WATERMARKING DATA OF RELATIONAL DATABASES ISSUES AND CHALLENGES

Palacholla Padmanabham
Hasan Al-Saedy
Mohd Abdul Salam

Abstract

Watermarking data for copyright protection is an accepted technique adopted for digital information. This subject has been quite exhaustively researched and several techniques have been established for protecting copyrights for still images, audio and video files. Recently attempts are being made to watermark data in relational databases. The aim of this paper is to look into the issues and challenges in the area of watermarking data of relational databases and to propose a possible new technique which needs to be studied in depth

Keywords : Watermarking, relational databases

1. Introduction

Watermarking has now become a standard technique to mark multimedia digital data to protect the privacy of the data. Inserting digital watermarks can now protect the privacy of digital data such as images, audio and video. Although water marking does not prevent data being stolen (illegal copying), but allows establishing the original ownership. Several techniques have been proposed and tested [2] , [3], [4] for water marking digital data and several commercial software are also available [5], [6], [7], [8]. All these techniques are based on one basic principle that the watermarking software introduces small errors into the object being watermarked. These intentional errors are the marks that constitute the total watermarking. However these marks (errors) are chosen so as to have an insignificant impact on the usefulness of the data and further a malicious user cannot precisely locate these marks. Therefore an important aspect of these techniques is that the malicious user cannot destroy the watermarks without considerably degrading the quality of the data and thus rendering it almost useless.

The usage of relational database data in data warehousing and data mining applications is creating a need for establishing the ownership of the data. Moreover it will be quite useful in the area of cyber forensics to establish the ownership of the data. It is a good idea to implement watermarking technique to claim the ownership of data in a relational database. However the question is to what extent it will be possible to extend the techniques used for multimedia digital data to the data in relational databases? In this paper we present a formal model for watermarking data in a relational database so as to bring out the pertinent issues and challenges. Further we examine the limitations of extending the existing techniques of watermarking of multimedia digital data to the data in relational databases and we conclude by suggesting a possible alternate solution, which we hope to implement and test.

2. Model for watermarking relational database

Now we present a formal algorithmic model for watermarking relational data based on the techniques used in watermarking multimedia digital data. Let $R(K,A_0,A_1 \ldots\ldots A_n\text{-}1)$ be the schema of a relation R. Let n be the number of attribute in the relation R. Let K be the primary key of the relation. Fig 1 shows the algorithm model, which is based on introducing watermarks randomly into the data of the relation probably controlled by a private key [1], which could be later on used by the owner of the data to detect the watermarks.

For each tuple r of the relation R at line 3 the function select_tuple(r) is called. This function returns true if this tuple is selected for marking otherwise false. It is necessary to randomize this function using a private key [1] of reasonable length, which will remain with the owner and later used to identify the watermarks. The detection of water marking will be presented in detail later in this section.

```
1. for each tuple r Œ R {
2.      // check if the tuple is to be maked
3.          If (select_tuple ( r ) ) {
4.                  // get which attribute is selected for marking
5.                  j = select_attribute( r );
6.                  // j is the index of the attribute to be marked
7.                  // get the bit to be marked
8.                  i = select_bit(Aj );
9.                  // mark the i th bit
10.                 mark( Aj , i);
11.                             }
12.                 }
13.          // the algorithm ends here
14.          // the details of the various functions are presented
15.            // later in this section
```

Fig 1. An algorithmic model for water marking data in relational database

At line 5 the function select_tuple(r) is called which returns the index of the attribute selected for marking. One of the requirements for this function is that it should select the attribute randomly controlled by the private key. However the type of attribute picked up is of greater importance since we cannot mark any bit in the attribute without changing the value of the attribute. One possibility is [1] to assume that all the attributes are numeric and mark in such a way that the change in the value does not affect the validity of the data. At line 8 the function select_bit(A_j) chooses the i^{th} bit to be marked assuming that this bit lies within x least significant bits [1] so that the value of the attribute is insignificantly altered. However this does not work for character or integer or date or Boolean type attributes, as the value of these types of attributes will be significantly changed if a bit is altered at any place. At line 10 the algorithm calls the function mark(A_j , i), to mark the selected bit. This function has to compare the bit selected with the watermark. If the bit coincides with the watermark, then the bit value is not changed else the bit value is changed to coincide with the watermark. The next question is how to get the Watermark? One way suggested [1] is use MAC (Message Authentication Code) which can be generated using one way hash function.

3. Model for watermark detection

Now we present an algorithmic model to detect the watermark. The algorithm is to be used when the ownership of the data is to be established and therefore needs the private key used in watermarking the data and the watermark. For each tuple in the relation R at line 3 it is tested whether the tuple has been marked or not. Line 5 and 8 locate the bit marked. At line 11 compare_bit is called and if the comparison succeeds we increment match_count . At line 12 the total_count is incremented. This could be necessary for determining the threshold. At line 16 the match_count is compared with the threshold and whether to suspect or not is decided. The threshold value can be based on probability [1].

```
1. for each tuple r Œ R {
2.     // check if the tuple is to be maked
3.       If (select_tuple ( r ) ) {
4.                    // get which attribute is selected for marking
5.                    j = select_attribute( r );
6.                    // j is the in dex of the attribute to be marked
7.                   // get the bit to be marked
8.                    i = select_bit($A_j$ );
9.                    // compare  the i th bit with watermark
10.               // match_count and total count are intialized\to 0
11.                  if (copare_bit( $A_j$ , i)) match_count++;
12.                     total_count++;
13.                            }
14.                    }
15.   // let _ be the threshold
16.    if(match_count > = _ ) suspect( );
17.         // the algorithm ends here
18.            // the details of the various functions are presented
19.               // later in this section
```

fig 2. An algorithmic model for detecting the watermark in relational database

4. Conclusions and a new proposal:

The algorithmic models presented above are based on the techniques adopted for watermarking digital data such as audio, video and other forms. The serious limitation of this method is that we cannot alter the bits in any type of attribute without causing substantial change. For example in a date field all the bits are important and change in any bit will totally change the field. This certainly indicates a necessity to conceive a different approach to this problem. One possible proposition is as follows:

Generally the watermarking algorithm needs a cryptographically secure pseudo-random sequence (watermark) and a private key. Instead of making the data using the private key we propose to do the opposite. We proceed to identify the water mark sequence within the data and create the private key, in which case the bits are not altered. The detection can be made using the private key and the watermark.

We propose to come up with a detailed algorithmic model for the above proposal (possibly implement and test) in future.

5. Reference:

[1] Rakesh Agarwal, Peter J Haas and Jerry Kiernan "A System for Watermarking Relational Databases" SIGMOD June 9-12 2003

[2] L. Boney , A.H. Tewfik and K.N. Handy "Digital Watermarking for Audio Signals", International conference on Multimedia Computing and Systems, Hiroshima, Japan June 1996

[3] F. Hartung and B.Girod , watermarking of uncompressed and compressed video. Signal Processing 66(3);283-301 1998

[4] Josep J. K . O Ruanaidh , W.J. Dowling , and F.M. Boland Watermarking Digital Images for copyright protection IEEE proceedings on Vision, Signallig and Image processing, 143(4);250-256 1997

[5] http://www.teletrax.tv/index.php-----teletrax-Watermarks TV footage and tracks wherever broadcast.

[6]http://www.alphatecltd.com/watermarking/watermarking.html-----Alpha-Tec makers of Eikonamark, Audiomark, and Videomark which watermark still images, audio and video files, respectively.

[7] http://www.verance.com/-----verance- provides digital rights management and copy/usage control tools to owners and distributors of sound recordings (including DVD audio and SDMI), television programming and motion pictures

[8] http://www.alpvision.com/products.html-Alpha Vision-watermarks printed docs/photos, digital images, and CD-ROM

Dr.Palacholla Padmanabham
PO Box:1797, Tel.: 00971 6 5439444
Skyline College, Sharjah, UAE

Dr. Hasan Al-Saedy
PO Box:1797, Tel.: 00971 6 5439444
Skyline College, Sharjah, UAE

Mr. Mohd Abdul Salam
PO Box:1797, Tel.: 00971 6 5439444
Skyline College, Sharjah, UAE