Scientific and Practical Cyber Security Journal (SPCSJ) 3(1): 12-20 ISSN 2587-4667 Scientific Cyber Security Association (SCSA)

DEVELOPMENT OF AN EFFICIENT HYBRID ENCRYPTION SCHEME FOR SECURING SHORT MESSAGE SERVICE (SMS)

Faisal A. Garba, Prof. A. A. Obiniyi, Prof. S. E. Abdullahi
[1]Department of Computer Science Education, Sa'adatu Rimi College of Education, Kano.
[2,3]Department of Computer Science, Ahmadu Bello University, Zaria
[3]Nigerian Turkish Nile University

ABSTRACT. Majority of mobile device users will prefer to preserve the privacy of their SMS communication using mobile device SMS encryption solutions. The mobile devices in use however, are highly constrained in terms of memory, power and computing capability to utilize the current SMS encryption solutions. As a result of this, there is a room for improvement in terms of the speed efficiency of the SMS encryption schemes proposed for use on mobile devices. This research proposed an end-to-end SMS encryption scheme ideal for use on mobile devices using a hybrid combination of cryptographic algorithms: Blowfish symmetric encryption algorithm, Elliptic Curve Diffie Hellman (ECDH) and Elliptic Curve Digital Signature Algorithm (ECDSA). The proposed scheme was implemented using Java programming language to develop SMS encrypting Android application. The time taken for the proposed scheme cryptographic operations was measured on five different android mobile devices with varying processor speed. The operation measured was the time taken for encryption, decryption and key generation. The research results revealed that the proposed scheme has a faster rate of key generation, encryption and decryption. This research has provided an end-to-end hybrid SMS encryption scheme ideal for use on constrained mobile devices using a hybrid combination of cryptographic algorithms: Blowfish symmetric encryption algorithm, Elliptic Curve Diffie Hellman (ECDH) and Elliptic Curve Digital Signature Algorithm (ECDSA) and is therefore an improvement in term of speed to the existing SMS encryption schemes on mobile devices.

KEYWORDS: SMS, encryption, Blowfish, cryptography, ECDH, ECDSA, Android

There are various ways of securing SMS and one of such is cryptography. In cryptography messages are encoded in such a way that only the sender and the receiver can know it's content (Jha *et al.*, 2016). Cryptography is of three forms: symmetric key cryptography (secret key cryptography) and asymmetric key cryptography (public key cryptography) and cryptographic hash functions. In symmetric key cryptography, the same key is use to encrypt as well as to decrypt data, whereas in asymmetric key cryptography two keys public key and private key are used to encrypt and decrypt data. Private key is only known to the owner but public key is made known to all intended communicating parties. Whereas symmetric key algorithm requires less computational power, asymmetric key algorithm requires very much computational power since it's computation requires the exponentiation of large numbers and consequently more memory for the computation and storage of keys. However, in symmetric key algorithm there lies the problem of key agreement and secure exchange of the agreed key. In addition, user authentication, non repudiation and message integrity cannot be provided with the use of symmetric key algorithm. The combination of symmetric and asymmetric encryption algorithms cover up for their individual weakness (Kuppuswamy and Al-Khalidi, 2014). Hash functions are also called message digests or one way encryption. In hashing, a unique hash value of a plaintext is produced. It is unique in the sense that no two different plaintexts can have the same hash value. It is also one way since we cannot recover the plaintext, given the corresponding hash value. Hash value is also called digital fingerprint (Kessler, 2017).

Proposed System Architecture

Figure 1 is the proposed scheme's system architecture. In the system architecture we have two communicating parties Aisha and Buhari. Either of the communicating parties can initiate the communication process. They used ECDH-ECDSA to generate a shared secret which serve as a temporary key. The temporary key is used alongside Blowfish encryption algorithm to encrypt and exchange the permanent Blowfish key. The permanent Blowfish key, can now be used with the Blowfish encryption algorithm to exchange SMS. Other entities in the architecture are the database which is used in storing the keys as well as the SMS messages and the mobile network operator.

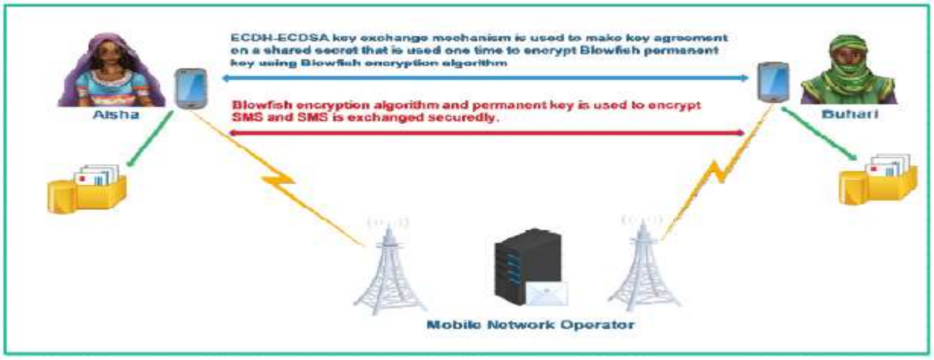

ECDH-ECDSA key exchange mechanism is used to make key agreement on a shared secret that is used one time to encrypt Blowfish permanent key using Blowfish encryption algorithm
Aisha
Buhari
Blowfish encryption algorithm and permanent key is used to encrypt SMS and SMS is exchanged securedly.
Mobile Network Operator

Fig.1: Proposed Efficient SMS Hybrid Encryption Scheme Architecture

Proposed Scheme's Pseudocode

```
Step 1: Aisha selects an integer $X_A$ to serve as her private key and go on to generate $Y_A = X_A \times G$ to serve as her public key.

Step 2: Aisha sends the public key $Y_A$ to Buhari signed with her ECDSA private key.

Step 3: Buhari verifies that the public key $Y_A$ is from Aisha by using Aisha's ECDSA public key and then picks an integer $X_B$ to be his private key and calculate his public key thus, $Y_B = X_B \times G$.

Step 4: Buhari sends the public key $Y_B$ to Aisha signed with his ECDSA private key.

Step 5: Aisha verifies that the public key $Y_B$ is from Buhari using Buhari's ECDSA public key, Aisha computes her secret shared session key thus $K = X_A \times Y_B$.
```

Step 6: Buhari also calculates his shared session key thus $K = X_B \times Y_A$.

Step 7: Aisha uses Blowfish encryption algorithm and K to encypt permanent Blowfish key K' and send it to Buhari.

Step 8: Buhari accept the encrypted message and decrypt it with his shared secret key generated in step 1 to recover the permanent Blowfish key.

Step 9: Aisha and Buhari can now exchange SMS encrypted with Blowfish encryption algorithm

Proposed System Design

Three Unified Modeling Language (UML) diagrams: use case diagram, activity diagram and sequence diagrams were used to illustrate the proposed system. Use case diagrams illustrates system's functionality, class diagram shows the different classes in the system as well as the relationship amongst them and sequence diagram to illustrate the interactions amongst the proposed system entities.

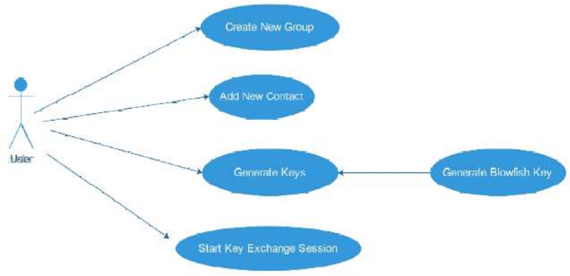


Fig. 2 System Pre-processing Use Case Diagram

Figure 2 is the system pre-processing use case diagram. It shows some tasks that are necessary to be performed before the user can fully utilizes the SMS application. First the user has to create a new group and add a contact to the group. As soon that is done, the app automatically initiates a key exchange session with the contact that has just been added to the group**.**

Proposed Scheme Implementation

To implement the proposed scheme as a proof of concept, a mobile application in Android was developed. Android mobile operating system was selected because it is open source and has a wider user base than any other mobile operating system. The target Android version is Android 4.0 (Ice Cream Sandwich). The developed Android program has been compiled into an Android Package Kit (APK) file. The apk file is installed into five Android devices for testing. The proposed work of Azaim et al. (2016) was also implemented in Android and the compiled apk file installed on five Android devices. The speed efficiency of the proposed scheme is then compared with that of Azaim et al. (2016) based on time taken for encryption and decryption versus the CPU clock rate of the five Android mobile devices.

System Testing

Azaim et al. (2016) proposal was also implemented on Android to give room for a fair evaluation. The tests for the encryption and decryption of the symmetric algorithms (Blowfish and AES-Rijndael) were carried out 100 times on five Android mobile devices with varying memory, processor and battery power shown in Table 1.

RESULTS

Results Analysis

This section reports on the result analysis of the proposed SMS scheme and Azaim et al. (2016) scheme. The analysis was conducted on five android mobile devices. The objectives of the analysis were to: 1. compare the operation of the proposed SMS scheme in terms of encryption execution time on five different mobile devices. 2. compare the operation of Azaim et al. (2016) scheme in terms of the encryption execution time on five different mobile devices.

Table 1: Specification of the Android devices used for the test

Mobile Device	Oppo A37F	Itel it1556	Tecno L9	LG Nexus 5	Tecno Camon C7
Android Version	Android 5.1 (Lolipop)	Android 5.1 (Lolipop)	Android 7.0	Android 6.0 (Marshmallow)	Android 6.0 (Marshmallow)
Central Processing Unit (CPU) count	Quad Core	Quad Core	Quad Core	Quad Core	Quad Core
Central Processing Unit (CPU) Type/Microprocessor	Snapdragon 801	Cortex A53	Cortex A7	Cortex A53	Cortex A53
Central Processing Unit (CPU) Clock Rate	1.2 GHz	1.2 GHz	1.3 GHz	2.3 GHz	1.3 GHz
System on a chip (SoC) /Microcontroller	Qualcomm Msm8974Ac	MediaTek 6572	MediaTek MT6572	Qualcomm MSM8974 Snapdragon 800	MediaTek MT6735
RAM	2GB	512MB	2GB	2GB	2GB
System Storage	16GB	8GB	16GB	16GB	16GB
Maximum Memory Card Size	256 GB	32GB	128GB	No Card Slot	128GB

3**.** compare the efficiency of the proposed SMS scheme with Azaim et al. (2016) in terms of encryption and decryption times using different SMS sizes. 4. compare the proposed SMS scheme with the Azaim et al. (2016) SMS scheme in terms of total time taken for cryptographic operations on 1 page SMS

Test for Efficiency of the Proposed SMS Scheme on Mobile Devices

This subsection presents the results obtained from the comparative analysis of the operation of the proposed SMS scheme using cryptographic operations on five mobile devices. Table 2 shows the obtained test result. From the Table 2 it can be seen that Camon C7 with a CPU clock rate of 1.3GHz has the lowest total overhead of 0.32ms followed by LG Nexus in with a CPU clock rate of 2.3 GHz having the total overhead of 0.36ms, followed by Oppo A37f with a CPU clock rate of 1.2 GHz having the total overhead of 0.45ms, followed by Tecno L9 Plus with a CPU clock rate of 1.3 GHz having the total overhead of 0.59ms and lastly ITEL IT1556 with a CPU clock rate of 1.2GHz having the total overhead of 0.71ms. The proposed efficient hybrid SMS encryption scheme chart is presented in Figure 3. The test for each of the cryptographic operations were ran one hundred times and average time recorded in millisecond.

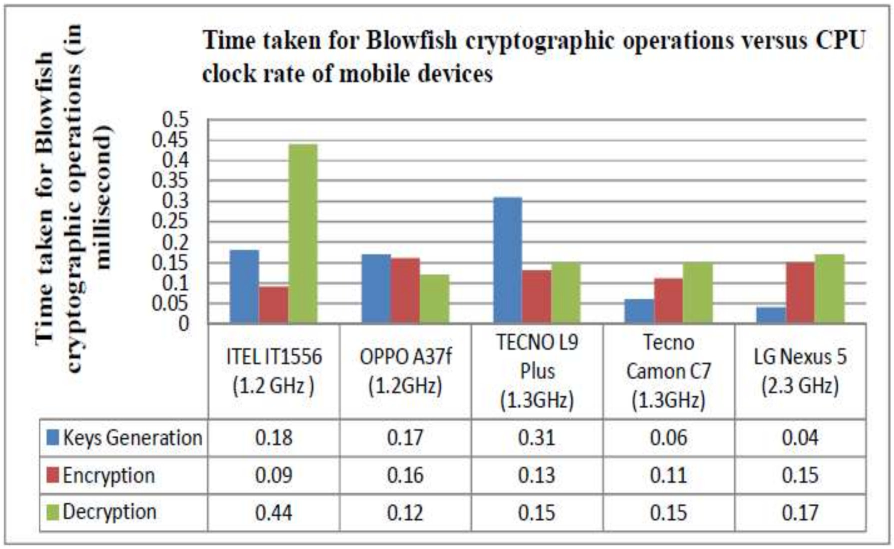
Test for Efficiency of Azaim et al. (2016) SMS Scheme on Mobile Devices

This subsection presents the result obtained from the comparative analysis of the cryptographic operations of Azaim *et al.*(2016) scheme using five mobile devices. Table 3 presents the test results obtained from running the Azaim *et al*. (2016) proposed SMS encryption scheme. From the table we can see that Tecno Camon C7 with a CPU clock rate of 1.3GHz has the lowest total overhead of 0.64ms, this is followed by LG Nexus 5 with a CPU clock rate of 2.3GHz having the total overhead of 0.76ms, followed by Oppo A37f with a CPU clock rate of 1.2GHz having the total overhead of 0.79ms, this is followed by Tecno L9 Plus with a CPU clock rate of 1.3GHz and having the total overhead of 0.95ms and lastly is ITEL IT1556 with a CPU clock rate of 1.2 GHz having a total overhead of 1.36ms. Azaim *et al*. (2016) scheme results chart is shown in Figure 4.

Table 2: Comparative Result in terms of Encryption Execution Time (in millisecond) of the Proposed SMS Hybrid Encryption Scheme Applied to Five Different Mobile Devices

Cryptographic Operations	OPPO A37f	ITEL IT1556	TECNO L9 Plus	LG Nexus 5	TECNO Camon C7
Blowfish Key Generation	0.17	0.18	0.31	0.04	0.06
Blowfish Encryption	0.16	0.09	0.13	0.15	0.11
Blowfish Decryption	0.12	0.44	0.15	0.17	0.15

Total Overhead	0.45	0.71	0.59	0.36	0.32



	ITEL IT1556 (1.2 GHz)	OPPO A37f (1.2GHz)	TECNO L9 Plus (1.3GHz)	Tecno Camon C7 (1.3GHz)	LG Nexus 5 (2.3 GHz)
🟦 Keys Generation	0.18	0.17	0.31	0.06	0.04
🟥 Encryption	0.09	0.16	0.13	0.11	0.15
🟩 Decryption	0.44	0.12	0.15	0.15	0.17

Fig 3: Proposed SMS Encryption Scheme Test Result Chart

Comparison of the Proposed SMS Encryption Scheme with Azaim et al. (2016) Scheme

This subsection presents the results obtained from the comparative analysis of the cryptographic operations of the proposed SMS scheme with the Azaim *et al*. (2016) SMS scheme using the encryption and decryption total time on different SMS size and the total time taken for the cryptographic operations. Table 3 shows the results obtained for encryption time (in milliseconds). Table 4 present the results obtained for decryption time (in milliseconds) while Table 5 presents the total time taken for cryptographic operations on 1 page SMS. From Table 3 it could be clearly seen that the proposed SMS Encryption Scheme takes less time to execute the cryptographic operations when compared with the proposed scheme of Azaim *et al*. (2016). The proposed SMS encryption scheme has the lowest encryption average time of 0.36ms with the proposed scheme of Azaim *et al*. (2016) having the average encryption time of 0.55ms. On the other hand, the proposed scheme has the highest throughput of 0.10 kb/ms with proposed scheme of Azaim *et al*. (2016) having the throughput of 0.62kb/ms.

Table 3: Comparative Result in Terms of Encryption Time (in milliseconds) using Blowfish and AES Rijndael using Different SMS Sizes.

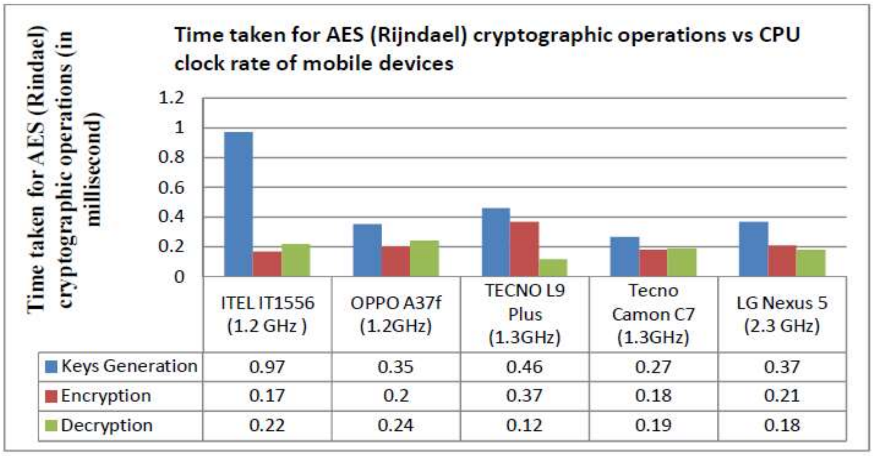
SMS Size(Kb)	Time(Millisecond)	
	Blowfish	Rijndael
0.1367	0.11	0.18
0.2734	0.30	0.51
0.4102	0.46	0.71
0.5469	0.56	0.81
Average Time	0.36	0.55
Throughput	0.10	0.62

From Table 4 results it could be clearly seen that the proposed SMS Encryption Scheme takes less time to execute its decryption when compared with the proposed scheme of Azaim *et al*. (2016). The proposed SMS decryption scheme has the lowest decryption average time of 0.34ms with the proposed scheme of Azaim *et al*. (2016) having the average encryption time of 0.48ms. On the other hand, the proposed scheme has the highest throughput of 1.0 kb/ms with proposed scheme of Azaim *et al*. (2016) having the throughput of 0.71 kb/ms.

Table 4: Comparative Result in Terms of Decryption Time (in Milliseconds) using Blowfish and AES Rijndael using Different SMS Sizes.

SMS Size(Kb)	Time(Millisecond)	
	Blowfish	Rijndael
0.1367	0.15	0.19
0.2734	0.28	0.38
0.4102	0.37	0.6
0.5469	0.57	0.76
Average Time	0.34	0.48
Throughput	1.00	0.71

Table 5 is a table of comparison of cryptographic operations on 1 page SMS between the proposed SMS encryption scheme and the proposed SMS encryption scheme of Azaim *et al*. (2016). From the table it could be seen that the lowest time difference for the execution of the cryptographic operations between the proposed SMS encryption scheme and that of Azaim *et al*. (2016) is achieved using Tecno L9 Plus (1.3 GHz), a time difference of 37.89% in favour of the proposed SMS encryption scheme and the highest cryptographic operations execution time achieved with LG Nexus 5 (2.3 GHz), a time difference of 52.63% in favour of the proposed SMS encryption scheme. A chart of the comparison is presented in Figure 4.



	ITEL IT1556 (1.2 GHz)	OPPO A37f (1.2GHz)	TECNO L9 Plus (1.3GHz)	Tecno Camon C7 (1.3GHz)	LG Nexus 5 (2.3 GHz)
🟦 Keys Generation	0.97	0.35	0.46	0.27	0.37
🟥 Encryption	0.17	0.2	0.37	0.18	0.21
🟩 Decryption	0.22	0.24	0.12	0.19	0.18

Fig 4: Test Result Chart for Azaim et al. (2016)

Table 5: Comparison of the total time taken for the cryptographic operations on 1 page SMS between the proposed scheme and Azaim et al. (2016)

	ITEL IT1556 (1.2 GHz)	Oppo A37f (1.2GHz)	Tecno L9 Plus (1.3GHz)	Tecno Camon C7 (1.3GHz)	LG Nexus 5 (2.3 GHz)
Proposed Scheme	0.71	0.45	0.59	0.32	0.36
Azaim et al. (2016)	1.36	0.79	0.95	0.64	0.76
Time Difference	0.65	0.34	0.36	0.32	0.40
Percentage Difference	47.79%	43.03%	37.89%	50%	52.63%

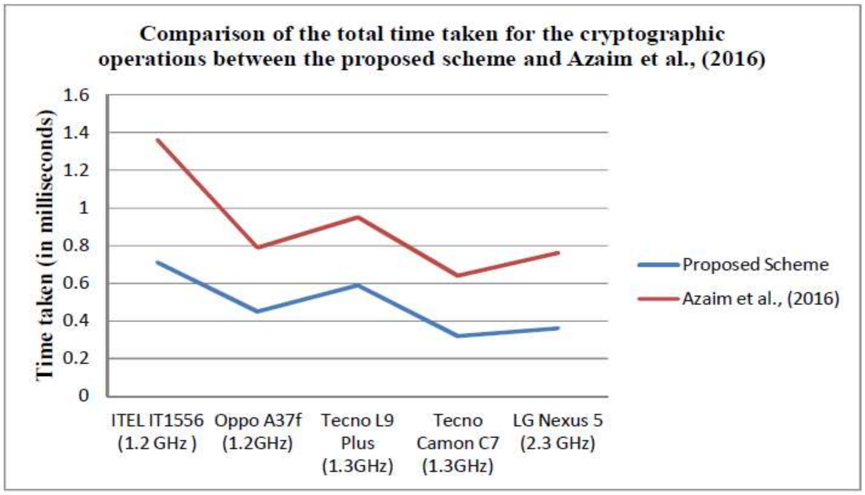


Fig 5: Comparison of the total time taken for the cryptographic operations

DISCUSSION

This research has developed an efficient hybrid SMS encryption scheme for mobile devices, using a combination of cryptographic algorithms—Blowfish encryption algorithm using ECDH-ECDSA key exchange mechanism.

The major findings from this work are:

a. The combination of the cryptographic algorithms—Blowfish encryption algorithm using ECDH-ECDSA key exchange mechanism provided more efficient SMS encryption scheme than the combination of AES (Rijndael) proposed by Azaim et al. (2016)

b. The combination of the cryptographic algorithms—Blowfish encryption algorithm using ECDH-ECDSA key exchange mechanism provided an appropriate scheme for encrypting other data in mobile device apart from SMS.
c. Blowfish encryption algorithm takes less time to compute its cryptographic operations than AES (Rijndael).
d. This research work has confirmed that clock rate should not be the only benchmark for evaluating the computing performance of mobile devices. Other factors such as pipeline depth and instruction sets should be put into consideration while comparing different processors. This is referred to as the megahertz myth (Linden, 2006).

Although there is a suspicion that the recommended Elliptic Curve Cryptography (ECC) which includes ECDH and ECDSA, parameters may likely contain backdoors as suggested by Bruce Schneir, a well known cryptologist who invented the Blowfish symmetric algorithm (Schneir, 2013).

For future work, there is the need for further research on curve parameters used by the ECC, which were recommended by the National Institute of Standards and Technology (NIST) Federal Information Processing Standards (FIPS) 186-3.

REFERENCES

[1]. Azaim, M. H., Sudiharto, D. W., & Jadied, E. M. (2016). Design and Implementation of Encrypted SMS on Android Smartphone Combining ECDSA - ECDH and AES. *The 2016 Asia Pacific Conference on Multimedia and Broadcasting (APMediaCast),* 18-23.

[2]. Jha, S., Dutta, U., & Gupta, P. (2016). SMS Encryption using NTRU Algorithms on Android Application. *International Journal of Scientific Engineering and Applied Science*, *2*(1), 331-338.

[3]. Kessler, G. C. (2017). *An Overview of Cryptography* (Updated version 26 February, 2017). Retrieved from https://commons.erau.edu/publication/412/

[4]. Kuppuswamy, P., & Al-Khalidi, S. Q. (2014). Hybrid Encryption/Decryption Technique Using New Public Key and Symmetric Key Algorithm. *International Journal of Information and Computer Security, 6*(4), 372-382.

[5]. Linden, G. (2006). *Instruction-level Parallelism*. Retrieved March 30, 2018 from: https://www.cse.unsw.edu.au/~cs9244/06/seminars/01-gvdl.pdf

[6]. Schneier, B. (2013). *The NSA Is Breaking Most Encryption on the Internet*. Retrieved February 19, 2018, from

a. https://www.schneier.com/blog/archives/2013/09/the_nsa_is_brea.html#c16759