International Journal of Engineering & Technology, 7 (1.9) (2018) 234-237

International Journal of Engineering & Technology

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

Research paper

Content characteristics based robust watermarking for relational database: a new approach to database security

Manoj Kumar [1] *, O. P. Verma [2]

[1] *Department of Computer Science and Engineering, Delhi Technological University, Delhi, India*
[2] *Department of Information Technology, Delhi Technological University, Delhi, India*
**Corresponding author E-mail: mkumarg@dce.ac.in*

Abstract

Digital data such as text, relational database, audio, video and software are intellectual property of creators/ writers/owners. The database services have become easily available and economical since the booming of internet. However, their outsourcing through the internet accompanies multiple threats like copying, modifying as well as unauthorized distribution. Relational Database has a wide-spread use in many real-life applications, hence, it is essential to maintain integrity and provide copyright protection. To counter the threats, watermarking techniques have been playing a vital role since the last decade. As a security measure, Relational Database Watermarking is becoming more popular and strengthened day-by-day. This is also one of the upcoming areas of interest among researchers of the Database Security. A technique earlier used for Image Watermarking is applied to watermark Relational Databases. In Image Watermarking technique, a pixel or a pair of pixels must satisfy certain characteristics. Usually, database watermarking techniques concentrate on hiding a watermark in database. Extraction and matching of hidden watermark with original watermark confirms ownership of database. This paper demonstrates the use of image watermarking technique for relational databases. Here we align some properties of attributes of database by changing some bit(s) in attribute value. Using secret key, we have ensured that values of two attributes of a tuples satisfy some bit-similarity property and to do so, we slightly alter values of attributes. Detection of such characteristic in a database using secret key can be done easily to verify the presence of a watermark.

1. Introduction

Nowaday, relational database systems are used in many real-life situations. Relational database presents the information in tables with rows and columns. A table is referred to as a relation in the sense that it is a collection of objects of the same type (rows). Data in a table can be related according to common keys or concepts, and the ability to retrieve related data from a table is the basis for the term relational database. Many of us share data through internet in the form of text, database, image, video and software. Sometimes, authors/ creators need to outsource their database across Internet for educational, commercial or official purposes. However, such sharing also has potential security threats. The data can be hacked and misused. The hacker/ attacker can modify/ transform the data and redistribute the same across the internet pretending to be the owner/ author. Such an act has multi-fold effects on the owner of the digital products, mainly financial and intellectual. Moreover, establishing the rightful ownership of the actual owner becomes difficult. Therefore, protection of data and copyright from such misuse is essential and extremely crucial. Watermarking is one of the most common security measures. Watermarking Multimedia Object is a comparatively mature field and a lot of material is available in this area. Many techniques are available which efficiently deal with the copyright issue of multimedia products. But the algorithms proposed for images, videos, texts, audio and software for multimedia objects are not useful for Relational Database Watermarking due to difference in nature of the database from other digital assets. Many techniques have been suggested to assure the integrity of Relational Database. The researchers in the database fields import the idea of Watermarking for the copyright protection of their Relational Database. The idea has been given for the first time by Agrawal and Kiernan[2]. Watermarking multimedia objects has successfully dealt with the problems of piracy and copyright violation.

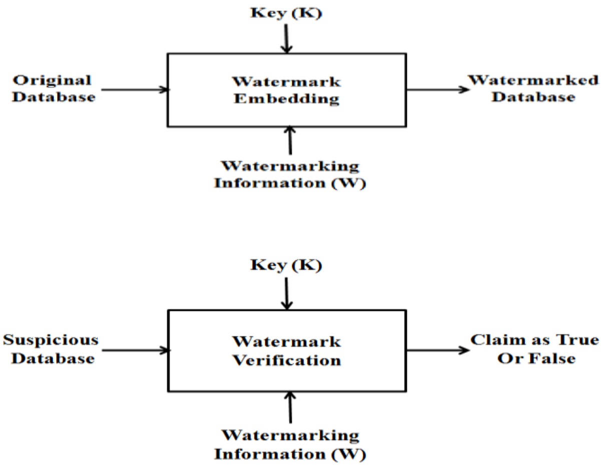

Fig. 1: Basic Database Watermarking Technique: Embedding and Extraction Process.

2. Related work

Relational Database Watermarking has been a big challenge for the researchers. Rakesh Agrawal and Kiernan [2] introduced the idea for watermarking Relational Database for the first time, many people have given their own algorithm for preserving relational database integrity. Hu,Cao and Sun [6] have suggested a scheme using an image as watermarks, while Pournaghshband[8] have tried to introduce 'fake' tuples among the original tuples. However, our technique successfully tested for reducing the chances of Subset Selection attack. For embedding the watermark, we have considered the entire database. Unnikrishnan and Pramod[10] proposed a new method for tamper detection, analysis and original data recovy by adding a set of bits as a watermark. Their algorithm provides a strong validation and recovery scheme to maintain data security and integrity.

Watermarking is the process of secretion of additional information within digital data (such as audio, image, and video), documents and software codes in such a way that it cannot be detected. Digital Watermarking technique refers to the activity of embedding the specified watermark information (such as signature, name, logo, ownership information, etc.) in the protective information (such as audio, text, video, or picture) and gathering the watermark information from protective information not perceived by human perceptual system.

In other words, it is a process of embedding a signal containing some digital watermark information unique to the copyright owner in object (image, audio, text, or video) which is required to be protected. As shown in Figure 1, a Digital Watermark is defined as a visible or invisible identification code that is permanently embedded in the data to transmit the hidden data. It remains present in the data and recipient of the data is unaware of such hidden watermark in the data received. Only owner of the data can have secret key and technique to extract the hidden watermark and prove his ownership. It usually provides the copyright protection of the intellectual property. The watermark is used to recognize the actual and rightful copyright owner of the object.

According to the human perception, the digital watermarks can be further divided into Source-based Watermarks and Destination-based Watermarking. It is desirable for authentication or ownership- identification where isolated watermark, recognizing the owner, is introduced in all the copies of specific software being distributed. A Source-based Watermark is used for authentication as well as to determine whether acquired software or other electronic data has been tampered with or not. The watermark could be destination-based where each allocated copy gets a distinctive watermark, identifying the specific buyer. The Destination based Watermark is used to detect the buyer in the case of unlawful reselling.

Wang, Cui and Cao [11] talked about describing a Speech based Watermarking system. Bhattavcharya and Agostino [4] have put forward a Distortion Free Database Watermarking technique. Ali and Odeh [3] have proposed a database watermarking algorithm based on inserting binary image watermarks in non-numeric multi-word attributes of selected database tuples. A large bit capacity is available to hide the marks in the database. Using image as watermarks introduces errors to the data.

Dong, Li,Ge and Lei [5] presented an algorithm resistive to inevitability attack in Watermarking Relational Databases .Xiang, Huang and Peng [12] have introduced a technique which uses image as watermark. This technique supports easy watermark identification. But it also has flaws; for example, whether the watermarked data is still usable according to their method is still unwarranted because they reset the whole decimal fraction and this kind of alteration may be so insignificant that normal application of data is bound to be affected. Zhang, Xiaoming, Wang, and Li [13] have put forward a method of watermarking database using image. Sion, Atallah and Prabhakar [9] specifically stated about watermarking numerical data. A new reversible database watermarking algorithm using Firefly Algorithm, a bio-inspired optimization algorithm, is introduced by Imamoglu, Ulutas, and Ulutas [7]. The method is reversible and uses an algorithm called firefly to determine the best candidate pairs to embed the watermark.

3. Proposed work

Watermarking a database is different from Digital Watermarking. Relational Database is a collection of unrelated tuples while the attributes are independent of each other. In case of digital watermarking, techniques which are mainly meant for still images, audio and videos are considered media as a whole, cannot be directly applied to databases. The conventional techniques apply watermark to that blob of data, while, in case of databases, each tuple must be secured independent of other tuples.

The fundamental objectives of watermarking techniques of relational data is to deliver with well-ordered performance with respect to the following important metrics proposed by F. Petitcolas [1]:

- Minimal alteration of database due to watermark,
- A watermark system should be blind i.e. at the time of extraction of watermarks there is no need of the original database,
- The time cost essential to detect the watermark and embed it, subsequently, in a doubtful relation,
- The cost of the preservation of the secret keys and some other functional information (if any) that are essential to maintain confidentiality for the detection phase,
- The capability of a relation to stay watermarked following modification operations (deletions ,insertions, and updates of database tuples), and,
- Robustness of the technique to malicious attacks.

Perhaps, in nearly all the well-known robust watermarking schemes for relational database, a small part of numeric data is altered based on a secret key in such way that these changes can be figured out for the motive of ownership proof.

In our approach, we take a database DB and produce a Watermarked Database W_{DB}. Let (A_0, A_1… A_m) be those attributes of database DB where altering one bit in lower order 8 bits of value does not affect the value significantly. This alteration can be applied to mantissa part of floating point values of database. Using single attribute at a time and select different attribute for different tuples, we spread the watermark evenly in m attributes. In this, for each tuple r $\in$ R encrypt the primary key P_K using AES as

$$E_{PK} = \text{Encrypt}\ (P_K, K) \tag{1}$$

After encrypting P_K, select n bits from E_{PK} to choose one of the $m=2^n$ attributes. Then, again select 3 other bits from E_{PK} to choose the position of bit from the selected attribute to apply watermark. This selected bit is set to '1' by performing a bitwise OR operation on that bit with '1'.

1) for each tupler $\in$ R
2) E_{PK} = Encrypt (P_K, K): encrypt primary key P_K using secret key K.
3) Select n bits b_1...b_n from E_{PK}. It is used to select attribute for watermark insertion.
4) Select another three bits (b_1, b2, b3) from E_{PK} to choose one of the bit from the above selected attribute.
5) Transform the above selected bit to '1'.

In watermarked database, we can use a similar process to verify if the database has been tampered. The user may select a threshold for percentage of violation to conclude whether the database has been tampered or not. Usually this threshold is set below 20 percent, that means if percent of violation is above 20 percent, we can conclude that database is tempered or not the one we have watermarked.

An alternate to above suggested scheme is to align ith bit of an attribute A_x with jth bit of another attribute A_y of a tuple. If ith bit of attribute A_x is matching with jth bit of attribute A_y, no changes are made. Otherwise, ith bit of attribute A_x is copied to jth bit of attribute A_y. It means for all tuples in database, i^{th} bit of attribute A_x is always equal to j^{th} bit of attribute A_y. But i and j are not fixed for all tuples. They are different for each tuple and are decided by selecting bits from E_{PK} as we have done above.

1. Select A_x and A_y, two attributes for hiding watermark.
2. For each tupler ∈ R
3. E_{PK} = Encrypt (P_K,K)
4. Select 3 bit from E_{PK} to decide value i, and select another 3 bits from E_{PK} to decide value of j.
5. Make j^{th} bit of attribute A_y identical to i^{th} attribute of A_x.

Comparing both the approaches, approach using two attributes performs better as few number of bits are altered in database. In first approach, altered bit is always set to '1', whereas in second approach using two attributes, bit can change from '0' to '1' or from '1' to '0'.

3.1. Analysis of error introduced during watermark embedding process:

Consider IEEE Standard 754 for double precision floating point number representation. This format uses 1 sign bit, 11 exponent bits and 52 bits in mantissa. Proposed method selects one out of 8 least significant bits of mantissa in attribute A_x and A_y and aligns them by changing j^{th} bit of attribute A_y. Thus one of the least significant 8 bits of mantissa changes its value from ‘1’ to ‘0’ or ‘0’ to ‘1’. For 52 bits of mantissa say “$m_1m_2\ldots m_{51}m_{52}$”, and 11 bit exponents e, and bias b, equivalent floating number is 1.f x $2^{(e-b)}$. As only one bit changes its value in mantissa that is one of the least significant 8 bits of mantissa ($m_{45}..m_{52}$), maximum change in mantissa value is $\pm {}^{1}/_{2^{45}}$ and minimum change in mantissa value is $\pm {}^{1}/_{2^{52}}$.

4. Results

The present scheme has suggested that using Single Attribute Modification for a tuple by setting a bit to ‘1’, is tested on a database with 10000 tuples. After inserting watermark in a test database, we perform test for various types of attacks on database that can alter our embedded watermark. In this, we are doing insertion , updating attack and checking the violation occurring in it. Deletion attack does not affect the watermark information as deleting a tuple containing watermark is simply reducing number of tuples having watermark.

4.1. Insertion

As insertion attack, upto 50% additional tuples are inserted into database and these new tuples do not have watermarking information. During watermark extraction process, some of these tuples will not be satisfying watermark characteristics i.e. do not have value '1' at designated bit position, and these violations are listed in Table 1.

Table 1: Insertion Attack

No. of Tuples Insert during attack	Total No. of Tuples After insertion	No. of violation detected	Violation percentage
1000	11000	460	4.18
2000	12000	944	7.86
3000	13000	1422	10.93
4000	14000	1881	13.43
5000	15000	2363	15.75

4.2. Updation

When a query in database updates some attribute(s) of database, our watermarking property of database tuples are disturbed. Amount of this distortion is usually dependent upon whether we are updating all attributes involved in database watermarking or we are updating single attribute or multiple attributes of database. In these cases also, watermaking property of some of the attributes will be disturbed and, hence, we see violations of characteristics in some of the tuples of database. Watermarking characteristics are spread into 8 attributes of database here but in experiment we have tried updation attacks for maximum 5 attributes as shown in Tables 2-6.

Table 2: Updation of One Attribute (Total 10000 Tupes in Database)

No. of Tuples Updated during Attack	No. of Violation Detected	Violation Percentage
1000	54	0.54
2000	71	0.71
3000	157	1.57
4000	210	2.10
5000	279	2.79

Table 3: Updation of Two Attributes

No. of Tuples Updated during Attack	No. of Violation Detected	Violation Percentage
1000	72	0.72
2000	92	0.92
3000	187	1.87
4000	262	2.62
5000	314	3.14

Table 4: Updation of Three Attributes

No. of Tuples Updated during Attack	No. of Violation Detected	Violation Percentage
1000	92	0.92
2000	101	1.01
3000	196	1.96
4000	299	2.99
5000	330	3.30

Table 5: Updation of Four Attributes

No. of Tuples Updated during Attack	No. of Violation Detected	Violation Percentage
1000	96	0.96
2000	141	1.41
3000	236	2.36
4000	317	3.17
5000	364	3.64

Table 6: Updation of Five Attributes

No. of Tuples Updated during Attack	No. of Violation Detected	Violation Percentage
1000	110	1.10
2000	171	1.71
3000	283	2.83
4000	392	3.92
5000	430	4.30

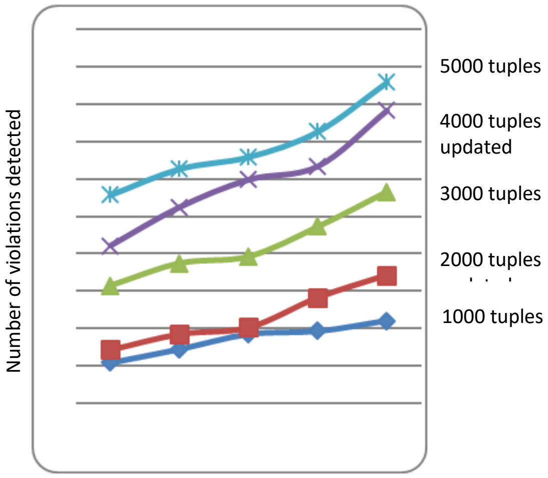

Fig. 2: Comparison of Different Updation Attacks.

Figure 2 shows comparison of results of updation attacks when updation attack is done on different number of attributes. As it is clear from the graph that when we perform updation on more number of attributes, chances of watermark modifications are more. Thus graph shows rise in distortions as we increase number of attributes. Comparing updation attack results with results of scheme proposed by Sion, Atallah and Prabhakar [9], proposed algorithm having violation percentage below 4.30% much less than violation percentage in range of 11% obtained in other scheme.

5. Conclusion

Our approach alters the data to an extent that data standard of altered database is acceptable. Reversible Watermarking Methods are used to serve to scenarios capable to recover the original data from the watermarked data. These methods, to some extent, also ensure that the data is up to the mark. However, these methods are not robust against malicious attack particularly those methods that target some chosen tuples for watermarking.

In this paper, a novel method for watermarking relational databases is presented which is detectable, robust, and follows properties of incremental update, blindness and imperceptibility. The results of the experimental study show that, even if an intruder adds, alters or deletes tuples, our approach is able to successfully detect the inserted watermark.

We have proposed two versions of the method

i). inserting watermark by selecting one of m attributes and setting a bit of selected attribute.

ii). fixing two attributes A_x and A_y and aligning ith bit of A_x with jth bit of A_y.

Few more possible variants are listed below:

i). We can select different A_x and A_y for each tuple, where value of x and y can be decided by bits in E_{PK}.

Inserting watermark by selecting one of m attributes and matching ith bit of E_{PK} with jth bit of selected attribute, where i can be fixed whereas j can be decided by bits in E_{PK}.

References

[1] F. Petitcolas, Watermarking schemes evaluation. IEEE Signal Processing, vol. 17,110.5, pp.58-64, 2000.

[2] Agarwal, R., &Kiernan, J. Watermarking Relational Databases. InVLDB '02 Proceedings of the 28th international conference on Very Large Data Bases (pp.155-166). ACM, 2002.

[3] Al-Haj, A., Odeh, A. (2008). Robust and Blind Watermarking of Relational Database Systems. Journal of Computer Science, 4 (12) 2008, pp.1024-1029. https://doi.org/10.3844/jcssp.2008.1024.1029.

[4] Bhattacharya, S.,and Agostino, C. A Distortion Free Watermarking Framework for Relational Databases. In Proceedings of the 4th International Conference on Software and Data Technologies (ICSOFT ’09) (pp.229–234). Sofia, Bulgaria: INSTICC Press.

[5] Dong,X., Li,X., Ge,Y.,and Lei, Z.(2009). An Algorithm Resistive to Invertibility Attack in Watermarking Relational Databases. IEEE Control and decision conference, China, 2009 (pp. 1532-1537). https://doi.org/10.1109/CCDC.2009.5192217.

[6] Hu,Z., Cao,Z., and Sun,J. An Image Based Algorithm For Watermarking Relational Database. International Conference on Measuring Technology and Mechatronics Automation, 2009 (pp. 425-428).IEEE.

[7] Mustafa Bilgehan Imamoglu, Mustafa Ulutas, and Guzin Ulutas, A New Reversible Database Watermarking Approach Firefly Optimization Algorithm. Mathematical Problems in Engineering, Volume 2017, Article ID 1387375.

[8] Pournaghshband, V. A New Watermarking Approach for Relational Data. In Proceedings of the 46th Annual Southeast Regional Conference on XX (ACM-SE ’08), (pp.127–131), Auburn, Alabama. ACM Press.

[9] Sion, R., Atallah, M., and Prabhakar,S. On Watermarking Numeric Sets. InProceedings of the 1st international conference on Digital Watermarking (IWDW’02) (pp.130-146).ACM.

[10] Unnikrishnan K., Pramod K. V., Dynamic Prediction Based Watermarking for Temporal Relational Databases. IEEE International Conference on Data Science and Engineering (ICDSE) 2016.

[11] Wang, H., Cui,Z. &Cao,Z. A Speech Based Algorithm for Watermarking Relational Databases.In Proceedings of the 2008 International Symposiums on Information Processing (ISIP ’08), (pp.603–606), Moscow, Russia. IEEE Computer Society.

[12] Xiang, Z., Huang,M., &Peng, Z. An Additive-attack-proof Watermarking Mechanism for Databases Copyrights Protection Using Image. In Proceedings of the 2007 ACM symposium on Applied computing (SAC ’07), (pp. 254–258), Seoul, Korea. ACM.

[13] Zhang, Z., Xiaoming, J., Wang, J., & De-Yi, Li. Watermarking Relational Database Using Image. In *Proceedings of the 3rd International Conference on Machine Learning and Cybernetics*, IEEE, Shanghai, PRC, vol. 3, 2004 (pp. 1739–1744).IEEE.