A New Technique for Relational Database Protection

Nahla El_Haggar
Faculty of Computers & Information
Helwan University, Egypt

Mahmoud M. El-khouly
Faculty of Computers & Information
Helwan University, Egypt

Samah S. Abu El Alla
Faculty of Computers & Information
Helwan University, Egypt

ABSTRACT
The great development of computer technologies and the Internet have made duplication and distribution of digital information simpler. This leads to a need for effective copyright protection tools. Watermarking database system is considered a vital technique for copyright protection of database systems. In this paper the proposed system is a robust technique to embed and detect watermark in a relational database. In the embedding watermarking stage the watermark is embedded in non numeric attributes for preserving the query results, then is compressed the database for increasing the transfer rate. In watermark detection stage the database will be decompressed, and then the distortion of the embedding watermark will be checked to identify pirated copies of original data. The proposed technique is considered a fully blind and has robustness against various types of attacks specially deletion and sub selection attack.

General Terms
Security, Relational Database, Watermarking.

Keywords
Relational database system; copyright protection; watermarking; non numeric attribute; database attacks

1. INTRODUCTION
Nowadays, the using of relational database systems is increased in many real-life applications; copy relational database and redistribute illegal copies become very easy. Therefore, copyright protection of database systems becomes very important research area. It identifies pirated copies of original data. It doesn't prevent copying, but it deters illegal copying by providing a means of establishing the ownership of a redistributed copy [8]. A watermark can be applied to any relational database having attributes which changes in their values will not affect the applications [1]. Databases have different types of attributes like, numeric, text, "date and time", etc. The proposed scheme applied for non-numeric attributes to preserve the value of attributes in database. Generally, database watermarking techniques consists of two main phases: watermark insertion and watermark detection as shown in figure 1 and figure 2.

This paper is organized as follow: in section 2; related work is presented, in section 3; overview and flow charts of proposed technique is described in details, in section 4; some important features of the proposed scheme related to security, transfer rate, blind detection and query preserving watermark are discussed, in section 5; analysis and the performance of the proposed system are evaluated with reference to different attacks and in Section 6; conclusion and future work.

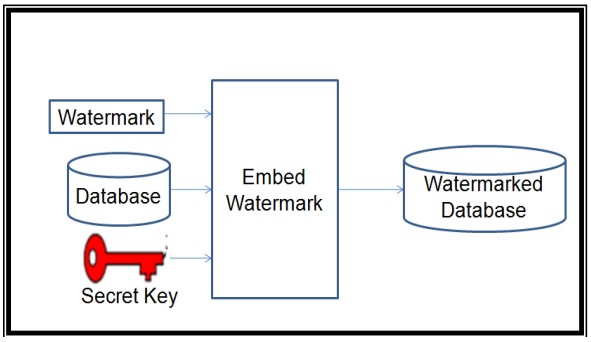

Figure 1: Digital watermarking –insertion

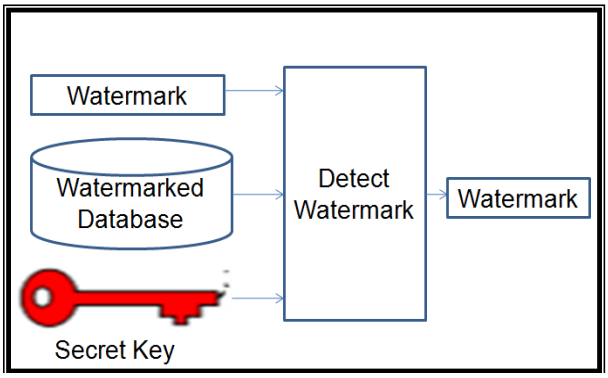

Figure 2: Digital watermarking –detection

2. RELATED WORK
Shah et al. [3], suggested a scheme to embed watermark in the alphabetic data attributes. This method is applicable to all the languages with upper/lower character cases. The main advantage of this technique is that it doesn't affect the semantic meaning of data if the case is changed from small to capital or vice versa. So the query result doesn't change.

Ali et al. [4], proposed technique based on using binary image to watermark the relational database. The bits of the image are segmented into short strings that are encoded in non-numeric, multi-word attributes of selected tuples of the database. A major advantage of using the space-based watermarking is the large bit-capacity available for hiding the watermark.

Rajneesh et al. [5], proposed a secure method which uses both semantic and syntactic techniques to watermark the tuple in a relation. The Watermarking technique is dependent on secret key and on the relation. The proposed algorithm is based on the concept of predefined signals of ASCII characters. A secret key is generated by using these signals only. To embed a watermark they used the concept of abbreviations for words and also one of syntactic approach.

Bedi et al. [6], proposed watermarking technique used for data authentication and integrity of relational database. For integrity verification of tables in the database, the watermark is depending on a secret key and the original copy of relation. This method used the concept of eigen values to generate the watermark for a record in tuple. Watermark embedding is done by using eigen values in a non-numeric attribute of a tuple. Detection of the watermark proves authenticate and integrities of data.

Pramod et al. [7], presented new scheme for watermarking non-numeric relational databases. This technique uses voice of a copyright holder to generate watermark, then corresponding insertion and detection algorithm had been applied.

3. PROPOSED TECHNIQUE

The main goal is to design a technique which has less transfer time, fully blind, robustness, and reliability. The following table (1) shows the symbols that used in the proposed technique.

Table 1: Symbols used in the proposed technique

Notation	Description
R	Relation to be marked
r	record of a relation
K	Secret Key known only to owner
r.p	Primary Key
Vp	virtual primary key
V	Number of attributes in the relation available for marking
Y	Used to determine the number of tuples to be marked. If w denotes number of tuples to be marked, then w=n/y
1/y	Fraction of tuples marked
r.Ai	Attribute Value
totalcount	Number of tuples are tested
matchcount	Number of tuples contain the expected value
α	Significance level of the test for detecting a watermark
Ґ	Minimum number of correctly marked tuples needed for detection
S	Suspected relational database
Attribute marked i	Index of selected attribute that will be marked
N	Number of tuples in relation

3.1 The watermarking insertion stage

The following flowchart illustrates insertion stage.as shown in figure 3.

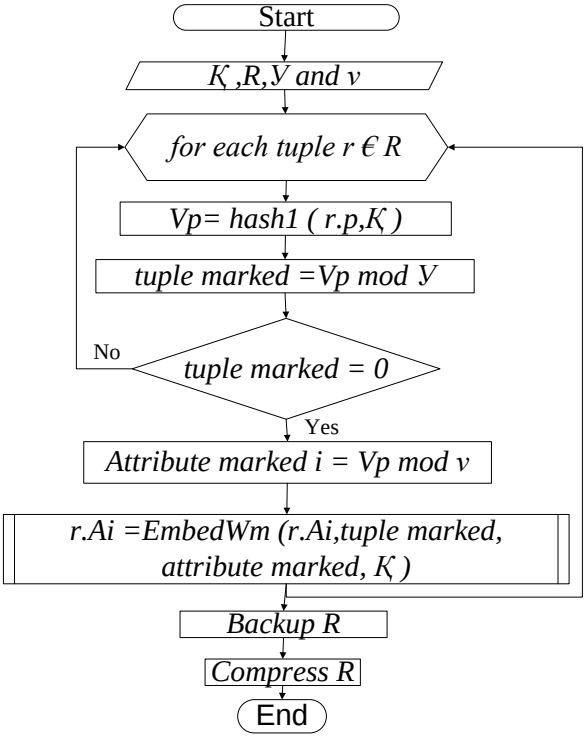

Figure 3: Watermarking insertion stage

In this stage, the watermark is embedded in non numeric attributes to preserving the query results then compress the database for increasing the transfer rate.
The watermark insertion stage composed of the following steps:

Step 1: Generate virtual key

To get the virtual key the following equation has been used

$$hash1\ (r.p, K) = H\ (K\ \&H(r.p\&\ K)) \qquad (1)$$

Where:

- hash1 is defined as a message authentication code which is a one-way hash function H that depends on a key. It operates on an input message (M) of arbitrary length and returns a fixed length hash value h= H (M).
- & represents concatenation.

Step 2: Generate marked tuples index and marked attribute

The algorithm generates marked tuple index for insert watermark in its selected attributes by using the following equation

$$tuple\ marked = Vp \bmod Y \qquad (2)$$

Where mod operation had been applied on *Vp* and y then, marked tuple had been checked if it is equal to zero or not. If it is not equal to zero anther tuple will be chosen and then mod operation will be applied again. If it is equal to zero, the following equation will be applied to get marked attribute.

$$Attribute\ marked\ i = Vp \bmod v \qquad (3)$$

Where mod operation had been applied on vp and v to get index marked attribute.

Step 3: Insertion watermark

In this step the algorithm gets the value of attribute after insert watermarking by using the following function *r.Ai =EmbedWm (r.Ai,tuple marked, attribute marked, K)*

(4)

Where : "EmbedWm" is insertion watermark function

This function has two main rules.

a) Generate watermark the algorithm uses the following equation to generate watermark

$$hash2(V, K) = H(K\&H(H(K) \text{ ¤ } H(V)) \quad (5)$$

Where:

hash2 a one-way hash function
¤ represents XOR operation.
V is the variable.

In The proposed system "V" is considered as marked tuple to get "T1". And it is considered as marked attribute to get "T2", where:

$$T1 = hash2(tuple\ marked, K) \quad (6)$$

$$T2 = hash2(attribute\ marked, K) \quad (7)$$

Watermark "T" is the value of 160 digit generating by XOR operation between T1 and T2.

b) Change case of selected attribute depending on value of 1th bit position of T.

Step 4: Backup the file and compress it.

3.2 The watermarking Detection

The following flowchart illustrates detection watermarking stage.as shown in figure 4.

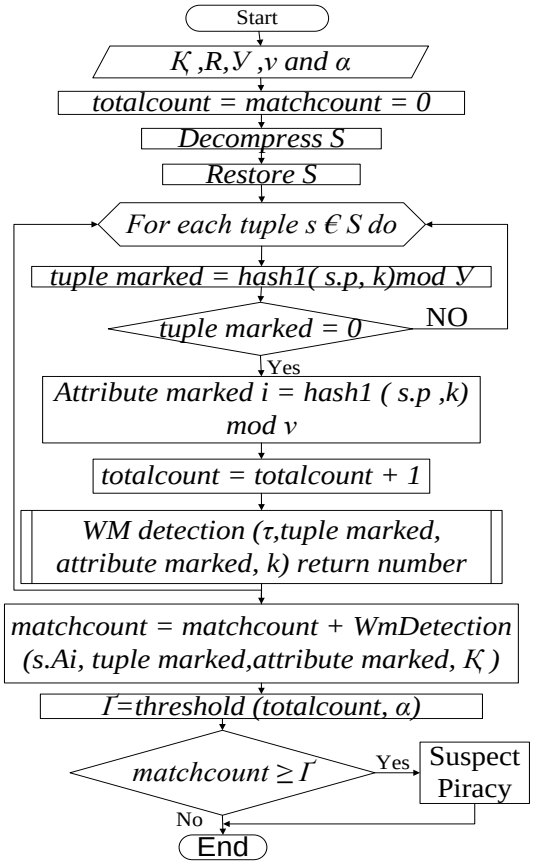

Figure 4: Watermarking detection flowchart

The watermark detection algorithm has the following steps:

Step 1: Restore and Decompress
Get the original database.

Step 2: Generate marked tuples index and marked attribute.
the algorithm gets selected attribute as explained in insertion stage. then increase (totalcount) by one.

Step 3: Detection the embedded watermark
This operation is done by a function called "WmDetection"

$$WM\ detection\ (\tau, tuple\ marked, attribute\ marked, k)\ return\ number \quad (8)$$

This function has two main steps:

a) **Generate watermark** as explained in insertion stage.
b) **Compare attributes case:** the function "WmDetection" compares the current attribute case with the case that must have been set for that attribute by the watermark insertion algorithm. If it returns "1" then "matchcount" will increase by one, as shown below:

$$matchcount = matchcount + WmDetection\ (s.Ai, tuple\ marked, attribute\ marked, K) \quad (9)$$

Step 4: repeat the previous steps for each tuple in relational database.

Step 5 check pirated copy
In this step, the algorithm checks if this database is pirated or not by getting the result of threshold function as follow:

$$\Gamma = threshold\ (totalcount, \alpha) \quad (10)$$

Then compare (matchcount) with the minimum count returned by the threshold function for the test to succeed at the chosen level of significance α. More details can be found in the paper [8].

4. DISCUSSION

In this section, some important characteristics of the proposed approach such as security, transfer rate, detection and query reservation had been discussed

4.1 Security

The secret key K only known by the owner has been used. Selection of tuples and attributes is based on K. Also, using a secure one-way hash function (SHA-1) for selecting marked attribute and generating watermark makes this scheme more secure.

4.2 Transfer rate

In earlier existing systems, transferring the watermarked relational database to the client had been sending without compressing its size. Therefore, the speed of transfer rate between client and server takes longer time. To overcome this limitation database will be compressed by using compression technique which will provide security as well as, increase the transfer rate [9]. The proposed technique reduces database size till 10% of its original size.

4.3 Blind detection

The proposed technique is fully blind since watermark detection should neither require the knowledge of the original database tables nor the watermark itself.

4.4 Query preserving watermark

Watermark embedding is done by changing the case of selected attribute according to algorithmic rules, which does not change the value of attribute, so the result of the queries will not be changed after embedding

5. ANALYSIS

This section analyzes the robustness of proposed system against various types of attacks. Experiments showed that, it is robust against deletion attacks. , subset selection attack, subset insertion attack, subset modification attack and finally subset case alteration attack. It had being tested by using SQL server database of 100,000 tuples on Windows 7 platform.

5.1. Detectability

The detectability of a watermark in the proposed system depends on the significance level(α),the number of marked tuples(w),the number of tuples in the relation(N) and the fraction of tuples marked (1/y).

5.1.1. Effect of "fraction, total number of tuples" in detectability

Figure 4 shows the proportion of marked tuples that must have the correct watermark value for successful detection (i.e. Ґ/ w). the results for relations of three different sizes (1182,10000,108800) tuples have been plotted.

Assuming that, α = 0.01 and V=3. The X-axis varies the percentage of tuples marked (i.e. the fraction 1/y expressed as percentage). The percentage of tuples marked 0.1%, 1.0%, and 10% correspond to the y values of 1000, 100, and 10 respectively. Figure 4, shows that the required proportion of the percentage of marked tuples increases as correctly marked tuples decreases, correctly marked tuples decreases as the number of tuples in the relation increases. And in large database even small fraction of tuples is chosen to be marked watermark can be detected.

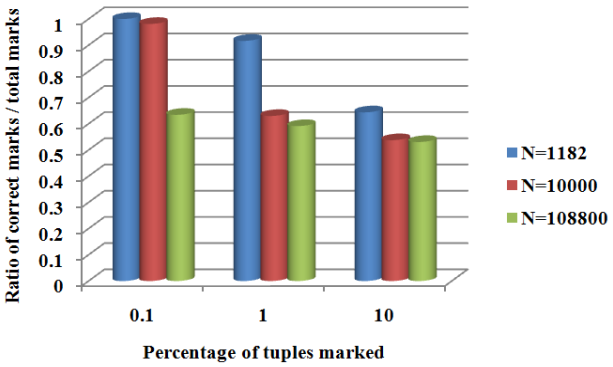

Figure 5: Proportion of correctly marked tuples needed for detectability

5.1.2. Effect of "Significance level of the test for detecting watermark" in detectability

Figure 6, shows the required proportion of correctly marked tuples for various values of α. The results are shown for 1182 tuples relation, V=3 and marked tuples =132. Clearly, it is possible to detect the watermark even for very low values of α,

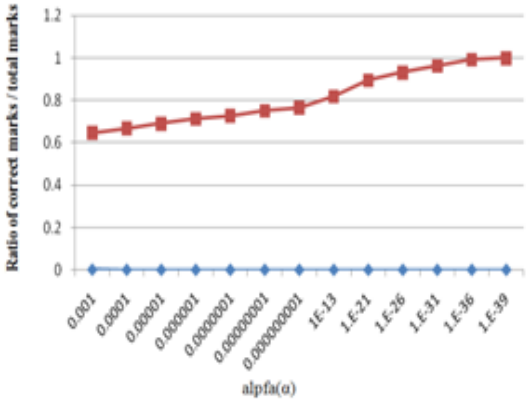

Figure 6: Proportion of correctly marked tuples needed for decreasing α.

5.2. Robustness

Robustness is a very important issue as it proves ownership of the data. Users have to detect their watermark in data without any damage or defect, i.e. the rate of correctly detection of watermark is also called robustness of the watermark [10]. Next subsections will explain the advantages and disadvantages of the proposed system against various forms of malicious attacks.

5.2.1. Advantage of proposed system

5.2.1.1. Preventing Subset Selection attack

The attacker can randomly select and use a subset of the original data set that might still provide value for the intended purpose. The attacker hopes that, the selected subset will not contain the watermark. However, the proposed algorithm embeds the watermark in the whole database randomly and depending on hash function, so it is difficult to guess where watermark is embedded. Also, the detection algorithm checks watermarked tuple by the same way of embedding, so the detection algorithm scans the primary key of subset selection attributes. This kind of attack doesn't change the primary key or the case of attribute, therefore even if attacker attacks only 10% of original database, the proposed system will detect watermark with 100%.

5.2.1.2. Preventing Deletion attack

In this type of attack, the attacker may delete a subset of the tuples of the watermarked database hoping that the watermark will be removed. the detection algorithm scans marked tuple by primary key, therefore if an attacker deletes tuple, then the detection algorithm doesn't effect because detection algorithm increases (totalcount) and (matchcount), if and only if detect marked tuple which still has marked. This means that Watermark can't completely remove from all the tuples because if only 5% tuples are there. Then a watermark can be extracted from the database. As a result the proposed system is considered as very robust against any kind of attacks that did not change the case of attributes or primary key.

5.2.2. Disadvantage of proposed system

5.2.2.1. Subset insertion attack

In this attack, an attacker might insert a set of tuple in the original database. The system can be executed with various percentages of insertion. These experiments run on N=13270, fraction =10, v=3 and α =0.01. The system had been run twice at each percentage then the results each time are recorded, then the maximum percentage of detection had been taken.

Figure 7 indicates that the watermark distortion is 46% even if 100 % of the tuples of the database were inserted. This is due to the fact that the proposed algorithm embeds the watermark everywhere in the database, making this type of attack ineffective.

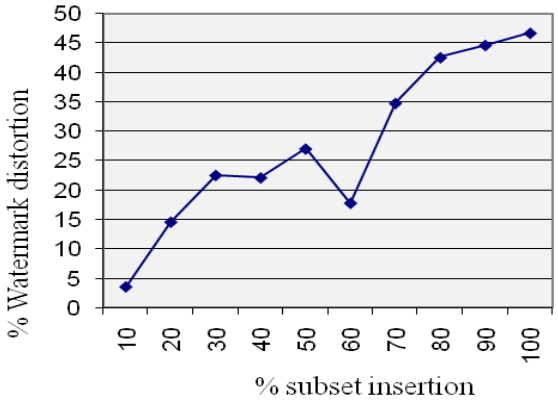

Figure 7: loss in watermark after insertion attack

5.2.2.2. Case Alteration attack

This attack can be considering special type of modification attack. This experiment run on N=13270, fraction =10, v=3and α =0.01. The case of attributes values are randomly and repeatedly changed, detect watermark, and get the percentage of loss. After that the percentage of loss in watermark and percentage of case alteration is plotted as shown in figure 8. It is observed that the scheme is robust against this kind of attack. After changing the case of up to 60% tuples, the watermark distortion is less than 20.6%.

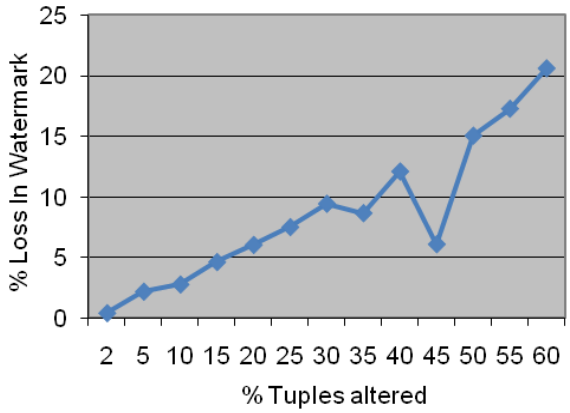

Figure 8: Loss in Watermark Detection after Change Case

5.2.2.3. Modification attack

In this type of attack, the attacker modifies the tuples of the database randomly. Attacker hopes by doing so to erase the watermark from the database. The system is executed with various percentages of modification twice the results are recorded each time then take maximum percentage of detection. The graph shown in Figure 9 indicates that the watermark distortion is 12.6% even if 34% of the tuples of the database were modified. This experiment run on N=13270, fraction =10, v=3 and α =0.01.

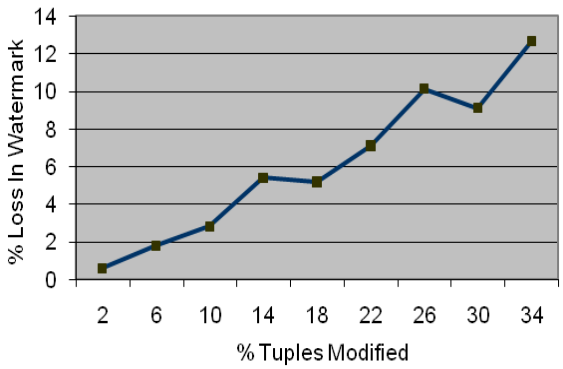

Figure 9: Loss In Watermark Detection After Modifications

this result can be improved by changing *α or (*1/y) as shown in figure 10.

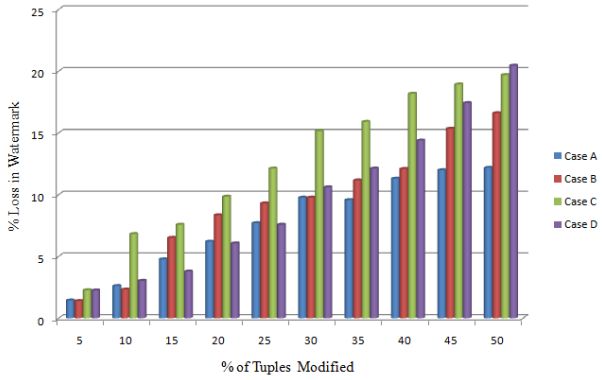

Figure 10: effect of detectability paramter on modification attack

The figure shows the precentage of loss in watermark in modification attack at different four cases this cases depend on choosing of owner's paramters.

- Case A owner choose α=0.01 and (1/y) =1. In this case the watermark will remain with 87.82% even if 50% of the tuples of the database were modified.
- Case B owner choose α=0.01 and (1/y) =2. In this case the watermark will remain with 83.39535% even if 50% of the tuples of the database were modified.
- Case C owner choose α=0.01 and (1/y) =10. In this case the watermark will remain with 80.303% even if 50% of the tuples of the database were modified.
- Case D owner choose α=0.001 and (1/y) =10. In this case the watermark will remain with 79.545% even if 50% of the tuples of the database were modified.

These results proved that the system is robust against modification attack.

5.3. Design Trade-Offs

The watermarking approach has three important valuable parameters:

- the test significance level
- the gap parameter that determines the fraction of tuples marked
- the number of attributes in the relation available for marking

Based on the analysis presented in this section, the following table is summarized the analysis.

α ↓	Missed Watermark ↑
y ↓	Robustness ↑
v ↑	Robustness ↑

6. CONCLUSION

In this paper, a new technique for robust watermarking relational database technique for copyright protection is introduced. A solution is proposed by:

- Reduce transfer rate of database.
- Building an algorithm for watermarking such that it introducing zero distortion in semantic meaning.
- Improving robustness by generate watermark using hash function and assume v=3 this leads to insert watermark in different three attributes. As a result, it is so difficult for attacker to guess the watermark.

Through experiments, the technique is robust against delete, subset, insertion and modification attacks as well alteration case attack. Also it is considered fully blind. In future the fingerprint features of database's owner that has unique characteristics can be inserted into database. Therefore, can easily detect it and prove that if database belongs to this owner or not.

7. REFERENCES

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

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

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

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

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

[6] Bedi R., Thengade A., Wadhai V., "A New Watermarking Approach for Non Numeric Relational Database", 2011.

[7] pramod a.kharade,vikramsinh m. pokale, vijay d. chougule and vishal v. mangave,"speech based watermarking for non-numeric relational database" , international journal of innovative research in engineering & science,april 2013.

[8] Nahla El_Haggar, Mahmoud M. Elkhouly, Samah S. Abu El Alla," Blind Watermarking Technique for Relational Database",COMPUSOFT, An international journal of advanced computer technology, May-2013.

[9] Abha Tamrakar, 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.

[10] Brijesh B. Mehta," A Novel approach as multiplace watermarking for security in database", Dept. of Computer Engineering, S. V. National Institute of Technology, Surat, Gujarat, INDIA-, 2010-2011.