Scientific Research and Essays Vol. 5(9), pp. 856-861, 4 May, 2010
Available online at http://www.academicjournals.org/SRE
ISSN 1992-2248 © 2010 Academic Journals

Full Length Research Paper

Performance evaluation of scalable encryption algorithm for wireless sensor networks

Murat Çakıroğlu*, Cüneyt Bayilmiş, Ahmet Turan Özcerit and Özdemir Çetin

Department of Computer Engineering, Faculty of Technology, University of Sakarya, Turkey.

Accepted 10 March, 2010

Developing effective security solutions for wireless sensor networks (WSN) are not easy due to limited supplies of WSNs and the hazardous nature of wireless medium. The implementation of encryption/decryption algorithms, which are the most essential part of the secure communication, can be very intricate in WSNs since, they incorporate routines, having very complex and intense computing procedures. Therefore, WSNs must be designed in such a way that a compromise should be established by balancing between the security level and the processor overhead. The aim of this paper is to investigate the suitability of the Scalable Encryption Algorithm (SEA) in use for a secure communication in WSNs. In order to confirm the effectiveness of SEA, a comparative performance evaluation with AES and RC6 algorithms are presented in terms of memory requirement, execution time, and bandwidth criteria. According to the results obtained from the target development platform, SEA has better performance than AES in respect of memory requirements and bandwidth, on the other hand, it has surpassed RC6 algorithm in terms of execution time and total memory requirement. As a result, SEA can be a strong alternative block cipher for WSNs.

Key words: Wireless sensor network (WSN), block cipher, data security, SEA, AES, RC6.

INTRODUCTION

WSNs consist of sensor nodes, which are low cost and small-embedded systems with limited capacity and processing capability, communicate with each other in ad-hoc manner (Akyildiz et al., 2002). WSNs have been usually utilized in many areas including military, health, plant automation and agricultural applications etc. The applications mentioned above have roughly similar *scenarios*: sensor nodes collect data from deployment environment and then it forwards collected data over non-secure wireless medium to the control center either directly or through base station. Because of the nature of non-secure wireless medium, WSNs are vulnerable against adversary attacks. These vulnerabilities cannot certainly be tolerated for security-critic applications including health-care, battlefield surveillance etc. (Bandirmali et al., 2009; Bayilmis and Cakiroglu, 2008; Perrig et al., 2002).

The encryption methods (WEP, WPA, WPA2 etc.) used in conventional wireless networks cannot provide sufficientsolutions for WSNs due to large memory requirements and excessive computations (Perrig et al., 2002). In addition, the processing and messaging over-head along with increased energy consumption are other drawbacks of the conventional encryption/decryption algorithms. The encryption algorithms used in WSNs must meet the expected security requirements together with minimum processing overhead and memory size. Considering the hardware constraints and security requirements of WSNs, Scalable Encryption Algorithm (SEA) can be utilized because of following features.

*Corresponding author. E-mail: muratc@sakarya.edu.tr.

(i) Developed for processors having limited instruction set and lower processing capability
(ii) Lower data memory size and register length in the implementation stage compared to most of the other block encryption algorithms
(iii) Lower code size compared to most of the other block encryption algorithms in equivalent platforms
(iv) Robust against linear/differential cryptanalysis techniques despite basic encryption routines
(v) Parametric key length and plaintext size (48-bit, 96-bit, 144-bit, etc.) in case of different processor used (Mace et al., 2007; Standaert et al., 2006).

The paper deals with the security features of WSNs. The details of SEA and compared encryption algorithms (Advanced Encryption Standard-AES, RC6) are given in this paper. The implementation details of the experimental setup are explained also. The performance evaluation of SEA, AES, and RC6 encryption algorithms are presented and introduce discussion and conclusions of the research work.

SECURITY IN WSNs

WSNs are special networks in terms of the number and type of constraints compared to traditional networks. Due to limited hardware resources, non-secure communication channel, and unattended operation complicate the use of conventional methods in WSNs. In addition, WSNs have special security requirements, for example, data confidentiality, data integrity, data freshness, authentication, self-organization, time synchronization, and secure localization (Bandirmali et al., 2009; Bayilmis and Cakiroglu, 2008; Perrig et al., 2002).

Encryption algorithms are used in order to meet some security requirements of WSNs such as data confidentiality and authentication. The protocols in the literature use various encryption algorithms: SPIN (Perrig et al., 2002) and INSENS (Deng et al., 2002) protocols use RC5, TinySec security packet uses Skipjack algorithm (Karlof et al., 2004), and 802.15.4 standard uses AES algorithm (Daemen and Rijmen, 2002).

In the literature, there are several researches on performance evaluation of block cipher algorithms for WSNs. For instance, Law et al. (2006) studied RC5, RC6, AES, MISTY1, KASUMI and Camellia encryption algorithms to evaluate their use in WSNs (Law et. al., 2006). In another study conducted by Guimarães et al. (2005), they evaluated the performance of Skipjack, RC5, RC6, DES, and TEA algorithms (Guimarães et al., 2005).

In another example, the help of TOSSIM simulation environment (Koo Woo et al., 2008) evaluated HIGHT, RC5, and Skipjack algorithms. In this paper, since we have focused on the use of SEA algorithm on WSNs, SEA and its corresponding alternative algorithms namely, AES and RC6 (Rivest et al., 1998), which are used in comparison analysis are explained briefly in the next section.

PRELIMINARY

In this section, we introduce the algorithms that we have used in this research work. We have selected AES and RC6 algorithms to be compared with SEA because of following reasons: (i) they are widely used in WSNs, (ii) they maintain sufficient security in most cases, and (iii) they are parametric encryption algorithm that is, supporting variable key and plaintext size.

The scalable encryption algorithm

SEA was designed for low cost embedded environments with limited resources (memory size, processor capacity) in 2006 by François-Xavier Standaert et al. 2007; Mace et al., 2007; Standaert et al., 2006. The design criteria of SEA, which is based on symmetric block cipher approach, are small memory size, small code size, and limited instruction set. To meet design criteria given, SEA uses basic bit operations such as XOR, bit/word rotations, modular addition, and s-box.

SEA, which is defined as SEA (n,b), has very flexible structure. It can operate on different plaintext and key sizes. In addition, SEA has Feistel structure with variable number of rounds. It is defined by following parameters (Standaert et al., 2006):

(i) n: plaintext and key size
(ii) b: word size
(iii) $n_b = \frac{n}{2b}$: number of word per Feistel branch
(iv) n_r: number of rounds

In the implementation of SEA, n and b parameters can be configured in respect of target processor attributes. However, the bit size of the key and plaintext must be in multiple of six such as 48, 96... 192 and so on. Another crucial point is that to meet an acceptable security level and the word size must provide the following conditions (Standaert et al., 2006):

b ≥ 8 and

$$n_r = \frac{3n}{4} + 2.(n_b + [b/2]) \quad (1)$$

The advanced encryption algorithm

AES was developed by Daemen and Rijmen (2002) and introduced by National Institute of Standards and Technology in 2001. This algorithm is also known as Rijndael algorithm and it is a symmetric block encryption algorithm, which is the successor of the Data Encryption Standard (DES) algorithm.

AES can encrypt 128-bit data blocks by using 128, 192 or 256-bit keys. AES operates on 4 x 4 matrixes, which are called as states, and each AES round is composed of four stages. AES performs encryption operations in 10, 12, or 14 rounds depending on predetermined key sizes (Daemen and Rijmen, 2002). AES, which can be implemented by hardware or software based methods effectively, is used in 802.15.4 standard and it is still accepted as the most secure block cipher.

Table 1. Implementation parameters of the encryption algorithms.

Algorithm	Plaintext and key size	No. of rounds
SEA	144	134
AES	128	10
RC6	128	20

RC6

RC6 is a block cipher proposed by Rivest et al. (1998) based on RC5 symmetric key approach. RC6, like AES, can encrypt 128-bit data blocks by using 128, 192, or 256-bit keys. RC6 can support various word/key sizes and number of rounds and it can be defined as RC6-w/r/b where w stands for bit size of word, r stands for the number of rounds, and b stands for key size in bytes. The most fundamental difference between RC6 and RC5 is that RC6 uses an extra multiplication operation to perform bit rotations in each word.

IMPLEMENTATION DETAILS

All algorithms presented in this paper used for performance evaluation have been implemented in C programming language. The implementation details of SEA (144, 8), AES–128, and RC6-32/20/16 algorithms are given in Table 1. While plaintext size was chosen as 128 bits for AES and RC6, 144 bits is determined for SEA.

The reason behind this arrangement is that plaintext sizes must be multiple of six in SEA. Therefore, the nearest plausible plaintext size for the combination of 128-bit plaintext and 8-bit word size is 144 bits for SEA. The numbers of rounds have been determined separately for each algorithm. For example, in case of SEA algorithm with a reasonable secure operation, the number of rounds is calculated in Equation 1. As for AES algorithm, the number of rounds depends on the size of plaintext and key size and AES uses 10 rounds for 128-bit plaintext and key sizes. The number of rounds in RC6 can be adjustable to desired security level and it was suggested as 20 rounds in acceptable level (Rivest et al., 1998).

Encryption and decryption procedures are mainly related to processor-oriented operations, and each sensor node in WSNs has a built-in microcontroller that can be used for this objective. For example, the MICAz (MICAz, 2009), which is a widely used sensor node in WSNs, has an integrated ATMEGA128L microcontroller from AVR family having 128 KB code memory and 4 KB data memory with the clock speed of 16 MHZ.

PERFORMANCE EVALUATION OF THE ALGORITHMS

In this study, we have used AVR Studio (Atmel, 2010) software platform accompanied with AVR GCC (AVR GCC, 2010), a popular c compiler for AVR micro-controllers, to evaluate the performance of the encryption algorithms. In WSNs, sensor nodes mainly use TinyOS operating system and all configurations and embedded programming implemented by a C-based development environment. Therefore, we have selected and used C programming language as a test language.

The MICAz sensor nodes have been used only and thoroughly as a reference platform in all performance measurements. In the performance evaluation of the algorithms, we have considered four parameters namely, memory requirement, execution time, bandwidth, and security level of encryption algorithms.

Memory requirements

The minimum data and code memory sizes of each algorithm for encryption/ decryption procedures obtained from AVR Studio are shown in Figure 1. SEA requires approximately one third of code memory in respect of AES. On the other hand, SEA requires 1.6 times less code memory with regard to RC6 algorithm. In addition, while SEA needs 43 times less data memory referring to AES; it requires 2.2 times more data memory with respect to RC6. The small size of code and data memory requirements of SEA is originated from the fact that its algorithm is based on basic bit operations.

Execution time and bandwidth

Execution times of each algorithm are presented in Figure 2 for both encryption and decryption (e/d) stages. While the execution time of SEA is much higher than AES, it is lower than RC6 for both e/d stages. This is certainly a serious drawback for both SEA and RC6 algorithms and this disadvantage is originated from the combination of three factors: key/plaintext size, the structure of rule tables used, and the number of rounds. For example, while SEA uses 144-bit key and plaintext, AES and RC6 use 128-bit key and plaintext.

The bandwidths of each algorithm are given in Figure 3. While AES can provide about 3.3 times higher bandwidth compared to SEA, SEA can deliver almost 1.3 times higher bandwidth than RC6 can.

Security

AES (Daemen and Rijmen, 2002), which is also used in 802.15.4 standard, is still most secure block cipher used in many applications. Law et al. (2006) proved that Rijndael (AES) is the most appropriate block cipher algorithm by means of high security and energy efficiency for WSNs. There is no known practical attack against full version of AES (Biryukov and Khovratovich, 2009).

There are several types of attacks designed against reduced version of RC6 in the literature. For example, Hinoue et al. (2007) suggested statistical attacks against versions of RC6 with 12 and 16 rounds (Hinoue et al., 2007). However, all attacks against RC6 in the literature are only for reduced versions. Therefore, RC6 with 20 rounds can be accepted as secure.

So far, there has been no known an attack for SEA in the literature. In addition, SEA is robust against linear/differential cryptanalysis techniques (Mace et al., 2007; Standaert et al., 2006; Bayilmis and Cakiroglu, 2008). As a result, all selected algorithms can maintain a security level required by WSNs.

RESULTS AND FUTURE WORKS

In this paper, we have implemented SEA algorithm for use in WSNs and the performance evaluation of this algorithm has been investigated by comparing with two alternative popular algorithms: AES and RC6. We have selected three crucial parameters to compare each algorithm's performance namely; memory requirements,

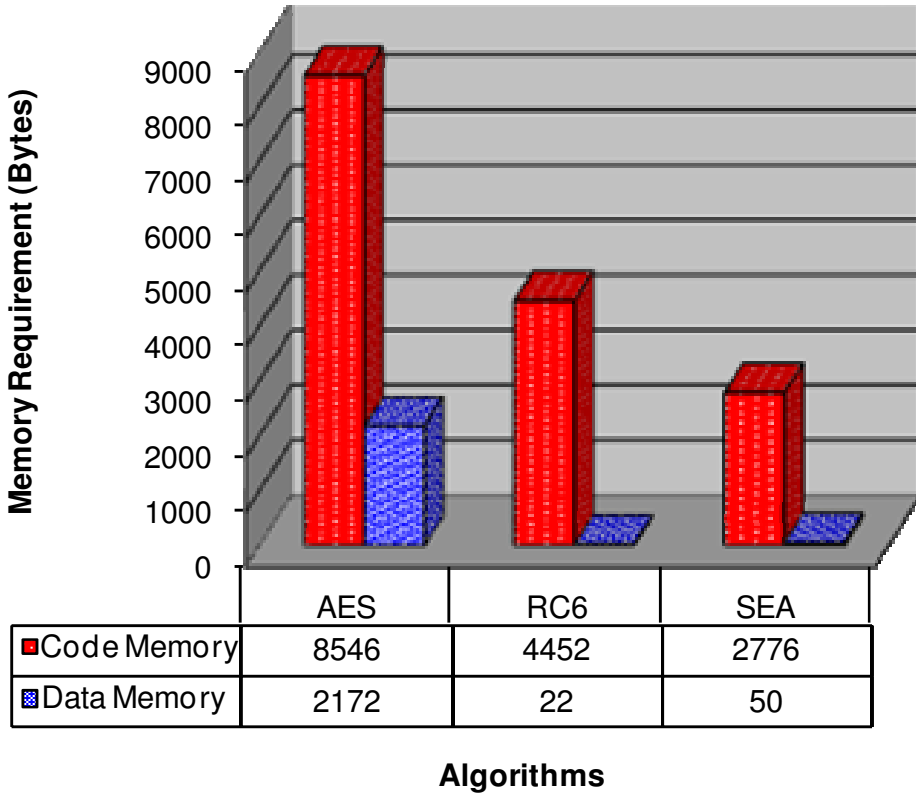

Figure 1. Memory requirements of each algorithm.

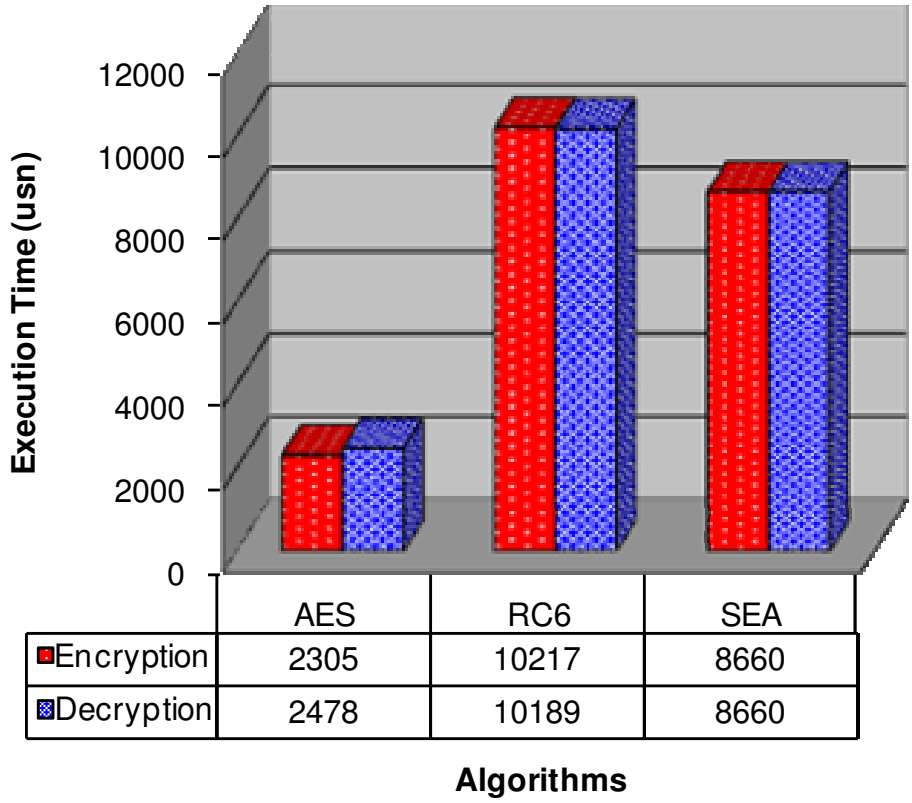

Figure 2. Execution times for each algorithm.

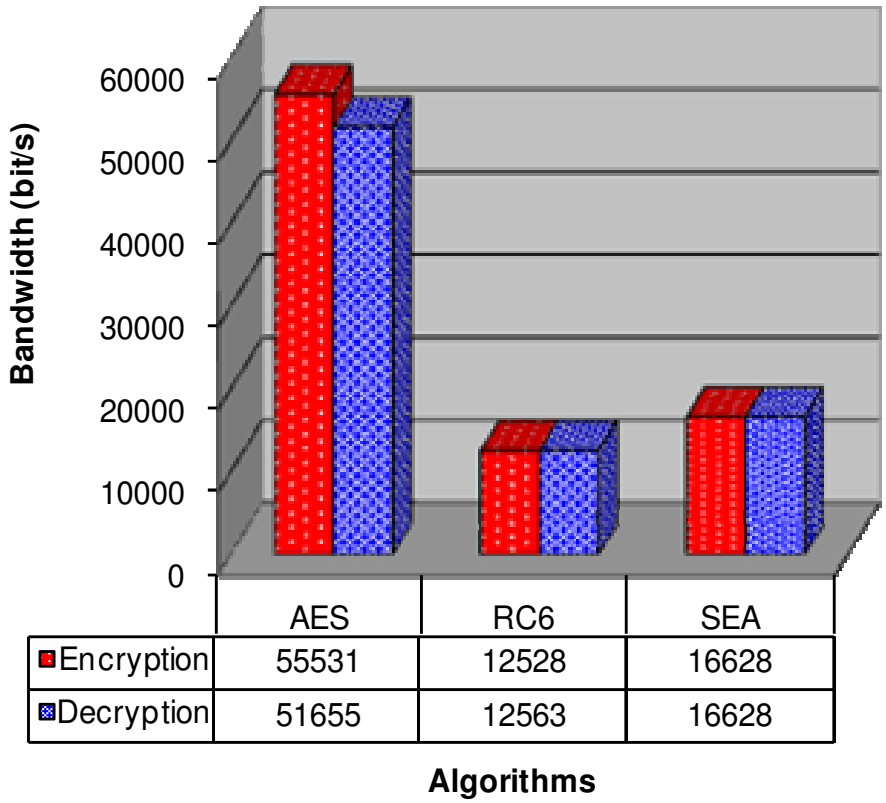

Figure 3. Supplied bandwidth of each algorithm.

execution time, and bandwidth. According to results, SEA requires less code memory size compared to both alternative algorithms. As for data memory, SEA needs much less memory size than AES; however, RC6 requires less memory size than SEA. Besides, although SEA was surpassed by AES, it has better performance compared to RC6 in terms of execution time and bandwidth. In addition, since, SEA can be configured to support many type of processors that have different word size (8-bit, 16-bit, and 32-bit) and can operate on adaptable size of key and plaintext, therefore, this algorithm can be used readily in many types of WSNs. As future works, SEA and other encryption algorithms should be implemented on real MicaZ sensor nodes to compare the simulation and real system performance.

REFERENCES

Akyildiz IF, Su W, Sankarasubramaniam Y, Cayirci E (2002). Wireless sensor networks: survey, Comput. Networks 38: 393-422.

Atmel Corporation. AVR Studio 4.18, Last visited: January 2010, Available: http://www.atmel.com/dyn/products/tools_card.asp?tool_id=2725.

AVR-GCC (2010). Free AVR C Compiler, Last visited: January, Available: http://sourceforge.net/projects/winavr/files/

Bandirmali N, Erturk I, Ceken C (2009). Securing Data Transfer in Delay-Sensitive and Energy-Aware WSNs using the Scalable Encryption Algorithm, Proceedings of international symposium on wireless and pervasive computing held at Melbourne, Australia. pp. 1-6.

Bayimiş C, Çakiroğlu M (2008). SEA şifreleme algoritması kullanarak güvenli kablosuz algılayıcı ağ haberleşmesinin gerçekleştirilmesi, Proceeding of 3rd international information security and cryptology conference held at Ankara, Türkiye.

Biryukov A, Khovratovich D (2009). Related-key Cryptanalysis of the Full AES-192 and AES-256, in Advances in Cryptology – ASIACRYPT. 59(12): 1-18.

Daemen J, Rijmen V (2002). The Design of Rijndael: AES - The Advanced Encryption Standard, Springer-Verlag. 2002, ISBN 3-540-42580-2.

Deng J, Han R, Mishra S (2002). INSENS: Intrusion-tolerant routing in wireless Sensor Networks, Technical Report CU CS-939-02, Department of Computer Science, University of Colorado, USA.

Guimarães G, Souto E, Sadok D, Kelner J (2005). Evaluation of security mechanisms in wireless sensor networks, ICW. pp. 428-433.

Hinoue T, Miyaji A, Wada T (2007). The security of RC6 against Asymmetric Chi-square Test Attack, IPSJ J. 48(9): 1-10.

Karlof C, Sastry N, Wagner D (2004). TinySec: a link layer security architecture for wireless sensor networks. Proceedings of the 2nd international conference on embedded networked sensor systems held at Baltimore, MD, USA, pp. 162-175

Koo Woo K, Lee H, Kim Yong H, Lee Dong H (2008). Implementation and Analysis of New Lightweight Cryptographic Algorithm Suitable for Wireless Sensor Networks, Information Security and Assurance, ISA, pp. 73-76.

Law YW, Doumen J, Hartel P (2006). Survey and benchmark of block ciphers for wireless sensor networks. ACM Trans. Sensor Networks 2(1): 65-93.

Mace F, Standaert FX, Quisquater JJ (2007). FPGA implementation(s)

of a Scalable Encryption Algorithm, IEEE Trans. Very Large Scale Integ. (VLSI) Syst. 16(2): 212-216.
Mica Z (2009). Data Sheet, Last visited: December 2009, Available: www.xbow.com/Products/Product_pdf_files/Wireless_pdf/MICAZ_Datasheet.pdf
Perrig A, Szewczyk R, Tygar JD, Wen V, Culler D (2002). SPINS: Security protocols for sensor networks. Wirel. Netw. 8: 521-534.
Rivest RL, Robshaw MJB, Sidney R, Yin YL (1998). The RC6 Block Cipher. Proceedings of First Advanced Encryption Standard (AES) Conference held in Ventura, CA, USA.
Standaert FX, Piret G, Gershenfeld N, Quisquater JJ (2006). SEA: A Scalable Encryption Algorithm for small-embedded applications. Lect. Notes Comput. Sci. 3928: 222-236.