International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-8 Issue-6, March 2020

A Semi-Fragile Watermarking Scheme for Integrity Checking of Relational Databases

Murugan R., John T. Abraham, Ibrahim Salim

Abstract: ***Extensive use of the Internet coupled with tremendous growth in database applications have created a huge demand for database security. The concepts of e-governance, e-commerce, e-business, e-learning, and digital libraries are already in place and evolving across poles. This raises various concomitant threats such as illegal copying, illegal redistribution, tampering, forgery and authentication of copyrighted digital assets. Digital Watermarking is an effective technique which can be introduced for solving the above mentioned threats. Based on the detection of the Watermark, the ownership and the integrity of the data can be asserted. Database Watermarking techniques are generally classified into two: i.e., Robust or Fragile. Robust watermarking techniques are designed for copyright protection and fragile watermarking techniques are for authentication or integrity checking of data. Watermarking schemes for relational databases authentication are almost fragile in nature. These algorithms do not allow any legitimate modification of the data. In most of the cases, innocent distortions such as tuple and/or attribute sorting may be considered as tampering. In this paper, the research work proposes a novel semi-fragile watermarking technique for tamper detection of relational databases. The new semi-fragile scheme can detect and localize malicious modifications, while allowing authorized updates. Theoretical analysis and experimental results demonstrate that common database attacks can be detected with high rate of success.***

Keywords: ***Digital Watermarking, Relational Databases, Copyright Protection, Information Security, Database Attack, Semi-Fragile Watermarking Technique.***

I. INTRODUCTION

Fragile watermarks and semi-fragile watermarks are primarily used to certify the integrity and authenticity of digital content. The fragile watermarking technique is used to achieve truthful authentication. It considers the digital contents as an entirety and does not allow any tampering. In the case of Relational Database even if there is only a single attribute value change, the data cannot pass the certification system. The semi-fragile watermark combines the advantages of both robust and fragile watermarking techniques together.

Algorithms for semi-fragile watermarking technique for Relational Database authentication focus on the ability of the detection of tamper attacks carried out on the marked data. Because the semi-fragile watermark has a certain degree of fragility, the algorithm can envisage the data authentication according to whether the marked data is tampered with or not. When the protected data suffers some kind of attacks, the information used for watermarking purpose will make a corresponding change as well. However, for authorized or innocent attacks, the semi-fragile watermark has a certain degree of robustness, so that the authorized and the malicious tampering or operations can be distinguished.

In this paper, a semi-fragile watermarking method has proposed for authentication or verification of Relational Databases. The proposed scheme has the following features:

- Robustness:
  Authorized updates in the marked data are possible without making any difference in the watermark information.
- Fragileness:
  Unauthorized modifications of the marked data can be detected. Even a minute change in the attribute value will make changes in the watermark information.
- Distortion Free:
  The insertion process in the proposed scheme does not perform a physical insertion, but does only a logical insertion of the marks to the underlying data. So the proposed technique is convenient for application which requires zero distortion on the attribute values of the database.
- No Constraints on Data Type of Attributes:
  The proposed scheme considers all the attribute values of the Relation for extracting the Watermark Key and the scheme does not put any restriction on the type of the attributes of the relation (data).

In this proposed scheme, a semi-fragile watermarking technique for relational databases, which authenticates or verifies the database even if the attacker tampers the data by changing the values of the attributes or tuples reordering of databases. The watermarking technique proposed is robust in the case of innocent attacks, since nothing will happen to the watermark even though the attacker tampers with the data by arranging them in different orders. And the watermark is fragile to malicious modifications of the data. This shows the semi-fragile nature of the watermark. The watermarking scheme proposed is robust since the watermark will not be lost even though the attacker changes the data values.

Manuscript received on February 10, 2020.
Revised Manuscript received on February 20, 2020.
Manuscript published on March 30, 2020.
* Correspondence Author
Murugan R*, Associate Professor, Department of Computer Applications, MES College Marampally, Ernakulam, Kerala, India.
E-mail: mes.murugan@gmail.com, Orcid Id: 0000-0003-3137-8236
John T Abraham, Assistant Professor in Computer Science, Bharata Mata College, Kochi, Kerala, India. E-mail: johntabraham@yahoo.com
Ibrahim Salim, Research Scholar, Research & Development Centre, Bharathiar University, Coimbatore, Tamilnadu, India and Assistant Professor, MES College Marampally, Ernakulam, Kerala, India.
E-mail: mes.ibrahimsalim@gmail.com

The experiments show the effectiveness of the proposed scheme of watermarking for maintaining copy right information there by ensuring right protection to relational databases.

The paper is structured as follows: Section 2 discusses the concept of digital watermarking, copy right protection, integrity checking and main challenges for watermarking relational databases and Section 3 presents the review of related works.

Section 4 constructs the solution for embedding and extraction of watermark in relational databases. Section 5 presents implementation details as well as experiments and evaluations of the proposed watermarking technique and Section 6 concludes.

II. DIGITAL WATERMARKING, COPY RIGHT PROTECTION AND INTEGRITY CHECKING

The main purpose of Digital Watermarking is to protect a certain content from unauthorized duplication and distribution by enabling provable ownership over the content. For protecting the copyright of databases, copyright information, message, or logo etc. are inserted or embedded in the databases, which is to be protected. The insertion algorithm embed copyright information which can be extracted later by the extraction algorithm to prove the authenticity and ownership. Even though, the copyright notice will not guarantee the ownership protection of the digital content, still it is used. Generally, the digital properties like databases, software, images, audio and videos contain copyright information, may be visible or invisible. It is desirable to achieve very high level of robustness for copyright protection watermarking.

The digital watermark can be considered as some kind of information that is embedded into digital assets for detection of tampering, authentication, traitor tracing, etc. [1]. The scheme which is used for digital watermarking can be perceptible or imperceptible, depends on its detectability in the watermarked content [2], [3]. Again, the scheme of a watermark may be robust or fragile. In robust watermarking technique, the changes to the marked content will not affect the watermark and in fragile technique, the watermark destroyed when the marked content is tampered or modified. Watermark can be classified into blind or informed on the basis of data requirement during extraction process. While extracting watermark in the case of blind watermarking, the original content is not required. But in the case of informed watermarking, the original content is required. Zero watermarking scheme is blind watermarking in which the original content is not modified when watermark is inserted into it.

Watermarking techniques used for text and multimedia could not be used for watermarking relational databases. The relational data defers from multimedia data in many respects: (i) Low redundancy: Relational databases, by design, focus on preventing redundancy within the data pool. Conclusively, low redundancy means high entropy, which minimizes potential possibilities of hiding additional information, (ii) High sensitivity to alterations: Certain databases, tables or attributes do not tolerate even slight modifications to the stored values, required to embed hidden information (e.g. medical, military and/or research data sets)., (iii) Frequent modifications Typical database tables are subject to frequent modifications, like table updates and row inserts. These processes can heavily distort or even delete an embedded watermark [4]. In the case of text watermarking, the watermarking scheme make use of special properties of text formatting and semantics. For instance, text watermarks are often introduced by altering the spacing between words and lines of text [5].

A major share of the traditional watermarking techniques developed for protecting the relational databases introduce some marks or errors into the data, to be watermarked, during the watermark insertion process [6]-[18]. Even though the distortions introduced by the insertion process are assumed to be minor, they inevitably reduce the usefulness and quality of the protected data. For example, attributes like date of birth, social security number, GPS coordinates, etc. might not tolerate such alterations in the data. Similarly, any distortion to the categorical data may be considered as significant. Another issue is the intrinsic strife between robustness and imperceptibility of the watermark information. Generally, the more changes introduced by a watermark scheme, the more secure is the watermarking technique.

III. RELATED WORKS

During the past two decades, numerous research papers have been published on digital multimedia watermarking techniques, including image, audio and video data as carrier. Database watermarking, on the other hand, is being in a rather early stage of research, whereas different approaches are of considerable interest. Many watermarking schemes have been published for integrity verification of relational database. Agrawal et al. proposed a watermarking scheme, which is based on attribute having numeric data type and marking is done at bit-level [4]. In this technique the watermark bits are hidden in the least significant bits of attributes in selected tuples.

Sion et al. proposed watermark scheme for numerical data for relational databases [2]. In this proposal, the scheme is dependent on a secret key which is used to insert in the normalized data set using the most significant bits. The data set is divided into partitions using markers and the partition statistics is varied to hide watermark bits.

Li et al. [19] proposed a fragile watermarking technique for tamper detection of categorical data. The tuples in the relation are first divided to partitions. Embed the watermark by physically modifying the order of the tuples without any legal updates to the database.

In order to mark any type of data, including integer, real, character and boolean, a public watermarking technique is proposed by Li and Deng [20] without dread of any error constraints. The unique feature of this scheme is that, no secret key is used and it can be verified in the public domain as many times as necessary. A public key, which is obtained by one-way hashing from various information like the identity of the owner(s) and characteristics of the database, is used in both the embedding and detection phase. The characteristics may the name of the database and its version.

Bhattacharya and Cortesi [21] proposed a watermarking technique which has the same algorithmic steps followed by Li and Deng's technique [20].

The two distinguishes in such a way that the scheme proposed by Li and Deng considers a private key instead of using it as public, and due to this, the watermark key cannot be verified in the public domain. Besides, the former is based upon partition and considers the binary watermark extracted as an image that is utilized to prove the authenticity.

Insertion of new tuples in the database relation being protected is the step on which the watermarking approaches proposed in [22]-[25] are based.

A process meant to create fake records and inserted them incorrectly into the database has been explained by Pournaghshband in [22]. The attributes which are treated as candidate key and susceptibility level of non-key attributes are taken care of by this fake records formation algorithm. In this technique, the Bernoulli sampling probability pi is applied for the ith non-key attribute Ai and the purpose being to decide its fake value which may be selected consistently or as the value with higher occurrence frequency in the existing set of values of Ai in the relation.

Safeguarding of both textual and numerical relational data was the objective of the method devised by El-Bakry and colleagues [23]-[25]. This technique hides only one tuple s watermark in the database relation, instead of adding many fake tuples. The process computes the values of the new record applying a function which is based on the numeric values of the data which is stored in the original records. The scheme proposed uses two different algorithms for numeric and non-numeric attributes. Besides, no extra storage space is necessary as needed when adding new columns as proposed in [26].

The fragile watermarking technique introduced by Prasannakumari in [27] is explained as: the database relations are altered by inserting a virtual attribute which will serve as watermark information containing parity checksum of other attributes. Obtain the aggregate value from any of the numeric attribute of all records. The process of virtual attribute insertion is then performed independently for each non-overlapping partition obtained from the original relation.

The proposal made by Kamel [28] is an R-tree structure based watermarking technique in contrast to the traditional watermarking schemes. The proposal explains fact that R-tree is not putting any conditions in the order of entries inside the node. Arrangement of entries inside the R-tree is made relative to a secret initial order, in this method in such a manner so as to correspond to the value of the watermark.

The objective of the scheme proposed by Tsai et al. [29] is maintaining the integrity of the information of databases. It is based on the mechanism of public authentication. In this scheme a watermark W is created first as an $n \times \sqrt{n}$ image, where n represents the number of records in the relation, besides four corners having mark of the owner.

In [30], Murugan R et al. proposed a new robust watermarking scheme which is used to protect the copyright information of relational databases. The proposed technique is based on the zero-watermarking approach. For generating Watermark, the scheme uses an image as secret key. The algorithm which embed watermark into the data will not insert any additional information into the database values.

IV. PROPOSED TECHNIQUE

Let us suppose the relation R(Pk, A1, A2,…, An-1) being watermarked, with primary key Pk and non-key attributes Ai (i = 1,…, n-1). Most of the watermarking techniques for checking the integrity of relational databases are of fragile in nature. But in this literature, the research proposes a semi-fragile scheme of watermarking which is robust to innocent type attacks and is fragile to malicious type attacks. Table-1 shows the notations used in various parts of this paper.

Table-1. Notations

Symbol	Description
R	Native Relation
R_m	Watermarked Marked Relation
P_k	Primary Key of the Native Relation
K	Private Key
m	Number of Records
n	Number of Attributes
A_i	Name of ith Attribute; i = 1 to n
t_j	j^{th} Tuple of the relation; j = 1 to m
H_t	Hash value of Relation
H_p	Hash value of Key Attribute
HA_i	Hash value of Attribute Names; i = 1 to n
AV_i	Attribute value of ith attribute
HAV_{ij}	Hash Attribute value of ith attribute in the jth column
Nt_i	i^{th} attribute value of the Newly Created Tuple
WM_r	Registered Watermark
WM_c	Computed Watermark
$Hash$	Hash Function
CA	Certification Authority

A. Overview of the Approach

All the attributes of a relation have equal participation in most of the algorithms which support fragile type watermarking schemes. And in many situations, the intended users of a database may require all the tuples of a relation, but need not be in the same order as in the original relation. The tuples may be in different sort orders on the basis of different tuples or attributes. In the proposed work, the Owner categorizes the users into two groups. The former is supposed to do some innocent attacks like changing sort orders and the latter does some kind of malicious attacks like data value modifications. That is, the scheme specified in this chapter allows authorized modifications of the database by restricting unauthorized updating.

In this proposed technique, the watermark is used to verify the integrity of the database on different types of attacks. The watermark is robust in the case of innocent attacks, since the mark will not lost even after the attacker modified the data by arranging them in different sort orders. And the watermark is fragile to malicious modifications of the data. This shows the semi-fragile nature of the watermark. In the proposed algorithm a watermark is logically embedded in all the attributes of the database.

All the attributes and all the tuples should be sorted before the watermark insertion process. The same order of the attributes and tuples should be maintained at the detection phase also.

This algorithm integrity of the data is maintained even while allowing the authorized updating of data for intended users. The two phases of the proposed watermarking schemes are:

- **watermark insertion scheme**
- **watermark detection scheme**

B. The Process of Watermark Insertion

The watermark insertion phase describes the algorithm used to embed watermark in the database is illustrated in the watermark insertion phase. The inputs in the insertion phase are the native database and the private key. The private key can be a logo or any image given by the database owner, which is known only to the database owner. The key attribute will be bring out from the metadata of the database. The process of insertion is portrayed in Fig. 1.

The watermark insertion process is illustrated as follows:

Step-1. Input the Native Database (R), Private Key (K).
Step-2. Find the Hash Value of the Database (H_t).
Step-3. Find the Key Attribute (P_k) using the metadata.
Step-4. Sort the Tuples in the ascending or descending order of Key Attribute.
Step-5. Find the Hash Name Values of all the attributes.
Step-6. Sort all the columns on the basis of the ascending or descending Hash Name Values of the attributes.
Step-7. Generate a New Tuple by processing each attribute value of the existing tuples.
Step-8. Obtain the watermark (WM_r) by combining the Hash Values of the newly created tuple, Hash value of the Relation and the Secret Key which is known only to the Owner of the database.
Step-9. Register the Native Database, Key Attribute (P_k), Private Key (K) along with the Watermark (WM_r) with the Certification Authority (CA).

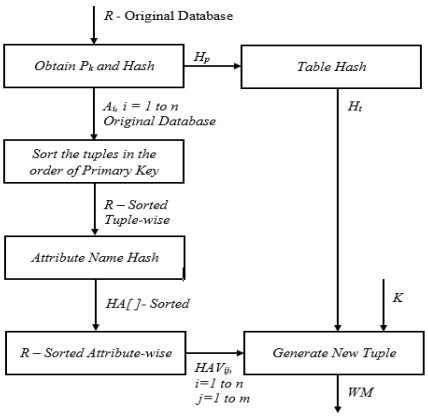

Fig. 1. Process of Watermark Insertion

C. Watermark Insertion Algorithm

The algorithm for watermark insertion is described in Algorithm 1. The Database and Private Key are the inputs and Watermark Key is the output.

Algorithm-1. Algorithm for Watermark Insertion

// Inputs are the Database, Private Key //

1 *Inputs R, K*

2 *Obtain the Key Attribute* P_k, H_p=*Hash(Pk),* H_t=*Hash(R* || H_p*)*

3 *Sort the tuples* $t_1, t_2, \ldots, t_m$ *in the order of* P_k.

4 *For i = 1 to n*

HA_i = *Hash*(A_i);

5 *Sort* AV_i, *i = 1 to n of each tuple in the order of* HA_i, *i = 1 to n;*

6 Nt_i=*Hash(sum*(HAV_{ij})*, j = 1 to m), for all i = 1 to n;*

7 WM_r = *Hash*(H_t || H_k || Nt_i*, for all i = 1 to n)*

8 *Register R, K, and* WM_r *with CA*

D. The Process of Watermark Detection

In detection process, the algorithm extracts the watermark and that can be used for checking the Integrity of the Database. The generated watermark key from the marked database, is compared with the registered key for substantiation. Fig. 2 shows the process of watermark detection.

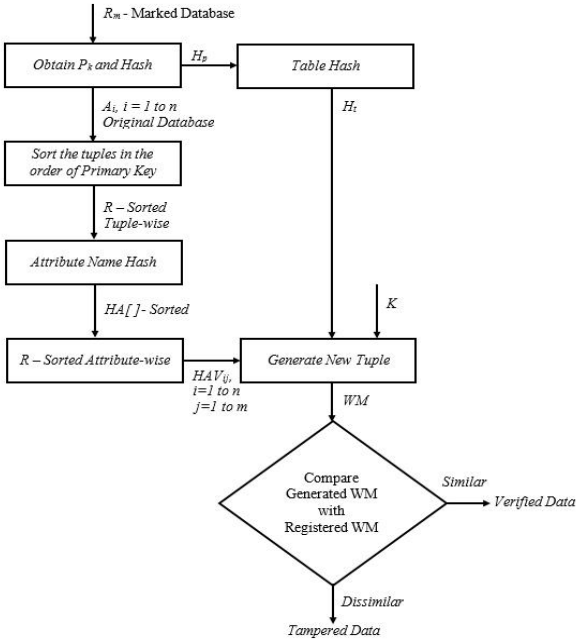

Fig. 2. Process of Watermark Detection

The process of watermark detection can be illustrated by the following steps:

Step-1. Input the Marked Database (R_m).
Step-2. Obtain the Private Key (K) and the Registered Watermark Key (WM_r) from the *Certification Authority*.

Step-3. Find the Hash Value of the Database (H_t).
Step-4. Find the Key Attribute (P_k) using the metadata.
Step-5. Sort the Tuples in the ascending or descending order of Key Attribute.
Step-6. Find the Hash Name Values of all the attributes.
Step-7. Sort all the columns on the basis of the ascending or descending Hash Name Values of the attributes.
Step-8. Generate a New Tuple by processing each attribute value of the existing tuples.
Step-9. Obtain the watermark (WM_n) by joining the Hash Values of the newly created tuple, Hash value of the Relation and the Private Key which is known only to the database owner.
Step-10. Watermark generated from the marked database is compared with the registered watermark (do compare WM_n and WM_r).
Step-11. If similarity exists, then display *"Verified Data"*. Or else if the keys are dissimilar then the out will be *"Tampered Data"*.

E. Watermark Detection Algorithm

Marked database is the input to this algorithm. The Key Attribute, Private Key and the registered Watermark Key will be accessed form the certification authority. The watermark obtained as the output of this algorithm can be used for similarity checking with the registered Watermark Key of the native database. Algorithm 2 illustrates the algorithm for watermark detection.

Algorithm-2. Algorithm for Watermark Detection

// Input – Watermarked Database//

1 *Input Watermarked database* R_m
2 *Obtain the Key Attribute* (P_k), *Private Key (K) and registered* (WM_r) *from resource center*
3 *Obtain the* $Hp=Hash(P_k)$, $H_t=Hash(R_m \| H_p)$
4 *Sort the tuples* $t_1,t_2,\ldots,$ *tm in the order of* P_k.
5 *For i = 1 to n*
    $HA_i = Hash(A_i)$;
6 *Sort* AV_i, *i = 1 to n of each tuple in the order of* HA_i, *i = 1 to n;*
7 $Nt_i=Hash(sum(HAV_{ij}), j = 1 \text{ to } m)$, *for all i = 1 to n;*
8 $WM_n = Hash(H_t \| H_k \| Nt_i$, *for all i = 1 to n)*
9 $WM_n = Hash(Combined\ HA[i]$ *for all i from 0 to n-1)*
10 *If* $WM_n == WM_r$ *then*
    Output "Verified Data";
Else
    Output "Tapered Data";

V. RESULT ANALYSIS

The proposed technique is of type semi-fragile, that is, the fragility of the watermark depends on the nature of attack on the attribute values of the relation. The proposed method is useful for both innocent and malicious attacks on the attribute values. Attribute reordering and sorting of records in a desired order constitute innocent attacks since these attacks will not change the attribute values of the relational databases. But changing the values of attributes, addition and deletion of tuples/attributes are of malicious in nature. The Watermark key in this type of scheme withstands the innocent types of attacks which do not change the attribute values.

A. Innocent Attacks

Innocent attacks will not make any changes to the data values. Attacks of this type may change the sort order of the records and/or the order of the attributes. In this proposed technique, formulating the watermark key is only after sorting the attribute values on the basis of attributes and records in a predefined order that is known only to the owner of the database. That is, the watermark scheme is robust to innocent attacks. Table-2 illustrate the outcome of innocent attacks on a Relational Database.

Table-2. Robustness Results due to Innocent Attacks

Basis of Reordering of Data	Criteria Taken	Watermark Detection Rate
Tuples	Ascending order of Primary Key	100 %
	Descending order to Primary Key	100 %
Attributes	Existing Order	100 %
	Arbitrarily disordered	100 %

B. Data Value Modification Attack

In data value modification, values of one or more attribute can be changed maliciously. The modifications in the data values of the relation result in the generation of a new watermark that will be different from the already obtained watermark using the original data. In other words, the watermark generated is fragile in nature. Table 3 shows the result analysis and the nature of Fragility and Tamper detection on different rates of Update Attack.

Table-3. Detection of Malicious Data Modifications with Different Rates of Attacks – Result Analysis

Insertion Attack Rate	Deletion Attack Rate	Update Attack Rate	Tamper Detection	Watermark Fragility
10 %	10%	10%	Yes (100%)	100 %
30 %	30 %	30 %	Yes (100%)	100 %
50 %	50 %	50 %	Yes (100%)	100 %
70 %	70 %	70 %	Yes (100%)	100 %
90 %	90 %	90 %	Yes (100%)	100 %

The result analysis shows that the tamper detection and watermark fragility is hundred percentage for any quantum of insertion, deletion or update attacks to the database as shown in Figure 3 and Figure 4.

C. Subset Attack

In this types of attacks, a set of new records or attributes are added to a watermarked database. This causes a change in the newly generated watermark during the detection process. The Table 3 gives the tamper detection and fragility rates on various volume of insertion attacks.

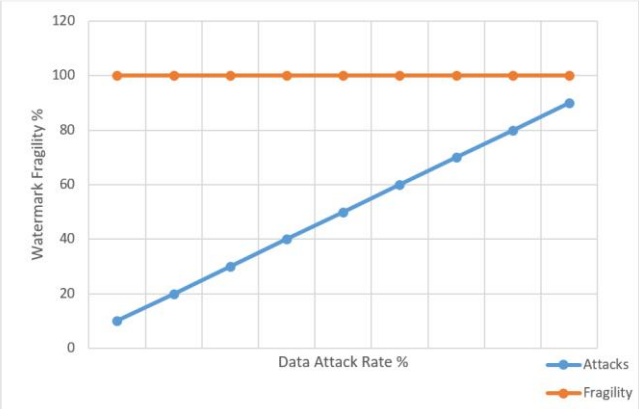

Fig. 3. Watermark fragility results on various rates of attack

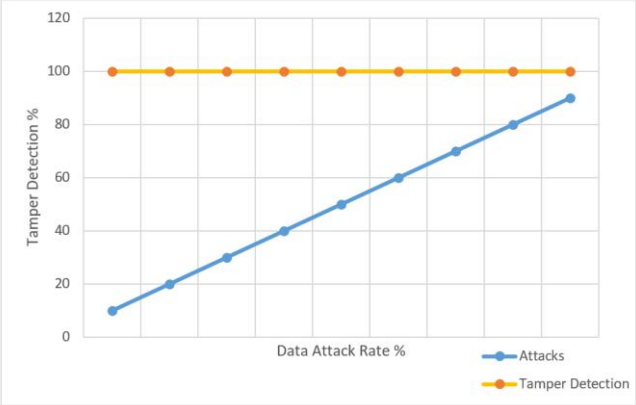

Fig. 4. Watermark tamper detection results on various rates of attack

The experimental results illustrated in this paper, as Table 2 and Table 3, show that the watermark is adversely affected by even minor malicious data value modifications and the watermark withstands for reordering of tuples and/or attributes of any kind. The new watermarking scheme presented in this chapter allow authorized modifications to the data to be protected without allowing unauthorized updates. The experiment conducted showed that the new scheme of this research is in the nature of Semi-fragile. Java and the Database Management System MySQL is used for implementing and testing the algorithm of the proposed scheme. Testing is conducted with a non-transactional data, from GitHub link https://github.com/datacharmer/test_db.

The dataset contains 2500 tuples, with attributes having datatypes such as numeric, non-numeric and date. A partial set of data values is given in Figure 6. The primary key "emp_no" of the relation "WM_EMPLOYEES" is obtained from the metadata. An image as in Figure 5 is selected from the workstation as the secret key for creating the watermark.

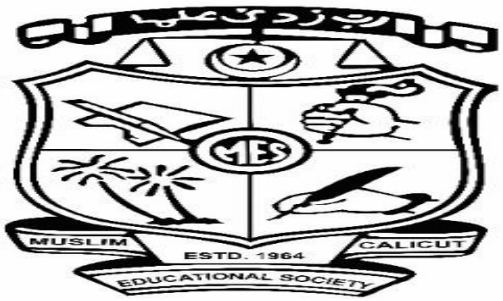

Fig. 5. Secret Key

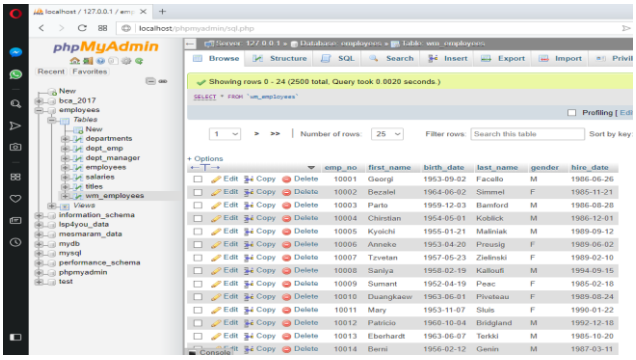

Fig. 6. Partial Employees Dataset

The research work verified to make sure that the proposed scheme is robust for innocent attacks and is fragile for data value modification, subset and superset types of attacks. This shows its Semi-fragility. This scheme allows authorized modifications but resists unauthorized data modification. In this process SHA256 hashing method is used and the generated Watermark for the purpose of registration is of 64 characters in length.

VI. CONCLUSION

In this paper, a Semi-Fragile Watermarking Scheme is proposed for Integrity Checking of Relational Databases. The proposed technique uses zero-watermarking approach and uses an image as private key for formulating the Watermark. The insertion algorithm does not induce any information into the cover data. The watermark computed using the proposed scheme, is registered with an Authority which can be later used for resolving dispute in connection with the authenticity or ownership. Without inserting any marks to the host data, the proposed scheme maintain its quality and usefulness. The proposed scheme is also suitable for watermarking database attributes of variant types. That is, there is no restriction on the data types of attributes selected for watermark insertion. Based on both theoretical analysis and experiments, the new proposed scheme demonstrated that the formulated watermark is Semi-Fragile and the scheme is effective in detecting any type of malicious attacks on the data.

REFERENCES

1. Raju Halder, Shantanu Pal and Agostino Cortesi.(2010). Watermarking Techniques for Relational Databases: Survey, Classification and Comparison. In Journal of Universal Computer Science, vol. 16, no.21 (2010), 3164-3190.
2. Sion, R., Atallah, M., and Prabhakar, S. (2005). Rights protection for categorical data. IEEE Transactions on Knowledge and Data Engineering, 17:912–926.
3. Z. Jalil, A. M. Mirza, "A Review of Digital Watermarking Techniques for Text Documents", IEEE, 2009.
4. R.Agrawal and J. Kiernan. Watermarking relational databases. In Proceedings of VLDB, 2002, pp 155–166.
5. N. Chotikakamthorn. Electronic Document Data Hiding Technique using Inter-character Space. In Proceedings of the 1998 IEEE Asia-Pacific Conference on Circuits and Systems (IEEE APCCAS). 1998, 419~422.
6. J.R.N. Baweu and H. Guo. Integrity Verification for XML Data. In Proceedings of the World Congress on Engineering and Computer Science (WCECS'07). San Francisco. 2007, 633~638
7. R. Yao, Q. Zhao, and H. Lu. A Novel Watermarking Algorithm for Integrity Protection of XML Documents. International Journal of Computer Science and Network Security. February 2006, 6(2): 202~207

8. R.C. Merkle. A Certified Digital Signature. In Proceedings of Advances in Cryptology (CRYPTO). 1989, 218~238
9. S.A. Shah, X. Sun, A. Hamadou, and X. Wang. Combined Watermarking Solution for XML Documents. International Journal of Digital Contents Technology and its Applications (JDCTA). November 2011, 5(11): 69~78
10. R. Liu and H. Wang. Integrity Verification of Outsourced XML Databases. In Proceedings of the International Conference on Computational Science and Engineering (CSE '09). 2009, 207~212
11. H. Guo, Y. Li, and S. Jajodia. Chaining Watermarks for Detecting Malicious Modifications to Streaming Data. Information Sciences. 2007, 177(1): 281~298
12. H. Xian and D. Feng. Leakage Identification for Secret Relational Data Using Shadowed Watermarks. In Proceedings of the 2009 International Conference on Communication Software and Networks (ICCSN'09). 2009, 473~478
13. Murugan R, Dr. John T Abraham, Aravind M J. A Study of Digital Watermarking on Relational Databases for Ownership Proofing and Tamper Detection. International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 3 (2018) Spl.
14. J. Guo, Y. Li, R. H. Deng, and K. Chen. Rights protection for Data Cubes. In Proceedings of Information Security Conference (ISC). 2006, 359~372
15. J. Guo and W.-D. Qiu. Watermarking Data Cubes. Journal of Shanghai Jiaotong University (Sci.). 2009, 14(1): 117~121
16. Jaseena K U, Murugan R, Dr. John T Abraham. A Zero Text Watermarking Algorithm for Protecting Plain Text Documents using Combined Image and Text through Compression and Encryption. International Journal of Applied Engineering Research ISSN 0973-4562 Volume 10, Number 10 (2015) pp. 26483-26498
17. S. A. Shah, X. Sun, A. Hamadou, and A. Majid. Semi-Fragile Watermarking Scheme for Relational Database Tamper Detection. In Proceedings of the 2011 3rd International Conference on Future Networks (ICFN'11). 2011.
18. Murugan R, Jaseena K U. John T Abraham. An Invisible Watermarking Technique for Integrity and Right Protection of Relational Databases. International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 24 (2017) pp. 15754-15758
19. Li, Y., Guo, H., and Jajodia, S. (2004). Tamper detection and localization for categorical data using fragile watermarks. In Proceedings of the 4th ACM workshop on Digital rights management (DRM '04), pages 73–82, Washington, DC, USA. ACM Press.
20. Y. Li and R. H. Deng. Publicly Verifiable Ownership Protection for Relational Databases. In Proceedings of the ACM Symposium on Information, Computer and Communications Security. 2006, 78-89
21. S. Bhattacharya and A. Cortesi. A Generic Distortion-free Watermarking Technique for Relational Databases. In Proceedings of the 5th International Conference on Information Systems Security (ICISS '09). LNCS. Kolkata, India. Springer. 2009, 5905: 252-264
22. H. M. El-Bakry and N. Mastorakis. A New Watermark Approach for Protection of Databases. In Proceedings of the 9th WSEAS International Conference on Applied Informatics and Communications (AIC '09). 2009, 243-248
23. H. El-Bakry and M. Hamada. A Novel Watermark Technique for Relational Databases. In Proceedings of the 2010 International Conference on Artificial intelligence and computational intelligence (AICI'10). Berlin Heidelberg. Lecture Notes in Computer Science. Springer-Verlag. 2010, 6320/2010: 226-232
24. V. Pournaghshband. A New Watermarking Approach for Relational Data. In Proceedings of the 46th Annual Southeast Regional Conference on XX (ACM-SE '08). Auburn, Alabama. ACM Press. 2008, 127-131
25. H.M. El-Bakry and M. Hamada. A Developed Watermark Technique for Distributed Databases Security. In Proceedings of the 3rd International Conference on Computational Intelligence in Security for Information Systems (CISIS'10). Advances in Intelligent and Soft Computing 85. Springer. 2010, 173-180
26. G. H. Gamal, M.Z. Rashad and M.A. Mohamed. A Simple Watermark Technique for Relational Database. Mansoura Journal for Computer Science and Information Systems. Jan 2008, 4(4): 1-7
27. V. Prasannakumari. A Robust Tamperproof Watermarking for Data Integrity in Relational Databases. Research Journal of Information Technology. 2009, 1(3): 115-121.
28. I. Kamel. A Scheme for Protecting the Integrity of Databases. Computers & Security. 2009, 28(6): 698-709
29. M. Tsai, H. Tseng, and C. Lai. A Database Watermarking Technique for Temper Detection. In Proceedings of the 2006 Joint Conference on Information Sciences (JCIS '06). Kaohsiung, Taiwan. Atlantis Press. 2006.
30. Murugan R, John T Abraham and Ibrahim Salim M. A Robust Watermarking Technique for Copyright Protection for Relational Databases. International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-8 Issue-3, September 2019: 4040-4046.

AUTHORS PROFILE

Dr. Murugan R received his PhD in Computer Science from Bharathiar University, Coimbatore. His other qualifications include MCA, MPhil (Computer Science), DRDBMS. He has 21 years of experience in teaching and currently works as an Associate Professor and HEAD of the Department of Computer Applications at MES College Marampally (NAAC reaccredited with A+ (3.38) in 2019), Aluva, Kochi, Kerala. His research interests include Information Security, Database Management System and Data Mining and has published research articles in International and National levels. He served the Department of Vocational Studies as a Nodal Officer for more than 6 years. He guides and mentors aspirants in research projects and software application developments with his academic and IT industrial experiences. He has achieved the Best Performing Teacher Award twice. He is also a member of various professional bodies.

Dr. John T Abraham received his PhD in Computer Science in 2001. His other qualifications include MCA, MSc(Information Systems and Management), MPhil, MTech(IT), DDBM, DAS400, DMF, DST. Till 2012 he worked in various colleges like Vaishnav College, Chennai, S A Engineering College Chennai, Vel Tech Engineering College Chennai, Saintgits College of Engineering Kottayam, KVM College of Engineering and Technology Cherthala, Mount Zion Engineering College Kadammanitta in various positions like Academic Director, Head of the Department etc. Two years he worked in the Faculty of Information Technology, Misurata University, Libya. From 2012 onwards he is working in Bharata Mata College, Kochi, Kerala. His research interests include Data Base Management Systems, Management Information Systems, Software Engineering etc. and published around 100 research articles in International and National level. He has also received many International and National level awards and is a member of various professional bodies.

Ibrahim Salim M was born in Kothamangalam in 1977. He is an assistant professor in the Computer Applications at MES College Marampally, Aluva, Kochi, Kerala. Currently he is doing his PhD in Data Mining at Bharathiar University, Coimbatore. His qualification includes MPhil (Computer Science) and MCA. His research and publication interests include Data Mining and Computer Networks. He has presented papers at various conferences. His teaching areas are Computer Networks, Operating Systems and various programming papers. He is the NCC officer of his college from 2010 onwards. He was the member of Board of studies of Computer Applications at Mahathma Gandhi University, Kottayam, Kerala. He conducted various seminars and conferences.