JKAU: Comp. IT. Sci., **Vol**. **8** No. 2, pp: 57 – 68 (1440 A.H. / 2019 A.D.)
Doi: 10.4197/Comp. 8-2.5

Secure Lightweight Routing Scheme for Energy Efficient Wireless Sensor Networks

Yasser R. Alselehibi, Mohammad H. Zafar and **Madini O. Alassafi**

Department of Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia

yselehibi@stu.kau.edu.sa

Abstract. The technology revolution in wireless communications and micro-electro-mechanical systems (MEMS) directly affects the development of wireless sensor networks (WSNs), which are used in several application areas, including the military, home, and environment. One of the best categories of routing networks in WSNs are hierarchical protocols (cluster-based). The well-known protocols in this category include the Low Energy Adaptive Clustering Hierarchy (LEACH). However, the LEACH is vulnerable to many attacks. To provide cryptographic protection against outsider attacks, a modified version of LEACH, called Enhancing Secure LEACH (MS-LEACH) protocol, is used. MS-LEACH enhances security but increases power consumption. To maintain an acceptable level of security and decrease the power consumption of secure LEACH protocols, the present research proposes a Secure Lightweight LEACH (SLW-LEACH) scheme. The simulation results show that this proposed SLW-LEACH protocol outperforms the MS-LEACH in terms of network lifetime, energy consumption, network throughput, and normalized routing load (NRL).

Keywords: Wireless sensor networks; LEACH; clustering; MS-LEACH; SLW-LEACH.

1. Introduction

A wireless sensor network (WSN) is a network of organized sensor devices that gathers information from a zone of concern. Certain types of networks are commonly organized in the military context, and some applications are related to monitoring. Sensor devices armed with nodes interact with each other to detect and process data and then forward that data to a base station (BS). Certain nodes are inhibited by restricted memory and battery life, and it has been claimed that they are also of low quality for calculations and communications. The routing protocol of a WSN presents the greatest challenge, as it exerts a direct effect on power compared to ad-hoc and cellular networks [1-2]. In WSNs, there are three main categories of routing protocols: flat, or data-centric; cluster-based, or hierarchical; and location-based [3-5]. The cluster-based protocol is considered the most suitable for routing protocols in WSNs, because it enhances power, stability, and minor network expectancy [6]. The Low Energy Adaptive Clustering Hierarchy Routing Protocol (LEACH) [3] assembles nodes into clusters that contain one cluster head (CH) and a few cluster members. Before transmitting data to the BS, all the data are first sent first to a CH. To maintain the energy load, the LEACH assimilates the random rotation of the high-power CH's location with the sensors. However, the LEACH is more sensitive to several types of attacks, including replaying, spoofing, and jamming of the WSN. Some current research is addressing improved

LEACH security. The Enhancing Secure LEACH (MS-LEACH) [7] is a well-known modified version of the LEACH that provides cryptographic security against outside attacks while maintaining comparable processing overhead and communication. By preventing an intruder from being a CH sinkhole, selective forwarding and MS-LEACH protect against HELLO flooding threats. They prevent intruders from sending bogus sensor data to CHs and CHs from forwarding bogus messages. However, MS-LEACH is not very energy efficient. The present work is inspired by the need to enable MS-LEACH to reduce the protocol's energy consumption. This paper proposes a Secure Lightweight LEACH (SLW-LEACH), which uses a more efficient function of lightweight cryptographic encryption to improve energy-efficiency performance compared with that of MS-LEACH.

This paper is organized as follows: Section 2 discusses related work. Section 3 presents the proposed SLW-LEACH scheme. Section 4 presents the simulation results and the analysis. Section 5 presents the paper's conclusions and future directions.

2. Related Work

A. Encryption Algorithms for WSNs

To meet the security requirements of WSNs, encryption algorithms are used to authenticate data and keep it confidential. Protocols use several encryption algorithms: the SPIN [8] and LEAP [9] protocols use TinySec [10], and the RC5 security packet uses the Skipjack algorithm. Encryption algorithms can be classified as having two types of key encryption: public (or asymmetric) and private (or symmetric). In addition, the lightweight algorithms of symmetric cryptography have recently become the targets of active research. Such algorithms, including SIMON [11], SPECK [11], and RECTANGLE [12], can be used in tiny devices that are embedded to provide powerful security and that have lower memory constraints and power costs than those containing the standard symmetric cryptography algorithms, which include RC5 [13], RC6 [14], AES [15], Blowfish [16], and 3DES [17]. Public key encryption is used to solve the issue of key distribution. Techniques for public key encryption are slower than those for private key encryption because they need more power for computation processing [18]. Therefore, public key encryption is not suitable for use in WSNs. Several studies have analyzed the performance of the lightweight block cipher algorithms used for WSNs [11-12, 19-23].

The SIMON-64 and SIMON-128 algorithms employ 32-bit and 64-bit words, respectively. SIMON comprises 10 different ciphers that support application security in a controlled environment. In addition, the SIMON ciphers that have block sizes of 2n bits and mn bits are called 2n/mn. Furthermore, SIMON uses a collection of rotation parameters and circular permutations of shift-bit.

The SPECK algorithm also comprises 10 different block ciphers that support application security in controlled environments. The round function of SPECK is analogous to the mixing operation of THREEFISH [24]. SPECK also uses a collection of rotation parameters and circular permutations of shift-bit.

The RECTANGLE algorithm proposes a new, lightweight SPN block cipher that is hardware friendly [12] and uses the bit-slice technique. The SERPENT [33-34] algorithm also uses this technique, which provides a 4X4 S-box [34]. RECTANGLE has a key size of 80/128 bits and a block size of 64 bits, with 25 rounds. Three operations occur in each round: ShiftRow (every row is rotated left over various offsets), AddRoundkey (XOR bitwise with a round key), and SubColumn (S-boxes, 4-bit, in parallel). In the substitution layer are 16 of the same 4X4 S-boxes, in parallel, and in the permutation layer

are 3 rotations. Because of the bit-slice technique, RECTANGLE has a good software speed [12]. To avoid slide threats in the key schedule, various round constants are added. The RECTANGLE mix of P-layer and S-box provides linear (or limited) differential trails. RECTANGLE also offers strong resistance against side-channel and mathematical attacks. Similar to AES [15], RECTANGLE has a matrix structure, which requires more computation cycles [25].

B. Secure LEACH Protocols

As explained above, research has increasing targeted the security of clustered WSNs. Due to space constraints, we provide examples of research aiming to secure the LEACH protocol [3] as it is commonly used in WSNs. This is quite suitable for meeting the objectives of our research. This section discusses several proposed modifications of LEACH. The protocol under which security is added to LEACH is referred to as SLEACH [26]. It is the first altered version of LEACH that has cryptographic protection against outside threats. SLEACH states that every node must have two symmetric keys, that the last key must be held by the BS, and that a pairwise key must be exchanged with the BS only. SLEACH implements authentication for broadcast to CHs in two stages, leveraging validated BS that has many other resources. In SLEACH, every CH sends a modified ADV message. The BS waits to be authenticated and to hear the ADV messages from each CH. Then, the BS compiles a list of legitimate CHs and sends it to the network, which uses the μTESLA broadcast scheme [8]. Normal nodes determine which ADV message they receive; authenticate that the message belongs to a legitimate node; and proceed with the rest of the protocols, selecting a CH from the list broadcast by the BS. Using only two keys at each node, SLEACH provides an efficient solution to authenticating node-to-CH messages, but it is not energy-efficient.

Subsequently, Oliveira *et al*. suggested SecLEACH [27], which integrates the key pre-distribution of Eschenauer *et al*. with LEACH [3]. SecLEACH uses a key pool that has S keys, similar to the random key of Eschenauer *et al*., which has ID for each node. The SecLEACH protocol has five steps, which look like those of the LEACH protocol.

The GS-LEACH protocol [28] is a secure version of LEACH. This protocol enhances SecLEACH. In SecLEACH, the aggregator is selected by the normal nodes. In addition, GS-LEACH has a shared key with the normal nodes. In some cases, a normal node that has more energy transmits directly to the BS, because the aggregator does not have the key. Furthermore, the next aggregator will be selected in the normal mode if the aggregator does not have a shared key. The assumptions made by GS-LEACH about sensor-node distribution are based on a grid. First, the sensing area is divided into squares, k, with sensor nodes, n, deployed in each square. The key pool, S is composed of partial groups of k. Then, the keys, m, from the S partial group are randomly assigned to each square. The shared key is shared with the BS by all nodes. GS-LEACH has five clustering and reporting steps, which are similar to those of SecLEACH with three main differences: 1) The clustering is carried out inside the grid, 2) Every grid has an aggregator, and 3) A node will sleep during a round, as it has not shared a key with an aggregator. Consequently, GS-LEACH has almost the same security strengths as Sec-LEACH, but the range of communication between the normal nodes and an aggregator is shorter than in Sec-LEACH. Also, energy consumption can be reduced if a node does not have an aggregator. GS-LEACH and SecLEACH have three disadvantages. First, these protocols do not provide broadcast authentication when an aggregator broadcasts a message. Second, due to the threats posed by

node compromise, GS-LEACH and SecLEACH are sensitive. Finally, pre-distribution of the random key causes incomplete connectivity within sensor nodes.

The Armor LEACH protocol [29] integrates solutions provided by the time-controlled clustering algorithm (TCCA) and SecLeach into a solution that provides WSNs a high security standard and low power consumption. The TCCA modifies the election of CHs by adding another election condition, the presence of energy, as distinct from the sensor's maximum energy, which is controlled by the timestamp. This protocol extends the one-hop cluster into the communication of a multi-hop cluster. The timestamp helps the CH by approximates the relative distance of every member, so that the best phase-setup time to use in future rounds can be identified. In creating a collision-free transmission schedule, the timestamp with the TTL helps the CH to create a view of multi-hops relative to its clusters. The security aspect of Armor LEACH is the same as that of SecLEACH, so the security analysis of both is the same.

MS-LEACH provides data confidentiality and authentication for the CH node using pairwise keys to enhance the SLEACH protocol. However, MS-LEACH's limitation is that it is consumes more energy because it uses a standard block cipher algorithm, which is appropriate for devices that have no constraints.

LS-LEACH [30] proposes a protocol that discourages attackers from joining a WSN using lightweight, energy-efficient authentication mechanisms in which the CH verifies the validity of nodes by asking to join the cluster.

H-LEACH [32] proposes a new protocol by using a hierarchical key-sharing structure. However, this security solution is not effective against insider attacks, and it is not energy efficient.

SS-LEACH [31] proposes a specification-based protocol that deals with security measures for authentications and to avoid sinkhole attacks, because there is no CH or non-CH authentication in LEACH, which makes a network vulnerable to attacks. However, SS-LEACH does not protect against node compromise.

3. Proposed SLW-LEACH Protocol

A. Assumptions

This section discusses some assumptions regarding the accurate actions of the protocol proposed. Initially, all nodes have power resources of equal value, and the base station is not restrained regarding power resources and computational power. The base station is secure against compromising attacks and impersonation attacks from adversaries. For freshness purposes, every node X is embedded with two keys: KX and KI. KX is a master symmetric key that each node transfers toward the BS. KI is a group key that is transferred by all network nodes and that is used for broadcast authentication. KI is the calculated last symmetric key that is produced from the BS. Each node transfers the counter CX of the BS. This protocol inherits the following assumptions of the MS-LEACH and LEACH protocols: nodes span the detecting field, and each sensor node has sufficient ability to transmit directly to the base stations. Any node in the network is accessible through any other node with a single hop, but communication from the node to the base station normally take place in two hops, first from the normal node to the CH and then from the CH to the main BS. All nodes are static, individual nodes are untrusted, and our protocol is used only to protect the network from outsider attackers.

B. SLW-LEACH Implementation

The MS-LEACH algorithm uses the Blowfish block symmetric cipher in its cryptographic functions [16]. We reviewed the

research on the performance of block cipher algorithms for WSNs, finding that symmetric block cipher algorithms that are more energy efficient and lightweight than Blowfish, including SIMON [11], SPECK [11], and RECTANGLE [12], are well-known in lightweight block cipher research. Therefore, we implemented our proposed SLW-LEACH using the following encryption algorithms: SIMON, SPECK, and RECTANGLE. Based on the results of the simulation, we decided to use SIMON as the basic cryptographic function in SLW-LEACH for encryption, decryption, and MAC computations.

C. SLW-LEACH Algorithm Design

In SLW-LEACH, each node is used to transfer a pairwise key with every node in the field. The pairwise key conducts source validation, which helps to protect the privacy of the data communicated. A pairwise key is created through a sensor node of the CH. Figure 1 shows a block diagram of SLW-LEACH. The rest of this section discusses its details.

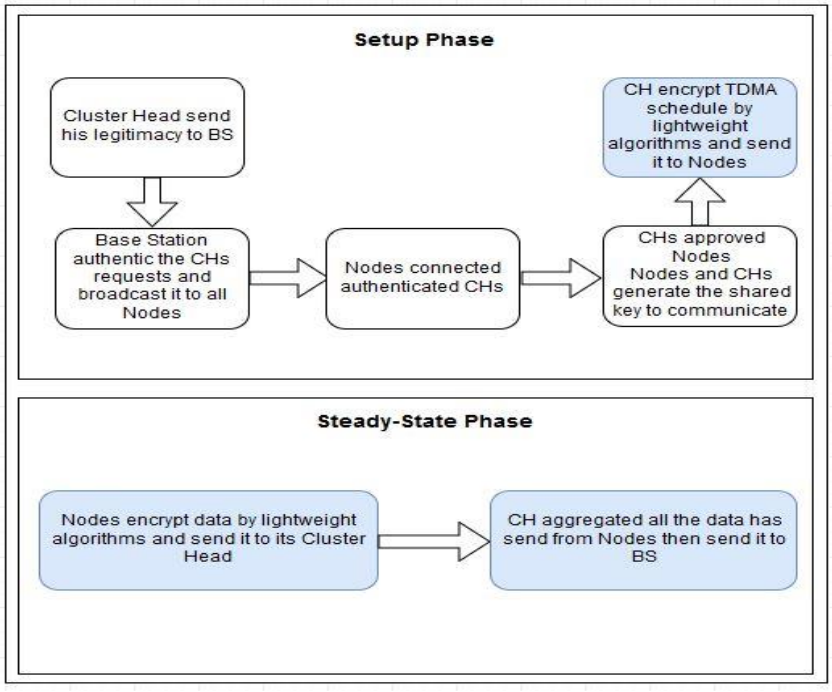

Fig. 1. SLW-LEACH Block Diagram.

1. Setup Phase

In the SLW-LEACH setup phase, each cluster creates the key between the CH and the BS, which is shared by the MAC and then transferred to the BS. Then, the BS checks the key's validity. If the key is valid, the CH automatically adds it to the list. Next, it highlights all valid CH node lists on the network. After identifying the valid CHs to which they are attached, the nodes transfer their applications to take part in the CH's clusters. Then, the CHs announce their validation memoranda to the sensor nodes, so that they can be approved. The algorithms used in the setup phase are SIMON, SPECK, RECTANGLE (for TDMA schedule encryption and decryption), and HMAC (for MAC generation). After the node sends the joint request, the CH and its child nodes generate a

pairwise key. The CH sends an encoded, form-protected TDMA schedule to each child node using a counter value and generates a pairwise key between the node and the CH. The CH also uses the pairwise key shared between the node and the CH to send an encoded TDMA schedule and the counter's MAC value. To enable communication between the CH and the node, the SLW-LEACH uses the same pairwise key that is used in the MS-LEACH. The pairwise key Kuv is generated between the nodes u and v [9].

$$Kv = fKI\ (v) \quad (1)$$

$$Kuv = fKv\ (u) \quad (2)$$

Where fk is a member of pseudo-random functions and KI is the symmetric final key series placed in the BS that is preloaded in every node.

2. SLW-LEACH Steady-State Phase

In the SLW-LEACH steady-state phase, nodes use the algorithms SPECK, SIMON, RECTANGLE, and HMAC to send encrypted IDs and data to the CH. The child node transfers the encrypted form of its data and their IDs to its CH using a pairwise key, and the MAC value of the encrypted message and the counter is created using the pairwise key. The CH transfers its estimated result for the calculation to the BS. The MACs from the group members are promoted in the MAC's array messages. The BS must approve both of the MAC values produced from the ordinary nodes by the CH. Unless this authentication is successful, the BS will remove the consistent total outcome and the creators of the unsuccessful MACs will be seen as intruders. In the case of intruders between the conventional nodes, the BS is responsible for validating the CHs and highlighting them in their network site, which includes sending a message to the BS about the performance.

3. Security Analysis

SLW-LEACH provides acceptance-level confidentiality using lightweight block cipher algorithms. In addition, the HMAC is used to provide data authentication and integrity. Furthermore, SLW-LEACH encrypts and authenticates all data-transmission and processing steps. Table 1 shows the security services and mechanisms used by SLW-LEACH.

Table 1. SLW-LEACH Security Services.

Security Services	Mechanism
Confidentiality	Encryption
Authentication and Integrity	MAC
Authenticated Broadcast	μTESLA
Freshness	Counter

4. Simulation Results and Analysis

Network simulator 2 (NS-2) [35] is used to analyze the proposed SLW-LEACH scheme. NS-2 is a discrete event simulator that was developed by the University of California at Berkeley and the VINT project [35] and that provides substantial support for simulating wireless networks. The simulation setup was conducted in a 1000 × 1000-meter area. The nodes were placed randomly, with a network of 25, 50, 75, and 100 nodes. The size of the packet was 1460 bytes (fixed). The UDP execution was simulated using the CBR traffic. The traffic load was varied by changing the CBR values. A new MAC protocol type, MAC Sensor, was created for LEACH using MIT's μAMPS project [36]. This protocol is an integration of a simple model of TDMA, Carrier-Sense Multiple Access (CSMA) and Direct-Sequence Spread Spectrum (DS-SS). As mentioned in the LEACH protocol [3], in the set-up phase, the CHs send their advertisement messages and the child nodes send their join requests using CSMA. In this steady-state phase, to decrease inner overload interloping, each group in the LEACH interconnects within the group nodes using DS-SS. Data is transferred from the manager of the group

nodes to the BS using CSMA. Table 2 shows the NS-2 configuration parameters, which are common to the protocols simulated.

This section presents the results of analyzing the simulation of the proposed SLW-LEACH protocol. Four measures were used to evaluate the performance of the SLW-LEACH and MS-LEACH protocols. The following subsections define those performance measures and present graphs that represent the results of the simulation.

Table 2. NS-2 Configuration Parameters.

Parameter	Value
Simulator	Network Simulator 2.35
Topology	Random
Interface Type	Phy-WirelessPhy-802.15.4
MAC Type	IEEE 802.15.4
Queue Type	Drop Tail/Priority Queue
Queue Length	50 Packets
Antenna Type	Omni Antenna
Propagation Type	Two Ray Ground
Routing Protocol	LEACH
Transport Agent	UDP
Application Agent	CBR
Network Area	1000 * 1000
Number of Nodes	25, 50, 75, and 100

A. Average Energy Consupmtion

Because energy is the most important restriction in a WSN, the protocols were matched in terms of their power outcomes. Figure 2 shows the average percentage of power consumed by all the nodes in SLW-LEACH, using various lightweight cryptographic functions (SIMON, SPECK, and RECTANGLE) and MS-LEACH with the various numbers of network nodes (25, 50, 75, and 100).

Figure 2 shows that all lightweight cryptographic algorithms used in SLW-LEACH consume less power than MS-LEACH. The reason for this lower power consumption is that Blowfish (the cryptographic algorithm used in MS-LEACH) is replaced with a cryptographic algorithm that has a lighter encryption and decryption process. Figure 2 also shows that SLW-LEACH and SIMON have an advantage over other algorithms in terms of average power consumption. Next is SLW-LEACH and RECTANGLE. SLW-LEACH and SIMON consume 0.94% less power than MS-LEACH. SLW-LEACH and RECTANGLE consume 0.5% less power than MS-LEACH. SLW-LEACH and SPECK consume l0.35% less power than MS-LEACH.

B. Average Network Lifetime

This measurement shows the result of the projected SLW-LEACH security enhancement on the WSN for the entire life associated with MS-LEACH. Figure 3 shows the network lifetime (in seconds) with SLW-LEACH, using the various cryptographic functions (SIMON, SPECK, and RECTANGLE) and MS-LEACH with various numbers of network nodes.

Figure 3 shows that all lightweight cryptographic algorithms used in SLW-LEACH demonstrate higher average network lifetimes because they consumes less power. Figure 3 also shows that SLW-LEACH using SIMON has a longer network lifetime than the other algorithms, followed by SLW-LEACH using RECTANGLE. SLW-LEACH using SIMON obtains a 30% higher average network lifetime than the MS-LEACH protocol. SLW-LEACH using RECTANGLE obtains a 27% higher average network lifetime than MS-LEACH. In addition, SLW-LEACH using SPECK obtains a 24% higher average network lifetime than MS-LEACH. Therefore, the proposed SLW-LEACH using SIMON obtains the best network lifetime performance.

C. Average Network Throughput

The average network throughput measures the effect of the proposed security enhancement in SLW-LEACH compared with MS-LEACH. Figure 4 shows the average network throughput (bytes/second) of SLW-LEACH using the various cryptographic functions (SIMON, SPECK, and

RECTANGLE) and MS-LEACH with various numbers of network nodes.

Figure 4 shows that the proposed SLW-LEACH protocol, in all the cryptographic functions used, increased the network throughout as compared with MS-LEACH. Figure 4 also shows that SLW-LEACH using SIMON has the highest network throughput compared with the other algorithms. Next is SLW-LEACH using RECTANGLE. SLW-LEACH using SIMON increased the network throughout by 1.17% compared with MS-LEACH. SLW-LEACH using RECTANGLE increased the network throughout by 1.13% compared with MS-LEACH. SLW-LEACH using SPECK had the same network throughout as MS-LEACH. Therefore, SLW-LEACH using SIMON obtains the best performance in terms of average network throughput.

D. Average Normalized Routing Load

This metric measures the overhead of network traffic. Figure 5 shows the average normalized routing load (NRL) of SLW-LEACH using the various cryptographic functions (SIMON, SPECK, and RECTANGLE) and MS-LEACH with various numbers of nodes.

Figure 5 shows that SLW-LEACH using SIMON and SLW-LEACH using RECTANGLE decreased the NRL by 6% and 3%, respectively, compared with MS-LEACH.

With SLW-LEACH using SPECK, the NRL decreased by 1% compared with MS-LEACH. Hence, Figure 5 shows the superiority of SLW-LEACH using SIMON over the other algorithms in terms of NRL. Next is MS-LEACH using RECTANGLE. Increasing the number of nodes, which necessarily increases the CBR (traffic load), decreases the NRL, which in turn results in a significant performance difference between the protocols. Therefore, SLW-LEACH using SIMON obtains the best performance in terms of average NRL.

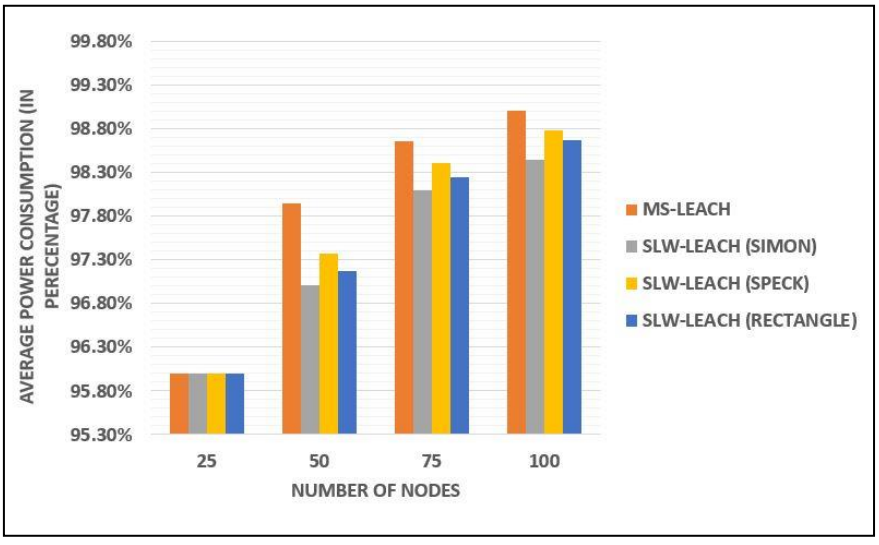

Fig. 2. Average power consumption of SLW-LEACH (SIMON, SPECK, RECTANGLE) and MS-LEACH with various numbers of nodes.

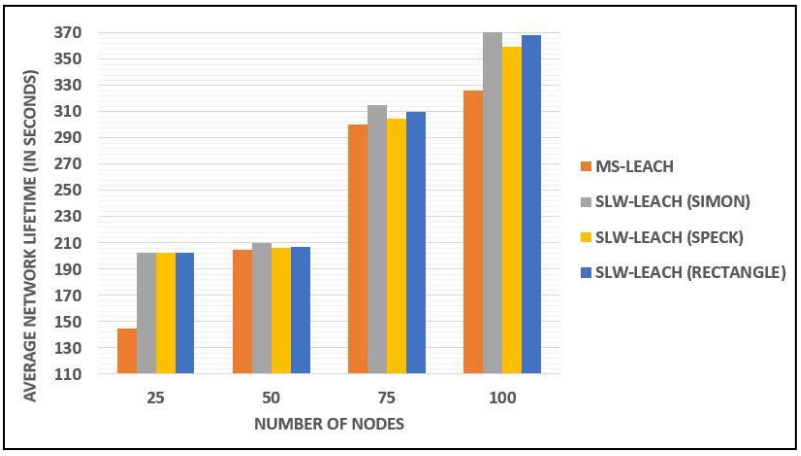

Fig. 3. Average network lifetime of SLW-LEACH (SIMON, SPECK, RECTANGLE) and MS-LEACH with various numbers of nodes.

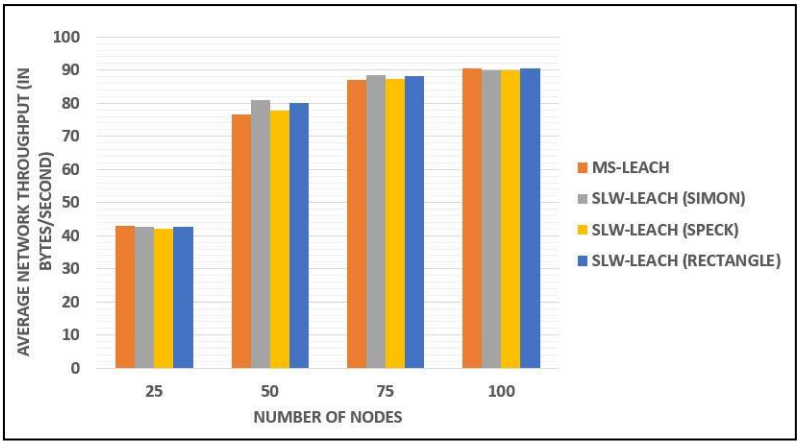

Fig. 4. Average network throughput of SLW-LEACH (SIMON, SPECK, RECTANGLE) and MS-LEACH with various numbers of nodes.

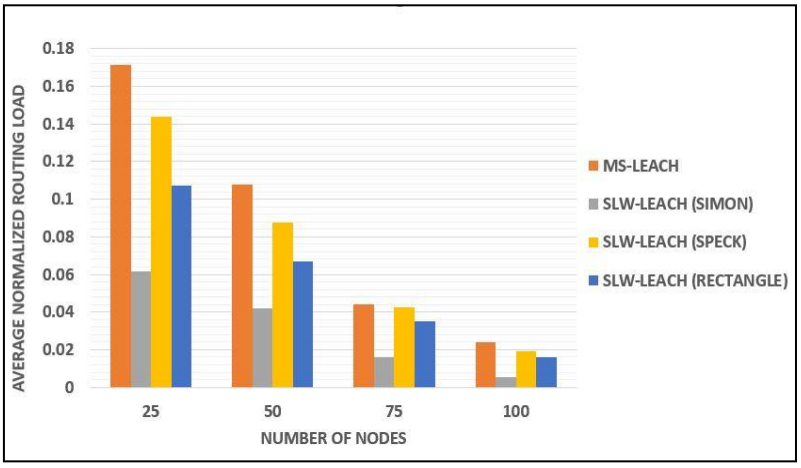

Fig. 5. Average normalized routing load of SLW-LEACH (SIMON, SPECK, RECTANGLE) and MS-LEACH with various numbers of nodes.

5. Conclusion

This paper presented a SLW-LEACH protocol that enhances the security of the original MS-LEACH by providing a lightweight block cipher algorithm that provides authentication and data confidentiality. In addition, SLW-LEACH provides more energy-efficient cryptographic functions than the original MS-LEACH for encryption and decryption. To the best of our knowledge, this is the first implementation of a lightweight block cipher algorithm in a LEACH protocol. The performance evaluation of the SLW-LEACH protocol is analyzed by employing various lightweight cryptographic functions, simulation is conducted, and the performance comparison against MS-LEACH is presented. The SLW-LEACH proposed can be deployed with the lightweight cryptographic algorithm SIMON, due to its efficiency over the other cryptographic functions. The proposed SLW-LEACH demonstrated that it achieves all WSN security goals and has the best safety properties. The experimental results validate the efficiency of the proposed SLW-LEACH protocol and show how that protocol attains a set of preferred safety objectives while care a satisfactory equal of energy income. The experimental outcome demonstrates that the SLW-LEACH proposed surpasses other protocols and allowances in terms of network lifetime, energy consumption, network throughput, and NRL.

References

[1] **Pandana, C.** and **Liu, K. R.** (2008) "Robust connectivity-aware energy-efficient routing for wireless sensor networks," *IEEE Transactions on Wireless Communications*, **7**(10): 3904-3916.

[2] Q. Cao, T. Abdelzaher, T. He and R. Kravets, (2007) "Cluster-based forwarding for reliable end-to-end delivery in wireless sensor networks," IEEE International Conference on Computer Communications, pp. 1928-1936.

[3] **Heinzelman, W. R., Chandrakasan, A.** and **Balakrishnan, H.** (2000) "Energy-efficient communication protocol for wireless microsensor networks", *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences*, p. 10.

[4] **Lindsey, S.** and **Raghavendra, C. S.** (2002) "PEGASIS: Power-efficient gathering in sensor information systems," *Proceedings IEEE aerospace conference proceedings*, vol. **3**, pp. 3-3.

[5] **Younis, M., Youssef, M.** and **Arisha, K.** (2002) "Energy-aware routing in cluster-based sensor networks", *Proceedings 10th IEEE international symposium on modeling, analysis and simulation of computer and telecommunications systems*, pp: 129-136.

[6] **Younis, O.** and **Fahmy, S.** (2004) "HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks", *IEEE Transactions on Mobile Computing*, **4**: 366-379.

[7] **El_Saadawy, M.** and **Shaaban, E.** (2012) "Enhancing S-LEACH security for wireless sensor networks", *IEEE International Conference on Electro/Information Technology*, pp: 1-6.

[8] **Perrig, A., Szewczyk, R., Tygar, J., Wen, V.** and **Culler, D. E.** (2002) "SPINS: Security protocols for sensor networks", *Wireless Networks*, **8**(5): 521-534.

[9] **Zhu, S., Setia, S.** and **Jajodia, S.** (2006) "LEAP+: Efficient security mechanisms for large-scale distributed sensor networks", *ACM Transactions on Sensor Networks (TOSN)*, **2**(4): 500-528.

[10] **Karlof, C., Sastry, N.** and **Wagner, D.** (2004) "TinySec: A link layer security architecture for wireless sensor networks", *Proceedings of the 2nd international conference on Embedded networked sensor systems*, pp: 162-175.

[11] **Beaulieu, R., Shors, D., Smith, J., Treatman-Clark, S., Weeks, B.** and **Wingers, L**. (2013) "*The SIMON and SPECK Families of Lightweight Block Ciphers Cryptology*", National Security Agency, ePrint Archive.

[12] **Zhang, W., Bao, Z., Lin, D., Rijmen, V., Yang, B.** and **Verbauwhede, I.** (2015) "RECTANGLE: a bit-slice lightweight block cipher suitable for multiple platforms," *Science China Information Sciences*, **58**(12): 1-15.

[13] R. L. Rivest, (1994) "The RC5 encryption algorithm.," International Workshop on Fast Software Encryption, pp. 86-96.

[14] **Rivest, R. L., Robshaw, M., Sidney, R.** and **Yin, Y.** (1998) "The RC6 block cipher", *First Advanced Encryption Standard (AES) Conference.*

[15] **Daemen, J.** and **Rijmen, V.** (2013) "*The design of Rijndael: AES-the advanced encryption standard*", Springer Science & Business Media.

[16] **Schneier, B.** (1993) "Description of a new variable-length key, 64-bit block cipher (Blowfish)," *International Workshop on Fast Software Encryption*, pp: 191-204.

[17] **Stallings, W.** (2006) "*Cryptography and Network Security*", Pearson Education India.

[18] **Hardjono, T.** and **Dondeti, L. R.** (2005) "*Security in Wireless LANS and MANS (Artech House Computer Security)*", Artech House Inc.

[19] **Zhang, X., Heys, H. M.** and **Li, C.** (2010) "Energy efficiency of symmetric key cryptographic algorithms in wireless sensor networks," *25th Biennial Symposium on Communications*, pp: 168-172.

[20] **Singh, S. Sharma, P. K., Moon, S. Y.** and **Park, J. H.** (2017) "Advanced lightweight encryption algorithms for IoT devices: survey, challenges and solutions", *Journal of Ambient Intelligence and Humanized Computing*, pp: 1-18.

[21] Koo, W. K., Lee, H., Kim, Y. H. and **Lee, D. H.** (2008) "Implementation and analysis of new lightweight cryptographic algorithm suitable for wireless sensor networks," *2008 International Conference on Information Security and Assurance*, pp: 73-76.

[22] **Çakıroglu, M., Bayilmis, C., Ozcerit, A. T.** and **Cetin, O.** (2010) "Performance evaluation of scalable encryption algorithm for wireless sensor networks", *Scientific Research and Essays*, **5**(9): 856-861.

[23] **Sehrawat, D.** and **Gill, N. S.** (2018) "Lightweight Block Ciphers for IoT based applications: A Review", *International Journal of Applied Engineering Research*, **13**(5): 2258-2270.

[24] **Ferguson, N., Lucks, S., Schneier, B., Whiting, D., Bellare, M., Kohno, T.** and **Walker, J.** (2010) "The Skein hash function family", *Submission to NIST*, Round 3.

[25] **Lim, C. H.** and **Korkishko, T.** (2005) "mCrypton–a lightweight block cipher for security of low-cost RFID tags and sensors", *International Workshop on Information Security Applications*, pp: 243-258.

[26] **Ferreira, A. C., Vilaça, M. A., Oliveira, L. B., Habib, E., Wong, H. C.** and **Loureiro, A. A.** (2005) "On the security of cluster-based communication protocols for wireless sensor networks," *International Conference on Networking*, pp: 449-458.

[27] **Oliveira, L. B., Wong, H. C., Bern, M., Dahab, R., Habib, R., E.** and **Loureiro, A. A. F.** (2006), "SecLEACH-A random key distribution solution for securing clustered sensor networks", *IEEE International Symposium on Network Computing and Applications*, pp: 145-154.

[28] **Banerjee, P., Jacobson, D.** and **Lahiri, S. N.** (2007) "Security and performance analysis of a secure clustering protocol for sensor networks", *IEEE International Symposium on Network Computing and Applications*, pp: 145-152.

[29] **Abuhelaleh, M. A., Mismar, T. M.** and **Abuzneid, A. A.** (2008) "Armor-LEACH-energy efficient, secure wireless networks communication". *Proceedings of 17th International Conference on Computer Communications and Networks*, pp: 1-7.

[30] **Alshowkan, M., Elleithy, K.** and **AlHassan, H.** (2013) "LS-LEACH: a new secure and energy efficient routing protocol for wireless sensor networks", *International Symposium on Distributed Simulation and Real Time Applications*, pp: 215-220.

[31] **Kumar, S. R.** and **Umamakeswari, A.** (2016) "SSLEACH: Specification based secure LEACH protocol for Wireless Sensor Networks," *International Conference on Wireless Communications, Signal Processing and Networking*, pp: 1672-1676.

[32] **Goto, S., Saito, S., Kang, H.** and **Iwamura, K.** (2015) "New secure LEACH protocol using hierarchy-based preshared key scheme", *Computer Science and its Applications*, pp: 99-106.

[33] **Biham, E., Anderson, R.** and **Knudsen, L.** (1998) "Serpent: A new block cipher proposal," *International Workshop on Fast Software Encryption*, pp: 222-238.

[34] **Zhang, W., Bao, Z., Rijmen, V.** and **Liu, M.** (2015) "A New Classification of 4-bit Optimal S-boxes and its Application to PRESENT, RECTANGLE and SPONGENT", *International Workshop on Fast Software Encryption*, pp: 494-515.

[35] **NS-2**, "*The Network Simulator*," [online]. Available: https://www.isi.edu/nsnam/ns/

[36] **NS-2 Code Extensions**, "*MIT μAMPS Project*", [online]. Available: homepage:www.mtl.mit.edu/researchgroups/icsystems/uamps/research/leach/MIT_uAMPS_changes_ns.tar.gz

حماية آليات التوجيه في شبكات الاستشعار اللاسلكية ذات الكفاءة في استخدام الطاقة

ياسر رزيق السليهبي و **محمد حسيب ظفر** و **مديني العساف**

كلية الحاسبات وتقنية المعلومات، جامعة الملك عبدالعزيز، جدة، المملكة العربية السعودية
yselehibi@stu.kau.edu.sa

المستخلص. انتشرت في الالفية الجديدة شبكات الاستشعار اللاسلكية، وهي شبكة من أجهزة الاستشعار يتم استخدامها للأغراض العسكرية وتطبيقات المراقبة. وهي محدودة الموارد من حيث الطاقة وحجم الذاكرة والقدرة على إجراء الحسابات والاتصال مع بعضها البعض. وأهم مورد من هذه الموارد هو الطاقة، لأن استهلاكها بشكل حاد يؤثر على الشبكة ككل. وآليات التوجيه في هذه الشبكة تنقسم إلى ثلاثة أقسام، وهي الآليات المسطحة، والآليات المجمعة، والآليات المعتمدة على الموقع. وأفضلها من حيث عدم استهلاك الطاقة هي الآليات المجمعة. ويوجد العديد من آليات التوجيه المجمعة، ولكنها معرضة إلى العديد من الهجمات وتفتقر إلى الحماية، ولذلك في هذه المقترح البحثي اقتراح عمل إصدار من هذه الآلية يحتوي على طبقة حماية، مع لأخذ في الاعتبار مشكلة استهلاك الطاقة.

الكلمات المفتاحية: شبكات الاستشعار اللاسلكية، LEACH، تجمع، LEACH-MS، SLW-LEACH.