International Journal of Electrical, Electronic and Communication Sciences
ISSN: 2517-9438
Vol:5, No:9, 2011

AES and ECC Mixed for ZigBee Wireless Sensor Security

Saif Al-alak, Zuriati Ahmed, Azizol Abdullah and Shamala Subramiam

***Abstract*—**In this paper, we argue the security protocols of ZigBee wireless sensor network in MAC layer. AES 128-bit encryption algorithm in CCM* mode is secure transferred data; however, AES's secret key will be break within nearest future. Efficient public key algorithm, ECC has been mixed with AES to rescue the ZigBee wireless sensor from cipher text and replay attack. Also, the proposed protocol can parallelize the integrity function to increase system performance.

***Keywords*—**AES, ECC, Multi-level security, ZigBee

I. Introduction

ZIGBEE technology is a low rate, low-power and low cost wireless sensor network that built over IEEE 802.15.4 standard [1]. ZigBee wireless sensor network can be utilizes in telecom services, home energy saving, health care and many other fields [2]. ZigBee technology is employing different frequency channels up to 17. The ZigBee technology uses Carrier Sense Multiple Access\Collision Avoidance (CSMA\CA) protocol to scan the free frequency channel [3]. The battery life time of ZigBee sensor is longer than normal one with around two years. ZigBee sensor has two modes which are Active and sleep and it can change from one state to other state fast [4].

ZigBee employs Advanced Encryption Standard (AES) algorithm and Counter CBC-MAC (CCM*) mode to provide defense against attack. Different types of attack can be compromise exchanged message for example attack on secret key of encryption. IEEE 802.15.4 provides message confidentiality for ZigBee via media access control (MAC) layer, which runs AES 16-octet block and key size 128-bit [5]. The analyst has to try 2^{128} combination key to detect the correct key, or to do 2^{64} steps of operation [6] where the time complexity to break it can be computed as O ($log_2 64$). In ZigBee technology, the AES algorithm applies static number of blocks with the same sequence of sub keys. The analyst can break system confidentiality by attacking the cipher text [7].

Saif Al-alak is with the Department of Computer Science, College of Science for Women, Babylon University, Babylon, Iraq (saif.shareefy@gmail.com).

Zuriati Ahmed is with Department of Communication Technology and Networks, Faculty of Computer Science and Information Technology, University Putra Malaysia, 43400 Serdang, Selangor (zuriati@fsktm.upm.edu.my).

Azizol Abdullah is with Department of Communication Technology and Networks, Faculty of Computer Science and Information Technology, University Putra Malaysia, 43400 Serdang, Selangor (azizol@fsktm.upm.edu.my).

Shamala Subramiam is with Department of Communication Technology and Networks, Faculty of Computer Science and Information Technology, University Putra Malaysia, 43400 Serdang, Selangor (shamala@fsktm.upm.edu.my).

II. Literature Review

The ZigBee wireless sensor network confidentiality can be improved by using robust encryption/decryption algorithm, for example using asymmetric algorithm instead of symmetric algorithm [1]. The practical result of comparison between Rivest, Shamir and Adleman (RSA) and Elliptic Curve Cryptography (ECC), two popular public key algorithms, for energy cost of authentication and key exchange, refers to that ECC is more optimize than RSA for computing time and amount of data transfer. The size of key for ECC is shorter than RSA, where ECC's key size equal to 160 bit is equivalent to RSA's key size equal to 1024 bit in terms of security. [8] In conclusion, ECC is more appropriate for ZigBee requirements and applications [9], in addition to it is more efficiency and less cost than RSA [10] but, ECC is not the accurate algorithm for power saving. When ECC runs over ZigBee wireless sensor network, it will consume more power than symmetric algorithm [11].Many researchers are worked to reduce the system's power consumption by developing different security protocols and chipset. In ZigBee wireless sensor network, an AES coprocessor is designed to reduce the power consumption and to raise the system performance, whose MAC layer executes AES encryption/decryption. The architecture of encryption/decryption integrate is optimized together via resource sharing fashion, where byte substitution (in encryption operation) and inverse byte substitution (in decryption operation) operations someway have similarity in their designing that allow both of them to share the same resource. As well as, mix column (in encryption operation) and inverse mix column (in decryption operation) operations have the same property of resource sharing [12].Elliptic curve algorithm is utilized to build secure and efficient identification protocol for multi-hop wireless sensor ZigBee, which is combine symmetric and asymmetric crypto technology to manage key. The identification protocol helps to transfer the master key over ZigBee network. The security of identification protocol is inherited the difficulty of solving the Elliptic curve discrete algorithm problem and the Elliptic curve computational Diffie-Hellman problem [13].System performance can be trade to security via a multi-level structure security, which is introduced to ZigBee wireless sensor network in terms of key length. The variety of security requirements of different applications leads to possibility of

classifying security to multiple levels according to their security need. ZigBee sensors are divided to groups based on their role, where each group has its own level of security that is based on sensors role. The key length affects the network bandwidth, communication, and computing cost, where the reduction of key length increases the system performance [14].

III. ZIGBEE TECHNOLOGY

A. ZigBee Structure

ZigBee is a wireless sensor network, whose node consists of four layers, with low power and low cost. The ZigBee layers are: physical (PHY) which is provide the basic capabilities of the physical radio, and media access control (MAC) layer, which is manage single-hop communication link between two devices, both of them are come from IEEE 802.15.4 standard, then network (NWK) layer which is added for packet routing and address management, and at a top is application (APL) layer that is responsible for define the node's role in the network, and establish and maintenance a secure link between nodes [5], [7], [15].

B. Zigbee Security

Security protocols of ZigBee wireless sensor are used to provide confidentiality and integrity, as well as defense against replay attack. Furthermore, AES algorithm provides confidentiality, CCM* mode is apply for integrity, and frame nonce is checked to prevent replay attack. There are three keys in ZigBee technology master key, network key, and link key. Each one of the keys has specified benefit, where link key is used for encrypting exchanged messages, network key is used to secure broadcast messages to all or group of nodes, and master key is used for transfer link key between nodes.

The ZigBee technology runs the AES 128-bit algorithm in MAC layer to ensure system confidentiality. The block cipher symmetric algorithm AES, which is published by National Institute of Standards and Technology (NIST) in 1997 as a secure and efficient cipher system, is encrypting the messages as blocks of 128-bit and the key size is 128, 192, or 256 bit and the number of round is 10, 12, and 14 in sequence [16]. However, although now many applications employs AES algorithm to protect their system from data disclosure, NIST's researchers predict the possibility of breaking AES 128-bit encryption algorithm in 2036 because of rapid development of computing technology [1], [5].

ZigBee technology apply CCM* mode to provide confidentiality, integrity and defense against replay attack; which is a developed version of CCM. CCM* provides ZigBee technology the ability to perform one of the following operation or both: encrypt the message with counter (CRT), and authenticate CBC-MAC the message. With this mode the maximum message size is up to 2^{64} byte. The authentication field T for the message with assumption that no additional authentication data is compute by the follow steps (see Fig. 1 message authentication):

- Define sequence of 16-octet blocks.
- Split the message m to 16-octet blocks $B_0, B_1, B_2, \dots, B_n$ and then padding the last block with zeros if necessary. (If message m is empty string, then blocks will not be added in this step).
- Compute:

$$X_1 = \mathrm{E}(K, B_0) \quad (1)$$

$$\mathrm{X_{i+1}} = \mathrm{E}(K, \mathrm{X_i} \oplus \mathrm{B_i}) \quad \text{for i} = 1, \dots, \mathrm{n} \quad (2)$$

$$T = \text{first\_M\_bytes}(\mathrm{X_{n+1}}) \quad (3)$$

Where

T: is the authentication code, which is truncated from $\mathrm{X_{n+1}}$.

K: is the block cipher key for (key size is 128 bit for AES in ZigBee technology).

M: is the size of authentication field.

E(): is the block cipher encryption function.

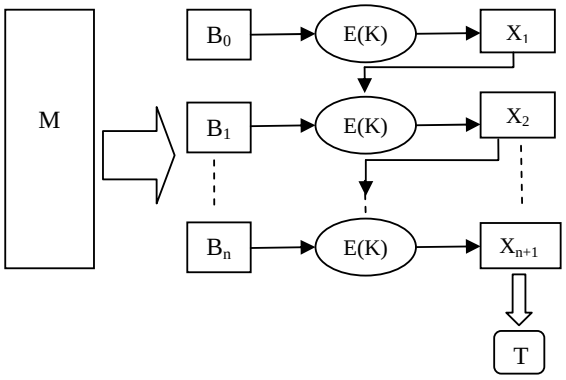

Fig. 1 Message Authentication in CCM* mode

For encrypting data with CCM*, CTR mode is applied, and can be summarized as following (see Fig. 2 Message Encryption):

- Define key stream

$$\mathrm{S_i} = \mathrm{E}(\mathrm{K}, \mathrm{A_i}) \quad \text{for i} = 0, 1, 2 \dots \quad (4)$$

Where Ai is 16-octet and its format is: flags (0 octet), Nonce N (1...15-L octet), Counter i (16-L…15). Symbol L represent the size of message length field. The flag format is: bits 0 to 3 to hold L value, bits 4 to 5 are set to zero to distinct all A blocks from B blocks in authentication, bits 6 and 7 are reserved for future used so that they set to zero.

- Encrypt the message m by XOR it with first l (m) octets of the concatenation of $\mathrm{S_1}, \mathrm{S_2}, \mathrm{S_3}, \dots$. l (m) stands for the length of the message.

However, with CCM* mode, many threats could weak the system security defense which must be enforced to keep the system safety. For AES block cipher, the pre-computation attack on secret key is possibly leads to expose the key, where for any block cipher algorithm, its strength computed as $2^{n/2}$ where n is a key size. Moreover, for AES 128 bit, the analyst needs to try 2^{64} different keys to expose the real key. One possible solution to avoid this type of attack is to strength the secret key via enlarging its size [6]. In contrast, when the length of the key is increased then the system performance is decreased. There is a tradeoff between the key size and system performance, where the number of round key is related to key size as well as run time and power consumption. The second threat is that the impossibility of ensure defense against replay attack. The

number of octets assigned to nonce is effective by the message size (number of blocks), where the octets assigned to counter is limited to the number of blocks. So that it is difficult to find distinct nonce for each message and this will weak the system against replay attack.

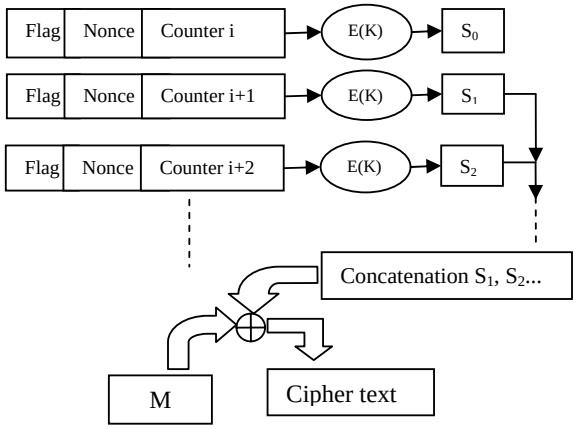

Fig. 2 Message Encryption

To provide a possible solution to remove the weakness of block cipher key encrypting algorithm, it could support the algorithm with multiple keys. Moreover, the system can encrypt each block of data with one key, where each block is encrypted with different key. For strength of security system, there will not be two plaintexts encrypted with the same cipher key, which will reduce the analyst chance to attack the cipher text. In other words, the message will be divided to sub-message and each one has its own individual key, where each sub-message is one block or more, as showing in (5), where the message M will be send from sender node to a receiver node. In sender, M is divided to n blocks with 16-octet size.

$$M = \sum_{i=1}^{i=n} B_i \qquad (5)$$

The difficulty of the proposed protocol is how to generate multiple keys and manage their secrecy in reasonable sense. It is not a practical to send all the secret keys over the network, so that it needs to transfer only initial values over network, where the keys are computed from two initial values (initial keys) as it will be show later. For any link key, it is computed from predecessor and successor secret key, so that each key is protected from attack by an ECC public key algorithm.

IV. Elliptic Curve Cryptography

A. Why Elliptic Curve Cryptography

ECC algorithm is used for protecting secret-key of the proposed protocol, which is highest secure public key algorithm. Moreover, according to the mathematical problem on which the public key algorithm based, three different systems can be find, Integer Factorization system (RSA, Rabin-William), Discrete Logarithm system (ElGamal, DSA), and Elliptic Curve Discrete Logarithm system (ECC). Security and efficiency are the main points for comparison to select the best public key system among three families. The security of the system is tied to robustness of the mathematical problem, where each one has known way to solve it, and as known to everyone ECC can be solved by fully exponential rather than sub-exponential for other public key systems, so that ECC is considered the highest security. In addition, it needs smaller key size than other systems and that refers to less memory size, processing time, bandwidth, and power consumption are needed [17].

V. Multiple Key Protocol

A. How Multiple Key Protocol Work

The proposed protocol tries to enforce the security system of ZigBee wireless sensor network in term of confidentiality, and integrity as well as defense against replay attack. ZigBee technology employs AES 128-bit algorithm and CCM* mode to improve its secrecy in MAC layer. The main threat is that 128-bit secret key of AES algorithm will be attacked in nearest future; where as the rising of key size is not efficient solution for low rate wireless sensor. Multiple-key protocol (MKP) is used ECC algorithm to protect the keys, where each key is computed from predecessor and successor keys.

In MKP a key size of AES algorithm is 128-bit, but the number of keys is different and trade off to level of security. From (5) the message is dividing to n 128-bit-blocks, and each block (sub message) is encrypted with 128-bit key K_i. The maximum number of secret keys is equal to number of blocks. The link keys must be distinct within each level, where $K_1, K_2, \ldots K_n$ are secret keys, for any secret key K_i

$$K_i \neq K_j, \qquad 1 \leq i, j \leq n \text{ and } i \neq j \quad (6)$$

The MKP assumes that there are two secure initial keys r_0 and k_0 which are established by a trust center for the communication. The list of keys is generated in sender node and receiver node, where two lists of parameters $r_1, r_2, \ldots r_n$, and $k_1, k_2, \ldots k_n$ are computed as following:

$$r_i = \mathrm{ECC_{enc}}\,(\mathrm{TC_{PK}}, r_{i-1}) \;\; 0 < i \leq n \quad (7)$$

$$k_i = \mathrm{ECC_{enc}}(\mathrm{TC_{PK}}, \mathrm{k_{i-1}}) \;\; 0 < i \leq n \quad (8)$$

Where ECC_{enc} is elliptic curve function; it ciphers the input with trust center public key (TC_{PK}). The trust center must established r_0 and k_0 to sender and receiver by node's public key securely.

Any link key K_i is computed by a hash function (f) as below:

$$K_i = \mathrm{f}(k_i, r_{n-i+1}), 1 \leq i \leq n \quad (9)$$

From (7), (8) and (9) it is clear that the secret keys are protected by ECC, where each key is generated from two different sub-keys (see Fig. 3 ECC MKP key generation model). The generated keys $K_1, K_2, \ldots K_n$ are used by AES algorithm for data encryption and decryption, where any block of data B_i is encrypted to be C_i by AES algorithm with key K_i

$$C_i = \mathrm{AES}_{K_i}(B_i) \quad (10)$$

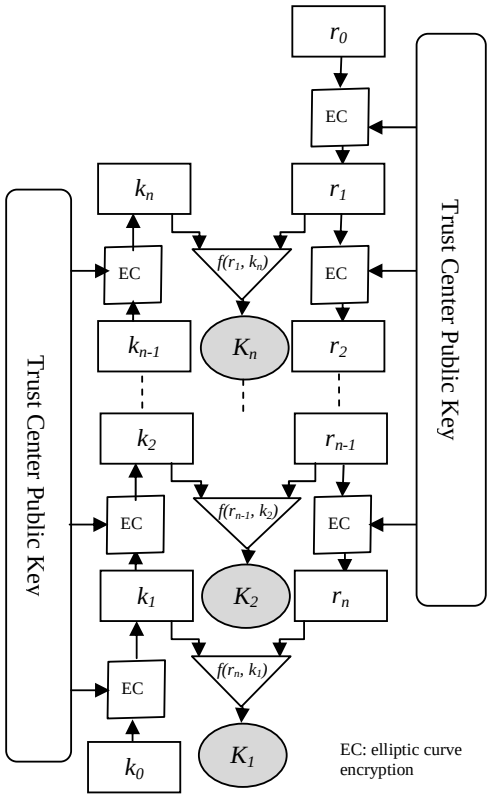

Fig. 3 ECC MKP key generation model

The receive node decrypts any block of cipher text C_i by AES with the same key K_i that computed by the same way in the sender node.

$$B_i = AES_{K_i}(C_i) \quad (11)$$

The proposed protocol assumes no changes with the original AES algorithm except that each block of data has its own key, and the encryption/decryption algorithm will be the same, but new key generation algorithm is build for MKP protocol. The algorithm can generate n keys, where the number of keys refers to number of blocks (also refers to level of security as will discuss later) as shown in Fig. 4 MKP algorithm.In MKP, secret keys are computed by ECC algorithm. Moreover, two initial values r_0 and k_0 are used to compute the parameters $k_1, k_2 \ldots k_n, r_1, r_2 \ldots, r_n$ (see (7) and (8)), which values are established by a trust center. A hash function (f) is used to generate secret keys $K_1 \ldots K_n$ (see (9)). Number of secret keys is computed by a security level, where different level of security can be proposed (see table 5.1.1 ZigBee security levels). A number of blocks in each group are computed based on message size and security level. Encryption and decryption operations of blocks in any group are done via that group key.

Input: k_0, r_0, Message, PK, Security_L
Output: C
Begin
BlockSize=128
for i= 1 to Security_L
$k_i = ECC(k_{i-1}, PK)$
$r_i = ECC(r_{i-1}, PK)$
$K_i = f(k_i, r_{n-i+1})$
No_of_Blocks = Message/BlockSize
No_of_Blocks_in_Group = No_of_Blocks /Security_L
Count = 1
Key = K_{count}
for j =1 to No_of_Blocks
$C_j = AES(B_j, Key)$
if (j % No_of_Blocks_in_Group == 0)
then Key = $K_{count++}$
End

Fig. 4 MKP algorithm

VI. ZigBee and MKP

MKP can be embedding in ZigBee technology, which applies CCM* mode. Each message is divided to sub-message with 128-bit block size then n keys are generated for AES algorithm. The number of keys refers to the level of security which refers to amount of system complexity and security strength. The highest level of security is n, where n is the number of block consist a message and block size is 16-octet. The blocks of message can classified to groups (maximum n) of blocks, whose keys are different, where each group has its own key.

CCM* mode allows ZigBee technology to perform message confidentiality and integrity together, as well as confidentiality or integrity alone. The authentication code will be computed as previous steps for each sub-message. The authentication code is computed for each sub-message (see Fig. 5 ZigBee Authentication Code in MKP). All authentication codes of sub-message are XOR with each other to produce one authentication code.

$$T = T_1 \oplus T_2 \oplus \ldots \oplus T_n \quad (12)$$

For encrypting data the CTR mode with nonce is applied, but with different key for each group to generate their key stream. The octets assigned to counter are reduced because each group has its own counter, where number of blocks of each group is less than before because in previous state all the blocks are in one group. The number of octets assigned for nonce will be increased that reduce the limitation for nonce generation.

VII. Benefits of MKP for ZigBee

The MKP protocol provides multi-level security in ZigBee technology. ZigBee applications with multi-role can get multi-level of security when using MKP. The numbers of generated keys in MKP are computed by a level of security.

From secrecy side, in previous protocol it used one secret key, while in MPK protocol it uses multiple-key. If the analyst needs $2^{keysize/2}$ operations to break one key in previous with time complexity O (log_2 $(keysize/2)$), then he will need

$n \times 2^{keysize/2}$ operations to break n keys in MKP protocol with time complexity O ($n \times log_2 (keysize/2)$).

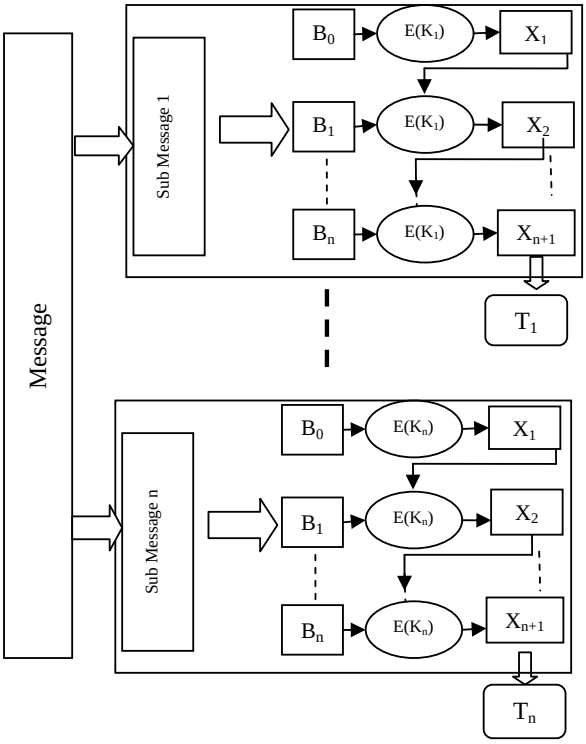

Fig.5 ZigBee Authentication Code in MKP

The important of system performance pushes to investigate the possibility of reducing the run time, where in previous protocol it is possible to run a parallel encryption, but not parallel authentication because it is run sequentially. However, the MKP provides parallel authentication and parallel encryption because it has multiple-key that enable it to run in parallel fashion, which means increasing throughput.

The raising of system performance leads to reduce the run time; also it will save sensor energy. Moreover, throughput increasing may minimize the active time of the node that means the node will be in sleep mode more time than previous. In addition, the amount of data in unit of time that is ready for send will be more than before because of parallel execution. MKP will exploit more bandwidth of the network.

REFERENCES

[1] Li Chunqing, Zhang Jiancheng, “Research of ZigBee's data security and protection”, *International Forum on Computer Science-Technology and Applications 2009*, IEEE, 2009, pp 298 - 302.

[2] ZigBee Alliance, Retrieved 2011, from http://www.zigbee.org/Standards/Overview.aspx

[3] ZigBee Alliance , “ZigBee and wireless radio frequency coexistence”, white paper, published by ZigBee Alliance, June 2007, Retrieved 2011, from http://www.zigbee.org/imwp/idms/popups/pop_download.asp?contentID=11745

[4] Patrick Kinney, “ZigBee technology: wireless control that simply works”, Communications design conference, vol 2, 2 October 2003. retrieved 2011, from http://www.zigbee.org/en/press_kits/latest/documents/white_papers/wp_zigbeetechwireless_final.pdf

[5] Bin Yang, " Study on security of wireless sensor network based on ZigBee standard." International Conference on Computational Intelligence and Security, IEEE, 2009, pp 426 - 430.

[6] Doug Whiting, Russ Housley and Niels Ferguso, “Counter with CBC-MAC (CCM) AES mode of operation”, submitted to NIST, retrieved 2011, from http://csrc.nist.gov/groups/ST/toolkit/BCM/documents/proposedmodes/ccm/ccm.pdf

[7] Meng Qianqian and Bao Kejin, “Security analysis for wireless networks based on ZigBee”, IEEE,vol 1, 2009, pp 158 - 160.

[8] Arvinderpal S. Wander, Nils Gura, Hans Eberle, Vipul Gupta, Sheueling Chang Shantz, “Energy analysis of public-key cryptography for wireless sensor networks”, Third IEEE International conference on Pervasive Compting and Communications, IEEE Computer Society, 2005, pp 324-328.

[9] Roy Pereira, “ZigBee and ECC secure wireless networks”, August 09 2004, retrieved 2011, from http://electronicdesign.com/article/embedded-software/zigbee-and-ecc-secure-wireless-networks8369.aspx.

[10] Jerry Krasne, “Using elliptic curve cryptography(ECC) for enhanced embedded security financial advantages of ECC over RSA or Diffie Hellman”, Embedded Market Forecasters American Technology International, Inc., November 2004, retrieved 2011, from http://embeddedforecast.com/EMF-ECC-FINAL1204.pdf

[11] ZigBee Alliance, "The enduring value of symmetric encryption”, white paper, August 2000, retrieved 2011 from http://www.partnerdata.it/files/WP-Symmetric%20Encryption.pdf.

[12] Zhen-rong Li, Yi-qi Zhuang, Chao Zhang and Gang Jin, "Low-power and area-optimized VLSI implementation of AES coprocessor for ZigBee system", The Journal of China Universities of Posts and Telecommunications (Elsevier), vol 16, Issue 3, June 2009, pp 89-94.

[13] Hyunjue Kim, Chang Hyun Kim and Jong-Moon Chung, "A novel elliptical curve id cryptography protocol for multi-hop ZigBee sensor networks", *Wireless Communication and Mobile Computing Wirel. Commun. Mob. Comput. (2010)*, Wiley Interscience, JohnWiley & Sons, Ltd, 2010.

[14] Tadiwa Elisha Nyamasvisva, Halabi Hasbullah, "Multi-level security algorithm for random ZigBee wireless sensor networks", *Information Technology (ITSim), 2010 International Symposium in Kuala Lumpur*, IEEE, vol 2, 2010, pp 612 - 617.

[15] Jelena Misic, Vojislav Misic, “Wireless personal area networks performance interconnections and security with IEEE 802.15.4., John Wiley & Sons Ltd, 2008.

[16] NFIPS, “197: Announcing the advanced encryption standard (AES)”, *Information Technology Laboratory, National Institute of Standards and Technology*, November 2001.

[17] S.A Vanston, "Next generation security for wireless: elliptic curve cryptography", *Computers and Security*, Elsevier, vol 22, no. 5, 2003,pp 412 - 415.