International Journal of Advanced Research in

Computer Science Engineering and Information Technology

Volume: 3 Issue: 1 26-Jun-2014,ISSN_NO: 2321-3337

Data Leakage Prevention System by Context based Keyword Matching and Encrypted Data Detection

Soumya S R[1] , Smitha E S[2]
Mtech Scholar, Dept. Of Computer Science & Engg, LBSITW, India[1].
Asso.Professor, Dept.Of Computer Science & Engg, LBSITW, India[2].

.

ABSTRACT— *Data Leakage is an important concern for the business organizations in this increasingly networked world days. Unauthorized disclosure may have serious consequences for an organization in both long term and short term. Data leakage is enhanced by the fact that transmitted data (both inbound and outbound), including emails, instant messaging, website forms, and file transfers among others, are largely unregulated and un monitored on their way to their destinations. The objective of this paper is to enhance the security of Data Leakage Prevention(DLP) system by finding documents containing confidential information even when most of the document consists of non-confidential content by context based keyword matching method and finding encrypted information in word documents by using Entropy method. The combined approach will help to enhance the security of the DLP system efficiently by detecting sensitive through text document or word document that contain encrypted information.*

Keywords— Data Leakage Prevention, Context, cluster graph, Entropy method.

I. INTRODUCTION

Information leakage is defined by [1] as the ‘‘accidental or unintentional distribution of private or sensitive data to an unauthorized entity’’. The definition of ‘‘sensitive data’’ is wide and can include (among other things) financial data, intellectual property, and customer details. In addition, once information has leaked it is nearly impossible to stop it from spreading. Confidential information can be sent to individuals outside the organization very easily and cause serious damage. One of the greatest threats to information security is the leakage of data by the organization’s own employees through emails. One way is to send sensitive information as a part of non-confidential communication. That is, only small part of message contain the confidential information but remain are non- confidential. Another way of spreading information by the employees of the organization by encrypting the sensitive information through some trusted applications like MS word, MS Excel etc. All these type of leakage is not usually detected by the DLP systems. Because most of the data leakage prevention mechanisms using keyword based or statistical based methods. The keyword based methods ignore context and statistical based method ignore the content of the documents. Also the current DLP systems does not consider encrypted contents of the documents. So, by keeping this factors in mind, we proposing a security enhancement approach for DLP system by finding

confidentiality of documents based on context of keywords and detecting encrypted information in word or text documents.

The proposed approach contain two phases; Learning phase and Detection phase. The input for the system are text or word documents. In learning phase , system generate document clusters for input documents contain both confidential and non-confidential documents. From them, it detect the confidential key terms and its context. Then prepare a confidential term graph for each cluster. It contain confidential key terms and context terms. Each cluster have an assigned level of confidential threshold value based on the confidential term that have. In Detection phase, Document to be tested is assigned to the relevant cluster based on similarity of confidential terms with the cluster graph and find the confidential score of the document. Then the score of document is above the threshold value of the assigned cluster then that document is blocked from sending. If the document to be tested is a word document, then system going to check for the encrypted information. Because the encrypted information contained files are wrapped out as doc or pdf files. The encrypted data detection is performed by checking the randomness of the ciphers in the document. Here Entropy method is adopted for the randomness testing of block ciphers produced by the encryption algorithm. If the document contain any encrypted information it is blocked from sending.

The remainder of this paper is organized as follows. In Section II, we present previous work on content protection and data leakage prevention. In Section III, we present the proposed method. Section IV concludes the paper.

II. RELATED WORKS

A. Information Leakage Prevention Methods

Considering the importance of the threat of information leakage, there has been relatively little published research on the matter. Existing research can be divided into two main areas: content and behavior-based methods.

The content-based approach includes the rule and classifier-based approaches. In the rule-based approach, various rules are defined with regard to words and terms that may appear in a scanned text. When used to protect information, these rules determine the ‘‘confidentiality level’’ of the scanned text based on the number of appearances of certain words and/or phrases. This technique is discussed in various academic studies [3,4,5,6,7]. The well explored classifier-based approach consists of various classification and other machine learning techniques, such as support vector machines (SVM) [4,5] and naïve Bayes [6,7]. In this approach, documents are represented as vectors, while the terms of the documents and their frequencies are the features of the vectors. These vectors form the learning set for a probabilistic model that classifies documents as confidential or not. These techniques are often used in spam detection, a field related to ILD&P.

Recent studies attempt to not only identify confidential content but to determine the level of threat its leakage presents to the organization; works such as [8] proposes a

''score'' for determining how detrimental a leak of the analyzed content would be and a framework for applying it. Others focus on the identification of entities (people, places, companies etc.') [6] as a mean of improving their detection abilities. Another approach, which does not exactly fit in either category is fingerprinting, used mostly in commercial products. In fingerprinting, documents are represented as a set of strings generated by using a hash function on a sliding window that spans X characters/words of a document. Each tested document is analyzed in the same manner and its hash values are compared against those in the database. If a sufficient number of matches are found, the document is considered confidential. Fingerprinting is used in commercial products offered by companies engaged in computer security, such as Symantec and Web sense, but is studied in academic literature in relation to plagiarism detection, finding near-duplicate files, authorship detection and even website summarization rather than in relation to leakage prevention.

The behavior-based approach [7] to ILD&P focuses on identifying anomalies in behavior. These anomalies can be tracked in the communication, in and out of the organization as a whole, or the analysis of past and current email communication for a single person. Other studies propose the use of decision trees to access illegitimate access to customers' personal data and the identification of similar behavior when accessing databases.

The behavior-based methods those that analyze communication patterns, are useless for this type of data leakage scenario since the email was sent to its intended recipient and, as noted previously, the vast majority of email's content is not confidential.

B. The use of graph representations in the field of information security.

To the best of our knowledge, graphs have not been used in the field of ILD&P for the purposes of text representation. However, they have been used in the fields of access control (AC) network vulnerability analysis. They have also been utilized in conjuncture with online social networks for the purpose of spam detection and organization mining (inferring the structure of an organization based on the social network of its employees). In all these studies, graphs are used to represent states and entities. The reasons graphs are used in these areas are that they are easy to modify and expand and because they assist users in understanding (using visualization), the connections between states, rules, and entities.

C. Encrypted data Detection

Filiol [10] defined a statistical test based on comparing a cryptographic function's algebraic normal form to that of a random Boolean function, and applied the test to the DES and AES block ciphers as well as several stream ciphers and hash functions. Katos [11] defined a statistical test to measure the diffusion of the block cipher's mapping, but did not apply the test to any actual block ciphers. In paper [13] propose novel approaches to the problem of classifying high entropy file fragments. This paper proposes two methods

that do not rely on such patterns. The NIST statistical test suite is used to detect randomness in 4 KiB fragments and also use the compressibility of a fragment as a measure of its randomness.

III. PROPOSED METHOD

The proposed method has two phases:- Learning phase and Detection phase. This method has the advantage of finding encrypted data in a word documents. This is normally not done by the current DLP system .So it will improve the capability of DLP system for securing sensitive information within an organization.

A. Learning Phase

The learning phase begins by performing an unsupervised clustering using k-means algorithm with the cosine measure as the distance function for identifying the various subjects represented by all documents (both confidential and non confidential). The next step is aimed at representing the confidential content for all clusters containing confidential documents. This is accomplished by applying the following procedures:

(1) Detect the confidential key terms - The purpose of this step is to identify the terms, which indicate with a high probability that a document in a cluster is confidential. We refer to these terms as confidential key terms. Our first intuition was to choose terms with a high probability of appearing in confidential documents and a low probability of appearing in non-confidential documents. The former probability could be divided by the latter in order to generate a ‘‘confidentiality score’’ for each term. This type of computation is referred to as language modeling and is usually used in the field of information retrieval. Here hierarchical language modeling is used and a separate language model for the confidential and non-confidential documents of the cluster are created. Then we calculate confidential score of each term in the text document based on the formula presented in Eq. (1), where Pconfidential_ LM(term) and Pnon_confidential_ LM(term) denote the probability of randomly sampling the analyzed term from the confidential and non-confidential language models, respectively.

$$\forall term \in Confidential\_LM,\ score_{term} += 1/iteration(P\_Confidential\_LM(term)) \Big/ (P\_nonConfidential\_LM(term)) \quad (1)$$

The term with confidential score greater than one is taken as confidential terms.

(2) *For each of the detected confidential terms, analyze the context* - The incorporation of the context in the representation of the confidential information is important because it enables a better understanding of the confidentiality of each term. Intuitively, the probability of a term being a part of confidential information is higher if it appears in similar contexts in other confidential documents. If a confidential term appears in an unrecognized context (or in a context only found in non-confidential documents), it is much less likely to be related to confidential content. In the proposed model, the context of a term is defined using a parameter referred to as the context span, which is used to

determine what the scope of terms surrounding the confidential term to be used as context should be. We calculated the probability of each context term to appear near the confidential term both in the confidential and non-confidential documents. This probability is defined as the number of documents in which the context term appeared near the confidential term divided by the total number of documents in which the confidential term appeared. This probability is calculated separately for the confidential and non-confidential documents. The score of each context term is computed by subtracting the probability of its appearance in a non-confidential context from the probability of its appearance in a confidential one.

$$score_{Context} += (1/iteration)(P\_Con(Context\_term/Confidential\_term)) - (P\_nonConf(Context\_term/Confidential\_term)) \quad (2)$$

The score of the context term is a positive value then that context is selected as the context for the confidential term.

(3) *Create a graph representation of the confidential content* - At the end of the process confidential terms and the context in which they appear are defined for each cluster. This information can easily be represented as a graph consisting of two types of nodes: confidential and context nodes. Confidential nodes can only be connected through their context nodes. That is, if two confidential terms have at least one common context term then they are connected in the graph.

Learning Phase Algorithm :-

Input :-Confidential & Non- Confidential documents.

Output :- Set of Clusters with confidential Cluster graphs.

Steps:-

1. *Perform Unsupervised Clustering on both dataset.*
2. *Create separate Language model for Confidential and non-Confidential documents.*
3. *Calculate the confidentiality Score of each term using eq (1).*
4. *Select terms with confidential score >1, as Confidential Key terms.*
5. *For each Confidential key terms, find context terms from context span.*
6. *Repeat steps 3, 4 & 5 for all documents in all clusters.*

B. Detection Phase

In the detection phase, we attempt to deal with detection of confidential documents as well as encrypted data. Bothe the processes are explained below;

(1)Confidentiality Detection:- It consists of the following steps:

i. *Assign the inspected document to relevant clusters*- The purpose of this step is to determine which of the confidential terms graphs (meaning, the clusters from which they were generated) will be used to determine whether the inspected document is confidential. The inspected document is transformed into a vector (after stemming and stop-words removal) and its similarity to each of the cluster centroids is calculated. This is done using the cosine distance measure . All the clusters whose similarity is above a predefined threshold are selected.

ii. *For each of the assigned clusters, identify all the confidential context terms that appear both in the document and in the cluster's confidential terms graph* - The text of the document undergoes stemming and stop-words removal and is then matched against the confidential terms represented in the cluster's confidential terms graph. The inclusion of a confidential term depends both on its own score and on the number and score of its context terms. The higher the score of the confidential term itself, the less we rely on its context terms in determining whether to include it.

iii. *For each of the assigned clusters, calculate the document's confidentiality score, based on the detected term*- The confidentiality score of the document for each cluster is calculated by summing up the scores of all the confidential terms.

iv. *Determine whether the document is confidential*-

At this point we have a confidentiality score for the document, more than one if the document was assigned to several clusters with confidential terms graphs. If the document's confidentiality score for a cluster is higher than the threshold defined for that cluster then the document is considered confidential and is blocked.

(2) Encrypted Data Detection

Data encrypted through good encryption algorithms should look random when observed at the bit level. Theoretically the data should be completely random so that a person should not be able to find any patterns that disseminate information about the data that is encrypted. Entropy is used to characterize the randomness of data. A string of all the same characters will give an entropy value of 0. The entropy will be higher as the string becomes more random. Entropy is used to detect whether a data is encrypted from the randomness measurement. Good encryption will produce very random sets of data, which means that its entropy will be higher.

Shannon's classical formula allows us to quickly place the observed (probability distribution of) data into three useful categories: low, medium, and high entropy. All compressed, encrypted, and random data would exhibit high entropy; text and coded medium; and sparse data, such as large tables, logs, etc., are usually of low entropy. So we are taking the assumption that the encrypted data will exhibit higher entropy value.

In detection phase, we will find the entropy value of each word of MSWord document for encrypted data detection. If any of the word entropy value is in the range of[5,8], it is

concluded that the corresponding MSWord document contain encrypted information. The equation for calculating entropy is,

$$H(X) = -\sum_{i=1}^{s} (P(X_i)\log_2(X_i)) \qquad (3)$$

where X is the word to which entropy is to be calculated and P is the probability of the word in the information.

Detection Phase Algorithm :-

Input:- Text documents & MSWord documents for confidentiality and encrypted data detection.

Output :- Blocking of Documents based on confidentiality level.

Steps:-

1. *Set confi_score=0.*
2. *For text documents, assign the document to be tested with relevant clusters based on keyword matching.*
3. *Compare terms in test document with assigned clusters graph for confidential key terms and context.*
4. *If any of the matched terms and its context present, then document is blocked.*
5. *For MSWord documents, calculate entropy of each word in the documents.*
6. *Check for entropy (terms) >5 then, corresponding word contain Word document is blocked.*

C. System Architecture

The architecture for the system shown below;

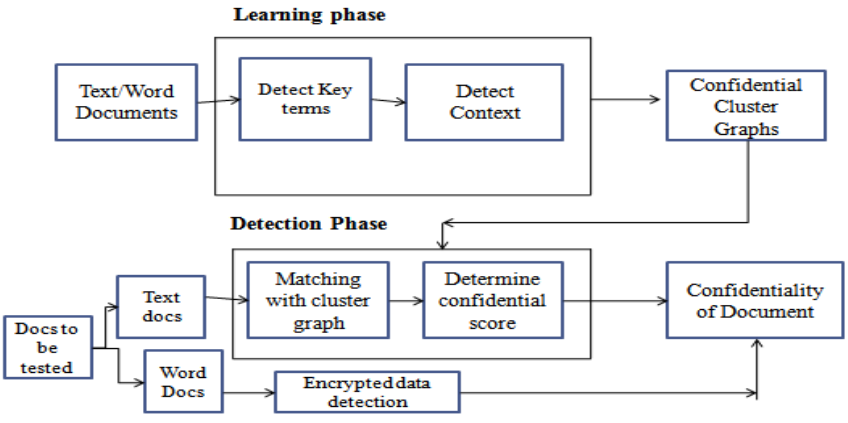

Figure 1. DLP System Architecture

IV. RESULTS AND DICUSSION

The system is implemented using Net Beans 7.0.1 IDE and Java was used to implement the system. XAMPP MySQL database is used for storing confidential key terms and their context for each of the cluster graph. The learning phase of the system is handled by administrator. Here administrator can perform the clustering on input documents. The input documents is a set of confidential and non-confidential documents. After clustering, cluster graph for each confidential key terms for each of the cluster is generated. In detection phase, confidentiality of documents to be checked. Here users given provision for sending text or word documents. After uploading their documents the system automatically check for confidentiality of document by comparing information stored on the database in learning phase. If any of the matching is found , document is blocked. In the case of word document, system check for encrypted data by computing entropy value of each of the terms.

The following are the interfaces showing when user attempt to send documents containing confidential information.

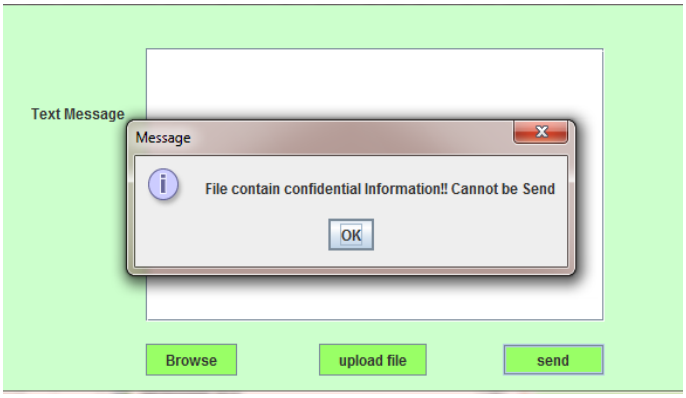

Figure 2. DLP system Interface showing blocking of text document contain confidential information.

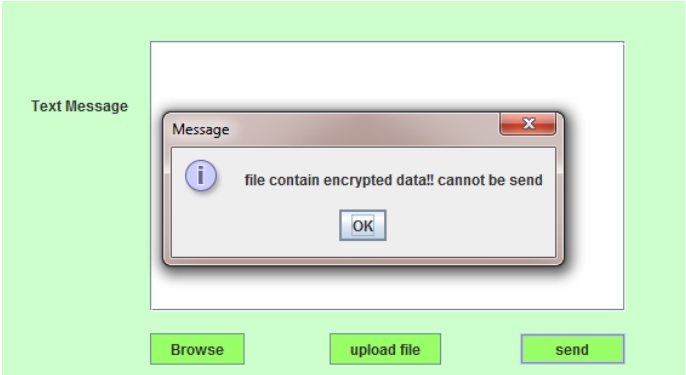

Figure 3. DLP system Interface showing blocking of MSWord document contain encrypted information.

The sample input of confidential and non-confidential text documents are collected from NYSK dataset , it is a collection of news articles. The input for encrypted data detection in word document is created by inserting DES encrypted ciphers into the word file. Both types of files are tested with our system and the performance of the system is impressive, ie., it identified the confidential and encrypted data in tested documents accurately.

V. CONCLUSION AND FUTURE WORK

An efficient Data leakage prevention with context based keyword matching for text documents and encrypted data detection of word documents is proposed. The proposed method will help to enhance the security capability of DLP system for protecting sensitive information within an organization. The method effectively check for information going out to the organization is confidential or encrypted. The current DLP systems does not deals with encrypted data detection. So it will improve the security features of the DLP system. Also for detecting small portion of confidential information in a non-confidential document can also be easily identified using the context based keyword matching method.

The following are suggestions for future work of this paper,

(i) we can able to include NLP(Natural Language Processing) for identifying synonyms of words in documents which represent any confidential information.

(ii) In this paper, we are only dealing presence of encrypted data not performing any cryptanalysis process for retrieving the information represented by the encrypted data. So this can also consider for future work.

ACKNOWLEDGMENT

I am greatly indebted to Dr.K.C Raveendranathan, Principal, LBS Institute Of Technology For Women and Dr Shreelekshmi R, Head of Department, Dept. of Computer Science & Engineering, for providing all the required resources for my thesis work. I would like to sincerely thank my project guide, Mrs Smitha E S, Dept of Computer Science &Engineering for her valuable suggestions and guidance. I would like to express my sincere gratitude to all teachers of computer science department for their moral and technical support throughout the course of this thesis work.

REFERENCES

[1] A. Shabtai, Y. Elovici, et al, A Survey of Data Leakage Detection and Prevention Solutions, Springer, 2012.

[2] “Data leak prevention,” Information Systems Audit and Control Association, Tech. Rep., 2010.

[3] W.W. Cohen, Learning rules that classify e-mail, in: Proceedings of the AAAI Spring Symposium on Machine Learning in Information Access, 1996, pp. 18–25..

[4] H. Drucker, D. Wu, et al, Support vector machines for spam categorization, IEEE Transactions on Neural Networks, vol. 10, no. 5, September 1999

[5] I. Androutsopoulos, J. Koutsias, et al, An experimental comparison of naive Bayesian and keyword-based anti-spam filtering with personal e-mail messages, in: Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, Athens, Greece, 2000, pp. 160–167.

[6] José María Gómez-Hidalgo, José Miguel Martín-Abreu, Javier Nieves, Igor Santos, Felix Brezo, Pablo G. Bringas, "Data Leak Prevention through Named Entity Recognition," socialcom, pp.1129-1134, IEEE Second International Conference on Social Computing, 2010.

[7] Zilberman, P., Katz, G., Elovici, Y., Shabtai, A., and Dolev, S., “Analyzing Group Communication for Preventing Data Leakage via Email”, In Proceedings of the IEEE Intelligence and Security Informatics (ISI 2011), Beijing, China, July 10-12, 2011.

[8] Amir Harel, Asaf Shabtai, Lior Rokach, and Yuval Elovici". M-Score: A Misuseability Weight Measure", IEEE Transaction on Dependable and Secure Computing, vol. 9, no. 3, may/june 2012.

[9] G. Katz et al., CoBAn: A context based model for data leakage prevention, Information Science, 2013, Elsevier.

[10] Filiol, E.: A new statistical testing for symmetric ciphers and hash functions. In: Deng, R., Bao, F., Zhou, J., Qing, S. (eds.) 4th International Conference on Information and Communications Security (ICICS 2002). LNCS, vol. 2513, pp. 342–353.Springer, Heidelberg (2002)

[11] Katos, V.: A randomness test for block ciphers. Applied Mathematics and Computation 162, 29–35 (2005)

[12] Mohammed M. Alani, Testing Randomness in Ciphertext of Block-Ciphers Using DieHard Tests, International Journal of Computer Science and Network Security IJCSNS), Vol.10, No.4, April 2010, pp. 53-57.

[13] Philip Penrose, Richard Macfarlane, William J. Buchanan"Approaches to the classification of high entropy file fragments",Digital Investigation ,372–384, 2013 Elsevier.