ISSN:2320-0790

An International Journal of Advanced Computer Technology

Blind Watermarking Technique for Relational Database

Nahla El_Haggar[1], Mahmoud M. Elkhouly[2], Samah S. Abu El Alla[3]

Information Technology Department, Faculty of Computers & Information, Helwan University, Egypt
[1]*Nahla_elhaggar@yahoo.com*
[2]*melkhouly@yahoo.com*
[3]*Eng_samahsaid76@yahoo.com*

Abstract: Recently, using of relational database system in many real life applications is increased. Therefore watermarking is considered as a vital technique for copyright protection of relational database. In this paper, we embedded watermark into some non numeric attribute. The generation of watermark depends on hash function and secret key to increase the security and the difficulty to guess by attackers. Our proposed technique is fully blind, which means that no need for the original tables to detect the watermark.

Keywords: Relational database system; hash function; copyright protection; watermarking; non numeric attribute

I. INTRODUCTION

The branch of watermarking technique appeared since 1995; this technique has been used for many propose such as copyright protection, rights management and copy control of digital contents, etc. Recently, due to great developments in computer and internet technology, digital watermarking becomes one of the best solutions to prevent illegal copying. In general, watermark is small, hidden perturbations in the database used as evidence of its origin. Inserting mark into original data is used to demonstrate the ownership. It should not significantly affect the quality of original data and should not be able to be destroyed easily. Database watermarking techniques consists of two stages: (a) watermark Insertion, (b) watermark Detection [1] as shown in figure 1.

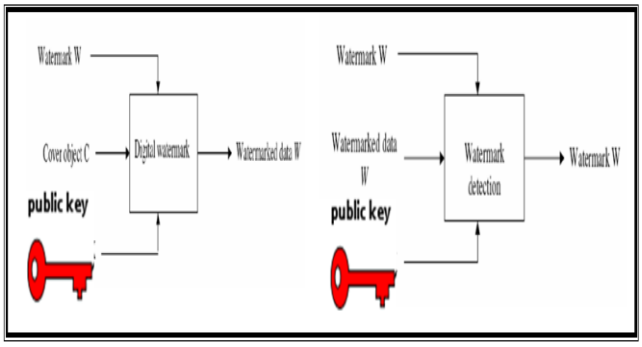

Figure 1: (a) digital watermarking – insertion, (b) digital watermarking – detection

In order to achieve the copyright protection, the algorithm should meet some basic requirements [2].

a. Imperceptibility: the watermark should not affect the usability of the original database or multimedia, thus it should be invisible /inaudible.

b. Robustness: the watermarked data should not be removed or eliminated by unauthorized distributors, so it must be robust against the various types of attackers.

c. Security: owner must use private key and hash function to increase the security of the system thus the watermark should only be detected by authorized person.

d. Blind detection: Watermark detection should be done without needing neither the knowledge about the original database nor the watermark. This property is critical as it allows the watermark to be detected in a copy of the database relation, and also later updates to the original relation.

e. The watermark should be undetectable without prior knowledge of the embedded watermark algorithm.

The watermark doesn't need to add data before or in the header of the original data it is directly embedded in the original relation

APPLICATIONS OF WATERMARKING

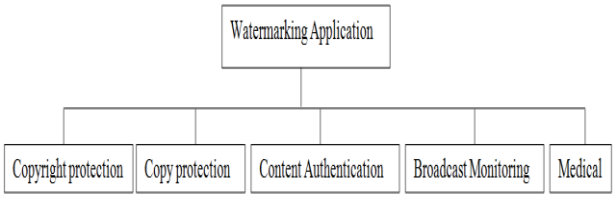

Figure 2: Watermarking Applications

Figure 2 shows some applications of watermarking, where:

- Copyright protection: watermarking does not prevent copying, but it deters illegal copying by using insertion and detection algorithms to detect the original ownership of a redistributed copy.
- Copy protection: it prevents unauthorized users to copy original data. So attackers cannot make illegal copy of data and redistribute it.
- Content Authentication: embedded a watermark to detect modifications to the host data [3].
- Broadcast Monitoring: watermark is embedded in commercial advertisements. Automated monitoring system can verify whether the advertisements are broadcasted as contracted or not. The main use of broadcast monitoring is to protecting the valuable TV products like news items from illegal transmission.
- Medical Applications: patient's information are inserted as watermark in medical images. It helps in avoiding ambiguities in searching the medical records [2].

.

B. Types of Watermarking Schemes

Various types of watermarking schemes have been developed for different applications according to their nature. Digital Watermarking schemes could be classified into three categories: fragile watermarking, semi-fragile watermarking and robust watermarking. The main difference between fragile and semi fragile watermarking is that semi-fragile watermarking is tolerant to non-malicious attack compression while fragile watermarking is intolerant to many manipulations. Robust watermarking is indented for application of copyright protection. Ours watermarking approach is robust watermarking where in watermarks should survive attacks aiming at weakening or erasing the attack [4].

Our technique deals with non-numeric data and it preserves query results by introducing almost zero distortion to semantic value of data. The goal is identify pirated copies of original data. It does not prevent copying, but it deters illegal copying by providing a means of establishing the ownership of a redistributed copy. El-Bakry [5], presents many approaches and algorithms. This paper is organized as follows: in section 2; background is presented, in section 3; proposed technique is described in details, and in section 4; conclusion is presented.

II. BACKGROUND

Current watermarking for relational database can be summarized as shown in figure 3, where:

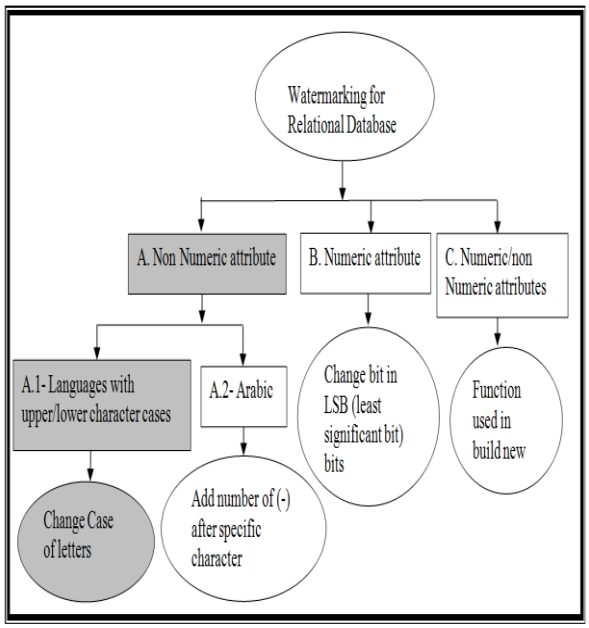

Figure 3: Types of watermarking for relational database

Schemes for non numeric attributes

A.1- English or languages with upper/lower character cases: Shah [6], described scheme to introduce a new embedding channel by embedding watermark in the alphabetic data attributes. Since the database, queries are case insensitive so it will not affect the semantic meaning of data if the case is changed from small to capital or vice versa. This method is applicable to all the languages with upper/lower character cases. Not all attributes need to be watermarked. Owner will decide which attributes are more suitable for watermarking.

Ali [7], used a binary image as watermark relational databases that segment into short binary strings that are encode in non-numeric, multi-word attributes of selected tuples of the database. The embedding process of each short string is based on creating a double space at a location determined by the decimal equivalent of the short string. Extraction of a short string is done by counting number of single-spaces between two separated double space locations.

A.2- Arabic: Ali [8], explained algorithm that hiding watermarking bits in Arabic characters of multiword attributes of subset tuples by adding (-) so we can add multi bit in the same attribute their extensions, without sacrificing the readability and appearance of the character. Image is used to watermark the relational database. The bits of the image are segmented into short binary strings that are encoded in non-numeric, multi-word attributes of selected tuples of the database. The embedding process of each short string is based on expanding the first character of a word whose location is determined by the decimal equivalent of the short string. Extraction of a short string

is done locating the word in which one of its characters was expanded. The image watermark is then constructed by converting the decimals into binary strings. A major advantage of using the space-based watermarking is the large bit-capacity available for hiding the watermark. This algorithm is robust against many kinds of attacks.

B. Scheme for numeric attributes

Rakesh [9], proposed a database watermarking scheme for watermarking numeric attributes of the database. This scheme is based on watermarking LSB (Least Significant Bit) of selected numeric attributes of a selected tuples these selections are algorithmically and secured by secret key that known only by the owner.

Rakesh [10], showed that some bit positions for some of the attributes of some of the tuples contain specific values. The tuples attributes within a tuple, bit positions in an attribute, and specific bit values are all algorithmically determined under the control of a secret key known only to the owner of the relation. This bit pattern constitutes the watermark. Only if one has access to the secret key then watermark can be detected with high probability. Their analysis showed that the watermark can withstand a wide variety of malicious attacks as well as benign updates.

Hossein [11], used LSB and SHA-1(Secure Hash Algorithm) to embed image into the selected bits, using XORed with MSB (Most Significant Bit). This scheme refers to the image elements directly in two dimensions. It generates two values for each tuple by using a hash function. These generated values will be used for elements coordinates, and the value of pixel in this coordinates will be embedded into relevant tuple. the value of attributes in database is the most important point during watermarking and should be noted.

Zaihui [12], Used images which encrypted by chaos as the copyright and embedded it into numeric attributes according to different weights. To be effective and convictive the watermark embedding process is imperceptible, secure and reliable, the watermark extraction process is a blind detection process and the watermark scheme is resistant to some malicious attacks.

Yossra [13], provide the effective technique to protect copyright of Relational data, it is robust against various forms of malicious attacks and updates because it depends on using new hybrid techniques, first technique MAC (Message Authentication Code) that used one way hash function SHA1, second technique is threshold generator base on simple combination of odd number of register and by using secret key this system is blind that means that to detect the watermark neither requires access to the original data nor the watermark. The watermark can be detected even in a small subset of a watermarked relation as long as the sample contains some of the marks.

C. Scheme for numeric/non-numeric attributes

El-Bakry, et. Al. [5], they added new record (altering the table) relies on the original data in each field of the relational database. The function used in constructing the new record as well as the secret key known only by the data owner. This technique has many advantages. First, it is available for any relational database. Second, it does not require any additional time because the calculations required for the new record are done off line. Third, it is not possible to delete the hidden record because it has been locked with a secret key. The values in the hidden record are known only by the data owner. Furthermore, there is no need for additional storage area as required when adding additional columns.

III. PROPOSED SCHEME

Our scheme has two sides: sender side and receiver side.

At sender side:

- the first step is embedding the watermark into the relational database.
- the second step is applying compression into watermarked database, as shown in figure 4.

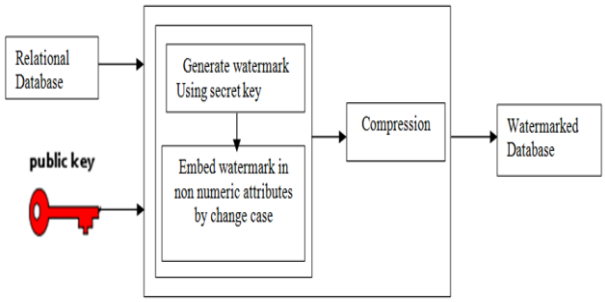

Figure 4: Block diagram for sender side

At the receiver side:

- The first step is applying decompression into compressed watermarked relational database.

The second step is detecting watermark from subset of the relational database, as shown in figure 5.

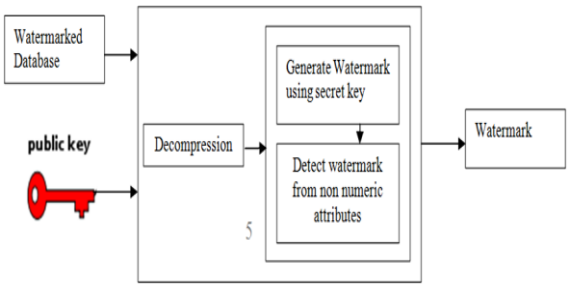

Figure 5: Block diagram for receiver side

The proposed technique uses compression to reduce storage space and transfer time, and embedded watermark in non numeric data to not affect semantic meaning of data. The proposed method is applicable to all the languages with upper/lower character cases. This technique marks only alphabetic attributes without

introducing any change in their semantic meaning. We use this method to avoid the manipulation of numeric data, which may have some margin of error. Our method does not need watermarking all attributes. Data owner will decide which attributes are more suitable for watermarking by using MAC (Message Authentication Code). That used one way hash function SHA1, so that we can say that our system consider robust against various type of attacks. After compression and embedded phases, watermark generation is used. It based on Message Authentication Code using secret key and marked tuple .then XORed with Message Authentication Code using secret key and marked attribute that generate random watermark which is difficult to guess by attacks. Detecting the watermark neither requires access to neither the original database nor the watermark. The watermark can be detected even in a small Subset of a watermarked relation as long as the sample contains some of the marks. Therefore our system considers blind system.

A. Message Authenticated Code (MAC)

MAC is one way hash function H for hash value calculation. There are number of hash function like MD4, MD5, SHA1, SHA256, etc. this function has important characteristics:

(1) If given message (M) of arbitrary length it is easy to compute a fixed length hash value (h) where h= H (M).

(2) Given hash value h, it is hard to compute the original message (M) such that H (M) =h.

(3) Given message (M) of arbitrary length, it is hard to find another message M` such that H(M) = H (M`).

For watermarking a database relation whose scheme is R (r.p, A_0, A_1,...,A_{v-1}) where r.p is the primary key, and A_0, A_1,.., A_{v-1} are attributes. For simplicity, assume that all v attributes A_0, A_1,.., A_{v-1} are candidates for marking

A. *Algorithm-1: The watermarking Insertion*

Input:
// Ķ only the owner knows the secret key.
// R is the relation to be marked.
// the parameter Ɣ and v are also private to owner.
Output: New value of r.A (relation of attribute) has change case.
Begin
1- for each tuple r € R do
2- Vp= hash1 (r.p,Ķ) // generate virtual primary key using secrete key
3- tuple marked =Vp mod Ɣ
4 - if (tuple marked equals 0) then // marked tuple
5 - Attribute marked i = Vp mod v // marked attribute
6- r.Ai =EmbedWm (r.Ai,tuple marked, attribute marked, Ķ)
// generate watermark and embedding in database.
EmbedWm (τ,tuple marked, attribute marked, Ķ) return string
T1= hash2(tuple marked , Ķ) // SHA2 hash function
T2= hash2(attribute marked, Ķ)
T=T1 XOR T2 // generate watermark
if 1th bit position of T=1 and (r.Ai is single word OR r.Ai is multi word) then
r.Ai has title case
if 1th bit position of T=0 and (r.Ai is single word OR r.Ai is multi word) then
r.Ai has all caps
7 - until ending of tuples in relation
8- Compress R // reduce size of database
End

Where:
r: record of a relation
K: secret Key known only to owner
v: number of attributes in the relation available for marking
1/y: fraction of tuples marked
R: relation
Vp: virtual primary key
α: significance level of the test for detecting a watermark
Ŕ: minimum number of correctly marked tuples needed for detection
T: watermark
τ: attribute Value
n: number of tuples in the relation
w: number of tuples marked

1. Generate Hash Values and select attribute: For embedding watermark into database the relation R (r.p, A_0, A_1,…..,A_{v-1}) with primary key r.p, use the cryptographic secure one-way hash function, with a secret key (KEY) in conjunction with the database table primary key (r.p) as hash function inputs in order to generate a virtual key as shown in line 2 .Lines 3,4 determine tuples index to be marked, using virtual key and mod operation . This determination depends on secret key K known to the owner, so only the owner can identify which tuple is to be marked. Line 5 determines the attribute that will be marked amongst the ν candidate attributes by using virtual key and mod operation. The selection of marked tuple and marked attribute depends on the private key of the owner. For erasing a watermark, therefore, the attacker will have to guess not only the tuples, but also the marked attribute within a tuple.

2. Embed watermark : embedding watermark function that called (EmbedWm) in algorithm 1 The inputs of this function are the value of marked attributes , index of tuple marked , index of attribute marked and the secret key. this function divided into two main parts as shown below
   a. Generate watermark :T1 is the hash value of 160 digit computed from hash2 function for inputs(tuple marked and secret key) , T2 is the hash value of 160 digit computed from hash2 function for inputs(attribute marked and secret key) T is the value 160 digit generating by T1 XORed T2
   b. Change case of selected attribute: check value of 1th bit position of T if this value =1 so attribute value become title case, if value of 1 th bit position of T=0 so change case of attribute value to upper case.

3. Compression: compress database before sending to the receiver to reduce the size and increase the transfer speed.

B. *Algorithm-2: The watermarking Detection*

Input:
// the parameter k, Ɣ and v have the same used for watermark insertion.
// α is the test significance level that the detector selects.
// totalcount = matchcount = 0
Output: detect Watermark
Begin
1- Decompress S // return the original database size
2- For each tuple s Є S do
3- tuple marked = hash1(s.p, k)mod Ɣ
4- if (hash1(s.p, k) mod Ɣ equals 0) then
5- Attribute marked i = hash1 (s.p ,k) mod v
6- totalcount = totalcount + 1 // increase tuples
7- matchcount = matchcount + WmDetection (s.Ai, tuple marked,attribute marked, Ķ)
WM detection (τ,tuple marked, attribute marked, k) return number // detect watermark
T1= hash2 (tuple marked, k)
T2= hash2 (attribute marked, k)
T=T1 XOR T2 //generate watermark
if (r.Ai has title case and 1th bit position of T=1) then return 1
Else return 0
if (r.Ai has all caps and 1th bit position of T=0) then return 1
Else return 0
8- goto 2
9- Ґ=threshold (totalcount, α)
10 - If (matchcount ≥ Ґ) then detect watermark
End

The watermark detection algorithm has three phases:

1. Decompress: to return database to original size at receiver side.
2. Generate Hash Values and select attribute: are explained in insertion algorithm from line 3 to line 5.

Detection: WmDetection subroutine compares the case of selected attribute with the case that must have been set for that attribute by the watermark insertion algorithm. Thus know how many tuples are tested (totalcount) and how many of them contain the expected case (matchcount).Line 6 increases the totalcount that determined how many tuples marked in insertion algorithm, line 7 increased matchcount when Wm detection subroutine returns 1. matchcount determined how many tuples match with marked tuples in insertion algorithm. In WmDetection subroutine the inputs of subroutine are the value of marked attributes and index of tuple marked, index of attribute marked and finally the secret key .this subroutine gets T1 ,T2 and T with the same method that mention in insertion algorithm. Line 8 goto to line 2 to take each tuple in relation and processing it from line 3 to line 7, until end of tuples in relation, line 10 the matchcount is compared with the minimum count returned by the threshold function in line 9 for the test to succeed at the chosen level of significance α

IV. CONCLUSION

Our technique embedding watermark into some non numeric attribute that selected secretly and algorithmically by using one way function and private key to protect the security of model. The generation of watermarking by using hash function depends on marked tuple, marked attribute, and secret key to increase robustness of the technique. Compressing the relation to decrease transfer rate has been used. This technique has several advantages: it decreases transfer rate, preserves query results, and it doesn't distortion to semantic value.

V. REFERENCES

[1] Rajneesh Kaur Bedi, Purva Gujarathi B.E., Poonam Gundecha B.E., and Ashish Kulkarni B.E., "A Unique Approach for Watermarking Non-numeric Relational Database", International Journal of Computer Applications (0975 – 8887) Volume 36– No.7, December 2011.

[2] Vipula Singh, "Digital Watermarking: A Tutorial" , Cyber Journals, Multidisciplinary Journals in Science and Technology, Journal of Selected Areas in Telecommunications (JSAT), January Edition, 2011.

[3] R.Manjula Nagarjuna Settipalli, VelloreTamil, "A new Relational Watermarking Scheme Resilient to Additive Attacks", International Journal of Computer Applications (0975 – 8887) Volume 10– No.5, November 2010

[4] Abha Tamrakar, and Chhattisgarh Swami, "Compression of Watermarked Relational Database for Security and Optimization of Storage Consumption ", International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-1, Issue-2, December 2011.

[5] Hazem El-Bakry and Mohamed Hamada, "A Novel Watermark Technique for Relational Databases",Japan © Springer-Verlag Berlin Heidelberg 2010.

[6] S.A. Shah, Sun Xingming and Hamadou Ali , "Query Preserving Relational Database Watermarking ",Network and Information Security Lab Hunan University, Changsha, Hunan, China ,2010.

[7] Ali Al-Haj, Ashraf Odeh, and Shadi Masadeh, "Copyright Protection of Relational Database Systems" Amman, Jordan,2010

[8] Ali Al-Haj and Ashraf Odeh," Robust and Blind Watermarking of Relational Database Systems", Journal of Computer Science 4 (12): 1024-1029, 2008,ISSN 1549-3636 ,© 2008 Science Publications

[9] Rakesh Agrawal Jerry Kiernan,"Watermarking Relational Databases",IBM Almaden Research Center,2002.

[10] Rakesh Agrawal, Peter J. Haas, Jerry Kiernan, "Watermarking relational data: framework, algorithms and analysis". The VLDB Journal 12, 2 (Aug. 2003).

[11] Hossein Moradian Sardroudi, Subariah Ibrahim, OmidZanganeh, "Robust Database Watermarking Technique over Numerical Data",Journal of Communications and Information Sciences. Volume 1, Number 1, April 2011

[12] Zaihui Cao, Jianhua Sun, and Zhongyan Hu," Image Algorithm for Watermarking Relational Databases Based on Chaos",Department of Art and Design, Zhengzhou Instiute of Aeronautical Industry Management,450015 Zhengzhou, China,2010.

[13] Yossra H. Ali, Bashar Saadoon Mahdi, "Watermarking for Relational Database by using ThresholdGenerator",Computer Sciences Department, University of Technology Baghdad,2011 .