties and responsibilities of each type of agent are discussed in following subsections.

3.1 Controller Agents (CA):

Controller Agents are responsible for dispatching subordinate agents and coordinating their respective activities in a designated network. Additionally, Controller Agents will coordinate the remote installation of the necessary mobile agent environment and other required software packages on target hosts with Environment Agents. Multiple instances of controller agents can be dispatched to ensure proper coverage of large networks as well as to accomplish load distribution for the purposes of performance optimization.[8]

3.2 Detection Agents (DA):

The main functionality of Detection Agents is to identify new hosts in the network and to verify the host's states.

In our initial design, a host's state will refer to the presence or absence of SELinux and the Colored Linux infrastructure. Once determined, a host's state will be reported to the[7] Controller Agent to aid in the identification of subsequent actions.

3.3 Queue Agents (QA):

To avoid overwhelming Controller Agents and to provide an orderly approach to dispatching agents to newly discovered hosts, Queue Agents will be useful. As stated above, when a Detection Agent identifies a new remote host, the host's state is reported to a Controller Agent. Rather than dispatching agents to a new host immediately, it may be preferred to defer such processing for some time, especially in the case when many such hosts are reported at once. In such cases, hosts are reported by Controller Agents to Queue Agents which prioritize hosts for subsequent processing by, and at the request of, Controller Agents.[8]

3.4 Monitor Agents (MA):

Monitor Agents will perform active monitoring on host file system through the notify kernel subsystem to identify file write and creation operations. Detail on the notify kernel subsystem will be discussed in the next section. When a write operation or file creation operation takes place, Monitor Agents notify Watermarking Agents which can then perform watermark analysis of the file in question. As comparable capabilities are already present in Colored Linux hosts, Monitor Agents will only reside in non-Colored SELinux.[7]

3.5 Environment Agents (EA):

Minimally, Watermarking and Monitor Agents require the necessary agent environment installed on a target host in order to reside and function there. Also, depending on the type of watermarking employed, certain water marking specific software dependencies which may not reasonably be accommodated by the Watermarking Agents themselves can exist. Environment Agents will be responsible for handling all such software dependencies without the intervention of the target host's administrator.[8]

In choosing an appropriate foundation for our agent community, we considered primarily the associated memory

TABLE II ENVIRONMENT AGENT COMMUNICATIONS (EA):

From:- Environment Agents (EA):
To: Controller Agent (CA) . Confirm with CA to perform environment checking and dependencies solution. (AGREE) . Notify CA of all resolved dependencies. (INFORM)

4. DATA LEAKAGE DETECTION:

There are a few popular content-based approaches that are used for detecting data leakage in outgoing information [4], [5]:

Global filters - concerning the whole file: File-based binary signature – hash value of the whole file. Can detect only an exact copy of a confidential file. Text-based binary signature – hash value of textual content of file. Offers more robustness compared to previous method. Can detect converted files e.g., txt to doc. Ignores text metadata like font, color, etc.Tokens - concerning special keywords or patterns: Keywords filter – used to build a policy based on keywords. For example, block files that mention “Project X". Pattern recognition – can block documents containing a match to a credit card number, phone number, etc.

Machine learning – can detect previously unseen confidential documents. Machine learning methods classify the documents according to their similarity to confidential or non-confidential documents[9].

Textual fingerprint – can detect full, near, and partial duplicates of confidential documents.

Global filters suffer from very low robustness e.g., even if a single character is replaced in a file, it cannot be detected. Token-based filters require a very accurate selection of keywords and patterns, which is not a realistic option in most cases. Machine learning (ML) methods require a lot of documents for training and suffer from high false positive rates. Additionally, ML methods cannot detect a partial duplicate of a confidential document since most of the document content is non-confidential and ML is based on statistics. Therefore, for content-based data leakage detection, the most natural choice is textual fingerprint. It is an efficient and effective approach and is commonly used by leading commercial DLD products; although, it also has some limitations as explained later in the section. [9]

A fingerprint of a document is a set of hash values of its features. In order to check whether document *d* is similar to one of the documents in a reference set *R* (when fingerprint is applied to data leakage detection, the set *R* consists of the confidential documents of the organization) indexing and detection phases are required. A pre-processing (indexing) phase is applied where each of the documents is *R[3]*

Fingerprinted. The fingerprints are stored in a special database (Figure 2). Then, during the detection phase, a fingerprint of the examined document *d* is extracted and is compared with fingerprints in the database. A list of documents that contain each of the hashes that are included in the fingerprint of document *d* is efficiently retrieved (using inverted index) from the database.[3] The documents that share a number (above some predefined threshold) of hashes with *d* are considered as similar. Therefore, there is no need to make a pairwise comparison of each document in *R* with *d* and the process time is linear to the length of *d*. This property makes fingerprinting highly scalable. As a case in point, Google's crawler employs their fingerprinting implementation to detect near duplicate web pages, while the reference set is the part of the Internet that Google is indexing [31]. Thus, it is naturally appropriate for real time environment.[4]

5. DATA LEAKAGE DETECTION USING ASP.NET:

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). [6]

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. We 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 we can also inject "realistic but fake" data records to further improve our chances of detecting leakage and identifying the guilty party.[6]

6. DATA LEAKAGE PREVENTION:

Over the last decade, enterprises have become increasingly reliant on digital information to meet business objectives. On any given business day, significant amounts of information fuel business processes that involve parties both inside and outside of enterprise network boundaries.[10]

There are many paths for these data to travel and they can travel in many forms—e-mail messages, word processing documents, spreadsheets, database flat files and instant messaging are a few examples. Much of this information is innocuous, but in many cases a significant subset is categorized as “sensitive” or “proprietary,” indicating that this information needs to be protected from unauthorized access or exposure. This need can be externally driven by privacy and other types of regulation, or internally driven by business objectives to protect financial, strategic or other types of competitive information. Most enterprises employ safeguards to control sensitive information. Often, however, these controls are inconsistent and are managed at different points in the enterprise with different levels of diligence and effectiveness. The result is that despite their efforts, enterprises around the globe leak significant amounts of sensitive information. These leaks create significant risk to enterprises, their customers and business partners with the potential to negatively impact an enterprise’s reputation, compliance, competitive advantage, finances, customer trust and business partnerships.[9]

6.1 Implementation:

Enterprises should strongly consider implementing DLP first in a monitoring-only mode. This will allow the system to be tuned and predict the impacts to business processes and the organizational culture. Allowing system-driven alerts to build awareness and to initiate behavioral changes is generally a better approach than to block traffic flows and potentially derail business processes. While leadership may have significant concerns regarding the amount of sensitive data “flying out the door” once the system is activated, initiating actual blocking too soon can cause even greater problems by breaking or severely impeding critical business processes. The hope is that these processes were identified during the preparation stage, but often things are overlooked that quickly come to light when the DLP solution is enabled.[7]

6.2 Remediation of Violations:

DLP (Data Leakage Prevention) solutions generally provide a great deal of useful information regarding the location and transmission paths of sensitive information. Sometimes, however, this can be a Pandora’s Box experience. An enterprise can be quickly dismayed at the volume and extent of its sensitive data footprint and loss, and may be inclined to rush forward to try to address all issues at once, which is a recipe for disaster. It is important that an enterprise be prepared to use a risk-based approach to prioritize and address findings in the most expedient

manner possible. All key stakeholders must be involved in this process since it frequently involves allowing one problem to continue temporarily while a larger one is addressed.[8][9]

The analysis and subsequent decisions regarding this process should be well documented and maintained in anticipation of future audits or regulatory inquiries.

6.3 Ongoing DLP Program:

The DLP solution should be closely monitored and periodic risk, compliance and privacy reports should be provided for appropriate stakeholders (e.g., risk management, compliance management, privacy team and human resources [HR]).[8]

DLP rules should continue to be reviewed and optimized. DLP solutions will not inform administrators that a rule is too broad and could have a significant performance impact on the DLP infrastructure. Enterprises should ensure that all stake [10] holders are diligent in reporting any new data formats or data types that may not be represented in the existing DLP rule set. A testing and staging environment should be available and used to test the impact of patches and upgrades on the DLP solution. Finally, it is important to continue training and awareness programs, which should be reinforced by the report and alert capabilities of the DLP solution.

7. DLP Limitations:

While DLP solutions can go far in helping an enterprise gain greater insight over and control of sensitive data, stakeholders need to be apprised of limitations and gaps in DLP solutions. Understanding these limitations is the first step in the development of strategies and policies to help compensate for the limitations of the technology. Some of the most significant limitations common among DLP solutions are:[7]

7.1 Encryption:

DLP solutions can only inspect encrypted information that they can first decrypt. To do this, DLP agents, network appliances and crawlers must have access to, and be able to utilize, the appropriate decryption keys. If users have the ability to use personal encryption packages where keys are not managed by the enterprise and provided to the DLP solution, the files cannot be analyzed. To mitigate this risk, policies should forbid the installation and use of encryption solutions that are not centrally managed, and users should be educated that anything that cannot be decrypted for inspection (meaning that the DLP solution has the encryption key) will ultimately be blocked.[7][8]

7.2 Graphics:

DLP solutions cannot intelligently interpret graphics files. Short of blocking or manually inspecting all such information, a significant gap will exist in an enterprise's control of its information. Sensitive information scanned into a graphics file, or intellectual property (IP) that exists in a graphics format, such as design documents, would fall into this category. Enteprises that have significant IP in a graphics format should develop strong policies that govern the use and dissemination of this information. While DLP solutions cannot intelligently read the contents of a graphics file, they can identify specific file types, their source and destination. This capability, combined with well-define traffic analysis, can flag uncharacteristic movement of this type of information and provide some level of control.[8]

7.3 Third-party service providers:

When an enterprise sends its sensitive information to a trusted third party, it is inherently trusting that the service provider mirrors the same level of control over information leaks since the enterprise's DLP solutions rarely extend to the service provider's network. A robust third-party management program that incorporates effective contract language and a supporting audit program can help mitigate this risk.[9]

7.4 Mobile devices:

With the advent of mobile computing devices, such as smart phones, invariably there are communication channels that are not easily monitored or controlled. Short message service (SMS) is the communication protocol that allows text messaging and is a key example. Another consideration is the ability of many of these devices to utilize [8] Wi-Fi or even to become a Wi-Fi hotspot themselves. Both cases allow for out-of-band communication that cannot be monitored by most enterprises. Finally, the ability of many of these devices to capture and store digital photographs and audio information presents yet another potential gap. While some progress is being made in this area, the significant limitations of processing power and centralized management remain a challenge. Again, this situation is best addressed by the development of strong policies and supporting user education to compel appropriate use of these devices.[9]

7.5 Multilingual support:

A few DLP solutions support multiple languages, but virtually all management consoles support only English. It is also true that for each additional language and character set the system must support, processing requirements and time windows for analysis increase. Until such time that

vendors recognize sufficient market demand to address this gap, there is little recourse but to seek other methods to control information leaks in languages other than English. Multinational enterprises must carefully consider this potential gap when evaluating and deploying a DLP solution.[6]

These points are not intended to discourage the adoption of DLP technology. The only recourse for most enterprises is the adoption of behavioral policies and physical security controls that complement the suite of technology controls that is available today such as:[6]

7.5.1 Solution lock-in:

At this time there is no portability of rule sets across various DLP platforms, which means that changing from one vendor to another or integration with an acquired organization's solution can require significant work to replicate a complex rule set in a different product.[6]

7.5.2 Limited client OS support:

Many DLP solutions do not provide end-point DLP agents for operating systems such as Linux and Mac because their use as clients in the enterprise is much less common. This does, however, leave a potentially significant gap for enterprises that have a number of these clients. This risk can only be addressed by behavior-oriented policies or requires the use of customized solutions that are typically not integrated with the enterprise DLP platform.[7]

7.5.3 Cross-application support:

DLP functions can also be limited by application types. A DLP agent that can monitor the data manipulations of one application may not be able to do so for another application on the same system. Enterprises must ensure that all applications that can manipulate sensitive data are identified and must verify that [8] the DLP solution supports them. In cases where non supported applications exist, other actions may be required through policy, or if feasible, through removal of the application in question.[7]

7.5.4 Business process:

Review business processes with access to confidential information and determine whether that access is required to perform each process. Identifying the need for access to confidential information from business processes is one of the strongest methods of protecting such data. In addition, appropriate processes for monitoring, detecting, qualifying, handling and closing data leakage incidents should exist.[7]

CONCLUSION:

In a excellent world, there would be no ought to pass sensitive information to agents that will unwittingly or maliciously leak it. And albeit we tend to had at hand over sensitive information, in an exceedingly excellent world, we tend to might watermark every object so we tend to might trace its origins with absolute certainty. [9] However, in several cases, we tend to should so work with agents that will not be 100% trustworthy , and that we might not be sure if a leaked object came from associate degree agent or from another supply, since bound information cannot admit watermarks.[1][3]

An enterprise's information can be among its most valuable assets. [7] DLP solutions offer a multifaceted capability to significantly increase an enterprise's ability to manage risks to its key information assets. However, these solutions can be complex and prone to disrupt other processes and organizational culture if improperly or hurriedly implemented. Careful planning and preparation, communication and awareness training are paramount in deploying a successful DLP program.[8]

REFERENCES:

1. R. Agrawal and J. Kiernan. Watermarking relational databases. In VLDB "02: Proceedings of the 28th International conference on Very Large Data Bases, pages55–166. VLDB Endowment, 2002.

2. P. Bonatti, S. D. C. di Vimercati, and P. Samarati. An algebra for composing access control policies. ACM Trans. Inf. Syst. Secur., 5(1):1–35, 2002.

3. Author undisclosed. (2007). Information Leak Statistics. Retrieved May 30, 2007, from Web sense.

4. Author undisclosed. (October 2006). Stop the Insider Threat.CSO Focus Vol.2 No.1

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

6. McAfee Data Loss Prevention. http://www.mcafee.com/japan/products/dlp.asp.

7. RSADLP. http://www.rsa.com/node.aspx?id=3426.

8. M. Alawneh and I.M. Abbadi, "Preventing information leakage between collaborating organisations", In Proceedings of the 10th international Conference on Electronic Commerce, vol. 342, pp. 1-10, 2008.

9. M. Alawneh and I.M. Abbadi, "Preventing Insider Information Leakage for Enterprises", The Second International Conference on Emerging Security Information, Systems and Technologies, pp. 99-106, 2008.

10. S. Cabuk, "Network Covert Channels: Design, Analysis, Detection, and Elimination", PhD Thesis, Purdue University, 2006.