University of Ibadan
Journal of Science and Logics in ICT Research

DATA ENCRYPTION SCHEME USING AN ENHANCED BASE64 ALGORITHM

[1]Logunleko K. B, [2]Adeniji, O.D, [3]Logunleko A. M, [4]Odufowora M. O.
[1]Corresponding Author

[1]Department of Computer Science and Statistics, DS Adegbenro ICT Polytechnic, Eruku-Itori, Ewekoro, Ogun state, Nigeria. logunleko.kolawole@dsadegbenropoly.edu.ng

[2] Department of Computer Science, University of Ibadan, Oyo State, Nigeria. od.adeniji@mail.ui.edu.ng

[3]Department of Computer Science, Gateway ICT Polytechnic, Saapade, Ogun State, Nigeria. adeyinkalogunleko@gmail.com

[4]151, Scherer Avenue, Newark New Jersey, 07112, USA. muyek2O14@gmail.com

Abstract

Privacy of data is a requirement to be critically considered when such data is to be transmitted or exchanged via a public network. It is therefore a significant challenge and a huge task to secure such data in transmission. This challenge can be addressed via data encryption. The use of encryption restricts unintended recipients from viewing the data which are deemed confidential and potentially dangerous if made known to adversary. Thus, encryption is a technique that secures relevant data or information from eavesdroppers, attackers and unauthorized users. The aim of this research is to develop an enhanced model of Base64 algorithm that can secure short message service (SMS) communication system. This research developed a model that solves the problem of non-availability of key in the existing model of Base64 algorithm (B64) which cannot adequately secure data. The developed model was implemented using phonegap technology, mobile android phone with HTML5, CSS and JavaScript. This enhanced base64 algorithmic model was then applied to secure SMS communication system. Hence, security of private and confidential data via SMS could be adequately guaranteed using an enhanced base64 (EB64) algorithm.

Keywords: *Cryptography, EB64, B64, Security, Encryption, Decryption, Cipher, Key*

I. INTRODUCTION

Many times when data is exchanged electronically the privacy of the data is a requirement [15]. Encryption is thus a technique that secures relevant data from eavesdroppers, attackers and unauthorized users. Users often exchange personal and sensitive information every second over a non secured channel which may not be safe. Thus, the security of data transmission is a great problem in communication networks; a communication system is only reliable as long as it provides high level of security. Furthermore, security, integrity and confidentiality of the exchanged data should be provided over the transmission medium. Therefore, it is essential to protect data from attackers. To protect the data cryptography techniques can be used Obaida [13]. In this study, an enhanced based64 algorithm model was developed to secure SMS communication system.

II. REVIEW OF RELATED WORKS

The existing Base64 algorithm is often used to protect data during transmission but it was not adequately secured because of non-availability of the "key" [1]. According to Baraka, *et.al.*[11], the security of a system should depend on the secrecy of the key and not of algorithms. Based on this, the research work develops an enhanced Base64 algorithm that uses "key" which will in turn solves the problem of SMS communication by making it more secured.

Isnar, *et. al.*, [1] introduced Base64 Character Encoding and Decoding Modeling. The model transforms a textual data into cipher text by using Base64 encoding technique and transforms the cipher text back to plaintext. The study revealed that base64 algorithm was unlike the other

Logunleko K. B, Adeniji, O.D, Logunleko A. M, and Odufowora M. O. (2019). Data Encryption Scheme Using an Enhanced Base64 Algorithm, *University of Ibadan Journal of Science and Logics in ICT Research (UIJSLICTR),* Vol. 3, No. 1 pp. 16-24

symmetric encryption techniques, simply because of security inadequacy. The security features of the model needs to be improved by enhancing the algorithm with the use of a key.

Nurdiyanto et al. [7] proposed symmetric stream cipher using Triple transposition key method and Base64 algorithm for security. The study pointed out that symmetric type cryptography algorithm was known with many weaknesses as compared with asymmetric type algorithm. The study further asserts that symmetric stream ciphers are algorithms that work on XOR process between the plaintext and the key so as to improve the security of the symmetric stream cipher algorithm.

Robbi *et, al*., [4] carried out a study titled combination Base64 Algorithm and EOF Technique for Steganography. Steganography consists of a set of methods and techniques to embed the data into another media so that the contents are unreadable to anyone who does not have the authority to read these data. The authors discussed steganography and encoding techniques using base64, which is encoding scheme that converts the same binary data to the form of a series of ASCII code. Also, the EoF technique is used to embed encoding text performed by Base64. The authors further explained that the usage of the two methods together will definitely increase the security level for protecting such data. Hence, the research aimed to secure many types of files in a particular media with a good security and not to damage the stored files and coverage media being used.

Hassinen [12] revealed that there was no security for the text messages which indirectly leads to a lot of problems where important, sensitive and confidential information such as passwords is being accessed by unauthorized individual. Apart from all that, there are also some other cases like the mobile phone owner accidently sends messages to the wrong number and it gets worse when the mobile get stolen [16].

Chandrashekar *et. al*., [2] did a research on Image File Security using Base-64 Algorithm. The paper mainly focused on embedding the data from one format to another by designing a data conversion application which converts image file to text file and text file to image file. Usually, image loses its resolution after conversion of image is done. The authors proposed a method such that the image remains unchanged in its resolution as well in size.

III. METHODOLOGY

A. An enhanced based 64 algorithm is an algorithm developed to encrypt and decrypt non-intelligible message using a private key. The scheme uses a concept of modern encryption algorithms. The key is XOR-ed into the existing Base64 algorithmic process to enhance the security features of the existing Base64 algorithm. This enhanced algorithm is a secret-key algorithm which means of the same key is used for both encrypting and decrypting the data. The developed model is implemented using sublime text, Javascript, HTML5 and Phonegap technology with minimum of 2.56 GHz processor size, 4 GB of RAM and 16GB of ROM.

B Data Flow Diagram (DFD) for Enhanced Base64 Algorithm (EB64)

The data flow diagram for implementing the enhanced Base64 algorithm is divided into two stages. In stage one, encryption was carried out as shown in figure 1 while in stage two, data flow diagram for decryption is shown in figure 2 respectively. Figures 3 and 4 present the flowcharts of the algorithms.

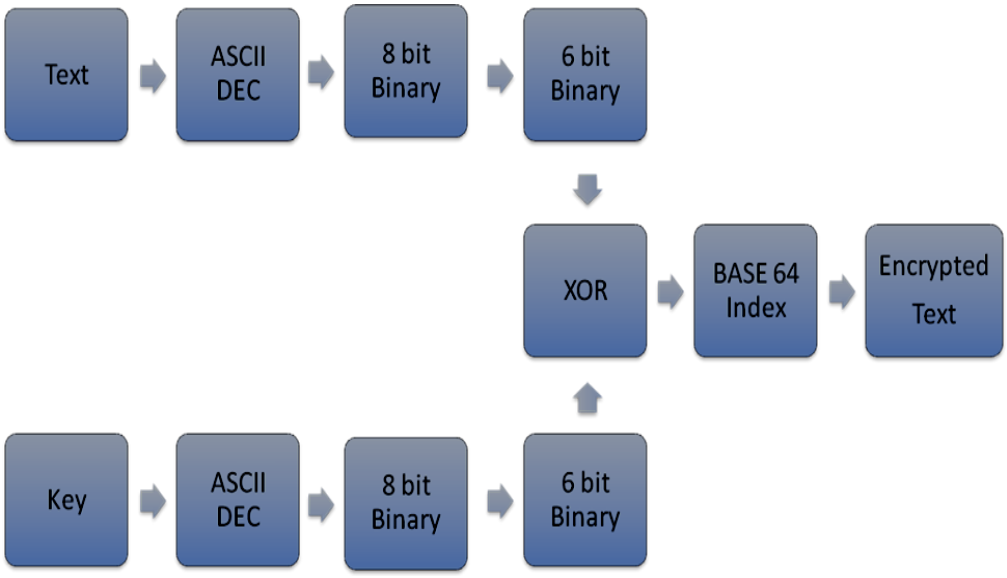

Figure 1: B.I. Enhanced BASE64 (EB64) Encryption Data Flow Diagram

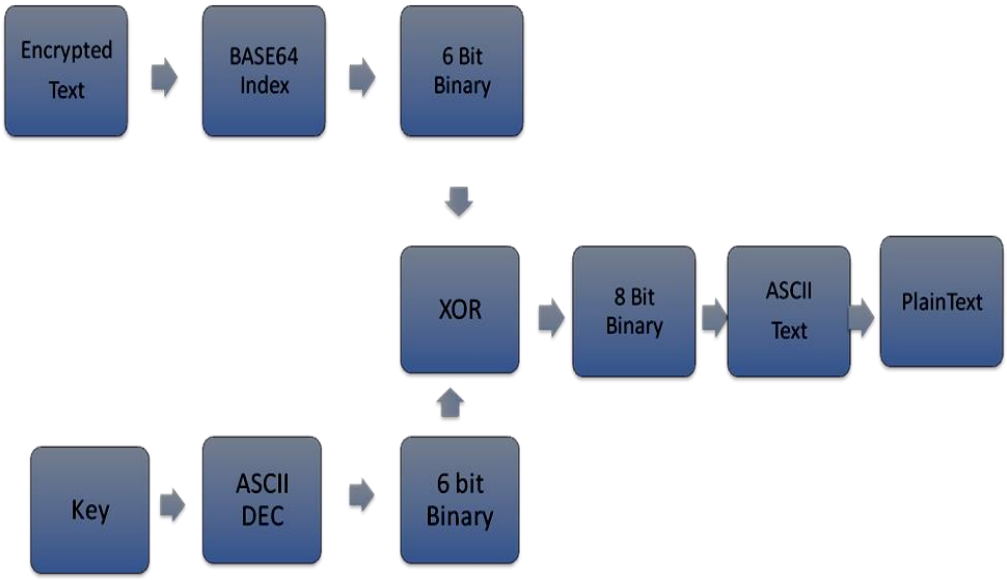

Figure 2: B.II. Enhanced BASE64 Decryption Data Flow Diagram

C. The Flow Chart of the Enhanced Base64 (EB64) Algorithm

The flow chart of the EB64 algorithm is divided into two; encryption and decryption. Encryption process was shown below in figure 3 while decryption was shown in figure 4 respectively.

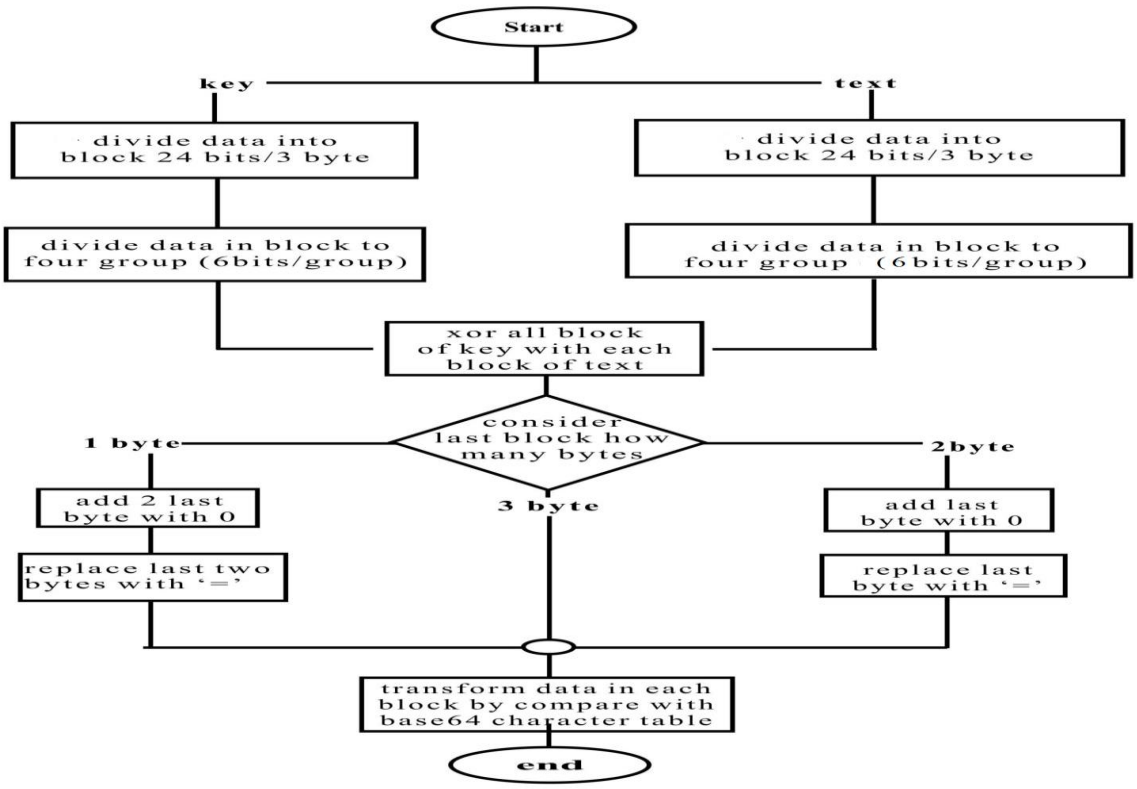

Figure 3: C.II. Flow Chart of the Base64 Decryption

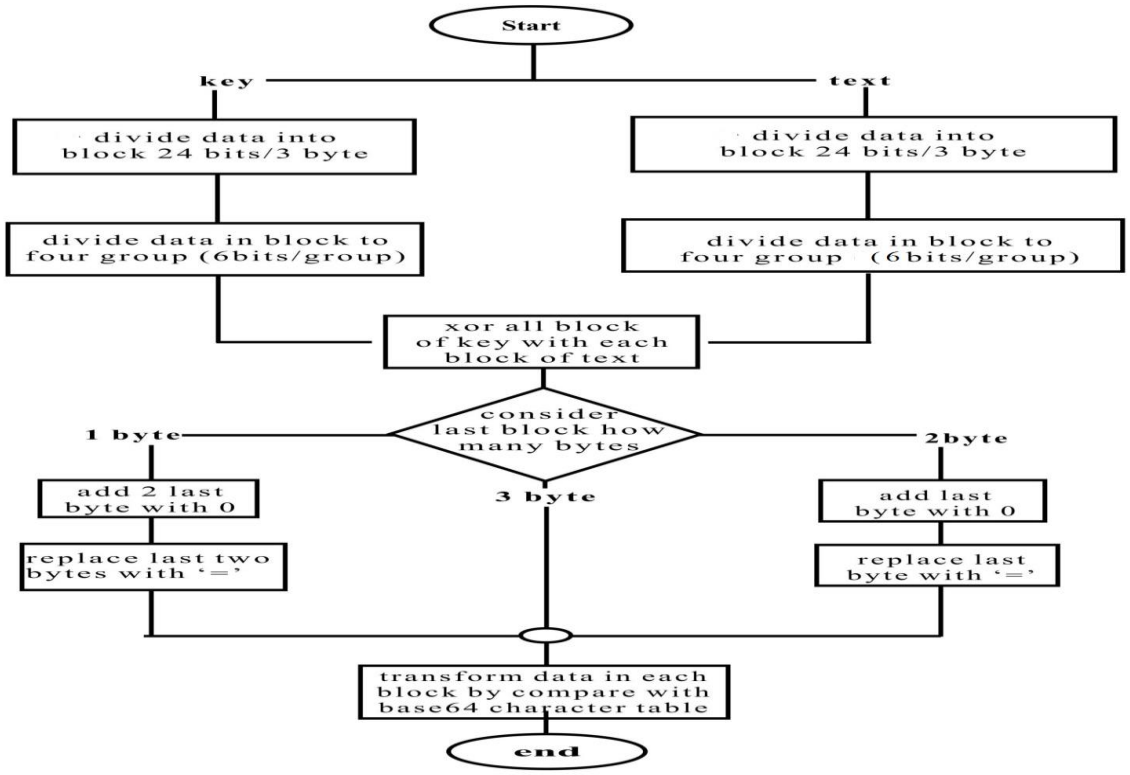

Figure 4: The Flow Chart of the Enhanced B64 Encryption Algorithm

D. The processes and steps taken in the design and implementation of this research work are explained as follows:

D.I. The steps for implementing the enhanced Base64 encryption algorithm are shown below:

i) Provide a plain text and a key
ii) Get ASCII number for each character of the plain text supplied
iii) Convert each ASCII number to 8-bits binary string
iv) Combine all binary strings together
v) Divide the combined string to each 6-bits binary string
vi) Repeat the process (ii-v) above for the given key
vii) XOR each 6-bits binary string of the plain text with all the 6-bits binary string of the key
viii) Convert each newly computed 6-bits Binary string to decimal
ix) Look for the equivalent character of the decimal value on the base64 index table.

D.II. Similarly, the steps used for implementing the enhanced Base64 decryption algorithm are as follows:

i) Supply the encrypted text and key
ii) Replace the each character of the text with its position in the base64 lookup table
iii) Convert each character of the base64 index above to a 6-bits binary
iv) Do the normal base64 encryption procedure for the key
a. Get ASCII number for each character of the key
b. Convert each ASCII number to 8-bits binary string
c. Merge all Binary Strings of the key accordingly
d. Split the merged string to each of 6-bits Binary String
v) XOR each 6-bits Binary String of the encrypted text with all the 6-bits Binary String of the key
vi) Merge all the newly computed 6-bits Binary String
vii) Break the Merged Binary String into each of 8-bits Binary String
viii) Convert each 8-bits Binary String into Decimal
ix) Get the equivalent character of each of the above decimal from the ASCII Table.

E. The Enhanced-Base64 Algorithm

The explanation of the processes and steps taken above for the design and implementation of the research were then turned to algorithm. Thus, the enhanced base64 algorithm for both encryption and decryption was shown below:

E.I Enhanced Base64 Algorithm for Encryption

$s = "a_0a_1a_2...a_n"$

$asc = array[\ ]$

$asc_i = ASCII(s_i)$

$asc_i = Binary(asc_i)$

$s = "asc_0asc_1asc_2...asc_i"$

$split6_0 = "s_0s_1s_2\ldots s_5"$

$split6_1 = "s_6s_7s_8\ldots s_{11}"$

$split6_2 = "s_{12}s_{13}s_{14}\ldots s_{17}"$

… … …

… … …

$split6_n = "s_{nx6}s_{nx6+1}s_{nx6+2}\ldots s_{nx6+5}"$

$split6_0 = split6_0 \oplus k_0 \oplus k_1 \oplus k_2 \oplus \ldots\ldots \oplus k_{last}$

$split6_1 = split6_1 \oplus k_0 \oplus k_1 \oplus k_2 \oplus \ldots\ldots \oplus k_{last}$

$split6_2 = split6_2 \oplus k_0 \oplus k_1 \oplus k_2 \oplus \ldots\ldots \oplus k_{last}$

} Key XORed

… … …

… … …

$split6_{lastn} = split6_{lastn} \oplus k_0 \oplus k_1 \oplus k_2 \oplus \ldots\ldots \oplus k_{last54}$

$split6_n = Decimal(split6_n)$

$Const = "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/"$

$split6_n = Const\ [split6_n]$

$Where\ f = 4 - (length\ (split6)(mod\ 4))$

$split6 = split6.add(=)_f$

$result = Text(split6)$

D.II. Enhanced Base64 decryption Algorithm

$Text = Text.replace(\ "=","")$

$Const = "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/"$

$Text_i = Const.indexof(\ Text_i\ )$

$Text_i = Binary6(\ Text_i\ )$

$k = "a_{1j}a_{1j}a_{1j}...a_{ij}\ldots a_{43}a_{44}"$

$asc = array[]$

$asc_i = ASCII(k_i)$

$asc_i = Binary(asc_i)$

$k = "asc_0asc_1asc_2...asc_i"$

$split6 = array[]$

$split6_0 = "asc_0asc_1asc_2\ldots asc_5"$

$split6_1 = "asc_6asc_7asc_8\ldots asc_{11}"$

$split6_2 = "asc_{12}asc_{13}asc_{14}\ldots asc_{17}"$

… … …
… … …

$split6_n$ = "$asc_{nx6} asc_{nx6+1} asc_{nx6+2} \ldots asc_{nx6+5}$"

$k = split6$

XOR each of the data in Text with all the data in k:

$Text_0 = Text_0 \oplus k_0 \oplus k_1 \oplus k_2 \oplus \ldots \oplus k_{last}$

$Text_1 = Text_1 \oplus k_0 \oplus k_1 \oplus k_2 \oplus \ldots \oplus k_{last}$

$Text_2 = Text_2 \oplus k_0 \oplus k_1 \oplus k_2 \oplus \ldots \oplus k_{last}$

Key XORed

… … …
… … …

$Text_{lastn} = Text_{lastn} \oplus k_0 \oplus k_1 \oplus k_2 \oplus \ldots \oplus k_{last}$

$Text\_merged = Text_0 Text_1 Text_2 \ldots .. Text_i$

$Text_n$ = "$Text\_merged_{8n+0} Text\_merged_{8n+1} \ldots Text\_merged_{8n+7}$"

$Text_n = Decimal(Text_n)$

$Text_n = ASCII[Text_n]$

$Merge \mathrel{+}= Text_n$

IV. RESULTS AND DISCUSSIONS

The system developed was done using phonegap technology. The system was able to encrypt and decrypt both intelligible and unintelligible messages respectively which were shown in figure 5 and 6 respectively.

a) Encrypt/Encode Model

This page allows user to encrypt and send encrypted text massages by supplying recipient phone number, encryption key (from sender name) and the message and then press "SEND" to send message. Figure 5 shows Encode Page.

b) Decrypt/Decode Page

This page allows user to decrypt encrypted text message by supplying the Encryption key and pasting the message into the message box then press "Decode" button to decrypt and read the original message. Figure 6 shows decrypt page.

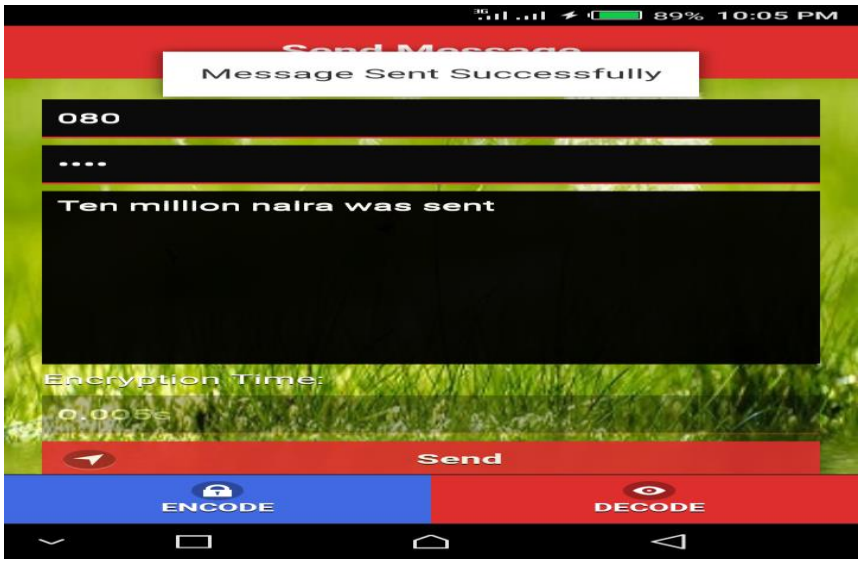

Figure 5a: Message Sent

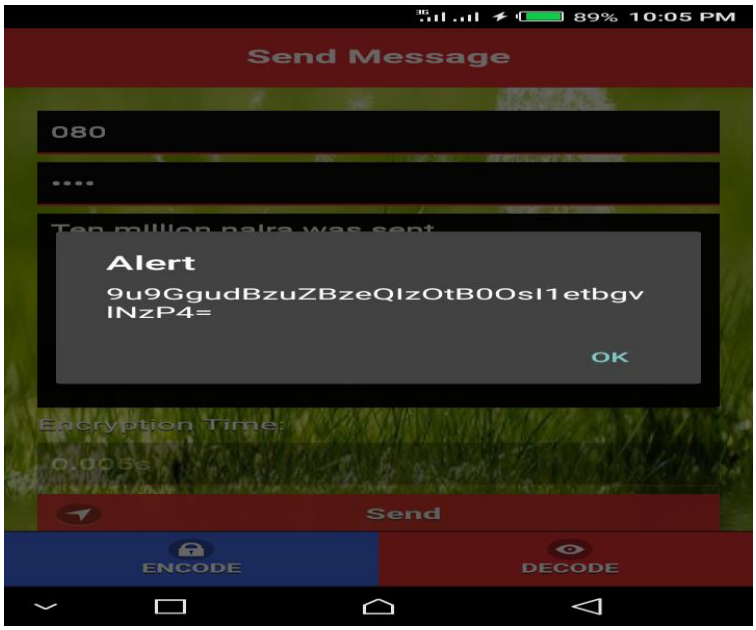

Figure 5b: Encrypted Message

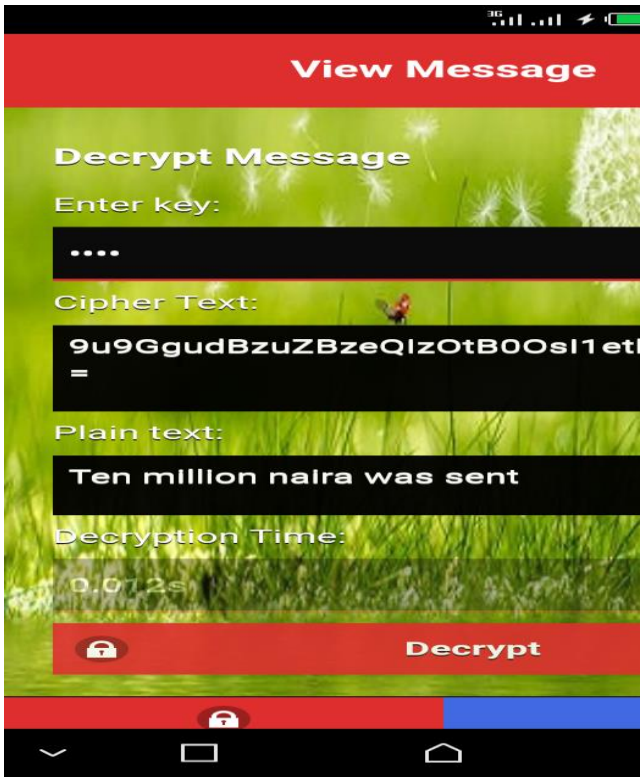

Figure 6: Decrypt Message

V CONCLUSION

The importance of encryption algorithm in communication security cannot be over-emphasized. This research work improves the security limitations of the existing Base64 Algorithm as it was not adequately secure because it was not a key-based algorithm. As a result, a key-based Base64 algorithm was developed called enhanced Base64 (EB64) algorithm which invariably solved the problem of a non-secured SMS communication system. However, the developed system runs very fast on the mobile devices as well, it does not need the addition of extra hardware. It can also be used by internet service provider for real-time application. Hence, it is no doubt that the developed model increases the security level of the existing Base64 Algorithm because of the key XORed into Base64 Algorithm which ascertains proper security of private and confidential text messages.

References

[1] Isnar S, Sumartono U , Andysah P, and Arpan (2016). Base64 Character Encoding and Decoding Modeling. *International Journal of Recent Trends in Engineering & Research*, Volume 02, Issue 12; [ISSN: 2455 1457], pp. 63-68.

[2] Guwalani P, Kala M, Chandrashekar R, Shinde dan J and Mane D (2014). Image File Security using Base-64 Algorithm, *Int. J. Computer Technology & Applications*, vol. 5, no. 6, pp. 1892-189.

[3] Solanki K, Vankani V , Pukle dan P and Iyer Y (2016). Multimedia Encryption Using Visual Cryptography. *International Journal of Recent Trends in Engineering & Research*, vol. 2, no. 9, pp. 261-264.

[4] Robbi Rahim *et. al.,* (2018). Combination Base64 Algorithm and EOF Technique for Steganography. *International Conference on Information and Communication Technology (IconICT): Journal of Physics*: Conf. Series 1007.

[5] Poonkuzhali S. M. and. Therasa M (2015). Data Hiding Using Visual Cryptography for Secure Transmission. *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 4, no. 4, pp. 440-441.

[6] Siahaan A. P., (2016). A Three-Layer Visual Hash Function Using Adler-32. *International Journal of Computer Science and Software Engineering*, vol. 5, no. 7, pp. 142-147.

[7] Nurdiyanto H,. Rahim R, and Wulan N (2017). Symmetric Stream Cipher using Triple Transposition Key Method and Base64 Algorithm for Security Improvement. *J. Phys. Conf. Ser.*, vol. 930, no. 1, p. 12005.

[8] Hariyanto E and Rahim R (2016). Arnold's Cat Map Algorithm in Digital Image Encryption. *International Journal of Science and. Research (IJSR)*, vol. 5, issue 10, pp. 1363–1365.

[9] Putera A, Siahaan U, and Rahim R (2016). Dynamic Key Matrix of Hill Cipher Using Genetic Algorithm. *International journal of Security and its Applications*, vol. 10, no. 8, pp. 173–180.

[10] Solms R.V and Niekerk J.V (2013). From information security to cyber security. *Computer. Security*, vol. 38, pp. 97–102.

[11] Baraka W, Anael S and Loserian S. (2013). Enhanced Security Model For Mobile Banking Systems In Tanzania. *International Journal of Technology Enhancements and Emerging Engineering Research*, Vol 1, Issue 4. ISSN 2347-4289.

[12] Hassinen M, Markovski S. (2009). Secure SMS messaging using Quasigroup encryption and Java SMS API. *Proceedings of the conference In SPLST'10*, Kuopio, Finland, Pp 187-200.

[13] Obaida M, Al-Hazaimeh (2013). Design of a New Block Cipher Algorithm. *Network and Complex System* 3, 1-6.

[14] Agoyi M and Seral D, (2010). SMS security: An asymmetric encryption Approach, *6th International Conference on Wireless and Mobile Communications(*ICWMC), Valencia, Spain, pp 448-452.

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

[16] De Santis A ,Castiglinone A Cattaneo G., Cembalo M., Petagna F and Petrillo U.F (2010). An extensible framework for efficient secure SMS. *Journal of systems and software* 78(1), pp 60-72.