A Lightweight Authentication Framework for Wireless Sensor Networks

Original Scientific Paper

Hakeem I Mhaibes
Middle Technical University,
Kut Technical Institute, Computer Center
Kut, Wassit, Iraq
hakeem.emade@mtu.edu.iq

Shahnawaz Qadir
University of Kashmir,
IT Department
Hazratbal, Srinagar, J & k, India
sqadir@uok.edu.in

Abstract *– Wireless Sensor Network (WSN) is emerging as a dominant technology with its applications in areas like agriculture, communication, environment monitoring, and surveillance. The inherited vulnerability and resource-constrained nature of sensor nodes led researchers to propose many lightweight cryptographic protocols for WSN. These sensors are low-cost, low energy, have low processing capability and have low storage restrictions. WSN suffers from many risks because of these unique constraints. This paper proposes a new lightweight security framework for WSNs and covers different lightweight cryptographic schemes for WSN applications. The aim is to provide cryptographic primitives for integrity, confidentiality, and protection from the man-in-the-middle and reply attacks. The work is based solely on symmetric cryptography and it has four phases; Network Initialization, Node Initialization, Nodes Communication, and Node Authentication. This work adopts the Low-Energy Adaptive Clustering Hierarchy (LEACH) framework, which deploys random rotation to distribute the energy among a group of nodes. The probability of attacking in LEACH is higher at cluster head and member nodes. Therefore, data transmission among communicated nodes is encrypted over multiple levels of protection by dynamic session keys to provide a high level of security. In addition, an authentication ticket is provided by a cluster head for each authenticated node to identify another node. The session keys are dynamically generated and updated during the communication to prevent compromising or capturing the keys. Through simulation and evaluation of the system, the results showed less energy consumption and efficient cryptographic primitive were compared with existing schemes.*

Keywords*: Information Security, Lightweight Cryptography, Key Management, WSN, LEACH, Mutual Authentication*

1. INTRODUCTION

In recent years, WSN applications have become well-known modern technologies and developed rapidly. Applications such as home automation [1], smart cities [2], healthcare applications, RFID tags [3], and sensor networks [4] led many companies to shift from general-purpose devices to resource-constrained devices [5].

WSN consists of numerous sensor nodes arranged in an organized manner called a cluster. Sensors collect information from the environment which may include buildings, people, transportation pathways, electrical lines, weather, health care etc. Highly confidential information is collected by these sensors and passed on through insecure channels for emergency response [6] and decision making.

In General, these sensors are low-cost and low energy, besides having low processing capability, communication restriction and storage restriction. Meanwhile, complex conventional computational data encryption and public-key cryptography systems are not applicable over WSNs due to these limitations [7].

Theoretically, these sensors may expose to multiple attacks (such as eavesdropping, interception, modification, and tampering) due to the nature of their deployment and the communication mode. Therefore, the probability of an attack is higher at WSN nodes [8].

Authentication can be efficiently used to verify identity of nodes in order to ensure that only authorized nodes have access to the data. In most cases, external and non-authenticated nodes could access secure data

[9], and hence, may be alter data and threaten data security. Therefore, it is necessary to build a secure WSN system to keep data secure from access violation and unauthorized access [10].

In this regard, the inherited vulnerability and resource-constrained nature of sensor nodes led researchers to propose many lightweight cryptographic protocols for WSN [11], [12], [13], [14].

The security schemes and algorithms for WSN are mainly focused on in this paper, to design and develop a new lightweight authentication framework (LWAF) that has an effective authentication and key management scheme with low computing and energy cost. This work adopts the Low-Energy Adaptive Clustering Hierarchy (LEACH) which deploys random rotation to distribute the energy among a group of nodes [15] and is based solely on symmetric cryptography [16]. The objective of key management is to dynamically establish and maintain secure channels among communicating sensors.

The symbols used in this paper illustrated in table (1).

Table 1. Notation Symbol Table

Notations	Description
BS, CH	Base Station, Cluster Head
K_M	Master key
CK^R	Cluster key for round R
C_{id}	Cluster id
N	Number of nodes
id	Node id
r	Random number
SK_{AB}	Session key between node A and B
t	timestamp
Kid	Symmetric key between CH and a node
$[1]_K$	Encrypted message M with K
[*M*]	Hash of message M
$[M]^j$	Message M is hashed j times

The above notations are used as scientific convention texts for illustration in this paper for node-to-node agreement description.

2. LITERATURE REVIEW AND PROBLEM STATEMENT

This paper presents the results of many existing cryptographic WSN schemes that have been done to secure WSNs.

Authors in [17] propose Localized Combinatorial Keying (LOCK) which is dynamic key management (Exclusion-Based Systems (EBS)) for cluster-based WSN. LOCK uses three keys; administrative key, group session key and cluster session key. LOCK selects a special node as a keys generator. LOCK is suitable for static networks only, whereas, the proposed LWAF is intended for dynamic networks.

Paper [18] presents a WSN protocol that is capable to prevent Denial of Service (DoS) and replay attacks. Their scheme is based on symmetric cryptography, where the sensor nodes shared a common secret key. Any compromised node can threaten the network and can send forged data.

SPINS proposed by authors in [19], which has two security protocols, SNEP and TESLA. Where, SNEP is responsible for authentication, confidentiality, evidence for data, the TESLA is responsible to provide authentication to broadcast fresh data to many nodes in the local cluster. In this scheme, the mobility of nodes leads to a topology change in a random way, that affects the security of the WSN application. This paper adopts symmetric cryptography exchange protocol and HMAC algorithm to authenticate sensor mobility in the network.

Authors in [20] considered DoS attack among multihop data transmission paths. In General, WSN is a tree structure, therefore, an attack on the node path affects the connected branches. They proposed a one-way hash chain (OHC) mechanism to prevent the connected paths and also a secure end-to-end communication of multihop data transmission path by adding a number to OHC to each transmitted message. Therefore, 8 bytes are added that cause an extra overhead on the resources-constrained devices.

Authors in [21] proposed BROadcast Session Key (BROSK) which is established through third party entities and thus lacks trust. This scheme is considered as an adhoc management scheme, where a specific node can communicate with a neighbor node for exchanging a session key. BROSK consumes less energy power by reducing data transmission as compared with SPINS.

Paper [22] developed a protocol to prevent replay attacks by maintaining a monotonical increment counter to keep track of all previous replayed messages. Here, each node has a counter that stores time information. This mechanism requires a huge amount of memory for memory-constrained sensor nodes.

Authors in [23] proposed AKMS, which consists of three phases; a key pre-distribution phase, a network initialization phase, and an authentication phase. AKMS solves the problem of malicious nodes that attack nodes during their transmission processes. The keys are dynamically generated and updated during the network communications to provide more protection and at the same time, it provides the ability for a new node to authenticate and enter into the cluster.

Authors in [24] proposed a key agreement protocol to protect a server from DoS attacks in a hierarchical WSN. In the protocol, the first level hierarchy is made by BS, the second level hierarchy is made by CHs and the third level is made by MNs. Each node in this scheme has its own built-in key used for key generation. The scheme uses timestamp with each mutual authentication and provides node authentication among hierarchy levels. Athough the researchers provide node authentication, but other security primitives are not considered.

Meanwhile, battery, communication bandwidth, computation complexities and memory constraints are major concerns in WSN. Therefore, providing security and authentication for these applications is crucial in open channel communications [25]. Authentication can be efficiently used to check reliable, fake and altered communication. In most cases, the external and non-authenticated nodes are interested in the data collected by the sensor node [9]. If these nodes access the data, its integrity and confidentiality can be compromised. Thus, it is necessary to stop data violation and unauthorized access [10]. Many other lightweight cryptographic schemes are proposed to overcome these limitations [26] [27].

3. THE AIM AND OBJECTIVES OF THE STUDY

The study is aimed at providing cryptographic primitives such as integrity, confidentiality and protection from the man-in-the-middle and reply attacks for WSN applications and emphasizes on CH selections to save node battery and make the network more reliable.

To achieve this, the following objectives need to be accomplished:

- Develop a lightweight authentication framework (LWAF), where the secret messages are encrypted by session keys to provide a high level of security. These session keys are dynamically generated and updated during the communication between nodes to prevent compromising or capturing the keys.
- Generate and use an authentication ticket provided by a CH to authenticate every node in the cluster. These tickets are used by a node to identify another node in the cluster.
- Adopt LEACH framework for dynamic WSN, in which the energy is distributed in an equal manner for all connected nodes in the cluster setup to focus on CH selection in order to save energy.

4. MATERIALS AND METHODS

In the proposed LWAF, a BS node can be located far away from the rest of the homogeneous nodes. In Fig. 1, the framework architecture of the proposed scheme is illustrated. The nodes are arranged in groups called clusters, where a specific node is selected as a controller node (CH) based on certain probabilities. The rest of the nodes are called member nodes (MNs) that send data to a corresponding CH, where a CH forwards data to the BS. Here, CH is responsible for the registration of MNs. We employ LEACH framework; the implementation of LEACH can reduce energy up to 8 factors when compared with traditional routing schemes.

Features of LEACH:

- Coordination and control for cluster set-up are done locally.
- Base stations or cluster heads are selected randomly.
- Data is compressed locally to reduce the amount of transmitted data [6].

Clustering is useful in WSN whereas the data travel small distances between the surrounding nodes in the same cluster. In addition, a node can determine the nearest CH and joins the cluster to reduce the amount of energy for data transmission. See Fig. 2.

In general, selecting a node as a CH in WSN applications drains its battery. But with LEACH framework, it dynamically spreads this energy over multiple nodes in the cluster, therefore, the CH nodes are not fixed and self-selected at different time intervals. For example, at time t1, a group of nodes might select themselves as cluster-heads and at t2 a new group of nodes selects themselves as another cluster-heads.

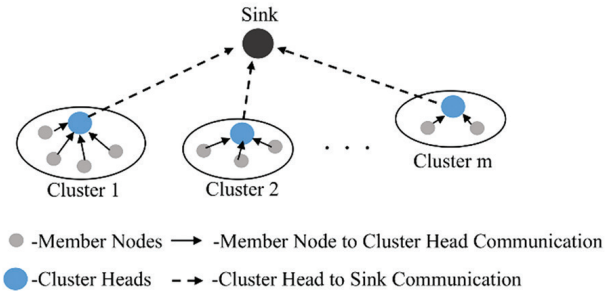

Fig. 1. WSN framework

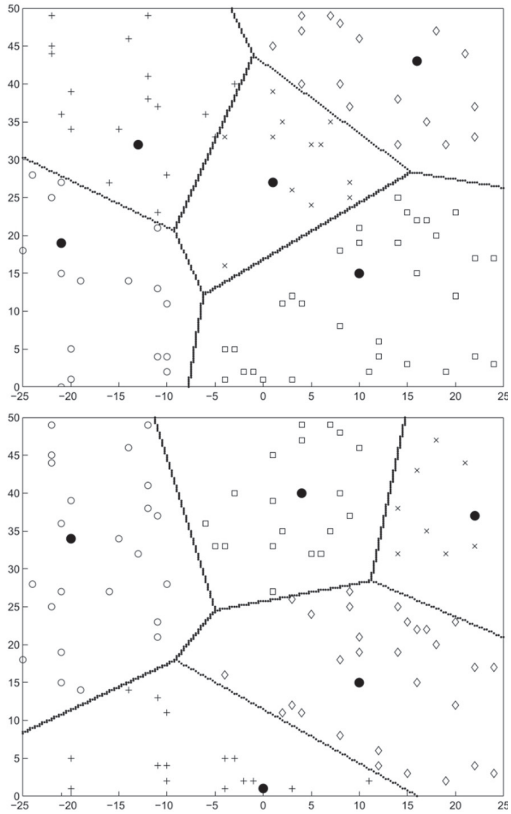

Fig. 2. Clustering in WSN: a - Group of nodes selected themselves as a CHs at time t1; b - Group of nodes selected themselves as a CHs at time t2. CH node is marked with a •, and all given symbol belong to same CH [6]

4.1. RESEARCH MATERIAL AND SIMULATION TOOLS

In this work, MATLAB (R2020a) was used as a software platform on Windows 10. Complete WSN system was built using MATLAB/Simulink software. The simulation process consists of building the nodes hardware architecture, model the communication channel, and receive master node architecture to analyze and evaluate the proposed LWAF. Bluetooth is used as a backbone to undertake communication of the physical layer. Bluetooth technology operates in a short-range radio with 2.4 GHz. The clustering techniques of this scheme are based on LEACH framework. The encryption class used in this scheme is based on symmetric cryptography only.

4.2. METHODS OF THE PROPOSED SCHEME

The general architecture of the proposed LWAF scheme includes four phases as given below:

4.2.1 Network Initialization Phase

This phase is enabled during network deployment, where each CH is selected and registered in the network. This phase assumes each node stores a master key K_M and it has to be long enough to tolerate against crack. This master key is 128 bits and is stored securely. At firsts, Round $R=0$;

A BS node starts to select n of *CH's* depending on their battery lifetime and positions and marks it with a unique number CH_n.

1. Each CH Sends a message to its BS separately, including the number of its node N and a cluster identity Cid, which is encrypted by a common master key.

$$\{C_{id}, N\}_{k_M}$$

BS now, received the encrypted message separately and uses k_M to decrypt them and generate a cluster key CK^R by hashing $CK^R=[C_{id},K_M, N]$ and send back to each individual *CH* an encrypted message containing its id and a private message encrypted by a corresponding cluster key CK^R.

$$\{C_{id}, M\}_{CK^R}$$

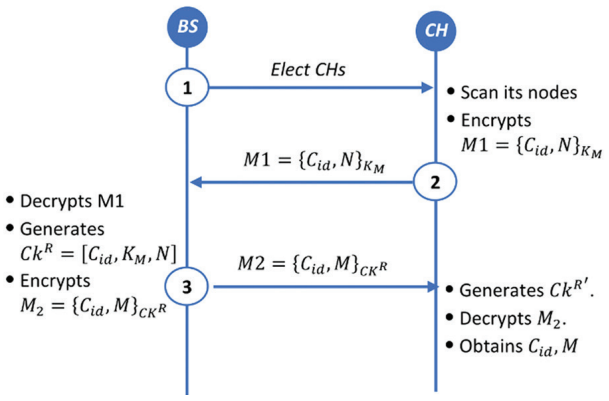

Fig. 3. Network initialization Phase.

2. Each corresponding CH receives an individual encrypt message from BS and uses its information to generate its cluster key. Then it decrypts the incoming message and checks the integrity of its identity and a common master key with a saved key. If it is equal, then the CH start a deployment process and the first phase is done, otherwise it cancels. Fig. 3. illustrates the whole processes of information exchange in this phase.

4.2.2. Node Initialization Phase

This phase is enabled when a node registers itself into the cluster.

1. For each node in the cluster;
   - Every node sends a requested message to its CH, this message contains a random number and its identity encrypted by a common master key such as:

$$\{r_i, id\}_{k_M}$$

2. After receiving massage by a CH;
   - It decrypts the incoming message and obtains node identity and a random number.
   - Generates a corresponding node key k_id by using a hash function of a requested message and a node id such as

$$k_{id} = [r_i, id].$$

   - And then encrypts two messages, the first encrypted message contains the round number, number of nodes, a cluster id and a cluster key which is encrypted by a corresponding key node such as

$$\{R, N, C_{id}, CK^R\}_{k_{id}}$$

   - The second encrypted message is a ticket T_id. This message contains the node's identity and is used to identify another node. This ticket is encrypted by its cluster key

$$\{Nodes\ idenity\}_{Ck^R}$$

These messages are sent back to the requesting node.

3. When a specific node received the incoming two messages, it does the following:
   - Generates a corresponding key node k'_{id} using the same hash function and checks the validity.
   - If it is equal, then it proceeds and decrypts the first message and obtains R,N,C_{id},and CK^R .
   - It uses the cluster key to decrypts the authentication ticket and obtains node's identities.
4. At this stage, each node stores other nodes identities, R, CKR, N, Cid and its kid and can communicate with other nodes in specified cluster.

5. After all nodes have been registered and authenticated. Now R=2, this means the authentication transmits to next round by hashing CKR and sets R=R+1, such as:

$$CK^{R+1} = [CK^R]$$

This obtained key is used for the next authentication round. See fig. 4.

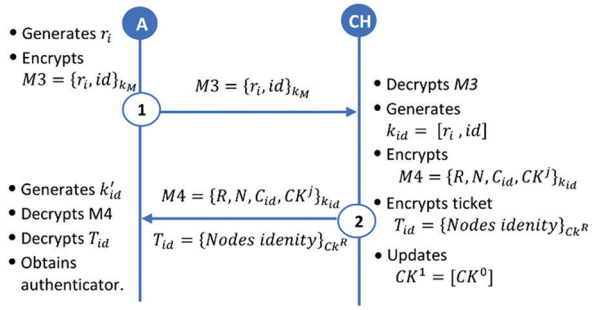

Fig. 4. Node Initialization Phase.

4.2.3 Node Communication Phase

At this stage, each node can find its neighbor and communicate within a safe mode. Mutual authentication protocol and specific excitation/response processes performed as follows:

1. If a node, A, wants to communicate with a node B, then a specific operation will perform:
   - Node A generates a common session key with B by hashing its identity, B's identity, and a common C_{id}. This key used to encrypt their further messages. sk_{AB}= $[C_{id}, A, B]$.
   - Generate a nonce and encrypts $M_5 = \{r_i\}_{sk_{AB}}$
   - Send an encrypted message (A, M_5) to B.
   - An attacker wanting to listen to the date being transmitted will get encrypted values.
2. When receiving a message from A, B performs the following operations:
   - Checks the identity of A.
   - If A exists, it generates the corresponding common session key using its common information to obtain sk_{AB}.
   - And, it decrypts the $\{r_i\}_{sk_{AB}}$ and obtain a nonce.
   - Then, it encrypts response M_6, which includes a nonce and timestamp $M_6 = \{r_i, t_1\}_{sk_{AB}}$, This timestamp is used for mutual authentication.
   - Then, B sends M_6 to A.

   After receiving M_6, node A will perform:
   - Decrypts M_6.
   - And checks its random number and the timestamp with a specific limit. If the current time $t_{(curr)}$- t_1<=Δt, otherwise stop the connection.

Now, each node can communicate with their neighbor nodes through their common session key.

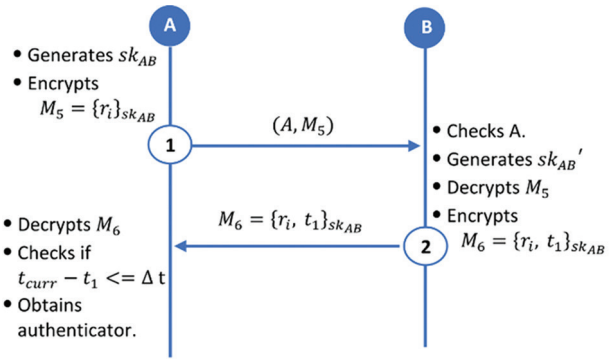

Fig. 5. Nodes Communication Phase.

4.2.4. Node Authentication Phase.

This phase is employed when a new node wants to join a network. Nodes use the authenticated ticket to authenticate and verify a new node. A pre-calculation for excitation/response based on mutual authentication protocol is done to avoid a situation of storing a common cluster key in the internal memory because keeping a common cluster key could face the network to a serious problem. Therefore, the LWAF suggests that a new node is verified using knowledge of the common master key of previous round R. Such that, the authentication of round 2, is constructed using a key derivation of round 1 without storing the previous key.

In this situation, when an attacker destroys a node and obtains its master key of the current round, he cannot compromise the identity and authentication messages of previous operations.

For example, if a new node C, wants to join the network, then it will start at a new round. Suppose a network is at the fourth round. Then the whole operation of this phase is as follows:

1. Node generates a random number ri and send a cipher request message to its surrounding CH. This message contains a random number and its identity encrypted by a common master key such as:

$$M_7 = \{r_i, id\}_{k_M}$$

2. CH decrypts M7, and obtains node identity and a random number.

$$k_{id} = [r_i, id].$$

   - Then, encrypts two messages, the first encrypted message contains a round number, number of nodes, a cluster id, and a cluster key which is encrypted by a corresponding key node such as:

$$M_8 = \{R, N, C_{id}, CK^j\}_{k_{id}}$$

   - The second encrypted message is a T_{id}. This ticket is encrypted by its current cluster key Ck^R and contains nodes identities in its cluster.

$$\{Nodes\ idenity\}_{Ck^R}$$

- These messages are sent back to the requesting node.

3. When *C* receives the two messages, it does the following:

- Generates a corresponding key node k'_{id} using the same hash function and checks the validity of its random number.
- If it is equal, then it proceeds and decrypts the first message and obtains *R, N,* C_{id}, and CK^R.
- Now, *C* hashes the *CK*, *R* times such as $CK^R = [CK]^{R-1}$, to synchronize with the current round.
- It uses CK^R to decrypts the authentication T_{id} and obtains node's identities.

Now, *C* has node's identities, *R,* CK^R, *N,* C_{id} and its k_{id} and can communicate with other nodes in specified cluster.

Node *C* acts as a requester and *CH* as an authenticator in above scenario. T_{id} and *R* parameters are used by *CH* to authenticate a node. In above case, the authenticator is at fourth round, but *C* is at the first round. Therefore, *C* should hash the key $[CK]^{R-1}$ times to synchronize with the current key. Fig. 6. shows the full operation of this phase.

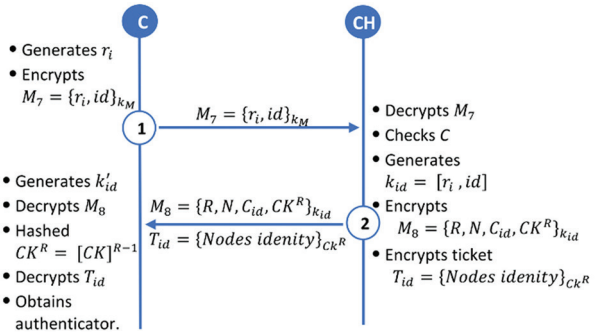

Fig. 6. Authenticate a new node C.

5. RESULTS OF LWAF DEVELOPMENT

The results are illustrated in terms of system simulation. The proposed scheme is based on symmetric cryptography only to overcome the complex computation of public-key cryptography since all nodes are resource constraints. Results are discussed in terms of stated aim and objectives, which encompasses the following:

5.1. DEVELOPING LWAF

Proposed LWAF scans sensors and elects *CH*s depending on their positions and corresponding battery life. It also creates cluster and scans its nodes and encrypts messages. The secret messages are encrypted by session keys. The session keys are dynamically generated and updated during the communication. This encrypted message contains cluster identity and a number of surrounding nodes. BS receives the encrypted message separately and uses its k_M for decryption, and creates a cluster key. Each corresponding *CH* receives an individual encrypt message from *BS*. Then, it decrypts the incoming message and checks the integrity of its identity and a common master key with a saved key. If it is equal, then the *CH* starts a deployment process and the first phase is done, otherwise, it cancels.

At the second phase, a node registers itself into the cluster by sending $\{r_i, id\}_{K_M}$. *CH* obtains node identity and a random number to create a corresponding node key $K_{id} = [r_i, id]$. *CH* lets a node to know its round, its neighbor nodes, and general cluster information $\{R, N, C_{id}, CK^R\}_{K_{id}}$. The second encrypted message is a ticket T_{id} that contains node's identity $\{node\_identity\}_{CK^R}$. A node cannot obtain the ticket unless it is registered in the network. At this stage, each node stores other node's identities, *R,* CK^R, *N,* C_{id} and its K_{id}, and can communicate with other nodes in a specified cluster. Then transmits to the next round by hashing CK^R and sets *R=R+1*, such as $CK^{R+1} = [CK^R]$.

The third phase ensures the security of message delivery. If node *A* wants to send a message to its neighbor *B*, each node must be updated and be synchronized with the current network round to ensure $CKR=[CK]^{R-1}$, then, mutual authentication protocol and specific excitation/response processes are performed to generate their common session key $sk_{AB}=[C_{id}, A, B]$ with a specific time $t_{cur}-t_1<=\Delta t$.

The fourth phase is enabled when a new node wants to join a network. Ticket authentication and a precalculation for excitation/response is done (requester and authenticator) $M_7 = \{r_i, id\}_{k_M}$, $M_8 = \{R, N, C_{id}, CK^j\}_{k_{id}}$, and $T_{id} = \{node\_identity\}_{CK^R}$. A new node *C*, is verified using knowledge of a common master key of previous round $CK^R=[CK]^{R-1}$ and using $k_{id}=[r_i, id]$. Such that, the authentication of a round 2, is constructed using a key derivation of round 1 without storing the previous key.

5.2. USING AUTHENTICATION TICKET

The novelty of this work is by adding an authentication ticket $T_{id} = \{node\_identity\}_{CK^R}$. Only registered node has this ticket, this ticket is created by a corresponding CH and sent to the authenticated node to ensure node authentication. It contains all the node identities in its cluster and is used to identify another node. This ticket is encrypted by its cluster key.

5.3. ADOPTING LEACH FRAMEWORK

Since the work is based on LEACH framework, the selection of CH is a major concern to keep a network live and reliable. This scheme emphasizes CH selection by updating the network dynamically. It spreads the CH's energy selection over multiple nodes, therefore, the CH nodes are not fixed and are self-selected at different time intervals.

Evaluation of this framework is carried out in order to measure the performance of the LWAF scheme, packet delivery rate [28], energy consumption, and access rate in presence of different types of attackers [29].

Simulation of the scheme is performed using MATLAB (R2020a). We considered 500 nodes. 20 runs of the system were carried out for different scenarios. Simulation parameters of the proposed system are shown in Table 2.

Table 2. Simulation Parameters.

Parameters	Values
Area size (m2)	500 × 500
Wireless bandwidth (Mbps)	2
Simulation duration	(sec) 300
Initial energy (J)	CH = 50, SN = 5
Initial *V* BP (J)	CH = 500, SN = 50
Radio range(m)	CH = 150, SN = 50
Number of *CHs*	6% of nodes

In order to evaluate the proposed work, at first, packet delivery rate (PDR) is simulated based on the mentioned parameters in Table 2, and compared with two existing competing schemes SPINS [19] and LOCK [17], as shown in Fig. 7. Malicious nodes for MNs and CH are eliminated by bidirectional malware detection. LWAF effectively reduces packets errors.

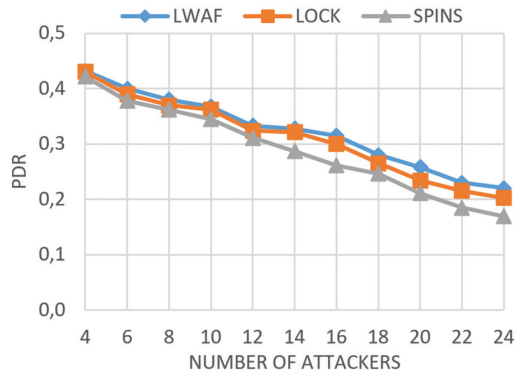

Fig. 7. Packet Delivery Rate.

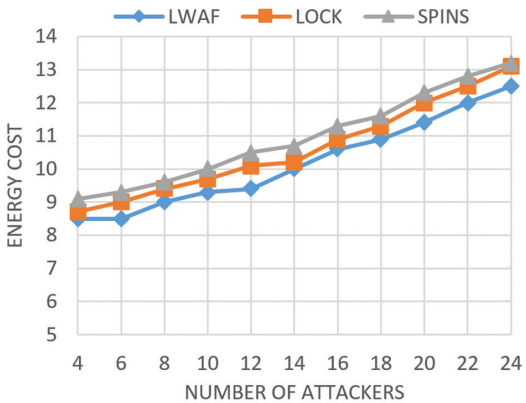

Fig. 8. Average Energy Consumption

As compared with LOCK and SPINS, more packets are sent to a destination sensor.

In Fig. 8 the average energy consumption for all nodes is measured during transmission in terms of sending, receiving and calculation complexity, and compared with LOCK and SPINS.

The average energy consumption increases when the number of attackers will increase and errors increase also, due to increase in error packets.

Fig. 9 shows the evaluation of network resilience ability. Number of nodes being evaluated are 500 and 25 % of them being attackers. The proposed system detects malicious nodes and can exclude them from the network.

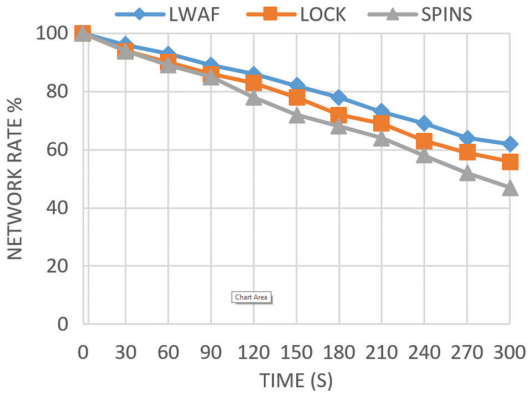

Fig. 9. Access Network Rate.

In addition, session key is dynamically re-generated and distributed to avoid node capturing by malicious nodes. Malicious nodes are unable to compromise other nodes because they don't have the master keys.

6. DISCUSSION OF EXPERIMENTAL RESULTS

Proposed LWAF ensures a CH has all the identities of surrounding nodes and uses a dynamic session key to encrypt each transmitted message. The key is dynamically generated. Therefore, if an attacker tries to compromise the previous key, he cannot either obtain the current session key or cannot synchronize with the current round, this done with the help of a simple calculation timestamp.

In addition, introducing the idea of the ticket increases security level, since it contains all nodes' identities. And hence, a node cannot obtain the ticket unless it is authorized and registered in the network.

Adopting the LEACH framework reduced energy consumption and ensured mobile nodes, therefore, increases system performance.

Practically, the average energy consumption increased when the number of attackers increased on the network, and hence, error packets increased also. A CH in LWAF filters out these errors and avoiding packets spreading to the attackers in the network and hence reduce energy consumption. In contrast with other schemes, they must initiate updating of the keys, which consume more energy.

Through simulation, the performance is better than SPINS and LOCK where it can avoid malicious nodes.

Besides, LWAF adopts multipath propagation routing technology to eliminate the selective forwarding attacks, which makes the PDR even higher.

The limitations of the proposed LWAF are the cryptographic algorithms used in the computation procedures since it is based only on symmetric cryptography. There may be other WSN environments where this framework is not applicable.

The disadvantages of the LWAF scheme includes; the fact that the scheme is affected by the number of nodes. Since the system provides scalability and mobility, increasing sensor nodes is inversely proportional to system performance.

7. CONCLUSION

1. This work emphasizes lightweight cryptographic systems and their important aspects in WSN. We developed LWAF with the intention to provide security primitives in WSN for all phases mentioned above. The system generates fresh random session keys for every authentication between BS, CHs, and MNs to prevent attackers. The simulation showed that; it provides a more efficient security primitive in less power consumption as well as communication complexity overhead as compared with other existing WSN schemes.

2. Generation of ticket is done only by an authorized CH. The novelty of using this ticket is to prevent an attacker from entering into the system. Taking into account, the only authenticated node receives this message through secure mutual excitation/response processes. A node can use this ticket to communicate securely with others in the cluster. Through system simulation, these tickets provide a high level of security.

3. Adopting LEACH framework hierarchy has made the system more reliable for dynamic WSNs and efficient in terms of energy-saving, low complexity overhead and scalability and robustness. The amount of data that must be transmitted to the BS is less compared with others. LWAF focuses on CH selection to distribute energy in an equal manner. Through the study of different WSNs, the selection of CH is a big concern, because a fixed CH drains its battery and hence archive reduction of energy.

Future work in this direction shall focus on applying additional procedures in terms of using asymmetric cryptography and other cryptographic algorithms and focus on the way to resist multiple attacks, robust routing. Besides, the future work shall consider the aspects to reduce the packet transmission time, latency and packet overheads.

8. REFERENCES

[1] S. G. Fatima, S. K. Fatima, "Home Automation System with WSN and IoT", International Journal of Advanced Research in Engineering and Technology, Vol. 10, No. 2, 2019, pp. 78-85.

[2] M. Peira, J. Fernando, Rocher, Javier, Perra, Lorena, Sendra, Lloret, Jaime, M. Ablanque, P. Vicente, "Autonomous WSN for lawns monitoring in smart cities", Proceedings of the 14th IEEE/ACS International Conference on Computer Systems and Applications, 30 October - 3 November 2017, pp. 501-508.

[3] A. Juels, "RFID security and privacy: A research survey", IEEE Journal on Selected Areas in Communications, Vol. 24, No. 2, 2006, pp. 381-394.

[4] K. Chintalapudi, T. Fu, J. Paek, N. Kothari, S. Rangwala, J. P. Caffrey, R. Govindan, E. Johnson, S. Masri, "Monitoring civil structures with a wireless sensor network", IEEE Internet Computing, Vol. 10, No. 2, 2006, pp. 26-34.

[5] J. Yick, B. Mukherjee, D. Ghosal, "Wireless Sensor Network Survey", Computer networks, Vol. 52, No. 12, 2008, pp. 2292-2330.

[6] W. R. Heinzelman, A. Chandrakasan, H. Balakrishnan, "Energy-efficient Communication Protocol for Wireless Microsensor Networks", Proceedings of the Hawaii International Conference on System Sciences, Maui, Hawaii, 4-7 January 2000, Vol. 2, pp. 10.

[7] M. Elhoseny. A. E. Hassanien, "Secure Data Transmission in WSN: An Overview", Dynamic Wireless Sensor Networks, Vol. 165, 2019, pp. 115-143.

[8] S. E. L. Khediri, N. Nasri, A. Wei, A. Kachouri, "A new Approach for Clustering in Wireless Sensors Networks Based on LEACH", Procedia Computer Science, Vol. 32, 2014, pp. 1180-1185.

[9] J. Sen, "A Survey on Wireless Sensor Network Security", International Journal of Communication Networks and Information Security, Vol. 1, No. 2, 2009, pp. 55-78.

[10] A. Ghosal, S. D. Bit, "A Jamming-Attack-Defending Data Forwarding Scheme Based on Channel Surfing in Wireless Sensor Networks", Security and Communication Networks, Vol. 6, No. 11, 2013, pp. 1367-1388.

[11] K. Haseeb, I. U. Din, A. Almogren, N. Islam, "An Energy Efficient and Secure IoT-Based WSN Framework: An Application to Smart Agriculture", Sensors, Vol. 20, No. 7, 2020, pp. 2081.

[12] A. Ghosal, S. DasBit, "A Lightweight Security Scheme for Query Processing in Clustered Wireless Sensor Networks" Computers & Electrical Engineering, Vol. 41, 2015, pp. 240-255.

[13] R. R. K. Chaudhary, K. Chatterjee, "An Efficient Lightweight Cryptographic Technique for IoT based E-healthcare System", Proceedings of the 7th International Conference on Signal Processing and Integrated Networks, Noida, India, 27-28 February 2020, pp. 991-995.

[14] C. Buratti, A. Conti, D. Dadari, R. Vedone, " An Overview on Wireless Sensor Networks Technology and Evolution", Sensors, Vol. 9, No. 9, 2009, pp. 6869-96.

[15] N. G. Palan, B. V. Barbadekar, S. Patil, "Low Energy Adaptive Clustering Hierarchy (LEACH) Protocol: A Retrospective Analysis", Proceedings of the International Conference on Inventive Systems and Control, Coimbatore, India, 19-20 January 2017, pp 1-12.

[16] S. Chandra, S. Bhattacharyya, S. Paira, S. K. Alam, "A Study and Analysis on Symmetric Cryptography", Proceedings of the International Conference on Science Engineering and Management Research, Chennai, India, 27-29 November 2014, pp 1-8.

[17] M. Eltoweissy, M. Moharrum, R. Mukkamala, "Dynamic key management in sensor networks", IEEE Communications Magazine, Vol. 44, No. 4, 2006, pp. 122-130.

[18] D. Liu, P. Ning, "Efficient Distribution of Key Chain Commitments for Broadcast Authentication in Distributed Sensor Networks", North Carolina State University, Department of Computer Science, 2002.

[19] A. Perrig, R. Szewczyk, J. D. Tygar, V. Wen, D. E. Culler, "SPINS: Security Protocols for Sensor Networks", Wireless Networks, Vol. 8, No. 5, 2002, pp. 521-534.

[20] J. Deng, R. Han, S. Mishra, "Limiting DoS Attacks During Multihop Data Delivery in Wireless Sensor Networks", International Journal of Security and Networks, Vol. 1, No. 3, 2006, pp. 167-178.

[21] B.-C. C. Lai, D. D. Hwang, S. P. Kim, I. Verbauwhede, "Reducing Radio Energy Consumption of Key Management Protocols for Wireless Sensor Networks", Proceedings of the International Symposium on Low Power Electronics and Design, Newport Beach, CA, USA, 11-11 August 2004, pp. 351-356.

[22] H. Soroush, M. Salajegheh, T. Dimitriou, "Providing Transparent Security Services to Sensor Networks", Proceedings of the International Conference on Communications, Glasgow, UK, 24-28 June 2007, pp. 3431-3436.

[23] D. Qin, S. Jia, S. Yang, E. Wang, Q. Ding, "A Lightweight Authentication and Key Management Scheme for Wireless Sensor Networks", Journal of Sensors, Vol. 2016, 2016, pp. 1-9.

[24] R. Nanda, P. V. Krishna, "Mitigating Denial of Service Attacks in Hierarchical Wireless Sensor Networks", Network Security, Vol. 2011, No. 10, 2011, pp. 14-18.

[25] J. Sen, "A Survey on Wireless Sensor Network Security", International Journal of Vommunication Networks and Information Security, Vol. 1, No. 2, 2009, pp. 55-78.

[26] J. Daemen, S. Hoffert, M. Peeters, G. V. Assche, R. V. Keer, "Xoodyak, a Lightweight Cryptographic Scheme", IACR Transactions on Symmetric Cryptology, Vol. 2020, No. S1, 2020, pp. 60-87.

[27] M. Kocakulak, I. Butun, "An Overview of Wireless Sensor Networks Towards Internet of Things", Proceedings of the 7th Annual Computing and Communication Workshop and Conference, Las Vegas, NV, USA, 9-11 January 2017, pp. 1-6.

[28] J. Dong, K. E. Ackermann, B. Bavar, C. N. Rotaru, "Mitigating Attacks Against Virtual Coordinate-Based Routing in Wireless Sensor Networks", Proceedings of the 1st ACM Conference on Wireless Network Security, 31 March 2008, pp. 89-99.

[29] A. Mitra, A. Banerjee, W. Najjar, D. Zeinalipour-Yazti, V. Kalogeraki, D. Gunopulos, "High Performance, Low Power Sensor Platforms Featuring Gigabyte Scale Storage", Proceedings of the 3rd International Workshop on Measurement, Modeling, and Performance Analysis of Wireless Sensor Networks, San Diego, CA, USA, 21 July 2005, pp. 148-157.