International Journal of Engineering &Technology, 7 (1.7) (2018) 230-235

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Information leakage detection and protection of leaked information by using the MAC-IP binding technique

B. Raja Koti[1]*, Dr.G.V.S. Raj Kumar[2]

[1]Research scholar
[2]Associate Professor
[1,2]Department of Information Technology, GITAM Institute of Technology,
GITAM (Deemed to be University), Visakhapatnam, 530045, India
[*]E-mail: rajakoti503@gmail.com.

Abstract

The Digital world is advancing in terms of technological development day by day, resulting in an instantaneous rise in Data. This massive amount of Data has introduced the thought of Big Data, which has attracted both the business and IT sectors leaving the scope for huge opportunities. In turn, securing this massive data has become a challenging issue in the field of Information and Communication Technology. In this paper, we have carried out the work on business information sharing data which contains some sensitive information to investigate the security challenges of data in the field of business communication**.** The article an attempt is also made to identify the user's intention or behavior during the navigation of data. The greatest challenge that is associated here is to prevent the integrity of the data while sharing the data from organization to the third party, where there exist huge chances of data loss, leakages or alteration. This paper highlights the concepts of data leakage, the techniques to detect the data leakage and the process of protecting the leaked data based on encrypted form.

Keywords: *AES, Data leakage, Data Protection, Data Privacy, Guilt Agent, MAC-IP Binding.*

1. Introduction

BIG data systems in large organizations today have become ever more complex, multi-tiered, multi-vendor, distributed physically or logically. The complexity gives rise to a multi-faceted network of devices and applications potentially representing an attack vector or entry point into the corporate critical network. In addition, organizations were constantly faced with an enormous volume of newly-discovered software vulnerabilities and exposures. The root cause for a majority of the cyber-security problems largely centered on software vulnerabilities, therefore, organizations must maintain effective vulnerability management programs including identification, assessment, reporting and remediation. Faced with large backlogs of unresolved vulnerabilities, organizations can become reactive and unprepared for new influxes of vulnerabilities and shift threat landscape, particularly, if the data is left untouched for consecutive months, the possibilities of coordinate attacks and exploiting attacks may increase proportionally. Technically, it is also very challenging for large organizations to secure their critical digital assets and cyber-infrastructure, due to the complex configurations and constraints [1], [2]. Therefore, it is imperative to deepen our understanding of the proliferation of newly discovered vulnerabilities [3]. A statistical framework to discover trends and patterns of invulnerability disclosures enables organizations to become more proactive in managing these vulnerabilities.

One of a major challenge being faced by most of the business companies is with respect to storage of the voluminous data generated globally at the affordable cost and makes available all the time. In order to manage the huge data, most of the organizations mostly relied on Cloud computing technologies. Cloud computing provides easy access and high-performance computing on the data. However, cloud computing technologies leave behind many unsolved queries, such as; where the data gets stored, who will manage the data, how to share the data and about the security concerns of the data being stored [4].

It is generally agreed that data security is not only a technical issue but also an important social issue. The status of a company can be majorly get damaged due to its well-given away approach towards data breaches. In recent years, there has been a large amount of research on information security and sensitive information (SI) protection. National Institute Standards and Technology (NIST) (2013)[5] defined as "the protection of information and information systems from unauthorized access, use, exposé, trouble, modification, or damage in order to provide confidentiality, integrity, and availability". In contrary, sensitive information is considered as " the loss, misuse, or unauthorized access to or modification of the data, that could adversely affect the national interest or the conduct of federal programs, or the privacy to which individuals are entitled under 5 U.S.C. Section 552a (the Privacy Act) [6], but that has not been specifically authorized under criteria established by an Executive Order or an Act of Congress to be kept classified in the interest of national defense or foreign policy [7]. In this paper, we present a methodology of protecting the information system from unauthorized access and data leakage.

Data leakage can be accomplished by simply remembering what was seen, by physical removal of devices like tapes, disks, and reports or by restrained ways such as data hiding. This can be achieved by using the electronic resources or by the usage of a physical method. The terms Data Leakage and Information Leakage are synonymous. The term "unauthorized" does not mean only means of intentional or malicious data. Unintentional or inadvertent data leakage is also deemed to be unauthorized. In order to execute the proper protective measures, one needs to understand the very objective of the data that needs to be protected in prior. Data leakages have become most frequent, leading to huge losses (in terms of finance and also confidential information) to organizations. In order to safeguard the information loss, security modeling systems are considered, where the security variables from different security evaluation techniques are taken into consideration, with the intuition of increasing the complexity of Sensitive information. However, if the data is leaked from within, identifying the guilt agent is a challenging task. Using mathematical probability, we are aware of the probable guilt agents and in some cases, we can also find the guilt agent, but we are unable to protect this leaked data.

In this article, a methodology is proposed to protect this leaked data from the unauthorized personnel and also it suggests a way to find the proper guilt agent with the usage of MAC-IP Binding technique, security algorithm and hash function also using for secure the leaked data. However, in this paper, we have confined ourselves for analyzing the cases to identifying guilt agents in specific to file transfer.

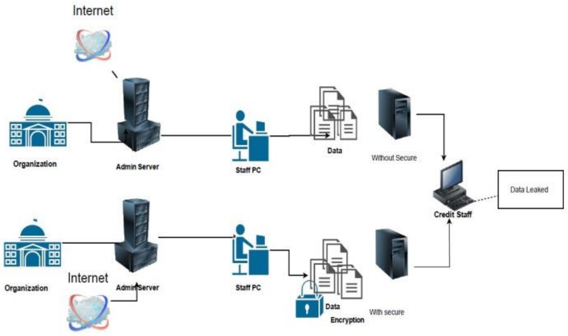


Fig.1: Data leakage with /without secure

The proposed methodology helps to identify such data leakages and find out the guilt agent. Also, it provides a platform to showcase how to protect this leaked data from being accessed by unauthorized sources. The rest of the paper is organized as follows; section II of the paper deals with the review of work carried out in this direction, in section III, the definitions and methodology of Data Leakage together with Guilt Agents identification are presented. Section IV highlights the results derived along with discussion and the concluding section V summarizes the article.

2. Related work

The review of Data Leakage techniques [8] highlights that; one can detect the guilty agent without changing the reliability of the original data. This Data Leakage detection concept was proposed by Papadimitriou and Hector Garcia Molina [9, 10], which can let us detect the guilty agent without changing the reliability of the original data. An initiation of data leak is a terrifying proposition. Security practitioners have always had to compact with data leakage issues that take place from email and other Internet channels. But now with the usage of mobile technologies, it's easier for data loss to happen, whether by chance or maliciously. There is a long history of research that aimed to enhance the safety of information flows. Recurring themes in this research include; information risk management, the costs and benefits of information security, and methods to determine the value of Sensitive Information. According [11] suitably stated that information security is information risk management. In the article [12], the Annual Loss Expected and Return on Security Investment have explained the proportional relationship between cost and security. Gordon [13] proposed a security model in using an economic viewpoint the author explains information security in relative to risk and describes quantitative and qualitative expressions of information risk. For this reason [14], the OCTAVE approach focused on risks, rather than vulnerabilities. Therefore, OCTAVE performed technical evaluations after organizational evaluations. Carnegie Mellon University (CMU) developed the OCTAVE (The Operationally Critical Threat, Asset, and Vulnerability Evaluation) approach [15]. The authors conclude that organizational risk is associated with technical vulnerability. Gordon suggested optimal security investigation ranges using a graph representing the correlation between security investigation level differences and the expected loss from every vulnerability, according to the information security level [16] proposed the utilization of information lifecycle management (ILM) according to its value discussed policy-based ILM for a large-scale information file system. In [17] described various information security approaches such as access control and secure communication, for information system security based on information value. In addition, [18] presented information valuation methods for ILM Altogether, it is clear that there are many approaches to information security and information modelling [19]. Information data leak detection/prevention solutions include Symantec DLP [20], Identity Finder [21], Global Velocity [22], and Go Cloud DLP [23]. However, no proposal was presented by the authors about the prevention of leaked data, and the present proposed article makes an attempt in this direction.

3. Definitions and Methodology

3.1. Guilt Agent

A guilt agent is a person who transmits authorized data (or information) to an unauthorized organization or person. This guilt agent may be an employee or a person with access to the sensitive data of the company. Recent years have seen an increase in the number of agents being in proportion to the population size**.** Those agents, with their autonomous reasoning and decision-making capabilities, can engage in complex interactions on behalf of their owners. There is no single-agent system in specific. Instead, agents usually live in a society of agents, which is known as a multi-agent system (MAS). Usually, agents in MAS represent various stakeholders, each with distinct interests and objectives. They try to pursue their own objectives, even at the cost of others. Various modalities of identifying guilt agents can be possible in E-mails, Chatting, using USBs, and Smartphone. However, in this article, we have confined ourselves towards analysing the cases of identifying guilt agents in specific to file transfer.

3.2. Identifying the guilt agent

Now, when the MAC-IP address is linked with the log file in the server, when there is an unauthorized record of a move(s) in the protocol, and the track of the record can be detected. Using the MAC-IP address linked to the particular log file and the timestamps allotted by these records, the guilt agent can be detected when he/she makes a move without the permission of the super-user or Admin [24]. When the wrong or mismatch move happens, the system will get the alert message with that alert, Admin will verify the mismatch user details and can find out the guilt agent among the network. During further analysis, the admin can trace whether the user has done it intentionally or by mistake.

3.3. Data Encryption processes

One of the means of protecting the data is to generate a key having a composition of isolated encryption and authentication primitives. The cipher-text is generated by encrypting the plaintext and then attaching a MAC of the plaintext. This resembles the working style of SSH (Secure Shell). The cipher-text is generated by attaching a MAC to the plaintext and then encrypting data and this resembles the working style of SSL (Secure Sockets Layer). The cipher-text is generated by encrypting the plaintext and for the encrypted plaintext MAC will be attached, in similar lines to IP-SEC (Internet Protocol Security). Among the above mentioned three methodologies, only the performance of an algorithm depends on the features considered during the Encryption and MAC functions. By using this type of Data Encryption one can provide the high degree of security to the data globally.

3.4. Protecting the leaked data

Even after finding the guilt agent, the transfer of data is successful to the external person/ company. To protect this data from being accessed by the unauthorized users, our method proposes the underlined system. The MAC-IP addresses that are linked to the data in the log file always checks for the correct MAC and IP address. If it is leaked and transferred to another organization or person that are trying to access this data, the file of data detects a mismatch in its own MAC and IP addresses and realizes it is out of its original user or authorized the user. Now, this authorized user that receives files containing that data gets automatically encrypted by the used encryption algorithm.

Table 1: Data protection for various data states

Type	Description	Proposal Goal
Data-at-rest	Information that in an organization like files, servers, document management Systems and email servers.	Content detection
Data-in-motion	Organization data is restricted to network traffic such as web traffic.	Identify the communication of sensitive data and encrypt the data
Data-in-use	Information currently used at the endpoints such as http, https, print, file to USB and outlooks.	Prevents unauthorized use of data

Data-at-Rest: In order to identify the content, different solutions are too developed. It helps to detect the sensitive data reside in separate locations by performing scanning in laptops, FTP servers, SMTP servers and in the database [33]. Techniques for content discovery are as follows: 1. *local scanning of data*- In this technique, an agent is installed on the host machine that regularly scans the content which is stored in the files. It relocates, encrypts and quarantines the content after finding anything malicious in it. During the process, agents are always active, execute a policy even when devices are not placed locally and are not connected to the network. 2. *Remote Scanning*- Scanning is performed by remotely located computers by maintaining a connection with server and application level protocols.

Data-in-Motion: Network-based solutions are deployed on company's gateway. Gateway computer searches the sensitive content and blocks the malicious activities immediately that violate the policy. These solutions capture the full data and perform the content analysis in real time [34], [35].

Data-in-Use: Local agents and host machines regularly check sensitive data, such as data copied from one location and pasted into another location, data from the print screen, unauthorized data transmission and copying data to a USB drive /CD drive or a DVD drive [36].

In this paper, the technique of MAC-IP binding, the bind value is considered by collecting all the MAC and IP values of all the clients in a separate file. For each of these values, a hash value is generated against each client. Basing on request, the hash value stored in the log file is compared to that of the request of the new IP and MAC values received and then comparing the hash values. This bind value is considered for the identification of the guilt agent and protects the data that is transferred without authentication. Then after for every request from a client to access the server, then our server will check the stored file every time that the bind value is correct or not. In the bind value is equal then connection established to communicate. If bind value is not equal it simply saves the MAC and IP addresses to stored in the other file and connection will be accepted, and the file which is downloading from the server or sharing from any other connected clients all those movements are recorded with time stamps and then all the file are also encrypted. In some cases, we are doing corrupting data and filling with garbage values. Even though it is secure, there is a chance of risk of data. To eliminate, this risk our article proposes a new method that data will be encrypted by using AES algorithm which can be accessed only by the admin. The data which was sent to users or which is available in the cloud to access for admin, in that data we will identify the important and passwords related data with the help of data analysis and then we will provide the security for that data. So, that should not be open in any other system even it received by the other authorized user, so for that we are using this encryption concept in this model, to encrypt the data which was delivered to other network user or the authorized user by doing this we can provide more security for the sensitive data with the help of the proposed algorithm in section-4.

AES algorithm is considered for model building, it is a mathematical symmetric cryptographic algorithm; its main strength is in key lengths in this AES we can choose the various key lengths. We can choose a 128-bit, 192-bit or 256-bit key, and to make it very stronger than the DES 56-bit key. During the implementation of DES, we have referred to the work carried out by the Feistel network where the network gets divided into two blocks during the encryption phase. AES is a symmetric algorithm which includes a series of substitution and permutation steps to create the block for encryption which is encrypted block. Firstly, DES implementation was made a great enrichment to data security, but one could say that for the AES algorithm has been better than the DES in terms of random key generation [37]. AES was also taken as the random bit key generated based on time so it shows more secure for the encryption to the data which are received by the unauthorized user. When the bind value is correct then the connection will be accepted and communication will be done normally in the below figure-2 shows the flow of data step by step whenever client request the server then server will verify the bind value of the requested client and allowed to connect to communicate with the server after completing of the client request connection will close. The miss-matches against the bind value all those files will be encrypted and sent so that data will not get access to miss-users.

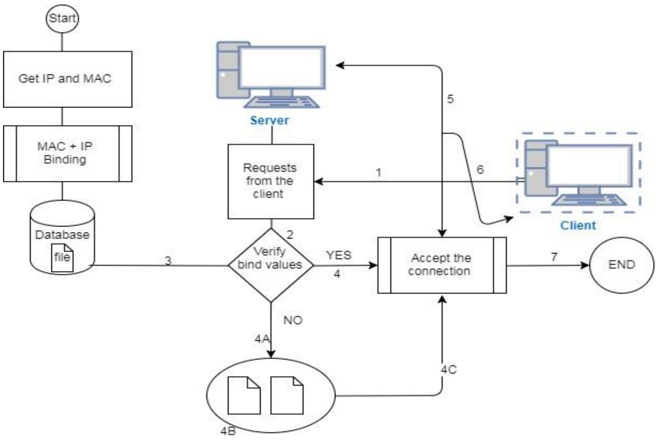


Fig. 2: Data flow Diagram

Step 1: Get IP & MAC Bind store in a file in the server

Step 2: If any request from a client

Step 3: Server check bind and response

Step 4: a) If the bind is matches or not matched, the server will respond and sent data to a client.

b) In the case of miss matched data will be encrypted and sent

Step 5: The server will separate miss matches requests and get alerts If in the case of client sent any file to another client Server gets IP and MAC of that client

Step 6: Server record all the movements of its users

Step 7: And checks all log records of it Sever identify that miss match request. Finalize the data leaked or not.

4. Results and Discussions

Let U denotes the group of users and C denotes the group complete dataset in organization, G_a the event that agent ‘a’ is guilty agent G_a and we have to compute the probability P_b of an agent for being a guilty agent G_a when the leak data L is given then the probability is denoted by $P_b = \{G_a \mid L\}$. We assume that $\forall S_i \in L$, where S is Sensitive data from the dataset, $i = \{\upsilon_1, \upsilon_2, \ldots, \upsilon_n\}$, there can be only two possible ways, one is that any single agent from the set of $Z\ S_i = \{a \mid S_i \in X_j\}$ has leaked S_i to target ‘t’, where $Z\ S_i$ is the set of agents having S_i in their allocated dataset X_j $\forall$ j={1, 2, . . . , m} or the target ‘t’ retrieved the data S_i by guess or through any other mean without the intervention of any agent ‘a’. The probability to leak any data object S_i from the leak dataset L. i.e., P_b {leak S_i to L} is equal $\forall$ a $\in$ Z S_i if it is leaked by any agent a $\in$ Z S_i otherwise P_b {leak S_i to L} is α if it is obtained by the target ‘t’. We consider that ‘a’ decision to leak any data S_i is autonomous to the leaking of other data S_i, $\forall S_i$, $S_i \in L$ where $S_i \neq S_i$. $P_b\{G_a \mid L\}$ of the agent ‘a’ to be a guilty agent G_a is computed as given in Eq below.

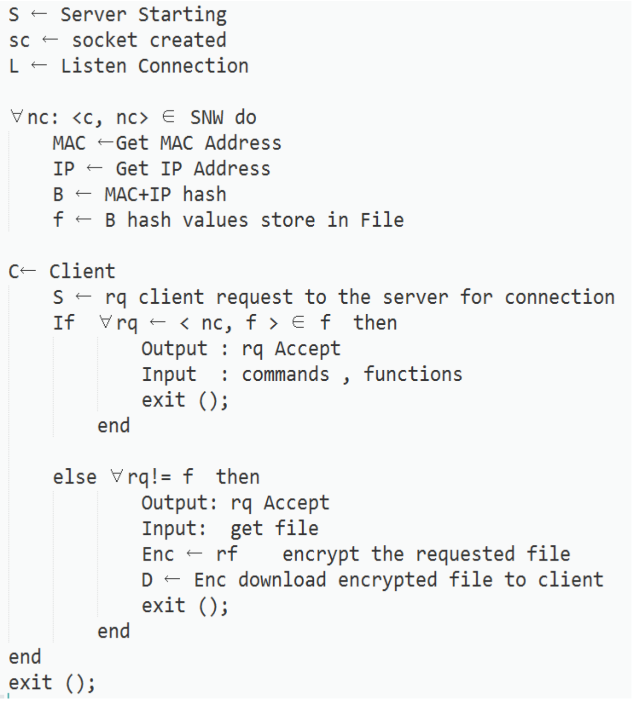
$$P_b\{G_a \mid L\} = 1\text{-} \prod_{s_i \in L \cap X_j} \left(1 - \frac{(1\text{-}\alpha)}{CS_i}\right)$$

Table 2: Evaluation of various parameters of α

α	$P_b\{G_a \mid L\}$	$P_b\{G_a \mid X\}$
0	1	1
0.1	0.931	0.998
0.5	0.905	0.987
0.9	0.217	0.368

The reason can be explained as chances of guessing the data becomes high. We observe that as α increase the value of all the two parameters decreases. The reason can be explained as more and more data is allocated and it becomes typical to identify the agent. We also observe that with increment in the value of α, the value of α firstly increases and then decreases. The reason can be explained as in starting difference value increases with increment in α. From Table 2, we observe, when α values are high i.e α = 0.5, then $P_b\{G_a \mid L\}$ = 0.905, $P_b\{G_a \mid X\}$ = 0.987, which are also very high. When α and weight factor are extremely high i.e α = 0.9 then $P_b\{G_a \mid L\}$ =0.217337, $P_b\{G_a \mid X\}$ = 0.386081, which are also acceptable that proves the efficiency of the approach.

4.1. Proposed Algorithm:



```
S ← Server Starting
sc ← socket created
L ← Listen Connection

∀nc: <c, nc> ∈ SNW do
    MAC ←Get MAC Address
    IP ← Get IP Address
    B ← MAC+IP hash
    f ← B hash values store in File

C← Client
    S ← rq client request to the server for connection
    If  ∀rq ← < nc, f > ∈ f  then
            Output : rq Accept
            Input  : commands , functions
            exit ();
        end

    else ∀rq!= f  then
            Output: rq Accept
            Input:  get file
            Enc ← rf    encrypt the requested file
            D ← Enc download encrypted file to client
            exit ();
        end
end
exit ();
```

Fig. 3: Proposed Algorithm

To evaluate the performance of the proposed approach, a simulated environment is created such that, it considers the sensitive data leakage problem. The proposed framework is implemented in Python environment. In our experimental setup, we have considered |S| = 200, |a| = 50 and |L| = 100. The performance of the approach is evaluated against the parameter weight factor W_F and defined in terms of relative value which can be defined as the ratio of the sum of all the allocated data to the total number of datasets.

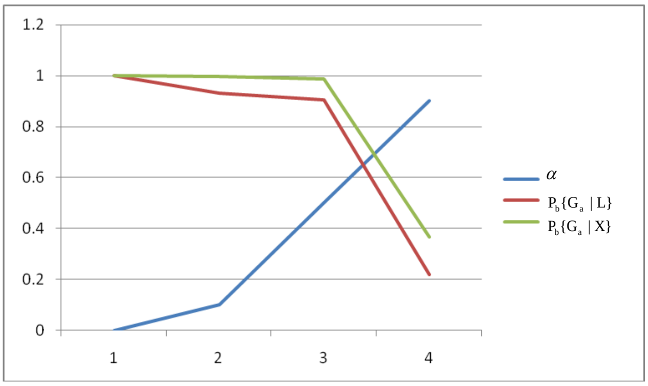


Fig. 4a: Evaluation of various parameters

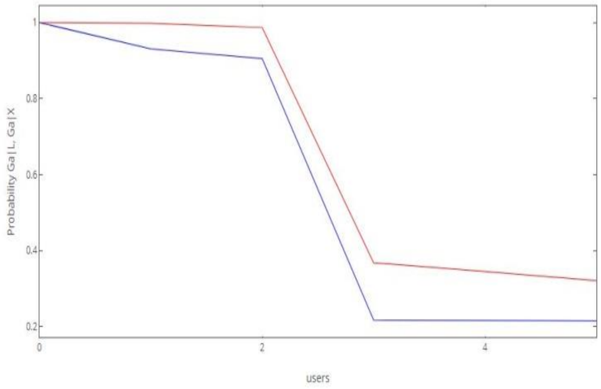
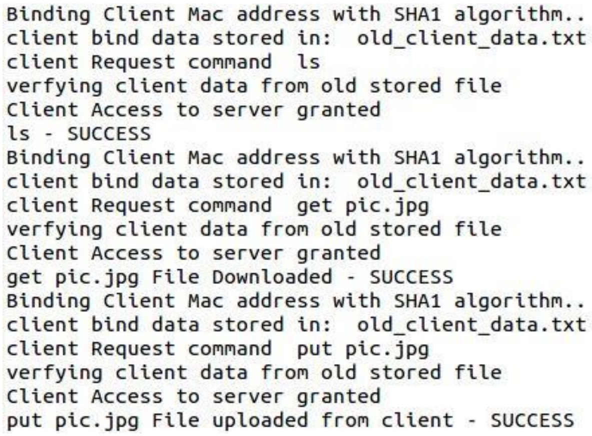


Fig 4b: Evaluation of probability $P_b\{G_a \mid L\}$, $P_b\{G_a \mid X\}$.

From the figure-4b which shows the two possible ways, one is that any single agent from the set, the probability P_b computes of an agent for being a guilty agent G_a when the leaked data L. Then the probability P_b leaked X to target the set of agents G_a having S_i in their allocated dataset.



```
Binding Client Mac address with SHA1 algorithm..
client bind data stored in:  old_client_data.txt
client Request command  ls
verfying client data from old stored file
Client Access to server granted
ls - SUCCESS
Binding Client Mac address with SHA1 algorithm..
client bind data stored in:  old_client_data.txt
client Request command  get pic.jpg
verfying client data from old stored file
Client Access to server granted
get pic.jpg File Downloaded - SUCCESS
Binding Client Mac address with SHA1 algorithm..
client bind data stored in:  old_client_data.txt
client Request command  put pic.jpg
verfying client data from old stored file
Client Access to server granted
put pic.jpg File uploaded from client - SUCCESS
```

Fig. 5: successfully connected to server and done few activities.

From the above figure-5, one can witness the way how a file can get stored and the process of getting the download in the server. As the first step, server will initiate and upon listening for connection, and consecutively upon receiving the requests from the client, it considers the IP and MAC addresses, and compares with the available allowed pool of connections available at the server using the bind value of it then if it matches with the bind value then server will get honouring the responding requests of the client. Even if there is a mismatch, the server will respond to the client requests but will be confined to operate with minimal functionalities. The restricted data, in this case, will be getting converted into an encrypted format, such that no data can be accessed. This bind value that is fixed will be generated among all the clients within the organization network, by using the SHA family algorithm which was more secure than others. In network traffic, each dataset have some sensitive data, and each set have some different type of data will be there from the traffic, and the public parameters, our model produces outputs as the alert (indicating possible of data leak) of possibility the only leakage is finalized. For all the data-leak matching instances detection alerts.

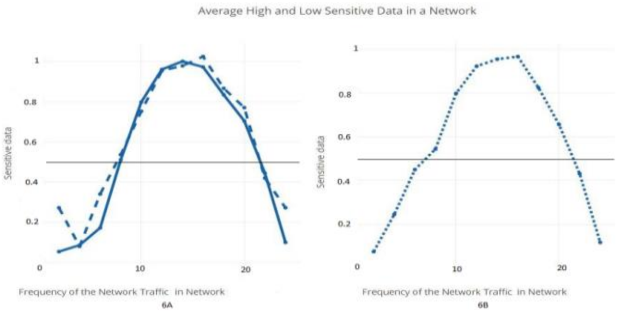


Fig: 6. Sensitive Data with in total data

From above figure-6 it is observed that the above three lines, in 6A lines, are related to the sensitive data in a network, the plain line indicates the total data in the network which is shared data to the users. The other line in 6B indicates the network traffic how the data was moving from server to client. The dotted line indicates the sensitive data that present in the total shared data for the user/client. So, by using the dotted line we can provide security to the data that should not access by any other user who does not belong to the organization.

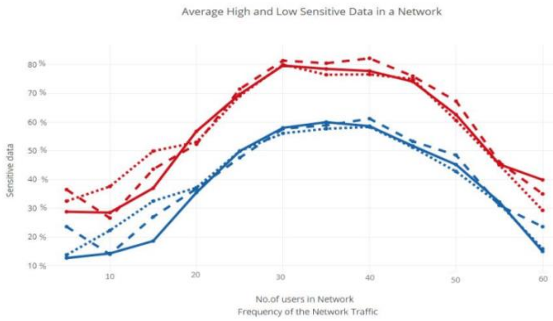


Fig. 7: Complete Network Traffic.

From above figure-7 we see the plotted lines it's related to the sensitive data in a network, the lines which are drawn in graph its shows the average low and high of the sensitive data in the network of the organization. Here on the X-axis is number of systems that are connected in the network for sharing that data from sever or one to another, on the Y-axis its network traffic and percentage of the sensitive data in the packet that was detected by an algorithm which is above 50% is the highly sensitive data and none of the packets has sensitivity value greater than 50% is low sensitive data in this algorithm performs as expected on plaintext files only. Based on the data sensitive we can provide the security in that storage and in that network also for the organization.

5. Conclusion

This article discusses Big Data vulnerabilities, the cloud computing "Information security" and "sensitive information", as to resolve SI issue by using the MAC-IP binding technique with an encryption application for protecting leaked data. In this article, an ideology is proposed for identifying the guilt agent(s) in a particular organization together with a mechanism for protecting the leakage of data from within the organizations. The so proposed method helps in providing security to the individual's data during ether sharing stage or transmission stage, together with the capability of identifying whether the data is leaked or not, if yes, the source of leakage can also be highlighted. Thus, providing data security using encryption algorithms ensures security, since this methodology is also bundled with the leakage detection technique, it will be of vital use in various organizations, where data is to be distributed via any of the public or private channel and shared with the third party. This method, once implemented within an organization or industries ensures a broad spectrum of security and detection.

References

[1] M. Alazab and R. Broadhurst Spam and criminal activity, Trends & issues in crime and criminal justice no. 526, Australian Institute of Criminology, 2016.

[2] H.M.J Almohri, L.T. Watson, D.F. Yao and X.M Ou, Security optimization of dynamic networks with probabilistic graph modeling and linear programming, IEEE Transactions on Dependable and Secure Computing, IEEE, Vol. 13, No. 4, 2016, 474-487.

[3] A. Carnielli, M. Aiash, M. Alazab, and others, on preserving privacy in cloud computing using ToR, London; San Diego; Cambridge, USA: Elsevier, 2016.

[4] S. Pearson, "Privacy, security and trust in cloud computing," in Privacy and Security for Cloud Computing (S. Pearson and G. Yee, eds.), Computer Communications and Networks, pp. 3–42, Springer London, 2013.

[5] National Institute Standards and Technology (NIST) (2013), Glossary of Key Information Security Terms, NISTIR 7298 Revision 2, NIST, Gaithersburg, MD.

[6] "Sensitive Data Classification and Protection" Overcoming the Challenges to Classify and Protect Sensitive Data at Federal Government Agencies.

[7] https://www.gpo.gov/fdsys/pkg/USCODE-2010-title5/pdf/USCODE-2010-title5-partI-chap5-subchapII-sec552a.pdf

[8] B. Raja Koti, Dr. G.V.S. Raj Kumar, Dr. Y. Srinivas, "A Comprehensive Study and Comparison of Various Methods on Data Leakages", International Journal of Advanced Research in Computer Science, Volume 8, No.7, July – August 2017, pp-627-631.

[9] Panagiotis Papadimitriou, "Data Leakage Detection", IEEE Transactions on Knowledge and Data Engineering, Vol. 23.

[10] P. Papadimitriou and H. Garcia-Molina, "Data leakage detection", Technical report, Stanford University, 2008.

[11] Blakley, B., Mcdermott, E. and Geer, D. (2001), Information Security is Information Risk Management, Proceedings of the 2001 Workshop on New Security Paradigms, ACM, Cloudcroft, New Mexico, pp. 97-104.

[12] Gordon, L.A. and Loeb, M.P. (2002), “The economics of information security investment”, ACM Transactions on Information and System Security (TISSEC), Vol. 5 No. 4, pp. 438-457.

[13] Anderson, R. (2001), “Why information security is hard-an economic perspective”, Computer Security Applications Conference, ACSAC 2001. Proceedings 17th Annual 2001, IEEE, pp.358-365.

[14] Alberts, C.J. and Dorofee, A. (2002), Managing Information Security Risks: The OCTAVE Approach, Addison-Wesley Longman Publishing Co. Inc., Boston, MA.

[15] Alberts, C., Dorofee, A., Stevens, J. and Woody, C. (2003), Introduction to the OCTAVE Approach, Carnegie Mellon University, Pittsburgh, PA.

[16] Gordon, L., Loeb, M. and Lucyshyn, W. (2003), “Information security expenditures and real options: a wait-and-see approach”, Computer Security Journal, Vol. 19 No. 2, pp. 1-7.

[17] Siponen, M.T. and Oinas-Kukkonen, H. (2007), “A review of information security issues and respective research contributions”, ACM Sigmis Database, Vol. 38 No. 1, pp. 60-80.

[18] Chen, Y. (2005), “Information valuation for information lifecycle management”, Autonomic Computing, ICAC 2005. Second International Conference on 2005, IEEE, pp.135-146.

[19] Sokolowski, J.A. and Banks, C.M. (2012), Handbook of Real-World Applications in Modeling and Simulation, John Wiley & Sons, Hoboken, NJ.

[20] Symantec. (2015). Symantec Data Loss Prevention. [Online]. Available: http://www.symantec.com/data-loss-prevention.

[21] Identity Finder. [Online]. Available: http://www.identityfinder.com

[22] Global Velocity Inc. (2015). Cloud Data Security from the Inside Out—Global Velocity. [Online]. Available: http://www.globalvelocity.com

[23] GTB Technologies Inc. (2015). GoCloudDLP. [Online]. Available: http://www.goclouddlp.com.

[24] B Raja Koti, GVS Raj Kumar, Y Srinivas, “Identification of Guilt Agent and Leaked Data by Using MAC-IP”, International Journal of Applied Engineering Research, 2017, Volume 12, Issue 22, pp 12237-12245.

[25] B. Hauer, "Data and Information Leakage Prevention within the Scope of Information Security," in IEEE, vol. 3, pp. 2554-2565, 2015.

[26] Jaymala Chavan and Priyanka Desai “Data Leakage Detection Using Data Allocation Strategies” International Journal of Advance in Engineering and Technology (IJAET), Volume 6 issue 6, Nov 2013.

[27] B. Hauer, "Data and Information Leakage Prevention within the Scope of Information Security," in IEEE Access, vol. 3, no. , pp. 2554-2565, 2015.

[28] Sandip A. Kale C, Prof.S.V. Kulkarni C, “Data Leakage Detection: A Survey”, IOSR Journal of Computer Engineering ISSN: 2278-0661 Volume 1, Issue 6 (July-Aug 2012), PP 32-35.

[29] Q. B. Hani and J. P. Dichter, "Data leakage presentation using homomorphic encryption in cloud computing," 2016 IEEE Long Island Systems, Applications and Technology Conference, Farmingdale, NY, 2016, pp. 1-5.

[30] DivyaChaube, Sonali Gandhi, Priyanka Gupta, “Implementation of Guilt Model and Allocation Strategy for Data Leakage Detection”, International Journal of Scientific and Research, Volume 5, Issue 4, April 2015, ISSN 2250-3153.

[31] S. Peneti and B. P. Rani, "Data leakage prevention system with time stamp," 2016 International Conference on Information Communication and Embedded Systems , Chennai, 2016, pp 1-4.

[32] C. Suresh Kumar, K. Iyakutti, Semantic Cluster-Based Classification for Data Leakage Detection for the Cloud Security, International Journal of Computer Applications, ISSN: 0975-8887, Vol 110 – No 6, January 2015.

[33] National Institute of Standards and Technology FIPS 197 Advanced Encryption Standard (AES) Published: November 2001.

[34] R. Tahboub and Y. Saleh, "Data Leakage/Loss Prevention Systems (DLP)," 2014 World Congress on Computer Applications and Information Systems (WCCAIS), Hammamet, 2014, pp. 1-6.

[35] S. Liu and R. Kuhn, "Data Loss Prevention," in IT Professional, vol. 12, no. 2, March-April 2010, pp. 10-13.

[36] G. Lawton, "New Technology Prevents Data Leakage," in Computer, vol. 41, no. 9, Sept. 2008, pp. 14-17.

[37] D. Kolevski and K. Michael, "Cloud computing data breaches a socio-technical review of literature," 2015 International Conference on Green Computing and Internet of Things (ICGCIoT), Noida, 2015, pp. 1486-1495.