Data Security In Relational Database Management System

R.Balasubramaniam *bala27788@gmail.com*
Assistant Professor/Department of Computer Science
Kathir College of Engineering
Coimbatore-641062, Tamilnadu, India

Abstract

Proving ownerships rights on outsourced relational database is a crucial issue in today's internet based application environments and in many content distribution applications. Here mechanism is proposed for proof of ownership based on the secure embedding of a robust imperceptible watermark in relational data. Watermarking of relational databases as a constrained optimization problem and discus efficient techniques to solve the optimization problem and to handle the constraints. This watermarking technique is resilient to watermark synchronization errors because it uses a partioning approach that does not require marker tuple. This approach overcomes a major weakness in previously proposed watermarking techniques. Watermark decoding is based on a threshold-based technique characterized by an optimal threshold that minimizes the probability of decoding errors. An implemented a proof of concept implementation of our watermarking technique and showed by experimental results that our technique is resilient to tuple deletion, alteration and insertion attacks.

Keywords: Embedding Algorithm, Watermarking & Threshold Value

1. INTRODUCTION

The rapid growth of the Internet and related technologies has offered an unprecedented ability to access and redistribute digital contents. In such a context, enforcing data ownership is an important requirement, which requires articulated solutions, encompassing technical, organizational, and legal aspects. Although we are still far from such comprehensive solutions, in the last years, watermarking techniques have emerged as an important building block that plays a crucial role in addressing the ownership problem. Such techniques allow the owner of the data to embed an imperceptible watermark into the data. A watermark describes information that can be used to prove the ownership of data such as the owner, origin, or recipient of the content. Secure embedding requires that the embedded watermark must not be easily tampered with, forged, or removed from the watermarked data. Imperceptible embedding means that the presence of the watermark is unnoticeable in the data. Furthermore, the watermark detection is blinded, that is, it neither requires the knowledge of the original data nor the watermark. Watermarking techniques have been developed for video, images, audio, and text data and also for software and natural language text. By contrast, the problem of watermarking relational data has not been given appropriate attention. There are, however, many application contexts for which data represent an important asset, the ownership of which must thus be carefully enforced. This is the case, for example, of weather data, stock market data, power consumption, consumer behavior data, and medical and scientific data. Watermark embedding for relational data is made possible by the fact that real data can very often tolerate a small amount of error without any significant degradation with respect to their usability. For example, when dealing with weather data, changing some daily temperatures of 1 or2 degrees is a modification that leaves the data still usable.[8]

To date, only a few approaches to the problem of watermarking relational data have been proposed. These techniques, however, are not very resilient to watermark attacks. In this paper, we present a watermarking technique for relational data that is highly resilient compared to these techniques. In particular, our proposed technique is resilient to tuple deletion, alteration, and insertion attacks. The main contributions of the paper are summarized as follows: We formulate the watermarking of relational databases as a constrained optimization problem and discuss efficient techniques to handle the constraints.

2. Existing System

Watermarking in least significant bits (LSB).This technique embeds the watermark bits in the least significant bits of selected attributes of a selected subset of tuple's. It uses secret key in watermarking. For each tuple's a secure message, authenticated code is computed using the secret key and tuple's primary key. The computed MAC is used select candidate tuple's attributes and the LSB positions in the selected attributes. This technique does not provide mechanism for multi bit watermarks. The watermark can be easily compromised by very trivial attacks [11].

2.1 Drawbacks

This technique does not provide mechanism for multi bit watermarks. The watermark can be easily compromised by very trivial attacks

3. Proposed System

Watermarking embeds ownership information in digital content. Watermark describes information that can be used to prove the ownership of relational database. Here the embedding is hidden that the presence of watermarking is invisible to the user. It is not resilient to watermark attacks. Optimal threshold reduces probability of decoding error. Multiple embedding of watermark bits in the dataset increases additional security.

3.1 Objective of Proposed work

It is not resilient to watermark attacks. Optimal threshold reduces probability of decoding error. Multiple embedding of watermark bits in the dataset increases additional security.

4. MODULE DESCRIPTION

4.1 SERVER (SOURCE) MODULE

- Main Form has controls like select Table Name and specify Destination System Name
- DBCON –
  This class establishes a connection to the database using SQL SERVER.
- ACTION CONTROLLER –
  This class initiates the process by accessing the data table.
- PARTITION –
  This class eventually divides the table records and assigns partition number to it.
- SINGLE BIT ENCODING –
  This class encodes the partitioned file by adding one bit to each record.
- ENCRYPTION –
  This class gets all the encoded records and encrypts them as a whole file along the secret key.
- WATERMARK SERIALIZATION –
  This class serializes (Convert The Object into file) and sends it to the destination via network connections.

4.2 CLIENT (DESTINATION) MODULE

- MAIN FORM –

This class receives the file from source through thread.

- WATERMARK DESERIALIZE –

This class reads the received file and convert the stream as file.

- DECRYPTION –

This class decrypts the file using the secret key.

- SINGLE BIT DECODING –

This class gets the file as an object from the Hash Table using the key value added to it and also decodes the file by removing the additional bit added to each record.

- REVERSE PARTITION –

This class merge the partitioned file or records into a single file.

- DATA INSERT –

This class reads each record and stores it in Hash Table and which in turn is inserted into the Database one by one using SQL SERVER.

4.3 DATA PARTITIONING

Data partitioning in relational data warehouse can implemented by objects partitioning of base tables, clustered and non-clustered indexes, and index views. Range partitions refer to table partitions which are defined by a customizable range of data. The end user or database administrator can define the partition function with boundary values, partition scheme having file group mappings and table which are mapped to the partition scheme. By using secret key the data set is partitioned into several non overlapping partitions.

4.4 WATERMARK EMBEDDING – SINGLE BIT ENCODING

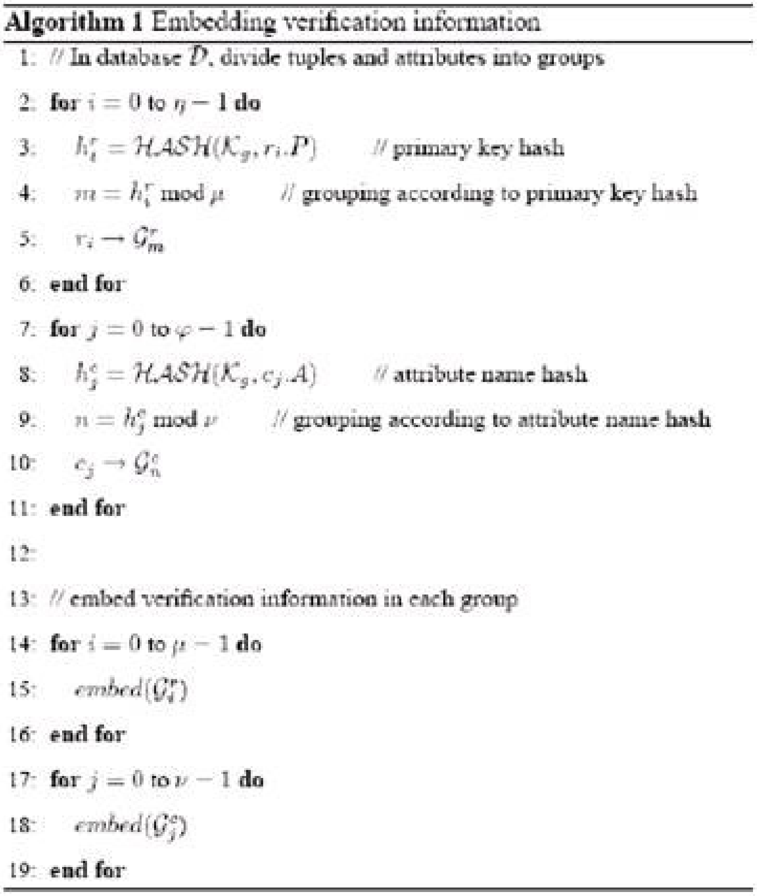
A watermark bit is embedded in each partition by Single Bit Encoding algorithm (figure.1). Watermarking is a technology for embedding various types of information in digital content. In general, information for protecting copyrights and proving the validity of data is embedded as a watermark. Watermarks are added to images or audio data in such a way that they are invisible or inaudible Ñ unidentifiable by human eye or ear. Furthermore, they can be embedded in content with a variety of file formats. Watermarking is the content protection method for the multimedia era.

4.5 OPTIMAL THRESHOLD EVALUATION

The bit embedding statistics are used to compute the optimal threshold that minimizes the probability of decoding error. The optimization technique used in this experiment is pattern search technique (PS). PS methods are direct search methods for non-linear optimization. It starts at an initial point and samples the objective function at a predetermined pattern of points centered about that point with the goal of producing a new better iterate

4.6 THRESHOLD BASED DECODING

The statistics of each partition are evaluated, and the embedded is decoded using a threshold based scheme based on the optimal threshold. The probability of bit decoding error is de.ned as the probability of an embedded bit decoded incorrectly. The decoding threshold T_ is selected such that it minimizes the probability of decoding error. The bit embedding stage is based on the maximization or minimization of the tail count; these optimized hiding function values computed during the encoding stage are used to compute the optimum threshold T.



```
Algorithm 1 Embedding verification information
// In database $\mathcal{D}$, divide tuples and attributes into groups
for $i = 0$ to $\eta - 1$ do
    $h_i^r = \mathcal{HASH}(\mathcal{K}_g, r_i.P)$        // primary key hash
    $m = h_i^r \bmod \mu$        // grouping according to primary key hash
    $r_i \rightarrow \mathcal{G}_m^r$
end for
for $j = 0$ to $\varphi - 1$ do
    $h_j^c = \mathcal{HASH}(\mathcal{K}_g, c_j.A)$        // attribute name hash
    $n = h_j^c \bmod \nu$        // grouping according to attribute name hash
    $c_j \rightarrow \mathcal{G}_n^c$
end for

// embed verification information in each group
for $i = 0$ to $\mu - 1$ do
    $embed(\mathcal{G}_i^r)$
end for
for $j = 0$ to $\nu - 1$ do
    $embed(\mathcal{G}_j^c)$
end for
```

FIGURE 1: Single Bit Encoding Algorithm

4.7 ENTITY RELATIONSHIP DIAGRAM

Entity-Relationship model (ER model for short) is an abstract and conceptual representation of data. Entity-relationship modeling is a database modeling method, used to produce a type of conceptual schema or semantic data model of a system, often a relational database, and its requirements in a top-down fashion. Diagrams created by this process are called entity-relationship diagrams or ER diagrams. The ER diagram for Sender Activity and Receiver Activity is shown in figure.2 & figure 3.

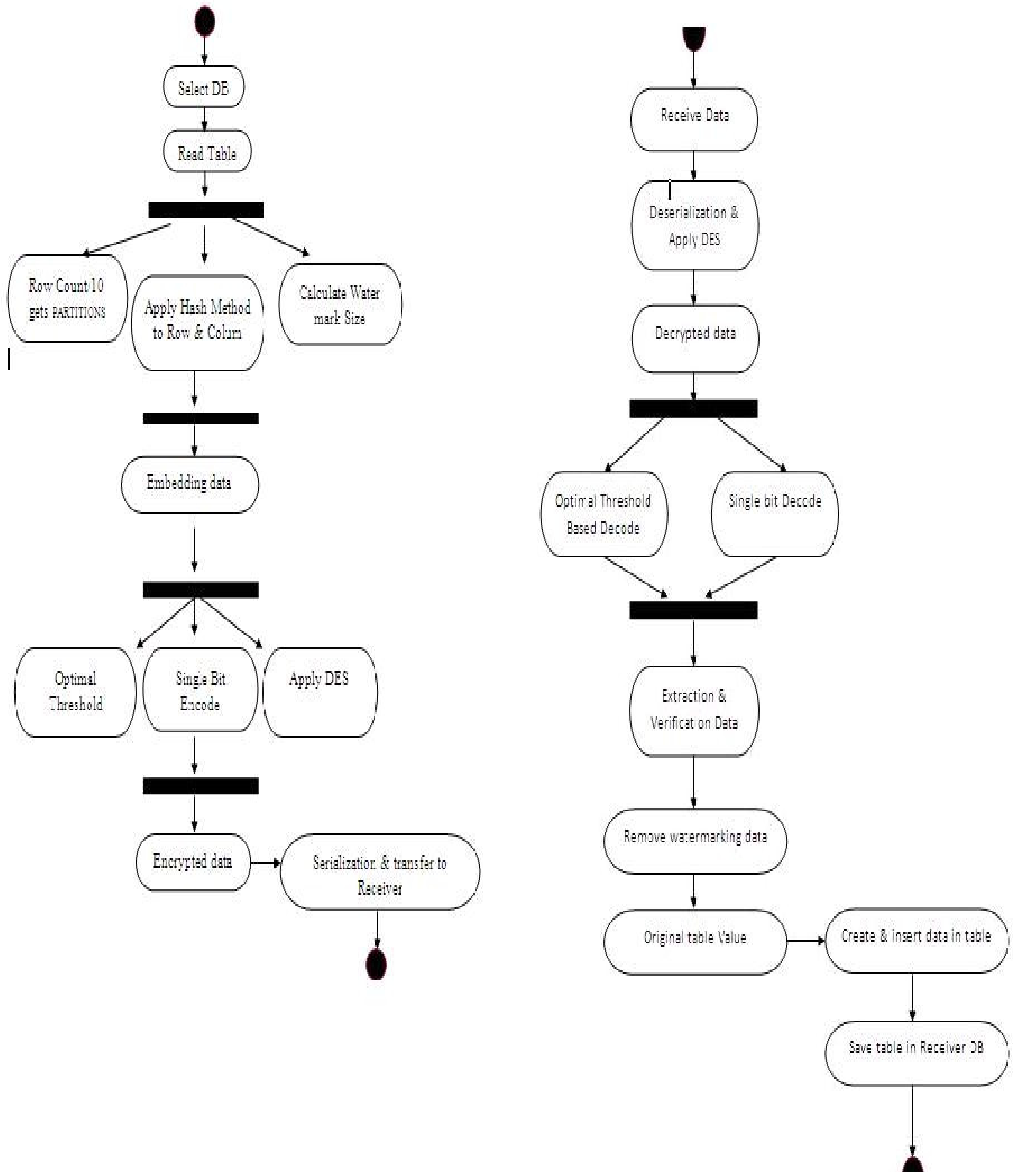

Select DB
Read Table
Row Count/10 gets PARTITIONS
Apply Hash Method to Row & Colum
Calculate Water mark Size
Embedding data
Optimal Threshold
Single Bit Encode
Apply DES
Encrypted data
Serialization & transfer to Receiver
Receive Data
Deserialization & Apply DES
Decrypted data
Optimal Threshold Based Decode
Single bit Decode
Extraction & Verification Data
Remove watermarking data
Original table Value
Create & insert data in table
Save table in Receiver DB

FIGURE: 2 & 3 Entity Relationship Diagram-Sender Activity and Receiver Activity

5. SYSTEM DESIGN

5.1 INPUT DESIGN

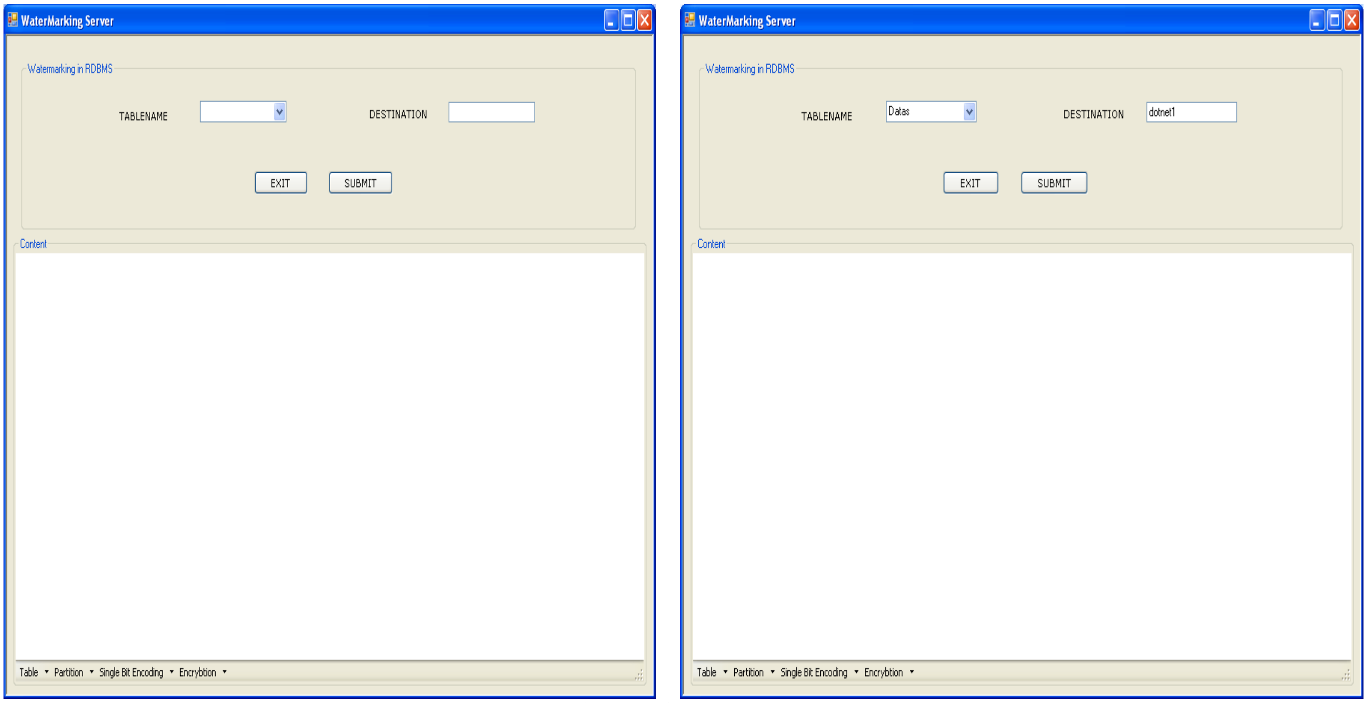
Input Design is the process of converting user oriented inputs to a computer based format. The quality of the system input determines the quality of the system output. Input design determines the format and validation criteria for data entering to the system. Input design is a part of overall system design, which requires very careful attention. If the data going into the system is incorrect then the processing and output will magnifies these errors. The analysis phase should consider the impact of the inputs on the system as a whole and on the other systems. In this project, the inputs are designed in such a way that occurrence of errors are minimized to its maximum since only authorized user are administrator can able to access this tool. The input is given by the administrator are checked at the entry form itself. So there is no chance of unauthorized accessing of the tool. Any abnormally found in the inputs are checked and handled effectively. Input design features can ensure the reliability of a system and produce results from accurate data or they can result in the production of erroneous information.

5.2 OUTPUT DESIGN

Computer output is the most important and direct source of information to the users. Designing the output should proceed in an organized, well thought out manner. The right output must be developed while ensuring that each output element is designed so that people will find easy to use the system. When analysts design the output, they identify the specific output that is needed to meet the information requirements. The success and failure of the system depends on the output, through a system looks attractive and user friendly, the output it produces decides upon the usage of the system. The outputs generated by the system are checked for its consistency, and output is provided simple so that user can handle them with ease. For many end users, output is the main reason for developing the system and the basis on which they will evaluate the usefulness of the application.

6. IMPLEMENTATION

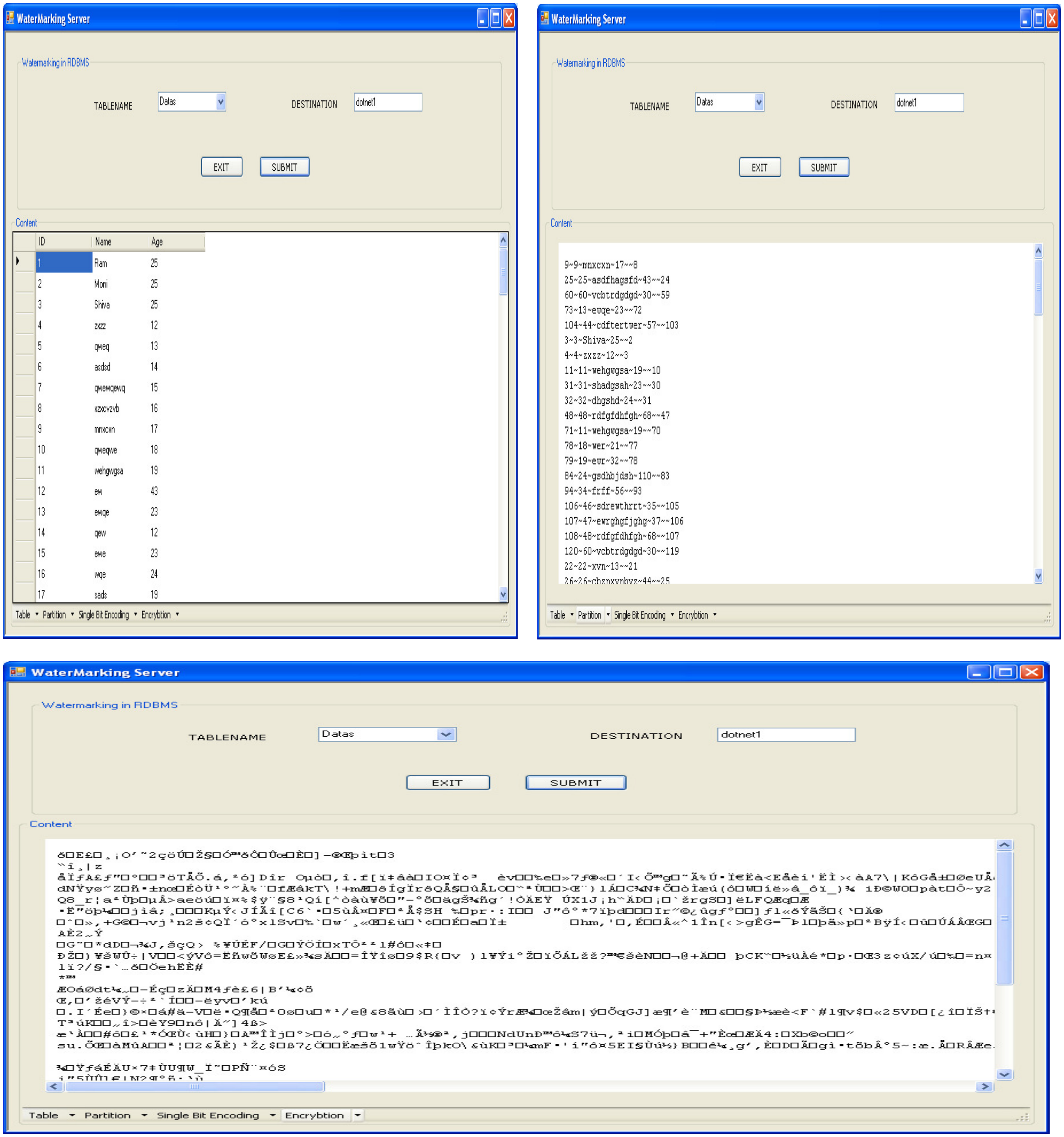
Once the system has been designed, the next step is to convert the designed one in to actual code, so as to satisfy the user requirements as excepted. If the system is approved to be error free it can be implemented.



TABLENAME Datas

DESTINATION dotnet1

Screen Shots



TABLENAME Datas

DESTINATION dotnet1

ID	Name	Age
1	Ram	25
2	Moni	25
3	Shiva	25
4	zxzz	12
5	qweq	13
6	asdsd	14
7	qwewqewq	15
8	xzxcvzvb	16
9	mnxcxn	17
10	qweqwe	18
11	wehgwgsa	19
12	ew	43
13	ewqe	23
14	qew	12
15	ewe	23
16	wqe	24
17	sads	19

TABLENAME Datas

DESTINATION dotnet1

```
9~9~mnxcxn~17~~8
25~25~asdfhagsfd~43~~24
60~60~vcbtrdgdgd~30~~59
73~13~ewqe~23~~72
104~44~cdftertwer~57~~103
3~3~Shiva~25~~2
4~4~zxzz~12~~3
11~11~wehgwgsa~19~~10
31~31~shadgsah~23~~30
32~32~dhgshd~24~~31
48~48~rdfgfdhfgh~68~~47
71~11~wehgwgsa~19~~70
78~18~wer~21~~77
79~19~ewr~32~~78
84~24~gsdhbjdsh~110~~83
94~34~frff~56~~93
106~46~sdrewthrrt~35~~105
107~47~ewrghgfjghg~37~~106
108~48~rdfgfdhfgh~68~~107
120~60~vcbtrdgdgd~30~~119
22~22~xvn~13~~21
26~26~cbznxvnbvz~44~~25
```

TABLENAME Datas

DESTINATION dotnet1

Screen Shots

7. CONCLUSION AND FUTURE ENHANCEMENT

The watermarking problem was formulated as a constrained optimization problem that maximizes or minimizes a hiding function based on the bit to be embedded. GA and PS techniques were employed to solve the proposed optimization problem and to handle the constraints. Furthermore, the data partitioning technique that does not depend on special marker tuples to locate the partitions and proved its resilience to watermark synchronization errors. Development of an efficient threshold-based technique for watermark detection that is based on an optimal threshold that minimizes the probability of decoding error. The watermark resilience was improved by the repeated embedding of the watermark and using majority voting technique in the watermark decoding phase. Moreover, the watermark resilience was improved by using multiple attributes. Media files can be transferred with secure and less packet loss. The ARMS system architecture with a focus on the extensions to the ISMA security standard to enable adaptive streaming of encrypted MPEG-4 content. Investigation on various optimizations in the coding and streaming to improve the bandwidth utilization while minimizing the distortion experienced by the clients in wired and wireless networks is going. Advances in compression and an increase in affordable bandwidth will allow for the streaming of higher resolution video and crisper audio. Developing better speech to text software and more adaptive technologies in streaming will offer greater accessibility.

8. REFERENCES

[1] R. Agrawal and J. Kiernan, "Watermarking Relational Databases," Proc. 28th Int'l Conf. Very Large Data Bases, 2002.

[2] M. Atallah and S. Lonardi, "Authentication of LZ-77 Compressed Data," Proc. ACM Symp. Applied Computing, 2003.

[3] M. Atallah, V. Raskin, C. Hempelman, M. Karahan, R. Sion, K. Triezenberg, and U. Topkara, "Natural Language Watermarking and Tamperproofing," Proc. Fifth Int'l Information Hiding Workshop, 2002.

[4] G. Box, "Evolutionary Operation: A Method for Increasing Industrial Productivity," Applied Statistics, vol. 6, no. 2, pp. 81- 101, 1957.

[5] E. Chong and S. Z_ ak, An Introduction to Optimization. John Wiley & Sons, 2001.

[6] D. Coley, "Introduction to Genetic Algorithms for Scientists and Engineers," World Scientific, 1999.

[7] C. Collberg and C. Thomborson, "Software Watermarking: Models and Dynamic Embeddings," Proc. 26th ACM SIGPLANSIGACT Symp. Principles of Programming Languages, Jan. 1999.

[8] I. Cox, J. Bloom, and M. Miller, Digital Watermarking. Morgan Kaufmann, 2001.

[9] E. Dolan, R. Lewis, and V. Torczon, "On the Local Convergence of Pattern Search," SIAM J. Optimization, vol. 14, no. 2, pp. 567-583, 2003.

[10] F. Hartung and M. Kutter, "Multimedia Watermarking Techniques," Proc. IEEE, vol. 87, no. 7, pp. 1079- 1107, July 1999.

[11] Darshana Mistry, "Comparison of Digital Water Marking methods," International Journal on Computer Science and Engineering - Vol. 02, No. 09, 2010, 2905-2909

[12] Dolley Shukla and Manisha Sharma, "WATERMARKING SCHEMES FOR COPY PROTECTION: A SURVEY," International Journal of Computer Science & Engineering Survey (IJCSES) Vol.3, No.1, February 2012

[13] K.Ganesan and Tarun Kumar Guptha, "Multiple Binary Images Watermarking in Spatial and Frequency Domains," Signal & Image Processing: An International Journal (SIPIJ) Vol.1, No.2, December 2010