A Decentralized Energy-Aware Key Management Scheme for Wireless Sensor Networks

Salwa Adriana Saab  Ayman Kayssi  Ali Chehab
Department of Electrical and Computer Engineering
American University of Beirut
Beirut 1107 2020, Lebanon
{sis08,ayman,chehab}@aub.edu.lb

Abstract — The popularity of Wireless Sensor Networks (WSNs) is increasing especially in applications where data needs to be remotely collected, such as in fire detection, health, or environmental monitoring. WSN nodes are limited in terms of processing capabilities and battery life. Thus, encryption is usually avoided and the readings are sent in the clear. This allows any eavesdropper to access data that could be confidential. Lightweight encryption techniques are proposed to overcome the limitations of sensor nodes. Identity-based encryption (IBE) that uses elliptic curve cryptography (ECC) seems to be very promising in terms of energy efficiency. Since the issue of key management is critical for a security system, we propose a novel decentralized IBE-based key management scheme that reduces the energy by using multiple base stations. The keys are pre-distributed in the WSN and refreshed at specific time intervals. The system ensures confidentiality of the messages and the availability of WSN service even when multiple nodes and base stations are compromised, at a significant reduction in overall system energy.

Keywords— Identity-based encryption, key management, confidentiality, energy-awareness, decentralization.

I. INTRODUCTION

Wireless sensor networks consist of a set of nodes that collaborate with each other to collect and forward data to a base station. Decisions and actions are performed based on the received information. WSNs have numerous applications that demand the collection of remote data, such as industrial and environmental monitoring, health care, home automation, and military surveillance [1, 2].

Sensors are microelectronic devices that are limited in terms of computation, storage, and energy. Thus, data which might be sensitive or private is sent in the clear as traditional encryption techniques require more than the available sensor resources, which allows any eavesdropper to effortlessly access the data. However, new cryptographic methods are being proposed that achieve security while taking into consideration the constrained capabilities of the sensor nodes. Identity-based encryption (IBE) attracted the attention of researchers due to its concept of making the identity of a sensor node the same as its public key [3]. Moreover, elliptic curve cryptography (ECC) consumes less energy as compared to other public key schemes without compromising the security of the system [1], and is therefore well suited for WSN environments.

In this paper, we propose a novel decentralized identity-based key management system that enhances the security of the network while reducing the energy consumption due to the existence of multiple base stations.

The rest of the paper is organized as follows: In section 2, we discuss the motivation behind the proposed scheme. We present related work and necessary preliminaries in sections 3 and 4, respectively. We provide a detailed explanation of our system architecture in section 5 and we evaluate its performance and security in section 6. We conclude our discussion in section 7.

II. MOTIVATION

The nature of sensor nodes and their severe limitations restrict the application of encryption-based security measures in them. After establishing a WSN, these nodes consume the energy stored in their batteries to perform the actions of collecting and forwarding data in the network. The lifetime of the WSN is determined by the sensor nodes that get eliminated once their batteries are drained. Also, sensors are not provided with large storage and processing capabilities. As such, many applications place the requirements of a secure system at a low level in the ladder of necessities for a WSN.

Nevertheless, security is critical in many WSN applications where sensitive data is exchanged. The values measured by the sensors can be very significant with respect to the implicit or explicit information that they carry. For example, WSNs can be deployed in a hostile environment that prevents any physical communication with the base station, such as battlefields. On the other hand, attackers can capture nodes and use them to gain access to confidential messages exchanged in the WSN [4].

The attacker model assumed in this paper is based on the physical access and hence compromise of any sensor node or base station. The attacker may also try to eavesdrop the data transmitted in the WSN. Service availability and data confidentiality are major requirements for any WSN but they should be achieved with minimal processing and energy requirements. Being energy efficient as compared to other public key schemes, IBE based on ECC will be the building block for our contribution in this paper.
We highlight the topic of key management in WSNs since it defines the robustness of the whole WSN against attacks that try to obtain the private keys of the sensor nodes. Usually, a single trusted base station is assumed to exist and to act as the central private key generator (C-PKG). This imposes a single point of failure in networks that are highly exposed to adversaries. Also, the C-PKG can publish the secret keys of any node without being detected. It can even impersonate the identity of any node with the help of a network master key by exploiting its private key to sign or decrypt its messages as there is no way to distinguish between the honest sensor node and the malicious C-PKG [5]. Moreover, the involvement of multiple base stations provides reduction in power consumption [6], which mitigates the effects of additional overhead caused by securing the WSN. All these factors motivate the use of a decentralized private key generator (D-PKG) which is described in section 5.

III. RELATED WORK

Different key management schemes were suggested to manage and establish keys among the sensor nodes. Most of these techniques are based on symmetric keys due to their relative resource efficiency. We present the most symmetric [7] and asymmetric key management schemes proposed for WSNs.

A. Symmetric-key Management Schemes

- Pair-wise key pre-distribution: Every node stores a pair-wise key with every other node in the WSN. If this node is compromised, it will be the only one affected as the communications among the other nodes remain secure. Obviously, such as solution is not scalable and cannot be applied for large WSNs.

- Master-key-based pre-distribution: A master key is pre-distributed to all the nodes in the WSN, so that the nodes can establish pair-wise keys using it along with random numbers exchanged with the other nodes. Although memory savings are obtained, the WSN is not resilient; once the master key is captured from a node, the whole WSN becomes exposed. LEAP [18] addressed this point by erasing the master key from a node after it establishes a pair-wise key. But if nodes are captured before the establishment of pair-wise keys, for example once added, LEAP becomes useless.

- Base station participation: The base station is the central authority as it often shares a secret key with every node in the WSN. This is called the “SPINS” mechanism [19]. When a node wants to communicate with another node, the base station sends both nodes a pair-wise key encrypted with the corresponding shared key. Other than assuming that a base station cannot be compromised, this scheme is not scalable due to having the traffic directed towards one point.

- Probabilistic key scheme: Every node is provided with a randomly chosen key ring \( k \) from a key pool \( P \) which is randomly selected from a huge key space. Two nodes can establish a pair-wise key by obtaining a common key in their key rings. The resilience of this mechanism is not perfect because when a node is compromised, the attacker has a probability \( k/P \) to successfully attack a link between the compromised node and another one [20].

B. Public-key Management Schemes

Public key-based key management schemes were also examined even though many researchers eliminated the potential of applying asymmetric algorithms in WSNs.

Using the public/private key pairs, the nodes can establish unique pair-wise keys with each other as was done in [8]. Koblitz and Miller developed Elliptic Curve Discrete logarithm Problem (ECDLP) which is a modification of ECC to establish pair-wise keys. Pairing-based cryptography (PBC) which is an extension of IBE is the practical implementation of ECC in WSNs. The secret key of every node is given as \( s.N \) where \( s \) is the master key of the network authority, while \( N \) is the key derived from its identity [9]. It should be impossible to deduce \( s \) from \( s.N \) in case the node was captured. The application of the scheme on WSNs is based on a common key between two nodes having public keys \( A \) and \( B \), which is found as \( H(e(s.A,B)) = H(e(s.B,A)) \) where \( e \) is a bilinear function and \( H \) is a key derivation function obtained from a one-way hash function. The scheme was proved to be efficient and promising for limited devices due to its low required computations and fast execution.

The issue of refreshing the keys in IBE systems was not tackled in the context of WSNs specifically but Balfe et al. [10] highlight the importance of performing this action and present a secure algorithm to achieve new pairs of keys in an identity-based scheme.

We note that all the suggested schemes assume that a single base station acts as the trusted authority and they do not account for possible attacks against it.

IV. PRELIMINARIES

Before explaining the architecture of our proposed system, we provide a background about IBE-based key management systems.

A. IBE-based Key Management Systems

IBE is an encryption technique that aims to provide public key encryption while avoiding the need for public key infrastructure for the distribution of keys. It makes use of the unique identities of the users in order to perform key distribution and encryption of messages. The authors of [3] argue that IBE is the ideal encryption scheme for WSNs although it only became feasible upon the introduction of PBC.
IBE-based key management offers the following advantages in the context of WSNs:

- It makes key management easier than traditional public key schemes since the public key is the identity itself or a function of it. The identities can be obtained from the exchanged messages without introducing additional communications overhead with its high energy requirements.

- It surpasses the symmetric key management systems that usually count on a pool of keys at every sensor node. This creates a problem of scalability in WSNs that can include hundreds or thousands of sensors, since the number of keys increases with the increase of number of nodes [12].

Four operations define the identity-based key management system:

1. Setup: Generation of the pair of public and private master keys using security parameters. The PKG is the only entity that possesses the master private key.
2. Extract: Generation of the private key specific to every node using its identity and the master private key of the system.
3. Encrypt: Producing the ciphertext of a plaintext message using the public master key and the identity of the node which is the node’s public key.
4. Decrypt: Producing the plaintext message of a ciphertext using the private key of the sensor node and the public master key [12].

Our IBE-based key system keeps all the phases and adds to them Refresh and Recover phases, as in [10]. A Revocation phase is also presented.

V. SYSTEM ARCHITECTURE

Our proposed solution is inspired from [10], [13], and [14] to achieve a decentralized wireless network with identity-based key management system. We extend the refreshing scheme suggested in [10] to match the nature of a distributed PKG (D-PKG).

The overall system architecture is shown in Figure 1. The WSN consists of m sensor nodes and n base stations such that n is much less than m. Every sensor node can be reached by r neighbouring base stations with minimal number of hop counts. The base stations can directly interact with each other via a separate channel, and are powerful nodes that do not suffer from the limitations of a regular sensor node. The identity of the node acts as its public key and its corresponding private key is initially stored upon deployment. However, these keys are not used during the whole lifetime of the WSN and thus need to be refreshed every period of time.

Our key management scheme includes phases for Initialization, Refresh, Recover, Revocation, and Encryption/Decryption. Note that we use the terms base stations and D-PKG nodes interchangeably.

Figure 1: Proposed System Architecture

A. Initialization

Upon deployment, the network manager stores in every sensor node its private key corresponding to the public key derived from its identity. In addition, a hash function \( H \) that satisfies the identity-based property [15] is installed inside all the entities of the network.

The keys are primarily calculated offline using a master key \( X \) which is then subdivided into \( n \) portions according to the threshold-cryptographic scheme suggested in [14]. The \( n \) shares of the master key \( X = (X_1, X_2, \ldots, X_n) \) are distributed among the \( n \) base stations in a \( t \)-over-\( n \) scheme such that the base station nodes act as a D-PKG. Thus, initially, every sensor node has its private key \( s.k \), its unique identity ID acting as its public key, the hash function \( H \), a list of the trusted base stations, the threshold number \( t \) that is needed to establish its new private key, and a timer that determines the lifetime of its keys. The public key of the sensor node is obtained as \( Pb = H(ID) \) and the keys are refreshed every time period \( T \).

B. Refresh

When the keys in time period \( T \) expire, each node needs to obtain new keys corresponding to period \( T \) following \( T \). It chooses a new identity (ID2) and sends a request message for refreshing the key to \( t \) trusted base stations:

\[
\text{Request}_{\text{Refresh}} = ID_1 || ID_2 || T_1 || T_2
\]

The sensor node chooses the base stations according to their proximity to it in order to avoid additional energy consumption due to contacting far ones. Every D-PKG node \( i \) uses Request_{Refresh} to obtain the value of \( ID_2 \). It then computes its part of the private key using its share \( X_i \) of the master key to get \( s.k_{2,i} = H(ID2)X_i \), and replies directly to the requesting node via:

\[
\text{Reply}_i = H(ID_1, ID_2, T_1, T_2) || \text{Encrypt}_{ID_1}(sk_{2,i})
\]

C. Recover

Each node receives \( t \) replies and recovers the key shares from base stations. Then, it obtains \( sk_{2,i} \) after
decryption $\text{Encrypt}_{ID}(sk_{2,i})$ with its private key $sk_1$ and calculates its new secret key $sk_2$ as:

$$sk_2 = \sum_{j=1}^{t} X_j H(ID_2) = \sum_{j=1}^{t} sk_{2,j}$$

\section*{D. Revocation}

A revocation list is issued by the base stations to identify malicious base stations and sensor nodes in the system. We use the idea of revocation threshold from [8] and modify it to fit into decentralized WSNs.

Every sensor node monitors its neighbouring nodes. Once it observes an abnormal activity of any of its neighbours, it sends an accusation message to the closest trusted base station. This message should be signed by the private key of the node. Each base station updates all other base stations to ensure that the whole network topology is covered. When the number of accusations reaches a certain threshold, the base stations classify the accused node as compromised and update all the sensor nodes through a revocation list. The criteria used to determine whether a node should accuse its neighbours are outside the scope of this paper, but many studies have been made to define a model of accusation [4, 16].

\section*{E. Encryption/Decryption}

At time period $T_i$, a node $n_1$ communicates with node $n_2$ by encrypting the message with $ID(n_2)$ at $T_i$. On the other hand, $n_2$ decrypts the message using the private key it has at $T_i$. We assume that encryption and decryption algorithms in our scheme are based on ECC.

\section*{VI. EVALUATION AND ANALYSIS}

We evaluate our proposed system in terms of security and performance. First, we analyse the security gains in general and provide sample scenarios for possible attacks that our scheme can endure. Then, we discuss the energy savings by evaluating the communications and computations overhead introduced to the WSN in order to achieve this level of security.

\subsection*{A. Security}

The following security gains are achieved by our proposed scheme:

- Confidentiality: Only authorized nodes can reveal the content of the messages sent to them. This is because the messages are encrypted using IBE/ECC that takes the identity of the node as the public key. Decryption is then performed by the node, whether it is a sensor node or a base station, since it is the only entity that has the corresponding private key.

Next, we introduce different possible attacks that the WSN can be subjected to and show how our architecture is resilient to them:

- When a base station is compromised, the attacker will only have access to this station’s share of the master key. No further information is available since the sensor nodes are the only ones that know their private keys. At least $t$ base stations should be compromised for the attacker to be able to calculate the private keys. When labelled as non-trusted, this base station will no longer receive updates from its peer base stations nor refresh requests from the sensor nodes.

- When a sensor node is compromised, the attacker will have access to its current private key only. Assume that the attacker has collected a set of messages previously exchanged with this node, it won’t be able to decrypt the messages since the current copy of the key is different from the one used to encrypt the previous messages. The node will be later isolated by the network through the revocation list and no future communications are performed with it.

- Assume that a malicious node wants to clone the identity of another sensor node. It uses its identity and requests its corresponding private key. The $t$ base stations will reply with $\text{Reply}_i = H(ID_1, ID_2, T_i, T_j) \mid \text{Encrypt}_{ID}(sk_{2,i})$. The cloner extracts $\text{Encrypt}_{ID}(sk_{2,i})$. However, it will fail to obtain $sk_{2,i}$ since it does not have the corresponding private key of ID. Encrypting $sk_{2,i}$ also protects against eavesdroppers that collect the Reply messages to build the new secret key.

\subsection*{B. Energy Consumption}

\subsubsection*{1. Computational Overhead}

In our scheme, the sensor nodes encrypt and decrypt the transmitted messages in WSN. They also sign the accusations sent to the base stations. These operations introduce additional computational overhead to the functionality of the sensor nodes in order to achieve a secure system and thus act as an additional source of energy consumption. However, we reduce this energy consumption by adopting ECC for encryption/decryption processes, which requires less energy as compared to other public key schemes. It has been shown that ECC-160 consumes 5 times less energy than RSA-1024 while achieving the same level of security [1, 2]. A further amount of energy is saved since the public key of the other nodes can be easily obtained from well-known information.
which is their identities. Of course, implementing a symmetric key scheme achieves better performance in terms of energy, but introduces the problem of key distribution, which is a very serious issue in WSNs.

2. Communications Overhead

More messages are exchanged by the sensor nodes as they need to contact multiple PKG nodes which results in additional power consumption. However, the impact of this cost is reduced by the fact that the closest base stations are favoured since the transmission energy is proportional to $d^2$, where $d$ is the distance to the base station.

We use the scenario described in [17] in order to compare the energy consumed in our system due to key refreshing with that of a single-base-station system that does not perform key refreshing. The WSN in this case consists of 30 sensor nodes and six (6) base stations distributed in a 30x30 meter squared field. We assume that the keys are refreshed every round in which 100 packets are exchanged. Every packet consists of 200 bits and every sensor needs to contact 3 base stations in order to obtain its new keys. We also assume that the transmission range of each sensor node is 10 meters, and that the energy consumed to transmit one bit over a distance of 1 meter is 0.1 nJ/bit-m$^2$ while the energy consumed to receive one bit is 50 nJ/bit.

![Figure 2: Energy Consumption per System](image)

It is very important to note that a careful distribution of the base stations will result in more enhancements concerning energy consumption [17]. As shown in Figure 2, by implementing our scheme for the above scenario, we achieve a secure WSN with 30% less overall system energy consumption than a regular WSN that does not apply key refreshing operations. This is justified by making use of the multiple base stations to achieve energy savings in regular operations such as forwarding. Thus, the communications overhead introduced by our system actually results in a more energy-efficient but highly secure WSN.

VII. CONCLUSION

In this paper, we presented a decentralized identity-based key management system for wireless sensor networks. The scheme was evaluated in terms of performance and security. The novelty of our approach lies in adding Refresh and Recover phases for WSNs with a decentralized private key generator represented by multiple base stations. We achieved a securely robust WSN that can be placed in hostile environments while improving energy-efficiency, as compared to other public key infrastructures.

REFERENCES


