ISSN: 0975-833X

Available online at http://www.journalcra.com

International Journal of Current Research
Vol. 3, Issue, 7, pp.160-164, July, 2011

INTERNATIONAL JOURNAL OF CURRENT RESEARCH

RESEARCH ARTICLE

WATER MARKING RELATIONAL DATABASES USING OPTIMIZATION - BASED TECHNIQUES

J. Mary Jenitha* and S.S. Dhenakaran

Department of Computer Science and Engineering, Alagappa University, Karaikudi-630003

ARTICLE INFO

Article History:
Received 21st April, 2011
Received in revised form
29th May, 2011
Accepted 5th June, 2011
Published online 16th July 2011

Key words:
Watermarking,
Optimization,
Genetic Algorithm,
Pattern Search.

ABSTRACT

This paper formulates the watermarking of relational database as a constrained optimization problem and to implement efficient techniques to handle the constraints like watermark synchronization errors and watermark detection. We present two techniques to solve the formulated optimization problem based on genetic algorithms (GAs) and pattern search (PS) techniques,and also a data partitioning technique that does not depend on marker tuples to locate the partitions and, thus, it is resilient to watermark synchronization errors.We develop an efficient technique for watermark detection that is based on an optimal threshold. The optimal threshold is selected by minimizing the probability of decoding error.

INTRODUCTION

The rapid growth of internet and related technologies has offered an unprecedented ability to access and redistribute digital contents. In such a context, enforcing data ownership is an important requirement which requires articulated solutions, encompassing technical, organizational and legal aspects. Though we are still far from such comprehensive solutions, in the last years watermarking techniques have emerged as an important building block which plays a crucial role in addressing the ownership problem. Such techniques allow the owner of the data to embed an imperceptible watermark into the data.

A watermark describes information that can be used to prove the ownership of data, such as the owner, origin, or recipient of the content. Secure embedding requires that the embedded watermark must not be easily tampered with, forged, or removed from the watermarked data. Imperceptible embedding means that the presence of the watermark is unnoticeable in the data. Furthermore, the watermark detection is blinded ,that is, it neither requires the knowledge of the original data nor the watermark. Watermarking techniques have been developed for video, images, audio, and text data , and also for software and natural language text. By contrast the problem of watermarking relational data has not been given appropriate attention. There are, however, many application contexts for which data represent an important asset, the ownership of which must thus be carefully enforced. This is the case, for example, of weather data, stock market data, power consumption, consumer behavior data, medical and scientific data. Watermark embedding for relational data is made possible by the fact that real data can very often tolerate a small amount of error without any significant degradation with respect to their usability. For example when dealing with weather data, changing some daily temperatures of 1 or 2 degrees is a modification that leaves the data still usable. Watermarking of relational databases is very important point for the researches; because the free databases available on the internet websites are published without copyrights protection and the future will exploding problems. If the DB contains very important numerical data; To add watermark in the numerical and relational database without affecting the usefulness and the quality of the data.

The goal is how to insert intended error /mark /data /formula/ evidence associated with secret key known only by the data owner in order to prove the ownership of the data without lossless of its quality, W is embedded into the relational database I with a secret key k, the watermarked relational database IW later pass through a distribution channel (computer network, internet, etc.), which are simulated under several kinds of common attacks. The watermarked DB after attack Iw, with the same secret key, will then extracted in order to recover the original watermark data W. To date only a few approaches to the problem of water marking relational data have been proposed . These techniques, however, are not very resilient to watermark attacks. In this paper, we present a watermarking technique for

Corresponding author: jenimphil@gmail.com

relational data that is highly resilient compared to these techniques. In particular, our proposed technique is resilient to tuple deletion, alteration, and insertion attacks.

II. RELATED WORKS

Agrawal et al. [7] proposed a watermarking algorithm that embeds the watermark bits in the least significant bits (LSB) of selected attributes of a selected subset of tuples. This technique does not provide a mechanism for multibit watermarks; instead only a secret key is used. For each tuple, a secure message authenticated code (MAC) is computed using the secret key and the tuple's primary key. The computed MAC is used to select candidate tuples, attributes and the LSB position in the selected attributes. Hiding bits in LSB is efficient. However, the watermark can be easily compromised by very trivial attacks. For example a simple manipulation of the data by shifting the LSB's one position easily leads to watermark loss without much damage to the data. Therefore the LSB-based data hiding technique is not resilient [5]. Moreover, it assumes that the LSB bits in any tuple can be altered without checking data constraints. Simple unconstrained LSB manipulations can easily generate undesirable results such as changing the age from 20 to 21. Li et al. [12] have presented a technique for fingerprinting relational data by extending Agrawal et al.'s watermarking scheme.

Sion *et al.* [11] proposed a watermarking technique that embeds watermark bits in the data statistics. The data partitioning technique used is based on the use of special marker tuples which makes it vulnerable to watermark synchronization errors resulting from tuple deletion and tuple insertion; thus such technique is not resilient to deletion and insertion attacks. Furthermore, Sion et al. recommend storing the marker tuples to enable the decoder to accurately reconstruct the underlying partitions; however this violates the blinded watermark detection property. The data manipulation technique used to change the data statistics does not systematically investigate the feasible region; instead a naive unstructured technique is used which does not make useof the feasible alterations that could be performed on the data without affecting its usability. Furthermore, Sion et al. proposed a threshold technique for bit decoding that is based on two thresholds. However, the thresholds are arbitrarily chosen without any optimality criteria. Thus the decoding algorithmexhibits errors resulting from the non-optimal threshold selection, even in the absence of an attacker.

Gross-Amblard [8] proposed a watermarking technique for XML documents and theoretically investigates links between query result preservation and acceptable watermarking alterations. Another interesting related research effort is to be found in [17] where the authors have proposed a fragile watermark technique to detect and localize alterations made to a database relation with categorical attributes. Cox [6] proposed a digital watermarking is a technology of embedding watermarking with intellectual property rights into images, videos and audios and other multimedia data by certain algorithm.

F.Harling [1] proposed the requirements and applications for watermark are reviewed. The applications are Copy right protection ,Data monitoring, Data tracking. Y. V. Swarup [13] Proposed a technique for fingerprinting relational data by extending Agrawal et .al watermarking scheme.fingerprinting is a class of imformation hiding techniques that insert marks into data with the purpose of identifying the recipients who have been provided data.

III. PROPOSED WORK

We propose a few approaches to the problem of watermarking relational data . In this paper, we present a watermarking technique for relational data that is highly resilient compared to these techniques. In particular, our proposed technique is resilient to tuple deletion, alteration, and insertion attacks. The main contributions of the paper are summarized as follows:

- We formulate the watermarking of relational databases asa constrained optimization problem, and discuss efficient techniques to handle the constraints. We present two techniques to solve the formulated optimization problem based on genetic algorithms and pattern search techniques.

- We present a data partitioning technique that does not depend on marker tuples to locate the partitions and thus it is resilient to watermark synchronization errors.

- We develop an efficient technique for watermark detection that is based on an optimal threshold. The optimal threshold is selected by minimizing the probability of decoding error.

With a proof of concept implementation of our watermarking technique, we have conducted experiments using both synthetic superiority of our technique with respect to all types of attacks.and real-world data. We have compared our watermarking technique with previous approaches [7], [12]. Figure 1 shows a block diagram summarizing the main components of the watermarking system model used. A data set D is transformed into a watermarked version DW by applying a watermark encoding function that also takes as inputs a secret key Ks only known to the copyright owner and a watermark W. Watermarking modifies the data. However these modifications are controlled by providing usability constraints referred to by the set G. These constraints limit the amount alterations that can be performed on the data, such constraints will be discussed in detail in the following sections. The watermark encoding can be summarized by the following three steps:

Encoding

Step E1. Data set partitioning: by using the secret key Ks the data set D is partitioned into m non-overlapping partitions {S0, . . . , Sm−1}.
Step E2. Watermark embedding: a watermark bit is embedded in each partition by altering the partition statistics while still verifying the usability constraints in G. This alteration is performed by solving a constrained optimization problem.

Step E3. Optimal threshold evaluation: the bit embedding statistics are used to compute the optimal threshold T□ that minimizes the probability of decoding error.

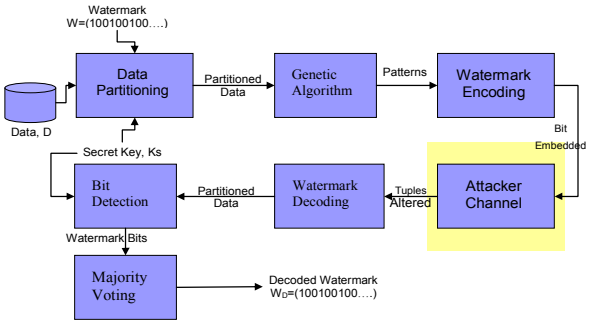

Fig. 1. Stages of watermark encoding and decoding

Decoding

Step D1. Data set partitioning: by using the data partitioning algorithm used in **E1**, the data partitions are generated.
Step D2. Threshold based decoding: the statistics of each partition are evaluated and the embedded bit is decoded using a threshold based scheme based on the optimal threshold T□.
Step D3. Majority voting: The watermark bits are decoded using a majority voting technique.

IV. DATA PARTITIONING

we present the data partitioning algorithm that partitions the data set based on a secret key Ks. The data set D is a database relation with scheme D(P,A0, . . . ,A_−1), where P is the primary key attribute, A0, . . . ,A_−1 are _ attributes which are candidates for watermarking and |D| is the number of tuples in D. The data set D is to be partitioned into m no overlapping partitions namely {S0, . . . , Sm−1}, Such that each partition Si contains on average |D| m tuples from the data set D. Partitions do not overlap, that is, for any two partitions Si and Sj such that i 6= j we have Si ∩ Sj = {}. For each tuple r ∈ D the data partitioning algorithm computes a message authenticated code (MAC) which is considered to be secure and is given by H(Ks||H(r.P||Ks)), where r.P is the primary key of the tuple r, H() is a secure hash function and || is the concatenation operator. Using the computed MAC tuples are assigned to partitions. For a tuple r its partition assignment is given by partition(r) = H(Ks||H(r.P||Ks)) mod m Secure Hash Algorithm (SHA-1) - takes data blocks up to 2^{64} bits long and produces a 160-bit message digest. A message digest is a number which is created by applying SHA-1 algorithm to the data partitions and represents that data partition uniquely. If the data partition changes, the message digest will change.The data partitioning algorithm that partitions the data set based on a secret key. The data set is a database relation with scheme which consists of primary key attribute , other attributes which are candidates for watermarking and number of tuples in dataset.

The data set is to be partitioned into a set of non-overlapping partitions, such that each partition contains on average number of tuples from the data set . For each tuple the data partitioning algorithm SHA -1 computes a message authenticated code (MAC) which is the concatenated message with the primary key of the tuple and the MAC. Using the computed MAC tuples are assigned to partitions. Using the property that secure hash functions generate uniformly distributed message digests this partitioning technique on average places MAC tuples in each partition. Our data partitioning algorithm does not rely on special marker tuples for the selection of data partitions, which makes it resilient to watermark synchronization attacks caused by tuple deletion and tuple alteration.

V. WATERMARK EMBEDDING

To encode bit into set, the bit encoding algorithm optimizes the hiding function . The objective of the optimization problem of maximizing or minimizing the hiding function is based on the bit such that if the bit is equal to 1, then the bit encoding algorithm solves using the maximization problem.However, if the bit is equal to 0, then the problem is simply changed into a minimization problem.

A. Genetic Algorithm Technique

A genetic algorithm (GA) is a search technique that is based on the principles of natural selection or survival of the fittest. Figure2 represents the steps involved in genetic algorithm process, The process start with a set of possible solutions (represented by chromosomes) the population. Solutions from one population are taken and used to form a new population. This is motivated by a hope that the new population will be better than the old one. New solutions (offspring) are selected according to their fitness - the more suitable they are the more chances they have to reproduce by mutations (crossover). Each chromosome has a corresponding value of the objective function, referred to as the fitness of the chromosome. To handle other types of constraints, we penalize the infeasible chromosomes by reducing their fitness value according to a penalty function , which represents the degree of infeasibility.

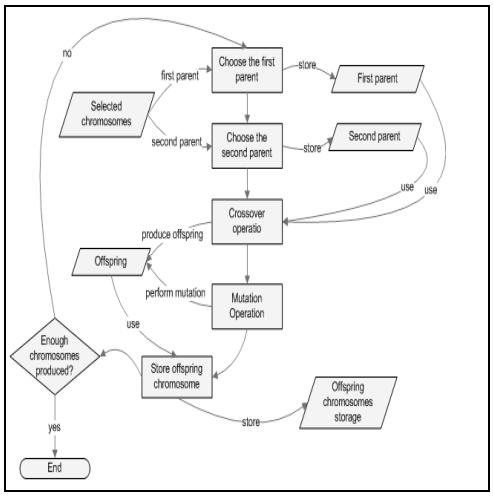

Fig. 2: Steps in Genetic algorithm

B.. *Pattern Search Technique*

Pattern search methods are a class of direct search methods for nonlinear optimization. Pattern search methods have been widely used because of their simplicity and the fact that they

work well in practice on a variety of problems. More recently, they are provably convergent [9][10]. Pattern search starts at an initial point and samples the objective function at a predetermined pattern of points centered about that point with the goal of producing a new better iterate. Such moves are referred to as exploratory moves. Pattern Search starts at an initial point and samples the objective function at a predetermined pattern of points centered about that point with the goal of producing a new better iterate. Such moves are referred to as exploratory moves. If such sampling is successful (that is, it produces a new better iterate), the process is repeated with the pattern centered about the new best point. If not, the size of the pattern is reduced, and the objective function is again sampled about the current point.

VI . WATERMARK DECODING

The algorithm starts by generating the data partitions using the watermarked data set , the secret key , and the number of partitions as input to the data partitioning algorithm.The watermark decoding will be performed in three different steps.
Step 1: (Sorting & Partitioning)
Partition data set using the same approach used in the encoding phase.
Step 2: (Bit Detection)
For each partition compute *HMAC* and decode the embedded bit.
Step 3: (Majority Voting):
Watermark bits are embedded in several partitions use majority voting to correct for errors.

VII. ATTACKER MODEL

In this section we discuss the attacker model and the possible malicious attacks that can be performed. Assume, Alice is the owner of the data set D and has marked D by using a watermark W to generate a watermarked data set DW. The attacker Mallory can perform several types of attacks in the hope of corrupting or even deleting the embedded watermark. A robust watermarking technique must be able to survive all such attacks. We assume that Mallory has no access to the original data set D and does not know any of the secret information used in the embedding of the watermark, including the secret key Ks, the secret number of partitions m, the secret constant c, the optimization parameters and the optimal decoding threshold T□. Given these assumptions Mallory cannot generate the data partitions {S0, . . . , Sm−1} because this requires the knowledge of both the secret key Ks and the number of partitions m, thus Mallory cannot intentionally attackcertain watermark bits. Moreover, any data manipulations executed by Mallory cannot be checked against the usability constraints because the original data set D is unknown. Under these assumptions Mallory is faced with the dilemma of trying to destroy the watermark and at the same time of not destroying the data. We classify the attacks preformed by Mallory into three types, namely *deletion, alteration and insertion* attacks

EXPERIMENTAL RESULTS

We report the results of an extensive experimental study that analyzes the resilience of the proposed watermarking scheme to the attacks. All the experiments were performed on Intel Pentium IV CPU 3.2GHz with 512MB RAM. We use real-life data from a relatively small database that contains the daily power consumption rates of some customers over a period of one year. The database size is approximately 5 Megabytes; for testing purposes only a subset of the original data is used with 150000 tuples. We used c = 75%, a 16 bit watermark, a minimum partition size _ = 10, a number of partitions m = 2048, the data change was allowed within } 0.5%. The pattern search algorithm was used for the optimization. The optimal threshold was computed using the technique usedI to minimize the probability of decoding error. The watermarked data set was subject to different types of attacks including deletion, alteration, and addition attacks. The results were averaged over multiple runs. Similar results were obtained for both uniform and normally distributed synthetic data. We show that it is difficult for Mallory to remove or alter the watermark without destroying the data.

A. *Deletion Attack*

In this attack Mallory randomly drops _ tuples from the watermarked data set, the watermark is then decoded and watermark loss is measured for different _ values. Furthermore, in this test we compare our implementation with Sion *et al.* (No Stored Markers) [4] approach. Figure3 shows the experimental results; they clearly show that our watermarking technique is resilient to the random deletion attack. Using our technique the watermark was successfully extracted with 100% accuracy

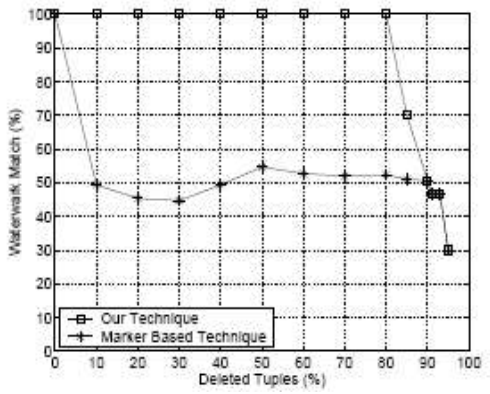

Fig. 3. Resilience to deletion attack.

CONCLUSION

In this paper, we have presented a resilient watermarking technique for relational data that embeds watermark bits in the data statistics. The watermarking problem was formulated as a constrained optimization problem, that maximizes or minimizes a hiding function based on the bit to be embedded. Genetic algorithm and pattern search techniques were employed to solve the proposed optimization problem and to handle the constraints. Furthermore, we presented a data partitioning technique that does not depend on special marker tuples to locate the partitions and proved its resilience to watermark synchronization errors. We developed an efficient threshold-based technique for watermark detection that is based on an optimal threshold that minimizes the probability of decoding error. The watermark resilience was improved by

the repeated embedding of the watermark and using majority voting technique in the watermark decoding phase. Moreover, the watermark resilience was improved by using multiple attributes.A proof of concept implementation of our watermarking technique was used to conduct experiments using both synthetic and real-world data. A comparison our watermarking technique with previously-posed techniques techniques shows the superiority of our technique to deletion, alteration and insertion attacks.

REFERENCES

[1] F.Harling and M.Kutter, " Multimedia watermarking techniques [MWT] proc.IEEE. Vol.87,no,7,pp 1079-1107 July 1999

[2] G. Langelaar, I. Setyawan, and R. Lagendijk. Watermarking Digital Image and Video Data: A State-of-the-Art Overview. *IEEE Signal Processing Magazine*, 17(5):20–46, September 2000.

[3] R. Lewis and V. Torczon. Pattern Search Methods for Linearly Constrained Minimization. *SIAM Journal on Optimization*, 10(3):917–941, 2000.

[4] R. Lewis and V. Torczon. Pattern Search Methods for Linearly Constrained Minimization. *SIAM Journal on Optimization*, 10(3):917–941,2000.

[5] I.Cox, J. Bloom, and M. Miller. *Digital Watermarking*. Morgan Kaufmann, 2001.

[6] R. Agrawal and J. Kiernan. Watermarking Relational Databases. In *Proceedings of 28th International Conference on Very Large Data Bases*, Hong Kong, China, 2002.

[7] D. Gross-Amblard. Query-Preserving Watermarking of Relational Databases and XML Documents. In *PODS '03: Proceedings of the 22nd ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, pages 191–201. ACM Press, 2003.

[8] E. Dolan, R. Lewis, and V. Torczon. On the Local Convergence of Pattern Search. *SIAM Journal on Optimization*, 14(2):567–583, 2003.

[9] L. Vaas. Putting a Stop to Database Piracy. *eWEEK, Enterprise News and Revs.*, Sept 2003.

[10] R. Sion, M. Atallah, and S. Prabhakar. Rights Protection for Relational Data. *IEEE Transactions on Knowledge and Data Engineering*, 16(6), June 2004

[11] Y. Li, V. Swarup, and S. Jajodia. Fingerprinting Relational Databases: Schemes and Specialties. *IEEE Transactions on Dependable and Secure Computing*, 02(1):34–45, Jan-Mar 2005

[12] M. Shehab, E. Bertino, and A. Ghafoor, "Watermarking Relational Databases Using Optimization Based Techniques," CERIAS Tech Report-(2006).

[13] "Digital Signatures in Relational Database Applications" online available at GRANDKELL systems INC. www.gradkell.com.(2007)

****