33rd Annual International Conference of the IEEE EMBS
Boston, Massachusetts USA, August 30 - September 3, 2011

Lossless Watermarking of Categorical Attributes for Verifying Medical Data Base Integrity

G. Coatrieux, E. Chazard, R. Beuscart*, IEEE Members*, C. Roux, *IEEE Fellow*

Abstract— **In this article, we propose a new lossless or reversible watermarking approach that allows the embedding of a message within categorical data of relational database. The reversibility property of our scheme is achieved by adapting the well known histogram shifting modulation. Based on this algorithm we derive a system for verifying the integrity of the database content, it means detecting addition, removal or modification of any t-uples or attributes. Such a content integrity check is independent of the manner the database is stored or structured. We illustrate the overall capability of our method and its constraints of deployment considering one medical database of inpatient hospital stay records. Especially, we reversibly watermark ICD-10 diagnostic codes.**

I. Introduction

THE development of medical information systems participates to the increase of care quality. Though these systems facilitate sharing, distribution and access of patient data among health professional, they also have the potential to seriously compromise medical data security. Security concerns like integrity and confidentiality of health data are surrounded by strict ethic and several legislative international and national rules (HIPPA and ISO/CEI 27001).

Watermarking is a security mechanism which has been proposed for protecting medical data [1]. It consists of the insertion of a message, also called watermark, in a host document in some multimedia format (image, XML, database …). It encodes the message within host data based on the principle of controlled distortion. For instance, in the case of an image, its pixels' gray values will be modulated for message embedding. Basically, it is required that the watermarked information remains hidden to any unauthorized user (as for data encryption, a secret key is needed to access the watermark), non-interfering with the use of the watermarked document (watermark imperceptibility) and fragile (integrity) or robust (authentication) to any attempt to suppress it. Originally devoted to copyright protection of multimedia documents, watermarking has also attractive properties that fit within the healthcare domain. Two main objectives of watermarking are foreseen in the medical domain [1]: data hiding for the purpose of inserting meta-data to make the host content more usable, and information protection with applications like integrity control. In healthcare, the essential of the literature focuses on watermarking medical images. A very few works have been devoted to other kind of medical data like database [3].

Several strategies for Database watermarking have been proposed. Most of them treat ownership protection issues [4-5] and a few have for interest integrity protection [6-7]. These methods can be differentiated depending on the way embedding is conducted and especially on the nature of the database record attributes used for message embedding: numerical or categorical attributes. Numerical attributes have been the first watermarked due to their similarity with image or video signal samples. The first method was proposed by Agrawal and Kiernan [4] who modify or force to a specific value the least significant bit of some numerical values from secretly selected records' or tuples' attributes. The watermark detection consists in verifying if tuples under investigation respect the watermark LSB distribution. Since, several other methods have been proposed [8]. Categorical attributes differ in the fact that there may not exist relationship between the values such an attribute can take. For example, if we consider the attribute "eye color" its values may be "brown", "blue", "green" … but between them there is no relationship like "greater than", "smaller than", "equal to", and so on. In order to make message embedding possible, Sion *et al.* [5] suggest to associate one numerical value to each attribute value. For instance, "green", "blue", "brown", "grey" can be associated to '0', '1', '2', '3' respectively. As a consequence, the scheme of Agrawal *et al.* can next be applied. Color-attribute-value change is thus assimilated to an LSB modification. Obviously, the reader needs to know the correspondence between categorical attribute values and their decimal representation.

Like in the multimedia case, the watermark can be made public or private (requiring or not the original database for watermark detection and message extraction); "single bit" or "multi-bit" (allowing the insertion of several bit of a message); robust or fragile to typical database attacks including tuple or attribute insertion, deletion, reorganization, modification ….

Whatever the method, watermarking makes the fundamental assumption that the database can tolerate some "errors" or some distortion for message embedding [9]. Since the two basic modulations presented above for numerical and categorical attributes, several other approaches have been proposed, taking into account more constraints, especially in terms of "distortion imperceptibility". However, this notion

G. Coatrieux and C. Roux are with the Institut Telecom; Telecom Bretagne; Unite INSERM 650 Latim, Technopole Brest-Iroise, CS 83818, 29238 Brest Cedex 3 France (e-mail: gouenou.coatrieux@telecom-bretagne.eu).

E. Chazard and R. Beuscart are with the Department of Medical Information and Archives, CHU Lille; UDSL EA 2694; Univ Lille Nord de France; F-59000 Lille, France

978-1-4244-4122-8/11/$26.00 ©2011 IEEE

of imperceptibility may differ and impose constraints different from the multimedia case. For instance, Bertino *et al.* aim at satisfying anonimyzation constraints when modifying attributes' values of a medical database [3].

Reversible watermarking can be used to overcome such a non-distortion requirement. The reversibility property offers the possibility to reconstruct the original data from its watermarked version by reversing watermarking distortions. Thus, concomitant risks of interferences with data interpretation or manipulation are reduced. Lossless schemes have been originally proposed for medical images. To our knowledge, only one experiment has been conducted by Gupta *et al.* on numerical attributes [7]. No solution has been yet proposed for categorical attributes of database. In this work, we extend the well known histogram shifting modulation (HS) to categorical attributes and we show up how this one can be used for verifying integrity of database.

The rest of the paper is organized as follows. In section II, we sum up the common process followed by most database watermarking schemes and introduce our integrity control system. In section III, we present histogram shifting modulation and how to adapt it to categorical attributes. Section IV illustrates the performances of our scheme on a database constituted of inpatient hospital stays records.

II. WATERMARKING OF DATA BASE

A. A common chain of database watermarking

Let us consider a relational database (DB) constituted of different tables. In the sequel, for simplicity reasons, we consider one database with one single table constituted of M_r records or tuples $\{r_u\}_{u=1...Mr}$. One tuple r_u is made of different attributes $r_u.A_i$, i=1 … M_A, where A_i si the i^{th} attributes of the u^{th} tuple. Each tuple r_u is identified by its primary key $r_u.P$.

Most watermarking schemes proposed in the literature follow the embedding procedure depicted in figure 1. It first starts by a process which objective is to organize tuples in a specific manner for watermark embedding or reading. This process is necessary in order to guarantee independency of the watermarking scheme from the way the database is organized or stored (i.e. independent to the records order in tables). Usually, groups of tuples are constituted. One usual strategy for building N groups of tuples consists in deriving the tuple group number n_u of a tuple r_u from its primary key:

$$n_u = hash(<r_u.P,K_w>) \bmod N$$

where *hash* is a cryptographic hash function like the SHA (Secure Hash Algorithm [ref]), <.,.> the concatenation operator and K_w a secret watermarking key.

The second step consists in the embedding of one or several bits of the message within each group of tuples $\{Gr_n\}_{n=1\ ...\ N}$,. Message bits are sequentially inserted/read. Thus insertion/reading depends on the way how tuples are browsed. Contrarily to images where signal samples are naturally organized we need to reorder tuples within one group. One common solution consists in ordering tuples in one group in the ascending or decreasing order based on their hash value (i.e. $hash(<r_u.P,K_w>)$). Once tuples gathered in their group and ordered, they are modified so as to encode bits of the message and then reorganized back to constitute the watermarked database (DB^w). Watermark detection is conducted in the same way. Groups of tuples are rebuilt and watermark detection or message extraction is conducted in each group of tuples. Our reversible watermarking algorithm works in the same way.

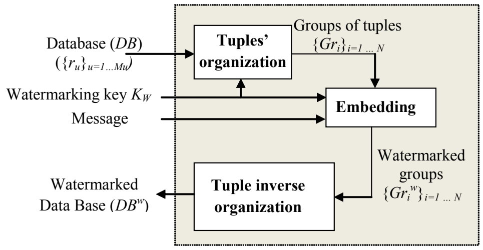


Fig.1. A common embedding procedure for database watermarking. .

B. Verifying integrity of database content based on lossless watermarking

The basic principle of our scheme is presented in figure 2. The idea is to embed a digital signature of the database itself. The detection sensitivity of cryptographic hashes to any modifications is such that it is normally not possible to modify or watermark attributes without changing their hash. Here comes the interest of lossless watermarking. The signature can be computed over the whole data base and next watermarked (see figure 2a). Obviously, it is required to remove the watermark from the database before verifying its integrity (see figure 2b). Thus, if the extracted hash is different from the one computed over the reconstructed database (i.e. un-watermarked database), then the database is not identical to the original one and an alarm is raised. The proposed method is fragile, any changes will impact extracted hashes.

III. LOSSLESS WATERMARKING OF CATEGORICAL ATTRIBUTES

A. Lossless watermarking modulations

Since 1999, several reversible watermarking methods for images have been proposed [10-14]. The solution proposed in [7] for database is derived from the method in [10].

Two main reversible modulations can be distinguished: additive and substitutive methods. Substitutive techniques come directly replace the signal by another one stemmed from a predetermined dictionary. We suggest classifying these methods into two subcategories: Lossless Compression Embedding techniques (LCE) and Expansion Embedding techniques (EE). The method proposed by Fridrich *et al.* [11] is one of the first LCE schemes. It substitutes one or more bit planes, by its compressed version concatenated with the message to be embedded. EE techniques expand the dynamic of the signal by shifting to the left the binary representation of one signal sample, thus creating a new virtual least significant bit (LSB) which can be used for data insertion [10].

With additive approaches, the message *m* to be embedded is first transformed into a watermark signal *w* next added to the host image *I*, leading thus to the watermarked image I_w. Without paying careful attention, I_w may include pixels with gray values outside the allowable image dynamic range introducing "underflows" or "overflows". To overcome this issue different additive approaches have been proposed. One of them refers to the Histogram Shifting (HS) modulation first suggested by Ni *et al.* in [12]. The basic principle of this modulation is given in figure 3. HS shifts a range of the data histogram (like image histogram) to create a 'gap' near the histogram maxima (C_1 in fig. 3b). Samples with values associated to the class of the histogram maxima (C_0 in fig. 3b) are then shifted to the gap or kept unchanged to encode one bit of the message, i.e. '0' or '1'. Data samples that belong to this class are named "*carriers*". Other samples, i.e. "*non carriers*", are simply shifted. At the reading stage, the reader just has to interpret the message from classes C_0 and C_1 and to invert watermark distortions (i.e. shifting back shifted value). Obviously, in order to restore exactly the original data, the watermark reader needs to be informed of the positions of data samples which have been shifted out of the dynamic range [v_{min}; v_{max}] ("*overflows*" in figure 3b). This requires the embedding of an overhead and reduces the watermark capacity. Typically this overhead corresponds to a location map (a vector) which components inform the reader if samples of value v_{max} are original values or shifted values.

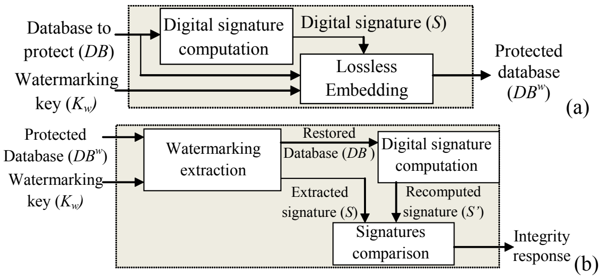

Database to protect (*DB*)
Watermarking key (K_w)
Digital signature computation
Digital signature (*S*)
Lossless Embedding
Protected database (DB^w)
(a)
Protected Database (DB^w)
Watermarking key (K_w)
Watermarking extraction
Restored Database (DB^r)
Digital signature computation
Extracted signature (*S*)
Recomputed signature (*S'*)
Signatures comparison
Integrity response
(b)

Fig.2. Verifying database integrity based on lossless watermarking: (a) protection procedure, (b) verification procedure.

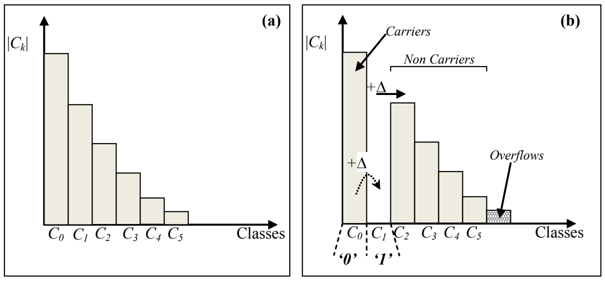


Fig.3. Histogram shifting modulation. (a) original histogram (b) histogram of the watermarked data.

To our knowledge, HS has only been applied to numerical data, i.e. type of data to which is associated a known dynamic range. Without such a relationship between the values one data sample can take, "shifting" cannot be applied. We show up in the next section how a "virtual dynamic" can be obtained between the values one categorical attribute can take.

Beyond the dynamic range issue, HS cannot be applied to data which probability density is uniform. In fact, HS payload (C), that is the number of bit of message embedded per sample of host data, is defined as:

$$C = \quad - \quad {}_{\max} + \quad {}_{\max -}$$

where C_0 is the class of carrier samples (see fig. 3), $C_{v_{\max}}$ and $C_{v_{\max -}}$ are classes associated to "overflows" and |.| gives the class cardinality. Thus; a host can be HS watermarked if the capacity given by $|C_0|$ is greater than the overhead length, i.e. $\left|C_{v_{\max}}\right| + \quad {}_{\max -}$.

B. Categorical histogram shifting

As depicted in the previous section HS is applied on data to which is associated a dynamic range. The basic idea of our proposal is thus to construct a "virtual dynamic" considering the statistical distribution of the attribute values and to order these values according to their occurrences within the database. Because HS insertion will impact the retrieval of this virtual dynamic at the detection stage, we propose to split one group of tuples Gr_n (see section II.A) into two sub-groups. One sub-group will be used as reference for building the virtual dynamic and will not be watermark. Consequently, the reader will be able to retrieve this virtual dynamic. The second sub-group is used for message embedding taking into consideration the virtual dynamic.

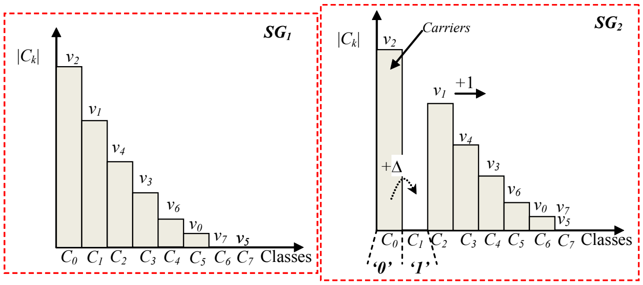
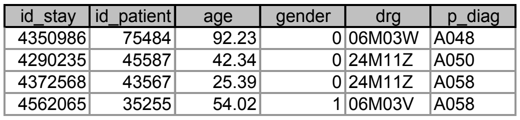


Fig.4. HS apply to one categorical attribute with values $\{v_i\}_{i=0,\ldots,7}$. - SG_1- tuple subgroup use to derive the histogram *x*-axis (i.e. "virtual dynamic"). $\{v_i\}$ are sorted depending of their occurrences in SG_1. - SG2 - tuple subgroup use for embedding applying HS based on SG1 virtual dynamic.

Thus, the strategy we propose is the following one. Let us assume that tuples of the database have been: i) distributed in groups $\{Gr_n\}_{n=1 \ldots N}$,and; ii) ordered in each group according to their hash value as exposed in section II-A. In one group Gr_n, each tuple is then secretly assigned to one subgroup: SG_1 or SG_2, making again use of the secret key K_w. Let us consider SG_1 as reference for building the histogram virtual dynamic. Values the attribute can take are organized depending on their occurrence in SG_1 and ordered in decreasing order (see fig. 4). If any, empty classes are arbitrarily distributed at the end of this virtual dynamique. As a consequence, the class C_0 corresponds to the value that appears with the highest occurrence in SG_1 and is identified as the "carrier class". Tuples of SG_2 which attributes belongs to this class correspond to the carrier tuples, others are a non-carriers. Message embedding is conducted sequentially in SG_2 browsing its tuples based on their organization in Gr_n. Each

SG_2 attribute refered as non-carrier is right-shifted according to the virtual dynamic. SG_2 carrier attributes are right shifted or kept unchanged depending on the message bit to be embedded (see fig. 4).

At the reading stage, the watermark reader just has to reconstitute groups of tuples $\{Gr_n\}_{n=1 \dots N}$. For each Gr_n, it retrieves subgroups SG_1 and SG_2. The virtual dynamic is re-identified from SG_1. Message extraction is conducted on SG_2 by directly interpreting attributes belonging to C_0 or C_1. SG_2 is then restored according to HS modulation principles.



id_stay	id_patient	age	gender	drg	p_diag
4350986	75484	92.23	0	06M03W	A048
4290235	45587	42.34	0	24M11Z	A050
4372568	43567	25.39	0	24M11Z	A058
4562065	35255	54.02	1	06M03V	A058

Fig.5. Sample view of our medical database. In this table one row or tuple corresponds to one stay or to one structured discharge record.

C. Discussion

The efficiency of this approach depends on the statistical distribution properties of an attribute. Especially, our approach stands on the hypothesis that these properties do not vary too much between sub-groups of tuples. Capacity performances are intimately related to the cardinality of C_0 ($|C_0|$) and on the accuracy of the virtual dynamic. If SG_1 is too small, its histogram may differ from the one of SG_2. Thus a compromise needs to be made between group and subgroup dimensions. Overflows may be critical to handle, especially if the number of values an attribute can take is somewhat limited. If this number is small, a large amount of overflows may occur reducing drastically payload performances.

As most reversible schemes, our method is fragile. Any modification will not allow us to reconstruct the original database. Thus this solution is well adapted to the integrity control framework. It may also be possible to localize group of tuples that have been modified if the integrity protection scheme we present in section II.B is applied independently to group of tuple (i.e. computation of a signature on one group followed by its embedding within the same group).

The security of the proposed scheme relies on the constitution of tuple groups and subgroups. Even though one hacker knows the information is carried by the attribute value with highest occurrence in the database, he or she will not be able to reorganize the message bit stream or counterfeit it easily without the knowledge of the secret key (K_w).

IV. Experimental results

We have applied our scheme to one medical database which contains all information in relation with the hospital stays of inpatients. For this experiment and sake of simplicity we have considered one table of this database. In this table of 24000 tuples, one tuple corresponds to a structured discharge record which includes the attributes (see figure 5): stay identifier ('id_stay'), patient identifier ('id_patient'), patient age and gender, ICD-10-encoded principal diagnosis ('p_diag') ….

The attribute "id_stay" was considered as the primary key and used for tuple groups and subgroups constitution. (see section II.A). Again for sake of simplicity we only have considered one group of tuples (i.e. N=1 – see section II.A). The attribute we retain for embedding is the principal diagnosis attribute ('p_diag') which takes values defined by the International Classification of Diseases (ICD-10) itself maintained by the World Health Organization (WHO) t. *p_diag* is a categorical attribute and, even though ICD-10 codes are included in a simple hierarchy, there is no algebraic relationship between ICD-10 codes.

Whence, in order to protect the integrity of this table, one hash/signature is computed applying the SHA-256 to the whole table. Then this signature of 160 bit long is embedded in the '*p_diag*' attribute in the table. The capacity of our scheme depends on the dimension of the subgroup SG_1 and SG_2. Capacity we achieved is 1597, 1370 and 1137 bits using $|SG_1|$ equals to 1756, 5212 and 8575 respectively. Such a capacity is large enough for embedding one SHA hash.

V. Conclusion

In this paper, we have proposed a new lossless watermarking algorithm for categorical attributes of databases and we have illustrated how this algorithm can be used for protecting efficiently the integrity of medical database content. The proposed scheme is independent on the way the database is structured and how records are organized. Furthermore, the reversibility property ensures the capability to recover exactly the original database. We also shown that ICD-10 codes can be watermarked. Achieved capacity gives access to a watermarking communication channel that can be used for medical database protection.

References

[1] G. Coatrieux , H. Maitre , B. Sankur , Y. Rolland , R. Collorec, " Relevance of Watermarking in Medical Imaging," in Proc. of IEEE Int. Conf. EMBS ITAB, 2000, pp. 250-255.

[2] A.U. Rajendra, U.C. Niranjan, S.S. Iyengar, N. Kannathal, Lim Choo Min, "Simultaneous storage of patient information with medical images in the frequency domain," Computer Methods and Programs in Biomedicine, 2004, 76, 13-19.

[3] E. Bertino, B.C. Ooi, Y. Yang, and R. Deng, "Privacy and ownership preserving of outsourced medical data." ICDE 2005.

[4] R. Agrawal, J. Kiernan, R. Srikant, Y. Xu, "Hippocratic Databases," Proc. of the 28th Int'l Conf. on Very Large Databases, 2002.

[5] R. Sion, "Proving ownership over categorical data, " in Proc. of IEEE Int. Conf. on Data Engineering ICDE, 2004.

[6] H. Guo, Y. Li, A. Liu, and S. Jajodia: A Fragile Watermarking Scheme for Detecting Malicious Modifications of Database Relations. IS 2006.

[7] G. Gupta, J. Pieprzyk, "Reversible and blind database watermarking using difference expansion," in Proc. of int. conf. on e-Forensics, 2008, pp24-30.

[8] S. Yige, L. Weidong,S. Jiaxing, "DCT Transform Based Relational Database Robust Watermarking Algorithm," 2nd Int. Symp. on Data, Privacy and E-Commerce (ISDPE) , 2010, pp. 61-65.

[9] Y. Li, R.H. Deng,"Publicly verifiable ownership protection for relational databases," in Proc. of ACM Symp. on information, Computer and Communications Security, 2006, pp.78-89.

[10] J. Tian, "Reversible data embedding using a difference expansion," IEEE Trans. on Circuits Syst. Video Technol., vol. 13, no. 8, pp. 890–896, 2003.

[11] J. Fridrich, J. Goljan, R. Du, "Invertible authentication," in Proc. of Int. Conf. SPIE, Security and Watermarking of Multimedia Content, 2001, pp. 197-208.

[12] Z. Ni, Y. Shi, N. Ansari, and S.Wei, "Reversible data hiding," in Proc. IEEE Int. Symp. Circuits and Systems, May 2003, vol. 2, pp. 912–915.