Implementation of a Hybrid Encryption Scheme for SMS / Multimedia Messages on Android

Hazem M. El bakry
Faculty of Computer Science
Mansoura University

Ali E. Taki_El_Deen
IEEE Senior Member
Alexandria University

Ahmed Hussein El tengy
Dept. of Communication
Alexandria University

ABSTRACT
SMS/Multimedia messages are one of the popular ways of communication. Sending an SMS/MMS is cheap, fast and simple. Because of mobile networks attack or smartphones hackers, the GSM networks are not secure, so that all information or SMS/MMS messages are vulnerable. This paper describe an android application that helps the user to encrypt the message (SMS/Multimedia files) before it is transmitted over the mobile network. The new idea of the program is to transmit encrypted messages and multimedia files via mobile networks or the internet as an alternative mean. To maintain intensive security, the program uses a Hybrid encryption algorithm based on Blowfish and S-Boxes of DES encryption. Moreover, it uses a private key encrypts the files and another private key encrypts file name. The transferring media is maintained online in the absence of mobile network coverage.

Keywords
Cryptography, Blowfish Encryption, Mobile System, SMS

1. INTRODUCTION
Mobile communication devices [1] have become commonplace during the past few years, integrating multiple wireless networking technologies to support additional functionality and services. One of the most important developments that have emerged from communications technology is SMS [2]. They are designed as part of Global System for Mobile communications (GSM) [3]. Banks worldwide are using SMS to conduct some of their banking services [4]. For example, clients are able to query their bank balances via SMS or conduct mobile payments. In addition, people sometimes exchange confidential information such as passwords or sensitive data amongst each other [5]. SMS technology suffers from some risks such as vulnerabilities, eavesdroppers and unauthorized access [6].Therefore, we need to secure SMS messages and keep their contents private, without increasing their size. This paper provides a solution to this SMS security problem. Our approach is to secure the SMS/Multimedia message using an encryption (Blowfish) system [7]. The proposed technique encrypts SMS with 16-round Feistel cipher and uses large key-dependent S-boxes.

Section 2 Short Message Service (SMS) technology
Section 3 Blowfish Encryption algorithm overview
Section 4 Data Standard Encryption (DES) overview
Section 5 Implementation of SMS/MMS Encryption program
Section 6 Conclusion

2. SMS TECHNOLOGY
SMS is a standard communication service in the Global System for Mobile Communications [8]. It is a technology that enables text messages to be sent and received as same as voice calls. The short message is transmitted over the radio channel using the signaling path [9]. The first appear of SMS is in Europe in 1992. The maximum size of SMS is (160 characters if 7-bit character encoding) or (70 character if 16-bit Unicode character) is used.

SMS contains some meta-data [10]:

a. sender number and Service center number.

b. Data coding scheme and Protocol identifier.

c. Timestamp.

3. BLOWFISH ALGORITHM
Nowadays cryptography becomes a numerous techniques for data security. There are two methods for data encryption, one of them uses one key for encryption and decryption, it's known as symmetric cryptography like DES and Blowfish. The other method uses two keys for encryption and decryption it is a symmetric cryptography like RSA Algorithm [14].

3.1 Blowfish Encryption Algorithm
Blowfish [15] is a 64-bit cipher and its key length extended from 32 bits to 448 bits, it has 16 rounds and generates the key dependent S-Boxes.

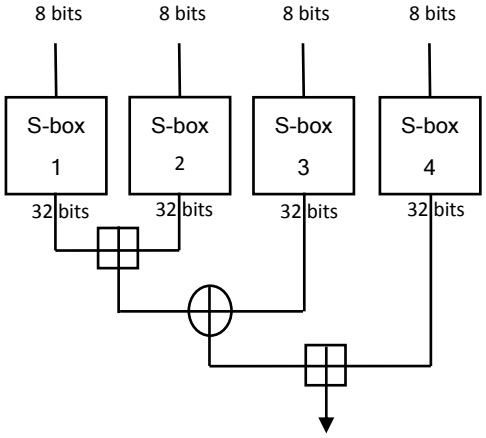

Fig.1 Substitution box of Blowfish

Figure (1) [16] shows a Function that uses four s-boxes S_1, S2, S3, S_4 derived from the encryption key. Each s-box contains 32-bit words. The s-boxes act as substitution boxes [17], by replacing an 8-bit input with a 32-bit output. *F* splits its 32-bit input into four 8-bit. It replaces each byte by the contents of an S-box, and combines the results as follows [18]:

Letting $\boxplus$ signify addition modulo 2^{32}:
$F(a, b, c, d) = ((S_1[a] \boxplus S_2[b]) \oplus S_3[c]) \boxplus S_4[d]$

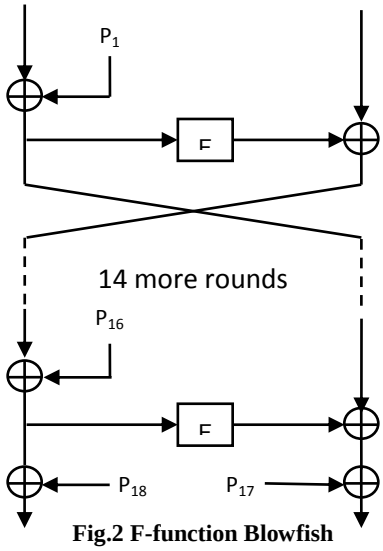

Fig.2 F-function Blowfish

Figure (2) [19] shows the block diagram of the Blowfish encryption algorithm like the Feistel network. There are 16 rounds (Feistel network); each round consists of a key-dependent permutation and a key and data-dependent substitution. All operations are XORs and additions on 32-bit words. The Key [20] is converted from 448 bits to several sub-key arrays totaling 4168 bytes. The keys are generated before data encryption or decryption.

The p-array [21] consists of (P_1, $P_{2,..}P_{18}$) sub-keys each one is 32-bit. And also four 32-bit S-Boxes each one consist of 256 entries (S_n, 0, S_n, 1,… S_n, 255).then 521 iterations are done to generate all sub-keys.

In this function, the only additional operations are four indexed array data lookup tables for each round.

4. DATA STANDARD ENCRYPTION

The data encryption standard (DES), known as the data encryption algorithm (DEA) by ANSI and the ISO, has been a worldwide standard for 20 years [22]. Although it is showing signs of old age, it has held up remarkably well against years of cryptanalysis and is still secure against all but possibly the most powerful of adversaries [23].

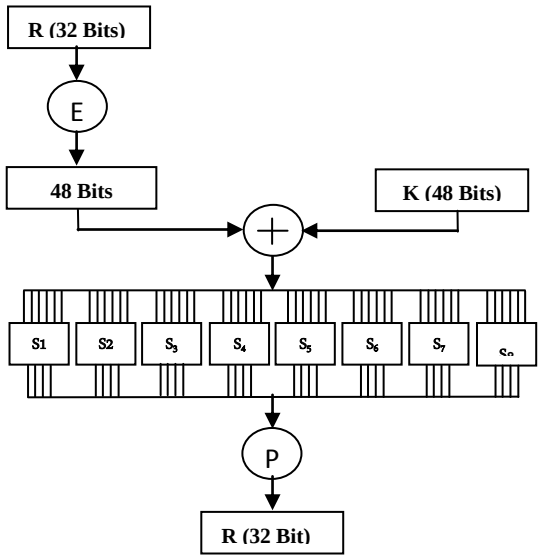

Fig.3 S-Boxes of DES

Figure (3) [24] shows the cipher function. This function is the main part of rounds consists of four separate stages as the following [25]:

4.1 Expansion

The 64-bit cipher text divided into two parts, each part contains 32-bit. Every part processes separately and expands to 64-bit using the expansion permutation.

4.2 Key generation

The result from previous process combined with a sub-key using XOR process. A 16-48 bit sub-key for 16-rounds for the DES algorithm.

4.3 Substitution Process

After mixing in the sub-key, the bits divided into eight 6-bit before s-boxes which can be called substitution boxes, each s-box provides 4-bit as output from 8-bit. The core security of DES is s-boxes.

4.4 Permutation

Finally, the 32 outputs from the S-boxes are rearranged according to a fixed permutation, the P-box. This is designed so that, after permutation, each S-box's output bits are spread across 4 different S boxes in the next round.

5. IMPLEMENTATION OF SMS/MMS ENCRYPTION PROGRAM

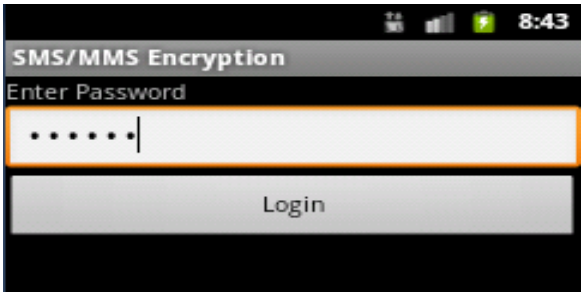

Fig.4 Login Screen of the program

Figure (4) shows the first screen in the application, which has one button, it is used for authentications login to allow authorized users only to use the application and prevent the unauthorized users from use it.

Fig.5 Start screen of SMS/MMS Encryption

Figure (5) shows the first screen in the application which has three buttons, the first button is used for encrypting and decrypting text messages, the second one is for encrypting and decrypting all files types on the mobile, the third is used for uploading or download files via Internet.

Fig.6 Main screen of encrypt/decrypt

Figure (6) shows how to encrypt or decrypt text messages.

Encrypting text by inserting private key into first textbox then the message inserted to second textbox, when pressing the "Encrypt" button, the result cipher text displayed in third textbox.

Decryption cipher text by inserting private key into first textbox, when pressing "Get Inbox SMS" button then the program opens the SMS inbox of the smart phone and picking out the income cipher message, it returns in the second textbox, when pressing the "Decrypt" button the result decrypted text message displayed in the third textbox.

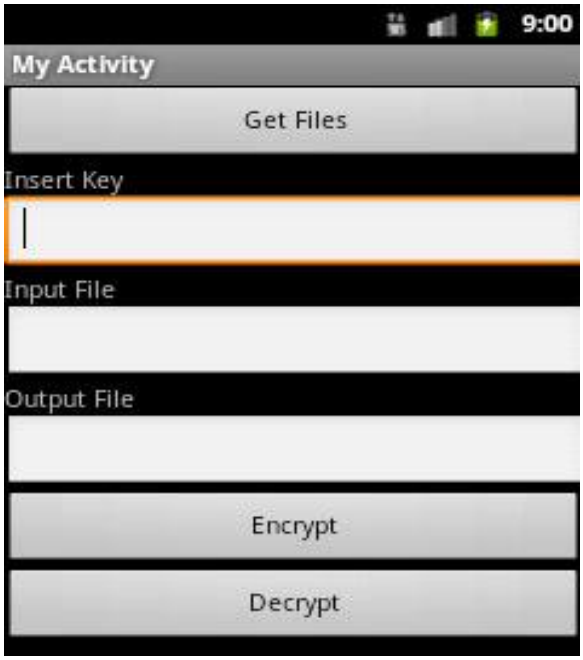

Fig.7 Main screen of encrypt/decrypt File

Figure (7) shows how to encrypt or decrypt multimedia or any type of files.

Encrypting files by inserting private key in the first textbox then when pressing the "Get Files" button, the program explores the mobile phone files and picks the file, its name and path returns in the second textbox, when pressing the "Encrypt" button, the encryption process starts, then the encrypted file name and path will be displayed in the third textbox.

Decrypting files by inserting private key into first textbox then when pressing the "Get Files" button, the program explore the mobile phone files and picks the encrypted file, its name and path returns in the second textbox, when pressing the "Decrypt" button, the decryption process starts, then the decrypted file name and path will be displayed in the third textbox.

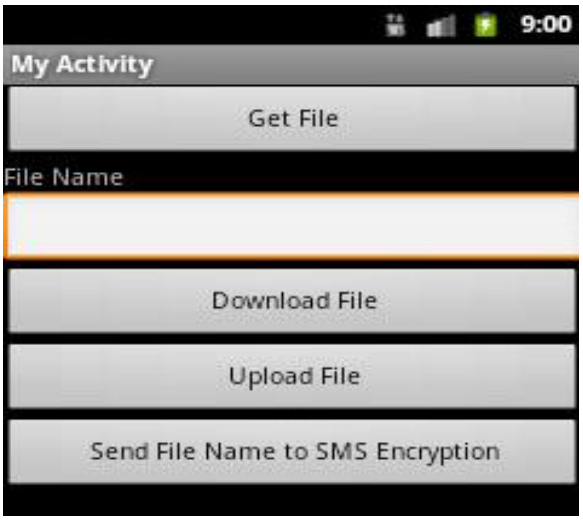

Fig.8 Alternative way to upload/download encrypted file

Figure (8) shows alternative way to upload/download files to a secured website with a username and password within the program to maintain more security so that the user has nothing to do with this information.

Uploading encrypted files: by pressing the "Get File" button, the program explores the mobile phone files and picks the file, its name and path returns in the first textbox, then when pressing the "Upload File", the program uploads the file to a secured website on the internet. When uploading completed, a notification message appears.

For high strength in the complexity of encryption, press the "Send File Name to SMS Encryption" button, as a result of that the program sends only the file name without path to the SMS encrypting screen to encrypt then send it. Downloading encrypted files: by opening the "Main screen of encrypt/decrypt SMS" window, pressing the "Get Inbox SMS" button, choosing the message including the encrypted file name, pressing the "decrypt" button to decrypt the file name, copying this file name to the clipboard, then switching back to the "Upload/download" screen, pasting the file name in the first textbox. By pressing the "Download File" button, the program will download the file from the secured website on the internet, and then when downloading process is completed, a confirming message appears.

Sample of encrypting plain text to cipher text

Example:

Figure (9) shows input text "Encrypted SMS by Blowfish" and private key "123"

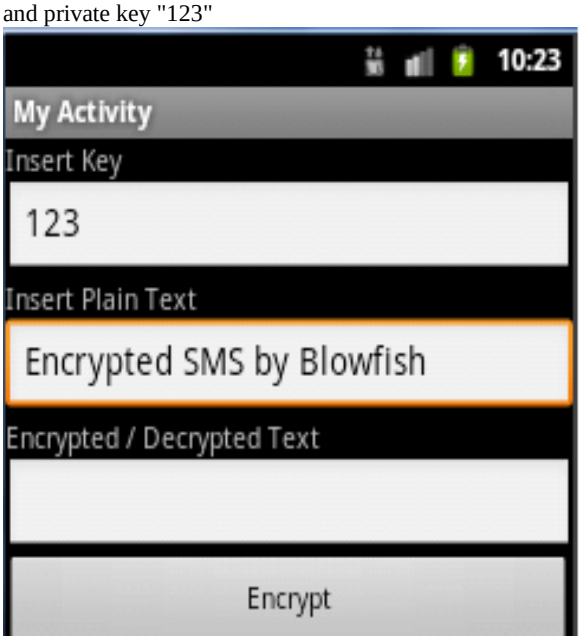

Fig.9 Input

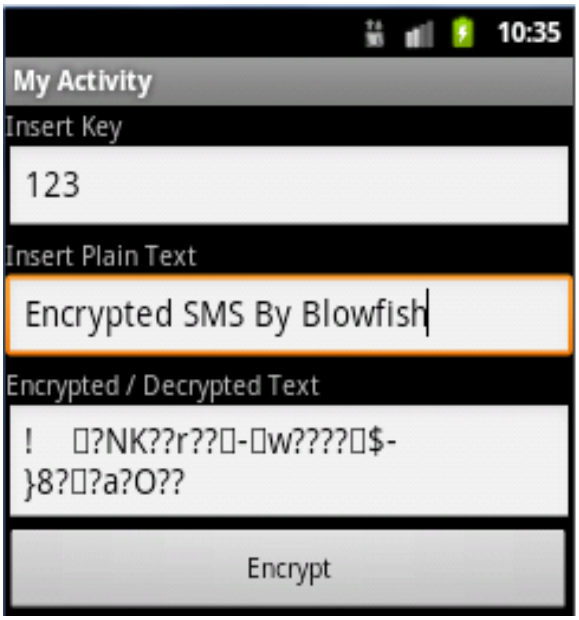

Fig.10 Output of cipher

Figure (10) shows Cipher text as a result by pressing the "Encrypt" button, the result appears in the third text box.

6. CONCLUSION

In this paper, the need of encrypting important information is shown, Because of mobile networks attack, the GSM networks are not secure. Thus, the paper described an android application that solved this problem, the program has a simple user interface and a startup screen with password added for authentication purpose. The program encrypts the text messages, image, voice, video files, and sends them via mobile networks. Two ways have been considered to send a message from one mobile to another; the first, the program sends messages via GSM networks as text messages or multi-media ones, the second, by connecting the mobile to internet through the mobile network or a wireless internet network, where the program uploads the files to a specific file hosting site that requires a user name and a pass word already created and hidden, to maintain more security, in the program. The receiver using the same program will be able to download the files. In case of weakness or losing mobile network signals, the internet connection is considered as alternative means for that.

7. REFERENCES

[1] Chin, E., Felt, A. P., Greenwood, K., and Wagner, D. "Analyzing Inter-Application Communication in Android". In Proc. of the Annual International Conference on Mobile Systems, Applications, and Services (2011).

[2] Marko Hassinen, "SafeSMS - End-to-End Encryption for SMS Messages", *IEEE International Conference on Telecommunications*, 2008, 359-365.

[3] S. Jahan, M. M. Hussain,M. R.Amin and S. H. Shah Newaz, "A Proposal for Enhancing the Security System of Short Message Service in GSM", *IEEE International Conference on Anti-counterfeiting Security and Identification*, 2008, 235-240.

[4] Mary Agoyi and Devrim Seral, "SMS Security: An Asymmetric Encryption Approach", *IEEE International Conference on Wireless and Mobile Communications*, 2010, 448-452.

[5] P. Traynor, W. Enck, P. McDaniel and T. La Porta. "Mitigating Attacks on Open Functionality in SMS-Capable Cellular Networks", IEEE/ACM Transactions on In Networking, 17(1):40-53, 2009.

[6] Mary Agoyi, Devrim Seral, "SMS Security: An Asymmetric Encryption Approach", Sixth International Conference on Wireless and Mobile Communications, 2010 IEEE, pp 448-452.

[7] Ferguson, N., Schneier, B. and Kohno, "Cryptography Engineering: Design Principles and Practical Applications", T. Indianapolis: Wiley Publishing, Inc. 2010.

[8] Roland Schloglhofer, "Secure and Usable Authentication on Mobile Devices", MoMM2012, 3-5 December, 2012, Bali, Indonesia. ACM 978-1-4503-1307-0/12/12 (pp 257-262).

[9] M. Toorani and A. A. Behesti, "SSMS – A Secure SMS Messaging Protocol for the M-Payment Systems", IEEE Symposium on Computers and Communications, 2012, 700-705.

[10] Marko Hassinen, "SafeSMS- End-to-End Encrption for SMS Messages", IEEE International Conference on Telecommunications, 2008, 359-365.

[11] Kuo-Tsang Huang, Jung-Hui Chiu, and Sung-Shiou Shen, "A Novel Structure with Dynamic Operation Mode for Symmetric-Key Block Ciphers". International Journal of Network Security & Its Applications (IJNSA) 5 (1): 19, (January 2013).

[12] Mary Agoyi and Devrim Seral, "SMS Security: An Asymmetric Encyrption Approach", IEEE International Conference on Wireless and Mobile Communication, 2010, 448-452.

[13] Jawahar Thakur and Nagesh Kumar, "DES, AES and Blowfish: Symmetric Key Cryptography Algorithms Simulation Based Performance Analysis", International Journal of Emerging Technology and Advanced Engineering, (ISSN 2250-2459, Volume 1, Issue 2, December 2011)

[14] Dripto Chatterjee, Joyshree Nath, Soumitra Mondal, Suvadeep Dasgupta and Asoke Nath, " Advanced Symmetric key Cryptography using extended MSA method: DJSSA symmetric key algorithm", Jounal of Computing, Vol 3, issue-2, Page 66-71, Feb(2011).

[15] Trisha Chatterjee, Tamodeep Das, Joyshree Nath, Shayan Dey and Asoke Nath, "Symmetric key cryptosystem using combined cryptographic algorithms- generalized modified vernam cipher method, MSA method and NJJSAA method: TTJSA algorithm", Proceedings of IEEE International conference: World Congress WICT-2011 t held at Mumbai University 11-14 Dec, 2011, Page 1179-1184(2011).

[16] Somdip Dey, Asoke Nath, "Modern Encryption Standard (MES) Version-I: An Advanced Cryptographic Method", Proceedings of IEEE 2nd World Congress on Information and Communication Technologies (WICT-2012), pp. 242-247.

[17] Dripto Chatterjee, Joyshree Nath, Sankar Das, Shalabh Agarwal and Asoke Nath, "Symmetric key Cryptography using modified DJSSA symmetric key algorithm", Proceedings of International conference Worldcomp 2011 held at LasVegas 18-21 July 2011, Page-306-311, Vol 1(2011).

[18] E. Barker and A. Roginsky, "Recommendation for the Transitioning of Cryptographic Algorithms and Key Sizes", NIST SP 800-131, 2010, Technical Report.

[19] Somdip Dey, Joyshree Nath, Asoke Nath, "An Integrated Symmetric Key Cryptographic Method - Amalgamation of TTJSA Algorithm, Advanced Caeser Cipher Algorithm, Bit Rotation and reversal Method : SJA Algorithm", International Journal of Modern Education and Computer Science (IJMECS), ISSN: 2075- 0161 (Print), ISSN: 2075-017X (Online), Vol 4, No 5, Page 1-9, 2012.

[20] Debanjan Das, Joyshree Nath, Megholova Mukherjee, Neha Chaudhury and Asoke Nath, "An Integrated symmetric key cryptography algorithm using generalized vernam cipher method and DJSA method: DJMNA symmetric key algorithm", Proceedings of IEEE International conference: World Congress WICT-2011 to be held at Mumbai University 11-14 Dec, 2011, Page 1203-1208(2011).

[21] Jiao Wentao, "Cloud computing environments cryptographic applications", Chinese Association for Cryptologic Research, vol. 5, no. 1, pp.20-29, 2011.

[22] Satyaki Roy, Navajit Maitra, Joyshree Nath, Shalabh Agarwal and Asoke Nath, "Ultra Encryption Standard(UES) Version-I: Symmetric Key Cryptosystem using generalized modified Vernam Cipher method, Permutation method and Columnar Transposition method", Proceedings of IEEE sponsored National Conference on Recent Advances in Communication, Control and Computing Technology-RACCCT 2012, 29-30 March held at Surat, Page 81- 88(2012).

[23] Asoke Nath, Saima Ghosh, Meheboob Alam Mallik, "Symmetric Key Cryptography using Random Key generator", Proceedings of International conference on security and management (SAM'10) held at Las Vegas, USA Jull 12-15, 2010, Vol 2, Page: 239-244(2010).

[24] Bruce Schneier, "Applied Cryptography: Protocols, Algorithms, and Source Code in C",(2011).

[25] William E. Burr, "Data Encryption Standard", in NIST's anthology",A Century of Excellence in Measurements, Standards, and Technology: A Chronicle of Selected NBS/NIST Publications,(2000).