I.J. Mathematical Sciences and Computing, **2018, 2, 34-48**
Published Online April 2018 in MECS (http://www.mecs-press.net)
DOI: 10.5815/ijmsc.2018.02.04

Available online at ***http://www.mecs-press.net/ijmsc***

Development of a Secure SMS Application using Advanced Encryption Standard (AES) on Android Platform

Muhammad Noman Riaz [a,] Adeel Ikram [b]

[A] *Assistant Professor, Department of Avionics Engineering, College of Aeronautical Engineering, National University of Sciences & Technology, Risalpur, 24090*
[B] *Undergraduate Student, Department of Avionics Engineering, College of Aeronautical Engineering, National University of Sciences & Technology, Risalpur, 24090*

Received: 31 December 2017; Accepted: 13 February 2018; Published: 08 April 2018

Abstract

When we live in a global village, then maintaining privacy and confidentiality becomes reasonably challenging. Short Message Service (SMS) is the oldest application for exchanging messages between communicating parties in cellular network used by mobile phones. These messages are encrypted over-the-air with A5/1 algorithm and stored as clear text at network operator. Recent developments have shown that this algorithm is not secure any more. Compromising an access to network operator registers gains access to SMS also. Current scenarios of hacks and exploitation demands confidentiality, and encryption is one of the techniques, which is used, in this subsequent project of designing a secure SMS android application. Cryptographic manipulation of the data is performed using AES 128 -bit algorithm to secure the data, which is essential to us and the safe transmission of confidential data over the GSM network. AES (Advanced Encryption Standards) algorithm is the considered impregnable even to super computers brute force attacks. The AES algorithm technique uses very befuddled and sporadic encryption making data impregnable to attackers or hackers. This android app will allow the user to encrypt and decrypt the SMS (Short Message Service) efficiently and just at one click. Subsequent explanation is given afterwards.

Index Terms: Encryption, Privacy, SMS, AES, Block Cipher, Mobile Application and Android.

1. Introduction

Communication has always been a crucial factor in development of human's everyday life. Since prehistoric times human society developed different forms of communication which made possible exchanging meaningful information between individuals, and, as a result, they put in function the society itself. At those times fire, smoke signals, horns etc. were used as communication tools and techniques. Moreover, the appearance of speech made a revolution in human communication. In addition, symbols and writing further revolutionized communication. They led to new communication techniques like mail, pigeon post etc. The technology innovations further improved and made communication more powerful. First the telegraph then the telephone made communication quite simple even in long distances. Furthermore, the internet extremely boosted the communication and made it easier than ever before. Telephone is considered to be one of the most important invention that revolutionized communication. First telephone was introduced with the landline version where all telephone were connected by wires but, later it evolved to the wireless version.

SMS is a text messaging service component of phone, web, or mobile communication systems, using standardized communications protocols that allow the exchange of short text messages between fixed line or mobile phone devices. Users can used SMS to send or receive from a single person, or several persons, personal messages, email notifications, information services [1], school activity alerts, notification from teacher, job dispatches, and also stock alerts. With these usable application, SMS is now more and more common among mobile phone users. However the security issue [12] of SMS's is still an open challenging task.

SMS is now a very common communication tool. The security protection of SMS messages is not yet that sophisticated and difficult to implement in practice. The confidentiality and integrity mechanisms are only specified as optional security measures that can be made available, but they are not mandatory requirements for SMS system implementation [14]. In this paper, there proposed the use of symmetric cryptography for SMS transfer securing.

2. Literature Review

Smartphones' market is growing exponentially as well as the operating systems like android and iOS, which run them. Regular updates are delivered to the users that provide services and security and keep them up to date in case of any new threat or breakthrough. Now communication is a big area itself and is of extreme importance considering military, governmental and political communication and also the daily personal information one don't want to share with others; there comes the secure methods of transferring information like encryption so that no unwanted listener hear or read what you have to say or write. There are multiple encryption schemes and algorithms available all around like blowfish, RC4, but being considered the most secure and efficient is AES (Advanced Encryption Standard) and is widely used for military grade encryption and other secure communication purposes. There are three variants available of the AES i.e. AES-128, AES-192, AES-256. The number represent the bits and the higher the bits more befuddling and mingling of data happens. For the lightweight apps like SMS encryption AES 128 is quite sufficient to handle and it do provide the fool proof security.

Recent trends in enterprise mobility have made mobile device security an imperative. IDC reported in 2010 that for the first time smartphone sales outpaced PC sales. Faced by this onslaught of devices and recognizing the productivity and cost benefits, organizations are increasingly implementing bring-your-own device (BYOD) policies. Research firm J. Gold Associates reports that about 25%-35% of enterprises currently have a BYOD policy, and they expect that to grow to over 50% over the next two years. This makes sense as mobility evolves from a nice-to-have capability to a business advantage.

But the competitive edge and other benefits of mobility can be lost if smartphones and tablet PCs are not adequately protected against mobile device security threats. While the market shows no sign of slowing, IT organizations identify security as one of their greatest concerns about extending mobility. Therefore, various

encryption techniques are used. [2]. Encryption has long been used by militaries and governments to facilitate secret communication. Encryption is now commonly used in protecting information within many kinds of civilian systems. For example, the Computer Security Institute reported that in 2007, 71% of companies surveyed utilized encryption for some of their data in transit, and 53% utilized encryption for some of their data in storage [3] Encryption can be used to protect data "at rest", such as files on computers and storage devices (e.g. USB flash drives). In recent years there have been numerous reports of confidential data such as customers' personal records being exposed through loss or theft of laptops or backup drives. Encrypting such files at rest helps protect them should physical security measures fail. [2] Digital rights management systems which prevent unauthorized use or reproduction of copyrighted material and protect software against reverse engineering (see also copy protection) are another somewhat different example of using encryption on data at rest.

In 2010, 6.1 trillion SMS text messages were sent. This translates into 192,192 SMS per second. SMS has become a massive commercial industry, worth over $81 billion globally as of 2006. The global average price for an SMS message is $0.11, while mobile networks charge each other interconnect fees of at least $0.04 when connecting between different phone networks.

The SMS industry being on such a great rise is vulnerable to attacks. Therefore it has now become more imperative to encrypt SMS before sending.[3] Various algorithms for encryption and decryption are in place. Out of the entire group of algorithm AES is the most preferred one. AES require very low RAM space and its very fast. On Pentium Pro processors AES encryption requires only 18 clock cycles/byte equivalent to throughput of about 11Mib/s for 200MHz processor. This was the main reason why we decided to use AES algorithm for encryption and decryption. [6].

The field of cryptography [15] can be divided into several techniques of study. There are two types of techniques in cryptography which are asymmetric key algorithm and symmetric key algorithm. Asymmetric key algorithm or sometimes called public key algorithm is usually based on complex mathematical problems. Symmetric key algorithm can be broadly grouped into block ciphers and stream ciphers [16]. Other symmetric key algorithms are cryptographic hash functions and Message Authentication Codes (MACs)

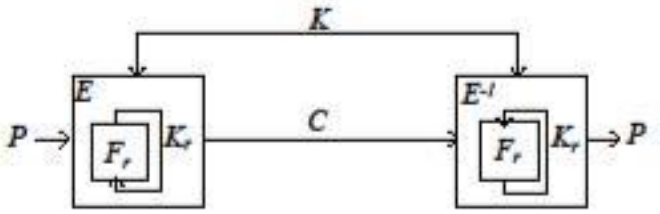

Fig.1. Diagram of Symmetric Block Cipher

There are few SMS application on Google Play which encrypts the SMS using AES algorithm. We have programmed our application meticulously considering various factors which might benefit the user. With only less than 200Kb size it is feather weight which effectively makes it faster. It provides functionality like conversation view, Inbox, Draft, Backup and restore; all the functionality which a standard SMS application should provide. The main advantage is that it is very simple app, easy to understand and very easy to operate. User interface is so simple and light weight that main functionality of encryption and decryption of SMSis carried out very efficiently.

The symmetric key block cipher technique operates on the same block or fixed-length groups of bits. The algorithm is illustrated in Fig. 1. The encryption function in (1), denoted as E, is a process of enciphering information called Plaintext, denoted as P, using some secret codes called secret Key, denoted as K, into an unreadable form called Ciphertext, denoted as C. The P, as it goes through each round of the cipher, is referred to as the cipher-state, denoted as F.

In the literature as shown in Tab. 1, many authors have used different cryptography algorithms in the SMS encryption application to provide confidentiality in sending and receiving messages. Even though there are several authors (2 out of 8) used DES, 3DES and AES block cipher algorithms in their works but most of these works are asymmetric key encryption techniques. Therefore it is advisable that can develop SMS Encryption using symmetric key encryption.

Table 1. Review on SMS Encryption

Author	Algorithm
Lisonek & Drahansky [17]	RSA
Albuja &Carrera [18]	DES, 3DES, AES and RSA
Toolani & Shirazi [19]	ECDLP
Zhao et al [20]	identity-based
Harb et al [2]	3DES
Sonam [3]	Elleptic Curve
Hosain et al [4]	SMS Sec

Owning from suggestion of Garza-Saldana & Diaz-Perez [6] that symmetric encryption could provide confidentiality to SMS, this paper perform an evaluation of three block cipher symmetric encryption techniques. This is done in order to find the most suitable block cipher symmetric encryption technique for securing SMS transmitted messages.

3. Short Message Service (SMS)

SMS stands for **short message service**. Simply put, it is a method of communication that sends text between cell phones, or from a PC or handheld to a cell phone. The "short" part refers to the maximum size of the text messages: 160 characters (letters, numbers or symbols in the Latin alphabet). For other alphabets, such as Chinese, the maximum SMS size is 70 characters.

Working of SMS

It is well-known that SMS service is a cell phone feature but indeed, SMS can also work on other computing devices such as PC, Laptop, or Tablet PC as long as they can accept SIM Card. SIM Card is needed because SMS service needs SMS center client which is built-in on the SIM Card.

BTS

A base transceiver station (BTS) is a piece of equipment that facilitates wireless communication between user equipment (UE) and a network. UEs are devices like mobile phones (handsets), WLL phones, computers with wireless internet connectivity, WiFi and WiMAX devices and others.

MSC

The mobile switching center (MSC) is the primary service delivery node for GSM/CDMA, responsible for routing voice calls and SMS as well as other services (such as conference calls, FAX and circuit switched data).[2] The MSC sets up and releases the end-to-end connection, handles mobility and hand-over requirements during the call and takes care of charging and real time pre-paid account monitoring.

SMSC

When SMS is transmitted from a cell phone, the message will be received by mobile carrier's SMS Centre (SMSC), do destination finding, and then send it to destination devices (cell phone). SMSC is SMS service centre which is installed on mobile carrier core networks. Beside as SMS forwarding, SMSC also acts as temporary storage for SMS messages. So, if the destination cell phone is not active, SMS will store the message and then deliver it after the destination cell phone is active. As additional, SMSC also notify the sender whether the SMS delivering is success or not. However SMSC cannot store the SMS message forever since the storage capacity is not unlimited. During the SMS delivering, sender cell phone and SMSC is actively communicating. So, if the non-active destination cell phones become active, SMSC directly notifies the sender cell phone and tell that the SMS delivering is success.

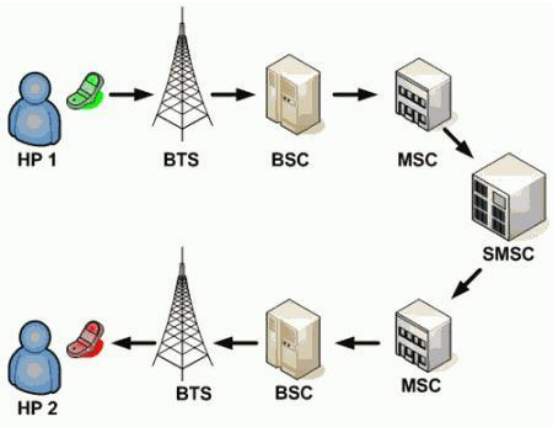

Fig.2. Transmission of SMS[13]

4. Advance Encryption Standards Algorithm/ Rijndael Algorithm

The Advanced Encryption Standard[9] comprises three block ciphers, AES-128, AES-192 and AES-256. AES has a fixed block size of 128 bits and a key size of 128, 192, or 256 bits. The block-size has a maximum of 256 bits, but the key-size has no theoretical maximum. The cipher uses number of encryption rounds which converts plain text to cipher text. The output of each round is the input to the next round. The output of the final round is the encrypted plain text known as cipher text. The input given by the user is entered in a matrix known as State Matrix. [2]

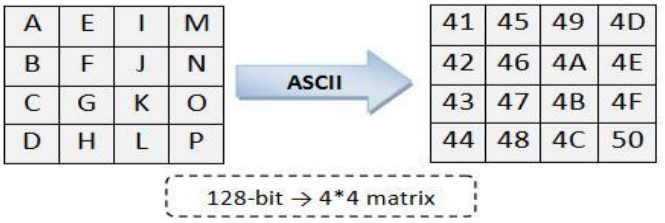

Fig.3. State Matrix

Following are the four steps.

SubBytes Step

This step is same as SubBytes step of AES algorithm. In the S-Box Substitution step, each byte in the matrix is reorganized using an 8-bit substitution box. This substitution box is called 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 (28), known to have good non-linearity properties[19]. 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. [7] This step causes confusion of data in the matrix. S-Box Substitution is carried out separately for LPT and RPT. This is the first step of iterative round transformation. The output of this round is given to the next round. [3]

ShiftRows Step

The ShiftRows step is performed on the rows of the state matrix. It cyclically shifts the bytes in each row by a certain offset. The first row remains unchanged. Each byte of the second row is shifted one position to the left. Similarly, the third and fourth rows are shifted by two positions and three positions respectively. The shifting pattern for block of size 128 bits and 192 bits is same[3].

MixColumns Step

In the MixColumns step, the four bytes of each column of the state matrix are combined using an invertible linear transformation [5]. A randomly generated polynomial is arranged in a 4*4 matrix. The same polynomial is used during decryption. Each column of the state matrix is XOR-ed with the corresponding column of the polynomial matrix. The result is updated in the same column. The output matrix is the input to AddRoundKey.[3]

AddRoundKey

A round key is generated by performing various operations on the cipher key. This round key is XOR-ed with each byte of the state matrix. For every round a new round key is generated using Rijndael‟s key scheduling algorithm. [3]

Decryption of the Proposed Algorithm

The encryption algorithm is referred to as the cipher and the decryption algorithm as the inverse cipher. In addition, the cipher and the inverse cipher operations must be executed in such a way that they cancel each other. The rounds keys must also be used in reverse order. [4] The Cipher Text which is formed of 256-bit 4*8 Matrix is the input for the decryption process. [8]

5. Implementation & Pseudo Code of Application

The algorithm can be implemented in any language. This algorithm can also be used in Image Processing. We have implemented it in java, java being an open source and platform independent language. The pseudo codes for the components of the cipher are given below. [3]

SMS SENDING

```
//      Getting an SmsManager

    SmsManager smsManager =
```

```
SmsManager.getDefault();

//    In case the Message exceeds 160 characters

//    Divide message

ArrayList<String> parts = smsManager.divideMessage(variable);
smsManager.sendMultipartTextMessage
```

ENCRYPTING FUNCTION

```
byte[] returnArray;

// Key generation from user input

Key key = generateKey(variable); // specify AES

Cipher c = Cipher.getInstance("AES");
// specify encryption mode c.init(Cipher.ENCRYPT_MODE, key);
```

Also adding a number of exceptions and limitations for invalid or incomplete secret key

```
// key generation from string

Key key = new variable (variable.getBytes(), "AES"); return key;
```

SMS RECIEVING

```
//    Getting the intent extra

Bundle variable = intent.getExtras()

//    Specify bundle to get "pdus" to retrieve message Object[] variable = (Object[]) bundle.get("pdus");

SmsMessage sms[] = new

SmsMessage[variable.length];

// Getting received SMS

variable = sms[i].getDisplayMessageBody(); //Getting sender's phone number

variable = sms[i].getDisplayOriginatingAddress();
```

SMS DISPLAY

```
// Getting phone number
variable = extras.getString("variable "); // Getting encrypted message
        variable = extras.getString("variable ");
        // Setting UI variable.setText(variable);
```

6. Working of Software Application

Fig.4. Sender's View

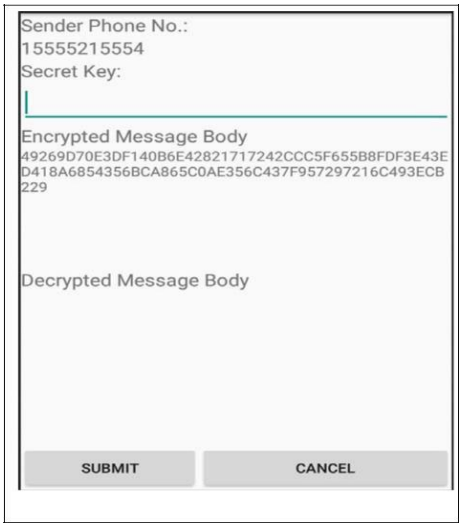

Fig.5. Receivers View

7. Modelling of Application

The Unified Modelling Language (UML) is used to approach the development of the app because of its general- purpose modelling language in the area of software engineering, and it provides a great view of the system and its architecture. The Class Diagram and Use Case are as follows [3].

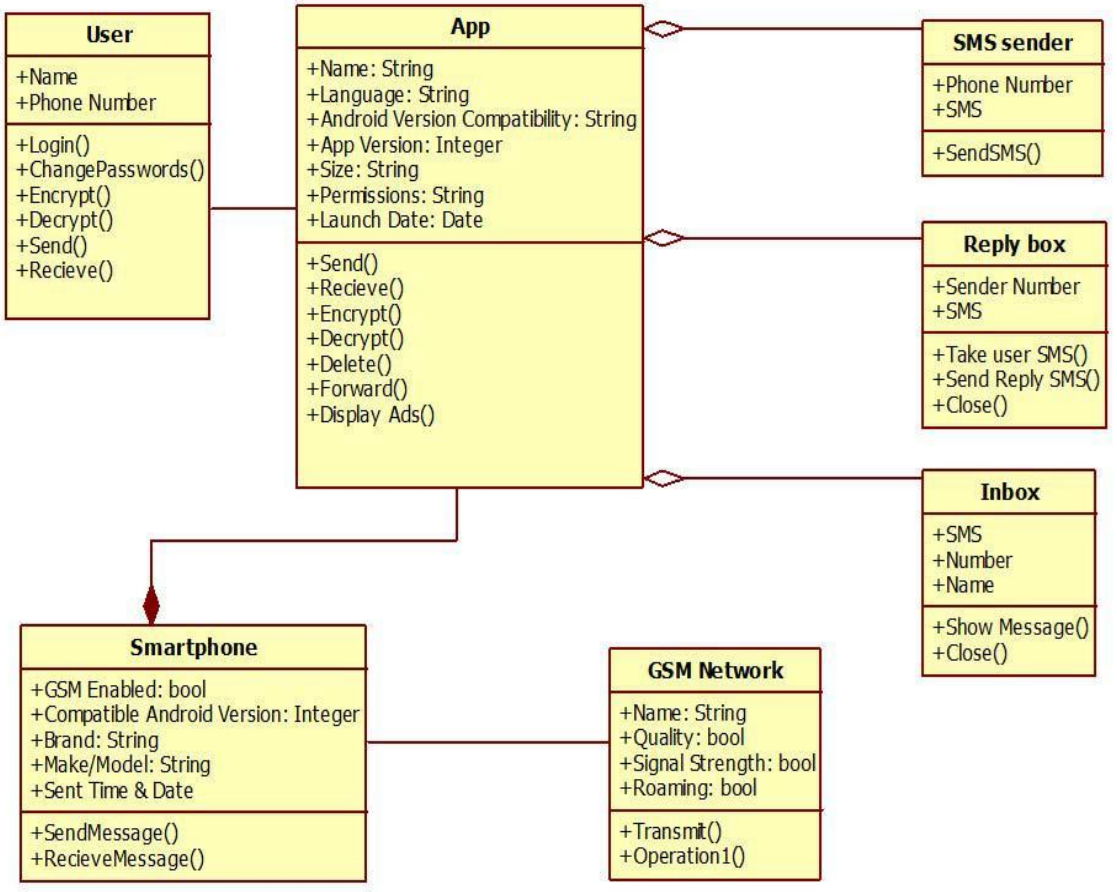

Fig.6. Class Diagram

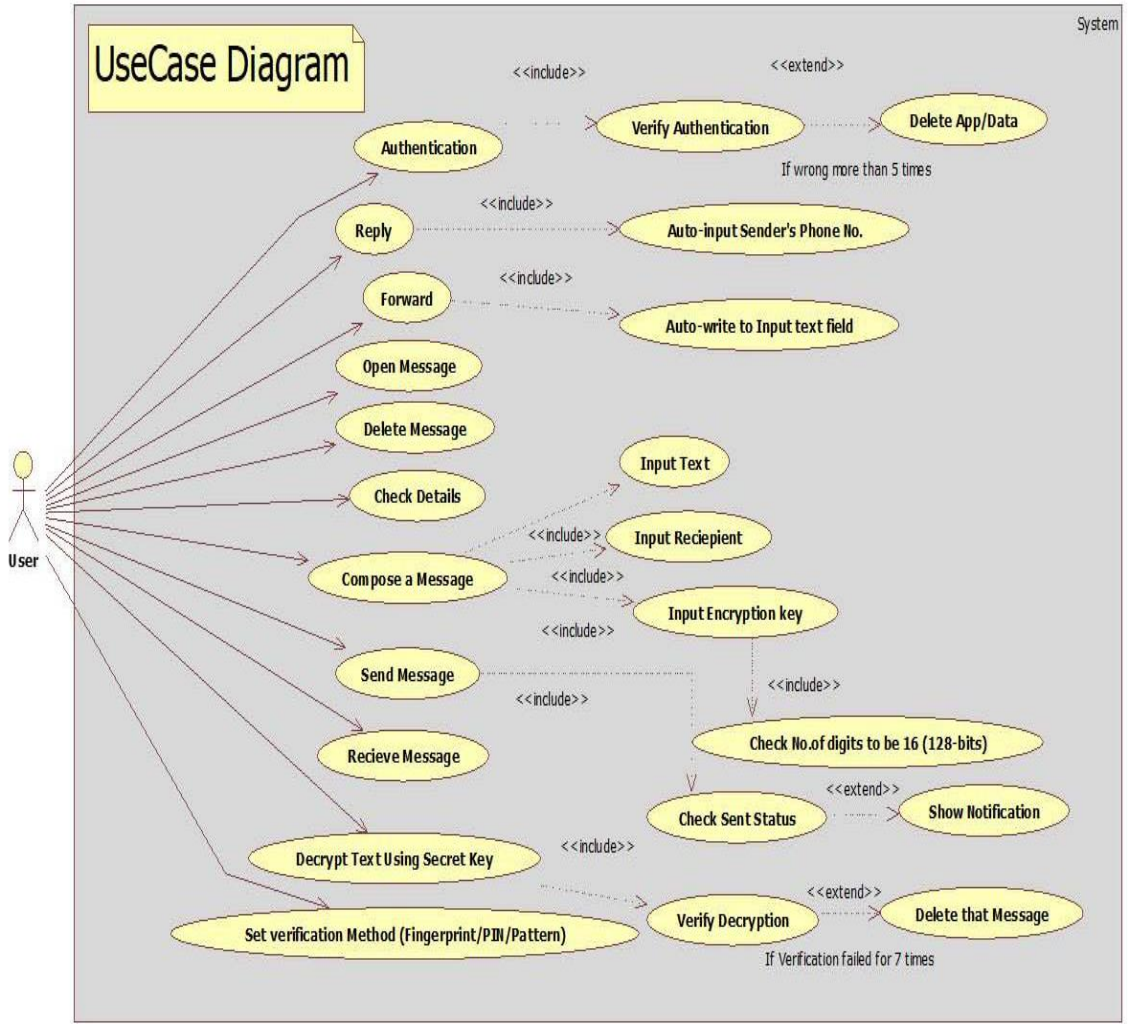

Fig.7. Use Case Diagram

8. Design of SMS Encryption

Programming Platforms for Mobile Phones

The SMS Encryption was developed for evaluating two symmetric encryption techniques which is AES and 3D-AES. It has been developed using a Java Programming Language, Java Micro Edition (Java ME) which is produced by Sun Microsystems[15]. Almost all mobile phones include this programming platform. The Eclipse IDE is the essential starting point for Mobile developers, including a Java IDE, C language support, a Git client, XML Editor and Mylyn.

Design flow

In this SMS Encryption is used a standardized facility defined as part as of the Global System for Mobile

Communications (GSM) series of standards [13] as shown in Fig. 3. Any message, sent via SMS, is not directly delivered to its destination, but it is stored into an SMS Center (SMSC) after passing through a Mobile Switching Center (MSC), which has the important role of message routing, according to the information provided by Home Location Register (HLR) and the Visitor Location Register (VLR).

The SMS Encryption application works only with SMS, which is encrypted in the first step, digitally signed in the second step and sent in the last step.

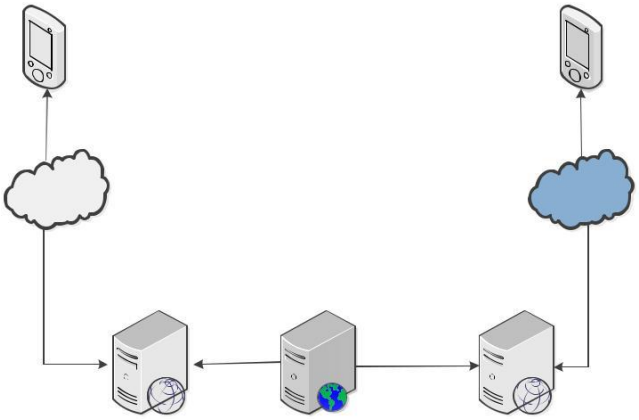

Fig.8. SMS Architecture[2]

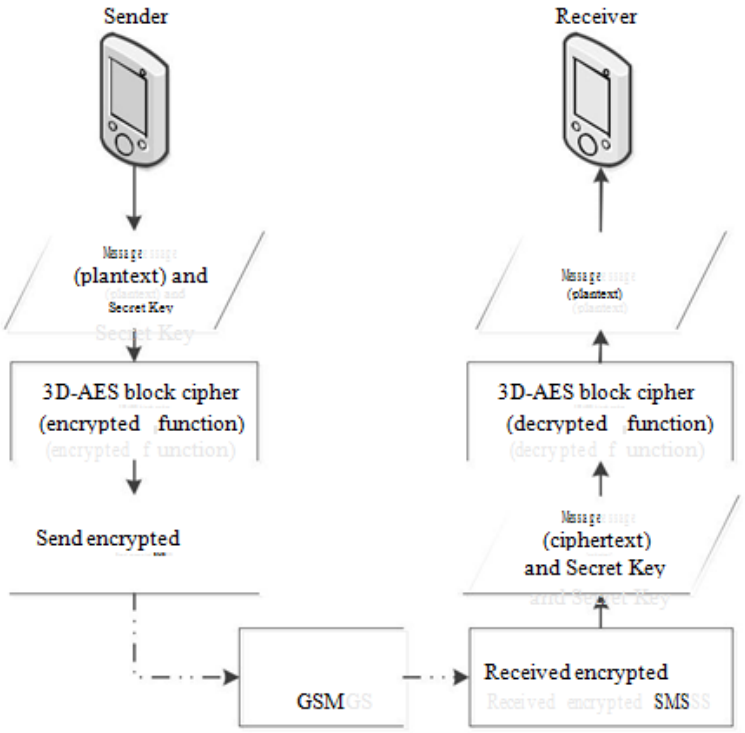

Fig.9. SMS Encryption[6]

9. Results & Discussion

This application was tested on Android operating system, v4.1.2 (Jelly Bean), Cortex-A5 processor mobile phone running at 1 GHz speed, with 4 GB internal Memory and 786 MB RAM. The performance data were collected by applying 100 sequences of random SMS message or plaintext for each sizes on the phone to get the encryption and decryption time for both algorithms[17]. The AES block cipher has a fixed block length of 128 bits and a key length of 128, 192, or 256 bits. It can be specified with block and key sizes in any multiple 35 of 32 bits with a minimum of 128 bits. The AES block cipher has 10 rounds for 128-bit keys, 12 rounds for 192-bit keys, and 14 rounds for 256-bit keys. The experiment only taking into consideration on 128-bit-keys only as well as 3D-AES blocks cipher.

Encryption time is the time taken to transform the SMS plaintext into cipher text. For each key size of same algorithm, random SMS message of different bit sizes was encrypted. The average of the encryption time is calculated using the formula in (1) and the results are tabulated in Tab. 2. where n is total number of encrypted message sequences, e_i is the consecutive encryption time and E_t is the average encryption time[10].

$$E_t = \frac{1}{n} \sum_{i=1}^{n=100} e_i \tag{1}$$

Table 2. Encryption Time in Milliseconds

Plaintext size	AES	3D-AES
32	45	243
64	73	243
128	145	243
256	298	243
512	412	243
1024	872	486

Decryption time is the time taken to transform the SMS cipherext into plaintext[11]. For each key size of same algorithm, random SMS message of different bit sizes was decrypted. The average of the decryption time is calculated using the formula in (2) and the results are tabulated in Tab. 3.

$$C_t = \frac{1}{n} \sum_{i=1}^{n=100} c_i \tag{2}$$

where n is total number of decrypted message sequences, c_i is the consecutive decryption time and C_t is the average decryption time.

Table 3. Decryption Time in Milliseconds

Ciphertext size	AES	3D-AES
32	47	241
64	71	241
128	142	241
256	287	241
512	409	241
1024	821	483

Tab. 3 indicates that decryption time and the ciphertext size are related. The rise in plaintext size of the AES block cipher increases the decryption time. Even though the 3D-AES block cipher has a high decryption time when the ciphertext size between 32 bit to 128 bits compared to the AES block cipher, the 3D-AES has low decryption time when plaintext size more then 256 bits. It can be indicate that SMS decryption using the AES block cipher will be proposed till 128 bits. Since the 3D-AES and AES have use a same key size to achieve high security, it can be concluded that the AES block cipher is the most cost effective algorithm for SMS encryption as compared with the 3D-AES block cipher.

10. Software Application Testing

For the sake of software testing both black box testing methodology as well as white box testing is implemented because there was a need to carry out tests to both sides of the application, the internal and the functional one. Black box testing includes wrong input of key, incomplete key and empty message scenario testing. The application does respond to these tests efficiently. White box testing helped to improve the code and removal of errors and bugs.

11. SMS Application

The application works in following way:

- The user opens the application and authenticates using pattern lock.
- User can either type new message or reply to an existing message.
- If new message is selected, user enters the message and presses encrypt button after inserting the recipient‟s name. The user has to enter a cipher key before the message is sent. The cipher key is auto-generated if the user does not enter one.
- If the user selects to reply to an existing message, he first decrypts the message by long pressing the message and then types in the reply. The user is asked to enter cipher key before the message is sent.
- Once the cipher key is entered, the message is successfully sent and is shown in encrypted form in the thread.

12. Conclusion & Future Work

The application of SMS Encryption of AES block cipher on android application has been designed and implemented. The application is running in the mobile phone and does not require any additional encryption devices. The result showed that suitable and easy to implement in mobile device for the proposed scheme. With the increasing use of SMS for communication and information exchange, care should be taken when sensitive information is transmitted using SMS. Users should be aware that SMS messages might be subject to

interception. Solutions such as encrypted SMS should be considered if there is a need to send sensitive information via SMS.

References

[1] 1S. Doyle, "Using short message service as a marketing tool", Journal of Database Marketing, vol. 8, no 3, 2001, pp. 273-277.

[2] H. Harb, H. Farahat, M. Ezz, "SecureSMSPay: secure SMS mobile payment model", 2nd International Conference on Anti-counterfeiting, Security and Identification, ASID. Guiyang, China, 2008, pp. 11- 17.

[3] R. Soram, "Mobile sms banking security using elliptic curve cryptosystem", International Journal of Computer Science and Network Security, vol. 9, no. 6, pp. 30-38.

[4] M. A. Hossain, S. Jahan, M. M. Hussain, M.R. Amin, and S.H. S Newaz, "A proposal for enhancing the security system of short message services in GSM", 2nd International Conference on Anti-counterfeiting, Security and Identification, ASID, Guiyang, China, 2008, pp. 235- 240.

[5] P. H. Kuaté, J. L. Lo and J. Bishop, "Secure asynchronous communication for mobile devices", Proceedings of the Warm Up Workshop for ACM/IEEE ICSE 2010,Cape Town, South Africa, 2009, pp. 5 – 8.

[6] J. J. Garza-Saldana and A. Diaz-Perez, "State of security for SMS on mobile devices", Proceedings of the Electronics, Robotics and Automotive Mechanics Conference, 2008, pp. 110 – 115.

[7] S. Ariffin, R. Mahmod, A. Jaafar and M.R.K. Ariffin, "Byte Permutations in Block Cipher Based on Immune Systems", International Conference on Software Technology and Engineering, 3rd (ICSTE 2011). ASME Press, New York, NY., 2011.

[8] NIST, "Fips197: Advanced Encryption Standard (AES)", FIPS PUB 197 Federal Information Processing Standard Publication 197, Technical report, National Institute of Standards and Technology, 2001.

[9] J. Daemen, V. Rijmen, V., "The Design of Rijndael, AES - The Advanced Encryption Standard", Springer-Verlag, 2002.

[10] S. Ariffin, R. Mahmod, A. Jaafar and M.R.K. Ariffin, "An immune system-inspired byte permutation function to improve confusion performance of round transformation in symmetric encryption scheme", Computer Science and Applications, Lecture Notes in Electrical Engineering, Springer, 2012.

[11] S. Ariffin, R. Mahmod, A. Jaafar and M.R.K. Ariffin, "Symmetric Encryption Algorithm Inspired by Randomness and Non-linearity of Immune Systems", International Journal of Natural Computing Research, IGI Global Publishing, 2012.

[12] D. Lisonek and M. Drahansky, "SMS encryption for mobile communication", International Conference on Security Technology, Hainan Island, 2008, pp 198 – 201.

[13] S. Redl, M. W. Oliphant, M. K. Weber, and M. K. Weber, "An Introduction to GSM", 1st ed. Norwood, MA, USA: Artech House, Inc., 1995.

[14] "Short Message Service Security on Febuary 2008", available http://www.infosec.gov.hk/english/technical/files/short.pdf dated on August 2013.

[15] B. Schneier, "Applied Cryptography: Protocols, Algorithms, and Source Code in C", John Wiley & Sons, Inc., New York, NY, USA, 2nd edition, 1995.

[16] W. Stallings, "Cryptography and network security", Prentice Hall, New Jersey, United State, 2006.

[17] D. Lisonek and M. Drahansky, "SMS encryption for mobile communication", International Conference on Security Technology, Hainan Island, 2008, pp 198 – 201.

[18] J. P. Albuja and E. V. Carrera, "Trusted SMS communication on mobile devices", 11th Brazilian Workshop on Real-Time and Embedded Systems, Pernambuco, Brazil, 2009, pp.165- 170.

[19] M. Toorani and A.A.B. Shirazi, "SSMS-A secure SMS messaging protocol for the m-payment systems", Proceedings of the 13th IEEE Symposium on Computers and Communications (ISCC'08), Marrakech,

2008, pp. 700-705.
[20] S. Zhao, A. Aggarwal and S. Liu, "Building secure user-touser messaging in mobile telecommunication networks", Proceedings of Wireless Telecommunications Symposium, Pomona, CA, 2008, pp.151-157.

Authors' Profiles

M Noman Riaz was born in September, 1982 at Karachi. He completed his B.S. in Electronic Engineering from Sir Syed University of Engineering & Technology, Karachi in 2006. He then joined government owned organization as an engineering officer and since then worked in different capacities that include Simulator Maintenance Manager, Manager (Telecom and Computer Networks), Manager (Training & Development), Senior Technical Manager (Air Field Electronics & Communication). Besides having professional experience in engineering domain, he had been affiliated with National University of Sciences & Technology, Islamabad as an Assistant Professor & Training coordinator from April 2014 till October, 2017. He did his M.E. in Telecomm Engineering in 2010. Also, he completed his M.S. degrees in Project Management and Software Engineering in 2017 and 2018 respectively. Currently, he is pursuing PhD in Computer Software Engineering from National University of Computer & Emerging Sciences, Islamabad, M.S. in Financial Engineering from World Quant University, Louisiana and MicroMasters in Data, Economics & Development Policy from MITx.

Muhammad Adeel Ikram joined College of Aeronautical Engineering, NUST in 2013. He achieves Bachelor Degree in the discipline of Avionics Engineering. Passionate about application designing and encouragement from the elders moved him to design an android application capable of transmission of encrypted SMS

How to cite this paper: Muhammad Noman Riaz, Adeel Ikram,"Development of a Secure SMS Application using Advanced Encryption Standard (AES) on Android Platform", International Journal of Mathematical Sciences and Computing(IJMSC), Vol.4, No.2, pp.34-48, 2018.DOI: 10.5815/ijmsc.2018.02.04