Data Leakage Prevention Solutions Based on state of the Data

Subhashini Peneti
Research Scholar, Dept. of CSE
Jawaharlal Nehru Technological University organization
Hyderabad, India

Pademaja Rani B
Professor, Dept. of CSE
Jawaharlal Nehru Technological University organization
Hyderabad, India

***Abstract*—Now days, identification of data leakage threat is a key task for every organization. In order to identify the threat, organization's need to identify the data which is leaked by the threat. This paper provides different Data Leakage Prevention Solutions, helps to identify the data leakages and prevent the confidential data from the leakages.**

Keywords-component; formatting; style; styling; insert (key words)

I. INTRODUCTION

According to the Forrester Wave report [1], most early Data Leakage Prevention (DLP) solutions focused on finding sensitive data as they left the organizational network by monitoring data-in –motion at the various network egress points. In the second stage ,as removable storage devices(e.g. USB sticks, external hard drives) proliferated, DLP solutions began to focus on detecting data leakage at the endpoint and on providing capabilities, for example, to subvert copying of sensitive information to USB devices or CD/DVDs even if the endpoint is not connected to the network.

The biggest DLP challenge lies in protecting the large amounts of sensitive data which exist in unstructured form .Therefore; DLP solutions providers are continuously improving their data improving their data discovery methods using approaches such as fingerprinting and natural- language processing [2].

According to three leading research reports, in the next few years.DLP products are expected to become as stable and as commonplace as existing security solutions such as firewalls, intrusion detection systems and anti-malware solutions [1, 2, and 3].

The DLP functionalities will be characterized based on the state of the data which they aim to protect (i.e., data-in-use, data-at-rest, and data-in-motion.

A. Protection for data-at-rest

Protection for data-at –rest is also provided by encryption of data at the endpoint. This can do using full-disk encryption or file-level encryption with access control. Encryption will protect sensitive data if, a laptop is stolen or lost.

B. Protection for data-in-use

Protection for data-in-use is provided by a local, host-based agent that locally monitors and prevents actions involving sensitive data such as copy-and-paste, print-screen ,copying to a USB/CD/DVD, unauthorized data transmission, or use of data in unapproved applications.

C. Protection for data-in motion

Protection for data-in motion is provided by means of a network-based solution that searches for and blocks content that violated a policy. Network monitoring components are often deployed at or near the enterprise gateway. They perform full packet capture, session reconstruction, and content analysis in real time.

II. DIFFERENT TYPES OF DATA LEAKAGE PREVENTION SOLUTIONS

The DLP solutions can be clustered into the following categories based on data state:

1. Email leakage protection
2. Network/Web-based protection
3. Detecting malicious insiders using honey pots and honey tokens.

A. Data Leakage Prevention solutions for protecting Data-at-Rest(DIR)

1. Misuse detection in information retrieval system.
2. Encryption and access control.
3. Hiddent data in files

1) Misuse detection in information retrieval systems.

The typical approach proposed for detecting data leakage in information retrieval systems is anomaly detection. Generally,

the system learns the normal behavioral profile of a user and detects deviations or anomalies with references to this profile. The main problem in an anomaly detection approach is how to model user's behavior. In [4] the author Cathey proposed four methods for modeling behavior in information systems: document clustering, clustering query results, relevance feedback and fusion method.

Document clustering operates in three phases. First, all documents in the Information retrieval system are clustered base on their content. Next during the training phase the misuse detection system builds a user profile based on the clusters of documents retrieved by a user's queries. During the detection phase, when the user attempts to access a document in that same cluster which he or she does not normally access and when the cluster is not similar to any one of the normal accessed clusters, an alarm is raised.

Clustering query results is similar to document clustering, only in this method, the documents being clustered are only those previously accessed by the user. The documents that user has accessed in the past are usually only a small portion of the full document, meaning that fewer documents are clustered.

In the relevance feedback approach, the submitted query is analyzed, and key words are identified, in addition the document currently being accessed by a user also analyzed. A user profile created as a combined set of keywords that a user has queried or accessed. When the user submits a new query, the query's keywords are extracted, and if the number of keywords that do not appear in the synthesized user profile exceeds some threshold, an alarm is raised.

Finally, the fusion method simply combines all previous methods in weighted average to determine when to raise the alarm.

Experimental evaluations performed by Cathey indicated that the relevance feedback and fusion methods provided the best overall results. In [5] the authors Ma and Gohrian tested the relevance feedback method for a user who submits a short(max four terms) or a long query(up to 17 terms).The results of this evaluation had an overall precision of 83.9% and 82.2% for short and long queries respectively.

2) Encryption and access control

Encryption and access control are two of the most common means for preventing leakage of confidential data through access restriction. Such frameworks use access control and encryption to secure sensitive data at-rest (e.g. stored on laptops, server, PCS, etc.), in-motion (e.g. transferred through the local network or on the web), and in-use (being accessed or modified).

Access control mechanisms in place can reduce the risk of data leakage; however, the amount of reduction is still limited because legitimate users such as employees and partners continue to access to sensitive data. Several related studies have addressed this issue.

One of the key questions in solutions that provide encryption of data or event of the whole disk 9for example True Crypt) is how encryption will affect data recovery in cases where the password has been forgotten or in the context of incident investigation and forensics [19].

Abbadi and Alawneh (2008) [20] presented a solution for preventing information leakage when the adversary is someone who is authorized to view the data. Generally, the proposed framework allows authorized users access to sensitive information from inside or outside an organization's premises (access from outside the organization is over VPN). The key concept is allowing access to sensitive data from unauthorized disclosure. This is achieved by creating a domain of devices which are authorized to access the data. Each domain has its own specific master controller that manages security administrator authentication, secure addition and removal of devices to and from the domain, and domain specific key distribution (denoted by KD). Only devices inside the organization's premises are authorized to join the domain: otherwise they cannot own a KD.The joining device has to trusted, i.e., to correspond to the expected state of the device and must be physically added by an authenticated security administrator. The only entity on a device that is authorized to manage encryption keys is a trusted software agent, who is assumed to use hardware that provides cryptographic functions.

While being transferred between domain devices, sensitive data are encrypted using the domain key KD. Because the KD key can be transferred only form trusted master controller to an authorized device, it is stored in a protected storage area and connote be copied between devices. This guarantees that if sensitive data reaches an unauthorized device, they cannot be disclosed.

While being stored in a device, the sensitive data encrypted using a device specific key, denoted as KC. KC is stored in a protected storage area. Before data transferred, they are decrypted using KC and re-encrypted using KD.

This framework prevents unprotected data from being transferred using the web or mass data storage (assumed to be prevented by the trusted software agent).It also prevents access to sensitive data on unauthorized devices. However, the proposed framework does not prevent an authorized user from rendering content on an authorized device with the physical presence of another unauthorized user (assuming that physical controls are not in place). It also does not prevent an authorized user from memorizing, writing or recording content and then transferring it to others.

Alwaneh and Abbadi(2008) further described a framework for protecting sensitive data shared between collaborating organizations. In such cases, one organization required sensitive data from another organization, but the data still needed to be protected from leakage to unauthorized users inside or outside the destination organization. The proposed solution is based on trusted computing, which provides a

hardware based root of trust. The shared data are protected while being sent to the collaboration organization by establishing VPN connections. Definitions of global domains in the destination organization ensure that the data can be shared between devices in the domain while still remaining protected from leakage outside the organization. The trusted computing platform ensures that the data are kept encrypted and that the encryption key is accessible only to devices in the domain and cannot be transferred to devices outside the organization. A software agent installed on the device will refuse to release sensitive content to other devices unprotected (even if they are a member of the global domain). Dynamic domains are used to specify subgroups of devices, which should be the only ones to share content using the domain-specific key.

Parno et al. (2009) [21] presented CLAMP (confidential Linux Apache MySQL PHP applications), which is a transformation performed on top of an existing LAMP based web application and which results in more data-leak-proof application. The transformation is based on taking the authentication process out of the application boundaries into a separate user authenticator (UA) module. In addition, each user who connects to the server will get a fresh and clean duplication of the server (called Web Stack) forked form a protected unchangeable copy. The new Web Stack runs in a separate virtual memory area, which provides total isolation between the servers seeing each user, and which in turn means that damage to one server instance, will not affect the rest of the servers. The Web Stack ID and the single unique UA which is attached to it are used by the query restrictor (QR). The QR is a database proxy that created a virtual database (using the database's "view" capability) which contains only the data that the user is allowed to see and which restricts SELECT, INSERT and UPDATE operations according to a predefined policy. The authors claimed that it is relatively easy to modify an existing Web application to work with CLAMP. On the other hand, the method consumes large amounts of memory and CPU resources on the server and cannot protect against insider attack.

Yasuhiro and Yoshik (2002) presented a Web-based framework aimed at preventing leakage of confidential information. This is done by encrypting confidential data and granting access only to authorized users, as well as by using a specialized viewer embedded in the Internet browser for decrypting and viewing content. The system operates in two phases: the download phase and the viewing phase. The download phase is based on a smart proxy that uses an authorization database to determine whether the current user can download the requested content and whether the content needs to be encrypted before sending it through. In the viewing phase, a smart viewed on the user's computer handles the request for the decryption key and decrypts the content. Following decryption, the smart viewer presents the content to the user (allowing the user to view the content once per key download), and the viewer is able to disable the save, print, and print-screen operations. The proposed framework is transparent to the user and protects confidential data while they are at-rest (encrypted in the database), in-motion (being sent encrypted over the network) and in-use (the user can watch the content in a specialized view that prohibits printing or saving). However, the print-screen option is not completely blocked and can be bypassed.

The concept of fine-grained access control for database systems was initially proposed to provide better data protection by controlling access at the granularity level of individual rows and columns. Fine grained access control at the database level (as opposed to the application program level) can be provided by modifying the original table being accessed by injecting a dynamically created temporary view between the query and the target table [Zhu, 2008].

De Capitani Di Vimercati et al [22] proposed the concept of selective encryption to provide selective access control to outsourced sensitive data by third-party partners. According to the proposed approach the data success authorization policy is processed to compute a hierarchical structure of tokens which are used to derive a set of cryptographic keys. This set of cryptographic keys, referred to as an encryption policy, enable selective encryption of and access to the data. The authors proved that the problem of computing a minimum encryption policy is NP-hard and presented a heuristic algorithm to solve the problem.

3) Hidden data in files

Documents written and stored in the Microsoft Word document format might contain hidden data.However; awareness of this problem is not sufficiently widespread, especially among non-technical computer users [23]. Examples of hidden data in word documents include the names and usernames of the document's creators and their collaborators and organizational information on the users involved.

Yixiang et al [24] claimed that publishing an XML document data with security requirements poses a multitude of challenges when users can infer data using common knowledge. Moreover, when two or more documents are involved, users can infer the sensitive data by combining the documents. The core of the Eliminate Inner Nodes algorithm, for use when publishing several related XML documents, is to find a maximal partial document which avoids information leakage while at the same time allows for publishing as much data as possible.

B. DLP Solutions for protecting Data-in-Use(DIU)

1) Honeypots for detecting malicious insiders

A honeypot is a mechanism which commonly used for detecting attacks from an outside source. It is an artificial resource set up as a trap which is aimed at detecting, deflecting, or in some sense counteracting attempts at unauthorized use of information systems.Generally,the trap consists of a computer database, web site or application server that appears to be part of a real production network, but is actually isolated, intentionally unprotected, and unobtrusively monitored. The honeypot should look genuine, be available and be vulnerable to draw the attacker who attempts to exploit it into the trap. Any interaction with the honeypot is by definition an anomalous situation that should be further reported and investigated. Forensic information provided by the honeypot is logged and analyzed to gain insight into various attack patterns(e.g., who the attacker is ; where, how and when was an attack launched; etc) The collected data enable inspection of attacks at various levels of abstraction, ranging from low-level network interface and routing protocols to higher application-level protocols[25].

Spitzner [26] noted the following main advantages of honeypots. First; honeypots collect data only when someone or something malicious interacts with them. This makes the data collected by the honeypots highly succinct, accurate, and easy to manage and simple to analyze. Second, honeypots can identify and capture new attacks. Because any activity with the honeypot is anomalous by definition, new or unseen attacks are detectable and result in a low false negative rate.

Honeypots usually focus on intercepting external attacks which attempts to compromise or penetrate a host or network. There are currently only a handful of academic articles on using honeypots to tackle insider threats. These studies discuss data (e.g. credit card number, a database entry, or bogus login credentials). Honeyfiles are files that contain fake information land that are planted in and organization's file system or in personal folder (e.g., PowerPoint presentation, an Excel spreadsheet, or an email message).

Internal attackers pose a much greater challenge to organizations because theory narrows the detection window available for existing countermeasures such as IDS, firewalls and IPS. Valli (2005) asserted that are stringent assumptions should be made when using honeypots against insider threats, for example: an insiders' legitimate access privileges: existence of high-speed network connections and access to the honeypot; deep acquaintance with the defense configuration and its weaknesses; and knowledge of earlier states of the application architecture, technologies and functionalities.

The concept of monitoring honeytokens has already been proposed by Story (2009). According to Storey, the first step is to learn how data item are legitimately used and, over around the organizational network. With this knowledge, honeytokens can be planted into genuine system resources. Using tools such as Snort, these honeytokens can then be monitored.

Spitzner proposed a two-stage approach for using honeypots. The first stage is planting honeytoken (i.e. user names, passwords) in an organization's applications i.e. Files, network traffic, etc. This information and then direct the insider to a ,ore sophisticated honeypot which can be further monitored and can be used to gather information on the perpetrator, to validate whether an insider has malicious or unauthorized intent, and to identify who the insider actually is and perhaps to determine his or her ,motives. For example, a honeytoken can be inserted into network traffic) e.g., a username and password which will be sent as part of an email text). If a sophisticated insider is passively monitoring network activity, he or she will encounter this honeytoken, which will point to a honeypot application into which the attacker will attempt to login using honeytoken he or she just obtained.

Bowen et al [27] presented the Decoy Document Distributor proof-concept system. The Decoy Document Distributor (DDD) system is a web-based service which first generates and sends decoy documents with embedded honeytokens to registered users and then, monitors any activity using the honeytokens. Multiple decoys are sent to increase the detection rate. An example of a honeytoken deployed by D^3 is a fake banking login account specifically created, published and monitored to attract and trap financially motivated attackers. The detection mechanisms used by the D3 system can be deployed at the network level, host level or both to detect the decoy documents. The authors of the decoy documents can be alerted whenever such a document is detected. For example D^3 will create s MS Word file containing login details for a Gmail account. The user downloads this file form the Web server to host laptop. When an attacker notices this file, he will try to login to the bait account. Custom script swill gather account activity information and an alert will be triggered. The honeyfiles created by the D3 system can be, monitored by the Decoy Documents Access sensor [] for masquerade attack (identify theft) detection.

C. DLP solutions for protecting Data –in-Motion(DIM)

*1) Email leakage protection :*Research in email leakage protection can be divided into two main categories: Content-based approaches and behavior-based approaches.

The content-based approaches can be further divided into:

- *Key-words based rules.* In these approach different rules are framed based on keywords appear in the body and the header of an email. These rules determine the confidentiality level of the scanned email based on the occurrences of the keywords [6] [7] [8].

- Machine learning techniques. The basic idea of this approach is to use machine learning techniques such as SVM [9] [10] and naïve Bayes [11] [12] [13] to determine the confidentiality level of the scanned email message.

Two methods are used to represent textual data in emails. The first method is the vector space model. Vectors represent documents, and vector features represent terms and their frequency of appearance [14].The vectors are used as learning sets to build a probabilistic model, on the bases of which decisions are made whether or not documents are confidential.

The second method for the representation of textual data is graphs. In general, words are represented as nodes in the graph and are connected by edges to words that appear in their vicinity. [15] Presented six major groups of algorithm for creating graphs from document text. The algorithms differ in their use of term-based techniques.

The behavior-based approach focuses on environment-related features such as organizational structure and which users send and receive email. For example in [16], the likelihood that an email has been sent by mistake is determined base on an analysis of past communications between email senders and recipients.

In [17] a sent email is identified as a leak based on the textual content of the mail and the likelihood that the recipient of the email should be receiving it. Messages sent to past recipients are modeled as (message, recipient) pairs, and a (message, recipient) pair is considered to be a potential leak if the message is sufficiently different form past messages sent to that recipient. To improve performance, Carvalho and Cohen (2007) used various social network features. Their proposed solution used two different techniques for detection. The first technique relies strictly on the message's textual content. It measures the similarity between two vector-based representations of email messages. The first vector is a TF-IDF representation of all previous messages from the current user u to recipient r (a different vector is created for each recipient).The second vector is a TF-IDF representation o the current message which is about to be sent. The distance between the two vectors is measured using one of two suggested algorithms: Cosine-similarity or K-nearest neighbors(KNN).If the computed similarity is less than a predefined threshold, a warning message is issued to the user who is about to send the message. The comparison is done separately for each recipient of the message which is about to be sent.

The second technique proposed was a classification-based method and was implemented using social network information (such as the number of received messages, the number of sent messages, and the number of times that a particular pair of recipients were copied in the same message). The idea was to perform leak prediction in two steps. In the first step, textual similarity scores were calculated using a cross-validation procedure on the training set. In the second step, network features were extracted, and then a function which combined theirs features with textual scores was calculated.

In the proposed method, email leaks were simulated on the basis of the Enron email corpus [] using different possible criteria. These criteria imitate realistic types of leaks, such as misspellings of email addresses, similar first/last names etc. The advantage of this approach is that it can be easily implemented in an email client and does not use by information which is available to the server only.

2) Network/web-based protection

Borders and Prakash [18] described a method for quantifying potential network-based information leaks. This approach uses the fact that a large portion of network traffic is repeated or constrained by protocol specifications. By ignoring these fixed data, the true information that flows from a client to the internet can be isolated. The authored focused on the Hypertext Transfer Protocol (HTTP) and computed the content of expected HTTP requests using only externally available information, including previous network requests, previous server responses and protocol specifications. The resulted in a measurement of the amount of unrepeated and unconstrained outbound bandwidth that represents the maximum amount of information that have been leaked by the client.

These leak quantification techniques were evaluated on web traffic form several legitimate web-browsing scenarios. The authors stated that this approach cannot handle malicious web requests from pages with active JavaScript code or Flash objects.

In 2009 Caputo et al. presented the Elicit system that monitors user's access to information on an intranet. The system uses network-based sensors that process network traffic to produce information-use events such as searching, browsing, reading, deleting and printing. The collected events are combined with contextual information and processed by various rule-based and statistical detectors that may issue alerts. Finally alerts form detectors are fed into a Bayesian network which produces a probability that a user's activity is malicious.

Conclusion

From analysis of different DLP technologies, we concluded that all these solutions are prescribed mainly for mitigating accidental leakage incidents. All these solutions typically provide no defense against intra-organizational data leakage (between departments), integration with virtualization frameworks should enable organizations to provide internal DLP with low effort.

1. Conclusion

From the analysis of different DLP technologies, we concluded that all these solutions are prescribed mainly for mitigating accidental leakage incidents. All these solutions typically provide no defense against intra-organizational data leakage (between departments), integration with virtualization frameworks should enable organizations to provide internal DLP with low effort. All the DLP solutions are suffering from high false positive rate.

References

[1]. Rascheke T The Forrester Wave: Data Leak Prevention 2008,Technical Report

[2]. Frost ,Sullivan World data leakage prevention market, Technical report.

[3]. Ouellet E, Proctor Magic Quadrant for content –aware data loss prevention Technical report, 2009.

[4]. Cathey R, Ma L, Goharian N Misuse detection for information retrieval system. Proceedings, 12 th ACM conference on Information and Knowledge Management, 2003.

[5]. Ma L, Goharian N Query length impact on misuse detection in information retrieval systems. Proceedings, ACM symposium on Applied Computing, 1070-1075.

[6]. Cohen W Learning rules that classify e-mail. Proceedings, AAAI Symposium on Machine Learning in Information Access, 1996, 18-25.

[7]. Helfman J, Isbell Ishmail: Immediate identification of important information Technical report, 1995.

[8]. Rennie j ifile: an application of machine learning to e-mail filtering. Proceedings, KDD workshop on text mining, 2000.

[9]. Cohen w, Singer Y Context-sensitive learning methods for text categorization. ACM transactions on Information systems, 1999, 17(2), 11-17.

[10]. Drucker H, Wu D Support vector machines for spam categorization. IEEE transaction on neural network, 1999, 10(50, 1048-1054.

[11]. Androutsopoulos I, Koutsias J An experimental comparison of naïve Bayesian and Keyword-based anti –spam filtering with personal e-mail messages. Proceedings 23 rd annual international ACM SIGIR Conference on Research and Development in Information Retrieval,2000,160-167

[12]. Hovold J Naïve Bayes span filtering using word-position-based attributes, Proceedings 2nd conference on email and anti-spam, 2005.

[13]. Sahami M, Dumais S A Bayesian approach to filtering junk email.AAAI-98 workshop on learning for text categorization.

[14]. Salton G, McGill M Introduction to modern Information retrieval .McGraw-Hill, INC, NewYork, USA.

[15]. Schenker a Graph Theoretic Technique for web content mining, PhD thesis, University of South Florida, 2003.

[16]. Kalyan C, Chandrasekaran k Information leak detection in financial emails using mail pattern analysis under partial information. Proceedings 7 th conference on 7th WSEAS International conference on applied informatics and communications, 2007, 104-109.

[17]. Carvalho V, Cohen w Preventing information leaks inemails.Proceedings, SIAM international conference on data mining, 2007.

[18]. Borders K, Prakash A Towards quantification of network-based information leaks via HTTP, Proceedings 3 rd conference on hot topics in security, 2008.

[19]. Forte D, Do encrypted disks spell the end of forensics? Computer frauds and security, 2009, 18-20.

[20]. Abbadi I, Alawneh M.Preventing inside information leakage for enterprises. Proceeding international conference e on emerging security information systems and technologies.99-106.

[21]. Parno B, McCune J.CLAMP: Practical prevention of large scale data leaks.Proceedins, IEEE symposium on security and privacy, 2009.

[22]. De capitani, Forest s.Encryption policies for regulating access to outsourced data. ACM transaction on database system, 35(2), 12:2-12:46.

[23]. Bayes S Information leakage caused by hidden data in published documents.IEE security and privacy, 2920, 23-27.

[24]. Yixiang S, Tao P Secure multiple XML documents publishing without information leaks. Proceedings international conference o convergence information technology, 2114-2119.

[25]. Valli Honeypot technologies and their applicability as a strategic internal countermeasure. International journal of information and computer security, 1(4), 30-436.

[26]. Spitzner L Honeypots: Catching the insider threat's Proceedings 19th annual computer security applications conference ,170179

[27]. Bowen B, Hershkop.Baiting inside attackers using decoy documents. Proceedings 5 th international ICST conference, 2009.