DATA PROVENANCE: A DATA LEAKAGE DETECTION MODEL

Neha Belekar [1]

[1] PG Student,Department of Computer Engineering, MET's IOE, Maharashtra, India

ABSTRACT

In today's era, information leakage is one of the most serious threats to companies. A data owner sends secret or confidential information to a group of trusted data consumers. Some of the information is lost and found in an inappropriate place. Thus data has been leaked. Data leakage means data distributed by the data owner is leaked by one or more agents. This causes a huge harm to the business. The distributor must assess whether data is leaked from one or more agents. To enhance the probability of detecting data loss, data allocation strategies (across the agents) are used. A data lineage framework is used for identifying a guilty entity. The digital watermarking is a technique in which vital information is kept hidden in the original data for protecting unauthorized copying and circulation of data. An accountable data transfer protocol can be built using transfer method, watermarking, and signature primitives. In some occasions fake data records can be injected in order to improve detecting data loss and identifying the guilty entity. The data sent by the data owner must be protected, secret and it must not be regenerated. The framework of data lineage is considered for transmission of data and is a key step towards achieving accountability.

Keywords *—Information leakage, data provenance, accountability, watermarking, distributor, agent, lineage framework, embed watermark, guilty attacker finding, accountable data transfer protocol.*

1. INTRODUCTION

In the course of technology and doing business, at many occasions sensitive important data is handed over to trusted third parties. For example, an organization may have associations with other organizations that share customer data. Another business enterprise may outsource its work to other companies where they require to send data to an external company. The entity who owns data is called as the data owner or distributor and the third parties are called as the agents or data consumer.

The aim is to detect leakage of sensitive data and identification of the guilty agent. In a very short time, large volumes of digital sensitive data can be copied by the attacker and can be spread across the internet. The chances of getting caught for data provenance is very less as currently there is no accountability method. The data leakage problem has reached new horizon recently. It is a concern for not only organizations, but individuals are also affected by data provenance. The situation get worse because of the advent in the technology of mobile phones and social networking. In today's environment, individuals usually expose their private data to various providers of service or advertising companies, in return for some services which are free of cost. In the nonappearance of right regulations and accountability methods, various online applications distribute entities data, through which entities can be easily identified by couple of advertising and online tracking companies. Even if there is restricted access to confidential data with the presence of access control methods, an intruder can post confidential data over the internet. Information security mechanisms such as encryption provide protection as long as the data is secured by encrypting it, but once the consumer decodes a message by decrypting it, nobody can prevent the disclosure of decrypted content. Therefore it is impractical to avoid data leakage fanatically.

Data provenance is the enormous threat in front of the companies and various different enterprises. Though there are number of different encryption mechanisms designed for securing information, there is a challenging problem of the integrity of the users of the systems. In order to offer security against data loss threat technologies like machine learning content/context based detectors, encryption, access control, firewalls and identity management have already been incorporated. The information distributed is considered as sensitive data when it consists of information about the client, budget, code and any design specification. The agents who get their hands on the sensitive data are also known as cyber criminals. Data leakage is done for their own prots which results in loss of the company. This problem can be overcome by a general method of data transmission is used. This mechanism is referred as accountability.

This accountability method can be directly correlated with detection of data transfer history across multiple nodes right from its origin. The distributor sends the data to the agent using strategies that increase the possibility of finding the agent by adding fake data to the information distributed. If any person receiving the data leaks the data then the distributor will find the agent by the help of number of fake objects released out and the distributor waits until he gets enough evidence and finally conform the agent and closes the business with him or takes any legal action on the agent.

2. RELATED WORK

G. Doerr et al. [1] digital watermarking recently elongated from still pictures to content of video. Further research in this area is strongly encouraged by an increasing requirement from the copyright data distributor in order to protect their rights assuredly. A watermark can be divided into two parts: one for copyright protection and the other for customer fingerprinting. Robustness has to be considered attentively. Applying watermarking to video is definitely a new area of research by making use of still images. The easier and simpler method is to consider a video as a series or chain of still images. An existing watermarking method can be reused for still images. A new robust video watermarking algorithm can be designed by exploiting additional temporal dimension. Another approach considers a specific video compression standard which can be used to compress a video stream.

M. A. Alsalami et al. [2] proposed the technique of digital audio watermarking for embedding data along with audio signal. Copyright owner uses embedded data for identification purpose. The main aim of watermarking systems is to insert a hidden robust watermark into digital media file. These systems have to appease two contrary needs. First, watermark must be unaffected from voluntary and involuntary removal. Second, watermarked signal should preserve a fair loyalty and robustness of watermarked signal.

V. M. Potdar et al. [3] as noted, a lot of research is being conducted currently in the field of watermarking. There is a lot of work begin conducted indifferent branches in this field. The changes are made to the image in such a way that only the data distributor and data agent can identify the message by using steganography. Modern printers use steganography. Intruders and intelligence services use steganography allegedly. The identity and authenticity of the owner of a digital image is verified using a technique called watermarking. Watermarking is a technique in which the identification information is inserted into the digital picture which reveals the owner of the digital image. Watermarks can also be embedded into videos or audios which are referred as signals. For example, popular artists watermark their paintings and portraits. The watermark is copied along with the image when somebody tries to copy the image. Watermarking is used for, source tracing, annotation of photographs and copyright protection.

A. Mascher-Kampfer et al. [4] discussed about the purpose of single watermarking schemes in a multiple re-watermarking scenarios. A surprisingly huge number of different watermarks may be detected and also robustness can be preserved up to a certain limit using this approach, however, detection association falls for an increasing number of embedded marks which restricts scalability to long selling series.

I. J. Cox et al. [8] defined spread spectrum scheme. This scheme uses injects noise into the watermark images. In order to provide robustness, the watermark is injected or embedded to the images most significant portion, so it would be impossible to destroy the image without removing the watermark.

R. Halder et al. [5] improved digital watermarking for relational databases. It arrived as an answer in providing protection, intrusion detection, culpable tracing, and maintaining integrity of relational data. The current modern approaches for relational databases are fingerprinting and watermarking. All the approaches are categorized on the basis of (i) Check if the underlying data is distorted (ii) Check where water mark is embedded and the cover type, and (iii) The watermark information type. For safeguarding the ownership distortion-based watermarking methods are practiced. And for maintaining integrity of the database distortion-free watermarking methods are adopted.

A. Adelsbach et al. [9] noted that for proving the security of multimedia applications, it combines cryptographic methods with watermarking and the formal definition of security properties or attributes is important.

M. Naor et al. [10] designed the oblivious transfer protocol using the PBC library [11]. The data distributor encrypts the message, when the transfer protocol is used to send the messages, sends both ciphertexts to the receiver and performs transfer just on the decryption keys. The messages are actually of arbitrary size, but the protocol can be used with a fixed message size.

Papadimitriou et al. [6] initiated the study of data leakage in which they defined that data owner has given confidential data to a group of data consumers (third parties). In some cases the data is lost and found in an inappropriate place (e.g., on the internet or somebody's mobile or computer). In order to increase the probability of identifying data leakage certain approach of allocating data (across the agents) is used. In some situations addition of fake data records will further enhance the likelihood of identifying leakage and detecting the culpable entity.

Michael Backes et al. [7] In this work, a LIME (Lineage In Malicious Environments) framework is implemented having two roles data distributor and agent. The data lineage approach is designed to identify accusable agent, and detect the non-repudiation and assumptions (honesty). A data transfer protocol is used between distributor and agent within a malicious environment. Transfer protocol, watermarking, and digital signature primitives mechanisms are developed and analysed.

[16]LSB (Least Significant Bit) substitution is the process of adjusting the least significant bit pixels of the carrier image. In this technique, in the image a message is embedded. The number of bits in an image decides the insertion of least significant bit. For an 8 bit image, the least significant bit i.e., the 8th bit of each byte of the image is changed to the bit of secret message. The most widely used and popular and algorithm for symmetric encryption is the Advanced Encryption Standard (AES).

3. PROPOSED WORK

The three vital parts or roles that can be allocated to the involved parties in lineage framework: data owner (distributor), data consumer (agent) and auditor (controller).

Data Owner: The distributor manages the files, documents and the agent is the recipient of the documents and can achieve some task or work by making use of the files or documents received.

Data Consumer: Data Consumer is the one which receives the document. The case of an untrusted data distributor can be examined where an agent can send a file to another agent. Every agent can expose new information which is inserted to the controller to indicate the further agent and to justify his own guiltlessness.

Auditor: which is not involved in the sending of documents or files, it is only call forth or requested when an exposure happens and then executes all actions that are needed to identify the accused entity.

Below given Fig. 1 shows the framework for tracing data provenance.

The procedure to find the accusable entity proceeds in the following manner:

1. Firstly the data owner will select an input file and the agent to whom the file will be sent.
2. Then the watermark will be embedded by collecting the original file and name of the recipient.
3. Fake objects are added and the watermark signature is placed into the file by applying accountable transfer protocols.
4. Further, in order to decode the file, it will be collected, loaded, analysed and decoded.
5. Guilty attacker is founded by tracing the data lineage and leakage.
6. Finally, auditor will detect and identify the guilty party involved in the data transmission.

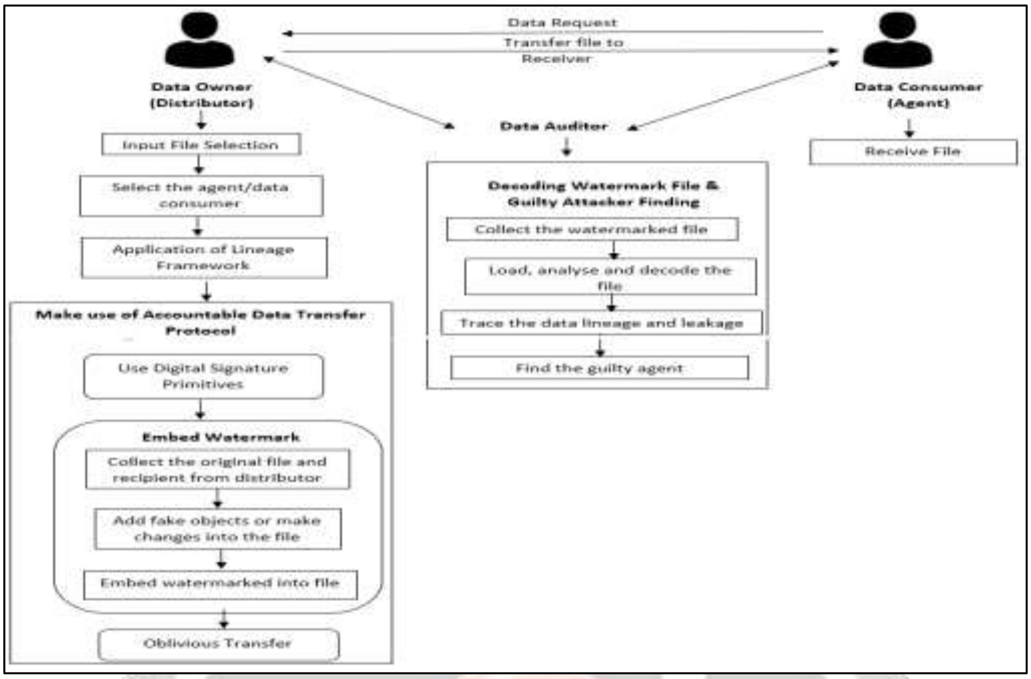
Data Request
Transfer file to Receiver
Data Owner (Distributor)
Data Consumer (Agent)
Data Auditor
Input File Selection
Select the agent/data consumer
Application of Lineage Framework
Make use of Accountable Data Transfer Protocol
Use Digital Signature Primitives
Embed Watermark
Collect the original file and recipient from distributor
Add fake objects or make changes into the file
Embed watermarked into file
Oblivious Transfer
Decoding Watermark File & Guilty Attacker Finding
Collect the watermarked file
Load, analyse and decode the file
Trace the data lineage and leakage
Find the guilty agent
Receive File

Fig. 1. Framework for tracing data provenance.

3.1 Algorithms:

1. Algorithm for Embedding Watermark

The original file is selected and loaded. The file contents are analysed. After finding the free position of pixels watermark is stored into the file by embedding watermark and signature. Various terms used in the given algorithm are C is the input image, Comp () function is to compute Cs, Um indicates original sender, Us represents secret image, Im denotes total no. of recipients, Stg is the segment and NBTH () represents number of bits to hide. IEncry () function is used to encrypt the Cs. The CDSIS () function is C Distance function which will check whether secret key can be applied to the pixel. And SS in embedding function EM () represent the total number of segments.

Following algorithm is used for embedding watermark into the file.

1: Input image files are: Secret image(S), Cover image(C) and Cipher Key (K)

2: Produce Cs by application of Comp(S, Um, Us, Im) function;

3: Produce CEs by applying IEncry (Cs, lCs) function;

4: For each color in C

4.1: Application of CDSIS(C, K) function to generate non uniform segment Stg by calculating XS, R and YS, R;

4.2: Call ByteCharacteristic () to find NBTH;

4.3: Execute function EM(C, SS, CES) for embedding watermark to the secret image which is compressed and encryption (CES) is performed;

5: The segment image Stg is transmitted over a transmission path which is insecure;

6: End

2. Algorithm for Guilty Attacker Finding

Auditor collects the leaked watermark file. Loads and analyses the file. Extracts the watermark position and checks whether data has changed. If data has changed then leakage has occurred according to the lineage trace.

Thus after decoding the file and checking the lineage trace guilty agent is found. Various terms used in the given algorithm are Stg is the segment and NBTH () represents number of bits to hide. IDcry() function is used to decrypt the Cs. The CDSIS () function is C Distance function which will check whether secret key can be applied to the pixel. And the extract function EX () is used.

Following algorithm is used for finding the guilty attacker or entity.

1: The input is Stg image and the cipher key K that are received from the insecure transmission path;

2: For each color in Stg;

2.1: The edges of each segment are found by applying CDSIS (Stg, K). SegS By calculating XS, R and YS, R this is done for each segment;

2.2: The NBTH is computed by scanning all bytes in image Stg;

2.3: For producing CES, EX function EX (Stg, SS) is applied for all bytes depending on characteristics of bytes;

3: Apply IDcry (CES ×lCS) function to produce CS;

4: Apply DEC (CS) function to find a secret image S;

5: The leakage is detected and guilty person is identified by analysing the secret image S and lineage trace.

6: End

3.2 Mathematical Model

The mathematical model for data provenance system S can be presented in terms of input, output and functions.

S = {I, O, F} where,

I: Input {C, K, S}

Where, C: Cover Image

K: Cipher Key

S: Secret Image

O: Output {G}

Where, G: Guilty Entity Identified

F: Set of Functions: {F1, F2, F3, F4, F5}

Where,

F1: Embed Watermark

F2: Transfer File

F3: Tracing Data Lineage

F4: Leakage Detection

F5: Guilty Attacker Finding

4. EXPERIMENTAL SETUP

Setup: The experiment is run on Operating System Windows 7 with i3 processor, speed is 2.4 GHz and RAM is 1GB. Dot Net programming language is used for implementation. C sharp is used for front end and SQL Server for backend, Programming Editor used is MS Visual Studio 2010, Dot Net Framework 4.0. The experiment has been executed for sender and receiver on the same program as well as in network environment.

Dataset: We use our own dataset collection for this experiments. In this dataset, three types of file format are used i.e., image file, audio file and video file. These files are collected using their file size. For image the maximum file size allowed is 4 MB. But for audio and video files, the file size should be between 1MB to 4MB in size. The experiment is executed for 40 image files that is 10 image files of size 65 KB,10 image files of 257 KB,10 image files of 1.1 MB and 10 image files of 3.1MB size. The experiment is executed for 30 audio files that is 10 audio files of 1 MB, 10 audio files of 2 MB and 10 audio files of 3 MB size. The experiment is executed for 30 video files that is 10 video files of 1 MB, 10 video files of 2 MB and 10 video files of 3 MB size.

5. PERFORMANCE RESULT AND ANALYSIS

5.1 Results for Image File.

The resulting image files after being transferred multiple times using the data provenance model is shown in Fig 2.

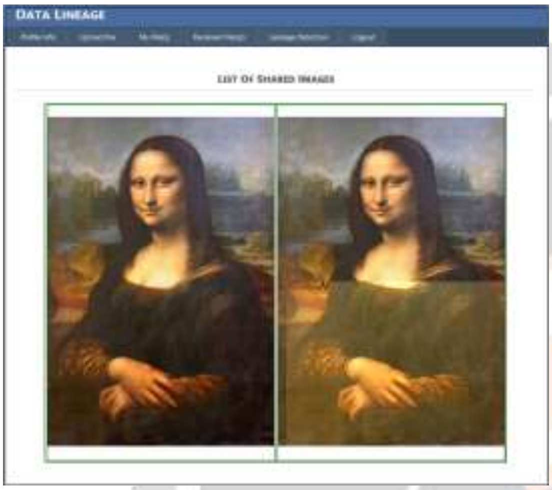

Fig. 2 (A) Image is transferred once

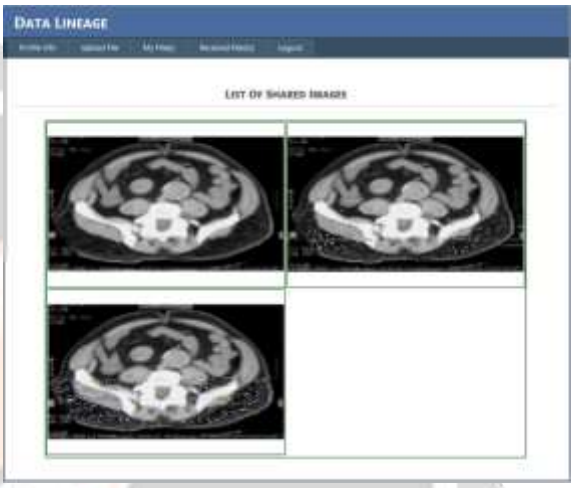

Fig. 2 (B) Image is transferred twice

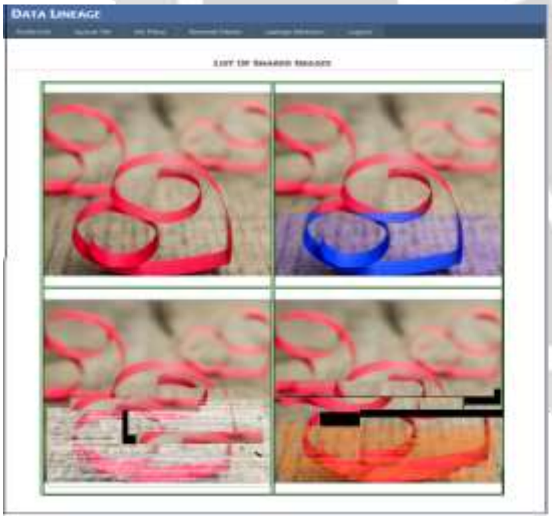

Fig. 2 (C) Image is transferred thrice

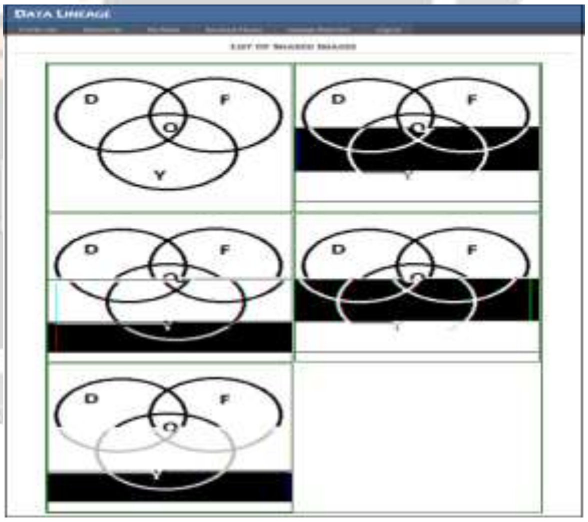

Fig. 2 (D) Image is transferred four times

Fig 2 shows the results of iterative experiments with embedded watermarks. A noticable distortion to the original file can be observed with these embedded watermarks . Fig. 2(A) shows the image transferred once using the data provenance model. Fig. 2(B) shows the image transferred twice using the data provenance model. Fig. 2(C) shows the image transferred thrice using the data provenance model. Fig. 2(D) shows the image transferred four times using the data provenance model.

5.2 Results for Audio File

The size for audio file should be between 1 MB to 4 MB. It is observed that a significant increase in size of audio file can be seen after applying the Watermark. The proposed system can efficiently detect the guilty agent for audio file.

5.3 Results for Video File

The size for video file should be between 1 MB to 4 MB. The size for video file should be between 1 MB to 4 MB. It is observed that a significant increase in size of video file can be seen after applying the Watermark. The proposed system can efficiently detect the guilty agent for video file.

5.4 Comparison with Existing System

Attributes	Reference No. [6]	Reference No. [10]	Reference No. [16]	Reference No. [17]	Proposed System
Signature	No	No	No	Yes	Yes
Encryption	No	No	Yes	No	Yes
Watermark Embedding	No	No	Yes	No	Yes
Transfer Protocol	No	Yes	No	No	Yes
Watermark Decoding	No	No	Yes	No	Yes
Leakage Detection	Yes	No	No	No	Yes

Table 1 . Comparison with Existing System

5.5 Analysing the Computation Time Performance

This section examines the effectiveness of the proposed system. The experiment is executed with different parameters to analyse the performance. The execution time is measured for different phases such as:

1. Signature Creation.

2. Encryption.

3. Watermark Embed.

4. Watermark Detection.

The first experiment shows the graph where number of files is used and the computation time is in milliseconds (ms). It is observed that the computation time is linear in the number of files. Fig 3 (A) shows the computation times for Number of Files of Image File. Fig 3 (B) The computation times for Number of Files of Audio File. Fig 3 (C) The computation times for Number of Files of Video File. In graph, X-axis represents number of files and Y-axis represents time in milliseconds.

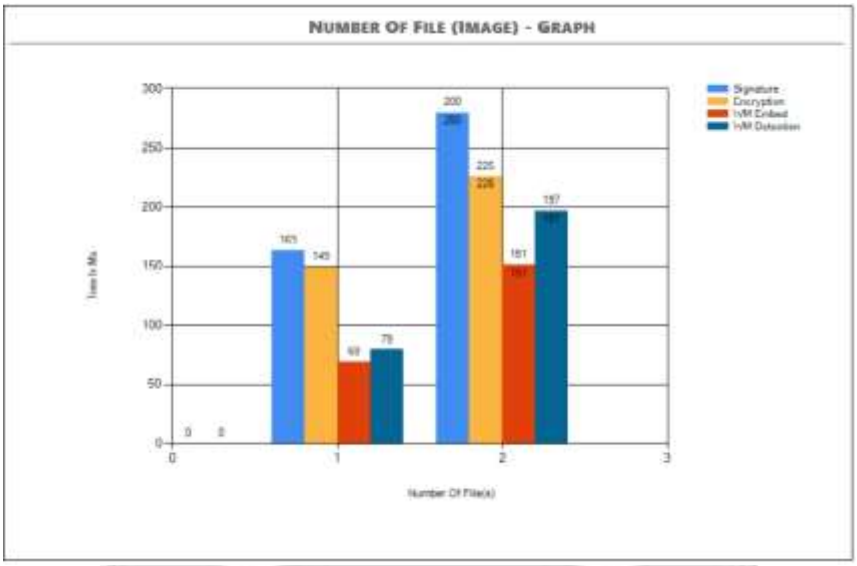

Fig 3 (A) The computation times for Number of Files of Image File.

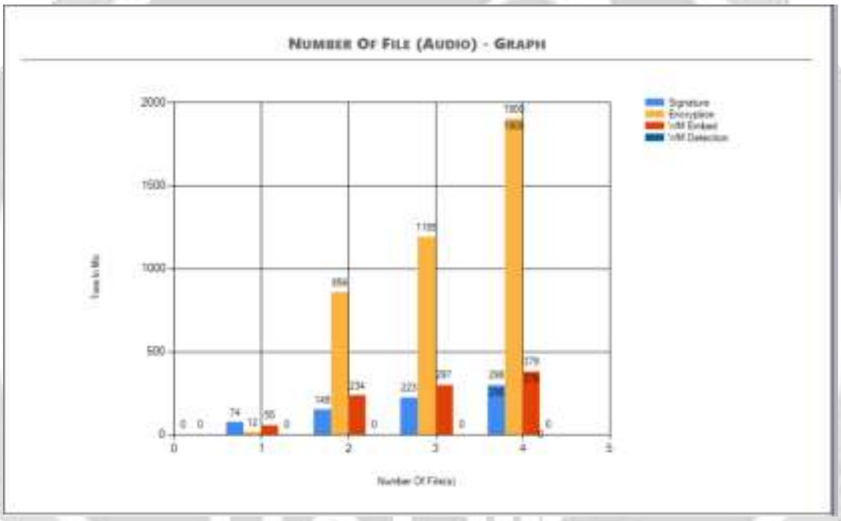

Fig 3 (B) The computation times for Number of Files of Audio File.

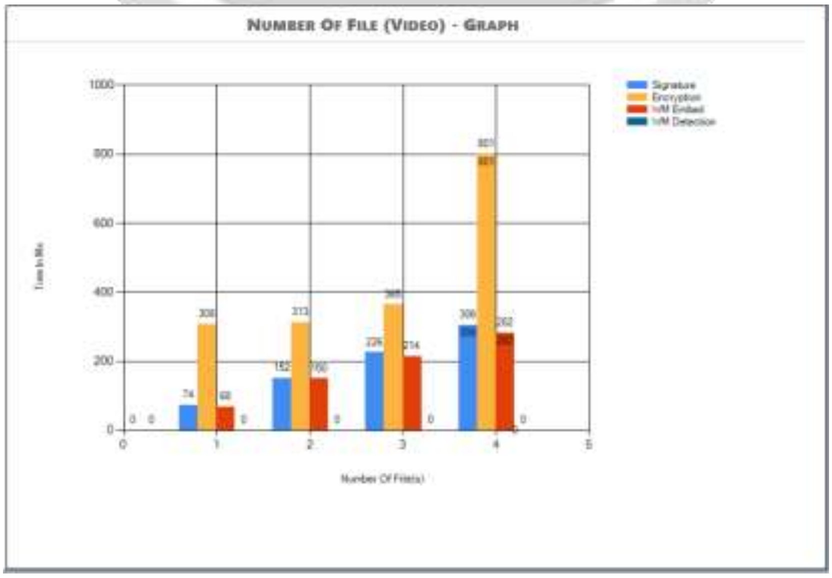

Fig 3 (C) The computation times for Number of Files of Video File.

The second experiment shows the graph where file size is considered. The file size is changing. The result can be seen in Fig 4 shows computation times for File Size. In graph, X-axis represents file size in KB (i.e. 1 indicates 100 KB) and Y-axis represents time in milliseconds.

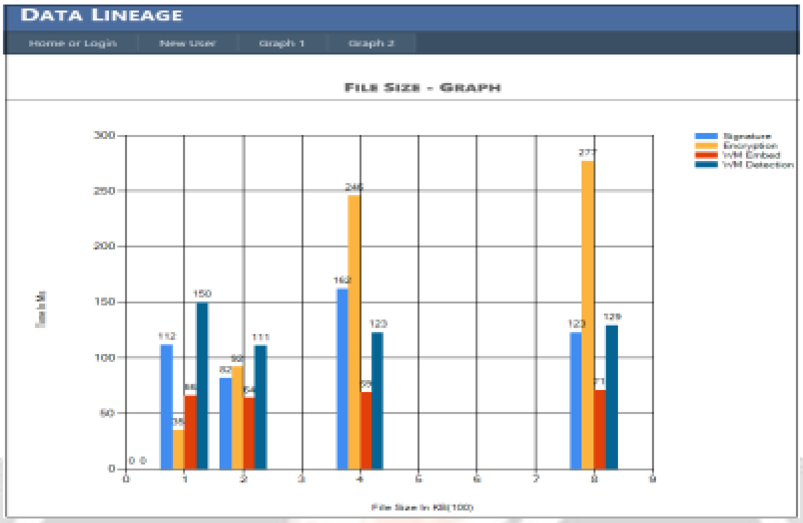

Fig. 4 The computation times for File Size.

The third experiment shows the Table 2 where various lineage chains along with file type and size are considered. The below Table 2 shows the computation time plotted for signature, encryption, watermark embedding and decoding in milliseconds (ms). In the below result there are users namely John, Jenny, Neha, Mary, Mike, Sam, Sui, Ben, Victor, Roy, Rita, Tom and Kim. These data users are part of the lineage chains. The table shows various scenarios for different file types and sizes. From the values plotted it is observed that for image file, as the file size and the number of participants in the lineage increases the computation time for encryption and watermark decoding increases linearly. The observation for audio file is that the computation time for encryption, watermark embedding and watermark decoding increases linearly, as the file size and the number of participants in the lineage is increased. The observation for video file is that the computation time for signature, encryption, watermark embedding and watermark decoding increases linearly, as the file size and the number of participants in the lineage is increased.

IJARIIE

Sr. No.	File Type Taken	File Size	Lineage Chain	Guilty Entity	Signature	Encryption	Watermark Embed	Watermark Decode
1	Image	65 KB	John- Neha – Kim	Neha	76	20	56	71
2	Audio	1 MB	Kim–Neha–Mary-Jenny	John	75	120	54	207
3	Video	1.56 MB	Sam-Tom-Sui-Ben-Kim & Victor	Kim	77	602	62	355
4	Image	257 KB	Victor-Neha-Ben-Roy & Tom ; Roy-Sam-Mike	Tom	75	17	55	212
5	Audio	2 MB	Roy-Tom-Sui-Ben-Mike & Neha;Mary-Sam-Victor	Sui	77	216	66	271
6	Video	2.1 MB	Neha-John-	Sam	75	850	96	398

			Mary-Sam-Ben-Victor-Kim-Roy					
				Kim	75	850	96	554
7	Image	1.1 MB	Mary-Sam-Ben-Sui-Tom-Victor-Rita-Kim-Neha	Sui	76	602	85	382
8	Audio	3 MB	Neha-Roy-Rita-Ben-Sam-Tom-Sui-Victor-Kim-Jenny	Tom	75	820	71	547
9	Video	3 MB	John-Jenny& Neha & Kim;John-Mary-Sam & Tom;Kim- Ben & Sui	Ben	157	940	90	288
10	Image	3.15 MB	Tom-Sui-Mike-Mary & Jenny & John &Rita	John	76	660	71	385

Table 2. Computation times plotted for various Lineage chains.

6. CONCLUSIONS

The chances that a data consumer is responsible for data loss is checked on the basis of overlay of his data with the exposed data and the data of other consumers, and based on the possibility that data items can be presumed by other modes. The lineage approach appliances a wide range of data circulation methodologies that can boost the owner's likelihood of finding leakage and diagnosing a data leaker. Thus the data provenance model can be made effective than the existing watermarking model. Data provenance model caters protection to data at the time of circulation or transmission of data and can also find if that gets leaked. Watermarking safeguards the data using techniques like encryption, whereas data provenance model provides prevention plus guilt identification. This model can prove to be advantageous to enterprise, where data is disbursed using any public or private medium and shared with the outsider (third party). Now, enterprise, numerous organizations can entrust or depend on this data provenance model.

7. FUTURE WORK

In this research attempt, the main focus is on detecting data leakages and identifying the guilty agent that leaked the data. However, there are various document types and formats for whom data leakage detection model is unavailable. Thus, this work motivates further research on data leakage detection techniques for various document types and scenarios such as relational database, Android apps, derived data, etc.

8. REFERENCES

[1] G. Doerr and J.-L. Dugelay, A guide tour of video watermarking, Signal Process.: Image Commun., vol. 18, no. 4, pp. 263 282, 2003.

[2] M. A. Alsalami and M. M. Al-Akaidi, Digital audio watermarking: Survey, School Eng. Technol., De Montfort Univ., U.K, 2003.

[3] V. M. Potdar, S. Han, and E. Chang, A survey of digital image watermarking techniques, in Proc. 3rd IEEE Int. Conf. Ind. Informat., pp. 709716, 2005.

[4] A. Mascher-Kampfer, H. Stogner, and A. Uhl, Multiple re-watermarking scenarios, in Proc. 13th Int. Conf. Syst., Signals, Image Process., pp. 5356,2006.

[5] R. Halder, S. Pal, and A. Cortesi, Watermarking techniques for relational databases: Survey, classification and comparison, J. Universal Comput. Sci., vol. 16, no. 21, pp. 31643190, 2010.

[6] P. Papadimitriou and H. Garcia-Molina, Data leakage detection, IEEE Trans. Knowl. Data Eng., vol. 23, no. 1, pp. 5163, Jan. 2011.

[7] Michael Backes, Niklas Grimm, and Aniket Kate Data Lineage in Malicious Environments, in IEEE Trans.Dependable and Secure Computing,vol. 13, no. 2,Apr. 2016.

[8] I. J. Cox, J. Kilian, F. T. Leighton, and T. Shamoon, Secure spread spectrum watermarking for multimedia, IEEE Trans. Image Process. , vol. 6, no. 12, pp. 16731687, Dec. 1997.

[9] A. Adelsbach, S. Katzenbeisser, and A.-R. Sadeghi, A computational model for watermark robustness, in Proc. 8th Int. Conf. Inf. Hiding , pp. 145160, 2007. 44

Tracing Data Provenance in Malicious Environments

[10] M. Naor and B. Pinkas, Efficient oblivious transfer protocols, in Proc. 12th Annu. ACM-SIAM Symp. Discrete Algorithms , pp. 448457, 2001.

[11] M. Backes, N. Grimm, and A. Kate, Lime: Data lineage in the malicious environment, in Proc. 10th Int. Workshop Security Trust Manage., pp. 183187, 2014. [12] R. Petrovic and B. Tehranchi, Watermarking in an encrypted domain, US Patent App. 11/482 ,519, Jul. 7, 2006.

[13] N. P. Sheppard, R. Safavi-Naini, and P. Ogunbona, On multiple watermarking, in Proc. Workshop Multimedia Security: New Challenges , pp. 36, 2001.

[14] A.-R. Sadeghi, The marriage of cryptography and watermarking Beneficial and challenging for secure watermarking and detection, in Proc. 6th Int. Workshop Digital Watermarking , pp. 218, 2008.

[15] Pairing based cryptography library (PBC) [Online]. Available: http://crypto.stanford.edu/pbc, 2014.

[16] W. Dai. Crypto++ Library [Online]. Available: http://cryptopp. com, 2013.

[17] D. Boneh, B. Lynn, and H. Shacham, Short signatures from the Weil pairing, in Proc. 7th Int. Conf. Theory Appl. Cryptol. Inf. Security: Adv. Cryptol., pp. 514–532, 2001.