S-FEAR: Secure-Fuzzy Energy Aware Routing Protocol for Wireless Sensor Networks

Iman Almomani[1 2] and Maha Saadeh[2]
[1] Computer Science Department, College of Computer and Information Sciences, P.O, Box 66833, Prince Sultan University, Riyadh 11586, KSA
[e-mail : imomani@psu.edu.sa]
[2] Computer Science Department, King Abdullah II School for Information Technology,
P.O. Box 13835, The University of Jordan, Amman 11942, Jordan
[e-mail: i.momani@ju.edu.jo, maha.k.saadeh@gmail.com]
*Corresponding author: Iman Almomani

Received July 27, 2017; revised October 8, 2017; accepted November 15, 2017; published April 30, 2018

Abstract

Secure routing services in Wireless Sensor Networks (WSNs) are essential, especially in mission critical fields such as the military and in medical applications. Additionally, they play a vital role in the current and future Internet of Things (IoT) services. Lightness and efficiency of a routing protocol are not the only requirements that guarantee success; security assurance also needs to be enforced. This paper proposes a Secure-Fuzzy Energy Aware Routing Protocol (S-FEAR) for WSNs. S-FEAR applies a security model to an existing energy efficient FEAR protocol. As part of this research, the S-FEAR protocol has been analyzed in terms of the communication and processing costs associated with building and applying this model, regardless of the security techniques used. Moreover, the Qualnet network simulator was used to implement both FEAR and S-FEAR after carefully selecting the following security techniques to achieve both authentication and data integrity: the Cipher Block Chaining-Message Authentication Code (CBC-MAC) and the Elliptic Curve Digital Signature Algorithm (ECDSA). The performance of both protocols was assessed in terms of complexity and energy consumption. The results reveal that achieving authentication and data integrity successfully excluded all attackers from the network topology regardless of the percentage of attackers. Consequently, the constructed topology is secure and thus, safe data transmission over the network is ensured. Simulation results show that using CBC-MAC for example, costs 0.00064% of network energy while ECDSA costs about 0.0091%. On the other hand, attacks cost the network about 4.7 times the cost of applying these techniques.

Keywords: Wireless Sensor Network, S-FEAR, Energy, Tree, Routing Protocols, Network Security

1. Introduction

WSNs have a wide range of civilian and military applications including area and battlefield monitoring, environmental/earth sensing, manufacturing/industrial monitoring, health care, smart homes/cities/transportation, and the Internet of Things (IoT), among others. Owing to the significant use of WSNs in such applications, which could involve the transmission of extremely sensitive information, maintaining data security becomes a major challenge in the implementation of WSN protocols. Consequently, it is important to consider security issues and possible threats in any protocol design. The reliability of the sensed data is reduced when few security requirements are in place. Breaking data security not only diminishes the importance of data but also affects the services provided by triggering false alarms and wrong reactions [1].

WSNs exhibit many characteristics such as low cost deployment, decentralized nature, easy setting up and tearing down of the network, multi-hop communication, self-configuration and routing, as well as limited resources in terms of energy, processing, memory, and communication bandwidth, which make them attractive in many application areas. However, such characteristics also introduce many challenges. In addition, WSNs are open-air in nature, and hence more vulnerable to security attacks [2-6]. To this end, proposing an energy-efficient secure routing protocol to prolong the lifetime of WSNs, while taking into consideration their characteristics and limited resources is a challenge.

The Fuzzy-based Energy Aware Routing (FEAR) protocol [2] was reviewed and studied as a part of the research described in this paper. A vulnerability assessment was performed and the resulting security model was proposed and mathematically analyzed. Subsequently, this model was implemented by applying security enforcers to produce a Secure FEAR (S-FEAR) protocol. The focus in this paper is to achieve both authentication and data integrity. Thus, both the Cipher Block Chaining-Message Authentication Code (CBC-MAC) and Elliptic Curve Digital Signature Algorithm (ECDSA) techniques were applied to the FEAR protocol and evaluated to measure their energy requirements and capabilities in preventing dangerous attacks, as well as limiting their occurrence. The use of these techniques protects the constructed routing topology among network nodes. Consequently, this topology can be used for safe data transmission. The FEAR protocol was chosen due to its proven efficiency in solving the problems of the Tree Routing protocol [7] and other tree-based protocols [8-10] in terms of reducing the number of messages exchanged among network nodes. In addition, it provides an energy-efficient solution to solve node or link failure problems. By applying energy-efficient security schemes to the FEAR protocol, a well-structured, energy-balanced secure routing protocol was built to protect the network from various security attacks that threaten the services it provides.

The rest of this paper is organized as follows: section 2 discusses the related work. Section 3 presents an overview of the FEAR protocol. The S-FEAR protocol is proposed in section 4, including a detailed FEAR attack analysis, S-FEAR security requirements, S-FEAR mathematical cost analysis and implementation. The S-FEAR evaluation results are presented and discussed in section 5. Finally, the paper is concluded and possibilities for future research are presented in section 6.

2. Related Work

This section summarizes the secure routing protocols in WSNs. Abuhelaleh et al. [11] proposed an armor leach protocol as a cluster-based WSN to secure the LEACH protocol using mechanisms to achieve authentication, confidentiality and integrity. Klaoudatou et al. [12] proposed another cluster-based security framework where two different scenarios for infrastructure and infrastructure-less WSN environments were described.

Pathan and Hong [13] proposed a secure routing protocol for tree-based networks by ensuring authentication and confidentiality using a one-way hash chain and a preloaded key. A sink rooted tree was constructed for use in the routing process. To ensure the authenticity of the transmitted data, all intermediate nodes were initialized with a basic one-way hash chain number during the tree construction. Additionally, the authors assumed that no node could be compromised during the tree construction.

Another secure protocol for tree based routing was proposed by Papadimitriou et al. [14]. This protocol uses Public Key Cryptography (PKC) to protect the network against sinkhole attacks by signing the message to prove its identity.

To evaluate security techniques, Söderlund [15] studied the impact of adding a message authentication code on the lifetime of the WSN based on different symmetric and asymmetric approaches. The results showed that the lifetime of a sensor node is not affected by the addition of authentication. In addition, the results also demonstrated that using symmetric-based authentication is more efficient than asymmetric approaches.

In the studies conducted by Ren et al. [16] and Yeh et al. [17], the authors presented asymmetric-based authentication schemes. Ren et al. [16] used different cryptographic techniques, such as a Merkle hash tree and an identity-based signature scheme to achieve immediate broadcast authentication and to minimize the computation and communication cost. Yeh et al. [17] provided an Elliptic Curve Cryptography (ECC)-based solution to achieve authentication. Gupta et al. [18] proposed another ECC authentication scheme.

This paper focuses on enhancing energy efficient tree-based routing protocols in WSNs, and mainly, the FEAR protocol, by applying security schemes to ensure a secure topology is constructed among the network nodes. In so doing, this topology can be used for safe data transmission. The FEAR [2] protocol has been chosen due to its efficiency in solving the challenges of tree-based routing protocols in terms of reducing the number of messages exchanged among network nodes. In addition, it provides a power efficient solution to solve node or link failure problems. The FEAR protocol has been reviewed, studied, analyzed, and compared with many other WSN routing protocols [19-31]. However, none of these studies have tackled the security aspects of this protocol. By applying energy-efficient security schemes to FEAR, a secure, well-structured routing protocol is constructed to protect the network from different threats and attacks, as well as provide many security services including authentication, data integrity, and confidentiality.

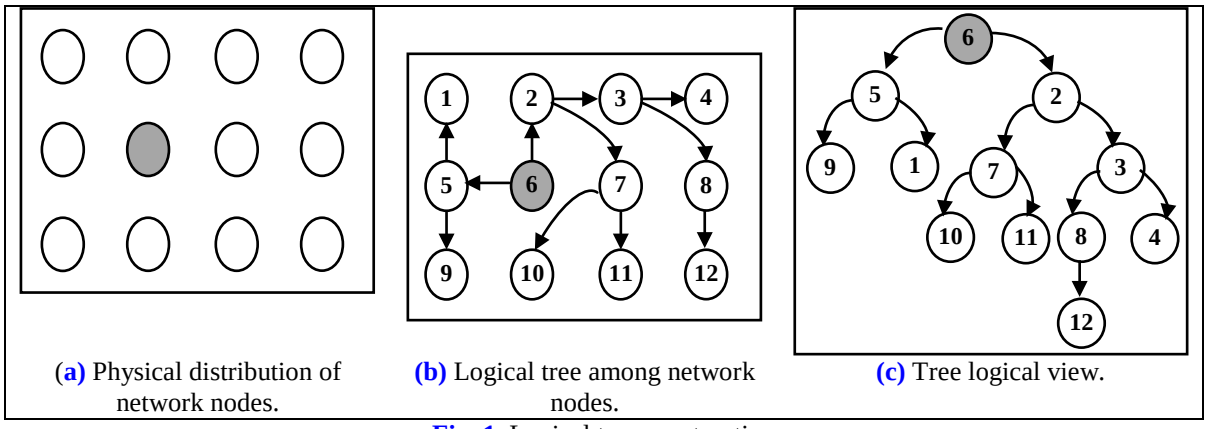
3. FEAR Overview

This section presents an overview of the FEAR protocol and introduces its main phases, advantages, and disadvantages.

FEAR is a tree-based routing protocol that improves the Tree Routing (TR) protocol, which is supported by IEEE 802.15.4 [7]. The TR protocol has two main drawbacks. Firstly, message transmission depends on the source depth, the deeper the node the longer the path. Secondly, it

suffers from node or link failure that causes the isolation of nodes. The FEAR protocol reduces excessive control messages transmitted among network nodes and eliminates the overhead of constructing and updating routing tables.

FEAR has three main phases; first, a logical tree is constructed among the network nodes. During the construction phase, each node gets a logical ID and creates its neighbors' table. **Fig. 1** shows an example of logical tree construction. Secondly, data packets are transmitted using neighbors' links convoyed with parent-child links. During this stage, both the energy of intermediate nodes and the depth are considered. Finally, the tree is reconstructed due to node or link failure or the entry of a new node. In tree construction and reconstruction phases, a fuzzy inference system is used to rank neighboring nodes according to specific factors. This ranking is used to associate a particular node with the best possible parent in terms of remaining energy, depth (number of hops to sink), and distance to maintain a balanced network.



(**a**) Physical distribution of network nodes. (**b**) Logical tree among network nodes. (**c**) Tree logical view.

Fig. 1. Logical tree construction

The FEAR protocol defines different control messages to control the tree construction and reconstruction phases. **Table 1** lists these messages with their corresponding purposes.

Table 1. FEAR control messages

Message	Purpose	Phase(s)
Ready	This message is sent by the node to broadcast its ID to its neighbors and inform other nodes that it is ready to accept children. It contains the sender ID, power, and status.	Tree Construction. Tree Reconstruction.
Engagement	This message is sent by the node to request an ID and a parent.	Tree Construction. Tree Reconstruction.
Engagement_ Acceptance	This message is sent as a reply to an *Engagement* message. It contains the sender ID, current energy, and the ID to be offered to the requesting node.	Tree Construction. Tree Reconstruction.
UnReady	This message is sent by the node to broadcast its ID to its neighbors.	Tree Reconstruction.
RequestParent	This message is sent when a node cannot reach its parent.	Tree Reconstruction.
NewNode	This message is sent when a new node wants to join the network.	Tree Reconstruction.
Inform	This message is sent by the node to inform the neighbors that the node will go down soon.	Tree Reconstruction.
ChangeID	This message is sent when a node changes its ID due to any failure (e.g. the node changed its parent). It informs other nodes to modify its ID in their neighbors' tables.	Tree Reconstruction.

The main disadvantage of the FEAR protocol is the lack of security countermeasures throughout its phases. Attackers can easily bring the network down by overrunning its limited resources or distributing fake topological information among network nodes. The following section discusses possible FEAR routing attacks and proposes the S-FEAR protocol.

4. S-FEAR

As mentioned in the FEAR overview section, attackers can affect FEAR's constructed topology by distributing the wrong topological information. Consequently, this topology cannot provide secure data transmission. In the following subsections, possible attacks that affect the routing process in the FEAR protocol and their consequences are discussed. In addition, security solutions to protect the network from these attacks are also presented and their costs are analyzed and measured.

4.1 FEAR Routing Attacks

The FEAR protocol uses control messages during the tree construction and reconstruction phases. Owing to the absence of security defenses, attackers can utilize these messages to affect the topology construction and disseminate the wrong control data. Additionally, attackers could influence data transmission by injecting false data or dropping important data. **Table 2** lists possible attacks and their consequences on the FEAR data routing and delivery processes.

4.2 S-FEAR Security Requirements

To protect WSNs from different types of attacks, prevention techniques must be applied to the FEAR protocol. Choosing the appropriate techniques depends on both the security requirements and the available resources. This study focuses on establishing a secure topology in the data transmission phase to ensure correct and safe data delivery. The secrecy of the data itself (even among normal sensors) is out of the scope of this paper because it depends on the application itself.

To establish a secure topology and exclude adversary nodes, all sensors participating in the construction phase are required to be benign. To achieve this, node authentication and data integrity are required to ensure the correct distribution of topological information. This paper demonstrates how both authentication and data integrity could be achieved, as well as the corresponding cost in terms of consumed energy.

4.3 S-FEAR Cost Analysis

The communication cost in WSN routing protocols in general and the FEAR protocol in particular is the main cause of sensor energy draining [2]. This includes the sending and receiving of messages performed by all sensors.

Applying security schemes to achieve security services such as authentication and data integrity requires the sensor to perform local operations to verify the authenticity and correctness of the messages, which also costs the sensors some of their energy.

This section analytically measures the cost of building a secure model to secure FEAR and produce S-FEAR, considering both the communication and processing operational costs.

Table 2. FEAR routing attacks and their consequences

Control Message	Purpose of the message	Attack	Attack Consequences
Ready	Sent when: 1. A node gets an ID; it is used to broadcast the ID to inform other nodes that it is ready to accept children 2. The node receives a *New-Node* or *RequestParent* messages	Send a large number of fake *Ready* messages.	- Overrun neighbors' resources due to the cost of receiving, processing, and storing fake *Ready* messages. - Neighbors add the adversary node to their neighbors' tables.
		Eavesdrop on neighbors' *Ready* messages.	Adversary node becomes aware of the network topology.
		Change the contents of the neighbors' *Ready* messages.	- Neighbors store wrong topological information in their tables. - Affect the engagement process.
Engagement	Sent when receiving a *Ready* message and used to request an ID and to assign a Parent	Send a large number of fake *Engagement* messages.	- Overrun neighbors' resources due to the cost of receiving, processing, and replying to fake *Engagement* messages. - Neighbors accept the adversary node as a child and send a reply (*Engagement_Acceptance*).
Engagement_Acceptance	Sent as a response to an Engagement message.	Send a large number of fake *Engagement_Acceptance* messages.	- Overrun neighbors' resources due to the cost of receiving and processing fake *Engagement_Acceptance* messages. - Neighbors accept the adversary node as a parent.
		Change the contents of neighbors' *Engagement_Acceptance* messages.	- Neighbors get wrong Offered IDs; and consequently, have wrong IDs. - Incorrect topological information is exchanged among the nodes.
UnReady	Used to broadcast the ID.	Send a large number of fake *UnReady* messages.	- Overrun neighbors' resources due to the cost of receiving, processing, and storing fake *UnReady* messages. - Neighbors add the adversary node into neighbors table.
		Eavesdrop on neighbors' *UnReady* messages.	Adversary node becomes aware of the network topology.
		Change the contents of neighbors' *UnReady* messages.	Neighbors store the wrong topological information in their tables.
NewNode / RequestParent	Sent when: a new node wants to join the network / a node cannot reach its parent	Send a large number of fake *NewNode/RequestParent* messages.	- Overrun neighbors' resources due to the cost of receiving, processing, and replying to fake *NewNode* / *RequestParent* messages. - Neighbors may accept the adversary node as a child.
Inform	Sent to inform neighbors that the node will go down.	Send a large number of fake *Inform* messages.	Overrun neighbors' resources due to the cost of receiving, processing, and responding to fake *Inform* messages.
		Eavesdrop on neighbors' *Inform* messages.	Adversary node becomes aware of the network topology.
		Change the contents of neighbors' *Inform* messages.	- Neighbors may exclude benign nodes from their neighbors table; consequently, wrong topological information will be stored. - Wrong topology will be constructed among network nodes.
ChangeID	Sent when a node changes its ID due to some failure to inform other nodes to modify the ID in their tables.	Send large number of fake *ChangeID* messages.	- Overrun neighbors' resources due to the cost of receiving, processing, and responding to fake *ChangeID* messages.
		Eavesdrop on neighbors' *ChangeID* messages.	Adversary node becomes aware of the network topology.
		Change the contents of neighbors' *ChangeID* messages.	- Neighbors change neighbors' ID. Consequently, wrong topological information will be stored. - Wrong topological information will be exchanged among network nodes.

Table 3 summarizes the list of notations used in the following equations and their meanings.

Table 3. Equation notations and meanings

Notations	**Meaning**
N	Number of nodes.
K	Message size in bits.
D	Distance between sender and receiver.
BC	Block Cost. Depends on the security mechanisms that are used.
BS	Block Size. Depends on the security mechanisms that are used.
RS_i	The size of *i'th Ready* message.
ES_i	The size of *i'th Engagement* message.
EAS_i	The size of *i'th EngagementAcceptance* message.
Ne_{ni}	Number of neighbors for node i.

In terms of the communication cost, Heinzelman et al. [32] demonstrated that a node needs *ETx(k,d)* to send a *k* bits message to a destination at distance *d*.

$$SendingCost\ (k, d) = ETx\ (k, d) = Eelec*k + Eamp*k * d^2 \quad (1)$$

Where,

- *Eelec = 50 nJ/bit*
- *Eamp = 100 pJ/bit/m²*

Additionally, a node needs *ERx(k)* to receive a *k* bits message

$$receivingCost\ (k) = ERx\ (k) = Eelec*k \quad (2)$$

According to the FEAR protocol, the maximum number of messages that might be exchanged among the nodes during the tree construction phase is calculated in Equations 3 and 4. This number was proved in [2].

$$SentMessages = 2N + \sum_{i=1}^{N} Ne(n_i) \quad (3)$$

$$receivedMessages = N + 2 * \sum_{i=1}^{N} Ne(n_i) \quad (4)$$

Where,

- *N = Number of sensor nodes*
- *$Ne(n_i)$ = Number of neighbors for a specific sensor*

Therefore, using *Equations 1- 4*, the S-FEAR ***communication cost*** is calculated as shown in *Equation 5*:

$$\text{S - FEAR ComCost} = 2N + \sum_{i=1}^{N} Ne(n_i) * ETx(k,d) + 2 * \sum_{i=1}^{N} Ne(n_i) * ERx(k) \quad (5)$$

In terms of the ***processing cost***, *Equation 6* shows the cost of one control message. This cost depends on two factors: number of message blocks and the cost of processing one block.

$$Message\ Cost\ (MsgCost) = NumOfBlocks\ ?loc\quad k\ Cost \quad (6)$$

Where,

$$NumOfBlocks = \left\lceil \frac{MessageSize}{BlockSize} \right\rceil \quad (7)$$

The Block Cost (BC) is the cost of processing one message block and it depends on the security mechanism(s) used. The processing cost of both sent and received messages in S-FEAR is calculated in ***Theorem 1*** and ***Theorem 2***.

Theorem 1: In the S-FEAR protocol, the maximum amount of energy that is consumed by processing the sent messages during secure tree construction phase is:

$$(\sum_{\mathrm{i=1}}^{\mathrm{N}}(\left\lceil \frac{\mathrm{RSi}}{\mathrm{BS}} \right\rceil + \left\lceil \frac{EASi}{BS} \right\rceil) + \sum_{i=1}^{N}(Ne_{ni} * \left\lceil \frac{ESi}{BS} \right\rceil)) * BC$$

Proof: According to the FEAR protocol, three control messages are sent during tree construction which are *Ready*, *Engagement*, and *EngagementAcceptance*. For *Ready* messages, it has been proved in FEAR that *N Ready* messages are needed to build the tree. Based on this, the number of blocks for all *Ready* messages is derived to be equal to $\sum_{\mathrm{i=1}}^{\mathrm{N}}\left\lceil \frac{\mathrm{RSi}}{\mathrm{BS}} \right\rceil$. The same calculations were performed for the *Engagement* and *EngagementAcceptance* messages with $\sum_{\mathrm{i=1}}^{\mathrm{N}}(Ne_{ni} * \left\lceil \frac{ESi}{BS} \right\rceil)$ and $\sum_{\mathrm{i=1}}^{\mathrm{N}}\left\lceil \frac{\mathrm{EASi}}{\mathrm{BS}} \right\rceil$ blocks, respectively. Knowing the number of blocks for all control messages, we can use *Equations 6 and 7* to calculate the total cost as $(\sum_{\mathrm{i=1}}^{\mathrm{N}}(\left\lceil \frac{\mathrm{RSi}}{\mathrm{BS}} \right\rceil + \left\lceil \frac{EASi}{BS} \right\rceil) + \sum_{i=1}^{N}(Ne_{ni} * \left\lceil \frac{ESi}{BS} \right\rceil)) * BC$.

Theorem 2: In the S-FEAR protocol, the maximum amount of energy that is consumed by processing the received messages during the secure tree construction phase is:

$$(\sum_{\mathrm{i=1}}^{\mathrm{N}}(\left\lceil \frac{EASi}{BS} \right\rceil) + \sum_{i=1}^{N}(Ne_{ni} * (\left\lceil \frac{\mathrm{RSi}}{\mathrm{BS}} \right\rceil + \left\lceil \frac{ESi}{BS} \right\rceil))) * BC$$

Proof*:* According to the FEAR protocol, three control messages are received during tree construction which are *Ready*, *Engagement*, and *EngagementAcceptance*. For *Ready* messages, it has been proved that FEAR receives $\sum_{\mathrm{i=1}}^{\mathrm{N}}(Ne_{ni})$ *Ready* messages to build the tree. Based on this, we derived the number of blocks for all *Ready* messages to be equal to $\sum_{\mathrm{i=1}}^{\mathrm{N}}(Ne_{ni} * \left\lceil \frac{RSi}{BS} \right\rceil)$. The same calculation was performed for *Engagement* and *EngagementAcceptance* messages with $\sum_{i=1}^{N}(Ne_{ni} * \left\lceil \frac{ESi}{BS} \right\rceil)$ and $(\sum_{\mathrm{i=1}}^{\mathrm{N}}(\left\lceil \frac{EASi}{BS} \right\rceil)$ blocks, respectively. Knowing the number of blocks for all control messages, we can use Equations 6 and 7 to calculate the total cost as $(\sum_{\mathrm{i=1}}^{\mathrm{N}}(\left\lceil \frac{EASi}{BS} \right\rceil) + \sum_{i=1}^{N}(Ne_{ni} * (\left\lceil \frac{\mathrm{RSi}}{\mathrm{BS}} \right\rceil + \left\lceil \frac{ESi}{BS} \right\rceil))) * BC$.

In conclusion, the overall cost (*Communication* + *processing*) of building the S-FEAR topology is calculated in *Equation 8*.

$$OverallCost(S\text{-}FEAR) = \tag{8}$$
$$(2N + \sum_{i=1}^{N} Ne(n_i)) * [ETx(k,d) + MsgCost]) + (N + 2 * \sum_{i=1}^{N} Ne(n_i)) * [ERx(k) + MsgCost])$$

This secure model was implemented by applying specific security techniques to achieve the required security services. Afterwards, experiments were conducted to test the protocol's performance in terms of attack prevention and energy consumption after implementing the selected security enforcers, as illustrated in the following sections.

4.4 S-FEAR Protocol Implementation

As this paper focuses on achieving authentication and data integrity, Message Authentication Code (MAC) or Digital Signature (DS) schemes could be used to achieve these two security requirements. MAC algorithms are either hash-based or block cipher-based. On the other hand, DS uses Public Key Cryptography (PKC) to sign the transmitted messages.

A detailed study about the energy cost estimation of different MAC and DS techniques has been conducted in [33]. Based on this study, and to balance security strength and energy consumption, both CBC-MAC using AES and Elliptic Curve Cartography (ECC) were examined in this study to develop a secure FEAR protocol (S-FEAR).

AES [34] is one of the most popular symmetric-key cryptographic algorithms. It uses a 4 x 4 array of bytes called the state array. Its main operations are substitution and permutation and it operates on a fixed block size of 128 bits and variable key sizes, 128, 192, or 256 bits, with 10, 12, and 14 rounds, respectively.

CBC-MAC [35] is a technique used to calculate the MAC value based on a block cipher. The message is encrypted using the block cipher algorithm in CBC mode to create a chain of blocks. The MAC value is added to the message and this value is the result of the encryption of the last block. It uses two different keys; one key is used by the block cipher algorithms (CBC) and the other for the MAC calculation [36]. In this way, any change to any plaintext bit(s) will cause a significant change in the final encrypted block that cannot be predicted without knowing the block cipher key.

The use of authentication and data integrity techniques requires that all nodes agree on either shared or public keys before applying the security schemes. Therefore, the following assumptions were made in this study:

- All required keys are preloaded into network sensors.
- All sensors participate in the security verification process.

Control messages during different phases of the FEAR protocol should be either signed or embedded with a MAC value. When a sensor receives a control message, it should verify whether an authenticated node sent this message or not. In addition, it should verify if it has been received without any modification, otherwise the constructed topology should not be considered secure. **Table 4** lists the main steps to achieve a secure network topology.

The following is an example of calculating the processing cost of one message after specifying the security techniques. Using CBC-MAC the cost of processing a 128-bit block is 112.2 µJ when an AES-128 block cipher is used with a key size of 128 bits [36]. On the other hand, signing the message using ECDSA-160 costs 6.26 mJ and verifying the signature costs 12.41 mJ [37].

Assuming that the size of a Ready message is 160 bits when using CBC-MAC, the message processing cost can be calculated using Equation 6 as:

Table 4. Main steps performed by each node to achieve a secure network topology in the S-FEAR protocol

Sending Process	***Step 1.*** Apply CBC-MAC or ECDSA for any sent control message. ***Step 2.*** Attach the calculated MAC value or the signature of the control message. ***Step 3***. Exit.
Receiving Process	***Step 1***. Apply CBC-MAC or ECDSA to verify the correctness of any received message. ***Step 2.*** If the message is successfully verified then go to step 3 otherwise go to step 4. ***Step 3***. Process the message according to the FEAR protocol procedures and go to step 5. ***Step 4.*** Ignore the message. ***Step 5.*** Exit.

- Number of blocks = 160/128 = 2
- Message cost = 2 * 112.2 = 224.4 µJ

While in the case of using ECDSA, the cost of signing and verifying the message is 6.26 + 12.41 = 18.67 mJ. Using *Equation 8*, the overall processing cost of the tree construction phase given the number of messages can be calculated.

The following section provides a detailed empirical study to test the performance of S-FEAR after implementing the secure model and measuring its cost. Moreover, the effect of attacks before and after injecting the security enforcers is studied.

5. Evaluation and Simulation Results

In this section, the estimated energy cost for applying security techniques and the simulation results are discussed. The Qualnet 5.0 Network simulator [38] was used to simulate both the FEAR and S-FEAR protocols. **Table 5** lists the hardware and software specifications in addition to the simulation parameters that were used to conduct the experiments. Different evaluation metrics are used to evaluate the two protocols (with and without applying security schemes).

Table 5. Hardware and Software Specifications and Simulation Parameters

Hardware/Software	**Description**
CPU	Intel (R) core (TM) i3
RAM	4 GB
OS	Windows XP
Simulation Parameter	**Value(s)**
Network Size	10, 20, 30, 40, and 50 nodes
Terrain Area	1500 m X 1500 m
Radio Range	250 m
Number of Sinks	1
Initial Energy	1000 J
Number of Attackers	10%, 20%, 30%, 40%, and 50 % of network size.
Attack start time	after 3000 s from the beginning of the simulation time
Experiment Simulation Times	20000 s

5.1 S-FEAR Security Cost

This section evaluates the cost of applying CBC-MAC or ECDSA in the S-FEAR protocol. Both network overhead and energy consumption are studied before and after applying the security mechanisms.

- **Impact of Applying Security Schemes on the Network Overhead**

The overhead is calculated in terms of the number of sent and received messages during the network's lifetime. Different network sizes: 10, 20, 30, 40, and 50 nodes were tested. For each size, the same node characteristics (initial energy, node distribution, and distance from sink) were used for FEAR, S-FEAR (MAC), and S-FEAR (DS). **Fig. 2** and **Fig. 3** show the number of sent and received messages of the FEAR protocol before and after applying the security mechanisms, respectively. As illustrated in the figures, no additional overhead is caused by the S-FEAR protocol after applying CBC-MAC or ECDSA. This is because it is implemented on the same number of control messages that are required to construct the tree without the need to exchange extra messages.

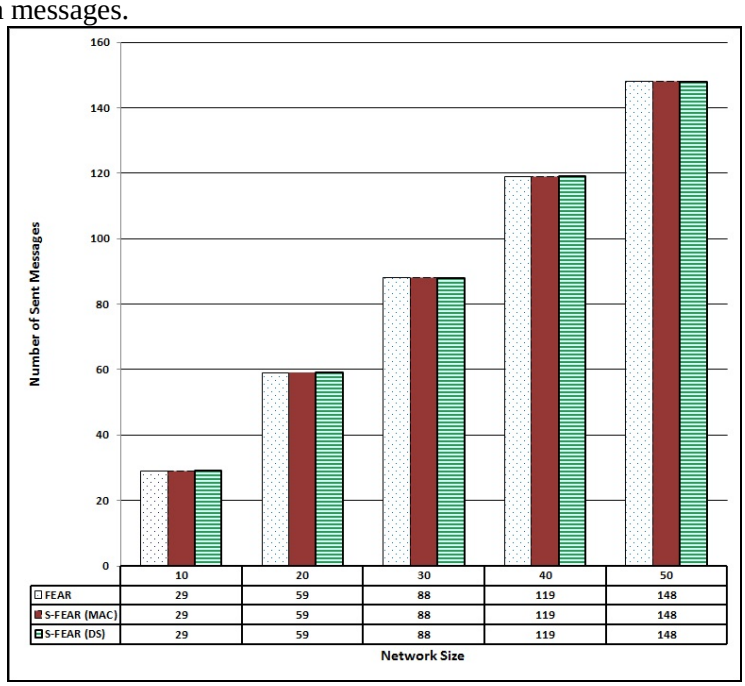


Fig. 2. Comparison between FEAR, S-FEAR (MAC), and S-FEAR (DS) in terms of the number of sent messages with different network sizes.

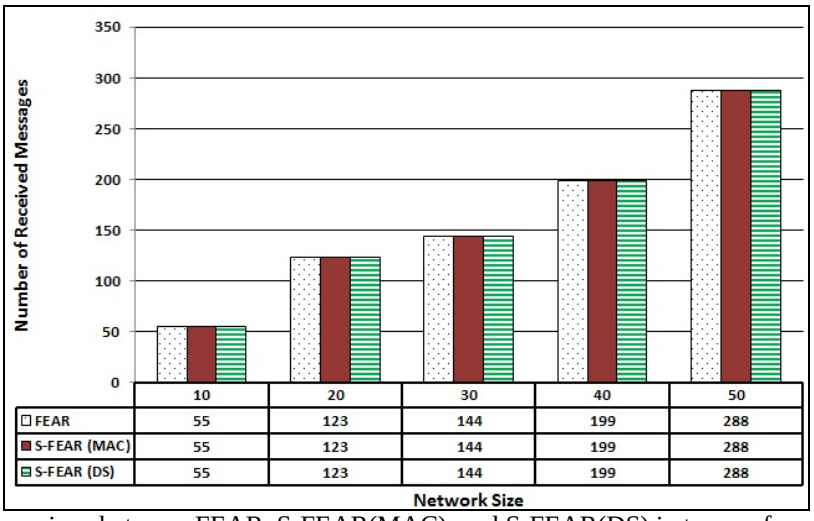
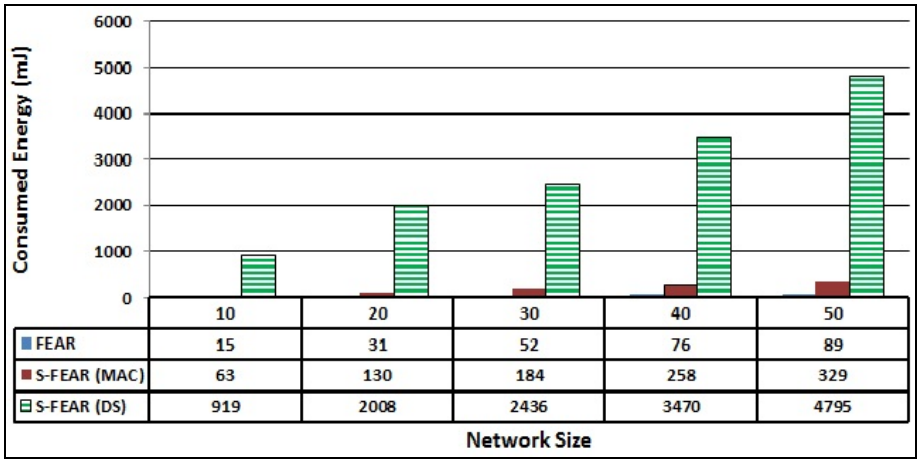


Fig. 3. Comparison between FEAR, S-FEAR(MAC), and S-FEAR(DS) in terms of number of the received messages with different network sizes

- **Impact of Applying Security Schemes on the Energy Consumed in the Network**

As discussed earlier, energy consumption due to local processing needs to be considered in addition to the communication cost. To obtain energy consumption results, the same simulation characteristics and scenarios considered in evaluating the overhead were used. **Fig. 4** shows that the consumed energy increases when applying the security schemes.



	10	20	30	40	50
FEAR	15	31	52	76	89
S-FEAR (MAC)	63	130	184	258	329
S-FEAR (DS)	919	2008	2436	3470	4795

Fig. 4. Comparison between FEAR, S-FEAR(MAC), and S-FEAR(DS) in terms of energy consumption against different network sizes.

The results illustrate that using CBC-MAC costs about 64 x 10^{-5}% of the network's energy whereas ECDSA costs about 91 x 10^{-4} %. Note that the asymmetric-based security enforcer consumes more energy than the symmetric enforcer does; due to the complexity of the signature generation and verification processes.

5.2 Effect of Security Attacks

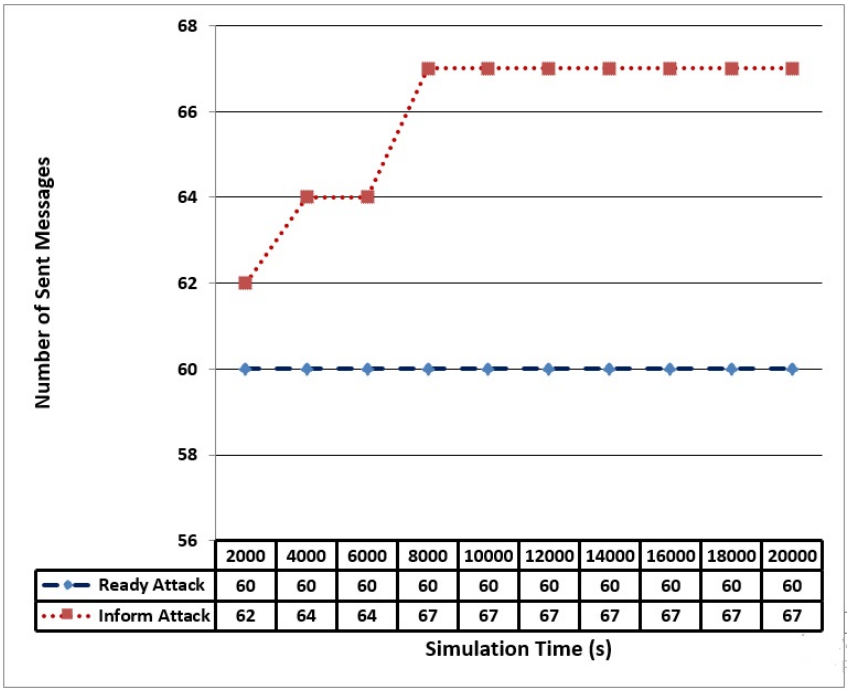
This section evaluates the impact of external attacks on the FEAR protocol in terms of network overhead and consumed energy. Five different attacks are studied: Ready Attack, Inform attack, *ChangeID* attack, *RequestParent* attack, and *NewNode* attack. Each attack is named according to the type of fake control message the attackers could send.

- **Effect of Attacks on Network Overhead**

In this section, the overhead is also calculated in terms of the number of sent and received messages. The assumed attacking scenario dictates that each attack starts after 3000 s from the beginning of the simulation time and sends one fake control message per second until the simulation stops after 20000 s. The resulting number of sent messages are shown in **Fig. 5** and **Fig. 6** (Note: the results are split into two figures due to a scaling gap).

In **Fig. 5,** the results of the Ready and Inform attacks are illustrated. For the Ready attack, the number of sent messages remains constant (60 messages) after 2000 s from the start of the simulation time. This is because Ready attacks are performed by sending fake Ready messages. The receivers of these Ready messages from normal network sensors are not required to send any control messages as replies. On the other hand, Inform attacks affected the number of sent messages during the time interval from 2000 to 8000 s. This was because some of the attackers

were parents to some of the sensor nodes. The parent attacker sends a fake Inform massage telling its children that it will go down, thus the children should find another parent through the tree reconstruction phase. This reconstruction requires extra messages to be exchanged among the network nodes.



	2000	4000	6000	8000	10000	12000	14000	16000	18000	20000
Ready Attack	60	60	60	60	60	60	60	60	60	60
Inform Attack	62	64	64	67	67	67	67	67	67	67

Fig. 5. Effect of *Ready* and *Inform* attacks on the number of sent messages versus simulation time.

After 8000 s, the reconstruction phase completed and all parent attackers were excluded. Consequently, the number of sent messages remained constant until the end of the simulation. The Inform attacks remained active until the end of the simulation, but after 8000 s, they would have only affected the number of received messages.

In **Fig. 6**, the results of the *ChangeID*, *RequestParent*, and *NewNode* attacks are illustrated. Unlike other attacks, *RequestParent* and *NewNode* attacks significantly increased the number of sent messages since all neighbors (attacker's neighbors) reply to these attacks by sending Ready or *UnReady* messages. On the other hand, the ChangeID attack affected only children nodes. If any child (attacker's child) received a fake *ChangeID* message, it also sends a *ChangeID* message. Thus, only children are affected by this attack in terms of sent messages.

The impact on the number of received messages is shown in **Fig. 7**. *RequestParent* and *NewNode* attacks increased the received messages significantly since the neighbor nodes broadcast replies to these attack messages by sending either Ready or *UnReady* messages affecting all nodes within their ranges.

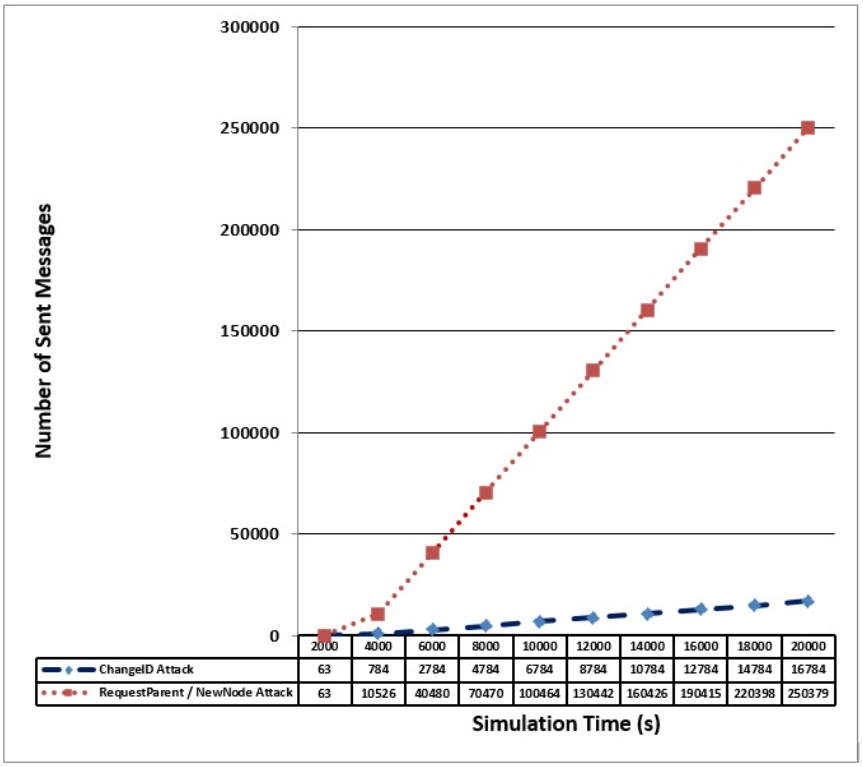


Fig. 6. Effect of *ChangeID*, *RequestParen*t, and *NewNode* attacks on the number of sent messages during simulation time.

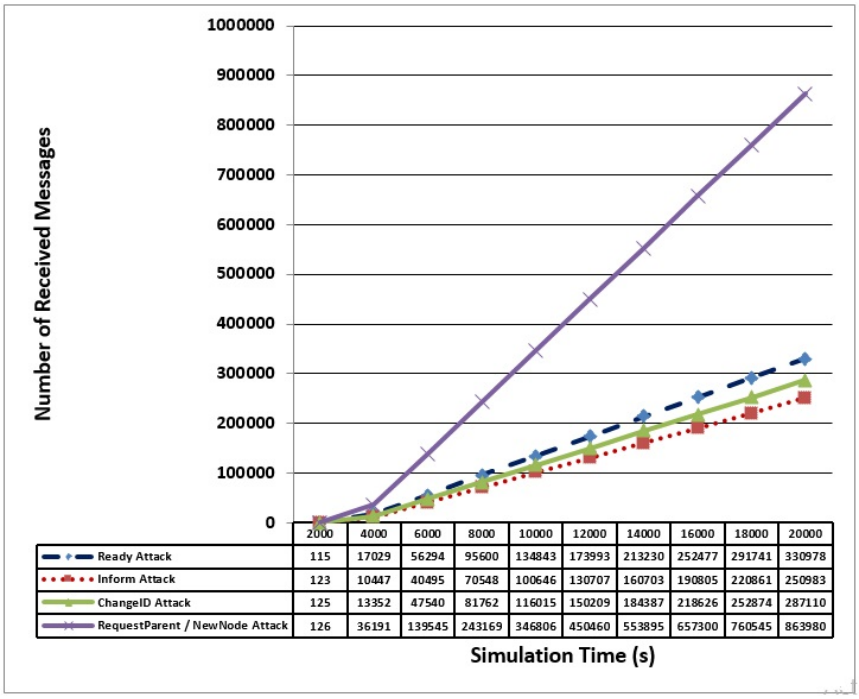


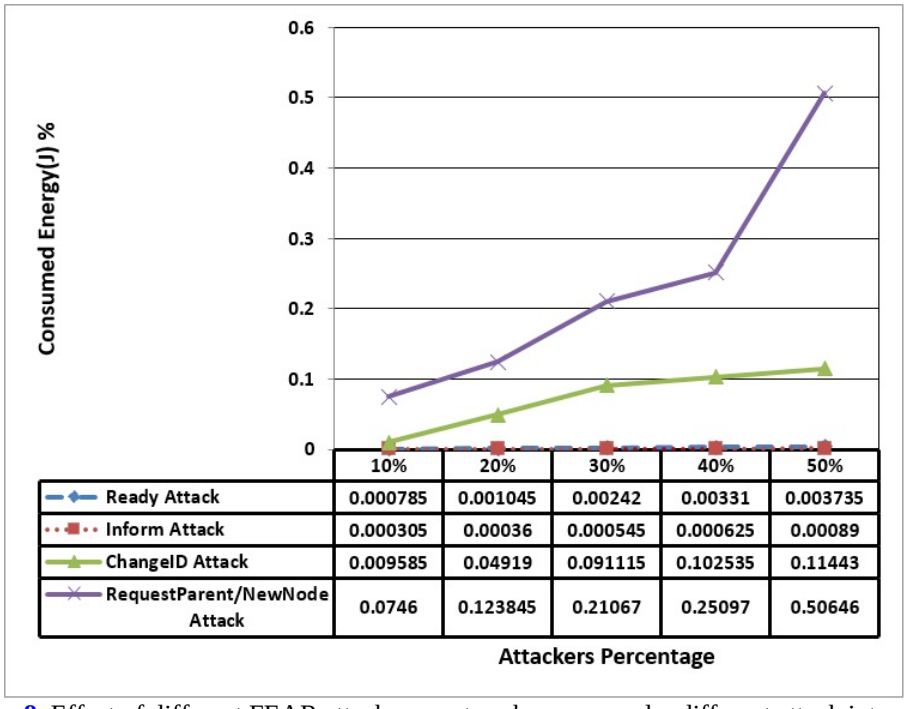
Fig. 7. Effect of different FEAR attacks on the number of received messages versus the simulation time.

- **Effect of Attacks on the Energy Consumed in the Network**

Since no local security processes and verifications are required in the FEAR protocol, the consumed energy depends mainly on communicating the sent and received messages, which is calculated in *Equations 1 and 2*. The results were obtained according to two different scenarios:

Firstly, the impact on network energy considering different attack intensities (10%, 20%, 30%, 40%, and 50% of network size) was studied. The simulation time was 6000 s to give attacks enough time to have an effect. The results are shown in **Fig. 8**. The larger the number of attackers, the more energy is consumed. Moreover, *RequestParent*, *NewNode*, and *ChangeID* attacks consume more energy than other attacks due to the large overhead caused by these attacks.

Secondly, the impact of attackers on network energy compared to simulation time was studied. The results are illustrated in **Fig. 9**. The number of attackers in this scenario was fixed for all types of attacks, which is 30% of the network size. As illustrated in the figure, increasing the duration of the attacks increases the consumed energy since the overhead increases as well. According to the conducted scenarios, the attacks could consume up to 1.3% of the network's energy after only 20000 s of network lifetime.



	10%	20%	30%	40%	50%
Ready Attack	0.000785	0.001045	0.00242	0.00331	0.003735
Inform Attack	0.000305	0.00036	0.000545	0.000625	0.00089
ChangeID Attack	0.009585	0.04919	0.091115	0.102535	0.11443
RequestParent/NewNode Attack	0.0746	0.123845	0.21067	0.25097	0.50646

Fig. 8. Effect of different FEAR attacks on network energy under different attack intensities.

5.3 Impact of Attackers on S-FEAR

This section studies the impact of attackers on S-FEAR after applying CBC-MAC or ECDSA and compares them with the unsecure FEAR protocol. Three evaluation metrics are considered: network overhead, energy consumed by the network, and network topology.

- **Network Overhead and Energy Consumption**

The overhead and the energy consumption that are caused by external attacks before and after applying the security mechanisms are compared. The same scenarios characteristics are used

to

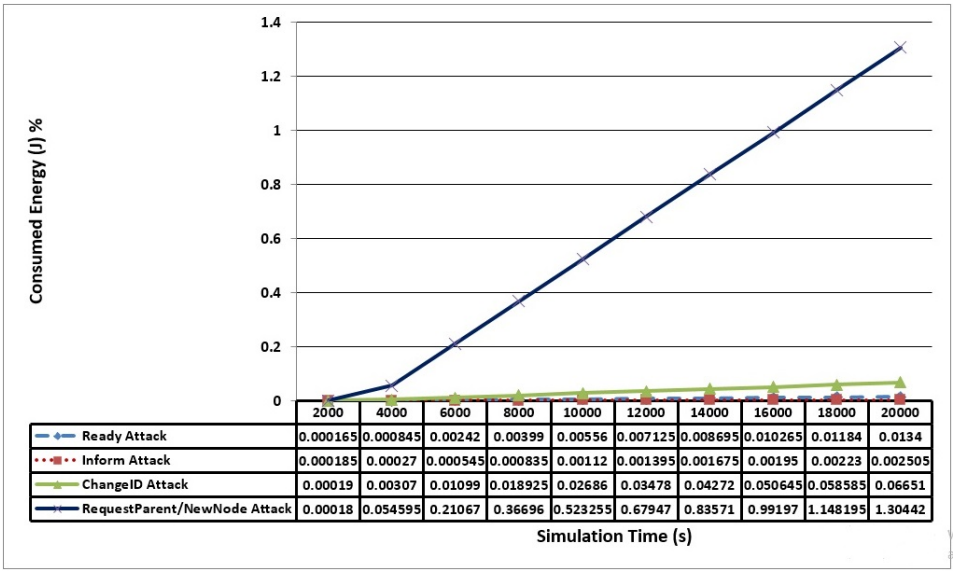


Fig. 9. Effect of different FEAR attacks on the network energy against simulation time.

evaluate three protocols: FEAR, S-FEAR (MAC), and S-FEAR (DS). The results of sent and received messages are shown in **Fig. 10** and **Fig. 11**, respectively, whereas the results of the energy consumption are shown in **Fig. 12**. According to the network overhead, applying security mechanisms (either CBC-MC or ECDSA) will decrease the effect of attacks since sensors will not respond to adversary nodes and consequently, the number of exchanged messages will be reduced. Please note that Ready and Inform attacks do not affect the number of sent messages (as mentioned earlier).

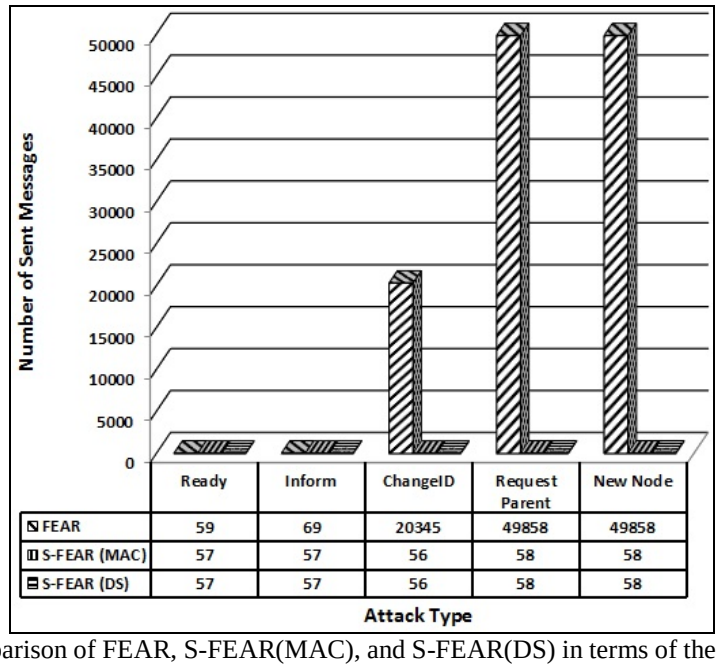


Fig. 10. Comparison of FEAR, S-FEAR(MAC), and S-FEAR(DS) in terms of the effect of different FEAR attacks on the number of sent messages.

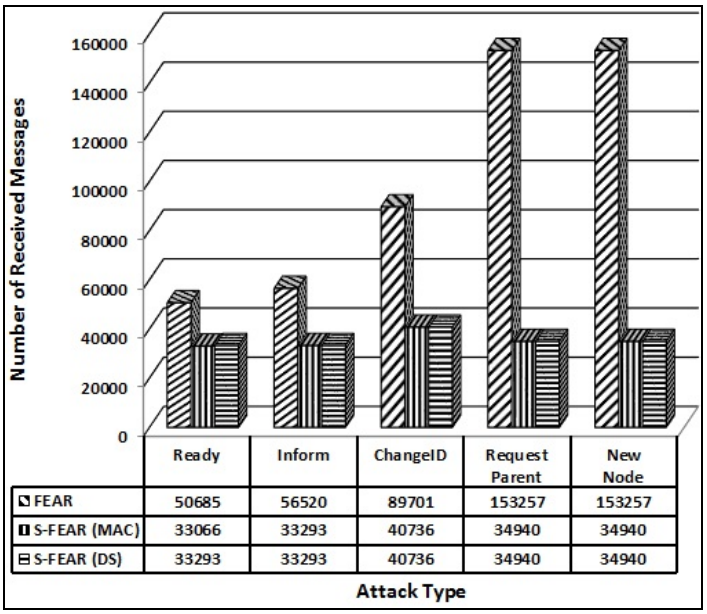


Fig. 11. Comparison of FEAR, S-FEAR (MAC), and S-FEAR (DS) in terms of the effect of different FEAR attacks on the number of received messages.

According to the consumed energy (**Fig. 12**), the results show that some attacks (*RequestParent* and *NewNode*) cost the network about 4.7 times the cost that is required when security requirements are applied using the CBC-MAC technique. This value is calculated by dividing 0.241125, which is the cost of the FEAR protocol under *RequestParent* and *NewNode* attacks, by 0.0513, which is the cost of the FEAR protocol under the same attacks using CBC-MAC. On the other hand, other attacks cost the network more when the security schemes are applied due to the verification process of fake messages. Moreover, as illustrated in the following figure, the DS scheme consumes the most energy due to its complexity in comparison with the MAC scheme.

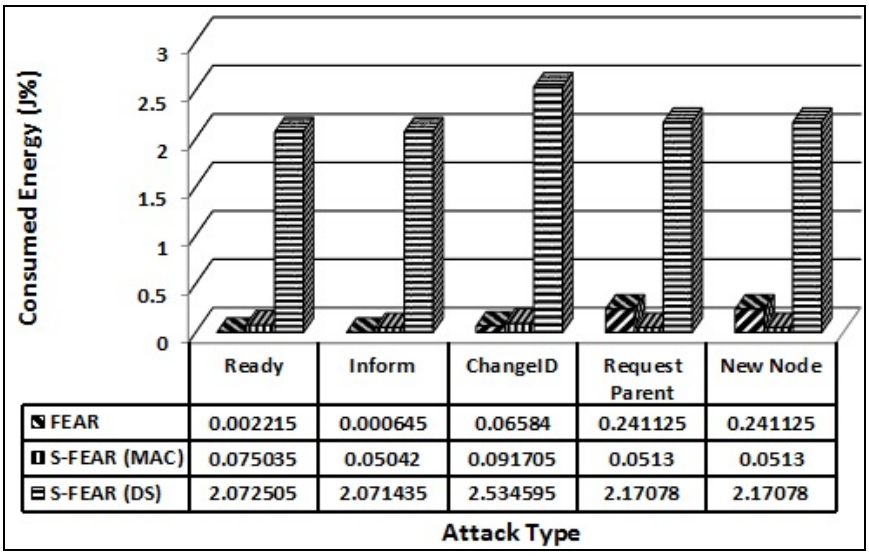


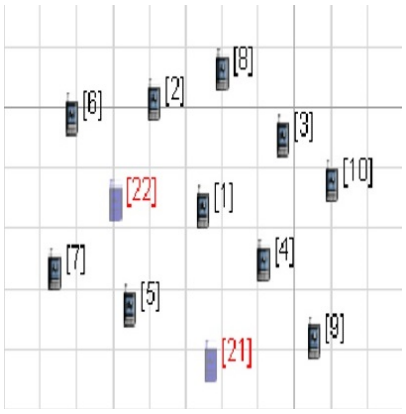
Fig. 12. Comparison of FEAR, S-FEAR (MAC), and S-FEAR (DS) in terms of the effect of different FEAR attacks on the energy consumption.

- **Network Topology**

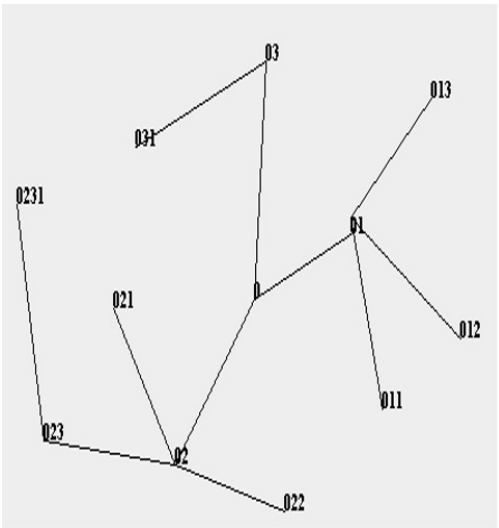
This section assesses the impact of attackers on the network topology before and after applying the security schemes. A scenario with a small network size was chosen to illustrate this impact. In **Fig. 13 (a)**, the network consists of 10 normal nodes and 2 attackers (20% of attacker intensity), which are nodes 21 and 22.

When the FEAR protocol is applied as shown in **Fig. 13 (b)**, all attackers had participated in the tree construction phase and were treated as normal nodes. In this scenario, the tree construction was started by node 1, with a logical ID of 0, which broadcasts a Ready message to the other nodes. Attacker 21 is engaged with node 0, with an ID of 03. The same is true for attacker 22 that is engaged with node 02 with an ID of 021. The tree construction continues until all nodes, including attackers, receive IDs. Thus, in the FEAR protocol, attackers can participate easily in the tree construction phase, which will lead to an unsecure topology construction.

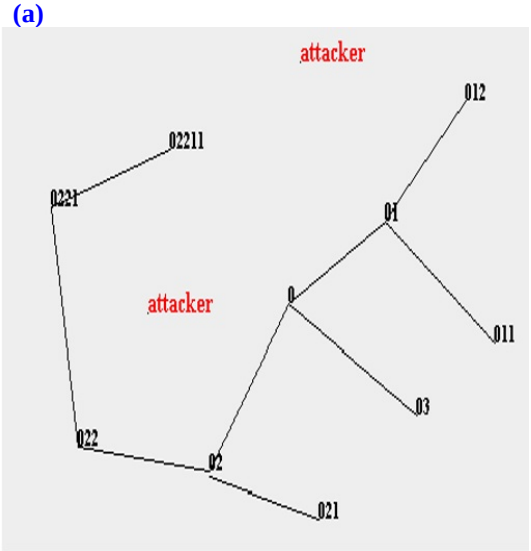
On the other hand, S-FEAR excludes all attackers from the constructed tree since they are not authenticated (**Fig. 13 (c)**). Consequently, the topology is kept safe and can be used to transmit data packets safely and securely among the network nodes.



(a)



(b)



(c)

Fig. 13. Effect of attackers on the network topology, **(a)** The network field with 20% of attackers, **(b)** The constructed topology in FEAR, **(c)** The constructed topology in S-FEAR.

6. Conclusions and Future Work

Considering security in the development of any protocol dedicated for WSNs is vital. This paper shows how malicious nodes can badly affect the services provided by WSNs and drain network energy. Including proactive security solutions will guarantee secure routing and correct, successful data delivery in addition to prolonging the network's lifetime.

This study introduced a Secure-Fuzzy Energy Aware Routing (S-FEAR) protocol for WSNs. S-FEAR improves the original FEAR protocol by applying security mechanisms to build an energy efficient secure routing protocol that protects the network from different external attacks. The main goal is to build a secure topology among network nodes to be used for secure data transmission. A secure model was proposed to secure the FEAR protocol. This model was evaluated analytically and empirically. Both authentication and data integrity were considered in this study and achieved by applying CBC-MAC or ECDSA based on symmetric and asymmetric approaches, respectively.

The Qualnet 5.0 Network simulator was used to assess the impact of attackers on the network topology and the routing process in the FEAR, S-FEAR (MAC), and S-FEAR (DS) protocols. In addition, the cost of attacks on the FEAR and S-FEAR protocol using different simulation experiments was assessed.

After applying the security schemes, the results show that CBC-MAC costs 0.00064% of network energy and ECDSA costs about 0.0091%. Although this cost is required to achieve the necessary security requirements, the results also show that some attacks cost the network about 4.7 times the cost of achieving these requirements. Moreover, the results demonstrate that considering only the resource limitations of the sensor nodes is not enough to develop an efficient routing protocol. Protecting the network from different routing attacks is important since attackers can drain the network energy over time. Requiring authentication and data integrity during tree construction excludes all attackers from the constructed network topology. Consequently, this topology can be used for safe data transmission, which will guarantee the success of services running over WSNs.

As part of future research, S-FEAR could include a more intelligent service that considers the application type, data sensitivity, remaining network energy, and the type of security techniques in terms of complexity, security strength, and latency before deciding in real time which security technique to apply.

Additionally, S-FEAR could also tackle internal attackers. This requires an additional intrusion detection service that might cost the network some of its resources; this is a problem worth investigating in a future study.

Moreover, other existing energy efficient protocols for WSN can be analyzed from a security perspective. In addition, new protocols could be proposed and evaluated to consider their security characteristics during the early design phase, as well as their impact on WSNs.

References

[1] Guoru Ding, Jinlong Wang, Qihui Wu and, Yu-Dong Yao, "Robust Spectrum Sensing With Crowd Sensors," *IEEE Transactions on Communications*, vol. 62, no. 9, pp. 3129-3143, September, 2014. Article (CrossRef Link)

[2] Iman Almomani and Maha Saadeh, "FEAR: Fuzzy-Based Energy Aware Routing Protocol for Wireless Sensor Networks," *International Journal of Communications, Network and System Sciences*, vol. 4, no. 6, pp. 403-415, June 2011. Article (CrossRef Link)

[3] M. Kocakulak and I. Butun, "An Overview of Wireless Sensor Networks towards Internet of Things," in *Proc. of IEEE 7th Annual on Computing and Communication Workshop and Conference (CCWC)*, pp.1-6, January 9-11, 2017. Article (CrossRef Link)
[4] Priyanka Rawat, Kamal Deep Singh, Hakima Chaouchi and Jean Marie Bonnin, "Wireless Sensor Networks: A Survey on Recent Developments and Potential Synergies," *Journal of Supercomputing*, vol. 68, no. 1, pp. 1–48, April, 2014. Article (CrossRef Link)
[5] Sanjeev Kumar Gupta and Poonam Sinha, "Overview of Wireless Sensor Network: A Survey," *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 3, no. 1, pp. 5201-5207, January, 2014. Article (CrossRef Link)
[6] MR. Ahmed, X. Huang, D. Sharma and H. Cui, "Wireless Sensor Networks: Characteristics and Architectures," *International Journal of Electrical, Computer, Energetic, Electronic and Communication Engineering*, vol. 6, no. 12, pp. 1398-1401, 2012. Article (CrossRef Link)
[7] IEEE 802.15.4, "ZigBee Specification Version 1.0", *ZigBee Alliance*, 2005.
[8] Yongsuk Park and Eun-Sun Jung, "Plus-Tree: A Routing Protocol for Wireless Sensor Networks," *Lecture Notes in Computer Science*, LNCS, vol. 4413, pp. 638-646, 2006. Article (CrossRef Link)
[9] Wanzhi Qiu, Efstratios Skafidas and Peng Hao, "Enhanced Tree Routing for Wireless Sensor Networks," *Ad Hoc Networks*, vol.7, no.3, pp. 638–65, May, 2009. Article (CrossRef Link)
[10] M. Al-Harbawi, MFA. Rasid, and NK. Noordin, "Improved Tree Routing (ImpTR) Protocol for ZigBee Network," *International Journal of Computer Science and Network Security*, vol.9, no.10, pp. 146-152, October, 2009. Article (CrossRef Link)
[11] MA. Abuhelaleh, TM. Mismar, and AA. Abuzneid, "Armor-LEACH – Energy Efficient, Secure Wireless Networks Communication", in *Proc. of 17th International Conference on Computer Communications and Networks (ICCCN '08)*, pp.1-7, August 3-7, 2008. Article (CrossRef Link)
[12] Eleni Klaoudatou, Elisavet Konstantinou, Georgios Kambourakis, and Stefanos Gritzalis, "A Survey on Cluster-Based Group Key Agreement Protocols for WSNs," *IEEE Communications Surveys & Tutorials*, vol. 13, no. 3, pp. 429-442, Third Quarter, 2011. Article (CrossRef Link)
[13] Al-Sakib Khan Pathan, Choong Seon Hong, "A Secure Energy-Efficient Routing Protocol for WSN," *Lecture Notes in Computer Science*, vol. 4742, pp. 407–418, 2007. Article (CrossRef Link)
[14] Anthonis Papadimitriou, Fabrice Le Fessant, Aline Carneiro Viana and Cigdem Sengul, "Cryptographic Protocols to Fight Sinkhole Attacks on Tree-based Routing in Wireless Sensor Networks," in *Proc. of 5th Workshop on Secure Network Protocols (NPSec 2009)*, pp.43–48, October 13-13, 2009. Article (CrossRef Link)
[15] Rickard Söderlund, Stefan Svensson, Tomas Lennvall, "Energy Efficient Authentication in Wireless Sensor Networks," in *Proc. of IEEE Conference on Emerging Technologies and Factory Automation*, pp. 1412-1416, September 25-28, 2007. Article (CrossRef Link)
[16] Kui Ren, Member, Wenjing Lou, Kai Zeng, and Patrick J. Moran, "On Broadcast Authentication in Wireless Sensor Networks," *IEEE Transaction on Wireless Communication*, vol. 6, no. 11, pp. 4136-4144, November, 2007. Article (CrossRef Link)
[17] Hsiu-Lien Yeh, Tien-Ho Chen, Pin-Chuan Liu, Tai-Hoo Kim and Hsin-Wen Wei, "A Secured Authentication Protocol for Wireless Sensor Networks Using Elliptic Curves Cryptography," *Sensors*, vol. 11, no. 5, pp. 4767-4779, May, 2011. Article (CrossRef Link)
[18] Sunil Gupta, Harsh Kumar Verma and AL Sangal, "Authentication Protocol for Wireless Sensor Networks," *International Journal of Electrical, Computer, Energetic, Electronic and Communication Engineering*, vol. 4, no. 6, pp. 947-953, 2010. Article (CrossRef Link)
[19] Patrick Olivier Kamgueu, Emmanuel Nataf, Thomas Djotio, Olivier Festor, "Fuzzy-based Routing Metrics Combination for RPL," *Doctoral Consortium Sensornets*, pp.1-8, 2014. Article (CrossRef Link)
[20] Seyed Hassan Mosakazemi Mohammadi and Reza sabbaghi-Nadooshan, "A Novel Comprehensive Taxonomy of Intelligent-Based Routing Protocols in Wireless Sensor Networks," *International Journal of Smart Electrical Engineering*, vol. 2, no. 2, pp. 103-109, Spring, 2013. Article (CrossRef Link)

[21] Amarappa Pagi, Abdul Lateef Haroon, Manjunath K M and Ulaganathan J., "Fuzzy Energy Aware Graph Based Routing (FEAGR) in Wireless Sensor Networks," *International Journal of Research in Engineering and Technology*, vol. 4, no. 5, pp. 133-139, May, 2015. Article (CrossRef Link)
[22] Ajai Kumar Mishra, Rakesh Kumar and Rakesh Kumar, "A Novel Cluster Head Selection Scheme Using Fuzzy Logic in Wireless Sensor Networks," in *Proc. of IEEE International Conference on Green Computing and Internet of Things (ICGCIoT)*, pp.203-208, October 8-10, 2015. Article (CrossRef Link)
[23] Aarti Jain, "Betweenness Centrality Based Connectivity Aware Routing Algorithm for Prolonging Network Lifetime in Wireless Sensor Networks," *Wireless Network*, vol. 22, no. 5, pp. 1605–1624, July, 2016. Article (CrossRef Link)
[24] A. B. Pagi, V. R. Budyal, and M.J. Sataraddi, "Fuzzy Based Energy Aware Flat Routing (FEAFR) in Wireless Sensor Networks," *International Journal of Emerging Technology and Advanced Engineering*, pp. 108-115, July, 2014. Article (CrossRef Link)
[25] A. Arya, A. Malik, and S. Kumar, "A Routing Protocol for Detecting Holes in Wireless Sensor Networks with Multiple Sinks," in *Proc. of Third ACM International Symposium on Women in Computing and Informatics*, pp.103-108, August 10-13, 2015. Article (CrossRef Link)
[26] Patrick-Olivier Kamgueu, Emmanuel Nataf and Thomas Ndie Djotio, "On Design and Deployment of Fuzzy-Based Metric for Routing in Low-Power and Lossy Networks," in *Proc. of IEEE 40th Local Computer Networks Conference Workshops (LCN Workshops)*, pp.789-795, October 26-29, 2015. Article (CrossRef Link)
[27] Ajai Kumar Mishra, Rakesh Kumar and Jitendra Singh, "A Review on Fuzzy Logic Based Clustering Algorithms for Wireless Sensor Networks," in *Proc. of IEEE International Conference on Futuristic Trends on Computational Analysis and Knowledge Management (ABLAZE)*, pp.489-494, February 25-27, 2015. Article (CrossRef Link)
[28] Veenu Mor and Harish Kumar, "Energy efficient wireless mobile networks: A review," in *Proc. of IEEE International Conference on Optimization, Reliability, and Information Technology (ICROIT)*, pp.281-285, February 6-8, 2014. Article (CrossRef Link)
[29] E. Golden Julie and S. Tamilselvi, "CDS-Fuzzy Opportunistic Routing Protocol for Wireless Sensor Networks," *Wireless Personal Communications*, vol. 90, no. 2, pp. 903–922, September 2016. Article (CrossRef Link)
[30] Firoj Ahamad and Rakesh Kumar, "Energy Efficient Region Based Clustering Algorithm for WSN using Fuzzy Logic," in *Proc. of IEEE International Conference on Recent Trends in Electronics Information Communication Technology*, pp.1020-1024, May 20-21, 2016. Article (CrossRef Link)
[31] Ajai Kumar Mishra, Rakesh Kumar, Vimal Kumar and Jitendra Singh, "A Grid-Based Approach to Prolong Lifetime of WSNs Using Fuzzy Logic," *Advances in Intelligent Systems and Computing Book Series*, Springer, pp.11-22, 2017.
[32] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-Efficient Communication Protocol for Wireless Microsensor Networks," in *Proc. of 33rd Hawaii International Conference on System Sciences (HICSS '00)*, Vol. 8. IEEE Computer Society, pp.3005-3014, January 7-7, 2000. Article (CrossRef Link)
[33] Iman Almomani, and Maha Saadeh, "Security Model for Tree-based Routing in Wireless Sensor Networks: Structure and Evaluation," *KSII Transaction on Internet and Information System*, pp. 1223-1247, vol. 6, no. 4, April, 2012. Article (CrossRef Link)
[34] Advanced Encryption Standard, *National Institute of Standards and Technology (NIST)*, 1997.
[35] ISO/IEC 9797-1, "Information technology – Security techniques – Message Authentication Codes (MACs) – Part 1: Mechanisms using a block cipher," *International Standardization Organization (ISO)*, 2011.
[36] Jongdeog Lee, Krasimira Kapitanova and Sang H. Son, "The Price of Security in Wireless Sensor Network," *Computer Networks*, vol. 54, no. 17, June, 2010. Article (CrossRef Link)

[37] P. Trakadas, T. Zahariadis, H.C. Leligou, S. Voliotis and K. Papadopoulos, "Analyzing Energy and Time Overhead of Security Mechanisms in Wireless Sensor Networks," in *Proc. of 15th International Conference on Systems, Signals and Image Processing (IWSSIP 2008)*, pp.137–140, June 25-28, 2008. Article (CrossRef Link)
[38] Qualnet 5.0, Qualnet Network Simulator, Article (CrossRef Link) (Accessed 31 May 2017).

Iman Musa Almomani is an Associate Professor and Associate Chair of the Department of Computer Science at Prince Sultan University, KSA. Iman is also the associate director of research and initiatives center. Before joining Prince Sultan University, Iman worked as an Associate Professor and Head of the Computer Science Department at the University of Jordan, in Jordan. Her academic qualifications include Bachelor and Master degrees in Computer Science from UAE and Jordan in 2000 and 2002, respectively, and a PhD degree from De Montfort University, UK in Wireless Network Security in 2007. Her research interests include Wireless Networks and Security, mainly Wireless Mobile Ad hoc NETworks (WMANETs), Wireless Sensor Networks (WSNs), Multimedia Networking (VoIP) and security issues in wireless networks. She is also interested in the area of electronic learning (e-learning) and mobile learning (m-learning). Iman has several publications in the above areas in many international and local journals and conferences. Iman is involved in the organizing and technical committees for a number of local and international conferences. In addition, she serves as a reviewer and a member of the editorial board in a number of international journals. Iman is also a senior member of IEEE and IEEE WIE.

Maha Saadeh, is a Ph.D. student in Computer Science at the University of Jordan. She worked as a research and teaching assistant at the Computer Science department at the University of Jordan from September 2009 to September 2010. Then, she received her M.Sc. degree in Computer Science from the same university in 2011. She has several publications in a number of local and international journals and conferences. Her research interests include wireless networks, network security, and the Internet of Things (IoT).