International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347 - 6435 (Online) Volume 2, Issue 4, April 2014)

Simulation of AES Based Data Encryption in Vb.NET

Ibrahim M. Harram[1], Mala U. M. Bakura[2], Zainab M. Gwoma[3]

[1]*Department of Electrical and Electronics Engineering Technology, The Federal Polytechnic, Damaturu, Yobe State, Nigeria*
[2,3]*Department of Electrical and Electronics Engineering, University of Maiduguri, Borno State, Nigeria*

***Abstract*- In the present computer era, the need to protect data in communications from spying eyes is greater than ever before. Since times immemorial, security of data to maintain its confidentiality, proper access control, integrity and availability has been a major issue in data communication. This paper focuses mainly on the realization of data encryption using the famous Advanced Encryption Standard Algorithm in Vb.net framework. The study is based on the development of a computer oriented encryption program, using Advanced Encryption Standard (AES) that will ensure the security of data on transit. The data encryption and decryption system described and realized in this work using AES codes is aimed at improving data security and integrity. It has been programmed, tested and proved satisfactory with all the needed modifications and upgrading.**

***Keywords*—AES Algorithm, Data Encryption, Data Decryption, Data Security, Vb.net framework**

I. INTRODUCTION

Cryptography, the science of encryption plays a central role in mobile phone communication, e-commerce, Pay-TV, sending private e-mails, transmitting financial information and touches on many aspects of daily lives. Today's technology can be traced back to earliest ciphers, and have grown as a result of evolution. Encryption was developed to a mechanical process.

With the advent of the computer, the mechanical encryption techniques were replaced with computer ciphers. They operate according to the same principles of substitution and transposition (where the order of letters or bits is altered). Again each cipher depended on choosing a key, known only by the sender and the receiver which defined how a particular message would be. This meant that there still was a problem of getting the key to the receiver so that the message could be deciphered.

For years, the key distribution problem haunted code makers. If you want to decipher a scrambled text you have to know the key in advance. But there was revolution in cryptography known as public key cryptography, which destroyed the key distribution problem. This was a technology made for the internet.

Customers could send credit card details and send them to retailers on the other side of the planet. It formed the basis of all kinds of modern day communications.

II. LITERATURE REVIEW

Before the modern era, cryptography was concerned solely with the message confidentially (i.e., encryption) conversion of messages from a comprehensible form into an incomprehensible one and back again at the other end, rendering it unreadable by interceptors of eavesdroppers without secret knowledge (namely the key needed for decryption of that message) [5]. Encryption was used to (attempt to) ensure secrecy in communications, such as those of spies, military leaders, and diplomats [1]. In recent decades, the field has expanded beyond confidentially concerns to include techniques for massage integrity checking, sender/receiver identity authentication, digital signatures, and interactive proofs and secure computation, among others [2].

Cryptography was recommended as a way for lovers to communicate without inconvenient discovery. Steganography (i.e., hiding even the existence of a message so as to keep it confidential) was also first developed in ancient times. More modern examples of Steganography include the use of invisible ink, microdots, and digital watermarks to conceal information [4].

Ciphertext produced by classical ciphers (and some modern ones) always reveal statistical information about the plaintext, which can often be used to break them. After the discovery of frequency analysis perhaps by the Arab mathematician and polymath, Alkindi (also known as *Akindis)*, in the 9th century, nearly all such ciphers became more or less readily breakable by any informed attacker. Such classical ciphers still enjoy popularity today, though mostly as puzzles [4].

Although frequency analysis is powerful and general technique against many ciphers, encryption has still been effective in practice; many a would-be cryptanalyst was unaware of the technique.

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347 - 6435 (Online) Volume 2, Issue 4, April 2014)

Breaking a message without using frequency analysis essentially required knowledge of the cipher used perhaps of the key involved, thus making espionage, bribery, burglary, defection, etc. more attractive approaches to the cryptanalytically uninformed. It was finally explicitly recognized in the 19th century that secrecy of cipher's algorithm is neither a sensible nor practical safeguard of message security; in fact it was further realized that any adequate cryptographic scheme (including ciphers) should remain secure even if the adversary fully understands the cipher algorithm itself. Security of the key used should alone be sufficient for a good cipher to maintain confidentially under an attack [4]. This fundamental principle was first explicitly stated in 1883 by Auguste Kerckhoff and is generally called Kerkchoff's principle; alternatively and more bluntly, it was restated by Claude Shannon, the inventor of information theory and the fundamental of theoretical cryptography, as *Shannon's Maxim* the enemy knows the system [2].

A. The Advanced Encryption Standard

In cryptography, the Advanced Encryption Standard (AES) is a symmetric-key encryption standard adopted by the U.S. government. The standard comprises three block ciphers, AES-128, AES-192 and AES-256, adopted from a larger collection originally published as Rijndael. Each of these ciphers has a 128-bit block size, with key sizes of 128, 192 and 256 bits, respectively. The AES ciphers have been analyzed extensively and are now used worldwide, as was the case with its predecessor, the Data Encryption Standard (DES).

AES was announced by National Institute of Standards and Technology (NIST) as U.S. FIPS PUB 197 (FIPS 197) on November 26, 2001 after a 5-year standardization process in which fifteen competing designs were presented and evaluated before Rijndael was selected as the most suitable. It became effective as a Federal government standard on May 26, 2002 after approval by the Secretary of Commerce. It is available in many different encryption packages. The Rijndael cipher was developed by two Belgian cryptographers, Joan Daemen and Vincent Rijmen, and submitted by them to the AES selection process. Rijndael is a wordplay based upon the names of the two inventors.

B. Description of the Cipher

AES is based on a design principle known as a substitution permutation network. It is fast in both software and hardware. Unlike its predecessor, DES, AES does not use a Feistel network.

AES has a fixed block size of 128 bits and a key size of 128, 192, or 256 bits, whereas Rijndael can be specified with block and key sizes in any multiple of 32 bits, with a minimum of 128 bits. The block-size has a maximum of 256 bits, but the key-size has no theoretical maximum.
Most AES calculations are done in a special finite field. The AES cipher is specified as a number of repetitions of transformation rounds that convert the input plaintext into the final output of ciphertext. Each round consists of several processing steps, including one that depends on the encryption key. A set of reverse rounds are applied to transform ciphertext back into the original plaintext using the same encryption key.

High-Level Description of the Algorithm: The Algorithm is processed in four (4) steps given below:

1. Key Expansion—round keys are derived from the cipher key using Rijndael's key schedule
2. Initial Round
    a) Add-Round-Key—each byte of the state is combined with the round key using bitwise XOR
3. Rounds
    a) Sub-Bytes—a non-linear substitution step where each byte is replaced with another according to a lookup table.
    b) Shift-Rows—a transposition step where each row of the state is 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)

The Sub-Bytes Step: In this step, each byte in the array is updated 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 inverse 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), and also any opposite fixed points.

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347 - 6435 (Online) Volume 2, Issue 4, April 2014)

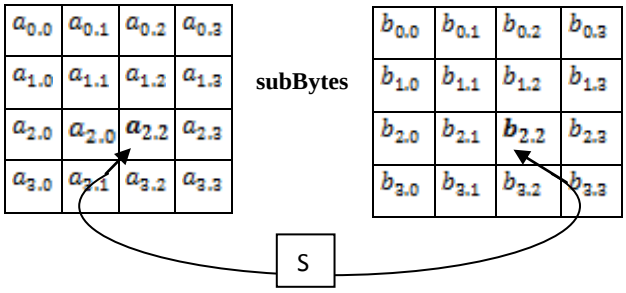

Fig. 1 The sub-Bytes Step

The Shift-Row Step: In the Shift-Rows step, bytes in each row of the state are shifted cyclically to the left. The number of places each byte is shifted differs for each row. The Shift-Rows 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.

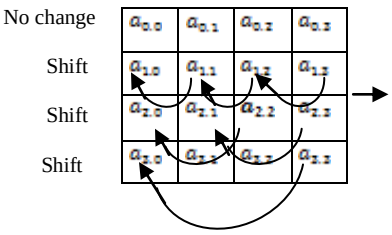

Fig. 1 The Shift-Row Step

For the block of size 128 bits and 192 bits the shifting pattern is the same. In this way, each column of the output state of the Shift-Rows step is composed of bytes from each column of the input state. (Rijndael variants with a larger block size have slightly different offsets). In the case of the 256-bit block, the first row is unchanged and the shifting for second, third and fourth row is 1 byte, 3 bytes and 4 bytes respectively.

The Mix-Columns Step: In the Mix-Columns step, each column of the state is multiplied with a fixed polynomial *c(x)*.In the Mix-Columns step, the four bytes of each column of the state are combined using an invertible linear transformation.

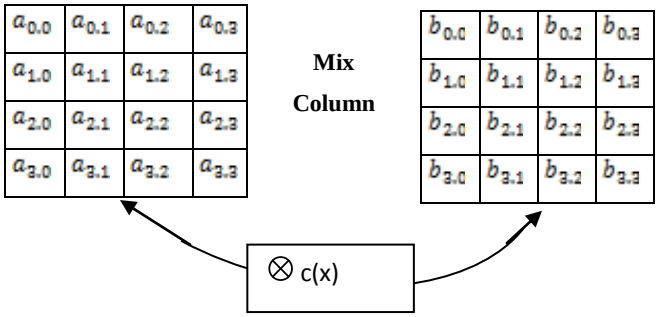

Fig. 3 The Mix-Column Step

The Mix-Columns function takes four bytes as input and outputs four bytes, where each input byte affects all four output bytes. Together with Shift-Rows, Mix-Columns provide diffusion in the cipher. During this operation, each column is multiplied by the known matrix that for the 128 bit key is:

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

The multiplication operation is defined as multiplication by 1 means leaving unchanged, multiplication by 2 means shifting bytes to the left and multiplication by 3 means shifting to the left and then performing XOR with the initial un-shifted value. After shifting, a conditional XOR with 0x11B should be performed if the shifted value is larger than 0xFF.

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)[8].

The Add-Round Key Step: In the Add-Round-Key step, each byte of the state is combined with a byte of the round sub-key using the XOR operation (⊕). In this step, the sub-key is combined with the state. For each round, a sub-key is derived from the main key using Rijndael's key schedule; each sub-key is the same size as the state.

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347 - 6435 (Online) Volume 2, Issue 4, April 2014)

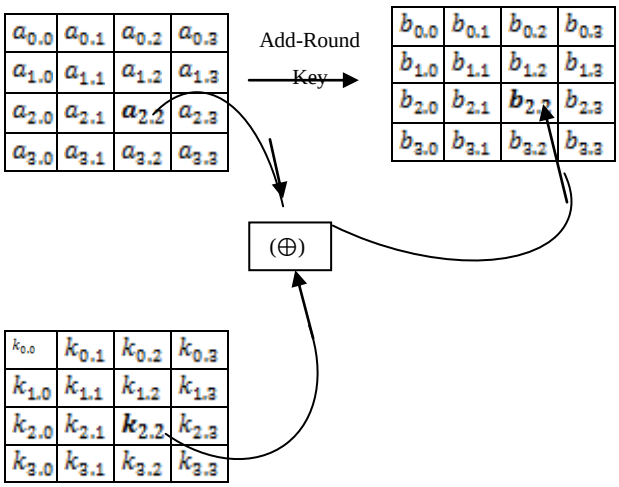

Fig. 4 The Add-Round Step

The sub-key is added by combining each byte of the state with the corresponding byte of the sub-key using bitwise XOR.

III. The AES Code In VB.NET Framework

The AES codes inVb.NET Framework is the abstract base code that extends the properties and methods for using the AES algorithm. This code is inherited by two other classes, *AesCryptoServiceProvider* and *AesManaged*- that represent and managed the implementations of the AES algorithm respectively. In the SecretKeyFile application, AES algorithm is used to demonstrate how to generate secret key, encrypt data with the key, and then decrypt data using the same key. The code-behind file of Form1 (Form1.vb) was opened and the appropriate codes were added to reference the necessary namespace.

A. The Reference Codes

There are four code statements used to reference the System IO; *System.Text, System.Security. Cryptography,* and *System.Runtime.InteropServices* in which the namespaces are added. The TextBox1 and Button1 allow you to specify an input file for encryption or decryption. In TextBox2, you need to specify the output file for encryption or decryption. The codes were added in the Form1 class to generate the secret key and encrypt a given file.

B. The Secret Key Function

The *GenerateSecretKey* function has the code to generate the secret key for the AES algorithm. This method uses the Create method of an *AESCryptoServiceProvider* object and returns the key as a string. In the Button 2 Click Sub procedure, the *GenerateSecretKey* function is called to generate the secret key. The Button 2 Click Sub procedure also contains a call to the *EncryptFile* Sub procedure that encrypts the given input file by using the *AESCryptoServiceProvider* object. The Key properties of the *AESCryptoServiceProvider* object is set to the secret key that was generated earlier. The encrypted data is saved in the given output file by using a *CryptoStream* object. The codes were added in the Form1 class of Form1.vb file to decrypt the encrypted data.

IV. Results And Discussion

Encryption has long been used by militaries and Government to facilitate secret communication and it is now commonly used in protecting information within many kinds of civilian systems. This has achieved the protection of data in transit. For example data being transferred via network (e.g. the internet, e-commerce) mobile telephones, wireless microphones etc. This was made possible using different cipher and computer oriented programs. After successfully loading the codes, the following program was realized. It was tested and proved satisfactory.

Fig. 5 First Image on Start-Up

Figure (5) above shows the first image on start-up as the program is initializing. The next image in figure (6) shows the real encryptor/decryptor interface where the data to be encrypted is loaded. It can be noticed that, as soon as the data is made available for encryption, the encrypt button becomes activated, ready to accept command and act upon.

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347 - 6435 (Online) Volume 2, Issue 4, April 2014)

Fig. 6.The Real Encryptor/Decryptor

As can also be seen in figure (7) below, the data has been encrypted and the Save Encryption button activated. This allows the processing of the encrypted data either for storage or possible transmission over any available medium. At the receiving end, the encrypted data need to be decrypted by loading it on the decryption portion and clicking on the decrypt button. In essence, both parties in the particular network must have the program installed in order to effectively use it.

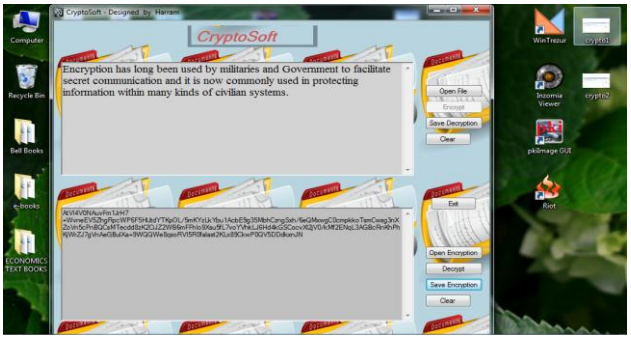

Fig. 7. The Encryption And Decryption Process

V. CONCLUSION

In conclusion, the data encryption and decryption system described and realized in this work using AES codes is aimed at improving data security and integrity. It has been programmed, tested and proved satisfactory with all the needed modifications and upgrading. It should be noted that an attempt by cryptanalyst to develop a technique that would break the effectiveness of any existing cipher technique will not be considered an offence, except if it is aimed at intension by adversary. Hence the development of such a system is imperative. The computer programmed software is also developed to suit the trend in data security. Most ciphers were successfully used to achieve data security, integrity, and privacy.

REFERENCES

[1] Becket, B (1988) Introduction to Cryptography. Blackwell scientific publication.

[2] David F. Wheeler (2009) A Bulk Data Encryption Algorithm computer laboratory, Cambridge University, UK.

[3] Deltart E. (2010), Data Encryption; mixing up the message in the name of security: http://www/acm/org/crossroads/xrds15-1/encryption.html

[4] Ibrahim A. Al-kadi (1992) "the Origins of cryptography": the Arab contributions, Crypto logia, vol. 16, no 2.

[5] Optical society of America (2010, October 20). Long distance, top secret messages: Science Daily http://www.sciencedaily.com/releases/2010/10/101019178 03.htm

[6] Retrieved February 11 ,2012 from: http://en.wikipedia.org/wiki/cryptography

[7] Varsha B. and Chandel S. G. S. (2012) Implementation of New Advance Image Encryption Algorithm to Enhance Security of Multimedia Component. International Journal of Advanced Technology & Engineering Research (IJATER)

[8] Zeghid M., Machhout M., Khriji L., Baganne A., and Tourki R., (2007). A Modified AES Based Algorithm for Image Encryption. International Journal of Computer Science and Engineering. Vol. 1 No. 1.