Data Leaker Detection Using Data Lineage Method

Upasana Shekhar[1], M. Srinithya[2], Ch. Sai Navya[3], Mrs.P.Mohamed Fathimal[4]

[1, 2, 3] *Undergraduate Students, Department of Computer Science and Engineering, SRM University, Tamil Nadu, India*

[4]*Assistant Professor, Department of Computer Science and Engineering, SRM University, Tamil Nadu, India*

Abstract - *Data leakage is not a new topic as it has its roots in before modern IT technologies but with emerging new technologies creates new ways for data leaking such Instant Messaging, VoIP, Social networks like Twitter, Facebook and etc. There has been an issue of sending any third party or someone else a private data. One may never know when it could be leaked without the permission of the owner and be misused. It could be personal photos, documents and many other sensitive data. Thus if data are leaked, our proposed system will try to find which organization have leaked it. An enhanced data lineage method is proposed to track down by whom and to whom the data has been passed to, thus enabling to find the leaker. The system uses steganography, encryption and digital signature concept which will help in creating the data lineage to find out the leaker. The key features of this proposed system are access control, notification system and bulk outsourcing system.*

Key Words: **Encryption, Steganography, Digital signature, leaker, lineage**

1. INTRODUCTION

The unauthorized transfer of data or information from anyone’s personal computer or social network sites or companies to another person without their permission is known as data leakage. There has been many cases of data leakages in today’s time where has been a loss to a lot of companies and individuals.

According to the Global Data Leakage Report H1 2016, InfoWatch Analytical Center registered 840 confidential data leaks, which is 16% more than in H1 2015. Among the data leaks logged, 506 (67%) are caused by internal offenders, while 250 (33%) of the cases are triggered by intruders from the outside.

In the scenario of Social networking, it was reported that third party applications widely uses online social network. Nearly all of the most popular apps on Facebook—including Farmville, Causes, and Quiz Planet—have been sharing users' information with advertising and tracking companies.

The problem is that the Facebook apps are—possibly inadvertently—revealing the addresses of users' Facebook pages, which contain the users' unique Facebook IDs, and in some cases, their names. The information being transmitted is one of Facebook's basic building blocks- the unique "Facebook ID" number assigned to every user on the site. Since a Facebook user ID is a public part of any Facebook profile, anyone can use an ID number to look up a person's name, using a standard Web browser, even if that person has set all of his or her Facebook information to be private. For other users, the Facebook ID reveals information they have set to share with "everyone," including age, residence, occupation and photos. Facebook ID numbers are sent to at least 25 advertising and data firms, several of which build profiles of Internet users by tracking their online activities.

It is possible to determine that several applications were leaking data by analyzing their behavior and so these applications could be disabled by Facebook. However, it is not possible to make a particular application responsible for leakages that already happened, as many different applications had access to the private data

Thus it is important to build a system where a method to be able to track to whom the data is passed along should be implemented in the design phase so that the culprit can be found out in later stages. Creating a data lineage can be a solution. Data lineage helps us to follow the data from the source of the data, intermediate flow of data i.e., hops till the final destination of the data.

2. RELATED WORKS

There are many related works which has dealt with the concept of finding the data leaker. Few are mentioned below:

In [1] they use the data provenance scheme where a system is used to enforce the logging of read and write actions in a tamper-proof provenance chain. This creates the possibility of verifying the origin of information in a document. However, the disadvantage is that the attacker can strip of the attribution information of a file, and use it and distribute it as they wish without any traces left it.

In [2] they raise the question of an insider attack. Usually methods like watermarking or fingerprinting are only applied after completion process of generating the documents, so the people involved in the generation process have access to the original document and could possibly publish it without permission of the authors, or also leak the document without being tracked. So they have proposed several possible architectures for watermarking with dishonest insiders, in which insiders have access only to a watermarked version of the object that they are working on. Hence the insider would be treated like an outsider.

In [3] the model introduced here is the addition of fake objects which are different from real objects in the documents or image. It can be a fake email id but that would require the creation of that email id since the guilty agent should not be suspicious of it. Since creating and monitoring e-mail accounts consumes resources, the distributor may have a limit of fake objects.

In [4] a pseudo random selection of two patches is carried out where the first one is A and the second is B. Patch A image data is brightened where as that of patch B is darkened. Then this changed image is sent to the insider of the organization, just like when dealing with outsiders. The major drawback is the capacity to hold data in the image.

In [5] they add a predetermined small luminance value to randomly selected image pixels. Then this changed image is sent to the insider of the organization, just like when dealing with outsiders. It has the lowest capacity to hold data.

3. EXISITING SYSTEM

In [6] the authors have designed a LIME framework where if any document say an image is needed by another consumer then a request is sent to the provider. The provider then watermarks the image using cox algorithm with the AES encrypted information along with CMA-secure signature of the consumer id and sends it to the consumer. If the provider finds their image in a malicious environment they contact the auditor who extracts and decrypts the embedded information in the image, which will lead him to the consumer. Repeating this process with each consumer, the auditor creates a lineage. The last consumer in the lineage is the leaker. The drawback is that they have no access control mechanism in their method and no notification system which would inform the owner that it has a request waiting.

4. PROPOSED SYSTEM

The proposed system implements an enhanced version of the model mentioned in [6]. When a consumer, let's say an advertising agency requests an image from the social network site which is the provider, the notification will sent to the provider through the SMS and the provider will accept the request. The provider will apply restriction according to which the consumer can forward the image. Then a digital signature is produced using HMAC on the information triple containing provider name, consumer name and filename which are encrypted using RSA encryption. Finally the information is embedded in the image using LSB. The image is then finally sent to the consumer. The auditor is the one who creates the data lineage to find the data leaker. The modules in our system are as follows:

- Key generation for organizations and auditor
- Encrypting and signing information
- Embedding encrypted and signed information in requested image
- Access control
- Data lineage method

5. MODULE DESCRIPTION

5.1 Key generation for organizations and auditor

The public and private key pairs are generated to be used in RSA encryption.

- **Generate the RSA modulus (n)**
  - Select two large primes, p and q.
  - Calculate n=p*q. For strong unbreakable encryption, let n be a large number, typically a minimum of 512 bits.
- **Find Derived Number (e)**
  - Number **e** must be greater than 1 and less than (p − 1)(q − 1).
  - There must be no common factor for e and (p − 1)(q − 1) except for 1. In other words two numbers e and (p − 1)(q − 1) are coprime.
- **Form the public key**
  - The pair of numbers (n, e) form the RSA public key and is made public.
  - Interestingly, though n is part of the public key, difficulty in factorizing a large prime number ensures that attacker cannot find in finite time the two primes (p & q) used to obtain n. This is strength of RSA.
- **Generate the private key**
  - Private Key d is calculated from p, q, and e. For given n and e, there is unique number d.
  - Number d is the inverse of e modulo (p - 1)(q – 1). This means that d is the number less than (p - 1)(q - 1) such that when multiplied by e, it is equal to 1 modulo (p - 1)(q - 1).

- This relationship is written mathematically as follows –

  e*d=1 mod (p-1)(q-1)

5.2 Encrypting and signing information

The following information needs to be encrypted using RSA: Provider name, consumer name and filename. The process for encryption is as follows:

- Encrypt using owners private key PR_o, to provide non repudiation.
- Encrypt again with the auditor's public key PU_{aud}, for confidentiality because we do not want the consumer to decrypt the data only the auditor.

$$E\ (E\ (PR_o,\ m),\ PU_{aud)}$$

The public key can be requested via a reliable, but not necessarily secret route.

Then using HMAC algorithm, the encrypted information is signed to ensure integrity.

5.3 Embedding encrypted and signed information in the Requested Image

LSB method is used to embed the encrypted information in the requested image.

Algorithm for embedding:

- Read the cover image and the encrypted signed information.
- Convert the secret message to binary or bit stream.
- Calculate the LSB operation on cover image to replace
- Result is the stego image.

Each time for new information is to be embedded the bit is incremented and next bit is used.

5.4 Access control

A restriction module where the provider accepts the request of the consumer then a restriction is set according to which the consumer can forward the image by the provider. Example, if restriction given by the consumer is 0 then the consumer cannot forward it to anyone else but can do so maliciously

5.5 Data lineage method

The provider gets a notification that his image is being used in some other websites using the online tool PlagHunter. This tool requires the user to upload the images first and then it will search for similar images on other websites. The provider would send a complaint to the auditor. The auditor then performs the following operations:

- The extraction process from the LSB of the image
  - Read the stego image.
  - Calculate LSB of each pixels of stego image.
  - Retrieve bits and extract the encrypted and signed information.
- Checks the HMAC to ensure integrity was maintained,
- Decrypts the information extracted.
  - Decrypt using auditor's private key $PR_{aud,}$ and then decrypt using public key of owner $PU_{o.,}$

$$D(D(PU_o,m)),\ PR_{aud}\ )$$

- From this extracted information the auditor will know to which consumer this image was sent. The auditor performs the same process for each disclosed consumer from the information. If one of the consumers has sent the image maliciously the auditor will be able to find it out

6. EXPERIMENT ANALYSIS

A website for LIME is created using MySql as database. The website requires each organization to register and upload images in their login. Whenever a consumer request is sent to the owner, the owner receives a SMS. The owner then accepts the request and sends it to the consumer using the scheme defined.

Whenever a lineage has to be proved, the key is generated for decrypting with HMAC and looked for the presence in the LSB part of the image; the auditor uses this extracted information as a lineage and can successfully find out who the malicious user is. The data can pass though multiple agents. Any agents can leak the data and using this method we can find the agent who leaked the data with high confidence.

The following figures show the image transferred each time using our scheme:

Fig -1: Original image

Fig -2: Image after frist transfer

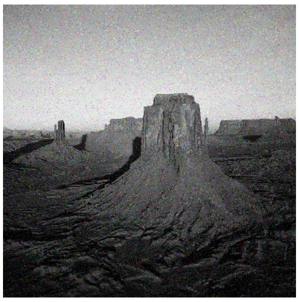

Fig -3: Image after second trasnfer

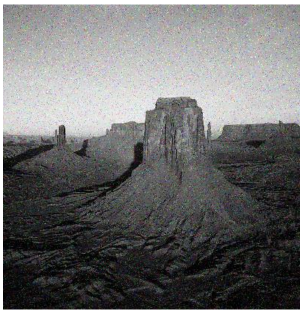

Fig -4: Image after third transfer

6. CONCLUSION AND FUTURE WORK

The proposed system can provide a way to find the leaker using steganography method by creating a lineage of the data. The scheme is simple and can work for environment with multiple levels of lineages. The advantage of this proposed method over the existing system is notification of request through SMS to the owner, restriction module for access control and bulk outsourcing to one user along with using asymmetric RSA algorithm and HMAC algorithm.

Possible future work would be to use new and improved steganography methods and to implement those new steganography methods in documents other than image.

REFERENCES

[1] R. Hasan, R. Sion, and M. Winslett, "The case of the fake picasso: Preventing history forgery with secure provenance", in FAST, 2009, pp. 1–14

[2] N. P. Sheppard, R. Safavi-Naini, and P. Ogunbona, , "Secure multimedia authoring with dishonest collaborators", EURASIP J. Appl. Signal Process., vol. 2004, pp. 2214–2223, 2004

[3]] P. Papadimitriou and H. Garcia-Molina, "Data leakage detection" , Knowledge and Data Engineering, IEEE Transactions on, vol. 23, no. 1, pp. 51–63, 2011. Imperial Journal of Interdisciplinary Research (IJIR) Vol-2, Issue-6, 2016 ISSN: 2454-1362, http://www.onlinejournal.in Imperial Journal of Interdisciplinary Research (IJIR) Page 1453

[4] W. Bender,D. Gruhl, N. Morimoto, A. Lu," Techniques for datahiding",IBM SYSTEMS JOURNAL, VOL 35, NOS 3&4, 1996

[5] Pitas, Ioannis. "A method for signature casting on digital images." *ICIP* (1996).

[6] Michael Backes, Niklas Grimm "Data Lineage in Malicious Environments" Dependable and Secure Computing, IEEE Transactions 2016 on Dependable and secure computing

[7] E.Divya, P.Rajkumar," Analysis of Data Hiding Algorithms", International Journal of Electrical, Computing Engineering and Communication

[8] Vol. 1, Issue. 2, April – 2015

[9] *C. A. V., D., L., V., N. P., and S., R. Ajai A., "Comparative Analysis of Image Steganography Using Lsb,Dct and DWT Techniques", International Conference on Signal and Speech Processing ICSSP '14. 2014.*

[10] Anupam, Jyoti Rani, "Analytical Comparison of DCT and LSB in Digital Watermarking", *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 3 Issue 9, September 2014*

[11] J. Domingo-Ferrer, "Anonymous fingerprinting based on committed oblivious transfer", in Public Key Cryptography. Springer, 1999, pp. 43–52

[12] S. Goldwasser, S. Micali, and R. L. Rivest, "A digital signature scheme secure against adaptive chosen-message attacks," SIAM J. Comput., vol. 17, no. 2, pp. 281–308, 1988.

[13] K. Brindha , Stiffy Sunny , K. Muralibabu,] "An novel architecture for information hiding using HMAC-MD5 L. Agilandeeswari" , International Journal of Engineering and Technology, 2 (2) (2013) 134-139

[14] Nawar S. Alseelawi1 , Tarik Z. Ismaiel2 , Firas A. Sabir, "High Capacity Steganography Method Based upon RGBA Image", International Journal of Advanced Research in Computer and Communication Engineering Vol. 4, Issue 6, June 2015

[15] K.Thangadurai and G.Sudha Devi, PG and Research Department of Computer Science, "An analysis of LSB Based Image Steganography Techniques", 2014 International Conference on Computer Communication and Informatics (ICCCI -2014), Jan. 03 – 05, 2014, Coimbatore, INDIA

[16] https://infowatch.com/report2016_half

[17] http://indiatoday.intoday.in/story/delhi-cyber-crooks-stealing-morhphong-photos-porn-sites-extrotion/1/907956.html

[18] http://www.plaghunter.com/

[19] http://gawker.com/5666325/how-to-stop-facebook-from-sharing-your-information-with-third-parties