International Journal of Latest Engineering Research and Applications (IJLERA) ISSN: 2455-7137

Volume – 02, Issue – 03, March – 2017, PP – 07-13

Fast and Dynamic Detection of Transformed data Leaks on Mail Server

A. Rohitha Alka Mary[1], P. Sindhiya[2], S. Sindhu[3], Dr. K. Valarmathi M.E. PhD[4]

[1]*Department of Computer Science and Engineering, Panimalar Engineering College, India*
[2]*Department of Computer Science and Engineering, Panimalar Engineering College, India*
[3]*Department of Computer Science and Engineering, Panimalar Engineering College, India*
[4]*Department of Computer Science and Engineering, Panimalar Engineering College, India*

Abstract: Sensitive data leaks are a serious threat to the organizations and companies. Sensitive Data of companies includes Intellectual Properties, Patient or Financial Information, Credit Card Data, and Other Information. Sensitive data leaks can be perceived only through the aid of some tools (i.e. to find the sensitive data being transmitted). This project provides security against Sensitive data, using data leak detection technique on the transformed data. Mail Server of an organization examines contents of outsourced email messages for sensitive data. This paper utilises Levenshtein distance algorithm or Edit distance algorithm for detecting the transformed data leaks pattern. It returns the results of computation of the extent between two strings (i.e. edits that transform the word from one to another (Insertion deletion or substitution)). Shuffling algorithm is for detecting n-grams sampling in mail content. It also uses Lucene search framework for fast indexing of sensitive documents of the organization. Matching of sensitive data and contents outsourced provides sensitivity percentage . Threshold assigned is used for allowing the contents to be outsourced. The system achieves accurate sensitivity measure in the documents being transmitted.

Keywords: Data leak detection, levenshtein distance, lucene, Quote request.

1. INTRODUCTION

Insights enroll different data leak cases. Among them botches by human is one of the primary drivers of information misfortune. As per a report from Risk Based Security (RBS) the quantity of released Sensitive information records has expanded incredibly amid the most recent couple of years, i.e., from 412 million in 2012 to 822 million in 2013. The greater part of the data leak episodes are lead by intentionally arranged attacks[6], botches by human (allocating the wrong privilege) and inadvertent leaks [4] (outsourcing secret messages to unclassified records of email).Among them email leaks are all the more generally common. A few missteps done by email [9] clients are:

1. Neglecting to encode the messages which are delicate.
2. Messages being sent to a wrong individual.
3. Sending corporate data from the customized mail records of representatives.
4. Neglects to spare or document sends.

A set of complementary solutions is needed for examining, preventing and detecting data leaks, which may include data-leak detection data confinement stealthy malware detection and policy enforcement. Deep packet inspection [14] (DPI) is typically performed by data-leak detection (DLD) in network and any occurrences of Sensitive data patterns are searched.

For instance, Mail Server of an association can inspect the substance of outsourced email messages hunting down touchy information showing up in decoded messages. Touchy information is inadvertently spilled in the outbound movement by an allowable client. This is inadvertent data leaks. Forgetting encryption, carelessly forwarding an internal email and attachments to outsiders are main reasons for inadvertent data leaks. A malicious insider discloses the organizational secrets or intellectual property to outsiders. In Malicious [6] data leak a rogue insider or a bit of stealthy programming may take Sensitive individual or organizational information from a host. A release recognized is according to the information stockpiling. The Data leak detection [4] has following difficulties [1]:

1.1 Data Transformation. The Data outsourced is adjusted or changed starting with one then onto the next, so that it no longer matches with the initial content. Change incorporates inclusions, increments or substitutions [5].

1.2. Scalability. Plan or example of Sensitive information which are long ones prompt to substantial workload.

2. EXISTING SYSTEM

In a current framework direct accomplishment of data leak recognition require the Sensitive information which is a plain content. Be that as it may, this request is hostile, as it might ruin the security of the Sensitive data [3]. On the off chance that the framework is powerless, then it might uncover the plaintext Sensitive data [3]. Additionally, information proprietor may need to externalize the information spill identification to supporters, however might be unwilling to uncover the plaintext. There was no privacy in existing framework, so promoters can get to the data without information proprietors consent.

Existing business information spill recognition [4] counteractive action arrangements incorporate Symantec DLP, Identity Finder, Global Velocity, and Go Cloud DLP. Worldwide Velocity utilizes FPGA to quicken the framework. Each answer is pretty much in view of n-gram set convergence. Character Finder scans document frameworks for short examples of numbers that might be touchy (e.g., 16-digit numbers that are charge card numbers). It will not supply any in-depth similarity tests. Symantec DLP depends on both n-grams and Bloom channels. The simplicity of Bloom channel is that it has less space multifaceted nature. By the by, as illustrated in presentation, Bloom channel enrolment testing depends on disarranged n-grams, which creates circumstantial matches and false alerts. Sprout channel organized with somewhat number of hash capacities has clashes, welcomes extra undesirable false positives.

3. PROPOSED SYSTEM

In the proposed system a data-leak detection solution allow the mail to be outsourced from organization by checking the sensitive data matching. Lucene search engine framework is designed and implemented for indexing the sensitive contents of the organization. Levenshtein [10]-distance technique is to avoid data leak and also provide privacy preserving [3] to Sensitive data. Two most important players in the proposed model is

- **Data Owner**. One who owns the Sensitive data and authorizes the DLD provider to inspect the network traffic from the organizational networks
- **Mail Server**. The network traffic is inspected by the DLD provider for probable data leaks. The content in file system or network traffic (over supervised network channels) is made available to the inspection system. A supervised network is where the content in it can be extracted and checked by an authority. Authority has the threshold for every categorized position of users

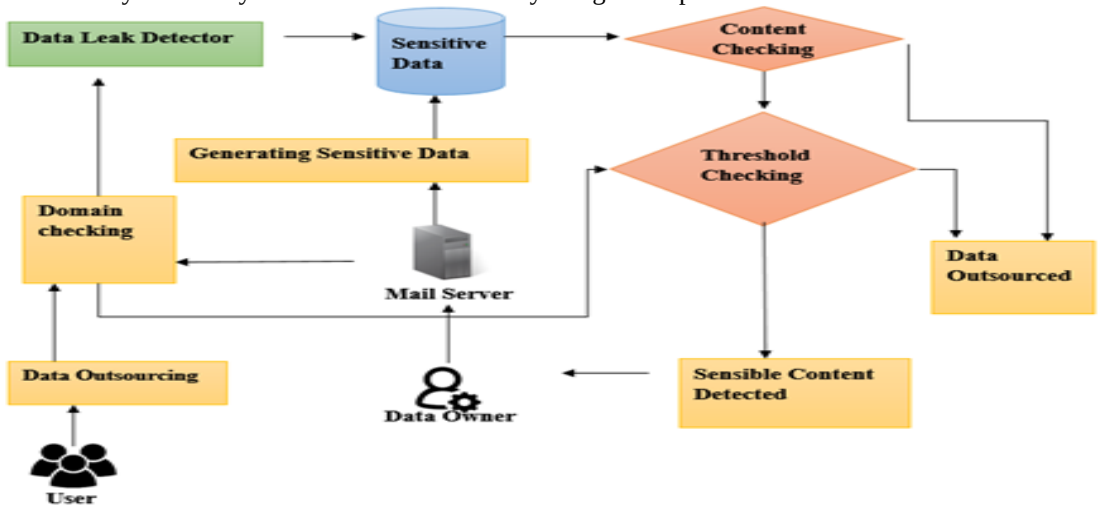

Fig 1.Architechture diagram

4. MODULES AND OVERVIEW

4.1 . User Registration

User registers in the mail server with their name, authorized job position and their authorized e-mail domain. Then users transfer their file to other users using their registered mail ids. The content can be of any file (text, document).

4.2. Build data leakage detection framework.

Data owner generates a Sensitive data and stores it in mail server. Data owner creates the directory for lucene search framework [8] and other data leakage detectors. Data owner contains much Sensitive information [3] about their authorized customer's details, information technology source, and database and server details. This Sensitive information is maintained by Data Leak Detector. Thus data leak detection [4] mechanism is performed. The DLD consist of lucene search engine framework, levenshtein [10] distance algorithm and our own shuffled checking algorithm.

4.2.1. Lucene search framework [8].

The Lucene search framework indexes the sensitive contents being stored or maintained by the data owner. Stop words are removed from the enlisted words. The remaining words are indexed along with its posting list. The indexing increases searching efficiency rapidly.

4.2.2. Levenshtein distance [10] algorithm.

This is also called as edit distance algorithm since it detects the number of edits to transform a content into another. The edits include insertion, deletion or substitutions.

4.2.3. Shuffling algorithm.

This detects the 3-grams sampled sequences in the transformed content.

4.3. Content Outsourcing with DLD Checker.

All the outsourced contents are checked with Sensitive data. Sensitive data's are maintaining in index file. Using this index file, DLD identifies the Sensitive data concurrently with domain filtering and threshold assigning based on their email domain.

4.3.1. Domain Filtering.

There is no filtering or checking of sensitive content if sender and receivers domain matches (Within the organization).The filtering is done when domain differs.

4.3.2. Threshold Filtering.

Each Employee or user is allotted a threshold percentage based on their job position .for example general manager-90% .This is done by the Admin.

DLD will check every line of the sending data with the Sensitive file. In proxy mail server the every occurrence of transformed contents are filtered. All users' details are retrieved using their email id. Then by using by lucene framework search engine, levenshtein [10] distance checking and shuffling [1] [2] algorithm the transferred content is being tested .

4.4. Sensitive Data Detection and Quote Request.

If any Data is leaked, DLD will detect the Sensitive data. Here DLD will check not only the Sensitive data but also it will check some access condition. Data owner (Admin) maintains common access condition to every file. For the purpose of false alert, we maintain threshold of every domain and users position. If the Sensitive content percentage of transferred file exceeds the threshold percentage, an alert mail is triggered to Data owner (Admin) of the proxy mail server. Alert mail consists of entire details about the users along with the Sensitive contents being transferred in mail being outsourced.

After the filtration of mail in Mail server the user can claim or quote the request to admin with the reasons or details for outsourcing the sensitive content. Finally the Mail server admin reviews the quote request mails from the quote users whether to allow or deny the mails.

5. RELATED WORK

In Gyrus: A Framework for User-Intent Monitoring of Text-Based Networked Application [18] a way that ensures the matching of system's behaviour to user's intent is proposed. It will perform better than traditional systems of security, since the approach used is attack agonistic. The key components to the approach mentioned above are: First, The objective of user is captured through their communications and interactions with application. Second, the resulting output of the system can be portrayed back to the user's interactions. To demonstrate its working Gyrus is developed, a research prototype that observes user interactions for tasks that commonly occur such as email sending, instant messaging, online financial services and online social networking. Gyrus provides security to the above applications from behaviours found malicious such as wire fraud and spamming by allowing only outgoing traffic with content that matches the user's intent. Gyrus captures user intent by displaying user's input on the screen so the users input are confirmed by the user. Gyrus also focuses on what is being displayed to the user instead of what typed or clicked by the user. This is called "what you see is what you send (WYSIWYS)" policy. The paper Quantifying Information Leaks in Outbound Web Traffic [19] information leak capacity in network traffic is analysed and quantified. The maximum volume of occurrence of sensitive data is measured and constrained not the presence of it. Most of the traffic in network is repeated or resolved by information which is external, such as protocol specifications or server sent messages .This is an important advantage. True information flowing from a computer can be isolated and quantified by this filtering technique. It also presents measurement algorithms for the Hypertext Transfer Protocol (HTTP), the web browsing protocol. When applied to the traffic in real web browsing, the algorithms were able to discount 98.5% of measured bytes and thus isolate information leaks efficiently. In the Sampling Techniques to Accelerate Pattern Matching in Network Intrusion Detection Systems [20] a large part of the text is skipped, thus organising less bytes. A slight number of false alarms are the price to pay and this requires a stage for confirmation. Therefore, a double stage matching scheme to two different new automata's are provided. The Rapid and parallel content screening for detecting transformed data exposure [2] Provides Transformed data leak detection idea. Here the data transformed from one form to another is detected by parallel screening of contents or data. This system achieves accuracy in transformed data recognition comparing with state-of-the-art inspections. The GPU is parallelised for efficient data matching.

6. ALGORITHM USED

Levenshtein [10] distance is contrived by Russian researcher Vladimir Levenshtein [10] in 1965. This likewise called as edit distance (inclusion, expansion, substitution).The zones where Levenshtein distance algorithm is utilized are

- Checks spelling
- Analysis of DNA
- Plagiarism location
- Speech acknowledgment.

It is a metric which measures the distinction between two string successions. The base number of character alters (i.e. inclusions, cancellations or substitutions) [5] that can change a word into the other.
Scientifically, the Levenshtein [10] separate between two strings a, b (of length |a| and |b |respectively) is given by

Lev (|a|, |b|) =

- max (i , j) if min(i,j)=0
- min(lev(i-1,j)+1,lev(i,j-1)+1,lev(i-1,j-1)+cost otherwise

Where **cost** is **1** if a character in comparison matches
0 if a character in comparison does not match.

Algorithm 1 Levenshtein Distance
Input: A matrix of two strings in row and columns respectively.
Output: Edit Distance d (n, m)

1. Read the two strings s and t
2. initialize n to length of string s
3. initialize m to length of string t
4. **if** n=0 **then**
5. return n
6. **else if** m=0 **then**

7. return m
8. **end if**
9. **end if**
10. initialize Row as 0..n
11. initialize Column as 0..m
12. **while** row(i) is 0..n and column(j) is0..m
13. **do**
14. examine string s row wise
15. examine string t column wise
16. **if** s[i]==t[j] **then**
17. cost is set to 1
18. cost is set to 0
19. **end if**
20. perform d(i,j) as min(above , left, diagonal)
21. set the above to d(i-1,j)+1
22. set the left to d(i,j-1)+1
23. diagonal to d(i-1,j-1)+cost
24. **end while**
25. return d(n,m)
26. **end**

7. EXAMPLE

Table 1 . Step by step process of levenshtein distance algorithm

		J	O	C	K	S
	0	1	2	3	4	5
L	1					
O	2					
C	3					
K	4					
E	5					
R	6					

Step1: Initialisation

		J	O	C	K	S
	0	1	2	3	4	5
L	1	1				
O	2	2				
C	3	3				
K	4	4				
E	5	5				
R	6	6				

Step2: Lev(a,b)of first column

		J	O	C	K	S
	0	1	2	3	4	5
L	1	1	2			
O	2	2	1			
C	3	3	2			
K	4	4	3			
E	5	5	4			
R	6	6	6			

Step3: Lev(a,b)of second column

		J	O	C	K	S
	0	1	2	3	4	5
L	1	1	2	3		
O	2	2	1	2		
C	3	3	2	1		
K	4	4	3	2		
E	5	5	4	3		
R	6	6	6	4		

Step4: Lev(a,b)of third column

		J	O	C	K	S
	0	1	2	3	4	5
L	1	1	2	3	4	
O	2	2	1	2	3	
C	3	3	2	1	2	
K	4	4	3	2	1	
E	5	5	4	3	2	
R	6	6	6	4	3	

Step5: Lev (a,b)of fourth column

		J	O	C	K	S
	0	1	2	3	4	5
L	1	1	2	3	4	5
O	2	2	1	2	3	4
C	3	3	2	1	2	3
K	4	4	3	2	1	2
E	5	5	4	3	2	2
R	6	6	6	4	3	3

Step6: Lev(a,b)of fifth column

8. RESULT AND FUTURE ENHANCEMENT

The Sensitive content percentage if any in the outsourced mails exceeds the threshold value assigned to the sender, the mail is not outsourced. Meanwhile alert message is sent to the admin .But the System Does not detect sensitive content in images or tables being outsourced. It can detect up to 3-gram sampled sequence in the transformed content. Hence as a future enhancement Images can be inspected for sensitive data. Content Sampling up to n-grams can be made detected.

9. CONCLUSION

Hence fast detection of data-leakage framework to avoid sensitive data exposure is developed and privacy-preserving to sensitive data is provided.

10. REERENCES

[1]. "Fast detection of transformed data leaks" Xiaokui Shu, Jing Zhang, Danfeng (Daphne) Yao and Wu-Chun Feng. (2016).

[2]. "Rapid and parallel content screening for detecting transformed data exposure" X.Shu, J.Zhang, D.Yao and W-C Feng, Apr. /May 2015.

[3]. "Privacy preserving detection of sensitive data exposure" X.Shu, D.Yao and E. Bertino ,May 2015.

[4]. "Data leak detection as a service," in Proc. 8thInt. Conf. Secure. Privacy Commun. Netw. (Secure Comm), Padua, Italy, Sep. 2012, pp. 222–240. X. Shu and D. Yao

[5]. "Secure and efficient outsourcing of sequence comparisons," inProc. 17th Eur. Symp.Res. Comput. Secure, 2012, pp. 505–522, M. Blanton, M. J. Atallah, K. B. Frikken, and Q. Malluhi,

[6]. "Towards mechanisms for detection and prevention of data exfiltration by insiders: Keynote talk paper," inProc. 6th ACM Symp. Inf., Comput. Commun. Secure. (ASIACCS), E. Bertino and G. Ghinita, 2011,pp. 10–19.

[7]. "iLeak: A light weight system for detecting inadvertent information leaks" V.P.Kemerlis, V.Pappas, G.Portokalidis and A. D. Keromytis, Oct 2010.

[8]. "Design of Data Duplicate Detection System Using Lucene", Yuahue Zing, KuiYi, Rihua Ziang June 2010

[9]. "Information leak detection in financial e-mails using mail pattern analysis under partial information", C.Kalyan and K.Chandrasekaran, in Proc. 7th WSEAS Int.Conf. Appl.Informat.Commun.2007

[10]. "Computation of Normalized edits distance and application" A.Marzel, E.Vidal, Aug 2002

[11]. X. Shu, J. Zhang, D. Yao, and W.-C. Feng, "Rapid screening of transformed data leaks with efficient algorithms and parallel computing," in *Proc. 5th ACM Conf. Data Appl. Secure . Privacy (CODASPY)*, San Antonio, TX, USA, Mar. 2015, pp. 147–149.

[12]. L. De Carli, R. Sommer, and S. Jha, "Beyond pattern matching: A concurrency model for stateful deep packet inspection," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secure*, 2014, pp. 1378–1390.

[13]. A. V. Aho and M. J. Corasick, "Efficient string matching: An aid to bibliographic search," *Commun. ACM*, vol. 18, no. 6, pp. 333–340, Jun. 1975.

[14]. H. A. Kholidy, F. Baiardi, and S. Hariri, "DDSGA: A data-driven semi-global alignment approach for detecting masquerade attacks," *IEEE Trans. Dependable Secure Comput.*, vol. 12, no. 2, pp. 164–178, Mar./Apr. 2015.

[15]. S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman, "Basic local alignment search tool," *J. Molecular Biol.*, vol. 215, no. 3, pp. 403–410, Oct. 1990.

[16]. Yeongjin Jang, Simon P. Chung, Georgia Institute of Technology,yeongjin.jang@gatech.edu, pchung34@mail.gatech.edu "Gyrus: A Framework for User-Intent Monitoring of Text-Based Networked Applications".

[17]. Kevin Borders, Web Tap Security, Inc. Ann Arbor, MI kborders@webtapsecurity.com "Quantifying Information Leaks in Outbound Web Traffic".

[18]. Domenico Ficara Dept. of Information Engineering, University of Pisa,ITALY "Sampling Techniques to Accelerate Pattern Matching in Network Intrusion Detection Systems"

