Ubiquitous Devices Safety using Cyber Security

Atanu Datta[1] and Somsubhra Gupta[2*]

[1&2]*School of Computer Science, Swami Vivekananda University, Barrackpore-700121, India*

Abstract: *Mobile data security [3] refers to the exchange of wireless information and services electronically with citizens' businesses between banks using information and communication technology. This paper imposes a security mechanism of data in the banking and financial sector with the help of AES algorithm.*

During the transaction of data and information, the hackers can listen to, modify or steal information in different ways. This paper proposes to find out a suitable encryption technique to prevent attacks caused by hackers in a financial transaction.

In this paper, we have wrapped AES[1] algorithm in Object-Oriented Model[1] and for implementation of confidentially of the message to improve the citizens' confidence on online banking transaction especially on client-side application-based mobile handheld devices[4]. As the mobile phones being raise in number on a day to day life it's being proportionally affecting other hands (i.e., mobile phone security) utmost cases the important credentials are being stored in their respective mobiles.

On this point, the smartphone consists of many sensitive data which should be safeguarded. This paper tells how to protect your mobile or smartphone in a cryptographic approach.

Keywords: *Object Oriented Model, AES, Data Security, Mobile handheld devices*

1. INTRODUCTION

Financial data transfer (from Banking, E-wallet, online purchase, or in a certain context government tax payment) refers to the use of information technology to exchange information and services with citizens, businesses, and other arms of financial houses. At present, the authentication of electronic documents and their efficient packaging is the main restriction for the financial transaction application development. There are also some other pending problems like protection validity, uniqueness traceability of e-documents, prohibition of illegal duplication, and tamper-proofing.

The national E-Transaction Plan [6] of the Indian Government seeks to lay the foundation and provide for long-term growth of E- transaction within and outside the country impetuously for Indian financial organizations. Some important and financial client-side application-based data are Income Tax filling, Banking services provident found status, passport and visa information, voter Id/ National citizen card status, Trade license key or agreement, pan, Aadhar card verification. Also, attacks can change the base information according to their requirement. So to ensure speedy communication and reduce unauthorized access in information and communication technologies (ICT), it is required to use some secured Cryptographic approach while data is transferred through the internet.

The cryptographic approach is being classified into three types asymmetric cryptosystem, symmetric cryptosystem, and digital signatures here in symmetric cryptosystem the sender and receiver get the same key which on the aspect of confidentiality and avoid privacy, asymmetric cryptosystem possess different keys one for encryption and the other for decryption which mainly works for key exchange and authentication and digital fingerprint will encrypt the data irreversibly and pushes for data integrity. Encryption is the new technology that protects through disturbing the data and acts unique for their respective key sizes and even stronger for specific from another adjacent here involves many algorithms like triple DES (data encryption standards), RSA, AES (advanced encryption standard) which on classifying to modern cryptography functions DES and AES under symmetric cryptosystem. RSA under asymmetric cryptosystem.

Cryptography [1] is the science of using mathematics to encrypt and decrypt data. Cryptography enables you to store sensitive information or transmit it across insecure networks (like the internet) so that it can't be read by anyone except the intended recipient. The objective of online financial transactions from Mobile devices is to enable different wings of the online transaction to interact effectively, with each other. To implement this we use an asymmetric cryptosystem.

1.1 Advantages of AES over DES

The AES is available in three key sizes: 128, 192, and 256 bits, versus the 64 bit DES. Therefore, there are around 1021 times more AES 128-bit keys than DES 56-bit keys.

As one expert put it, assuming that you could recover a DES key in a second (trying 255 keys per second), it would take the same machine approximately 149 trillion years to recover a 128-bit AES key.

AES has more elegant mathematical formulas behind it and only requires one pass to encrypt data. AES was designed from the ground up to be fast, unbreakable, and able to support the tiniest computing devices imaginable. The big differentiators between AES and DES are not the strength of security, but superior performance and better use of resources.

The next para ie para 2 deals with the necessity of Object Orientation. Para3 discussed over AES Algorithm. Para4 shows the proposed Design. Para5 is the conclusion of this paper. Para6 shows the references of the paper. And finally, Appendix deals with the program input

2. GAP Analysis

But these digital devices are badly used also. Sometimes, students join their online classes but they are doing other activities in the background of the device at the same time without concentrating on their class, sometimes also unexpected voice or video calls terminate the online classes. Another aspect we can see is that, in the students of rural/village areas, they do not get good quality devices due to price and they are unable to maintain network cost for their online learning. So, research on digital learning is widely circulated, but research to enhance traditional devices learning using newer types of ubiquitous and pervasive devices as an example of "New-normal Object-Oriented Dynamic Learning Environment (NOODLE)" for collecting resources is yet to be widely circulated in the education.

3. Methodological Aspects

3.1 Need of Object orientation

Simplicity: software objects model real-world objects, so the complexity is reduced and the program structure is very clear. Here the key is an object by which the program creates the secret key.

Modularity: Each object forms a separate entity whose internal workings are decoupled from other parts of the system;

Modifiability: it is easy to make minor changes in the data representation or the procedures in an OO program. Changes inside a class do not affect any other part of a program, since the only public interface that the external world has to a class is through the use of methods.

Here modification requires every day, because financial transaction policy by banks or the government may change from time to time. This is shown in figure 1.

Extensibility: adding new features or responding to changing operating environments can be solved by introducing a few new objects and modifying some existing ones. For the betterment of civilization, the government introduces new policies every day.

Maintainability: objects can be maintained separately, making locating and fixing problems easier;

Re-usability: objects can be reused in different programs. Cryptography [1,5] is the science of using mathematics to encrypt and decrypt data. Cryptography enables you to store sensitive information or transmit it across insecure networks (like the Internet) so that it cannot be read by anyone except the intended recipient. While cryptography is the science of securing data, cryptanalysis is the science of analyzing and breaking secure communication. Classical cryptanalysis involves an interesting combination of analytical reasoning, application of mathematical tools, pattern finding, patience, determination, and luck.

Cryptology embraces both cryptography and cryptanalysis.

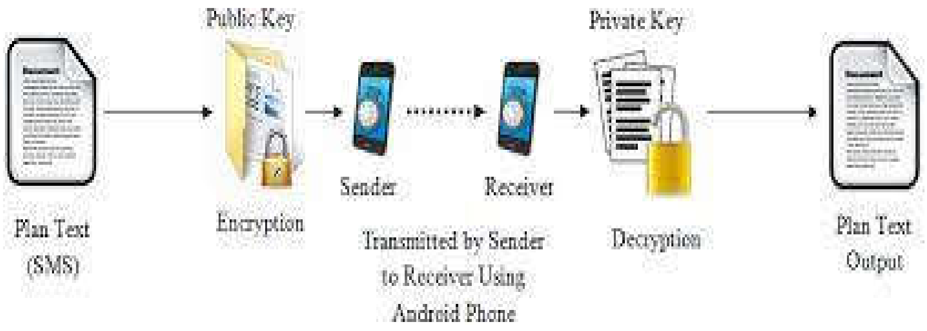

Figure 1. Block Diagram

The Techniques are presented in the following Section 4.

4. Algorithms and Techniques

4.1 Outline of AES Algorithm

AES is a block cipher with a block length of 128 bits.

AES allows for three different key lengths: 128, 192, or 256 bits. Our discussion will assume that the key length is 128 bits.

Encryption consists of 10 rounds of processing for 128-bit keys, 12 rounds for 192-bit keys, and 14 rounds for 256-bit keys.

Except for the last round in each case, all other rounds are identical.

Each round of processing includes one single-byte-based substitution step, a row-wise permutation step, a column-wise mixing step, and the addition of the round key. The order in which these four steps are executed is different for encryption and decryption.

To appreciate the processing steps used in a single round, it is 2 best to think of a 128-bit block as consisting of a 4×4 matrix of bytes, arranged as follows:

byte$_0$ byte$_4$ byte$_8$ byte$_{12}$
byte$_1$ byte$_5$ byte$_9$ byte$_{13}$
byte$_2$ byte$_6$ byte$_{10}$ byte$_{14}$
byte$_3$ byte$_7$ byte$_{11}$ byte$_{15}$

Therefore, the first four bytes of a 128-bit input block occupy the first column in the 4 × 4 matrices of bytes. The next four bytes occupy the second column, and so on.

The 4 × 4 matrices of bytes are referred to as the state array.

A word consists of four bytes that are 32 bits. Therefore, each column of the state array is a word, as is each row.

Each round of processing works on the input state array and produces an output state array.

The output state array produced by the last round is rearranged into a 128-bit output block. The Encryption Key and Its Expansion.

AES Decryption process is just the reverse procedure of the AES Encryption process. With the help of the Decryption process, encrypted data convert to the plain test ie original data.

4.2 The Encryption Key and Its Expansion

Assuming a 128-bit key, the key is also arranged in the form of a matrix of 4 × 4 bytes. As with the input block, the first word from the key fills the first column of the matrix, and so on.

The four-column words of the key matrix are expanded into a schedule of 44 words. (As to how exactly this is done, we will explain that later.) Each round consumes four words from the key schedule.

The figure below depicts the arrangement of the encryption key in the form of 4-byte words and the expansion of the key into a key schedule consisting of 44 4-byte words.

The Overall Structure of AES

The overall structure of AES encryption/ decryption is shown in Figure 1.

The number of rounds shown in Figure 2, is for the case when the encryption key is 128 bits long. (As mentioned earlier, the number of rounds is 12 when the key is 192 bits, and 14 when the key is 256.)

Before any round-based processing for encryption can begin, the input state array is XORed with the first four words of the key schedule. The same thing happens during decryption — except that now we XOR the ciphertext state array with the last four words of the key schedule.

For encryption, each round consists of the following four steps: 1) Substitute bytes, 2) Shift rows, 3) Mix columns, and 4) Add round key. The last step consists of XORing the output of the previous three steps with four words from the key schedule.

For decryption, each round consists of the following four steps: 1)Inverse shift rows, 2) Inverse substitute bytes, 3) Add round key, and 4) Inverse mix columns. The third step consists of XORing the output of the previous two steps with four words from the key schedule.

The last round for encryption does not involve the "Mix columns" step. The last round for decryption does not involve the "Inverse mix columns" step.

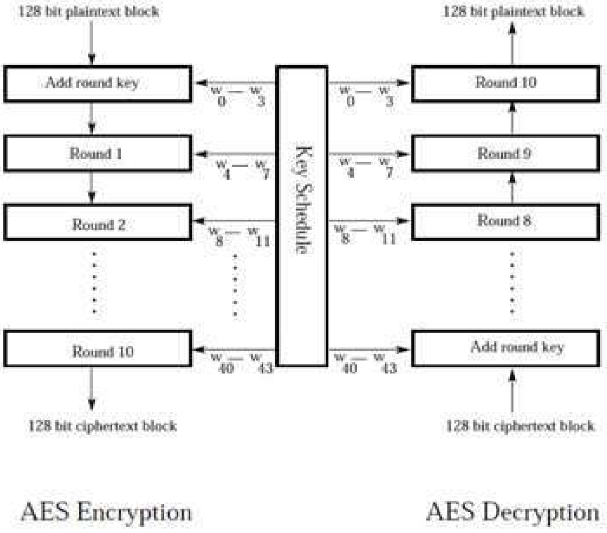

Figure 2. Rounds of AES

4.3 The Four Steps in Each Round of Processing

STEP 1: Generate Sub Bytes for byte-by-byte substitution during the forward process.
STEP 2: Shift Rows for shifting the rows of the state array during the forward process. The corresponding transformation during decryption is denoted Inv Shift Rows for Inverse Shift-Row Transformation.
STEP 3: Mix Columns for mixing up of the bytes in each column separately during the forward process.
STEP 4: Add Round Key for adding the round key to the output of the previous step during the forward process.

The Substitute Bytes Step

This is a byte-by-byte substitution and the substitution byte for each input byte is found by using the same lookup table.

- The size of the lookup table is 16 × 16.

The Shift Rows Step

- This is where the matrix representation of the state array be- comes important.
- The ShiftRows transformation consists of (i) not shifting the first row of the state array at all; (ii) circularly shifting the second row by one byte to the left; (iii) circularly shifting the third row by two bytes to the left; and (iv) circularly shifting the last row by three bytes to the left.

The Mix Columns Step

- This step replaces each byte of a column by a function of all the bytes in the same column.
- More precisely, each byte in a column is replaced by two times the value of that byte, plus three times the next byte, plus the byte that comes next, plus the byte that comes next. The word 'next' means the byte in the row below; the meaning of 'next' is circular in the same column.

Adding the Round Key

- The 128 bits of the state array are bitwise XOR'ed with the 128 bits of the round key.
- The AES Key Expansion algorithm is used to derive the 128-bit round key from the original 128-bit encryption key.

5. Conclusion

In this paper, we have concentrated on the modeling of the secret key cryptographic algorithm AES in the Object-Oriented Programming paradigm. The Proposed Object-Oriented model designed in this paper uses the data hiding feature, the secret key. The reuse of code is made possible by implementing the inheritance feature of objects. Generally, all secret key cryptographic algorithms use one secret key in the sender end as well as the receiver end. This secret key can be easily defined in OOP. In our proposed object-oriented model of AES algorithm, different objects of sender class are defined where the secret key of sender class object cannot be known to any third party objects but should be known to any objects of receiver class. Another advantage of this object-oriented model is that the roles of sender and receiver classes may be interchanged simply by declaring the objects interchangeably in the Driver Program main ( ).

OOP offers several benefits to both the program designer and the user. Object-Orientation contributes to the solution of many problems associated with the development and quality of software products. The new object-oriented technology promises greater programmer productivity, a better quality of software, and lesser maintenance cost.

Through inheritance, we can eliminate redundant code and extend the use of existing classes, Sender and Receiver. We can build programs from the standard working modules. This leads to saving of development time and higher quality of productivity. Object-Oriented systems can be easily upgraded from small to large systems. For example in our present object-oriented model of AES, we can add another secret key in the sender class in addition to the existing one that communicates with one another, rather than having to start writing the code from scratch.

ACKNOWLEDGEMENT

The authors gratefully acknowledge the Authority of Swami Vivekananda University, Barrackpore for their kind help and support during the course of research

REFERENCES

[1] Bechrouz A. Forouzan, Cryptography and Network Security Special Indian Edition, pp.-191-224

[2] Dr. Sunil Karforma, Dr. Sripati Mukhopadhyay, Siddartha Sen, An object oriented approach of Elgamal Digital Signature Algorithm , EAIT 2006 Kolkata. pp.-259,260

[3] Liza M. Lowery, Executive Director Department of Telecommunications & Information Services, San Francisco, Developing a Successful E-Government Strategy

[4] Alan G. Konheim, Computer Security and Cryptography , Wiley-Interscience A John Wiley & sons, Inc. Publication

[5] www.exforsys.com/tutorials

[6] www.vocal.com/cryptography/aes.html

[7] Russell, D. and Gangemi, Computer Security Basics, G.T, O' Reilly & Associate ,wikipedia.org/wiki/ Advanced_EncryptionSt

[8] *C Kaufman, R Perlman, M Speciner, Network Security - PHI, Second Edition, 2002*

[9] Ankit Fadia, Network Security Macmillan Publication

[10] Rajib Mall, Fundamental of Software Engineering, Prentice-Hall of India Pvt. Ltd 2004