Database Record Encryption for Multiuser Environment

By

Sasikala. V

Reg. No. 71204621038

Of

KUMARAGURU COLLEGE OF TECHNOLOGY

COIMBATORE

A PROJECT REPORT

Submitted to the

FACULTY OF INFORMATION AND COMMUNICATION ENGINEERING

In partial fulfillment of the requirements

for the award of the degree

of

MASTER OF COMPUTER APPLICATIONS

July, 2007

Certificate

Kumaraguru College of Technology

Coimbatore – 641006.

Department of Computer Applications

Bonafide Certificate

Certified that this project report titled **Database Record Encryption for Multiuser Environment** is the bonafide work of **Ms. Sasikala V** who carried out the research under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

Project Guide

Head of the Department

Submitted for the University Examination held on 3.7.2007

Internal Examiner

External Examiner 3/7/07

Company Certificate

ROFTR
FUTURE TECHNOLOGIES

Date :

06th June 2007

TO WHOMSOEVER IT MAY CONCERN

Sub.: Student Project Completion – Reg.

This is to inform that **Ms. Sasikala V** (04MCA38) of M.C.A. department of **Kumaraguru College of Technology, Coimbatore** has finished her final year project training in our concern under the guidance of our staffs in the following period: 18th December 2006 to 1st June 2007. The performance of the student during the training period is good and the project title is "**Database Record Encryption for Multi-User Environment**".

For ROFTR

Authorised Signatory

32-A, Bharathi Colony, Peelamedu, Coimbatore - 641 004.
Phone : 0422 4398003, E-mail : info@roftr.com, Website : www.roftr.com

Abstract

Abstract

In general, a database is a sharable resource among many users or applications. A multi-user application in a distributed system complicates the data security problem imposed upon a database. Hence, the issue of data security for database application is an actively researched field.

Researchers pointed out that the encryption scheme is an important measure for providing security for a database. It seems that an asymmetric cryptosystem (two key cryptosystem) is suitable for this case. Decryption keys represent the access rights to read the protected fields, and encryption keys represent the access rights to write the protected fields. Any database user having a decryption key of a protected field has the access right to read the field, and any user having an encryption key of a protected field has the access right to write the field. By the definition of an asymmetric cryptosystem, the computation of one key from another one is very difficult. Hence, the uses of an asymmetric cryptosystem can distinguish the access rights between read rights and write rights to a protected field.

The simplest database record encryption scheme based on the RSA public-key scheme and the RSA master key pair is the establishment of a RSA public-key system per field of a database that requires protection. Then, the RSA encryption keys represent the rights of write operation, and the RSA decryption keys represent the rights of read operation to the corresponding field. The RSA master key pair of all fields represents the right of the database manager. And, the RSA master keys of subsets of fields combine the access rights to the various combinations of fields. Based on the access control scheme provided from the concept of the master key our scheme provides an efficient method to allocate the access right to users according to their needs. In the database record encryption scheme, the RSA systems which are associated with each field of the database are established by the database manager.

Acknowledgement

Acknowledgement

I express my grateful thanks to our beloved principal, **Dr. Joseph V Thanikal** and our former principal **Dr. K.K.Padmanabhan**, Kumaraguru College of Technology, Coimbatore, for giving me an opportunity to take up this project.

I express my deep sense of gratitude to **Dr. S. Thangasamy**, Professor and Dean of Computer Science and Engineering department and **Dr. M. Gururajan,** Professor and Head of Department of Computer Applications for extending their help in providing all the facilities at college for the successful completion of the project.

I would like to express my sincere gratitude to **Mr. S. Ganesh Babu, MCA.,** Lecturer, Department of Computer Applications, for his guidance, support, cooperation and valuable suggestions during the course of this project.

I also thank our external guide **Mr. S. Prabhu,** Roftr Technologies, Coimbatore, for providing me with adequate technical support and for his excellent guidance during the course of the project.

Contents

Table of Contents

List of Figures

Introduction

CHAPTER 1

INTRODUCTION

1.1 SYSTEM OVERVIEW

The project titled "**Database Record Encryption for Multiuser Environment**" is done to enhance security in mulituser environment.

For the first few decades of their existence, a computer network was primarily used by university researchers for sending mail, and by corporate employees for sharing printers. Under this condition, security did not get a lot of attention. Nowadays, millions of ordinary citizens are using network for banking, shopping and filing tax returns, etc. Hence security is of major concern. Information is secured by encrypting the contents.

Cryptography is the art and science of keeping information secure from unintended audiences, of encrypting it. Cryptanalysis is the art and science of breaking it. The art of encoding and breaking is known as Cryptology. The two kinds of cryptography are:

- Symmetric Cryptosystem
- Asymmetric Cryptosystem

Symmetric cryptosystem uses the same key (secret key) to encrypt and decrypt a message whereas asymmetric cryptosystem uses one key (public key) to encrypt a message and a different key (private key) to decrypt it. Asymmetric cryptosystem is also called as public key cryptosystem.

Whitfield Deffie and Martin Heilman introduced the concept of public key cryptography in 1976. In this cryptosystem, each person gets a pair of keys, called as public and private key. Each person's public key is published while the private key is kept secret. Communication involves only public keys, and a private key is never transmitted or shared.

Anyone can send a confidential message by just using public information but the message can only be decrypted with a private key.

The simplest database record encryption scheme based on the RSA public-key scheme and the RSA master key pair is the establishment of a RSA public-key system per field of a database that requires protection. Then, the RSA encryption keys represent the rights of write operation, and the RSA decryption keys represent the rights of read operation to the corresponding field. The RSA master key pair of all fields represents the right of the database manager. And, the RSA master keys of subsets of fields combine the access rights to the various combinations of fields. Based on the access control scheme provided from the concept of the master key our scheme provides an efficient method to allocate the access right to users according to their needs. In the database record encryption scheme, the RSA systems which are associated with each field of the database are established by the database manager.

1.2 COMPANY PROFILE

Roftr was founded with the motto to serve the world with the solution that is quite sensible and feasible. They believe in three things. They are: Smart work, exceeding our clients' expectations positively and offering the complete enterprise solution on time. They are building value, making their customers profit not tomorrow, but today.

Roftr provides quality solution that are sensible and that makes a positive impact about you amongst your customers. They do have the professionals who know what they are doing and who truly believe in what they do, not just in their paychecks.

They don't have employees who are just developing the sites for hobbies or as part-time. They have true professionals who believe in experiments and values. They assure that you will find elegant craftsmanship in our solutions which makes sense and satisfies your need.

They have defined several benchmarks and measured great heights in very short period that speaks volume about their development. Now they are on the course to obtain the ISO-9002 certification, which is an independent universal standard to certify the standard maintained in the organization.

Roftr believes in long-term partnership with their clients. Clients and client satisfaction is our ultimate goal, so they will satisfy their clients at any cost.

1.2.1 Roftr Approach

Roftr believe in building a long-term partnership with our customers to iteratively implement solutions to their business needs. It is the belief that they must understand their customers business, what their goals are, and what

success means to them. It is not so much about the features and functionality, but how their customers will use the system to achieve their business needs. Their approach requires clear definition of business goals and most importantly the closure and assessment of how these goals were met - they are focused on what success means to them at the end of the day. The approach is to partner with our customers for implementations.

1.2.2 Professional Services

The services organization will work with each customer to understand their unique business needs and challenges and develops a plan that leverages the available resources within the customers organization and supplements those with key skills within Roftr, to minimize project risk and maximize business benefit. Providing skills related to system configuration, project management, report creation, course development, change management, we have successfully worked with the customers.

1.2.3 Hosting

They offer comprehensive-hosting services that allow extended enterprises to successfully deploy Valiant InfoTech applications with minimal IT infrastructure and resources. Through the dedicated application hosting services, the customers receive worldwide access.

System Study and Analysis

CHAPTER 2

SYSTEM STUDY AND ANALYSIS

2.1 OBJECTIVE OF THE PROJECT

- ✓ To enhance security for a multi user database environment.
- ✓ Encrypts each record in the database using key.
- ✓ Provides the key to decrypt the record and provides access rights only to acknowledged users.

2.2 EXISTING SYTEM

- ➢ File Encryption

The existing system was done to encrypt the whole database as a whole. In such cases, the system will not allow multiple users to access the database. If multiple users are allowed, then the key used for encryption will not be kept secure and the system will lack in security. Therefore, the system will face various drawbacks. To overcome such drawbacks, we proceed with the proposed system.

2.2.1 Drawbacks of Existing System

- ✓ Multi users cannot access the database.
- ✓ Does not provide much security.
- ✓ The key generated for encryption cannot be kept secure.

2.3 PROPOSED SYSTEM

- Database Record Encryption

The proposed system is done to encrypt each record in the database separately. This allows multiple users to access and work with the same database. The data is kept much secure. Only the data entered by that particular user can be viewed by the user. In that case, security plays a major role. The database and the contents of the database are secure. The large random prime numbers are used for generating the key, so there is no possibility of unauthorized users to find the key.

2.3.1 Advantages of Proposed System

- ✓ Each record can be encrypted separately. So multi users can access the database.
- ✓ Provides much security to the database.
- ✓ The key generated for encryption can be kept much secure.

2.4 FEASIBILITY ANALYSIS

Feasibility analysis is the measure of how beneficial or practical the development of Information System will be to the Organization. Once the problem is explained information is gathered about the system to test whether the system is viable Technically, Financially and Operationally. Thus, feasibility study is carried out in three phases as follows:

2.4.1 Technical Feasibility

Technical Feasibility is the measure of practicality of a specific technical solution and the availability of technical resources and expertise. It centers on the existing computer system (hardware, software, etc.) and to what extent it can support the new addition.

The proposed system is to be developed using .Net, which is one of the leading technologies in the market. These resources are easily available and the company does not need to acquire any development license. These features of the selected technology are quite beneficial to the proper functioning of the system in different environments.

2.4.2 Operational Feasibility

Operational Feasibility asks if the system will work when it is developed and installed. It checks for the support of the management, the current business methods, user's involvement and their attitude towards the proposed system, etc.

The proposed system has found encouraging support from the company staff and management as it will be of great use to them. The users of the project finds the system very secure.

2.4.3 Economic Feasibility

Economic Feasibility is the measure of the cost-effectiveness of the proposed system. The investment to be made in the proposed system must prove a good investment to the organization by returning benefits equal to or exceeding the costs incurred in developing the system.

The proposed benefits of the system will outweigh the costs to be incurred during system developed since the system does not require procurement of additional hardware facilities it is economically feasible.

Development Environment

CHAPTER 3

DEVELOPMENT ENVIRONMENT

3.1 HARDWARE REQUIREMENTS

Processor : Pentium IV Processor/Board or higher

RAM : 256MB RAM

Memory : 64 Megabytes

Hard Disk : 80GB HDD

Monitor : 15" Monitor

I/O Devices : Standard Keyboard/Mouse

3.2 SOFTWARE REQUIREMENTS

Front End : Microsoft VB.NET 2003

Back End : Oracle 9i and Microsoft Access

Operating System : Windows

3.3 PROGRAMMING ENVIRONMENT

3.3.1 RSA Algorithm

The RSA algorithm is named after Ron **R**ivest, Adi **S**hamir and Len **A**dleman, who invented it in 1977. The RSA algorithm can be used for both public key encryption and digital signatures. Its security is based on the difficulty of factoring large integers.

3.3.1.1 Key Generation Algorithm

1. Generate two large random primes, p and q, of approximately equal size such that their product $n = pq$ is of the required bit length, e.g. 1024 bits.
2. Compute n = pq and (φ) phi = (p-1)(q-1).
3. Choose an integer e, 1 < e < phi, such that gcd(e, phi) = 1.
4. Compute the secret exponent d, 1 < d < phi, such that ed ≡ 1 (mod phi).
5. The public key is (n, e) and the private key is (n, d). The values of p, q, and phi should also be kept secret.

- n is known as the *modulus*.
- e is known as the *public exponent* or *encryption exponent*.
- d is known as the *secret exponent* or *decryption exponent*.

3.3.1.2 Encryption

Sender A does the following:-

1. Obtains the recipient B's public key (n, e).
2. Represents the plaintext message as a positive integer m < n.
3. Computes the ciphertext $c = m^{\wedge e} \bmod n$.
4. Sends the ciphertext c to B.

3.3.1.3 Decryption

Recipient B does the following:-

1. Uses his private key (n, d) to compute $m = c^{\wedge d} \bmod n$.
2. Extracts the plaintext from the integer representative m.

3.3.1.4 Summary of RSA:

- n = pq where p and q are distinct primes.
- phi, $\varphi = (p-1)(q-1)$
- e < n such that gcd(e, phi)=1
- $d = e^{\wedge -1} \bmod phi$.
- $c = m^{\wedge e} \bmod n$.
- $m = c^{\wedge d} \bmod n$.

3.3.2 Visual Basic .Net

Visual Basic .Net (VB .Net) is an object-oriented computer language that can be viewed as an evolution of Microsoft's Visual Basic (VB) implemented on the Microsoft .Net framework. Its introduction has been controversial, as significant changes were made that broke backward compatility with VB and caused a rift within the developer community.

The great majority of VB .Net developers use Visual Studio .Net as their integrated development environment (IDE).

Microsoft Visual Basic .Net used as front end tool. The reason for selecting Visual Basic .Net as front end tool as follows:

- Visual Basic .Net has flexibility, allowing one or more language to interoperate to provide the solution. This Cross Language Compatibility allows to do project at faster rate.
- Visual Basic .Net has Common Language Runtime, that allows the entire component to converge into one intermediate format and then can interact.
- Visual Basic .Net has provide excellent security when your application is executed in the system
- Visual Basic .Net has flexibility, allowing us to configure the working environment to best suit our individual style. We can choose between a single and multiple document interfaces, and we can adjust the size and positioning of the various IDE elements.
- Visual Basic .Net has Intelligence feature that make the coding easy and also dynamic help provides very less coding time.
- The working environment in Visual Basic .Net is often referred to as Integrated Development Environment because it integrates many different functions such as design, editing, compiling and debugging

within a common environment. In most traditional development tools, each of separate program, each with its own interface.

- The Visual Basic .Net language is quite powerful – if we can imagine a programming task and accomplished using Visual Basic .Net.
- After creating a Visual Basic .Net application, if we want to distribute it to others we can freely distribute any application to anyone who uses Microsoft windows. We can distribute our applications on disk, on CDs, across networks, or over an intranet or the internet.
- Toolbars provide quick access to commonly used commands in the programming environment. We click a button on the toolbar once to carry out the action represented by that button. By default, the standard toolbar is displayed when we start Visual Basic. Additional toolbars for editing, form design, and debugging can be toggled on or off from the toolbars command on the view menu.
- Many parts of Visual Basic are context sensitive. Context sensitive means we can get help on these parts directly without having to go through the help menu. For example, to get help on any keyword in the Visual Basic language, place the insertion point on that keyword in the code window and press F1.
- Visual Basic interprets our code as we enter it, catching and highlighting most syntax or spelling errors on the fly. It's almost like having an expert watching over our shoulder as we enter our code.

3.3.3 Microsoft Access

Access is a versatile application for creating databases. Although it may not be as powerful as industrial strength database management systems, Access is very popular due to the vast number of features it provides. Many businesses also favor Access because it is integrated with all the other Microsoft Office products.

As with all of the Microsoft Office products, there have been numerous updates to Access over the years. The most recent version of Access, known as Access 2000, was made available to the public in June of 1999. New features of this version include easy organizing and sharing of work over intranets and networks, integrating databases with the World Wide Web, interface improvements, and many more.

3.3.4 Oracle

Oracle is a fourth generation relational database management system. In general, a database management system (DBMS) must be able to reliably manage a large amount of data in a multi-user environment so that many users can concurrently access the same data. All this must be accomplished while delivering high performance to the users of the database. A DBMS must also be secure from unauthorized access and provide efficient solutions for failure recovery. The Oracle Server provides efficient and effective solutions for the major database features.

Oracle is an object relational database. A relational database is an extremely simply way of thinking about and managing the data used in a business. It is nothing more than a collection of tables of data. Tables are with column headings and rows of information. Oracle database supports all of the features of a relational database and object oriented concepts and features.

Oracle consists of many tools that allow you to create an application with ease and flexibility. You must determine how to implement your requirements using the features available in Oracle, along with its tools. The features and tools that you choose to use to implement your application can significantly affect the performance of your application.

A database is simply a container of information. To find an example of a database that most people use everyday, you need look no further than your address book or the yellow pages. Websites use databases to keep track of information, mostly for Internet commerce or recorded user information and usage. In fact, before we bring computers into this, lets cover some basic database terms that are universal.

Oracle implements the object-type system as an extension of the relational model. The object-type interface continues to support standard relational database functionality such as queries (SELECT...FROM...WHERE), fast commits, backup and recovery, scalable connectivity, row-level locking, read consistency, partitioned tables, parallel queries, cluster database, export and import, and loader. The result is an object-relational model, which offers the intuitiveness and economy of an object interface while preserving the high concurrency and throughput of a relational database.

Oracle object types are user-defined types that make it possible to model real-world entities such as customers and purchase orders as objects in the database.

Oracle object technology is a layer of abstraction built on Oracle relational technology. New object types can be created from any built-in database types and any previously created object types, object references, and collection types. Metadata for user-defined types is stored in a schema that is available to SQL, PL/SQL, Java, and other published interfaces.

Object types and related object-oriented features such as variable-length arrays and nested tables provide higher-level ways to organize and access data in the database. Underneath the object layer, data is still stored in columns and tables, but you are able to work with the data in terms of the real-world entities, such as customers and purchase orders, that make the data meaningful.

Internally, statements about objects are still basically statements about relational tables and columns, and we can continue to work with relational data types and store data in relational tables as before. But now we have the option to take advantage of object-oriented features too. We can begin to use object-oriented features while continuing to work with most of our data relationally, or we can go over to an object-oriented approach entirely. For instance, we can define some object data types and store the objects in columns in relational tables, which enable us to extend the system built-in types with user-defined ones. We can also create object views of existing relational data to represent and access this data according to an object model. Or we can store object data in object tables, where each row is an object.

System Design and Development

CHAPTER 4

SYSTEM DESIGN AND DEVELOPMENT

4.1 ELEMENTS OF DESIGN

System Design is the most creative and challenging phase in the development of a software system. Design implies to a description of the final system and the process by which it is developed. The first step is to determine what input data is needed for the system and then to design a database that will meet the requirements of the proposed system. The next step is to determine what outputs are needed from the system and the format of the output to be produced.

Design is concerned with identifying software components specifying relationships among components. Specifying software structure and providing blue print for the document phase.

Modularity is one of the desirable properties of large systems. It implies that the system is divided into several parts. Design will explain software components in detail. This will help the implementation of the system. Moreover, this will guide the further changes in the system to satisfy the future requirements.

The steps carried out in the design phase are as follows:

- Modular Design
- Input Design
- Output Design

4.1.1 Modular Design

It is always difficult for any System Development team to grasp a system without breaking it into several smaller systems. These smaller systems will be a part of the original system yet they will be independent in the sense that they will incorporate within them the major functionalities of the proposed system.

A software system is always divided into several subsystems, which make it easier to develop and perform tests on the whole system. The subsystems are known as the modules and the process of dividing an entire system into subsystems is known as Decomposition.

The modules identified in the Database Record Encryption system are as follows:

- Database Management
- GUI Creation
- User Management
- RSA Cryptography
- Multi-User Record Encryption

4.1.1.1 Database Management

Database Management deals with handling the database and to perform various other operations in the database. This module deals with database design and creation. Basic operations with the database can also be performed. This includes data source creation, connection establishment. New table creation using query execution and update statements.

4.1.1.2 GUI Creation

Graphical User Interface (GUI) is needed to interact with the users. With the help of GUI, the user can carry out all the operations with the project. GUI acts as an interface between the user and the project. VB .Net components are used to create the GUI.

4.1.1.3 User Management

This module deals with managing the users of the project. The administrator has the full control of the users. The administrator can manage with user creation, editing and deletion of each user. When a new user is created, a private key is assigned for each user. This key can be used for encryption and decryption of the project.

4.1.1.4 RSA Cryptography

This module deals with implementation of RSA Algorithm. This deals with implementation prime number generation. Greatest Common Divisor (GCD) generation algorithm and RSA cryptography algorithm are also implemented as a part of RSA algorithm.

4.1.1.5 Multi-User Record Encryption

Each user is assigned with a private key. This private key is used to create cipher text from a database field value. Only the data entered by the current user can be decrypted and viewed by the respective user. Record viewing can be done only by the user.

4.1.2 Input Design

Input design is the link between the information system and the user. It comprises developing specification and procedures for data preparation and those steps that are necessary to put transaction data into a usable form for processing data entry. Instructing the computer to read data from a written or printed document can achieve the activity of putting data into the computer for processing or it can occur by having people key data directly into the system. The design of inputs focuses on controlling the amount of inputs required, controlling errors, avoiding delay, avoiding extra steps and keeping the process simple.

The input design requirements such as user friendliness, consistent format and interactive dialogue for giving the right message and help for the user at right time are also considered for the development of the project.

4.1.3 Output Design

Output generally refers to the results and information that are generated by the system. 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. Most end-users will noticeably operate the information system or enter data through workstation, but they will use the output from the system.

When designing output, system analyst must accomplish the following: Determine what information to present, decide whether to display, print the information and select the output medium.

The user can view only the data entered by the respective user. Other users cannot view the information entered by other users. Only the current user who is working with the login, can view the information entered. The information can be viewed by the decrypting the record. Records entered by other users will be in encrypted form.

4.2 Data Flow Diagram

Data flow diagrams are graphical representation depicting information regarding the flow of control and the transformation of data from input to output. The DFD may be used to represent the system or software at any level of abstraction. In fact, DFD can be partitioned into levels.

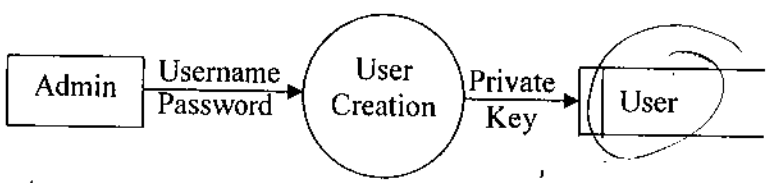


Figure : 4.2.1 – Key Generation

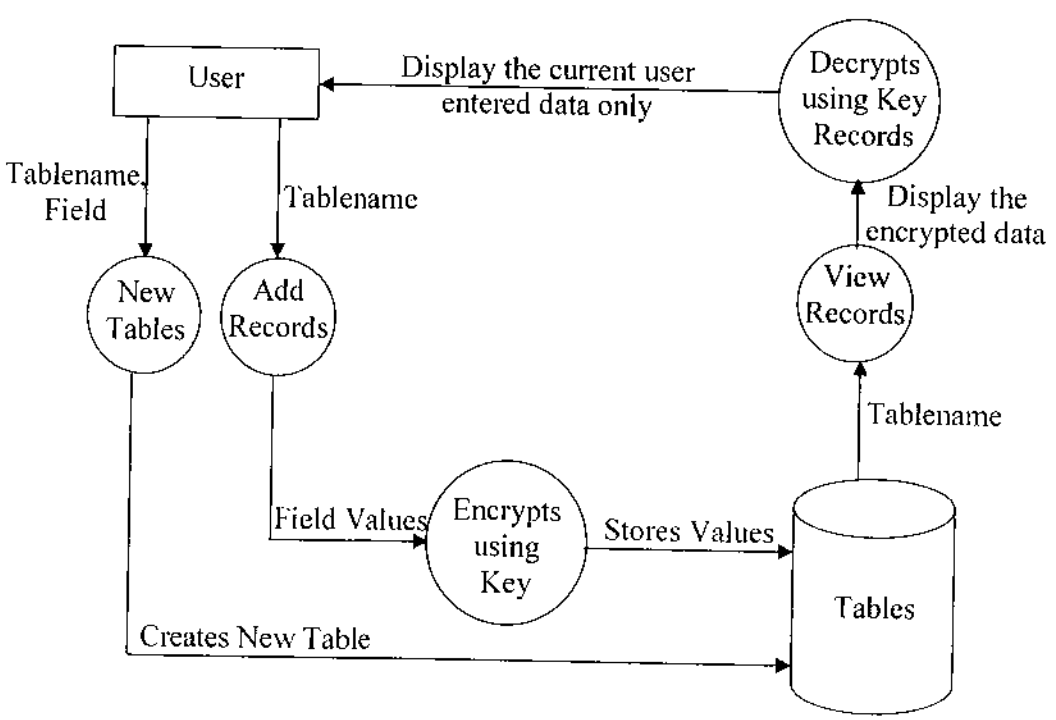


Figure : 4.2.2 – Encryption and Decryption Process

Implementation

CHAPTER 5

IMPLEMENTATION

5.1 SYSTEM IMPLEMENTATION

Implementation is the stage of the project when the theoretical is turned into a working system. If the implementation stage is not clearly planned and controlled, it can cause chaos. The term implementation has different meanings, ranging from the conversion of the basic application to a compatible replacement of a computer system. Implementation is used here to mean the process converting a new or a revised system design into an operational one. During the implementation stage we convert the detailed code in a programming language.

5.1.1 Implementation Procedures

The implementation procedures include pre and post implementation. The pre-implementation denote procedures that are done before implementation. Here the changes are made manually and given to the programmer. The programmer implements these changes using post-implementation procedures.

A post-implementation review measures the system performance against predefined requirements. It is an evaluation of system in terms of extent to which the system accomplishes stated objectives and actual project costs exceed initial estimates. It is usually a review of major problems that need converting and those that surfaced during implementation phase.

For the installation of the software the setup of the software has to be created which will help us to install all the components used in the project and with the help of which only the work can run successfully. The setup wizard will

setup the product. This will automatically includes all files to setup kit. The database entry and updating should be done manually. Since we place the files in a network server there is a chance to miss or damage the files due to the trespassing. So, we have to keep the backup copies of setup files to compact disks or in the hard disks and run the file setup which will install all the required components to computer. These files are stand-alone and don't need development software to access it.

5.1.2 User Training

After the system is implemented the user is informed and all the modules of the system are explained. Documentation is given for any future reference of the software.

5.1.3 Operational Documentation

Computer software includes the software code for a system and all the supporting documents generated during analysis, design, implementation, testing and maintenance of the system. Internal documentation includes standard prologues for compilation units and sub programs, the self documenting aspects of the source code, and the internal comments embedded in the source code.

5.2 SYSTEM VERIFICATION

System Verification answers the question "Am I building the product right?" It includes the review of interim work steps and interim deliverables during a project to ensure they are acceptable. Verification also determines if the system is consistent, adheres to standards, uses reliable techniques and prudent practices, and performs the selected functions in the correct manner. In data access, it verifies whether the right data is being accessed, in terms of the right place and in the right way.

For e.g., the drop downs gather data from the database, so each dropdowns should be verified whether they are bound to the correct database field. It is done during development of the key artifacts. Verification is a demonstration of consistency, completeness, and correctness of the software at each stage and between each stage of the development life cycle. In Result Analysis, verification is done during the development itself. Each database bindings are verified after binding to test whether the control is bound to the right data field.

5.3 SYSTEM VALIDATION

Validation answers the question "Am I building the right product?" This checks whether the developer is moving towards the right product, whether the development is moving towards the actual intended product that was agreed upon in the beginning. Validation also determines if the system complies with the requirements and performs functions for which it is intended and meets the organization's goals and user needs. It is traditional and is performed at the end of the project. In data access, it checks whether we are accessing the right data, in terms of data required to satisfy the requirement.

Validation is performed after a work product is produced against established criteria ensuring that the product integrates correctly into the environment. It determines the correctness of the final software product by a development project with respect to the user needs and requirements.

Functional validation is done to check whether each of the functions are done correctly as expected in every page. Each control in a Screen is designed to do some function. These functions are checked against the requirements stated for them. Clicking the Edit icon should allow one to edit the contents that are being currently displayed. This level of validation can continue to all the

controls in the system. This checking is usually done after the system is developed so that all activities that are affected can be checked.

Field level validation is done to check whether each of the fields either accepts the data as expected and do the client side validation of data entered. For e.g. a field level validation on a text box would check against the type of data entered and follow rules such as length of entry etc. The data type validation checks are conducted after the form is submitted. It takes place in the Action Form class of the struts framework. If the validation check fails then the processing stops and the control returns back to the original form that was submitted.

The validation is done in a step by step process. First the screen is loaded with the controls. When the user moves between controls on the screen, the validation events for the control that lost the focus are fired and appropriate error messages (if any) are displayed. If the user generates a form save request, the entire form is evaluated for any validation controls that are not valid. If even one control is not valid, the form will not be submitted.

5.4 SYSTEM MAINTENANCE

The term 'Software Maintenance' is to describe the software engineering activities that occur following delivery of a software product to the customer. The maintenance phase of the software is the time period in which a software product performs useful work.

Maintenance activities involve making enhancement to the software products, adapting products to new environments and correcting problems. Adaptation of the software to a new environment may involve moving the software to different machine, or for instance, modifying the software to

accommodate to a new protocol or additional disk drive. Problem correction involves modification and validation of software to correct errors.

5.4.1 Corrective Maintenance

Maintenance activity that could take place since it is not reasonable to assume that testing will uncover all errors in such a large system, by process of including the diagnosis and correction of one or more errors.

5.4.2 Adaptive Maintenance

Maintenance that occurs device rapid change encountered in every aspect of computing. Therefore modification of software to properly interface with changing environment is necessary.

5.4.3 Perfective Maintenance

Recommends for new capabilities, modifications to the existing functions and generates enhancement when the system is used.

5.4.4 Preventive Maintenance

Maintenance activities occur when system is changed to improve future maintainability or reliability.

5.5 SYSTEM TESTING

It is the stage of implementation, which ensures that system works accurately and effectively before the live operation commences. It is a confirmation that all are correct and opportunity to show the users that the system must be tested with text data and show that the system will operate successfully and produce expected results under expected conditions.

Before implementation, the proposed system must be tested with raw data to ensure that the modules of the system work correctly and satisfactorily. The system must be tested with valid data to achieve its objective.

The purpose of system testing is to identify and correct errors in the candidate system. As important as this phase is, it is one that is frequently compromised. Typically, the project the schedule or the user is eager to go directly to conversion.

Actually, testing is done to achieve the system goal. Testing is vital to the parts of the system are correct; the goal will be successfully achieved. Inadequate testing or non testing leads to errors, which may not appear until months later. This creates two problems:

A small system error can conceivably exploded into much larger problem. Effectively early in the process translates directly into long term cost savings from a reduced number of errors.

5.5.1 Unit Testing

A program represents the logical elements of a system. For a program to run satisfactorily, it must compile and test data correctly and tie in properly with other programs. Achieving an error free program is the responsibility of the programmer. Program testing checks for two types of errors: syntax and logical. Syntax error is a program statement that violates one or more rules of the language in which it is written. A improperly defined field dimension or error messages generated by the computer. Logic error deals with incorrect data fields, out-of range items, and invalid combinations. Since diagnostics do not determine logic errors the programmer must examine the output carefully.

When a program is tested, the actual output is compared with the expected output. When there is a discrepancy the sequence of instructions must be traced to determine the problem the process is facilitated by breaking the program down into self-contained portions, each of which can be checked at certain key points. The idea is to compare program values against desk-calculated values to isolate the problem.

Unit testing has been performed on all the form modules. The syntax and logical errors have been corrected then and there. All the syntax errors have been rectified during compilation. The output has been tested with the manual.

5.5.1.1 Unit Test Consideration

The module "interface" is tested to ensure that information properly flows in and out of the programs unit under test. The local data structures are examined to ensure that the data stored temporarily maintains its integrity during all steps in an algorithm execution.

Boundary conditions are tested to ensure that the modules operated properly at boundaries established to limit or restrict processing. All independent paths through the control structures are exercised to ensure that all statements in modules have been executed at least once. Finally paths are tested.

5.5.1.2 Unit Test Performance

Unit test is considered an equivalent to the coding step. After the source level code has been developed, reviewed and verified for correct syntax, unit test case begins, since a module is not stand-alone program. Drivers or stub software must be developed for each unit test.

5.5.2 Integration Testing

Programs are invariably related to one another and interact in the total system. Each program is tested to see whether it conforms to related programs in the systems. Each portion of the system is tested against the entire module with both the test data and the live data before the entire system is tested as a whole.

Integration testing is systematic techniques for conducting the program structure. While at the same time conducting tests to uncover errors associated with the interfacing.

There are two types of Integration steps,

- Top down Integration.
- Bottom up Integration

5.5.3 Validation Testing

The validation testing is performed for all the data in the system. The data are completely validated according to the companies requested and requirement.

In these testing, software is completely assembled as package, interfacing errors have been uncovered and correction testing begins after each one of the two possible conditions exists. They are,

The function or performance characteristic's is confirm specification and are accepted. A deviation from the specification is uncovered and efficiency list is created.

5.5.4 Output Testing

Various outputs have been generated by the system. The system generated output and the desk-calculated values have been compared. All the output is perfect as the company desires. It begins with low volumes of transactions based on live tone. The volume is increased until the maximum level for each transaction type is reached. The total system is also tested for recovery and fallback, after various major failures to ensure that no data are lost during the emergency time.

Conclusion

CHAPTER 6

CONCLUSION AND FUTURE ENHANCEMENT

6.1 CONCLUSION

In this project, we have pointed out how to directly update the protected fields. We support the user directly decrypting and encrypting the protected fields. We use the concept of the RSA master key to reduce the key management problem. The use of master keys also provides the feasible combination of access rights according the user's need and security policy of the database. The mechanism of generating key can solve problems like access control and password authorization.

6.2 FUTURE ENHANCEMENT

The project has been developed to decrypt only the records which the user entered. This can be enhanced in the future with more security by providing access rights to the user and to view the records can be according to the access rights of the user.

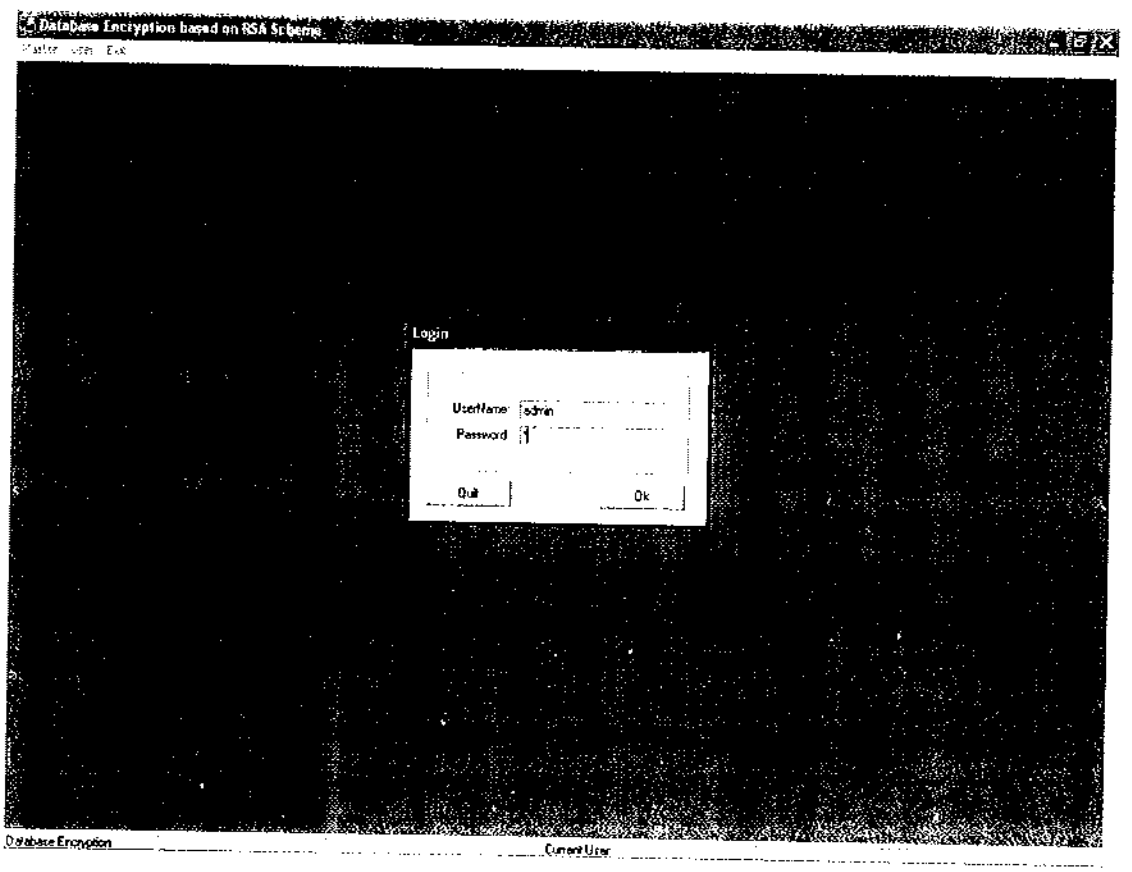
Further, in the future, field-oriented encryption system with user master keys can be used to encrypt the database fields, so that the decryption involved correspond to the access rights of multiple fields.

Appendices

APPENDICE

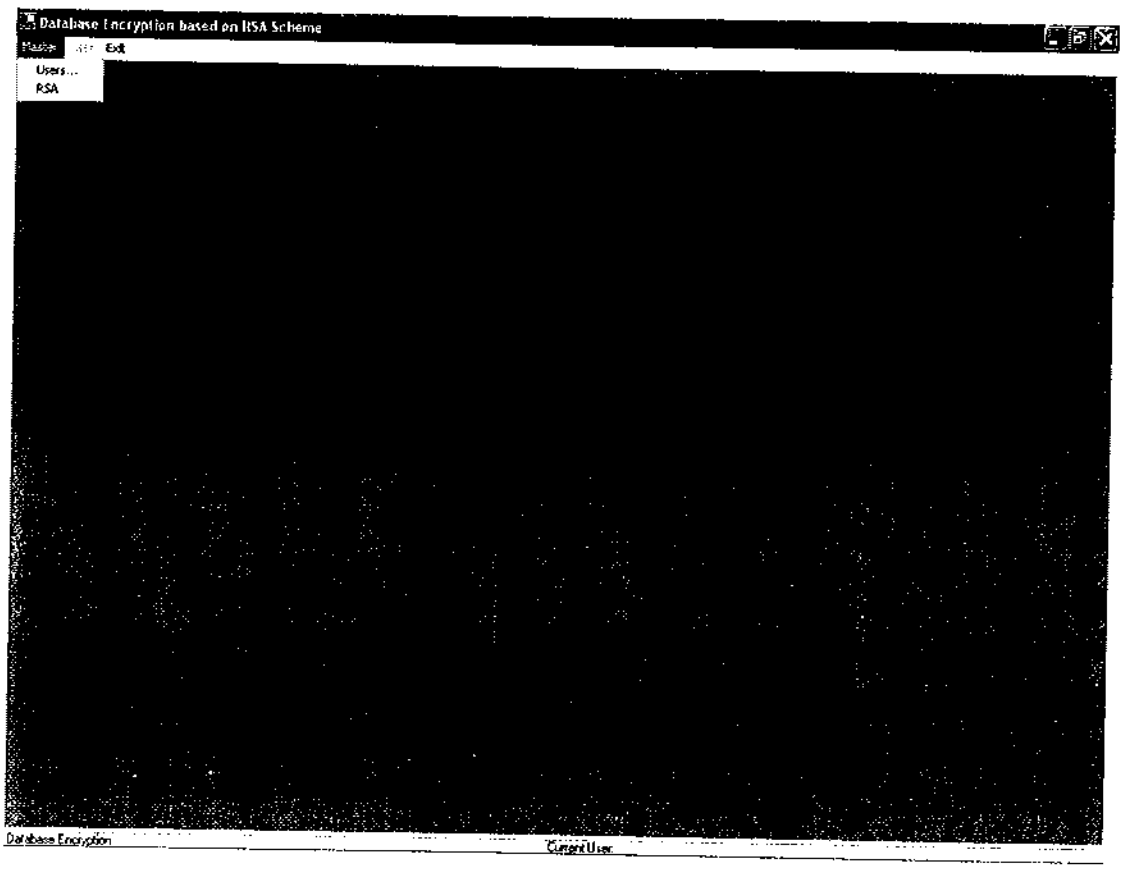
Administrator Login

This form is used by the administrator to log in to the system.



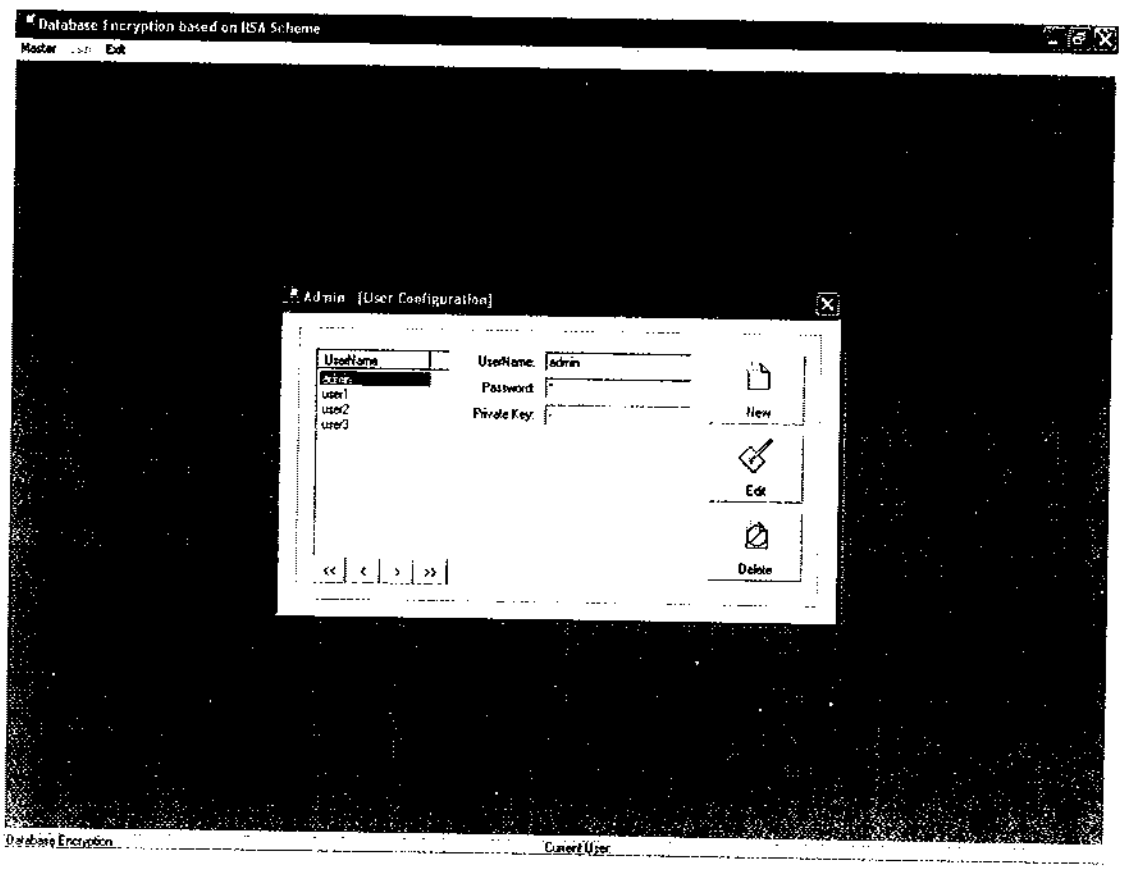
MDI Form

This form displays the operations that the administrator can perform.



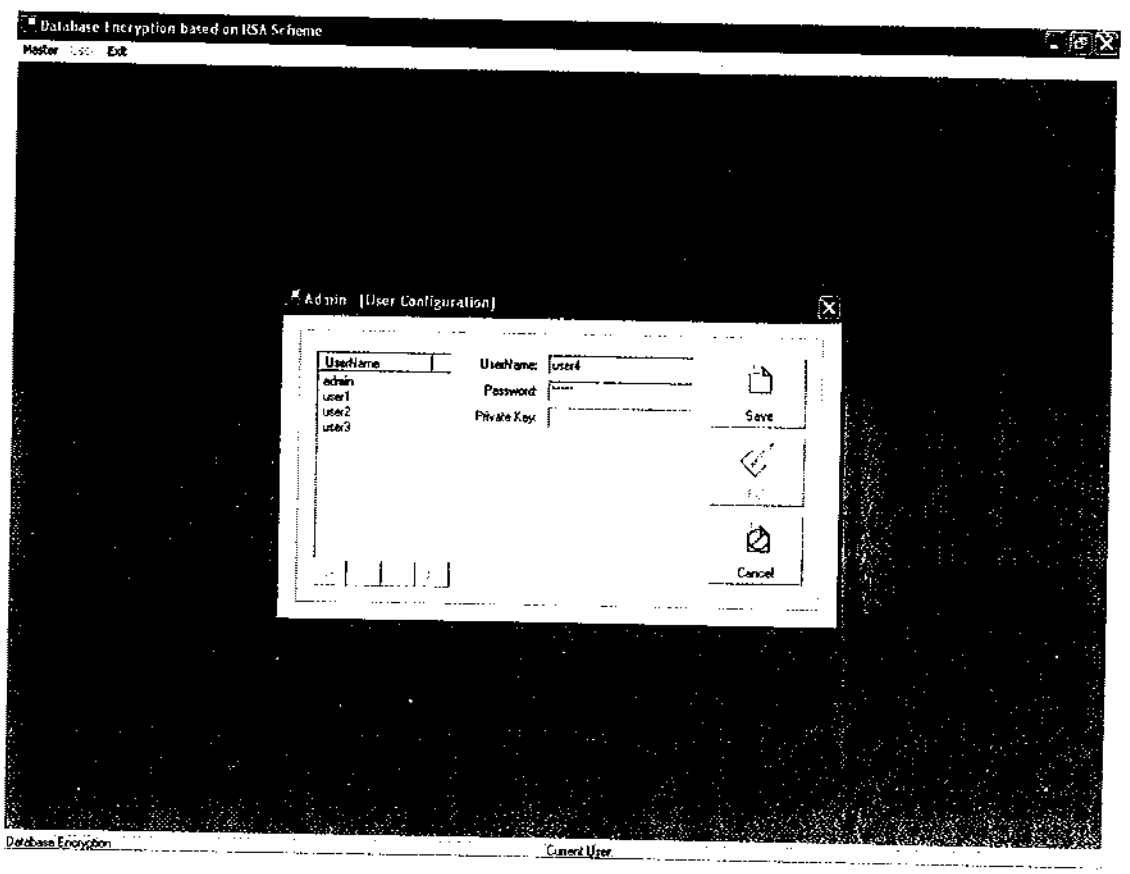
User Configuration Form

This form is used by the administrator to create new users, edit and delete existing users.



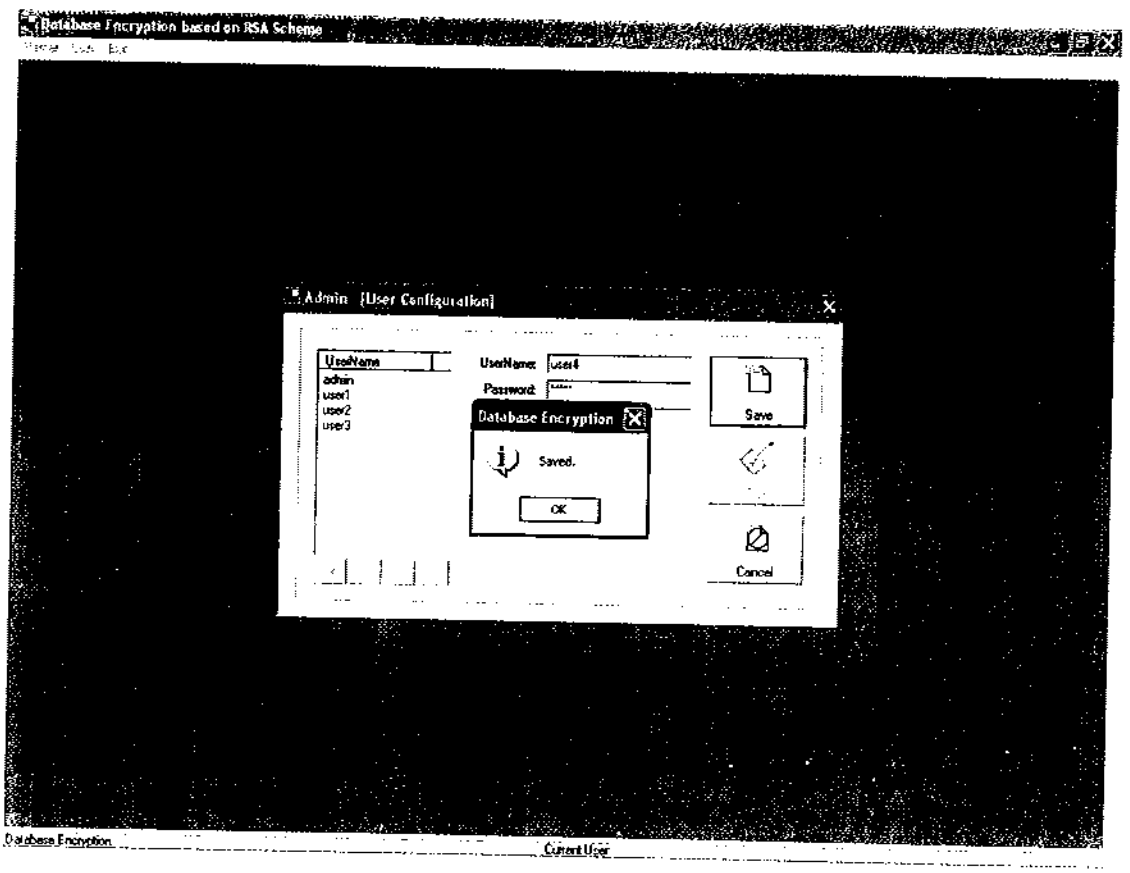
User Configuration Form

This form creates a new user by specifying username and password for the user.



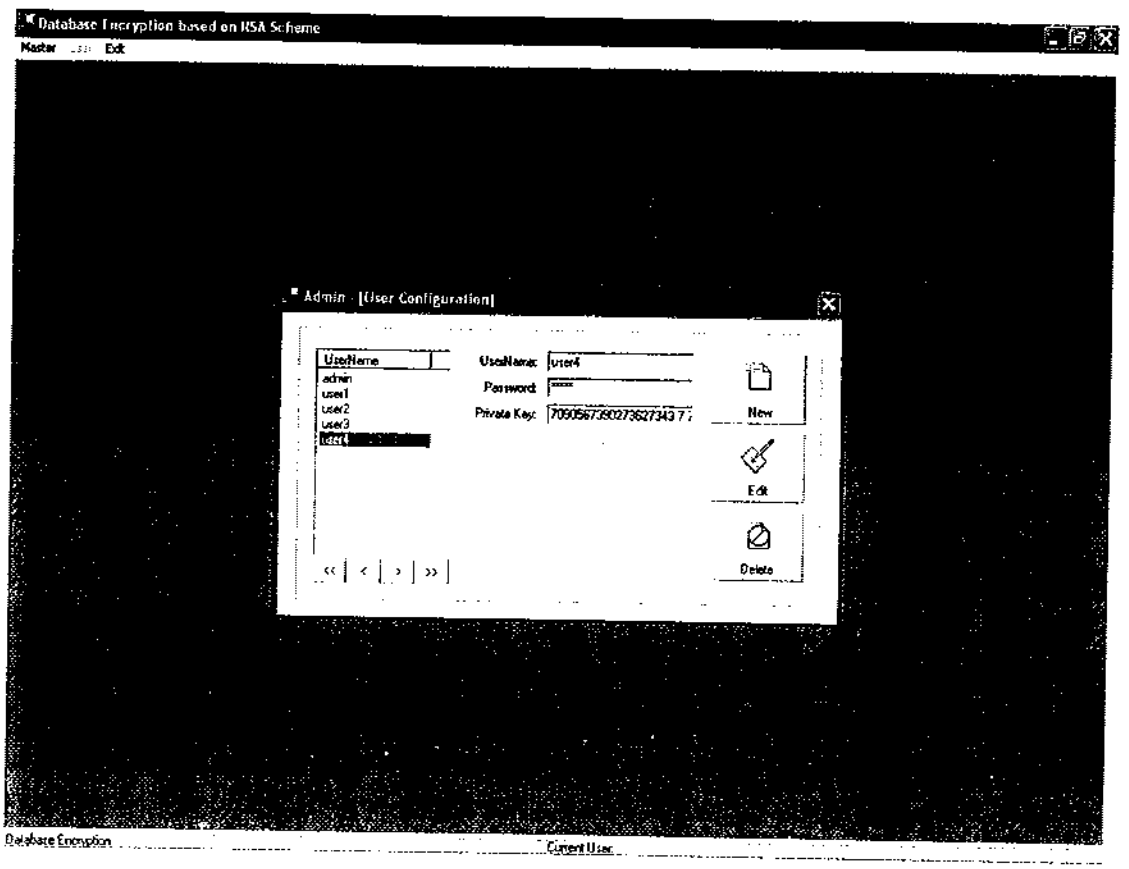
User Configuration Form

This form saves the username and password entered by the administrator.



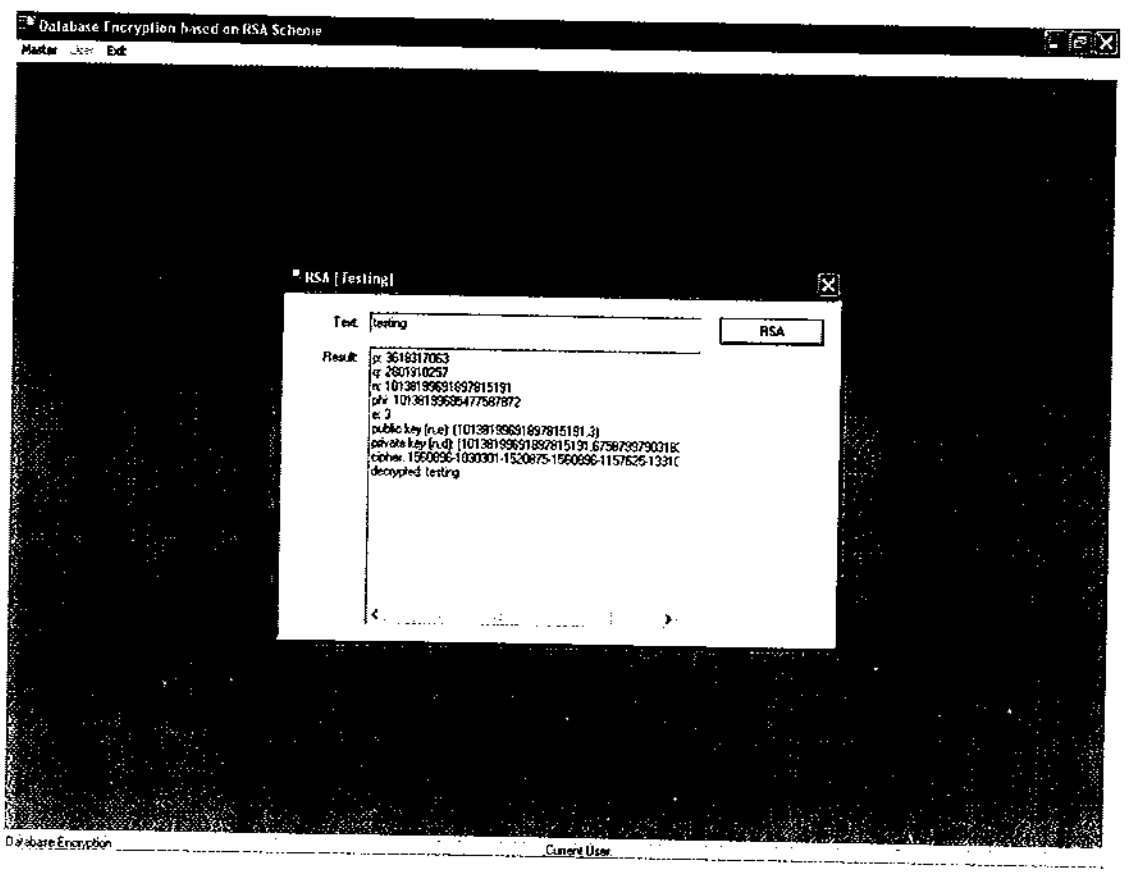
User Configuration Form

This form displays the private key generated when a new user is created.



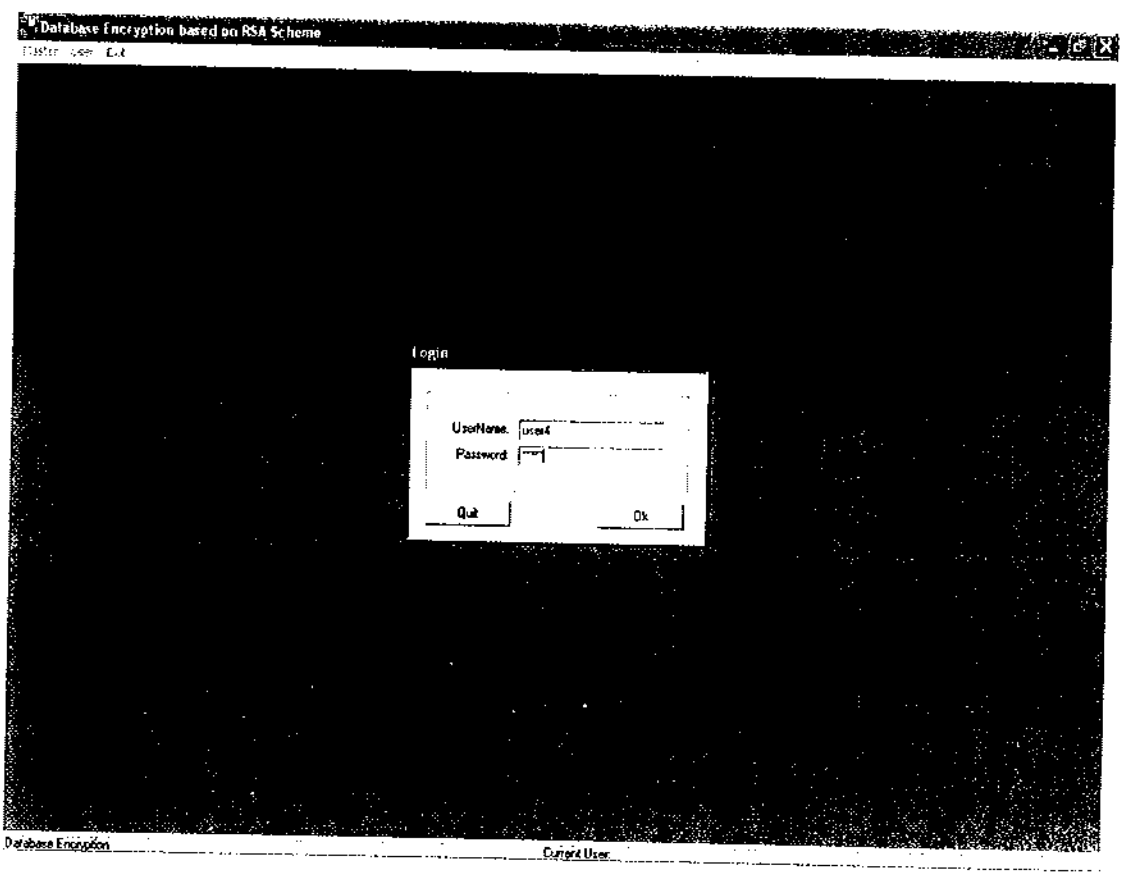
RSA Form

This form displays the RSA implementation and generates the cipher text.



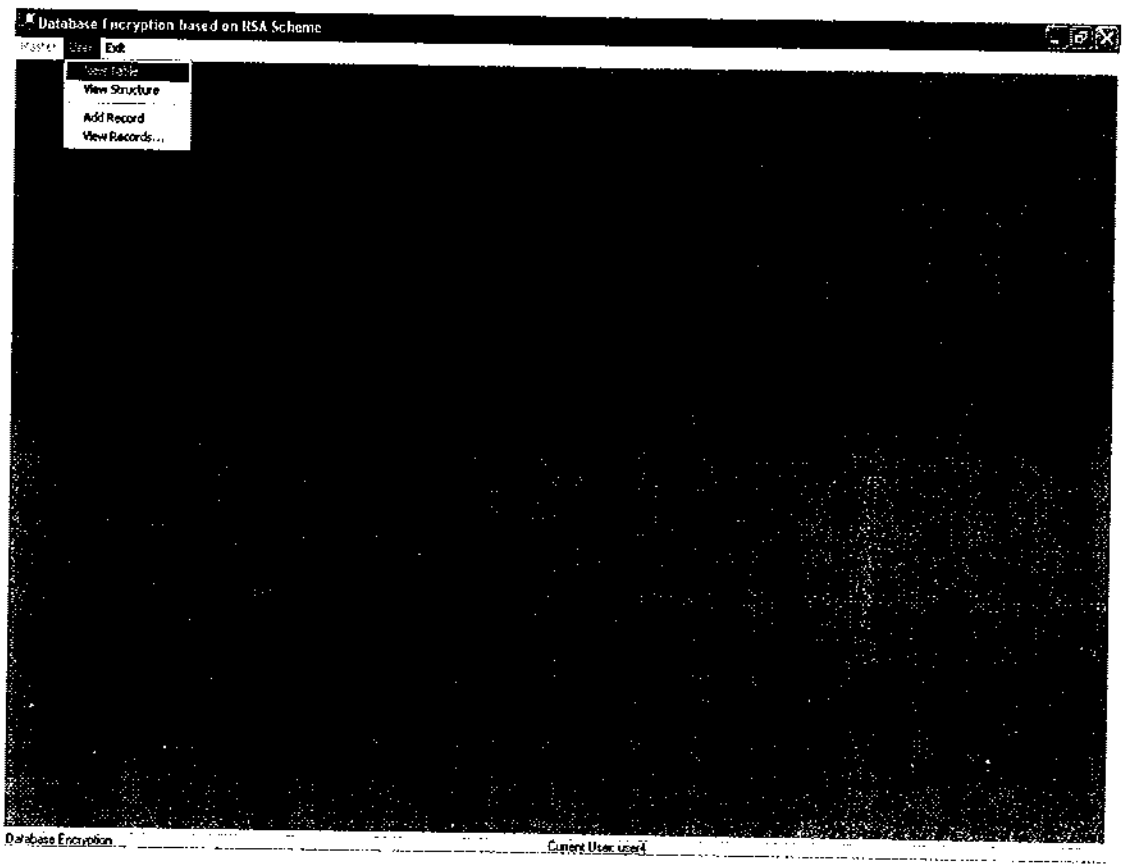
Login Form

This form is used by the user to log in to the system.



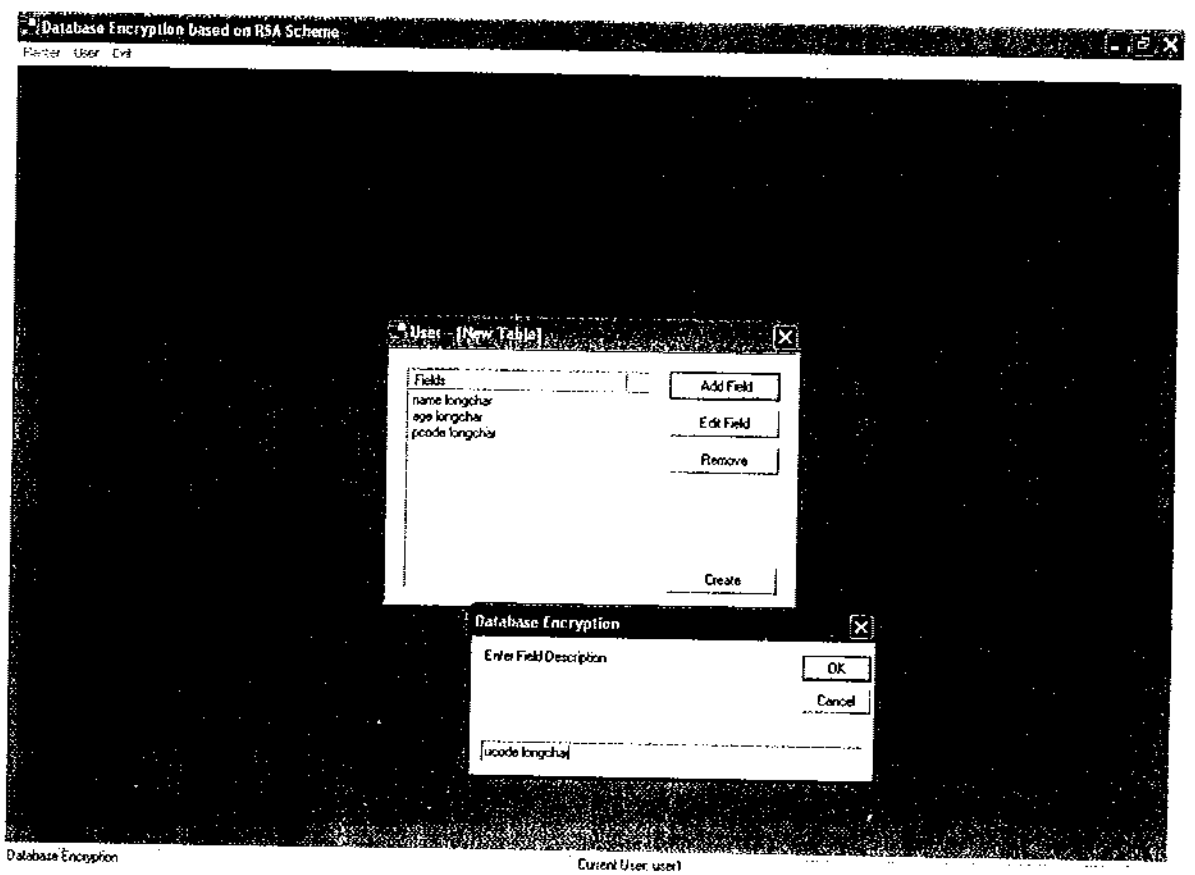
MDI Form

This form displays the operations that can be performed by the user.



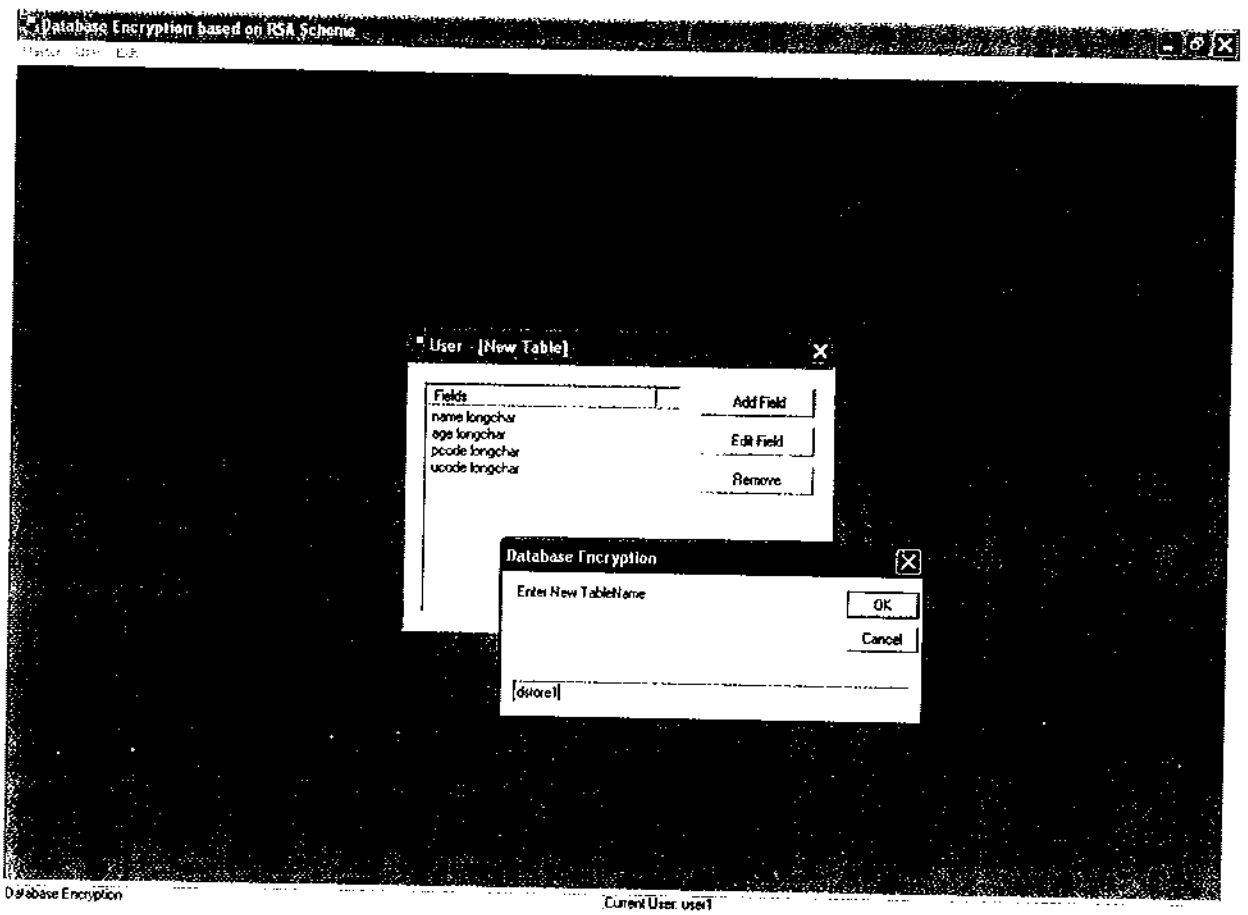
New Table Creation Form

This form is used to create a new table. The fields are specified along with the data type.



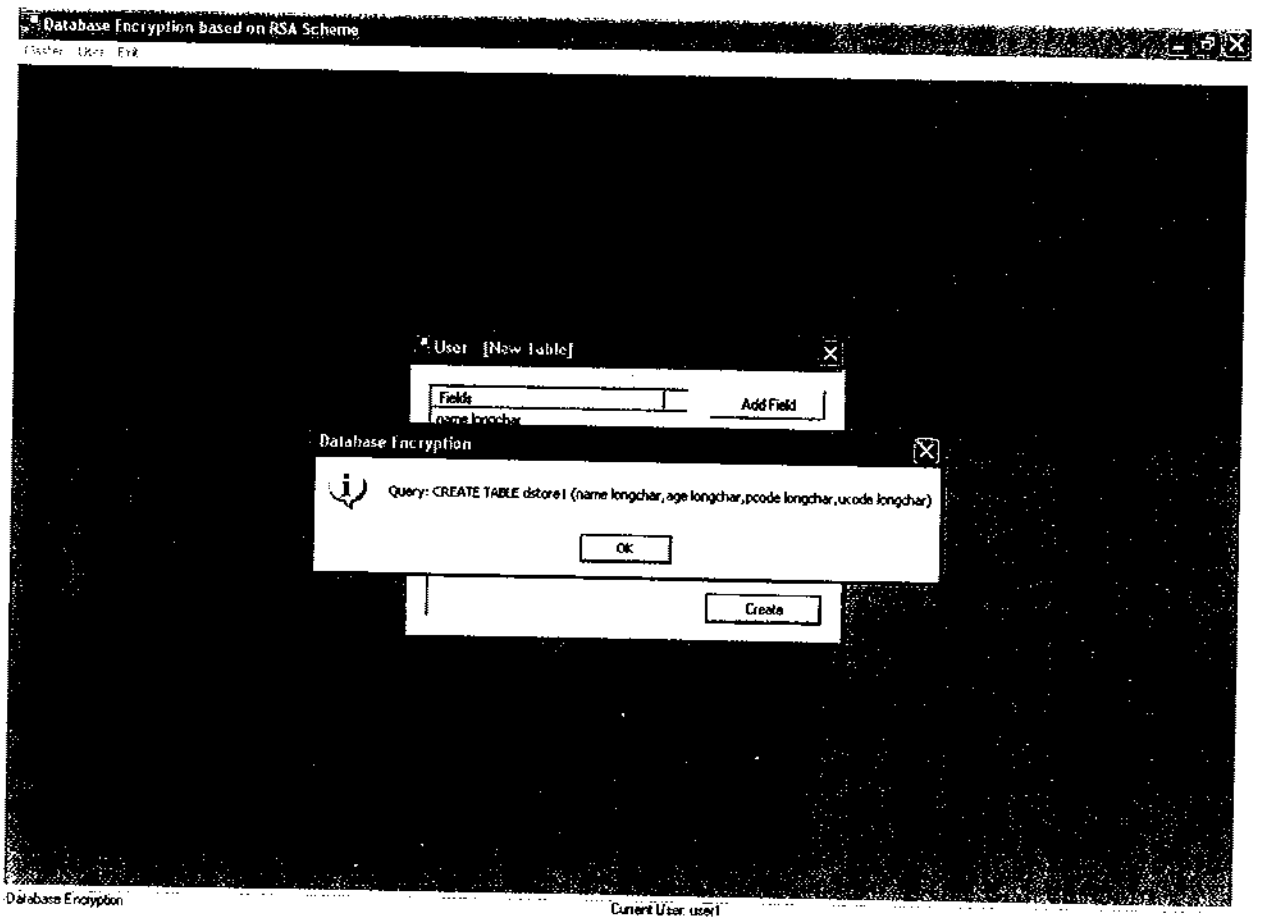
New Table Creation Form

This form is used for specifying a name for the table.



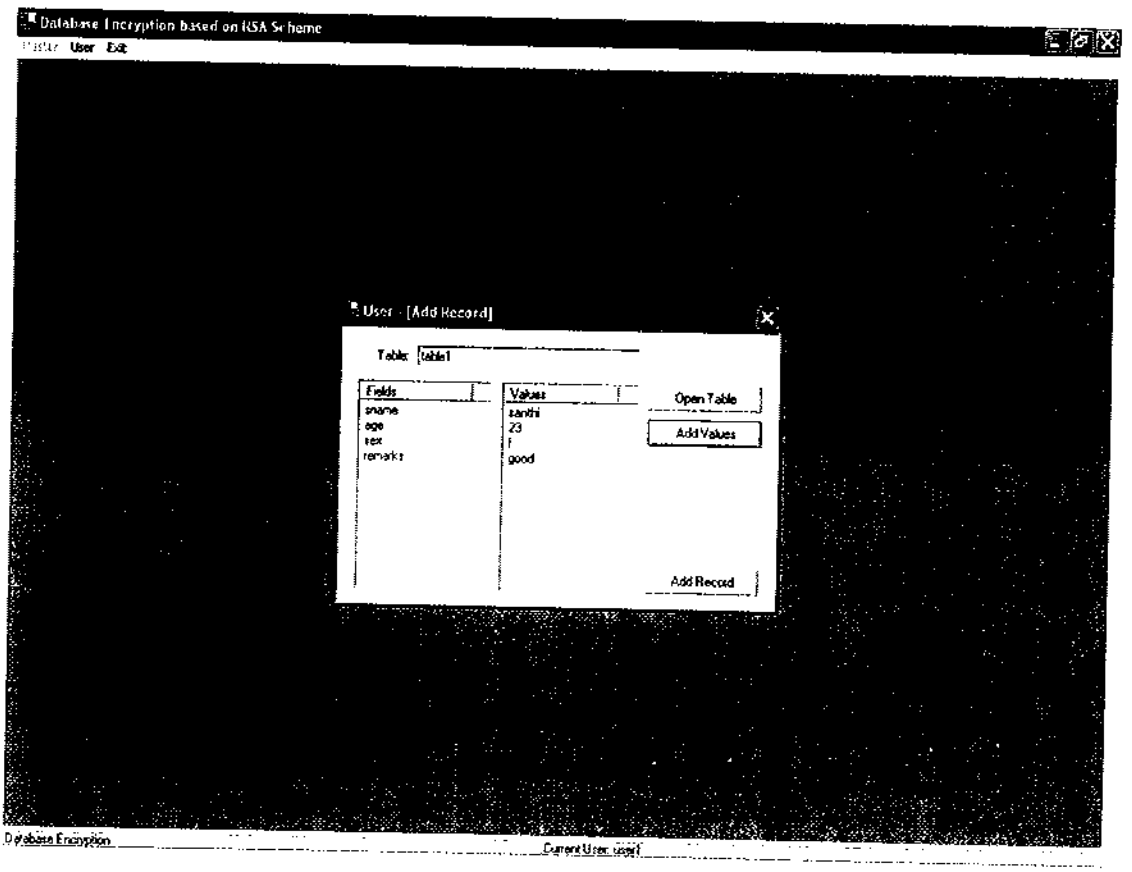
New Table Creation Form

This form displays the query when the new table is created.



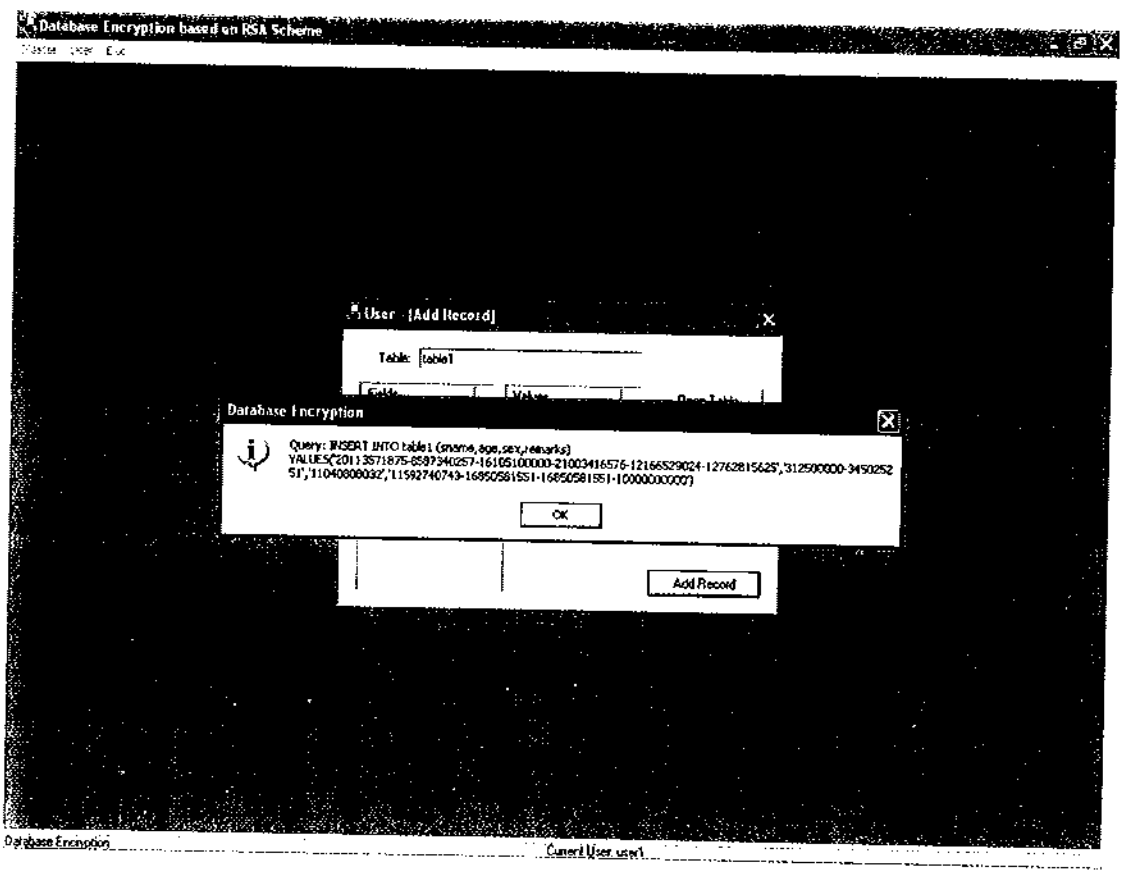
Add Record Form

This form is used to add records to the existing tables.



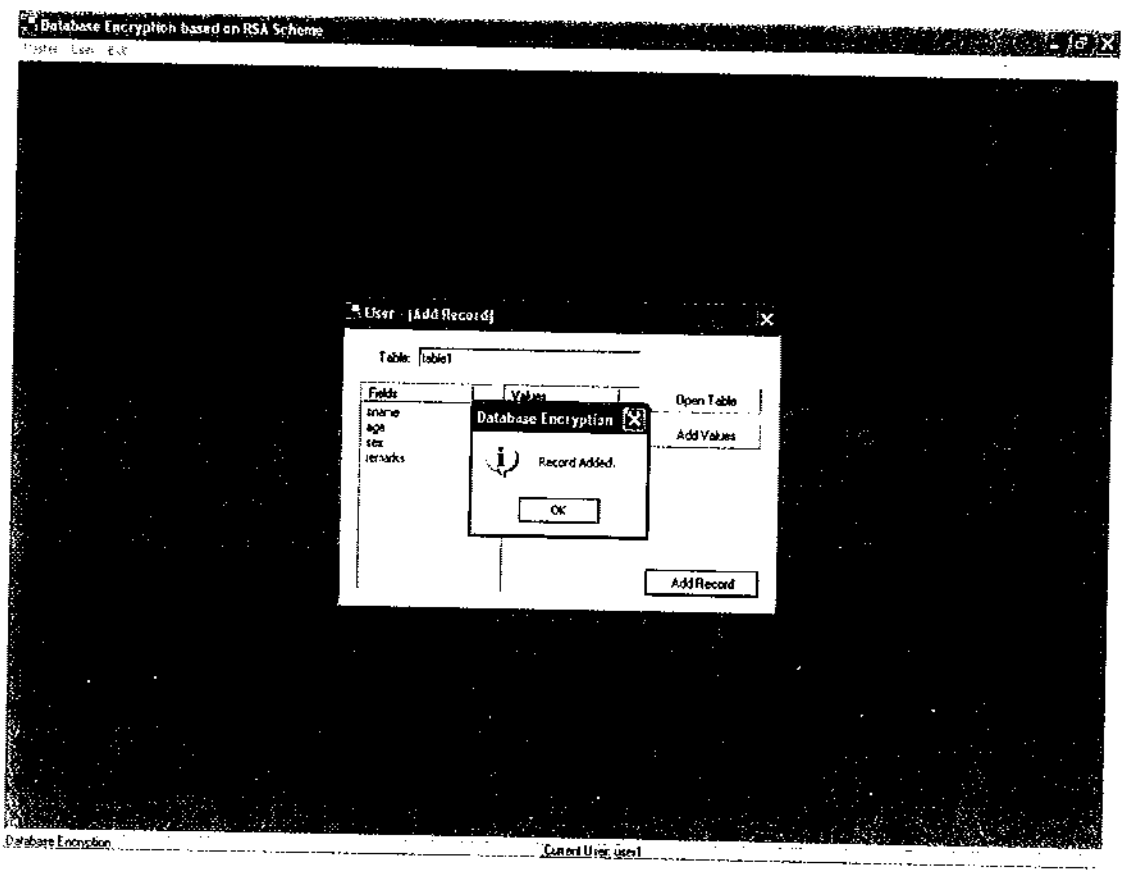
Add Record Form

This form displays the query when a new record is entered into the database. The new record is stored in encrypted form.



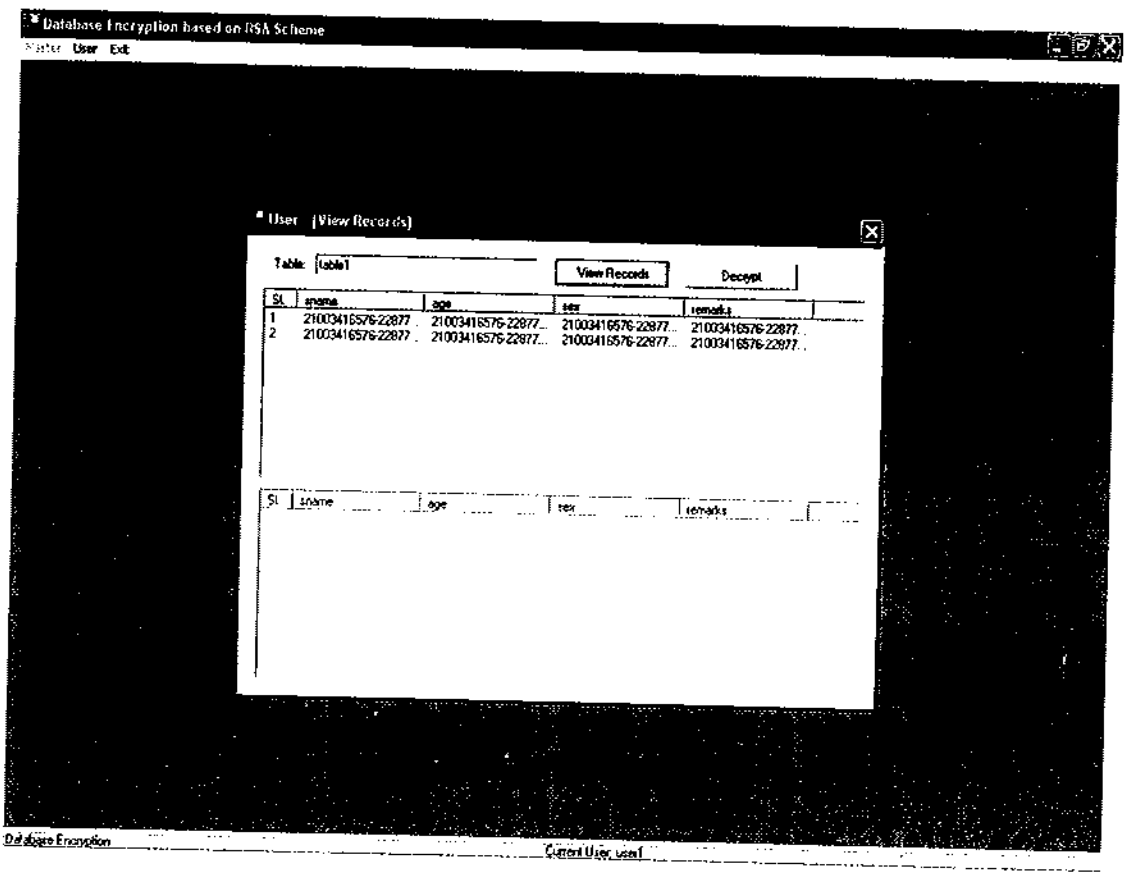
Add Record Form

This form specifies that the specified record has been added to the database.



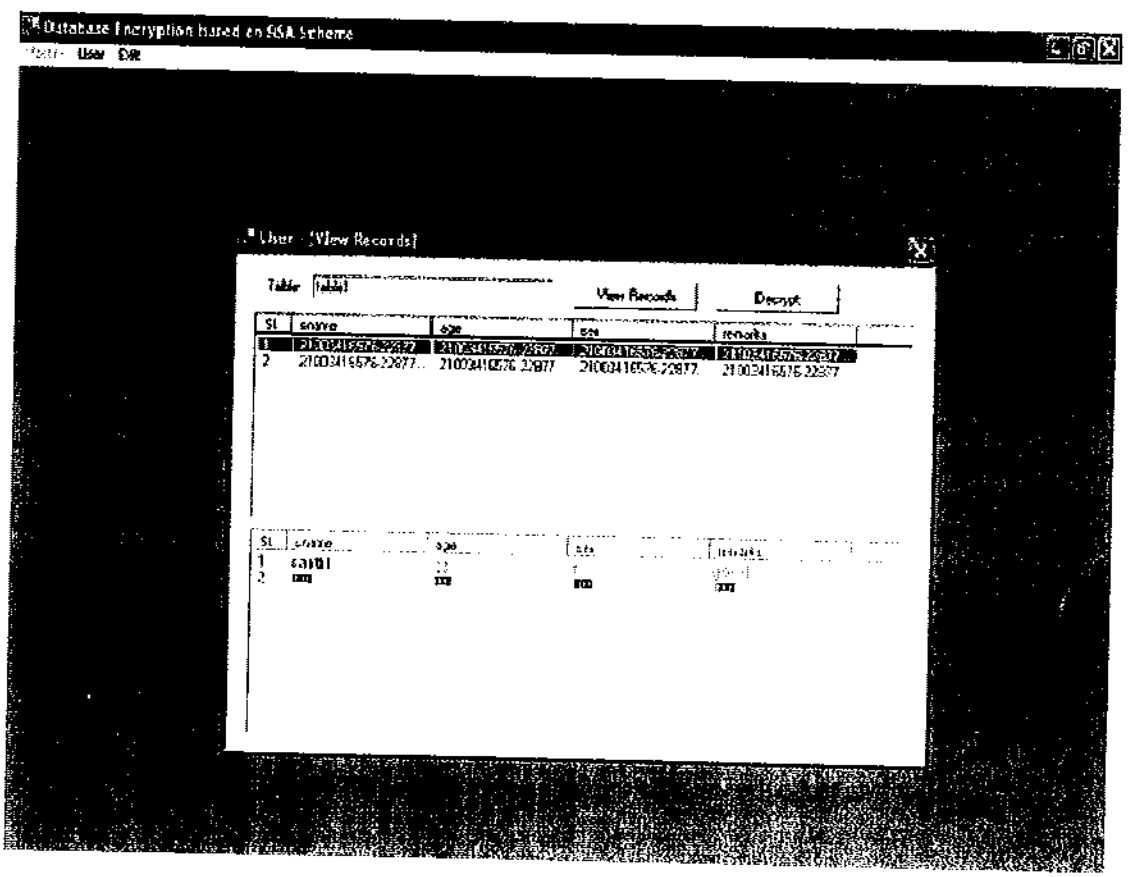
View Record Form

This form is to view the records in the database.



View Record Form

This form decrypts and displays the records entered by the current user and the other records are displayed in encrypted form.



Reference

REFERENCE

1. Steven Holzner , "**Visual Basic .Net Programming** " Black Book - Dream Tech Press, New Delhi.
2. "**Software Engineering**" By Roger Pressman
3. Lee **" Introduction to System Analysis and Design"** Galgotia Book Source Publications
4. Li Gong and Ravi Sandhu, Nov / Dec 2000, "**Security Solutions**", " IEEE Internet Computing", pages 38 – 41.
5. Luca Salgarelli, Milind Buddhikot, Juan Garay, Sarvar Patel and Scott Miller, Dec 2003, "**Efficient Authentication and Key Distribution in Wireless IP Networks**", "IEEE Wireless Communications", pages 52 – 61.
6. Nick Galbreath, 2002, "**Cryptography for database and Internet applications**", Wiley Computer Publishing.
7. William E. Burr, Mar / Apr 2003, "**Selecting the Advanced Encryption Standard**", "IEEE Security and Privacy, Vol.1, No: 2, pages 43 – 52.
8. William Perry,1995, "**Effective Methods of Software Testing**", John Wesley and Sons Inc.
9. www.ugweb.cs.ualberta.ca
10. www.startvbdotnet.com
11. http://en.wikipedia.org/wiki/RSA#History