IOP Conf. Series: Journal of Physics: Conf. Series **1175** (2019) 012073 doi:10.1088/1742-6596/1175/1/012073

Effect of the Application of TEA Algorithm on the Development of Secure Phone Application Android Smartphones

Ryan Ari Setyawan[1], Selo[2], Bimo Sunarfri Hantono[3]

[1]Department of Informatics Engineering, Janabadra University, Yogyakarta, Indonesia 55231.
[2,3]Department of Electrical Engineering and Information Technology, Gadjah Mada University, Yogyakarta, Indonesia.

ryan@janabadra.ac.id

Abstract. Android smartphone technology is growing rapidly. It is indicated that by utilizing the features of internet telephone service via the Voice over Internet Protocol (VoIP), users are able to make conversation easily and inexpensively. However, with the development, the internet telephone service used is unnecessarily safe for use. One of the solutions proposed is to make end-to-end encryption. This research applied TEA algorithm for encryption into the VoIP client of Sipdroid in Android smartphones. The objective of the research is to find out the effect of the application of TEA algorithm for encryption into the VoIP client of Sipdoid by investigating the performance of TEA algorithm in view of mean opinion score (MOS), delay, and throughput in VoIP. The results of measurement and calculation show that the application of TEA algorithm produced quality speech with R factors of 72.11676-75.01362 smaller than no TEA algorithm application with those of 84.79064-85.38539. Meanwhile, the effect of VoIP performance on such application generated a much larger delay of 0.018084 to 0.024327 seconds compared to no TEA algorithm application with a delay of 0.006071458 to 0.009465105 seconds. The results indicate that the application of TEA algorithm could more effectively contribute to the security of VoIP client practice, particularly in Android smartphones.

1. Introduction

The recent features of communication using internet phone service are growing rapidly. Various applications utilizing the internet phone services such as *Yahoo Messenger*, *Kakao Talk*, *We Chat*, *Line* and *Bee talk*, has increasingly developed. These applications are very easily used as the users can communicate with each other if the internet network connection is available. Moreover, such communication through internet phone service has several advantages, including low cost and no limitation of distance [1]. Actually, the concept of internet phone service is to build communication by utilizing the network technology of VoIP to access Internet Multimedia System (IMS). However, the development of technology did not necessarily provide the guaranteed security system. Therefore, security is necessary for the application. Various techniques can be applied in order to collect users' information by *eavesdropping* technique using a software of *spied-phone on* [2]. Another technique is to apply the Denial-of-Convenience (DoC) by installing the counterfeit Wireless Fidelity (WIFI) to tap information [3]. Such tapping of information can be done when the smartphone connects to the WIFI. In addition, the attack can be done by breaking the encryption code through searching keywords by

Brute Force Attack (BFA) [4]. BFA is done by giving the first plaintexts of (100-(50/2n))% bit in a random and unique manner.

It shows that the secure communication using the internet phone service is necessary. One solution is to make end-to-end encryption. By the encryption technique, the data of voice are randomized before being sent and described before getting to destination address. Various cryptographic algorithms have been applied to make encryption. The selection of an appropriate cryptographic algorithm to develop mobile devices is required. The Algorithm used in this study are Tiny Encryption Algorithm (TEA) algorithm. TEA algorithm was chosen because it has the characteristics suitable for mobile devices, i.e. minimizing the memory and maximizing the speed [5]. The algorithm is inserted into the source code of Sipdroid. Sipdroid is a source code for the VoIP client in the Android operating system that is open for the developers [6]. Such insertion of TEA code is a form of the development of secure phone application in Android smartphones.

2. Literature Review

Tiny Encryption Algorithm (TEA) is a cipher algorithm that was created by David Wheeler and Roger Needham from the Computer Laboratory, Cambridge University, England in November 1994 [5]. It is a block cipher encryption algorithm that was designed to use a minimum memory with a maximum speed. Some studies on the application of TEA algorithm such as that by Hunn et al (2012) [7] proposed the development of TEA crypto-core for mobile system. Other study by Guillen et al (2009) [8] addressed security mechanism and the feasibility of TEA applied in sensor nodes for information authentication and encryption. Venugopal et al (2013) [9] conducted a study about the application of TEA algorithm in terms of latency, throughput, gate equivalence, the cost and ease of mapping in the Field Programmable Gate Arrays (FPGA) and Graphics Processing Units (GPU). Bagbaba et al (2015) [10] applied the TEA algorithm for image encryption and compression. Ge et al (2015) [11] conducted a study about the encryption of TEA algorithm for public communication network wireless remoter. The algorithm only has a maximum key length of up to 128-bit [5]. The key length is the key number that is too long for modern cryptographic algorithm. David Wheeler and Roger's purpose in applying the maximum key length of up to 128 bits was that TEA algorithm was created just for mobile devices with the computing capacity lower than personal computers.

TEA encryption system used the Feistel network by adding mathematical functions such as addition and subtraction as the inverting operator other than XOR. It is intended to create the property of nonlinearity. The shifts in both directions (left and right) made all the bit keys and data repeatedly mixed. TEA processed 64-bit input at a time and produce 64-bit output [7]. TEA stored 64-bit input into L_0 and R_0 of each 32-bit. Finally, 128-bit keys are stored into k(0), k(1), k(2), and k(3), each of which contains 32-bit. It is expected that the technique could prevent the use of exhaustive search technique effectively. The output will be stored in L_{16} and R_{16}. Constant delta number to be used is 9E3779B9 derived from the golden number $(5/4^{1/2}-1/2-0.618034)\ 2^{32}$ or delta=(√5-1) 231. Different double delta number is used in each rotation, so that no bit of multiplication did not change regularly. Different from Feistel's structures that originally operated one side only, i.e. the right side with a function F, in TEA algorithm both sides were operated with a similar function. The TEA System Encryption can be shown in Figure 1.

Figure 1. shows TEA encryption system with the process as follows [7].

1) *Shift*
   Light text block on both sides of each of 32-bit is shifted to the left as many as four times and is shifted to the right as much as five times.
2) *Addition*
   The next step after the blocks were shifted to the left and to the right is that Y and Z to be shifted were added the keys of k[0]-k[3], while the initial Y and Z were added the sum (delta).

3) *XOR*

The next process after they were operated by the addition in each of register was XOR processed by the formula for one round as shown in Equation 1 and Equation 2.

$$y = y + (((z << 4) + k[0]^2 z + sum^2 ((z >> 5) + k[1])) \quad (1)$$

$$z = z + (((y << 4) + k[2]^2 y + sum^2 ((y >> 5) + k[3])) \quad (2)$$

The results of encoding in one cycle of a light text block of 64-bit to be 64-bit ciphertext were obtained by combining Y and Z. For the encoding at the next cycle, positions of Y and Z were interchanged so that Y1 was in Z1 and Z1 was in Y1, then continued by the process with the steps above until 16 cycles (32 rounds).

4) Key Schedule
TEA algorithm used very simple key schedule, i.e. keys of k [0] and k [1] were used in odd round, while the constant keys of k [2] and k [3] were used in even round.
5) Description and Encryption
Decryption process was the same with that in other Feistel cipher-based encoding process. The principle was the same at the time of encryption process. The thing different was the use of ciphertext as input and the keys used were in a reversed order. In the entire decryption process with even round, k[3] was firstly used and then k[2].

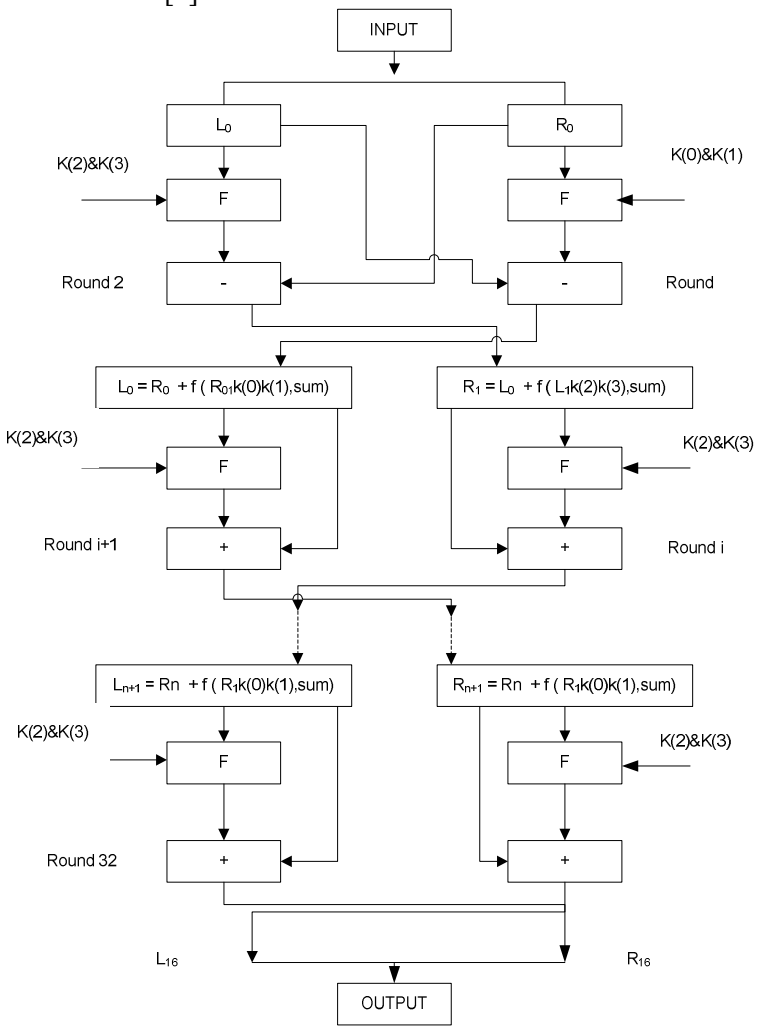

Fig. 1. Tiny Encryption Algorithm (TEA)[7]

The formulas for encryption and decryption were shown in Equations 3 and 4:

$$L_0 = L_0 + f(R_0.k[0], k[1], sum) \quad (3)$$

$$R_0 = R_0 + f(L_0.k[2], k[3], sum) \quad (4)$$

Thus, L0 is the sum of L0 added by f (R0, k[0], k[1], sum). The encryption process for one round was run by using the formula as shown in Equations 1 and 2:

$$y = y + (((z << 4 + k[0]^2 z + sum^2((z >> 5) + k[1])) \quad (5)$$

$$z = z + (((y << 4) + k[2]^2 y + sum^2((y >> 5) + k[3])) \quad (6)$$

The decryption process was run by using the formula as shown in Equations 7 and 8:

$$L_0 = L_0 + f(R_0.k[1], k[0], sum) \quad (7)$$

$$R_0 = R_0 + f(L_0.k[3], k[2], sum) \quad (8)$$

Thus, L_0 is the sum of L_0 added f (R_0, k[0], k[1], sum). The decryption process for one round was run by using the formula as shown in Equations 9 and 10:

$$y = y + (((z << 4 + k[1]^2 z + sum^2((z >> 5) + k[0])) \quad (9)$$

$$z = z + (((y << 4) + k[3]^2 y + sum^2((y >> 5) + k[2])) \quad (10)$$

The formula of Y the above explained that Y was the result of Y added Z that was shifted to the left four times with the addition of key k [1]. Then, the sum was made to be XOR where Z was summed with the sum (delta). The results of XOR process of both summations were made XOR again of shifted again with Z to be shifted to the right five times with the addition of key k [0]. Thus, the formula of Z was similar to that of Y, only the keys of k[3] and K[2] were used.

3. Method and Materials

This research was conducted by the following methods to measure the quality of services in terms of MOS and VOIP.

3.1. Mean Opinion Score (MOS)

The study was conducted using qualitative and quantitative methods to measure VoIP. Qualitative test was carried out by a survey among a number of respondents about the quality of speech, while qualitative test was carried out by using a mean opinion score (MOS) method [12].

The qualitative test was carried out to find the perception of average voice quality of a system. The test could be done by a survey among a number of respondents ask their opinion. They were asked to rate the voice quality by providing a value between 1 and 5 [12]. Based on the result, mean opinion score (MOS) can be calculated [10]. The difficulty encountered in the research was the subjectivity of different respondents, so that the quality of voice system was difficultly determined. In addition, the test required a lot of respondents, making the test expensive and time consuming [12].
The following equation was used to calculate MOS.

$$MOS = \frac{N_a + N_b + N_c + ... + N_n}{N} \quad (11)$$

$N_{a...j}$: Speech Quality Value
N : Number of Respondents

Table 1. Categories of MOS.

Value	Quality	Voice
5	Excellent	No noise
4	Good	Slightly noise
3	Fair	Lot of noise
2	Poor	Difficult to understand
1	Bad	Not understood

3.2. E-Model (ITU-T G.107)

The research also applied E-Model method to support the results of measurements by using the MOS method. In a VoIP network, the reduced level of quality due to data transmission played an important role in the resultant quality of voice. The underlying causes of the reduced level of voice quality were delay, packet loss, and echo. The mathematical approach used to determine the quality of voice based on the reduced level of voice quality in VoIP networks was modeled in E-Model in accordance with ITU-T G.107[13]. Final value of estimation in the E-model was called R factor. R factor is defined as the factor of transmission quality to be affected by parameters such as signal-to-noise ratio and echo devices, codec and compression, packet loss, and delay. R factor was defined in Equation 12.

$$R = 94.2 - I_d - I_e \tag{12}$$

I_d value was determined from the following equation.

$$4.\ I_d = 0.024d + 0.11(d - 117.3)H(d - 1773) \tag{13}$$

Meanwhile, I_e value depends on the compression method used. The overall R factor value was calculated by Equation 14.

$$R = 94.2 - [0.024d + 0.11(d - 117.3)H(d - 1773)]I_e \tag{14}$$

where

R : factor of transmission quality
D : one-way delay (ms)
H (x) : 0 if x <0, and so on
H (x) : 1 for x ≥ 0

The value of R factor refers to the standard of MOS, where the relationships can be seen in Table 2. The following rules were applied to change estimation of values of R factor into MOS.

- For R<0, MOS=1
- For R=100, MOS=4.5
- For 0<R<100, MOS=1+0.035R+7x10^{-6}R(R-60)(100-R)

Thus, a correlation between E-Model and MOS (in accordance with ITU-T) was shown in Table 2.

Table 2. Correlation Between E-Model and MOS

R-Value	Mean Opinion Score	User Satisfication
90 or higher	4.34 or higher	All users were very satisfied
80 or higher	4.03 or higher	All users were satisfied
70 or higher	3.60 or higher	Some users were dissatisfied
60 or higher	3.10 or higher	Many were dissatisfied
50 or higher	2.58 or higher	Almost all users were dissatisfied

3.3 Parameter of Quality of Service (QoS) in VoIP Network

In general, there are important parameters that affect QoS of voice services in VoIP network. The parameters are used as the measures for the performance of VoIP network [14]. The parameters were delay, jitter, packet loss, and throughput. In this research, only two parameters were used, i.e. delay and throughput. This was done because delay is a part of the jitter (variation in delay time) and throughput is a part of packet loss rate.

3.3.1 Delay

Delay (latency) is a delay time the data needed to cover the distance from origin area to destination. In designing a VoIP network, the time delay is a problem that must be taken into account because the quality of good or bad voice depends on time delay [13]. The maximum delay time to be recommended by ITU-T G.711 for voice applications was 160 ms, while the maximum delay time with the voice quality that was accepted by the user was 250 ms. Equation 15 was used to calculate the delay time.

$$Delay = \frac{Twf}{P} \tag{15}$$

Twf : Time between first and last packet
P : Total packet

3.3.2 Throughput

Throughput is the rate of effective data transfer, which is measured in bps. Header in the data packet reduces this value. It is calculated by looking at the number of incoming packets to be sent [13]. It is also the actual ability of a network for data transmission. Usually it is always associated with bandwidth, because it can be referred to as bandwidth in real conditions. Bandwidth is more definite and throughput is more dynamic, depending on the ongoing traffic. Equation 16 was used to calculate throughput.

$$Throughput = \frac{Data}{Time} \tag{16}$$

4. Result and Disscusion

The study aims at analyzing the effect of the application of TEA algorithm for a comparison of the resultant quality of conversation voices between application with TEA encryption and that without TEA by MOS method as well as delay and throughput measurement method. The study used a bandwidth of 1 Mbps. The data collection was carried out 30 times using the key length of 128-bit. Each session of communication was held in a one-way manner for 1,280 seconds.

4.1. The Results by MOS Method

MOS method was used as a qualitative test based on the quality of voices heard by the respondent. Each respondent did experiments and given values in accordance with the category of the quality of voices heard. The following was the results of measurement by MOS method.

Table 3. MOS in Application without TEA

10 Experiment	Value of MOS										MOS	RR
	A	B	C	D	E	F	G	H	I	J		
1	5	5	4	5	5	4	4	5	5	5	4.7	
2	4	5	4	5	4	4	5	4	4	4	4.3	4.4
3	4	5	5	5	5	4	4	4	4	4	4.4	

From the data in Table 3, it can be concluded that the quality of conversation voices was still high as they could be very clearly heard by respondents. This was seen from average value (RR) of MOS, i.e. 4.4. Meanwhile, the measurement by MOS in application with TEA was as follows.

Table 4. MOS in Application with TEA

10 Experiment	Value of MOS										MOS	RR
	A	B	C	D	E	F	G	H	I	J		
1	4	4	4	3	3	3	4	4	4	4	3.7	
2	4	5	3	4	4	3	5	4	3	4	3.6	3.8
3	3	3	3	5	5	4	4	3	4	4	3.8	

From the data in Table 4, it can be concluded that most respondents assessed that the application with TEA produced the sufficient good conversation, although it was rather noisy. They perceived that the noise was generated in a delay time between voice sent by the caller A and voice received by the caller B and the conversation voice was intermittently heard.

4.2. The Results by E-Model Method (ITU-T G.107)

The E-model method was used in favor of the results of measurement by the MOS method. Final estimation value in the E-model was called as R factor. R factor is defined as the factor of transmission quality to be affected by delay time. The reduced value of voice quality due to one-way delay time must be recognized before in order to find out the values of R factors. Equation 13 was used to calculate the delay time.

Table 5. Values I_d in Application Without TEA

10 Experiment	*Delay* (ms)	H(x)	I_d
1	90.64	0	2.17536
2	81.78	0	1.96272
3	60.7	0	1.45680

Meanwhile, the results of the calculation of Id value in the application with TEA value were shown in Table 6.

It was explained that Ie value is closely related to packet loss where E is a decimal value of packet loss. In this case, the rate of packet loss should be as low as possible because what to be sent is a voice packet wherein the transmission process there was no retransmission.

Table 6. Values I_e in Application without TEA

10 Experiment	*Packet Loss*	I_e
1	0.00000	7
2	0.00100	7.44664
3	0.00080	7.35781

References The results of the calculation of value I_e in the application with TEA are shown in Table 7.

Table 7. Values I_e in Application with TEA

10 Experiment	*Packet Loss*	I_e
1	0.01571	0.01571
2	0.01932	0.01932
3	0.02365	0.02365

After I_d and I_e values were obtained, the values of R factors could be obtained.

Tabel 8. Values of R Factors in Application without TEA

10 Experiment	I_d	I_e	R Factor
1	2.17536	7	85.32464
2	1.96272	7.44664	84.79064
3	1.45680	7.35781	85.38539

The values of R factors in application with TEA were shown in Table 9.

Tabel 9. Values of R Factors in Application without TEA

10 Experiment	I_d	I_e	R Factor
1	5.83848	13.3479	75.01362
2	7.44864	14.6346	72.11676
3	4.34016	16.1085	73.75334

The results of correlation between the measurements by the E-Model and the BOS method (in accordance with ITU-T) were shown in Table 10.

Tabel 10. Results of Correlation Between MOS and R Factor

10 Experiment	Application Without TEA		Application TEA	
	R Factor	MOS	R Factor	MOS
1	85.32464	4.2112	75.01362	3.8791
2	84.79064	4.1894	72.11676	3.6933
3	85.38539	4.2218	73.75334	3.7930

From the results of calculation, it can be concluded that the application without TEA had the MOS values of 4.1894 to 4.2218. According to ITU-T recommendation, the application was in a good category. Meanwhile, the application with TEA had the MOS values of 3.6933 to 3.879.

It can be clear from Table 2.4 in accordance with ITU-T recommendation, explaining that the results of the calculation of R factor in the application with TEA were in an acceptable category. Therefore, the effect of the application of the TEA algorithm showed that the application is feasible for use.

4.3. The Results of the Measurement of QoS VoIP

In general, there are important parameters that affect QoS of voice services in VoIP network. The parameters are used as the measures for the performance of VoIP network. The parameters were delay, jitter, packet loss, and throughput. In this research, only two parameters were used, i.e. delay and throughput. This was done because delay is a part of the jitter (variation in delay time) and throughput is a part of packet loss rate.

4.3.1 Delay

Delay is a time the data needed to arrive at the destination. The amount of delay can be measured at the time of the first bit transmitted until the last bit received at the receiver side.

Table 11. Delay

10 Experiment	***Delay* (s) App Without TEA**	***Delay* (s) App TEA**
1	0.009465105	0.024327
2	0.008178191	0.031036
3	0.006071458	0.018084

The resultant average delay as shown in Table XI indicates that the application with TEA had delay time greater than that without TEA. This proved that the delay was caused by the application of TEA algorithm. The encryption led to long data due to the addition of keys and XOR process. Thus, the delay was larger. However, the delay resulted from the applications with TEA was averagely ±0.03 seconds or, if converted, 30 millisecond, indicating that it met the standard of ITU-T. The standard of ITU-T categorizes that the delay of 0-150 milliseconds had still been acceptable for users.

4.3.2 Throughput

Throughput is the effective rate of data transfer measured in bps. It also refers to the amount of data that can be sent in a time unit. It is highly dependent on availability of bandwidth in the network. The following is throughputs generated from the applications with and without TEA.

Table 12. Throughput

10 Experiment	***Throughput* (kbps) App Without TEA**	***Throughput* (kbps) App TEA**
1	481.979	53.74
2	457.600	169.011
3	374.487	126.173

Table 12 shows that the throughput generated from the application with TEA has throughputs smaller than that from the application without TEA. Such process occurred because the formation of packet in the application with TEA had bits larger than the usual bits. This was due to encryption process that changed the length of the packet.

5. Conclusion

From the results of analysis, measurement, calculation, and comparison between the applications with and without TEA, it can be concluded as follows:

1 The application of TEA algorithm caused the lower values of R factors (from 72.11676 to 75.01362) compared to that without TEA (from 84.79064 to 85.38539). However, in accordance with the standard of ITU-T, the values of R factor were included into to an "acceptable" category. Thus, the application was feasible to use.

2 The effect of the application of TEA algorithm resulted in the delay rates from 0.018084 to 0.024327 seconds higher than that of the application without TEA resulting in the delay rates from 0.006071458 to 0.009465105 seconds. It proved that the application of TEA algorithm led to long data due to the addition of keys and XOR process. Thus, the delay was larger. However, according to the standard of ITU-T, the delays resulted in the application with TEA were in an "acceptable" category.

3 The effect of the application of TEA algorithm led to lower values of throughput, ranging from 53 to 126 kbps. This occurred due to the formation of packet with larger than usual bits. This is because the encryption process occurred changed the length of packet. The results proved that the application of TEA affected the quality of VoIP.

Reference

[1] H. P. Singh, S. Singh, J. Singh, and S. A. Khan, "VoIP: State of art for global connectivity - A critical review," *J. Netw. Comput. Appl.*, vol. 37, no. 1, pp. 365–379, 2014.

[2] Yi-Bing Lin and Meng-Hsun Tsai, "Eavesdropping Through Mobile Phone," *Veh. Technol. IEEE Trans.*, vol. 56, no. 6, pp. 3596–3600, Nov. 2007.

[3] E. Dondyk, L. Rivera, and C. C. Zou, "Wi-Fi access denial of service attack to smartphones," *Int. J. Secure. Networks*, vol. 8, no. 3, pp. 117–129, 2013.

[4] C. Li, S. Li, D. Zhang, and G. Chen, "Cryptanalysis of a data security protection scheme for VoIP," *Vision, Image Signal Process. IEE Proc.* -, vol. 153, no. 1, pp. 1–10, Feb. 2006.

[5] S. J. Shepherd, "The Tiny Encryption Algorithm," *Cryptologia*, vol. 31, no. 3, pp. 233–245, 2007.

[6] A. Wahab, R. B. Bahaweres, M. Alaydrus, M. Muhaemin, and R. Sarno, "Performance analysis of VoIP client with integrated encryption module," in *Communications, Signal Processing, and their Applications (ICCSPA), 2013 1st International Conference on*, 2013, pp. 1–6.

[7] S. A. Y. Hunn, S. Z. binti Md Naziri, and N. binti Idris, "The development of tiny encryption algorithm (TEA) crypto-core for mobile systems," 2012, pp. 45–49.

[8] B. K. Mishra, M. C. Nikam, and P. Lakkadwala, "Feasibility of TEA in wireless sensor networks," in *IT in Business, Industry and Government (CSIBIG), 2014 Conference on*, 2014, pp. 1–6.

[9] V. Venugopal and D. M. Shila, "High throughput implementations of cryptography algorithms on GPU and FPGA," in *Instrumentation and Measurement Technology Conference (I2MTC), 2013 IEEE International*, 2013, pp. 723–727.

[10] A. C. Bagbaba, B. Ors, O. S. Kayhan, and A. T. Erozan, "JPEG image Encryption via TEA algorithm," in *Signal Processing and Communications Applications Conference (SIU), 2015 23th*, 2015, pp. 2090–2093.

[11] Y. T. Ge, X. M. Liu, and X. T. Yin, "Study on TEA Encryption for Public Communication Network Wireless Remoter," in *Applied Mechanics and Materials*, 2014, vol. 565, pp. 179–182.

[12] A. Lakaniemi, J. Rosti, and V. Räisänen, "Subjective VoIP speech quality evaluation based on network measurements," in *Communications, 2001. ICC 2001. IEEE International Conference on*, 2001, vol. 3, pp. 748–752.
[13] E. P. Guillen and D. A. Chacon, "VoIP Networks Performance Analysis with Encryption Systems," vol. 16, no. 2, pp. 688–695, 2009.