CHAPTER 1
INTRODUCTION

The title of the project is "**Three Layer Encryption using New Key Management and Bluetooth MAC Address**". It is a monolithic application based on non-traditional idea which utilizes symmetric key encryption algorithm in conjunction with mac address of a Bluetooth device to keep the private data of user secure against unauthorized access and undetected mutilation.

Confidentiality has always played an important role in every aspect of profession. In order to keep this symmetric key secret the proposed software encrypts this confidential symmetric key with the public cryptographic function. This system uses the Bluetooth technology as the main technique to secure the data of the user. The Bluetooth device's MAC address which will be stored in the registry of user's personal computer plays a major role in securing the data. The aim of the software is to ensure data integrity and confidentiality to the authorized parties only.

The front-end will be designed using Windows Forms with C# for client side validation and all business logics will be in C# programming language reside at middle layer. And these layers will interact with third layer of database, which will be in an xml file. To start working on this project environment required is a Visual Studio IDE (10 or above) and Windows Operating System as development environment.

CHAPTER 2
MOTIVATIONAL SURVEY

Recent advances in IT Sector in the field of security has led to the development of many file security softwares which provide their users with the option of encrypting their data to prevent purloin of private data. But still, the private information of the users is not fully protected. The existing system still is not acceptable due to various security issues which leaves the users discontented. Since information can be accessed, manipulated and misused in numerous ways, it can pose as a threat to the user's details.

In order to overcome this, DES has been developed. But this technology still has many shortcomings some of which are: Vulnerability to Key logger attacks, no software which provides absolute protection of user's data, the domain of most the softwares is limited to only DES, Blowfish algorithm, most of the softwares provide the option of encrypting the text files only, no software which uses AES Rijndael Algorithm and Bluetooth mac address for encryption.

This project uses the AES Rijndael Algorithm as a base cryptographic method for the encryption and decryption of the user's private data. The end product (software) of this project overcomes the limitations of the existing system due to features such as this software uses AES Algorithm combined with Bluetooth mac id for security, Decentralized security system, Automatic key generation and destruction termed as "Blind Technology", Priority based encryption, Customization options for the customer, Option of encrypting multiple types of file and folders, Three Layered Encryption with a New Key Management system.

CHAPTER 3
PROBLEM DEFINITION AND SCOPE

3.1 PROBLEM STATEMENT

The overall aim of the proposed system is protecting the user's private data from illegal access or from damage and keeping the used keys in the encryption process safe against any pilfering by applying a new method. This method is a three layer encryption with a new key management system approach using windows registry and a Bluetooth device's MAC address.

3.2 STATEMENT OF SCOPE

Bluetooth is one of the most appropriate technology for securing the data in near future. This software uses three layer encryption with a new key management system approach using windows registry and MAC address of a Bluetooth device. It is a customizable software for securing user's private data from unauthorized access. Rijndael will begin to supplant the Data Encryption Standard (DES) - and later Triple DES - over the next few years in many cryptography applications. Rijndael algorithm is successor to what is currently used the data encryption standard which has proved to be deciphered, given enough computing resources.

3.3 GOALS AND OBJECTIVES

The goal of the software is to protect the user's private data from all kinds of pilfering using the AES algorithm. This software combines the two technologies together as one i.e. wireless Bluetooth technology combined with the AES security algorithm. The Bluetooth technology is used for preliminary authentication of the user after which the AES algorithm is used to encrypt and decrypt the user's data according to the user's requirement.

CHAPTER 4
SOFTWARE REQUIREMENT SPECIFICATIONS

4.1 INTRODUCTION

This document is the Software Requirement Specification (SRS) for the project "Three Layer Encryption using New Key Management and Bluetooth MAC Address". The purpose of this document is to describe the functionality, requirements and general interface of the project.

4.1.1 Scope for Development of this Project:

The requirement of the user is to:

- Absolute protection of private data
- Encryption facility for multiple file types
- Vulnerability to Key logger attacks should not be there
- Simple Interface to use complex security methods

4.1.2 Overview of Responsibility of Developer:

The responsibility for the various functions as described below:

- **Planning:** In this phase, the project scope is defined and the appropriate methods for completing the project are determined.
- **Requirement Analysis:** The functional and non-functional requirements are gathered in this phase.
- **Design:** The designing and implementation of Graphical User Interface is done in this phase.
- **Development:** All the coding and implementation in each module is done in this phase.
- **Integration and test:** All software modules are combined and tested as a group. It occurs after unit testing and before validation testing. Integration testing takes as its input modules that have been unit tested, groups them in

larger aggregates, applies tests defined in an integration test plan to those aggregates, and delivers as its output the integrated system ready for system testing.

- **Maintenance:** In this phase, modification in application is done software delivery to correct faults and to improve performance or other attributes.

4.2 GENERAL DESCRIPTION

4.2.1 User Characteristics:

The target audience for this software is any type of user be it beginner, amateur or professionals (Technical/Non-technical) .The users for this softwares are

- Noncommercial users – The users who use the software for personal use.
- Commercial – The users who use the softwares for commercial use i.e. for business, or any organization.

4.2.2 Product Perspective:

The product will be a monolithic application and can be run on multiple systems once installed. The product will require a keyboard, mouse and monitor to interface with the users. The minimum hardware requirements for the product are specified in this document.

4.2.3 Overview of Functional Requirements:

The client requires the following features:

- Automatic key generation and destruction termed as **"Blind Technology"**
- The software should be able to encrypt the user's data successfully.
- Three Layered Encryption with a New Key Management system
- The software should use latest **AES** for encryption and decryption process.
- The software should restrict any unauthorized access on user's data
- Customization options for the customer
- Priority Based Encryption process.

- Combine the functionality of AES Rijndael Algorithm with Bluetooth mac address for security
- The system should have a mac address reset functionality.

4.2.4. Nonfunctional requirements

Hardware Requirements:

1. Processor	: Pentium IV
2. Hard Disk	: 10GB
3. RAM	: 2GB
4. Device	: A Bluetooth enabled device

Software Requirements:

1. Language	: C#
2. Front End	: Visual Studio IDE
2. Backend	: XML Database
3. Operating System	: Windows XP
4. Framework	: .NET

4.3 USER VIEW OF PRODUCT USE

The front view of the software consists of different services provided by the software when user runs the software for the first time he/she is prompted to register the Bluetooth devices whose MAC address will be used in authentication stage, next the user will be asked to map the path of data to encrypt. From next execution the software will startup automatically and search for the user's registered Bluetooth device's MAC address, if found the user will be able to access the encrypted data else the user will be denied of all the access to data. As soon as the software is unable to find the MAC address of Bluetooth Device it will automatically encrypt the user's data. In this way the software provides a secure way for the user's to prevent their data from unauthorized access.

4.4 SPECIFIC REQUIREMENTS

4.4.1 External Interface Requirements:

- Simple, Attractive, User Friendly
- Self-Contained, Consistent, Self-Explanatory
- Robust

4.4.2 Quality Attributes:

- **Security:** This feature is provided by AES algorithm and MAC id authentication modules.
- **Reliability:** Must maintain data integrity. Computer crashes and misuse should not affect a user's data.
- **Simplicity:** Must be driven by a simple user interface.
- **Portability:** The product can be used with multiple systems.
- **High performance:** The product takes very less time to complete its tasks

CHAPTER 5
SYSTEM DESIGN DOCUMENT

5.1 INTRODUCTION

The design of the document provides the complete description of the design for data, architecture, interfaces and component. It also includes class diagram. This document seeks to provide the software requirement specification.

5.1.1 Purpose and Scope

Purpose of **Three Layer Encryption using New Key Management & Bluetooth MAC Address** Design Document is to describe the design and the architecture of software. The design is expressed in sufficient detail so as to enable all the developers to understand the underlying architecture of this software**.** Logical architecture of software and Data Flow Diagrams are explained.

5.2 TARGET AUDIENCE

This Design document is intended to act as a technical reference tool for developers involved in the development of this project. This document assumes that you have sufficient understanding of the following

Concepts:

- Cryptographic Functions
- C#
- MAC Addresses
- Data Flow Diagrams
- Classes and Interfaces

5.3 PRE-REQUISITES

This software requires no additional plugin to run. Since the software is written in C#, it can run on any platform that supports the Windows Environment, C# or Visual Studio IDE. The compiled files are contained in Executable File (**EXE'**s) and have to be installed on the system for any usage.

5.4 ARCHITECTURAL STRATEGIES

The architectural design of a software project is simply the design of the entire software system. This includes the hierarchy of the modules and also which modules are present in the system. A good architectural design will create a clear and fair balance between cohesion (each module has only one distinct purpose), coupling (no two modules depend completely on each other), abstraction (seeing modules in full and not in detail), hierarchy (logical modules stem from others) and partitioning (logically grouping modules together) of the software modules.

5.5 LOGICAL VIEW

It provides the user with an abstract view of the overall system functionality.

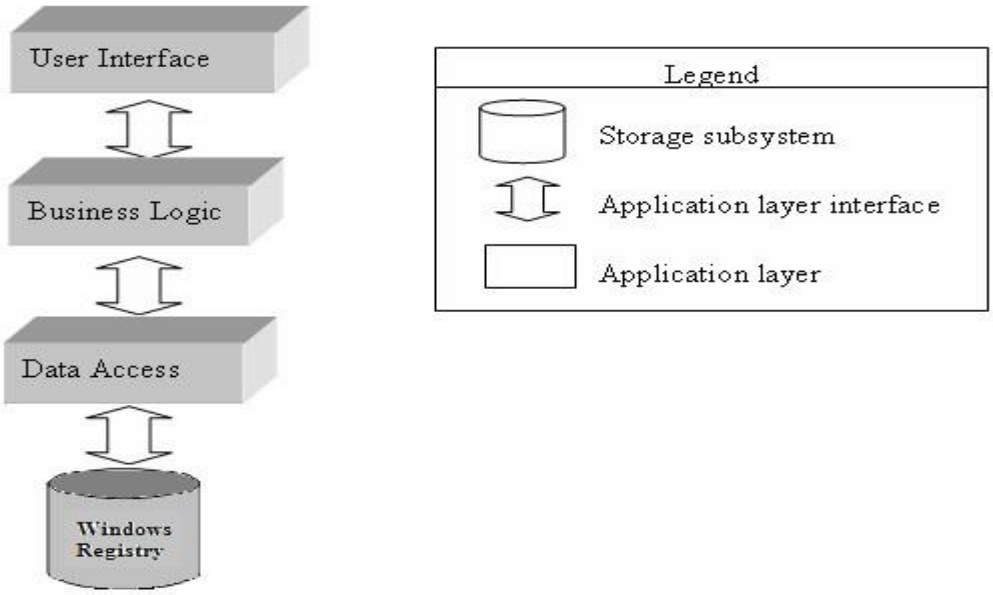

Fig 1: Abstract view of Software

5.6 DATA FLOW DIAGRAMS

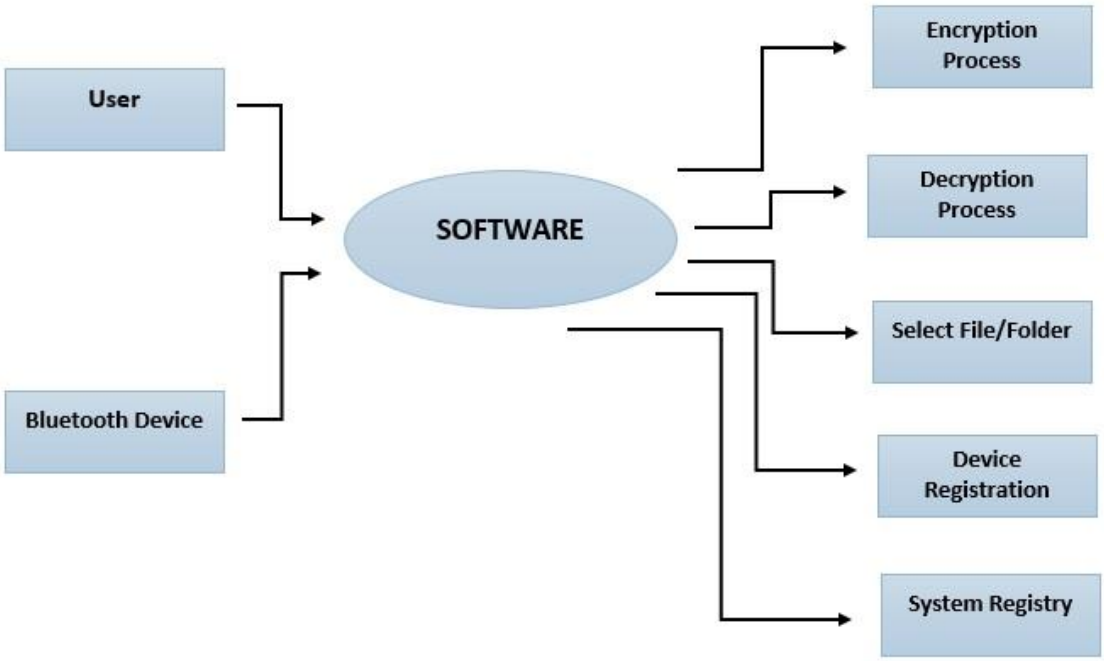

Fig 2: Level 0 DFD

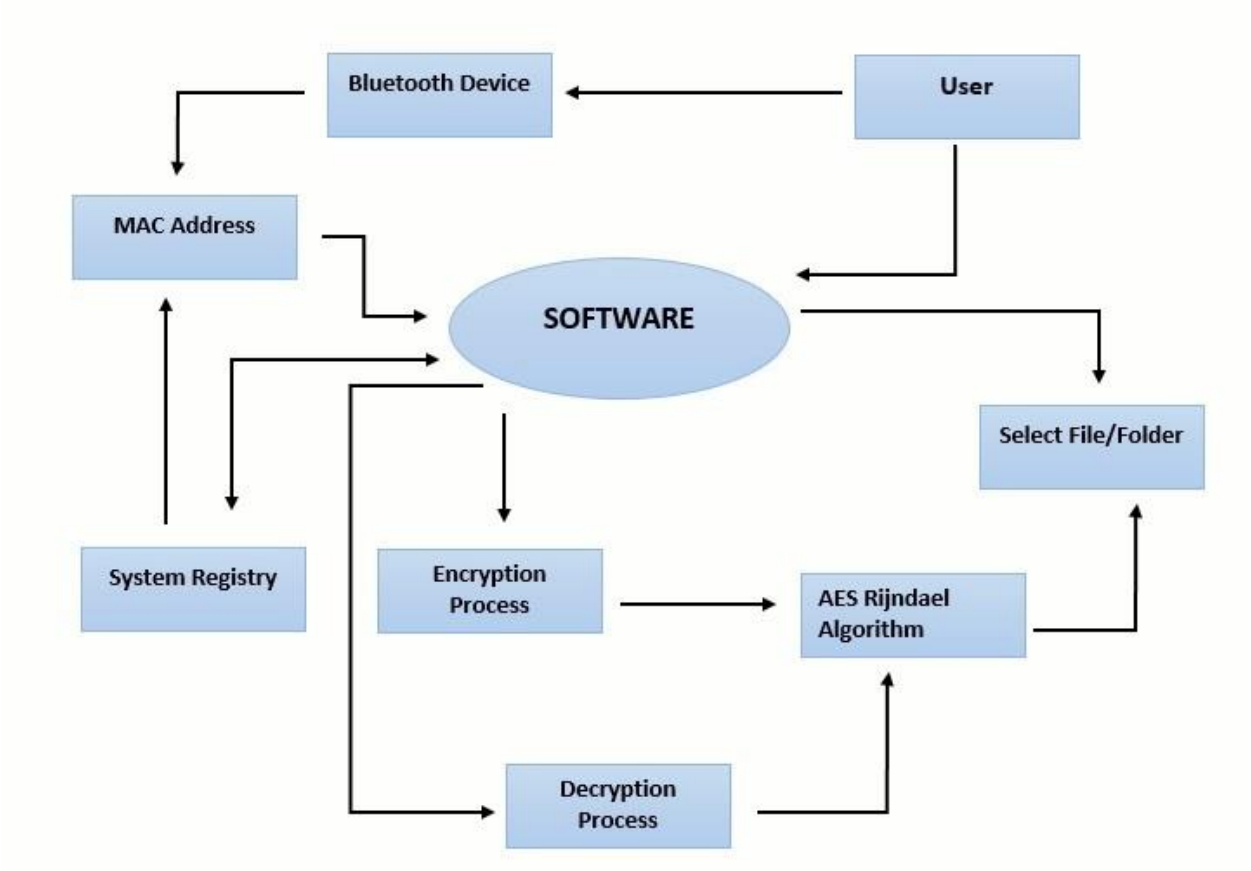

Fig 3: Level 1 DFD

5.7 UML DIAGRAMS

1. Package Diagram

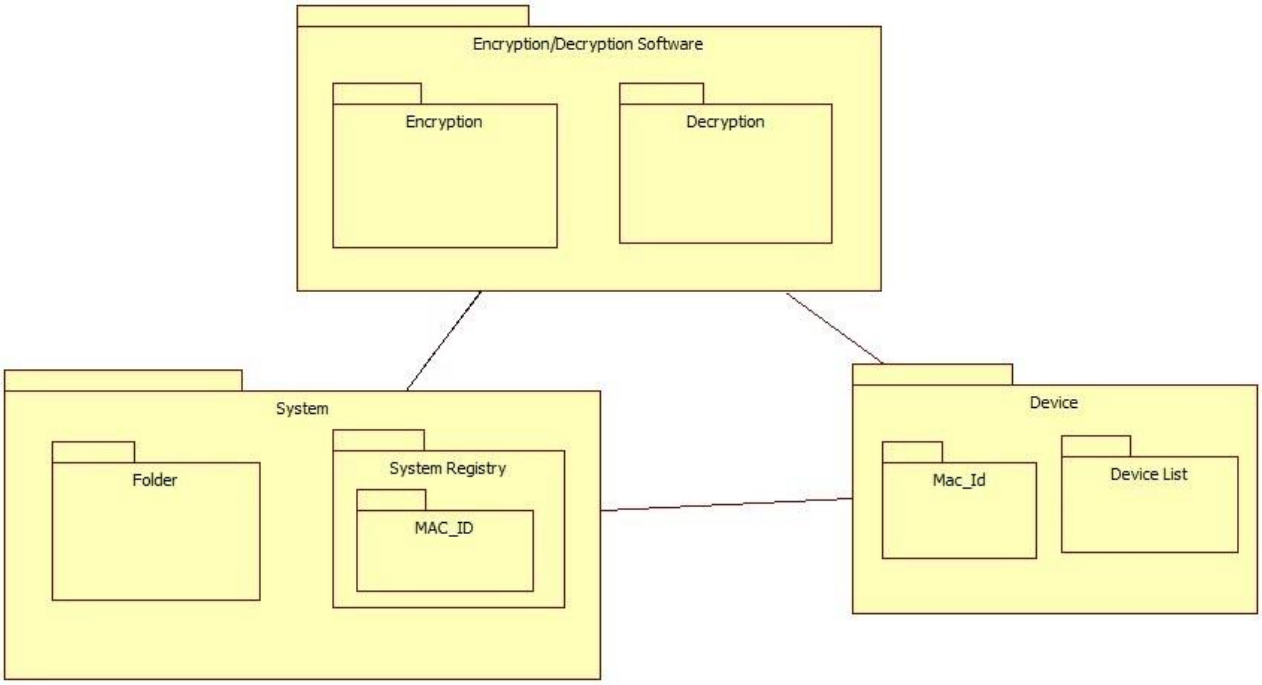

Fig 4: Package Diagram

2. Class Diagram

The component used to model the physical things that may reside on node.

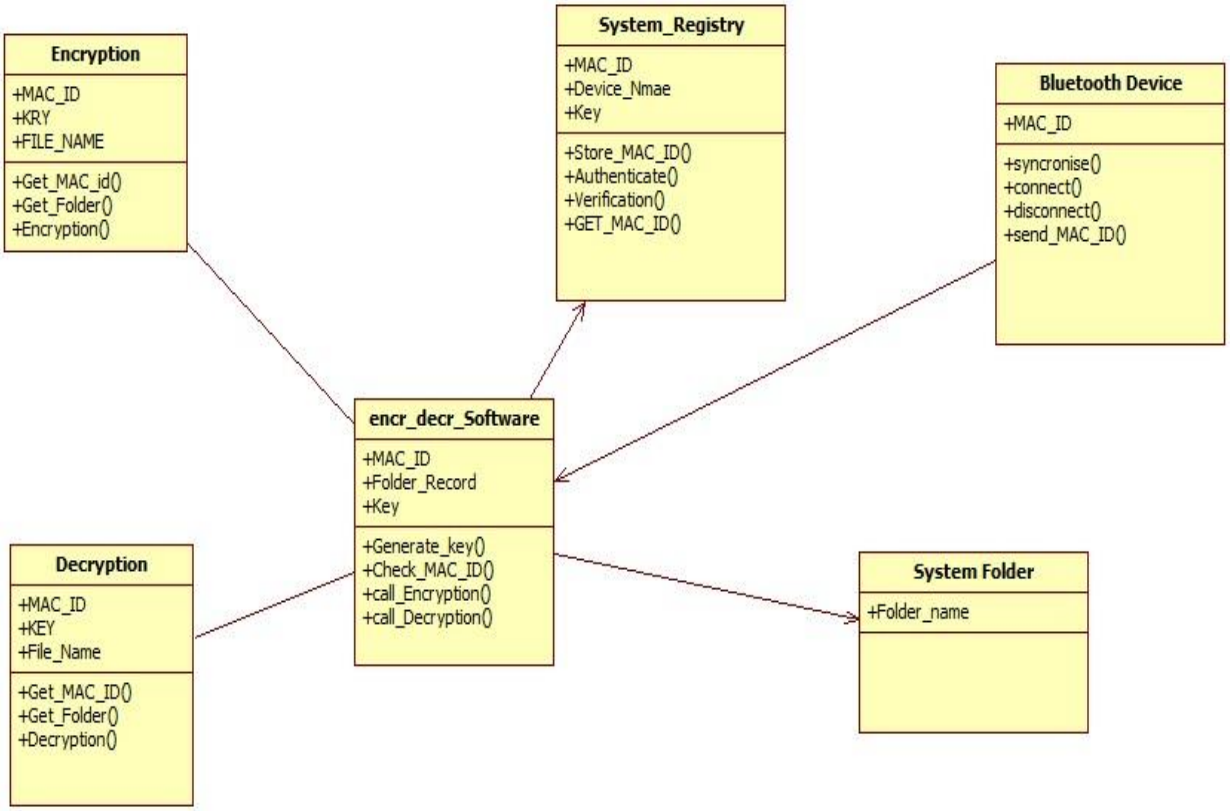

Fig 5: Class Diagram

3. Use Case Diagram:

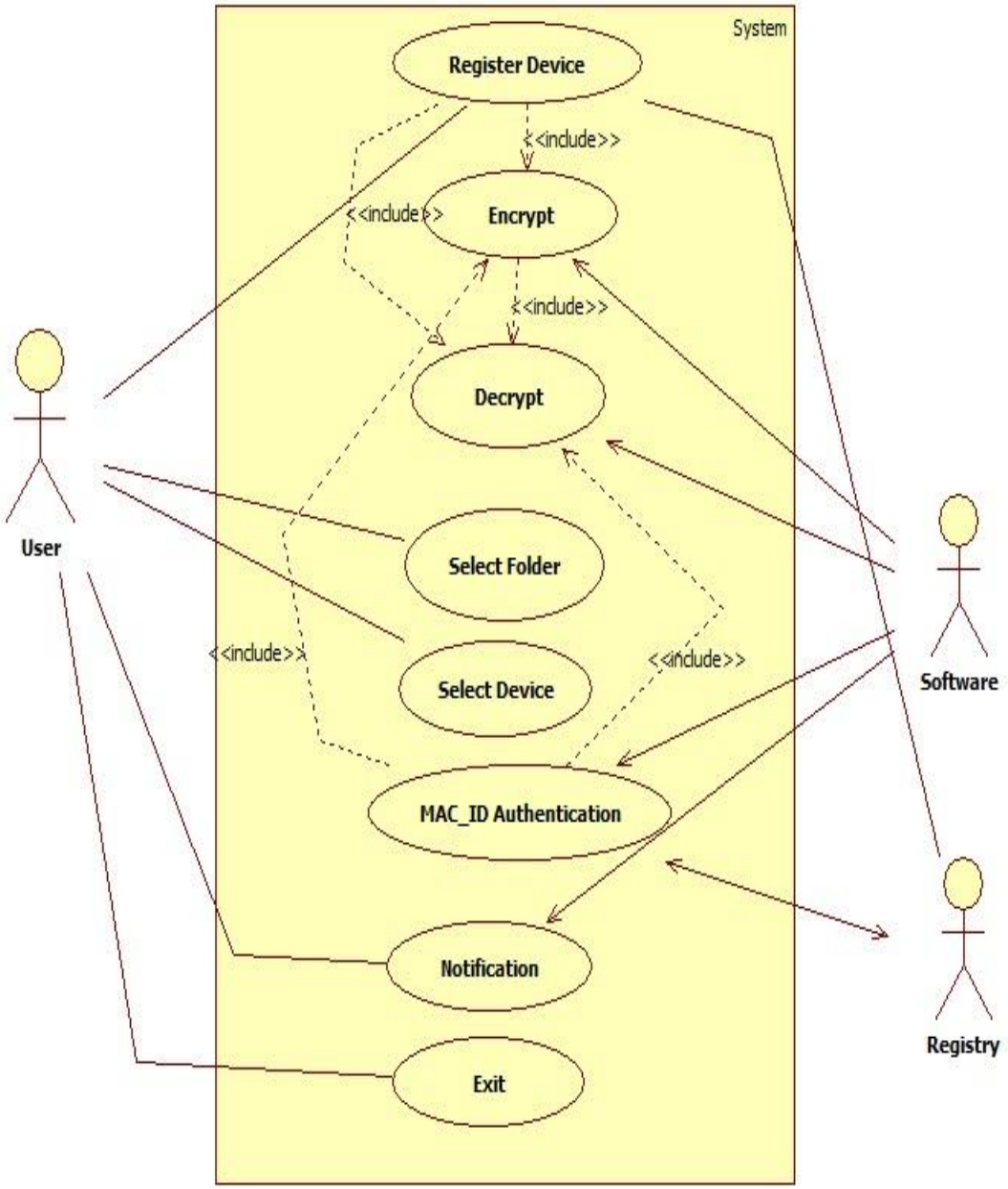

Fig 6: Usecase Diagram

4. State Chart Diagram

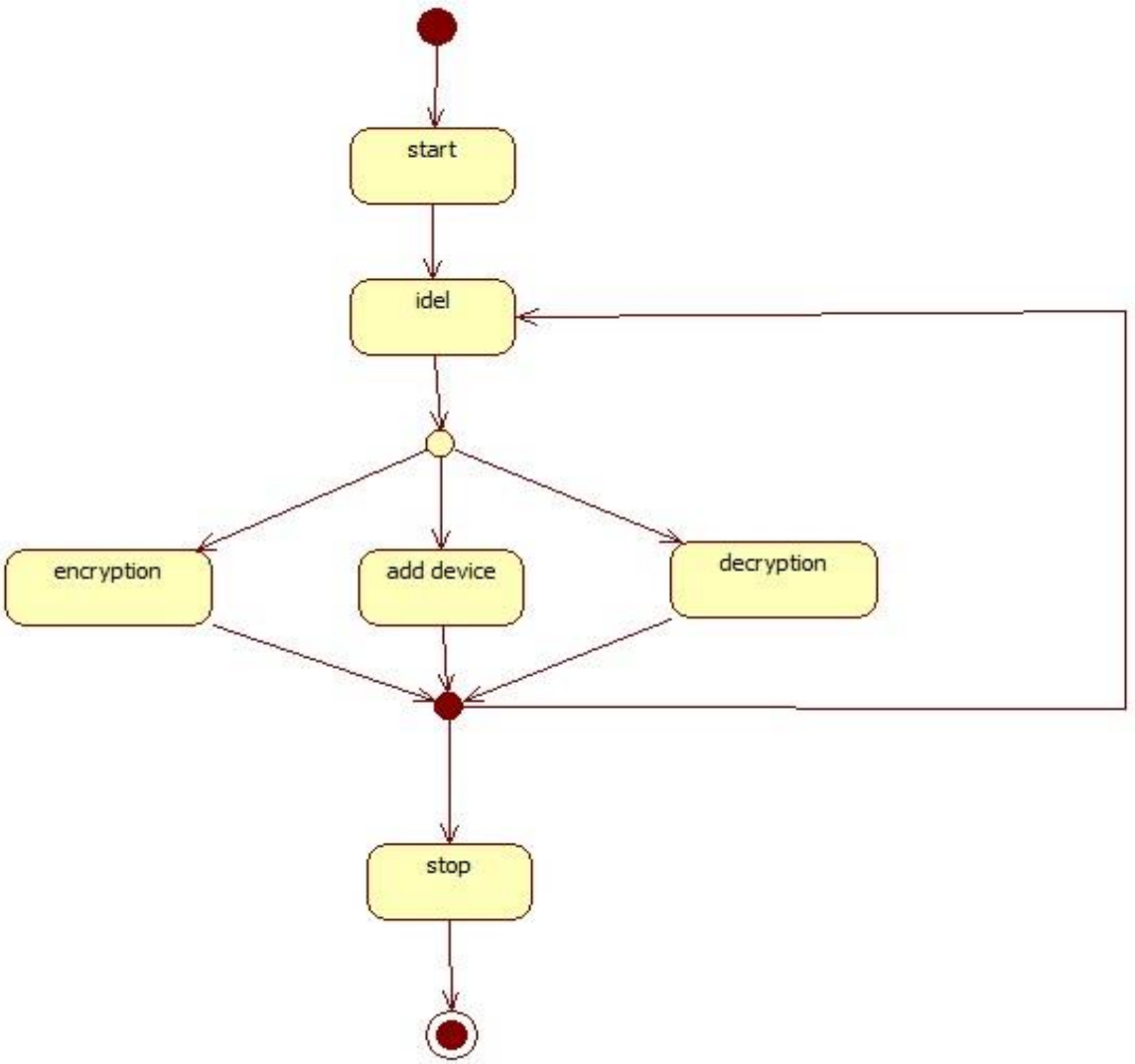

Fig 7: State Chart Diagram

5. Sequence Diagram

Sequence diagram species the sequence of actions that are under taken while operating the system.

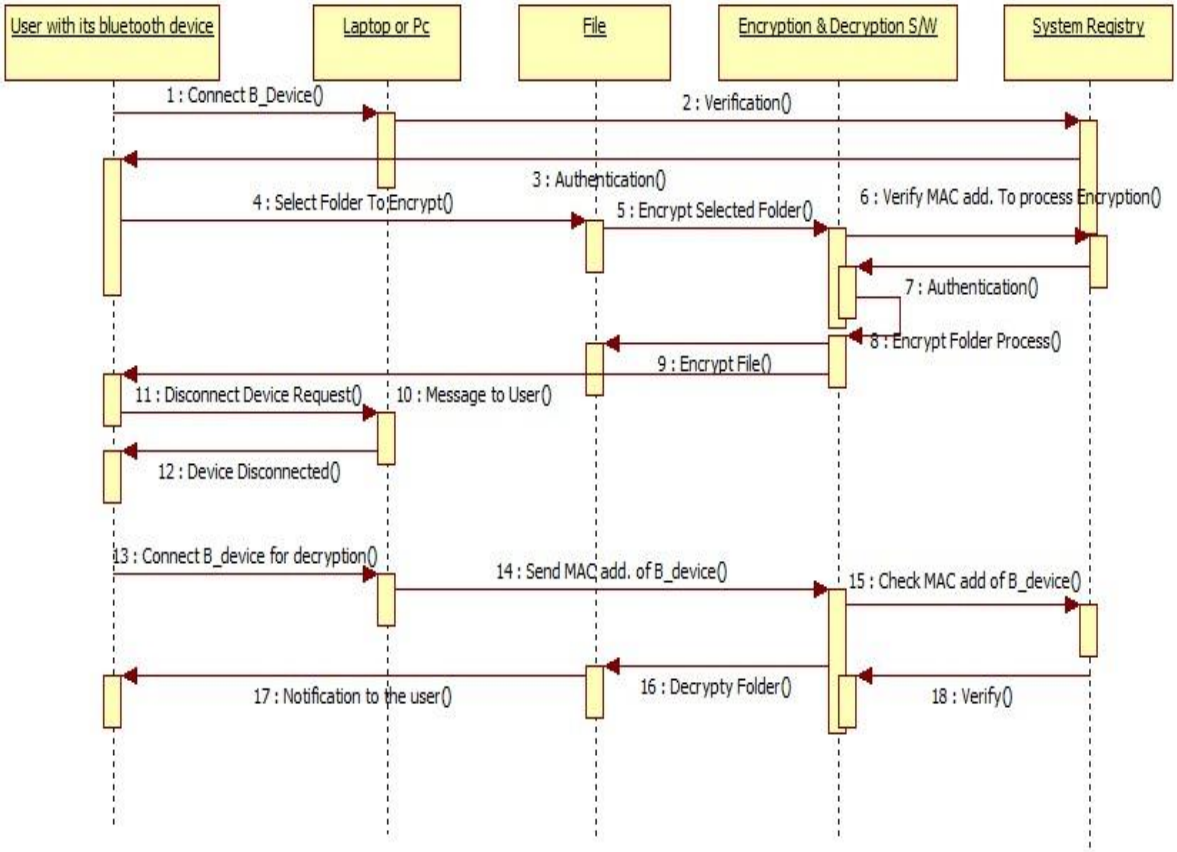

Fig 8: Sequence Diagram

6. Communication Diagram

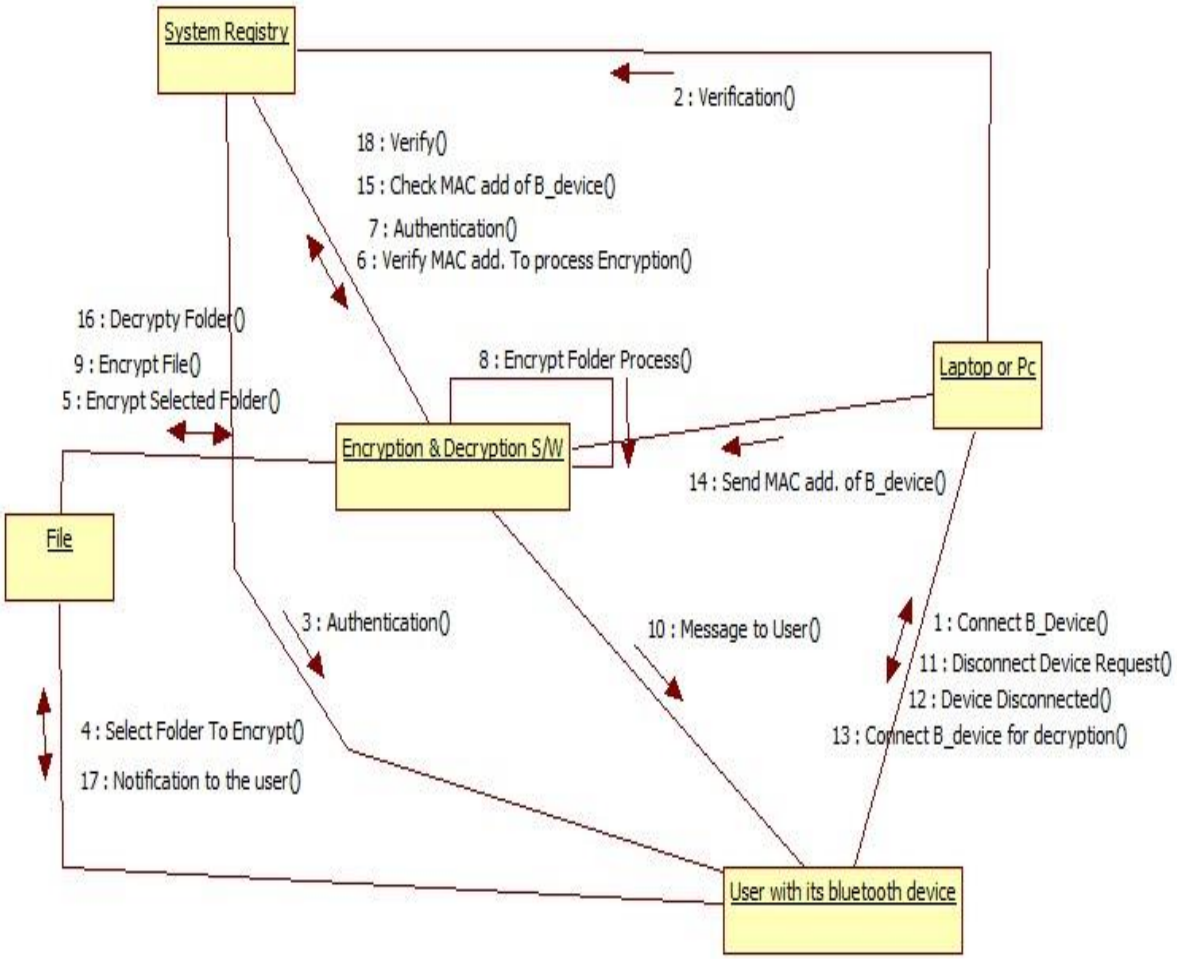

Fig 9: Communication Diagram

7. Activity Diagram

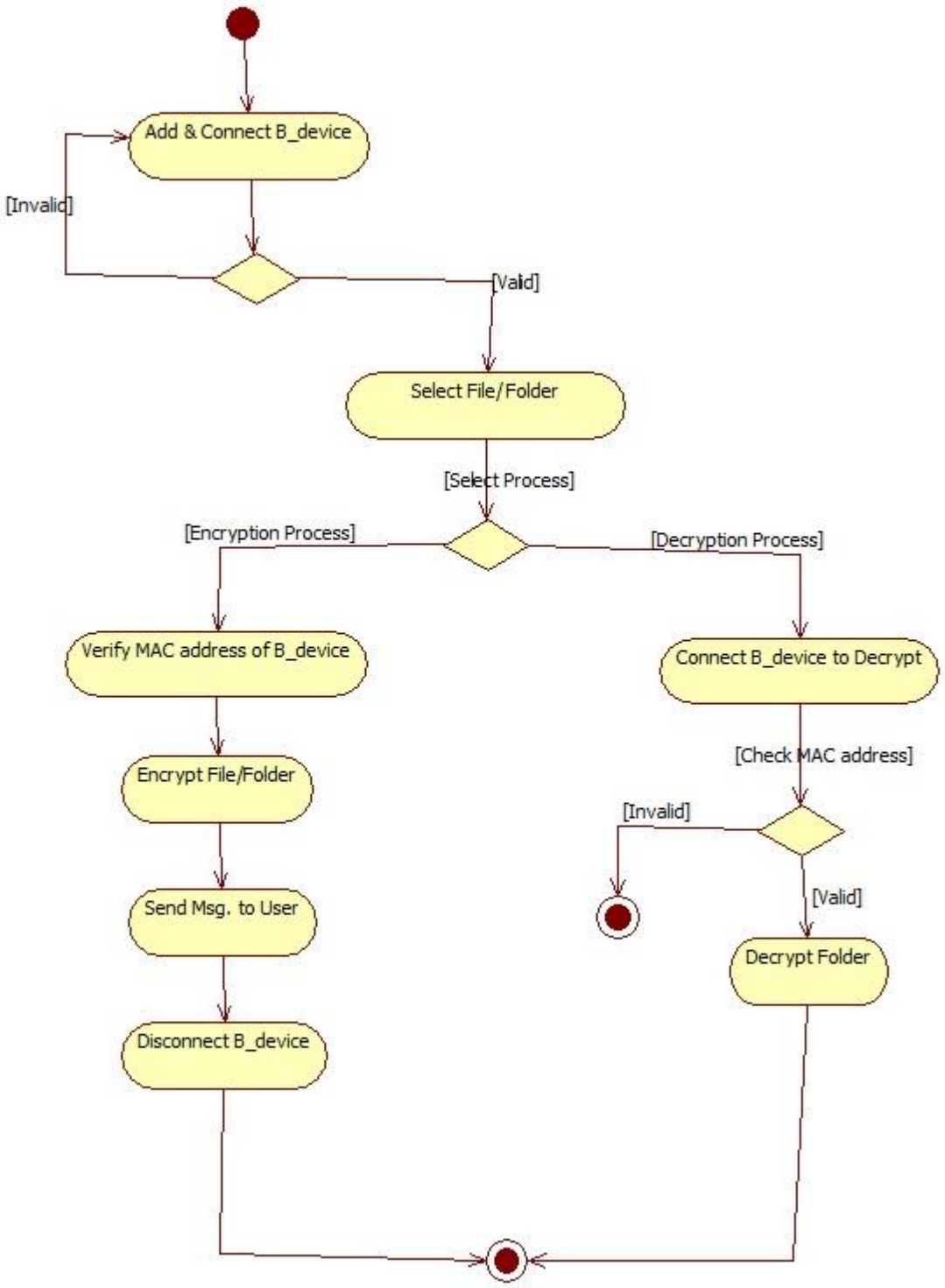

Fig 10: Activity Diagram

8. Component Diagram

The data structures used are represented by the class diagram.

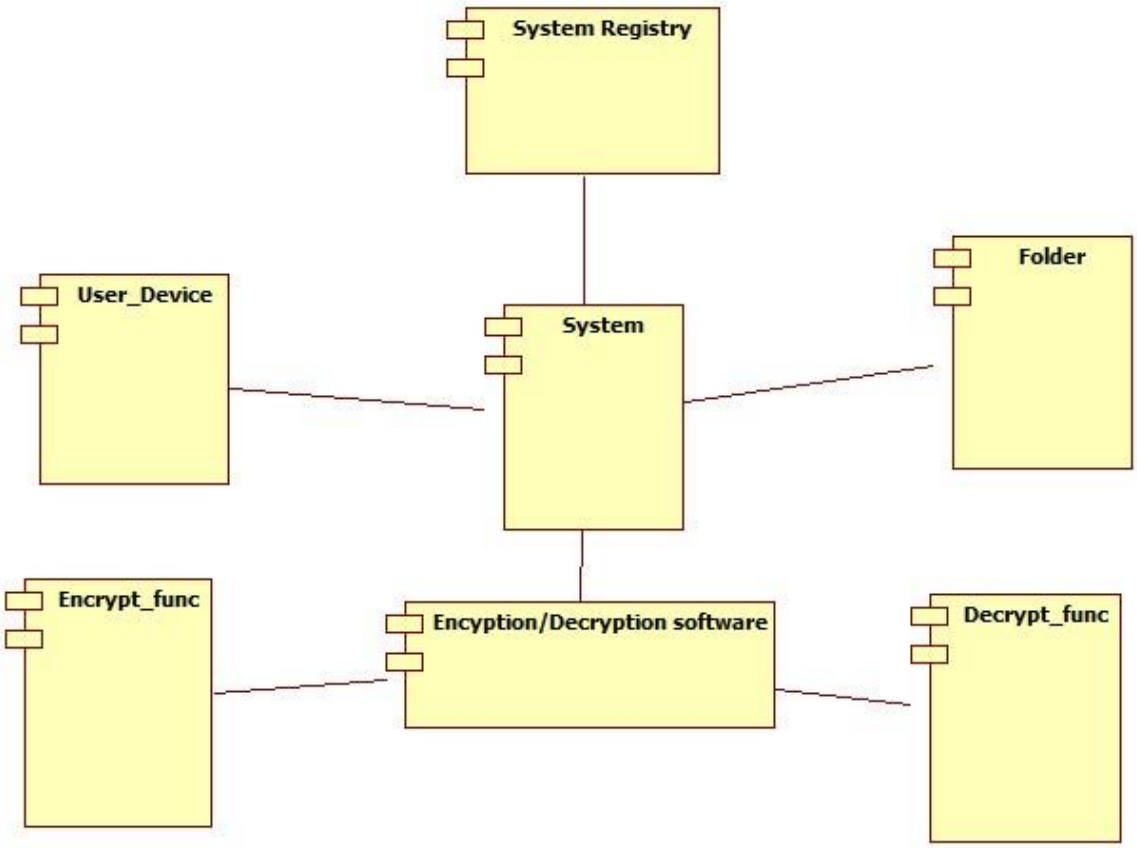

Fig 11: Component Diagram

CHAPTER 6
ALGORITHMIC STRATEGIES

6.1 INTRODUCTION

Rijndael (pronounced rain-dahl) is the algorithm that has been selected by the U.S. National Institute of Standards and Technology (NIST) as the candidate for the Advanced Encryption Standard (AES). It was selected from a list of five finalists that were themselves selected from an original list of more than 15 submissions. Rijndael will begin to supplant the Data Encryption Standard (DES) - and later Triple DES - over the next few years in many cryptography applications. The algorithm was designed by two Belgian cryptologists, Vincent Rijmen and Joan Daemen, whose surnames are reflected in the cipher's name. Rijndael has its origins in Square, an earlier collaboration between the two cryptologists.

The Rijndael algorithm is a new generation symmetric block cipher that supports key sizes of 128, 192 and 256 bits, with data handled in 128-bit blocks - however, in excess of AES design criteria, the block sizes can mirror those of the keys. Rijndael uses a variable number of rounds, depending on key/block sizes, as follows:

	Key Length *(Nk words)*	**Block Size** *(Nb words)*	**Number of Rounds** *(Nr)*
AES-128	4	4	10
AES-192	6	4	12
AES-256	8	4	14

Fig 12: Key-Block-Round-Combinations

Rijndael is a substitution linear transformation cipher, not requiring a Feistel network. It use triple discreet invertible uniform transformations (layers). Specifically, these are: Linear Mix Transform; Non-linear Transform and Key Addition Transform. Even before the first round, a simple key addition layer is performed, which adds to security. Thereafter,

there are Nr-1 rounds and then the final round. The transformations form a State when started but before completion of the entire process.

The State can be thought of as an array, structured with 4 rows and the column number being the block length divided by bit length (for example, divided by 32). The cipher key similarly is an array with 4 rows, but the key length divided by 32 to give the number of columns. The blocks can be interpreted as unidimensional arrays of 4-byte vectors.

The exact transformations occur as follows: the byte sub transformation is nonlinear and operates on each of the State bytes independently - the invertible S-box (substitution table) is made up of 2 transformations. The shiftrow transformation sees the State shifted over variable offsets. The shift offset values are dependent on the block length of the State. The mix column transformation sees the State columns take on polynomial characteristics over a Galois Field values (28), multiplied x4 + 1 (modulo) with a fixed polynomial. Finally, the roundkey transform is XORed to the State. The key schedule helps the cipher key determine the round keys through key expansion and round selection.

Overall, the structure of Rijndael displays a high degree of modular design, which should make modification to counter any attack developed in the future much simpler than with past algorithm designs.

6.2 ALGORITHM SPECIFICATION

1. **KeyExpansions—**round keys are derived from the cipher key using Rijndael's key schedule. AES requires a separate 128-bit round key block for each round plus one more.
2. **InitialRound**
    1. **AddRoundKey—**each byte of the state is combined with a block of the round key using bitwise xor.
3. **Rounds**
    1. **SubBytes—**a non-linear substitution step where each byte is replaced with another according to a lookup table.

2. **ShiftRows**—a transposition step where the last three rows of the state are shifted cyclically a certain number of steps.
3. **MixColumns**—a mixing operation which operates on the columns of the state, combining the four bytes in each column.
4. **AddRoundKey**

4. **Final Round (no MixColumns)**
    1. SubBytes
    2. ShiftRows
    3. AddRoundKey.

6.2.1 The SubBytes Step

In the SubBytes step, each byte $a_{i,j}$ in the *state* matrix is replaced with a SubByte $S(a_{i,j})$ using an 8-bit substitution box, the Rijndael S-box. This operation provides the non-linearity in the cipher. The S-box used is derived from the multiplicative over **GF** (2^8), known to have good non-linearity properties. To avoid attacks based on simple algebraic properties, the S-box is constructed by combining the inverse function with an invertible affine transformation. The S-box is also chosen to avoid any fixed points (and so is a derangement), i.e.$S(a_{i,j})$! = $a_{i,j}$, and also any opposite fixed points, i.e. $S(a_{i,j}) \oplus a_{i,j} \neq 0\text{xFF}$. While performing the decryption, Inverse SubBytes step is used, which requires first taking the affine transformation and then finding the multiplicative inverse (just reversing the steps used in SubBytes step).

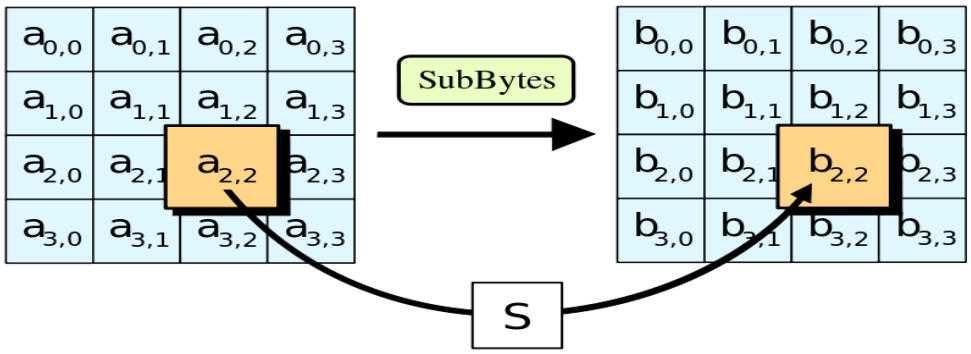

Fig 13: SubBytes Step

6.2.2 The ShiftRows step

The ShiftRows step operates on the rows of the state; it cyclically shifts the bytes in each row by a certain offset. For AES, the first row is left unchanged. Each byte of the second row is shifted one to the left. Similarly, the third and fourth rows are shifted by offsets of two and three respectively. For blocks of sizes 128 bits and 192 bits, the shifting pattern is the same. Row n is shifted left circular by n-1 bytes. In this way, each column of the output state of the ShiftRows step is composed of bytes from each column of the input state. (Rijndael variants with a larger block size have slightly different offsets). For a 256-bit block, the first row is unchanged and the shifting for the second, third and fourth row is 1 byte, 3 bytes and 4 bytes respectively—this change only applies for the Rijndael cipher when used with a 256-bit block, as AES does not use 256-bit blocks. The importance of this step is to avoid the columns being linearly independent, in which case, AES degenerates into four independent block ciphers.

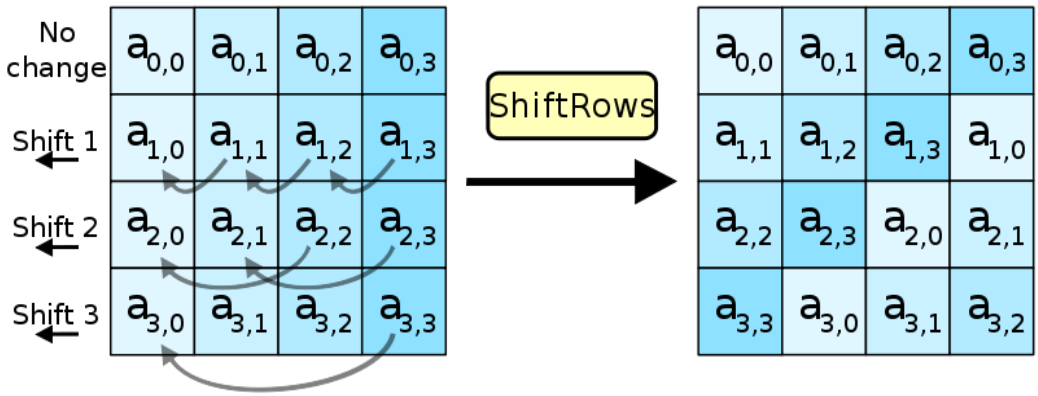

Fig 14: ShiftRows Step

6.2.3 The MixColumns Step

In the MixColumns step, the four bytes of each column of the state are combined using an invertible linear transformation. The MixColumns function takes four bytes as input and outputs four bytes, where each input byte affects all four output bytes. Together with ShiftRows, MixColumns provides diffusion in the cipher.

During this operation, each column is multiplied by a fixed matrix:

$$\begin{bmatrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \end{bmatrix}$$

Matrix multiplication is composed of multiplication and addition of the entries, and here the multiplication operation can be defined as this: multiplication by 1 means no change, multiplication by 2 means shifting to the left, and multiplication by 3 means shifting to the left and then performing XOR with the initial unshifted value. After shifting, a conditional XOR with 0x1B should be performed if the shifted value is larger than 0xFF. (These are special cases of the usual multiplication in **GF** (2^8).) Addition is simply XOR.

In more general sense, each column is treated as a polynomial over **GF** (2^8) and is then multiplied modulo x^4+1 with a fixed polynomial $c(x) = 0x03 \cdot x^3 + x^2 + x + 0x02$. The coefficients are displayed in their hexadecimal equivalent of the binary representation of bit polynomials from **GF** (2) [x]. The MixColumns step can also be viewed as a multiplication by the shown particular MDS matrix in the finite field **GF** (2^8). This process is described further in the article Rijndael mix columns.

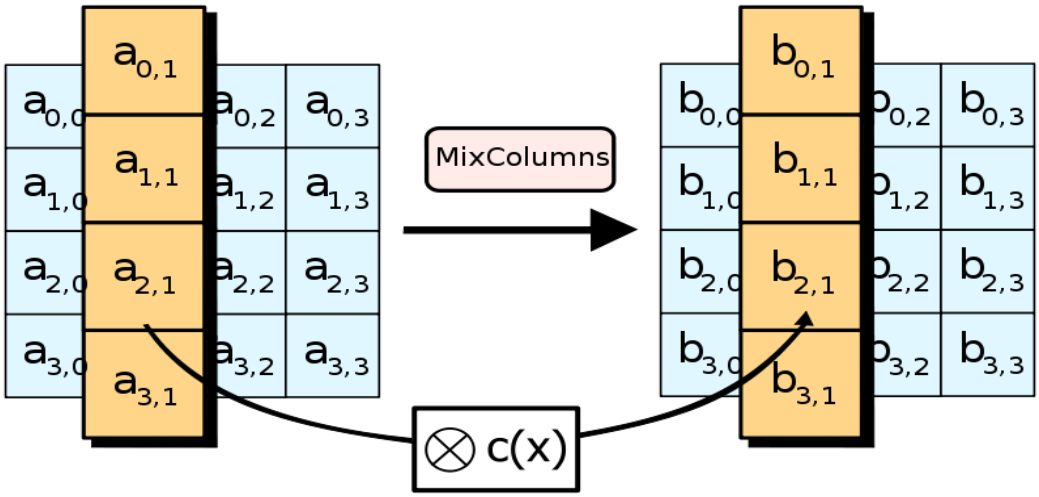

Fig 15: MixColumns Step

6.2.4 The AddRoundKey Step

In the AddRoundKey step, the subkey is combined with the state. For each round, a subkey is derived from the main key using Rijndael's key schedule; each subkey is the same size as the state. The subkey is added by combining each byte of the state with the corresponding byte of the subkey using bitwise XOR.

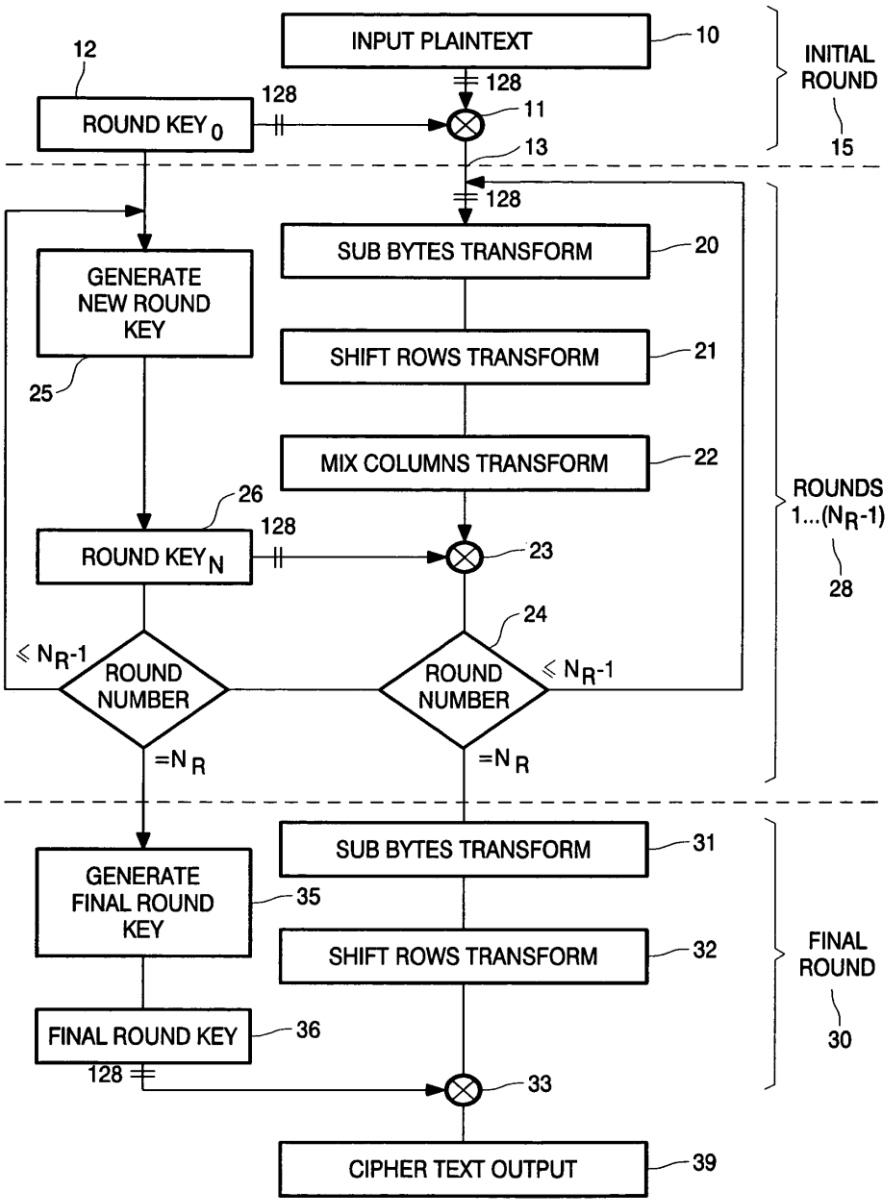

Fig 16: Algorithm Flowchart

CHAPTER 7
TEST SPECIFICATION

7.1 INTRODUCTION

The document is procedural guide for listing the testing activities that should be carried out the "**Three Layer Encryption using New Key Management and Bluetooth MAC Address**". It describes the software test environment for testing, identifies the tests to be performed and provides schedules for the test activities.

7.2 PURPOSE OF THE DOCUMENT

The purpose and objectives are:

- Identify all the activities involved in testing.
- Resources required for executing testing activities and monitoring mechanisms.
- The main goal includes testing of protection of user's data by trying various kinds of attacks against the software.
- Software should resist each and every kind of attacks against the system e.g. Brute Force attack which is the most common form of attack against the security softwares.

ADVANTAGES

- Customized Security
- Decentralized security system
- Secure private data from unauthorized user
- Three layer encryption using new key management
- Priority based encryption
- Automatic key generation and destruction termed as “Blind Technology”
- Bluetooth MAC address is used to provide additional security.

LIMITATIONS

- Bluetooth device’s MAC address is a must to access the data.
- If Windows Registry is damaged then no method of data recovery.
- Bluetooth device should be present in the vicinity.

CONCLUSION

The proposed system applies a decentralized security system by using the unique MAC Address which guarantees that the decryption did not occur until the Bluetooth device is connected to the system. The used keys in the encryption process are generated internally in the system and no one have authority to get their contents which is called a ―blind technology that guarantees the keys safeness against key logger attack. The proposed system prevents any insider or outsider attacks in the private data by encrypting the specified data with a Rijndael block cipher algorithm and prevents any access without having the used keys.

FUTURE ENHANCEMENT

The proposed system applies a decentralized security system by using the unique MAC Address which guarantees that the decryption did not occur until the Bluetooth device is connected to the system. The used keys in the encryption process are generated internally in the system and no one have authority to get their contents which is called a ―blind technology, which guarantees the keys safeness against key logger attack. The proposed system prevents any insider or outsider attacks in the private data by encrypting the specified data with a Rijndael block cipher algorithm and prevents any access without having the used keys.

REFERENCES

- An Innovative Approach to File Security Using Bluetooth , International Journal of Scientific Engineering and Technology (ISSN : 2277-1581)Volume No.2, Issue No.5, pp : 417-423 1 May 2013
- Securing Bluetooth Communications, International Journal of Network Security, Vol.14, No.4, PP.229-235, July 2012
- Multilevel Security Algorithm for Bluetooth Technology, International Journal for Research in Technological Studies ISSN: - Applied (Online) Vol-1, Issue - 1, Dec 2013
- Web Sites: www.sves-srpt.ac.in
  www.kings.cam.ac.uk
  www.wellington-college.school.nz
  www.homeandlearn.uk.com