A Computerized Database Encryption System for Standalone Applications

Omorogiuwa, O. & Ohiagu, K.
Department of Computer Science & Information Technology,
College of Natural & Applied Sciences
Igbinedion University
Okada, Edo State, Nigeria.
E-mail: ask4osas@yahoo.com, kingsleyohiagu@aol.com
Phone No: +234080338347172; +2348066583142

ABSTRACT

Information security uses cryptography to transform usable information using an algorithm (called cipher) into a form that renders it unusable by anyone other than an authorized user through a process called encryption. Information that has been encrypted can be transformed back into its original meaningful form by an authorized user, who possesses the cryptographic key, through the process of decryption. In recent times, most cipher algorithms have being implemented in internet based online applications to prevent the various attacks by hackers. Users of standalone applications also suffer attacks to their stored databases and information by malicious users. To make standalone applications more secured, there is a strong need to also implement cryptographic mechanism to protect its information and stored databases from intruders. Small scale organizations that do not need the internet to carry out activities still need some measure of database security. This research work provides an overview of computer security techniques with special reference to data encryption. Ultimately, encrypted database software was designed using Rijndael AES algorithm. The software was designed using Visual BASIC programming language.

Keywords: Rijndael algorithm, cryptography, encryption, DES, AES, database security& standalone applications.

1. BRIEF BACKGROUND INFORMATION

Information security uses cryptography to transform usable information into a form that renders it unusable by anyone other than an authorized user; this process is called encryption. Information that has been encrypted can be transformed back into its original usable form by an authorized user, who possesses the cryptographic key, through the process of decryption. Cryptography is used in information security to protect information from unauthorized or accidental discloser while the information is in transit (either electronically or physically) and while information is in storage. Cryptography provides information security with other useful applications as well as including improved authentication methods, digital signatures, non-repudiation, and encrypted network communications. Cryptographic solutions need to be implemented using industry accepted solutions that have undergone rigorous peer review by independent experts in cryptography. The length and strength of the encryption key is also an important consideration. A key that is weak or too short will produce weak encryption. The keys used for encryption and decryption must be protected with the same degree of rigor as any other confidential information. They must be protected from unauthorized disclosure and destruction and they must be available when needed.

In cryptography, a Caesar cipher, also known as a Caesar's cipher, the shift cipher, Caesar's code or Caesar shift, is one of the simplest and most widely known encryption techniques. It is a type of substitution cipher in which each letter in the plaintext is replaced by a letter some fixed number of positions down the alphabet. For example, with a shift of 4, A would be replaced by E, B would become F, and so on. The method is named after Julius Caesar, who used it to communicate with his generals. This research study will seek to design and implement an encrypted database cryptographic system using Rijndael AES algorithm.

Today's cryptography is vastly more complex than its predecessor. Unlike the original use of cryptography in its classical roots where it was implemented to conceal both diplomatic and military secrets from the enemy, the cryptography of today, even though it still has far-reaching military implications, has expanded its domain, and has been designed to provide a cost-effective means of securing and thus protecting large amounts of electronic data that is stored and communicated across corporate networks worldwide. Cryptography offers the means for protecting this data all the while preserving the privacy of critical personal financial, medical, and ecommerce data that might end up in the hands of those who should not have access to it. There have been many advances in the area of modern cryptography that have emerged beginning in the 1970s as the development of strong encryption-based protocols and newly developed cryptographic applications began to appear on the scene. On January, 1977, the National Bureau of Standards (NBS) adopted a data encryption standard called the Data Encryption Standard (DES), which was a milestone in launching cryptography research and development into the modern age of computing technology [1]. Moreover, cryptography found its way into the commercial arena when, on December, 1980, the same algorithm, DES, was adopted by the American National Standards Institute (ANSI). Following this milestone was yet another when a new concept was proposed to develop Public Key Cryptography (PKC), which is still undergoing research development [2].

When we speak of modern cryptography, we are generally referring to cryptosystems because the cryptography of today involves the study and practice of hiding information through the use of keys, which are associated with Web-based applications, ATMs, Ecommerce, computer passwords, and the like. Cryptography is considered not only a part of the branch of mathematics, but also a branch of computer science. There are two forms of cryptosystems: symmetric and asymmetric. Symmetric cryptosystems involve the use of a single key known as the secret key to encrypt and decrypt data or messages. Asymmetric cryptosystems, on the other hand, use one key (the public key) to encrypt messages or data, and a second key (the secret key) to decipher or decrypt those messages or data. For this reason, asymmetric cryptosystems are also known as public key cryptosystems. The problem that symmetric cryptosystems have always faced is the lack of a secure means for the sharing of the secret key by the individuals who wish to secure their data or communications. Public key cryptosystems solve this problem through the use of cryptographic algorithms used to create the public key and the secret key, such as DES, and a much stronger algorithm, RSA.

The RSA algorithm is the most popular form of public key cryptosystem, which was developed by Ron Rivest, Adi Shamir, and Leonard Adleman at the Massachusetts Institute of Technology in 1977 [3]. The RSA algorithm involves the process of generating the public key by multiplying two very large (100 digits or more) randomly chosen prime numbers, and then, by randomly choosing another very large number, called the encryption key. The public key would then consist of both the encryption key and the product of those two primes. Ron Rivest then developed a simple formula by which someone who wanted to scramble a message could use that public key to do so. The plaintext would then be converted to ciphertext, which was transformed by an equation that included that large product. Using an algorithm developed through the work of the great mathematician, Euclid, Ron Rivest provided for a decryption key, one that could only be calculated by the use of the original two prime numbers. Using this encryption key would unravel the ciphertext and transform it back into its original plaintext. What makes the RSA algorithm strong is the mathematics that is involved. Ascertaining the original randomly chosen prime numbers and the large randomly chosen number (encryption key) that was used to form the product that encrypted the data in the first place is nearly impossible [2].

A very popular public key cryptosystem is known as Pretty Good Privacy (PGP), developed by Phil Zimmerman beginning in early 1991 [2]. The strength of the keys that are created to encrypt and decrypt data or communications is a function of the length of those keys. Typically the longer the key, the stronger that key is. For example, a 56-bit key (consisting of 56 bits of data) would not be as strong as a 128-bit key. And, consequently, a 128-bit key would not be as strong as a 256- or 1024-bit key. Database encryption system can help to protect organizational staff who may want to perform some ill practices to organizational stored databases. Database security includes a broad knowledge such as access control, application access, vulnerability interference and auditing mechanisms. These knowledge areas are required in developing a secured database system for organizations [4]. Pistoia et al., [5] conducted on system security and concluded that the three areas of security vulnerability in software systems are access-control, information flow, and application-programming interface conformance.

1.1 Research Problem

Most existing cryptographic systems have gained so much usefulness in curbing online, real time systems. This is because most organizations tend to be geared towards using internet based services in enhancing and facilitating their activities. However, not much emphasis is laid on standalone applications.Databases in standalone applications, if not effectively protected by encryption mechanisms is prone to insiders attack. Despite the growth in internet technology, mobile computing and ubiquitous computing, some organizations still depend on standalone applications in carrying out there day to day information processing. This is premised on various advantages such as; standalone applications cannot be attacked by hackers on the internet because it is not internet based; the level of intruders is reduced to organizational staffs that are given access by way of authorization to use such applications. Therefore, there is the strong need to develop a database encryption mechanism and organizations can use in information security.

1.2 Research Direction

The research is intended to design and implement a computerized encrypted database system for standalone applications using Advanced Encryption Standard (AES), also known as Rijndael algorithm. This will provide likely solutions or techniques that can curb or reduce the insider threat to computer systems databases. It will also provide facilities for storing user information, encrypting and decrypting user information and provision of facilities for protecting information from unauthorized or accidental disclosure while the information is in transit (either electronically or physically).

2. RELATED WORKS

Malhotra et al., [6] proposed a hybrid cryptography approach known as a symmetric key technique. Also a new process where the principle of computer graphics and properties of geometrical shape were presented to encrypt data. Two geometrical shapes, the ellipse and the rectangle were taken as cover where the information is placed and series of geometric transformation operations are defined to encode the information which uses the properties of ellipse, rectangle and symmetric key algorithm. The work also included geometric shape and alternative transformation to enable information security to be achieved to a greater degree.

Kenekayoro [7] based on an investigative research on DES, which was the first encryption system to meet the National institute of standards and technology's requirement and certified as an encryption system and also incidentally standardized thereafter. Also, a critical review of some of the challenges and criticisms encountered by industry experts as they use encryption system came to the conclusion that DES still had the capacity to effectively secure our confidential information based on already published cryptoanalysis, [7]. The security goals of the proposed algorithms by many researchers such as AES, DES, RSA were enhanced to maintain the security on the communication channels employed for encrypting and decrypting data. This was done to make it difficult for an attacker to predict a pattern for encrypting and decrypting and the speed of the encryption/decryption scheme [8]. With respect to Chosen Ciphertext Attacks (CCA), Boneh et al., [9] proposed a CCA-secure public-key encryption scheme based on Identity-Based Encryption (IBE). These schemes provide for a new paradigm for achieving CCA-security, which avoids "proofs of well-formedness" that formed the basis of developing a strong encryption mechanism.

It is important to state here that employing encryption based on cryptographic algorithms to secure consumer data is of paramount importance to organizational database. It is in this vain that Toubba [10] stresses the importance of strong encryption key management and granular access control to computer applications. In the study carried out by Toubba, it shows that corporations that store, transmit, and use consumer data must take steps to choose strong cryptographic solutions to protect this data, and to employ complementary network security procedures to maximize the overall effectiveness of the encryption product. Strong key management and granular access control are viewed as the complementary network security procedures. It was further shown in a study [11], that the use of public key cryptography based on asymmetric key ciphers overcomes the shortcomings of using symmetric key ciphers in isolation by enabling confidentiality, message integrity, and authentication. The ability to break a cryptosystem was further demonstrated and that the authentication problem of their protocol that allowed them to break this seemingly "unbreakable data encryption" is fixable, [12]. Despite the usefulness of cryptography in securing information systems there is till skepticism on its ability to actually protect organization database [13].

3. OVERVIEW OF THE RIJNDAEL AES ALGORITHM

The Advanced Encryption Standard (AES), also known as Rijndael (its original name), is a specification for the encryption of electronic data established by the U.S. National Institute of Standards and Technology (NIST) in 2001 (Boneh et al., 2006). AES is based on the Rijndael cipher developed by two Belgian cryptographers, Joan Daemen and Vincent Rijmen, who submitted a proposal to NIST during the AES selection process. Rijndael is a family of ciphers with different key and block sizes (Boneh et al., 2006). For AES, NIST selected three members of the Rijndael family, each with a block size of 128 bits, but three different key lengths: 128, 192 and 256 bits. AES has been adopted by the U.S. government and is now used worldwide. It supersedes the Data Encryption Standard (DES), which was published in 1977. The algorithm described by AES is a symmetric-key algorithm, meaning the same key is used for both encrypting and decrypting the data. AES is included in the ISO/IEC 18033-3 standard. AES is available in many different encryption packages, and is the first publicly accessible and open cipher approved by the National Security Agency (NSA) for top secret information when used in an NSA approved cryptographic module.

3.1 Description of the Cipher

AES is based on a design principle known as a substitution-permutation network, combination of both substitution and permutation, and is fast in both software and hardware. Unlike its predecessor DES, AES does not use a Feistel network. AES is a variant of Rijndael which has a fixed block size of 128 bits, and a key size of 128, 192, or 256 bits. By contrast, the Rijndael specification per se is specified with block and key sizes that may be any multiple of 32 bits, both with a minimum of 128 and a maximum of 256 bits [9]. AES operates on a 4×4 column-major order matrix of bytes, termed the state, although some versions of Rijndael have a larger block size and have additional columns in the state. Most AES calculations are done in a special finite field.

The key size used for an AES cipher specifies the number of repetitions of transformation rounds that convert the input, called the plaintext, into the final output, called the ciphertext.

The number of cycles of repetition are as follows [9]:

- 10 cycles of repetition for 128-bit keys.
- 12 cycles of repetition for 192-bit keys.
- 14 cycles of repetition for 256-bit keys.

Each round consists of several processing steps, each containing four similar but different stages, including one that depends on the encryption key itself. A set of reverse rounds are applied to transform ciphertext back into the original plaintext using the same encryption key.

3.2 High-Level Description of the Algorithm

1. Key Expansions - 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 [8-9].
2. Initial Round
    a. AddRoundKey - Each byte of the state is combined with a block of the round key using bitwise xor.
3. Rounds
    a. SubBytes – A non-linear substitution step where each byte is replaced with another according to a lookup table.
    b. Shift Rows - A transposition step where the last three rows of the state are shifted cyclically a certain number of steps.
    c. Mix Columns - A mixing operation which operates on the columns of the state, combining the four bytes in each column.
    d. Add Round Key
4. Final Round (no Mix Columns)
    a. Sub Bytes
    b. Shift Rows
    c. Add Round Key.

4. THE PROPOSED SYSTEM

4.1 Systems Analysis

The systems we are investigating are the existing cryptography software, which are usually internet-based systems. That is they are applicable or useful in the protection of internet files or systems in the global network. This is not too appropriate because intruders do not only attack or intrude into system within a network or internet files but can also attack standalone systems and files or database in this standalone system. A review of literatures revealed that most existing cryptography system has a lot of preventive facilities but their major disadvantage is that they are functional online. That is they are internet-based. A situation when a user of a standalone system wants his system to be protected and prevented from unauthorized users or intruders may not get a solution to his need. This will result in the user's system being accessed by any one and as such his files and information can easily be attacked or affected and information stolen.

In view of the above weaknesses or problems identified in existing cryptography systems used for protecting systems information and preventing intruders from using the system, there is need for the development of a cryptography system that will protect standalone systems information and also protect intruders from having access to such standalone systems.
Such alternative system will seek to provide prevention and protective facilities for standalone systems which others are not able to provide. Standalone applications/systems are software that are not internet based and as such cannot be used online. They are mostly customized applications developed to meet the information processing needs of mainly small scale organizations.

4.2 The Proposed System

The proposed system is a computerized encrypted database system. The proposed system adopted the Rijndael AES algorithm for encrypting and decrypting information. The system will be user friendly. It will be designed with features, which provides users with input screen such that a user can enter his/her access code to login to the system. The system generally will provide facilities for storing, encrypting and decrypting stored database; provide facilities for protecting information from unauthorized user access. The encrypted database system framework is given below:

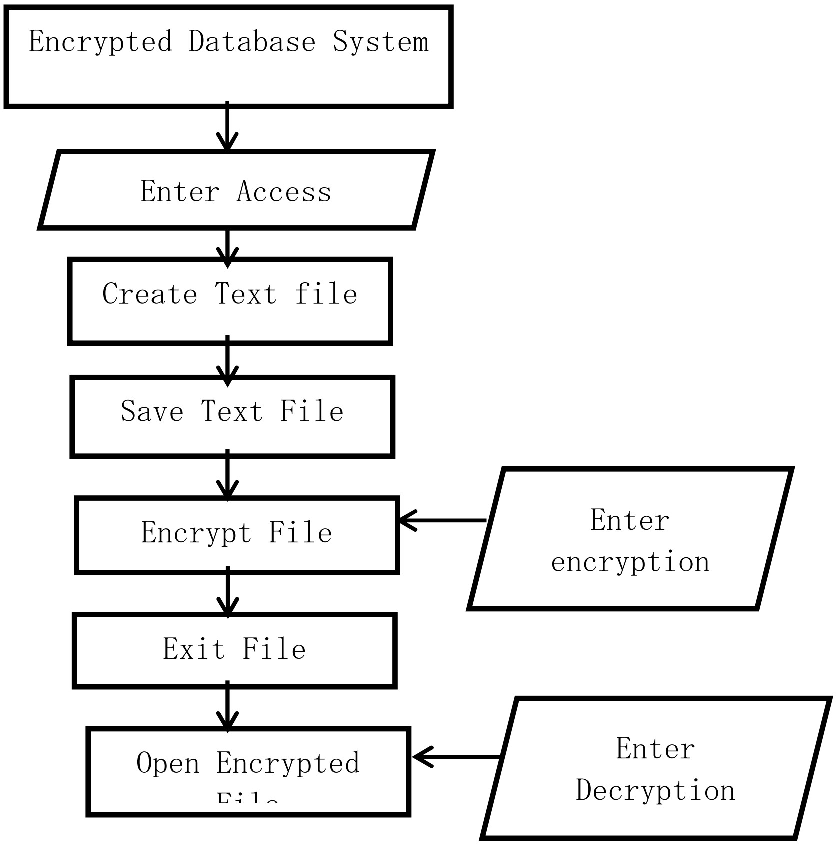


Figure 1: Encrypted Database System framework

4.2 Proposed System Algorithm

An algorithm is a finite logical sequence of steps for solving a problem that can be translated into a computer program. The processing function of the Encrypted Database System is broken down into three major tasks namely: information storage, information retrieval, information encryption and decryption. Algorithms are then designed for these tasks and these algorithms coded in the Visual BASIC 6.0 programming language.

The algorithms for the system are presented below:

Task 1: Algorithm for Saving User Information

```
i.)     Enter Information
ii.)    Click on File Menu
iii.)   Click on Save Menu
iv.)    Enter File name
v.)     Check File Name if Already Exist
vi.)    If Yes, Display Message “Filename already exist, overwrite [Y/N]”
        a.  If user response is Yes, systems saves information and go to step ix, otherwise go to step iv
            to re-enter filename.
vii.)   Else
viii.)  Save Information
ix.)    Stop
```

Task 2: Algorithm for Opening Existing Information

```
i.)     Click on File Menu
ii.)    Click on Open Menu
iii.)   Enter Filename
iv.)    Check system for Filename
v.)     If Found, Display “ Enter Access Key”
        a.  Check System Key if Valid
                i.  If key entered is valid, then open file for user to go to step i in Task 3, otherwise
                    display message “invalid access key entered, access denied, go to step vi.
        b.  Otherwise, display “ filename does not exist”
vi.)    Stop
```

Task 3: Algorithm for Encrypting Information

```
i.)     Enter Information
ii.)    Click on Encrypt Data Menu
iii.)   Enter Access Key
iv.)    Encrypt Information
v.)     Save Information
vi.)    Stop
```

Task 4: Algorithm for Decrypting Information

```
i.)     Click on File Menu
ii.)    Click on Open Menu
iii.)   Enter File Name
iv.)    Check for Filename
v.)     If Found, then prompt user to enter access key
vi.)    Check access key entered
        a.  If Okay, decrypt information and display information for user view, otherwise, display message
            “invalid access key, access denied” goto step viii.
vii.)   Otherwise, display message “File not found”.
viii.)  Stop
```

4.3 Program Design

The software was designed using modular approach and Visual Basic 6.0 programming language was used for the program development. The software was ran to determine whether it achieves the objectives of development. The running of the program usually reveals errors that need to be corrected. However, with careful design of the software, the errors were minimized. Also, testing was done in the following four phases; unit testing, integration testing, system testing and user acceptance testing. In other words, the system comprises of several modules coming together to form the entire encrypted database system. Each module was designed and developed separately and later linked together to form the coherent system. There is a built in security function in the system using the password approach which allows only an authorized user to have access to the systems information. A user wishing to decrypt and encrypted file will first have to enter his or her access key (password).

5. SYSTEM IMPLEMENTATION

The software can be installed in any system with at least windows 98 operating system and above. To launch the software; Click on the start button, Select All Programs, Select the software named Encrypted Database System, Click on the software name Encrypted Database System. Clicking on the software name launches the splash screen. After some seconds, the main menu is displayed (Figure 2).

To encrypt data or information, the following steps should be followed; Click on File menu, and next on New menu, Enter the textual information to be encrypted, Click on Key menu, Click on Encrypt Data, Clicking on the Encrypt Data menu encrypts the information (Figure 3).

To decrypt data or information, the following steps should be followed; Click on Key menu, Click on Decrypt Data menu, Enter Access code (Figure 4), Click on Login button, Click on Ok button to respond to the prompt message, Click on decrypt Data button, Clicking on Decrypt Data button, decrypts the encrypted information (Figure 5). Also, a user can as well save his database information for future reference. This can be achieved by following these steps; Click on File menu, Click on Save As menu, Enter File name and Click on Save button.

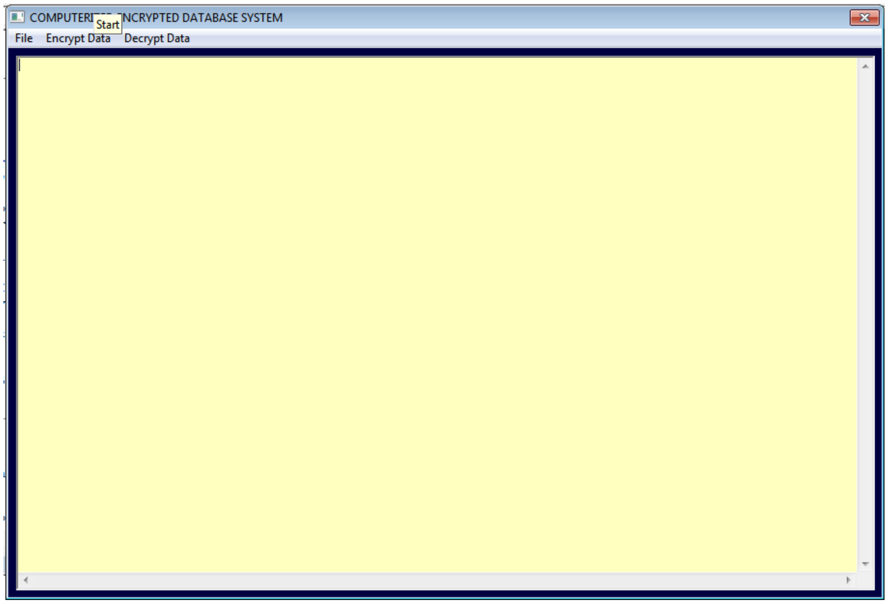
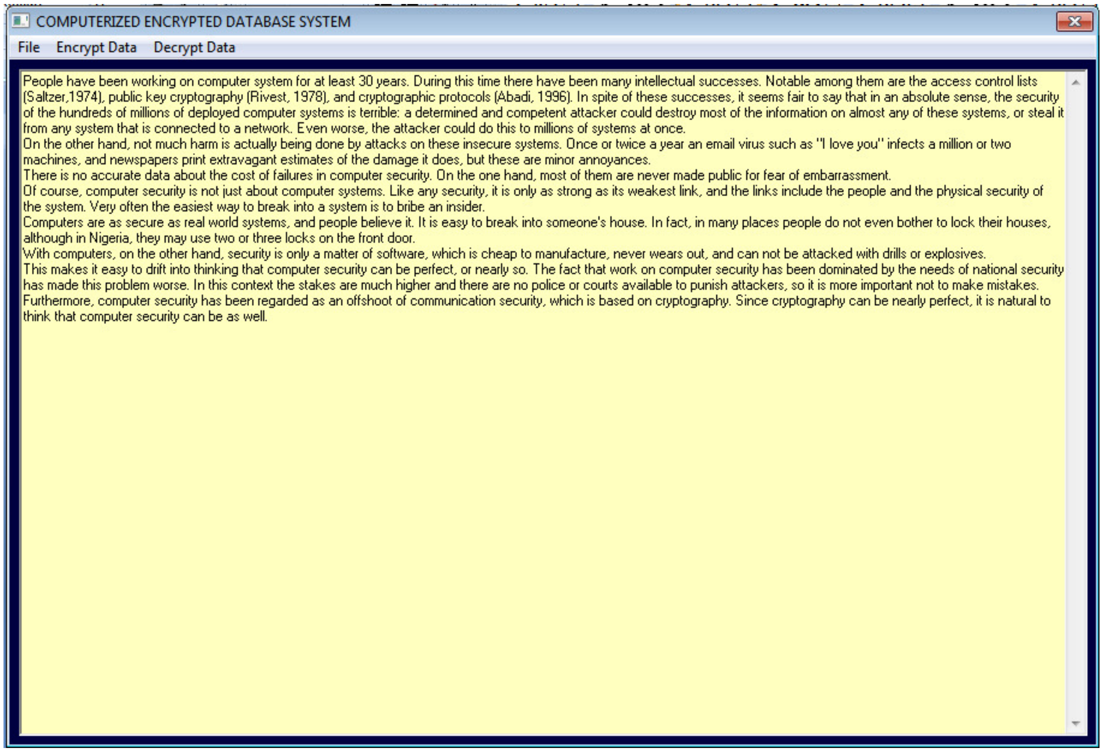
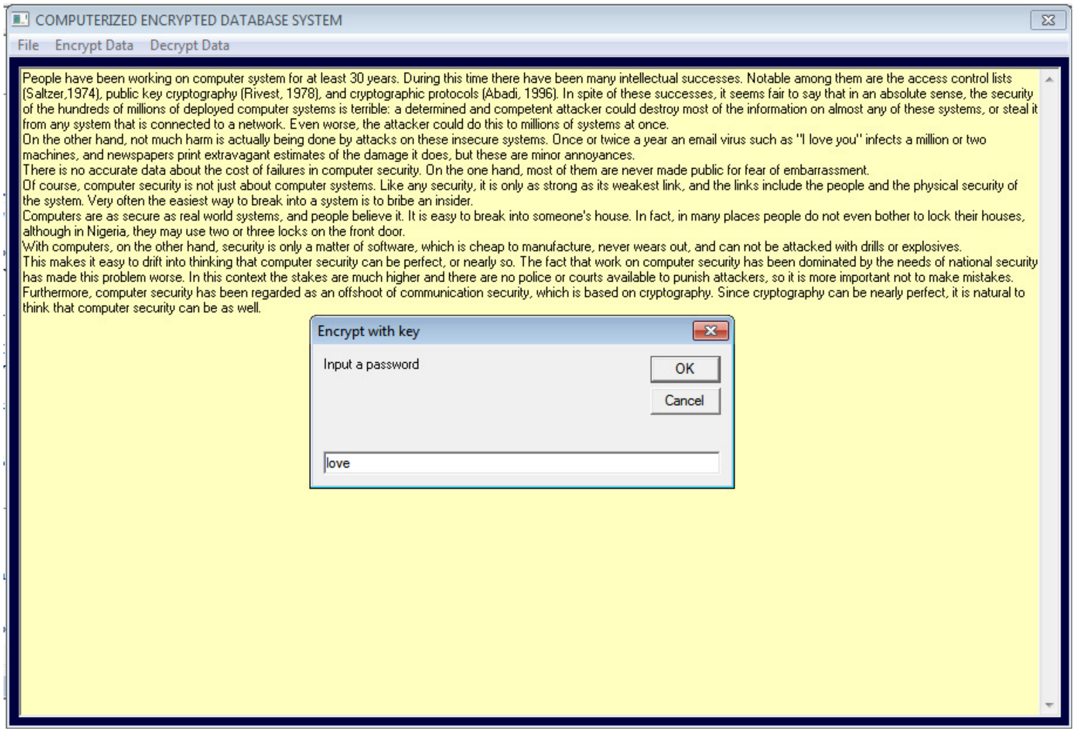


Figure 2: Main Menu Window of the Computerized Encryption Database System



People have been working on computer system for at least 30 years. During this time there have been many intellectual successes. Notable among them are the access control lists (Saltzer,1974), public key cryptography (Rivest, 1978), and cryptographic protocols (Abadi, 1996). In spite of these successes, it seems fair to say that in an absolute sense, the security of the hundreds of millions of deployed computer systems is terrible: a determined and competent attacker could destroy most of the information on almost any of these systems, or steal it from any system that is connected to a network. Even worse, the attacker could do this to millions of systems at once.
On the other hand, not much harm is actually being done by attacks on these insecure systems. Once or twice a year an email virus such as "I love you" infects a million or two machines, and newspapers print extravagant estimates of the damage it does, but these are minor annoyances.
There is no accurate data about the cost of failures in computer security. On the one hand, most of them are never made public for fear of embarrassment.
Of course, computer security is not just about computer systems. Like any security, it is only as strong as its weakest link, and the links include the people and the physical security of the system. Very often the easiest way to break into a system is to bribe an insider.
Computers are as secure as real world systems, and people believe it. It is easy to break into someone's house. In fact, in many places people do not even bother to lock their houses, although in Nigeria, they may use two or three locks on the front door.
With computers, on the other hand, security is only a matter of software, which is cheap to manufacture, never wears out, and can not be attacked with drills or explosives.
This makes it easy to drift into thinking that computer security can be perfect, or nearly so. The fact that work on computer security has been dominated by the needs of national security has made this problem worse. In this context the stakes are much higher and there are no police or courts available to punish attackers, so it is more important not to make mistakes.
Furthermore, computer security has been regarded as an offshoot of communication security, which is based on cryptography. Since cryptography can be nearly perfect, it is natural to think that computer security can be as well.

Figure 3: Sample Input Data for Encryption

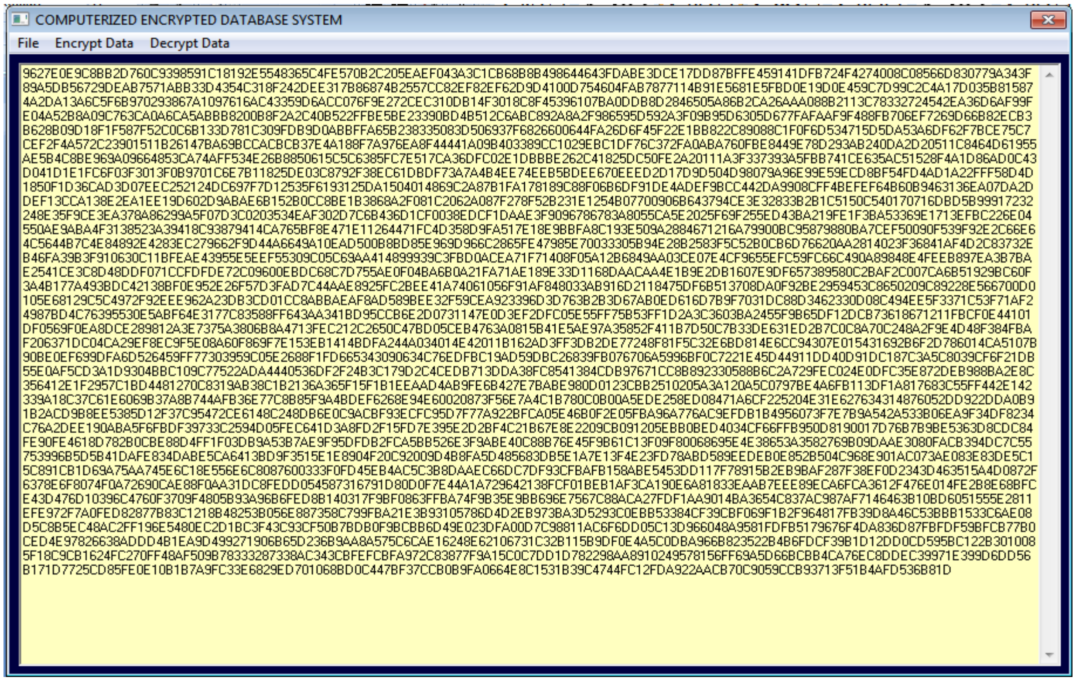


People have been working on computer system for at least 30 years. During this time there have been many intellectual successes. Notable among them are the access control lists (Saltzer,1974), public key cryptography (Rivest, 1978), and cryptographic protocols (Abadi, 1996). In spite of these successes, it seems fair to say that in an absolute sense, the security of the hundreds of millions of deployed computer systems is terrible: a determined and competent attacker could destroy most of the information on almost any of these systems, or steal it from any system that is connected to a network. Even worse, the attacker could do this to millions of systems at once.
On the other hand, not much harm is actually being done by attacks on these insecure systems. Once or twice a year an email virus such as "I love you" infects a million or two machines, and newspapers print extravagant estimates of the damage it does, but these are minor annoyances.
There is no accurate data about the cost of failures in computer security. On the one hand, most of them are never made public for fear of embarrassment.
Of course, computer security is not just about computer systems. Like any security, it is only as strong as its weakest link, and the links include the people and the physical security of the system. Very often the easiest way to break into a system is to bribe an insider.
Computers are as secure as real world systems, and people believe it. It is easy to break into someone's house. In fact, in many places people do not even bother to lock their houses, although in Nigeria, they may use two or three locks on the front door.
With computers, on the other hand, security is only a matter of software, which is cheap to manufacture, never wears out, and can not be attacked with drills or explosives.
This makes it easy to drift into thinking that computer security can be perfect, or nearly so. The fact that work on computer security has been dominated by the needs of national security has made this problem worse. In this context the stakes are much higher and there are no police or courts available to punish attackers, so it is more important not to make mistakes.
Furthermore, computer security has been regarded as an offshoot of communication security, which is based on cryptography. Since cryptography can be nearly perfect, it is natural to think that computer security can be as well.

Input a password

love

Figure 4: Sample Output Data to be Decrypted Using a Specified Access Key



```
9627E0E9C8BB2D760C9398591C18192E5548365C4FE570B2C205EAEF043A3C1CB68B8B498644643FDABE3DCE17DD87BFFE459141DFB724F4274008C08566D830779A343F
89A5DB56729DEAB7571ABB33D4354C318F242DEE317B86874B2557CC82EF82EF62D9D4100D754604FAB7877114B91E5681E5FBD0E19D0E459C7D99C2C4A17D035B81587
4A2DA13A6C5F6B970293867A1097616AC43359D6ACC076F9E272CEC310DB14F3018C8F45396107BA0DDB8D2846505A86B2CA26AAA088B2113C78332724542EA36D6AF99F
E04A52B8A09C763CA0A6CA5ABBB8200B8F2A2C40B522FFBE5BE23390BD4B512C6ABC892A8A2F986595D592A3F09B95D6305D677FAFAAF9F488FB706EF7269D66B82ECB3
B628B09D18F1F587F52C0C6B133D781C309FDB9D0ABBFFA65B238335083D506937F6826600644FA26D6F45F22E1BB822C89088C1F0F6D534715D5DA53A6DF62F7BCE75C7
CEF2F4A572C23901511B26147BA69BCCACBCB37E4A188F7A976EA8F44441A09B403389CC1029EBC1DF76C372FA0ABA760FBE8449E78D293AB240DA2D20511C8464D61955
AE5B4C8BE969A09664853CA74AFF534E26B8850615C5C6385FC7E517CA36DFC02E1DBBBE262C41825DC50FE2A20111A3F337393A5FBB741CE635AC51528F4A1D86AD0C43
D041D1E1FC6F03F3013F0B9701C6E7B11825DE03C8792F38EC61DBDF73A7A4B4EE74EEB5BDEE670EEED2D17D9D504D98079A96E99E59ECD8BF54FD4AD1A22FFF58D4D
1850F1D36CAD3D07EEC252124DC697F7D12535F6193125DA1504014869C2A87B1FA178189C88F06B6DF91DE4ADEF9BCC442DA9908CFF4BEFEF64B60B9463136EA07DA2D
DEF13CCA138E2EA1EE19D602D9ABAE6B152B0CC8BE1B3868A2F081C2062A087F278F52B231E1254B07700906B643794CE3E32833B2B1C5150C540170716DBD5B99917232
248E35F9CE3EA378A86299A5F07D3C0203534EAF302D7C6B436D1CF0038EDCF1DAAE3F9096786783A8055CA5E2025F69F255ED43BA219FE1F3BA53369E1713EFBC226E04
550AE9ABA4F3138523A39418C93879414CA765BF8E471E11264471FC4D358D9FA517E18E9BBFA8C193E509A2884671216A79900BC95879880BA7CEF50090F539F92E2C66E6
4C5644B7C4E84892E4283EC279662F9D44A6649A10EAD500B8BD85E969D966C2865FE47985E70033305B94E28B2583F5C52B0CB6D76620AA2814023F36841AF4D2C83732E
B46FA39B3F910630C11BFEAE43955E5EEF55309C05C69AA414899939C3FBD0ACEA71F71408F05A12B6849AA03CE07E4CF9655EFC59FC66C490A89848E4FEEB897EA3B7BA
E2541CE3C8D48DDF071CCFDFDE72C09600EBDC68C7D755AE0F04BA6B0A21FA71AE189E33D1168DAACAA4E1B9E2DB1607E9DF657389580C2BAF2C007CA6B51929BC60F
3A4B177A493BDC42138BF0E952E26F57D3FAD7C44AAE8925FC2BEE41A74061056F91AF848033AB916D2118475DF6B513708DA0F92BE2959453C8650209C89228E566700D0
105E68129C5C4972F92EEE962A23DB3CD01CC8ABBAEAF8AD589BEE32F59CEA923396D3D763B2B3D67AB0ED616D7B9F7031DC88D3462330D08C494EE5F3371C53F71AF2
4987BD4C76395530E5ABF64E3177C83588FF643AA341BD95CCB6E2D0731147E0D3EF2DFC05E55FF75B53FF1D2A3C3603BA2455F9B65DF12DCB73618671211FBCF0E44101
DF0569F0EA8DCE289812A3E7375A3806B8A4713FEC212C2650C47BD05CEB4763A0815B41E5AE97A35852F411B7D50C7B33DE631ED2B7C0C8A70C248A2F9E4D48F384FBA
F206371DC04CA29EF8EC9F5E08A60F869F7E153EB1414BDFA244A034014E42011B162AD3FF3DB2DE77248F81F5C32E6BD814E6CC94307E015431692B6F2D786014CA5107B
90BE0EF699DFA6D526459FF77303959C05E2688F1FD665343090634C76EDFBC19AD59DBC26839FB076706A5996BF0C7221E45D44911DD40D91DC187C3A5C8039CF6F21DB
55E0AF5CD3A1D9304BBC109C77522ADA4440536DF2F24B3C179D2C4CEDB713DDA38FC8541384CDB97671CC8B892330588B6C2A729FEC024E0DFC35E872DEB988BA2E8C
356412E1F2957C1BD4481270C8319AB38C1B2136A365F15F1B1EEAAD4AB9FE6B427E7BABE980D0123CBB2510205A3A120A5C0797BE4A6FB113DF1A817683C55FF442E142
339A18C37C61E6069B37A8B744AFB36E77C8B85F9A4BDEF6268E94E60020873F56E7A4C1B780C0B00A5EDE258ED08471A6CF225204E31E627634314876052DD922DDA0B9
1B2ACD9B8EE5385D12F37C95472CE6148C248DB6E0C9ACBF93ECFC95D7F77A922BFCA05E46B0F2E05FBA96A776AC9EFDB1B4956073F7E7B9A542A533B06EA9F34DF8234
C76A2DEE190ABA5F6FBDF39733C2594D05FEC641D3A8FD2F15FD7E395E2D2BF4C21B67E8E2209CB091205EBB0BED4034CF66FFB950D8190017D76B7B9BE5363D8CDC84
FE90FE4618D782B0CBE88D4FF1F03DB9A53B7AE9F95DFDB2FCA5BB526E3F9ABE40C88B76E45F9B61C13F09F80068695E4E38653A3582769B09DAAE3080FACB394DC7C55
753996B5D5B41DAFE834DABE5CA6413BD9F3515E1E8904F20C92009D4B8FA5D485683DB5E1A7E13F4E23FD78ABD589EEDEB0E852B504C968E901AC073AE083E83DE5C1
5C891CB1D69A75AA745E6C18E556E6C8087600333F0FD45EB4AC5C3B8DAAEC66DC7DF93CFBAFB158ABE5453DD117F78915B2EB9BAF287F38EF0D2343D463515A4D0872F
6378E6F8074F0A72690CAE88F0AA31DC8FEDD054587316791D80D0F7E44A1A729642138FCF01BEB1AF3CA190E6A81833EAAB7EEE89ECA6FCA3612F476E014FE2B8E68BFC
E43D476D10396C4760F3709F4805B93A96B6FED8B140317F9BF0863FFBA74F9B35E9BB696E7567C88ACA27FDF1AA9014BA3654C837AC987AF7146463B10BD6051555E2811
EFE972F7A0FED82877B83C1218B48253B056E887358C799FBA21E3B93105786D4D2EB973BA3D5293C0EBB53384CF39CBF069F1B2F964817FB39D8A46C53BBB1533C6AE08
D5C8B5EC48AC2FF196E5480EC2D1BC3F43C93CF50B7BDB0F9BCBB6D49E023DFA00D7C98811AC6F6DD05C13D966048A9581FDFB5179676F4DA836D87FBFDF59BFCB77B0
CED4E97826638ADDD4B1EA9D499271906B65D236B9AA8A575C6CAE16248E62106731C32B115B9DF0E4A5C0DBA966B823522B4B6FDCF39B1D12DD0CD595BC122B301008
5F18C9CB1624FC270FF48AF509B78333287338AC343CBFEFCBFA972C83877F9A15C0C7DD1D782298AA8910249578156FF69A5D66BCBB4CA76EC8DDEC39971E399D6DD56
B171D7725CD85FE0E10B1B7A9FC33E6829ED701068BD0C447BF37CCB0B9FA0664E8C1531B39C4744FC12FDA922AACB70C9059CCB93713F51B4AFD536B81D
```

Figure 5: Sample Output Data of Decrypted Information

6. CONCLUSION

This research study presented a framework for the development and implementation of an Encrypted Database System using Rijndael AES algorithm. A background review of some cryptographic systems was conducted with a view of understanding their strength and weaknesses. Ultimately, a Computerized Encrypted Database System for standalone applications was developed. The implementation of this system will help to ensure that intruders into computers systems files and computer users who are involve in carrying out malicious acts on system being used are prevented. Once used according to specification with the authentication checks in place, the system will no doubt provide security facilities for system users. Future research work in this direction can be done by implementing framework using other ciphering algorithms.

REFERENCES

1. Callas, J. (2007). “The Future of Cryptography”, Information Systems Security, 16(1), pp 15-22.
2. Lee, S. and Lee, P. (2004). Crypto: How the Code Rebels Beat the Government - Saving Privacy in the Digital Age. New York: Viking Penguin Publishing.
3. Robinson S. (2008). “Safe and Secure: Data Encryption for Embedded Systems, EDN Europe, Vol. 53(6), pp 24-33.
4. Murray M. C. (2010). “Database Security: What students need to know”. Journal of Information Technology Education, vol. 6 pp 61 – 77.
5. Pistoia, M., Chandra, S., Fink S., and Yahay E. (2007). “A Survey of Static Analysis Methods for Identifying Security Vulnerabilities in Software Systems. IBM Systems Journal, vol. 46(2), pp 265-288.
6. Malhotra R. T. (2015). “A Hybrid Geometric Cryptography Approach to Enhance Information Security”. Journal of Network Communications and Emerging Technology, vol 3(1) pp 23 – 36.
7. Kenekayoro Patrick T. (2010). “The data encryption standard thirty years later: An overview”. African Journal of Mathematics and Computer Science Research, vol 3(10), pp 267-269.
8. Obaida, M. (2013). “A New Approach for Encrypting and Decrypting Data”. International Journal of Computer Networks and Communication, Vol. 5(2), pp 15 -29.
9. Boneh, D., Canetti, R., Halevi, S., and Katz, J. (2006). Chosen-Ciphertext Security from identity-Based Encryption. *SIAM Journal on Computing*, 36(5), 1301-1328.
10. Toubba K (2006). “Employing Encryption to Secure Consumer Data”. Information Systems Security, vol. 15(3), pp 46-54.
11. Kodaganallur, V. (2006). “Secure E-Commerce: Understanding the Public Key Cryptography Jigsaw Puzzle. Information Systems Security, vol14(6), pp 44-52.
12. Klappenecker, A. (2004). “Remark on a Non-breakable Data Encryption Scheme by Kish and Sethuraman”. Fluctuation and Noise Letters, vol 4(4), pp 25-26.
13. Fagin, B., Baird, L., Humphries, J., and Schweitzer, D. (2008). Skepticism and Cryptography. Knowledge, Technology and Policy, 20(4), 231-242.