Watermark based Copyright Protection for Relational Database

Ramani Sagar V.
Dept. Of Information Science & Engg.
100 Feet ring Road, Banshankari III Stage, Banaglore -560085

ABSTRACT

As information hiding, the digital watermark techniques have been attracting more interests in both research and industrial fields. Relational database is widely used in many information systems, as a tool for storing and managing a data. Major issue to protect the copyright of relational data. In order to make watermarking information more intuitive and easier to identify and to give authorization to the database this idea has been proposed. Ownership verification of a database by inserting an imperceptive watermark in such a way to provide robustness and security against attempts to remove the watermark. To prove ownership of the database watermarking is done using image. Image is converted into the row bits and row bits is encrypted using MD5 security algorithm .This row bits will be embed into the database attribute in terms of watermark. To check the performance of this scheme will be tested based on different types of attacks on the database which shows the robustness of this scheme.

General Terms

Ownership verification, security algorithm ,database watermarking.

Keywords

Database watermarking, image , relational database,MD5 algorithm

1. INTRODUCTION

Recently, copyright protection of database is important in the field of information technology. This is partly because that the digital data is very easy to be tampered, peculated and illegality copied. Digital watermarking is an approach to cope with this situation. Protection from the piracy of digital assets is usually based upon the embedding of digital watermarks into the data.

1.1 Introduction about watermarking

Ownership verification for digital product by inserting watermark in such a way that it provides robustness against attempts to remove or modification of watermark. It provides a promising method of protecting digital data from illicit copying and manipulation by embedding a secret code directly into the data. In general, the database watermarking techniques consist of two phases: Watermark Embedding and Watermark Verification. During watermark embedding phase, a private key *K* (known only to the owner) is used to embed the watermark *W* into the original database. The watermarked database is then made publicly available.

1.2 Copyrighting & Protection

Copyright is a legal concept, enacted by most governments, giving the creator of original work exclusive rights to it, usually for a limited time. One such mechanism is based on *Information Hiding*. By concealing a resilient rights holder identity "signature" (*watermark*) within the digital Work(s) to be protected, Information Hiding for Rights Assessment (*Watermarking*) enables as an evidence court-time to prove associating particular Works with their respective rights .

1.3 Motivation

In many applications where relational databases are publicly available on the Internet. For example, to provide convenient access to information for users, governmental and public institutions are increasingly required to publish their data on the Internet . The released data are usually in tabular form. They may be statistical data produced by Census Bureau demographic surveys and Federal agencies such as National Center for Education Statistics and Energy Information Administration; they may also be databases released by the Department of Motor Vehicles and Health Maintenance Organizations . In these cases, all released data are public; what is critical for the owner of the data is to make sure that the released data are not tampered with it [1].

Databases are the foundations of business systems such as web servers and ERP applications. The data stored within the databases often includes sensitive information such as employee and client details, financial and confidential data. Databases have not been subject to the same security of networks and operating systems and are extremely complex systems. Confidential information kept in databases becomes the purpose of criminals very often. From data of CSI/FBI Computer Crime and Security Survey in 2006 32% organizations found out the unauthorized access to information, 10% organizations –theft of confidential information . Amount of incidents outside of organization and inside it was approximately equal. Respondents of survey estimate the losses caused by various types of computer security incident dropped significantly this year. Total losses are related to the unauthorized access to information and theft of proprietary information for 2006 – $16,651,000 for the 313 spondents. These digits are showing the seriousness of the problem [1].

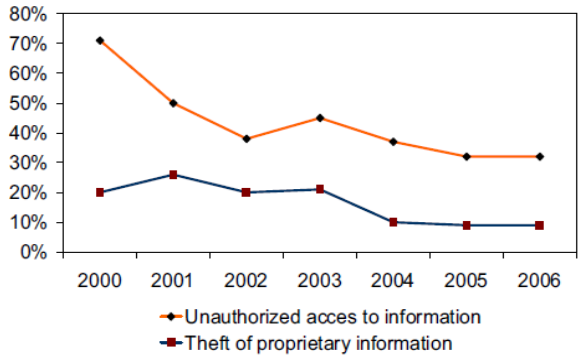

Figure 1: CSI/FBI Computer Crime Security Survey Data[1]

2. LITRETURE SURVEY

In the year of 2000, S. Khanna proposed the idea of controlling the security of database with digital watermark which arouses the researchers' interests on watermarking database[3]. For embedding images there are two major methods. First method was proposed in 2004 by Zhang[2] . This method uses the order of the marked data for embedding image. Due to the fact that the pixels of image have relative position, and they are sure to be placed in order, this method is vulnerable to conflicting order of embedded marks by some subset attacks. In 2008 Sun introduced another robust technique for embedding images into the database. This method transforms some meaningful images as the copyright information into a bit flow. In this method unlike other algorithms, location of each mark bit (pixel) is calculated according to the hash value that is related to one of the database tuples. Hence this technique can resist against mark out of ordering. On the other hand as this algorithm uses the remainder of hash value modulo length of watermark, to calculate and find specific mark bit location, the algorithm will have a problem if length of watermark (image) increases, or number of database tuples decreases. So this method is vulnerable in placing all of the mark (image) bits and the probability of missing mark bits is notable, especially in case of embedding big size image into the small size database.

Watermarking relational databases is a relatively new research area that deals with the legal issue of copyright protection of relation databases. Therefore, literature in this area has been very limited, and focused on embedding short strings of binary bits in randomly selected bit locations in numerical databases. In fact, most bit-level database watermarking algorithms suffer a lack of robustness against bit-level attacks such as bit-setting, bit-resetting and bit-flipping. Other database watermarking algorithms embed watermark information in the statistical properties of the database rather than in the relational data itself.

2.1 Applications of Digital Watermark for Relational Databases

2.1.1 Ownership Assertion:

Watermarks can be used for ownership assertion. To assert ownership of a relational database, watermark can embed a into database *R* using some private parameters (*e.g.* secret key) which is known only to user. Then the watermarked database can make publicly available. Later, suppose any suspects that the relation *S* published by some has been pirated from the relation *R*. The set of tuples and attributes in *S* can be a subset of *R*. To defeat ownership claiming, one can demonstrate the presence of her watermark in its relation. For such a scheme to work, the watermark has to survive intentional or unintentional data processing operations which may remove or modify the watermark.

2.1.2 Fingerprinting:

Fingerprinting aims to identify a traitor. In the applications where database content is publicly available over a network, the content owner would like to discourage unauthorized duplication and distribution by embedding a distinct watermark (or fingerprint) in each copy of the database content. If, at a later point in time, unauthorized copies of the database are found, then the origin of the copy can be determined by retrieving the fingerprint.

2.1.3 Fraud and Tamper Detection:

When database content is used for very critical applications such as commercial transactions or medical applications, it is important to ensure that the content was originated from a specific source and that it had not been changed, manipulated or falsified. This can be achieved by embedding a watermark in the underlying data of the database. Subsequently, when the database is checked, the watermark is extracted.

2.2 Types of Database Watermarking

Need for watermarking database relations to refer their piracy, identify the unique characteristics of relational data which pose new challenges for watermarking, and provide desirable properties of a watermarking system for relational data. Proving ownership rights on outsourced relational databases is a crucial issue in today's internet-based application environments and in many content distribution applications.

2.2.1. Distortion base Watermarking:

The watermarking techniques in this category introduce small changes in the underlying data of the database during embedding phase. The degree of changes should be such that any changes made in the data are tolerable and should not make the data useless. The watermarking can be performed at bit level, or character level, or higher such as attribute or tuple level, over the attribute values of type's numeric, string, categorical, or any[5].

2.2.2 Distortion frees Watermarking:

Most of the distortion free watermarking techniques are fragile in the sense that in addition to the ownership claiming, they aim at maintaining the integrity of the information in the database. The watermark insertion phase does not depend on any specific type of attribute and does not introduce any distortion in the underlying data of the database.

2.3 Cryptography Overview:

Modern cryptographic mechanisms are mainly based on different, unproven assumptions, concerning easy and hard computation of functions. Cryptography, then, not only protects data from theft or alteration, but can also be used for user authentication. Three types of cryptographic schemes typically used to accomplish these goals: secret key (or symmetric) cryptography, public-key (or asymmetric) cryptography, and hash functions, each of which is described below.

- **Secret Key Cryptography (SKC):** Uses a single key for both encryption and decryption Examples: Data Encryption Standard (DES), Advanced Encryption Standard (AES), Rivest Ciphers, Seeded.

- **Public Key Cryptography (PKC):**

Uses one key for encryption and another for decryption . Examples: RSA,Diffie hellman, ElGamal

- **Hash Functions:**

Uses a mathematical transformation to irreversibly "encrypt" information Examples: Message Digest (MD) algorithms, hash of variable length,whirpool etc.

2.4 Different Types of Attacks:

The digital watermarking for integrity verification is called fragile watermarking as compared to robust watermarking for copyright protection. In a robust watermarking scheme, the embedded watermark should be robust against various attacks which aim at removing or distorting the watermark. While in a fragile watermarking scheme, the embedded watermark should be fragile to modifications so as to detect and localize any modification in presence of different attacks[5].The

watermarked database may suffer from various types of intentional and unintentional attacks which may damage or erase the watermark, as described below:

2.4.1 Benign Update:

In this case, the tuples or data of any watermarked relation are processed as usual. As a result, the marked tuples may be added, deleted or updated which may remove the embedded watermark or may cause the embedded watermark undetectable . This type of processing are performed unintentionally[5].

2.4.2 Value Modification Attack:

Bit Attack: This attack attempts to destroy the watermark by altering one or more bits in the watermarked data. More information about the marked bit position makes attack more successful. However, in this case usefulness of data is crucial: more alternation may result the data completely useless.

Rounding Attack: Mallory may try to lose the marks contained in a numeric attribute by rounding all the values of the attribute. Success of this attack depends on the estimation of how many bit positions are involved in the watermarking. Underestimation of it may cause the attack unsuccessful, whereas overestimation may cause the data useless[5].

2.4.3 Subset Attack:

Consider a subset of the tuples or attributes of a watermarked relation and by attacking (deleting or updating) on them he may hope that the watermark has been lost[5].

2.4.4Superset Attack:

Some new tuples or attributes are added to a watermarked database which can affect the correct detection of the watermark.

2.4.5 Subset Reverse Order Attack:

Attacker enjoys this attack by exchanging the order or positions of the tuples or attributes in relation which may erase or disturb the watermark[5].

Table 1: Comparison between Distortion base watermarking[5]:

Proposed Schemes	Watermark Information	Cover Type	Granularity Level	Verifiability	Intent
AHK Algorithm	Meaningless bit pattern	Numeric	Bit level	Blind, private	Ownership proof
Gupta et. Algorithm	Meaningless bit pattern	Numeric	Multi bit level	Blind , reversible, private	Ownership proof
Image Based watermarking	Image	Numeric or non numeric multi word	Bit level ,whole attribute value or char level	Blind private	Ownership proof and temper detection
Speech based	Owners speech	Numeric	Bit level	Blind , private	Ownership proof
Content based	Database content	Numeric	Multi bit level	Blind ,private	Ownership proof and/or temper detection and localization
Cloud model based	Cloud model with three characteristics Expected value ,entropy, hyper entropy	Numeric	Whole attribute value	Non blind private	Ownership proof
Categorical attribute based	Meaning full binary string	Categorical	Bit level	Blind and private	Ownership proof
Fake tuple based	Fake information obtain from the database contain	Database table	Tuple level	Blind private	Ownership proof
Virtual attribute based	Database contain	Database table	Attribute level	Blind , deterministic, private	Temper detection
Others	Meaningful information of any type	Numeric	Bit level	Blind or non blind or private	Ownership proof
Fingerprinting techniques	Meaningful fingerprint identifying buyers uniquely	Numeric	Bit level	Blind private	Trailer detection

Table 2: Comparison between distortion free watermarking [5]:

Proposed Schemes	Watermark Information	Cover Type	Granularity Level	Verifiability	Intent
Permutation Based	A part of group level Hash value	Tuples positions	Tuple level	Blind private	Temper detection
Characteristics based	White image with owner's mark at four corner	Nil	Nil	Blind public	Temper detection
Binary form relation	Relation in binary form	Nil	Nil	Blind public or private	Ownership proof or temper detection
R tree based scheme	Numeric value identifying owner	Order of entries in r tree nodes	R tree nodes	Blind private	Temper detection

3. DESIGN

Data flow diagram

3.1 Data Flow Diagram –Level 0:

The level 0 is the initial level Data flow diagram and it's generally called as the context level diagram. . It is common practice for a designer to draw a context-level DFD first which shows the interaction between the system and outside entities. This context-level DFD is then exploded to show more detail of the system being modeled.

The Fig shows the Level 0 data flow diagram. In this level 0 DFD, three external entities have been identified Encrypter who encrypt the given bit image ,Decrypter who identify the given embedded given image & Database.

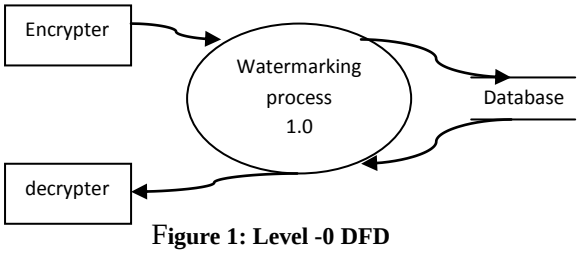

Figure 1: Level -0 DFD

3.1.1 Data flow diagram for Image embedding in the database :

The Level 1 DFD gives more information than the level 0 DFD. Basically there two main process in the database watermarking. First is image embedding into the database and second is image extraction from the database.
This diagram shows the processes contained in to the image embedding phase .Using external entity converter , image is converted into the image raw bits. And the second process shows the image raw bits is encrypted where external entity is encrypter is connected with this process. After in the third process encrypted image raw bits are embedded into the database.

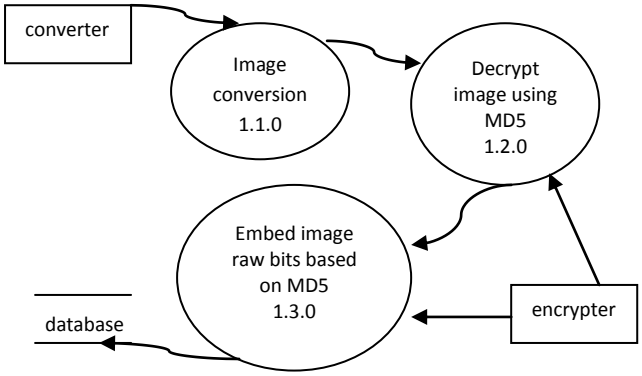

Figure 2: Level -1 DFD

3.1.2 Data flow diagram for watermark detection:

This dataflow diagram is related to process of image extraction from the database. First process is showing that extract the image from the database. After in the decryption process the decrypt will decrypt the image using the private key. Third process is detection of the image base on the raw bits identification.

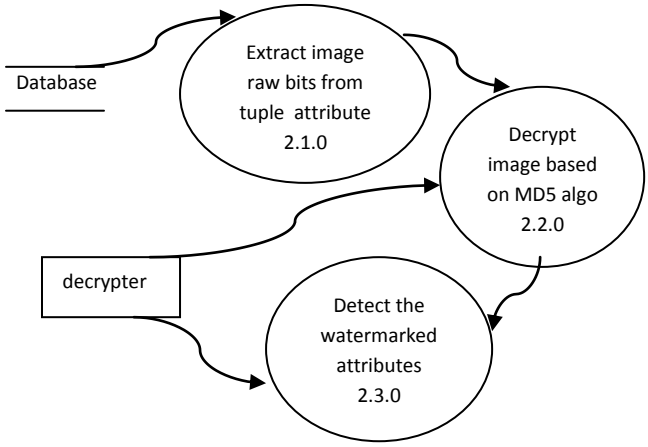

Figure 3: Level -1 DFD

3.2 Class Diagram

The class diagram is the main building block of object oriented modeling. It is used both for general conceptual modeling of the systematic of the application, and for detailed modeling translating the models into programming code.

3.2.1 Watermark Insertion:

In this section of the class diagram is related to encryption process. Main class will contain below methods which method is related to which functionality.
Main(): method will initialize the main application
getMD5(): to generate hash value using secret key this encryption method is used
GetConnection(): method will connect the database with application
Concate(): method will concatenate the private key with raw bits for encryption
Image(): this method will get the image
Connectionclose():method will close the connection with the data base
getheight( ): return height of the image
getwidth( ): return width if the image

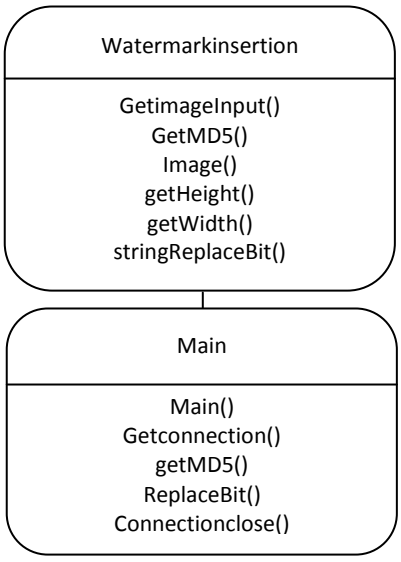

Figure 4: Class Diagram for watermark insertion

3.3 Watermark Detection

In this section of the class diagram is related to decryption process. Detect Main class will contained below methods which method is related to which functionality.

main():initializing method for application
getconnection():method will connect the database with application
image():method will get the image
connectionClose( ):method will close the connection with the database
GetBinaryImage( ):get the binary raw bits of image from the database
compareImage( ):compare the image with original image
getheight( ):return height of the image
getwidth( ):return width if the image
Comparebit( ): compare the extracted raw bits with the original raw bits
Replacebit ( ):replace the modified bit with original bit

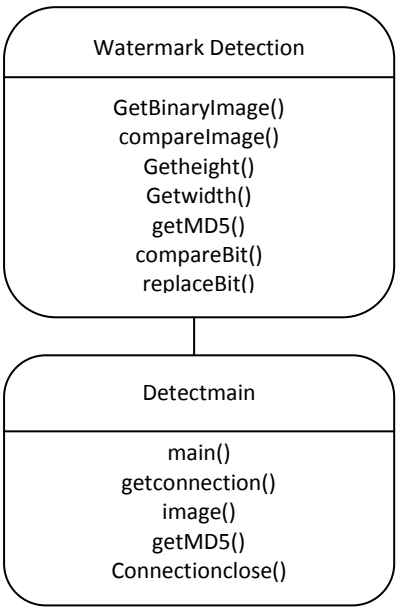

Figure 5: Class Diagram for watermark detection

5. Implementation

5.1 Modules for database watermarking:

Basically in this project there are two modules and they are

1) watermark insertion into the database
2) watermark detection into the database

For both classes method description is as bellow tables:

5.2 Pseudo code:

5.2.1 For insertion Watermark:

1) Take a path for a given database to be watermarked
2) Based on database count no of tuples & no of columns to be watermarked
3) Obtain *secret key* Sk for MD5 hashing
4) For MD5 hashing obtain primary key P and concatenate with secret key Sk
5) Using MD5 algorithm obtain hash value using combination of P+Sk
6) Mark the tuple based on the hash value *If(H_hashvalue mod F= 0)* then mark the tuple
   Where F= database partitioning value which will be private to the database owner
7) Obtain ***attribute indexA*** based on (*H_hashvalue mod V =0*)
   Where V =No of columns of the database
8) Obtain ***bit indexB*** which will mark particular bit of attribute based on(*H_hashvalue mod ξ*= 0) where *ξ=private to the owner*
9) Select the image bit procedure for the given image bases on hash value generated by the MD5
   - Select row of image based on (*A*V*) mod *H where* H = image height
   - Select column of image based on *H_hashvalue mod W where* W = image width
   - Convert row value into decimal value
   - Convert column value into decimal value
   - Obtain height & width for marked bit from the image based on hashing algo MD5 concatenated value of row
   - Select bit of image for marking based on marked bits
10) Select field from database based on hash function
11) Covert attribute field into the binary field
12) Replace in LSB of binary value of attribute with selected image bit
13) Replace new attribute value in the database

5.2.2For watermark Detection:

1) Take a path for a given database
2) Initially totalcount=matchcount=0
3) Obtain MD5 hashing obtain hash value using P+Sk
4) Detect the tuple based on the following condition

- IF (*H_hashvalues mod F*=0)then mark the tuple
  Where F=data partitioning value which will be private to the owner
- Obtain *attribute index i* = (H_hashvalue mod V=0) where V =No of columns of the database
- Obtain *bit index B* which will mark particular bit of the attribute based on (*H_hashvalue mod ξ*=0) where *ξ*=private to owner
- If(bit index B<=length of selected field)
  then
  totalcount++
  matchcount=matchcount+match(selected field ,bit index, selected bit)

6. RESULT & DISCUSSIONS

6.1 Results Of Diff Attacks Tested on the local Database:

6.1.1 Attribute Addition attack:

In this type of attack randomly attributes has been added in to the attribute subset of the tuples. Here in this graph it has been shown that even if also attributes added equal to the total attributes subset of the database then also watermark will not be removed using given watermarking scheme.

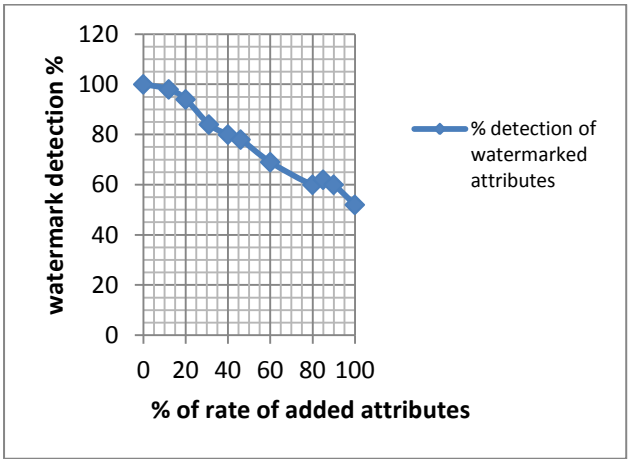

Figure 6:Graph showing attributes addition attack

6.1.2Attribute deletion Attack:

In the given bellow figure it has been shown that eventually when attributes has been removed according to that watermark has been removed. So up to 94% attributes has been deleted watermark has been detected .But when 100 % attribute has been removed it full watermark is removed from the database.

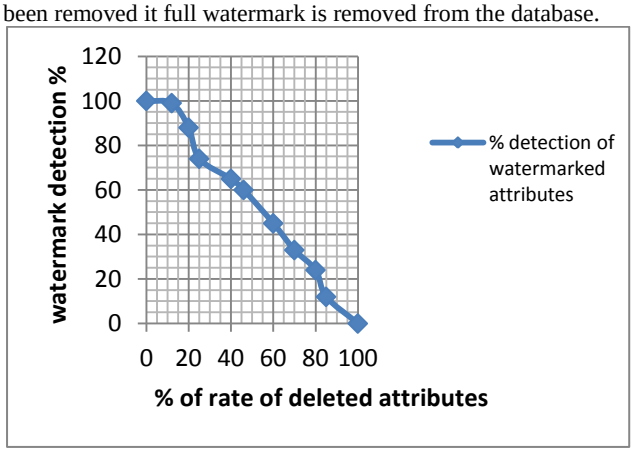

Figure 7: Graph showing Attribute deletion attack

6.1.3Subset Reverse Order Attack:

In this attack in the subset of the attributes randomly chosen and after that position of the attribute are randomly disturb or changed. According to that in the given bellow graph up to 100 % reverse ordering of the attribute is done then also 64% of watermark has been detected from the given subset of the database.

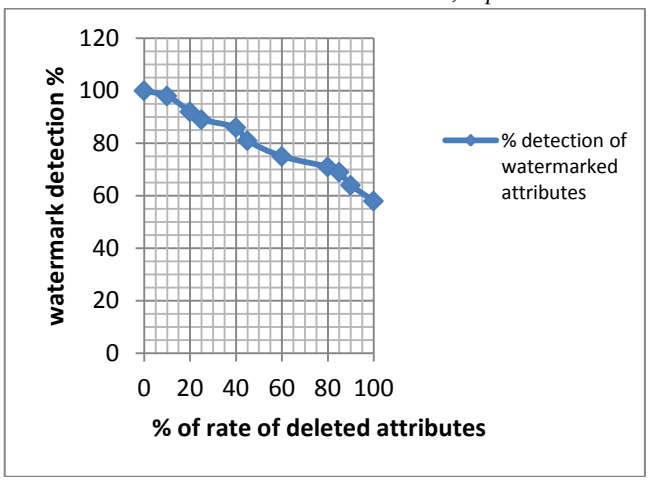

Figure 8: Graph Showing Subset reverse order attack

6.2 Screenshots

6.2.1 Watermark insertion phase:

```
selected_field of Original database : 90
Binary of this field : 1011010
length of binary:7
0
selected bit from selected field 0
```

Figure 9: For each and every tuple it based on MD5 generate hash value and calculate attribute index and bit index for each tuple

```
k_cvalue:67108863
k_rvalue:2147483647
```

Figure 10: Column and Row value of image

```
selected_field of Original database : 90
Binary of this field : 1011010
length of binary:7
0
selected bit from selected field 0
```

Figure 10: Based on MD5 hashing it will select randomly attribute from the database

```
selected bit from selected field 0
this bit is updated with selected bit of an image :1
replaced field without minimize variation method : 91
After mv, Replaced attribute in binary : 1011011
After mv, Replaced attribute in decimal : 91
```

Figure 11:For watermarking it will replace LSB with the selected bit of the image and generate new value

```
stock
33
UPDATE watermarking.sheet1 SET stock = null WHERE stock = null
UPDATE watermarking.sheet1 SET stock = '33' WHERE stock = '32'
```

Figure 12:Same value is replaced in the database in column stock

```
for tuple no. 2:
value:27086490266497182828000697388348595341 8
tfmod:4
Value of J is: 0
```

Figure 13:For next tuple it will check tuple attribute value and here it is not 0 so it will not mark the tuple

6.2.2 Watermark detection phase:

```
selected bit of an image: 1
selected field of Watermarked database : 91
Binary of this field : 1011011
length of binary:7
1
selected bit of an image:1
selected bit of database:1

Both bits are matched
```

Figure 14:based on selected bit of the image compare a bit of the watermarked attribute

```
Total_count : 16
Match_count : 16
```

Figure 15:count the total no of watermarked attributes and matched attributes

7. CONCLUSION & FUTURE ENHANCEMENT

7.1 Conclusion

In the detection phase of the watermarking it is partition based, we are able to detect and locate watermarked tuple attributes as we watermarked in the insertion phase neither watermark generation nor detection depends on any correlation or costly sorting among data items. Each tuple in a table is independently processed; therefore, the scheme is particularly efficient for tuple oriented database operations. This watermarking scheme is robust for two attacks which are attribute addition attacks and subset reverse order attack .and for that result has been shown in the graphs.

7.2 Future Enhancement

For hashing to watermark the database MD5 algorithm is used with concatenation of private key of the relational database instead of that AES algorithm can be used. This scheme is applied on the numeric type attributes only to achieve minimize variation on the attribute .So this scheme can be extended for categorical type attributes also. Here watermarking is done using image and this type of watermarking scheme is distortion base so for the same scheme can be extended for distortion free watermarking also.

8. ACKNOWLEDGEMENT

My guide Mrs. Mamatha H.R. , Associate Professor, IS&E DEPT., PESIT, for the constant help and support extended towards me during this paper work. I would like to take this opportunity to thank her

9. REFERENCES

[1] Andriy Furmanyuk, Mykola Karpinskyy, Bohdan Borowik," Modern Approaches to the Database Protection", IEEE International Workshop on Intelligent Data Acquisition and Advanced Computing Systems" 2007

[2] Zhi-Hao Zhang , Xiao-Ming jin, jain-Min wan , " Watermarking Relational Database Using Image ", IEEE, Proceedings of the Third International Conference on Machine Learning and Cybernetics, Shanghai, 26-29 August 2004

[3] Sanjeev Khanna, Francis Zane." Watermarking maps: hiding information in structured Data".Int'l Conf. SODA 2000, San Francisco, California, USA

[4] R. Barnett , "Digital watermarking: applications, techniques and challenges", Electronics & Communication Engineering Journal AUGUST 1999

[5] Raju Halder, Shantanu Pal , Agostino Cortesi,"Watermarking Techniques for Relational Databases:Survey", Classification and Comparison, Journal of Universal Computer Science, vol. 16, no. 21, 2010

[6] Radu Sion, "Database Watermarking for Copyright Protection, Network Security and Applied Cryptography Lab Computer Science", Stony Brook University

[7] Hossein Moradian Sardroudi, Subariah Ibrahim," A New Approach for Relational Database Watermarking Using Image," Universiti Teknologi Malaysia (UTM)",2006

[8] Yingjiu Li, Robert Huijie Deng, "Publicly Verifiable Ownership Protection for Relational Databases", ACM 1-59593-272-0/06/0003, ASIACCS'06, March 21-24, 2006

[9] Hamed khataeimaragheh and Hassan Rashidi," A Novel Watermarking scheme for detecting and recovering distortions in database tables", International Journal of Database Management Systems (IJDMS) Vol.2, No.3, August 2010

[10] Xiaomei Dong , Xiaohua Li, Ge Yu, Lei Zheng, An Algorithm Resistive to Invertibility Attack in Watermarking Relational Databases, 2009 Chinese Control and Decision Conference