Secure and efficient Key Exchange Mechanism for Heterogeneous Connected Objects

Sofiane Aissani, Tarek Fettioune, Nafaa Maizia, Mohamed Mohammedi, Mawloud Omar

Noname manuscript No.
(will be inserted by the editor)

Secure and efficient Key Exchange Mechanism for Heterogeneous Connected Objects

Sofiane AISSANI · Tarek FETTIOUNE · Nafaa MAIZIA · Mohamed MOHAMMEDI · Mawloud OMAR

Received: date / Accepted: date

Abstract The fast evolution in microelectronics and the emergence of wireless communication technologies, have allowed the appearance of the promising field of Internet of Things (IoT). The latter is more and more present in the human life, that is why it becomes essential to secure the communications done with the connected objects. Almost all communicating systems attach great importance to security, consequently, on the cryptographic key management. The existing key management schemes for conventional networks are relatively resource-intensive, that is why they are not adequate for resource-constrained networks like IoT, especially since the nodes' capabilities are heterogeneous. In this paper, we focus on exchanging and updating of cryptographic keys among the IoT objects often limited in resources, where we propose a new form of key exchange based on the mechanism of concealing encryption keys, while exploiting the misused spaces in the header fields of the exchanged packets by the communication standards, such as ZigBee, BLE, WiFi.

Keywords Internet of Things · Key Exchange · BLE · Zigbee · WiFi · BAN logic.

1 Introduction

The Internet of Things (IoT) is a technical achievement of ubiquitous computing, where technology is naturally integrated into everyday objects. Very promising this concept opens the way to a multitude of scenarios based on the interconnection between the physical and virtual world. The IoT, machine to machine and connected objects covers a wide range of applications either in the industrial world or in everyday life, such as home automation, e-health, agriculture, smart city, logistics, transport, industry, etc. [3]. Security is a very important predicative in the communications of IoT objects. Therefore, in order to guarantee authentic communication, it is necessary to establish very powerful key management schemes allowing the IoT objects to exchange information in full security. Likewise, to establish keys by respecting the heterogeneity of resource constrained IoT objects.
Numerous solutions to this issue have been proposed in the literature, some of them are based on the reduction of communication cost to have a minimum energy consumption. Some others, focus on reducing the storage cost. There are also the solutions that apply to reduce both storage and communication. However, all these protocols are implemented as independent services where each of them generates its own traffic. This leads us to wonder if it is possible to hide encryption keys in data packets sent out by IoT objects. This means that we do not generate specific messages to send these keys to avoid the network saturation.

Sofiane AISSANI [1] E-mail: aissani.sofiane@gmail.com ·
Tarek FETTIOUNE [1] E-mail: fettiounetarek@gmail.com ·
Nafaa MAIZIA [1] E-mail: otherman94@yahoo.fr ·
Mohamed MOHAMMEDI [1] E-mail: mohammedi.mohamed88@gmail.com ·
Mawloud OMAR [2] E-mail: mawloud.omar@univ-eiffel.fr ·
[1] LIMED laboratory, Faculty of Exact Sciences, University of Bejaia, Bejaia, Algeria.
[2] LIGM, ESIEE Paris, University Gustave-Eiffel, Noisy-le-Grand, France

In order to remedy the aforementioned issues, we must find unused or poorly exploited spaces in the data packets sent out by the IoT objects. In this paper,we present an extended version of the paper we published in [14], where we have studied the possibilities of concealing keys in a message sent out by an IoT object in certain communication standards such as ZigBee, BLE and WiFi, we propose a key Exchanging scheme called Secure and efficient Key Exchange Mechanism for heterogeneous connected objects (SKEM), based on the concealment of encryption keys in data packets of communication protocols (ZigBee, BLE and WiFi). In other words, our proposal is a new form of key exchange based on the mechanism of hiding encryption keys, which exploits the misused spaces in the header fields of the packets exchanged by the communication standards in the IoT. The purpose of the proposed scheme is twofold, the sending of keys in a hidden way and without suspecting that they are there, and also the exploitation of some sent messages to exchange the cryptographic keys to reduce the communication overhead. In this paper we have enriched the work we published in [14] by:

- A formal analysis of some security properties by using the formalism of BAN logic.
- More simulation results are presented and discussed.
- More explanations in different sections and subsections are done.

BAN logic is a decidable logic of belief, proposed on 1989 in [12], to analyze the security of authentication protocols. It allows the construction of proofs on the reliability and authenticity of the exchanged information, using a set of deduction rules. In BAN logic, it is assumed that all communications are vulnerable to passive and active attacks. The properties that can be verified by the BAN logic are: authentication of message senders, verification of the messages freshness, verification of the messages reliability. A demonstration of a given property in BAN logic involves the following four steps: First, we transform the transmitted messages into formulas of the BAN logic; second, we fix the security objectives to achieve in logical formulas; third, we represent the hypotheses in the form of logical formulas, and fourth and last, we proof the fixed objectives using the deduction rules, the assumptions and the messages.

The following parts of the paper are organized as follows. In Section 2, we discuss some research work in the framework of key management in the IoT. In Section 3, we present our key management scheme detailing the steps which make it up. In Section 4, we formally validate the proposed scheme by using the formalism of the BAN logic. Afterwards, in Section 5, we provide the obtained results following the performance evaluation. Finally, in Section 6, we conclude the paper with a conclusion summarizing the key points, which were discussed, as well as the prospects we hope to achieve soon.

2 Related work

The existence of a key management system in any communication system is paramount to provide the security service. Consequently, numerous reflections and research work have been conducted to mend the security issues related to IoT. The aim is to propose a high-performance security mechanism, which ensures not only a high level of security, but also shows a significant energy conservation. In [2], Kim et al. proposed a new Authentication and Key Management (AKM) mechanism to ensure security between IoT devices and access point. Kim et al.'s scheme is based on two communication standards, 802.11 to provide key management process and the 802.1X standard to ensure the authentication service. The Station-side authentication server (SAS) provides both of these communication standards. In [5], Sciancalepore et al. proposed a key management protocol called KMP (Key Management Protocol) for mobile and industrial systems of IoT. This protocol is integrated into the second layer of the 802.15.4 protocol stack to provide security services for the different IoT scenarios. As well, it relies on an Elliptic Curve Diffie Hellman (ECDH) fixed exchange with Elliptic Curve Qu-Vanstone (ECQV) implicit certificate, stamps exchange, authentication of exchanged messages. All this, is to ensure mutual authentication among IoTs objects and freshness in the key derivation. In [6], Porambage et al. proposed a key management

scheme based on an implicit authentication mechanism, and uses a certificate for WSNs in distributed IoT applications. The proposed authentication scheme allows sensor nodes and end users to authenticate and communicate securely. Consequently, end users (human beings or virtual entities) collaborate with on-board devices to obtain particular information or service. In[8], Zhang and Pengfei proposed a key management scheme called EHKM (Efficient and Hybrid Key Management) for heterogeneous wireless sensor networks. The proposed scheme is based on a new key management method, which is a combination of both elliptic curve cryptography (ECC) and symmetric cryptography. The aim behind this is to improve the network security, while taking into account the limited resources of sensor nodes. In [9], Bi and Xu proposed a key management scheme called SNKM (security node-based key management) based on the cluster for WSN. This scheme is based on the security of nodes which constitute clusters in the WSNs. It relies on different sort of keys, where nodes can choose different keys for encryption and authentication depending on the types of data packets. The major goal of this solution has been to improve security and reduce the energy consumption of nodes in general, especially cluster-head, which plays a very important role in WSN. Because if a bad behavior is detected at the cluster-head level, it must be immediately replaced. In [10], Zhang et al. proposed a key management scheme called EDDK (energy-efficient distributed deterministic key management scheme) with certificate, which can support the mobility of sensors in the WSN. This proposal operates on a clustering sensor network composed of simple static sensors, simple mobile sensors, cluster-heads, Base station, and certification authority. In [7], Suganthi and Sumathy proposed a key management scheme termed EEKM (Energy Efficient Key Management) for fixed WSNs. This protocol is based on non-energy consuming polynomial functions for managing all cryptographic keys. As well, it operates on a homogeneous WSN composed of simple sensor nodes and a base station. In this protocol the reduction of energy consumption is due to the use of polynomial functions, which are not energy intensive. Besides, the transmission cost is optimized by the fact that no cryptographic key is transmitted instead of less expensive parameters which are transmitted. In [4], Seo et al. proposed a certificateless key management scheme termed CL-EKM (CertificateLess-Effective Key Management). CL-EKM operates on a clustered WSN consisting of a number of sensors, clusters-heads and a base station. This protocol manages five types of keys, namely: public/private key pairs, individual keys, master key pairs, cryptographic key pairs, as well as cluster keys. This proposal seems to be very resistant to cloning, and provides security in the present and the past with the regeneration of cluster keys as soon as a node joins or leaves the cluster. This uncertificated scheme reduces the communication cost due to authentication and certificate-based encryption. However, it is expensive in terms of storage because of the multiple keys, which a node must store particularly the key pair (master/cryptographic) that connect him to each of his neighbors. The main shortcoming of the reviewed schemes is that they are implemented as independent modules, which generate their own traffic when creating or updating cryptographic keys. To overcome the aforementioned security pitfalls and address other aforementioned issues, we propose a key management scheme based on the hide hybrid key management system concealment. The goal is to ensure both, the sending of the keys in a hidden way and without suspecting that it is there. As well, the exploitation of some messages sent out to exchange cryptographic keys.

3 The proposed scheme

3.1 Network model and assumptions

We consider several types of networks using different communication standards,in this work, we focused just on three standards that are WiFi, ZigBee and BLE, but our approach can be applied to other communications standards. Each network type is supervised by a base station which is assumed to be sufficiently secure and has no constraints in terms of energy and computation capacities. Each standard consists of a base station, which collects data from IoT objects, and communicates with other base stations of other communication

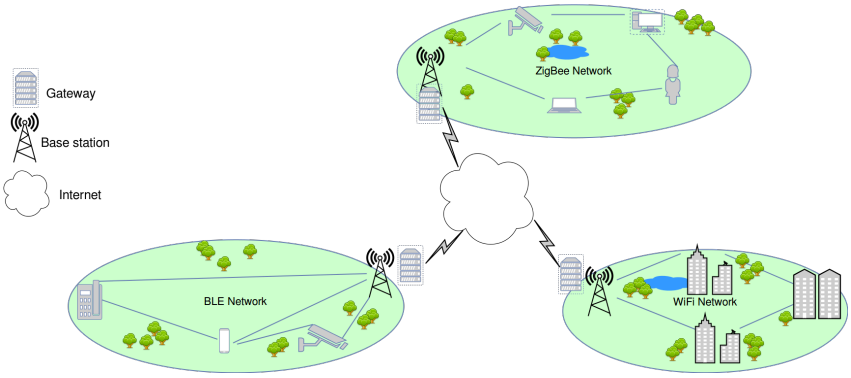

Fig. 1: Proposed scheme architecture

standards through a gateway (see Fig. 1). We did not use this technique only to hide the transmitted key parts, but also to exploit space in some fields, so as not to have surpluses in communication.
In this paper we are interested only in how different objects exchange their keys, we are not interested in steps like predistribution or key generation. accordingly, we assume that there are secret keys shared between two IoT objects which want to update their keys. The key exchange process is performed by the objects themselves, and this process is executed periodically after each time T. At each round T, an object O_i generates a new private key $\widehat{K}$, sends it to the object O_j, eventually, through the base stations. Afterwards, it waits for an acknowledgment to overwrite the old key K. Since the parameter T has a strong influence on the key update frequency. Consequently, it must be adapted according to the application type for which the network is deployed. Therefore, a compromise between the desired security level and the required performances must be found. In fact, a reduced value of T offers a good resistance of the system against passive security attacks. However, it causes and adds an extra computational burden. If T increases, the computation will be light, but with a relatively lower robustness. The message transmission is carried out through a gateway from where it is used to transfer data packets to much more advanced actions, such as traffic filtering or protocol translation to different network layers.

3.2 The proposed scheme operations

In this sub-section, we explain the different operations of the SKEM scheme, which is a key exhanging mechanism for IoT, which generates no communication overhead. The proposed scheme is based on the same principle as that of μKMS proposed in [1], which is a new form of key exchange based on the mechanism of concealing encryption keys. Our proposal exploits the misused spaces in the header fields of the packets exchanged by IoT communication standards, such as ZigBee, BLE and WiFi. In order to illustrate the proposed scheme operation, we first study the concealment spaces possibilities according to the communication standards. Consequently, in our study, we are only interested in the following communication standards:

1. According to ZigBee 802.15.4 *standard:* As explained in [13], we can exploit four fields in a ZigBee frame: the sequence number, radius, source and destination Addresses.

– **Sequence number:** The sequence number is used twofold, first: to reorder the packets of the same message in the reception, and second: to to acknowledge a messages. For every message, it is generated randomly, then it is incremented in each packet. This is why it is exploitable for dissimulation. We use eight bits to code this field. As discussed in [1], the number of exploitable bits in the sequence number field is given by the following equation:

$$ES_{Seq} = 8 - \log_2(M \cdot 102^{-1}) \quad (1)$$

– **Radius:** this field represents the number of jumps that a packet can make on the network, it is fixed by the transmitter, and it is decremented in each node. The worst case is that the packet can cross all the nodes before reaching its destination. If η is the network size, we need $\log_2(\eta)$ bits, so the usable bits number on this field, E_{Rad} is given by the following formula :

$$ES_{rad} = 8 - \log_2(\eta) \tag{2}$$

– **Source Address and Destination Address:** In [1], we proposed to modify the addresses locally, so as to exploit only the necessary space, in environnement which uses few objects (as in a body area network), we can use only the necessary space and not all the space allocated to addresses. The gate will keep a mapping table between local addresses and global addresses. Each address is encoded on 16 bits. The necessary space for addressing all the objects is $\log_2(\eta)$ bits. Therefore, the exploitable space on the two addresses ES_{adr} is given by the following formula:

$$ES_{adr} = 32 - 2 \cdot \log_2(\eta) \tag{3}$$

Total exploitable bits of ZigBee standard

The total exploitable space in the different fields is formulated as follows:

$$T_{ZigBee} = 48 - 3 \cdot \log_2(\eta) - \log_2(|M| \cdot 102^{-1}) \tag{4}$$

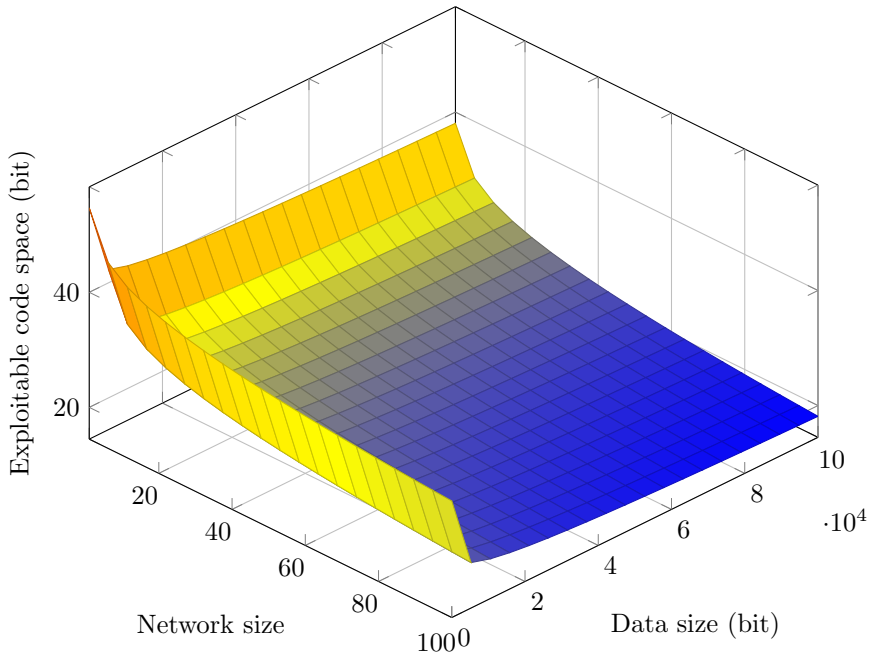

Fig. 2: Exploitable space for concealment in ZigBee standard

Fig. 2 illustrates the exploitable bits number, for concealment, by the ZigBee communication protocol depending on the network size and that of the messages. We observe from Fig. 2 that the exploitable space, sometimes exceeds the 45 bits.

2. According to WiFi standard The WiFi frame contains four exploitable fields including: Duration ID, Sequence control, Source and destination addresses.

– **Duration ID:** this field role is like that of radius in ZigBee standard, it is coded on 16 bits, if η is the network size and the packet must pass through all the objects, then we

need $\log_2(\eta)$ bits to encode it. That is why exploitable space $ES_{DurationID}$ on this field is expressed by the following formula:

$$ES_{DurationID} = 16 - \log_2(\eta) \tag{5}$$

– **Sequence control:** this field have the same utility of the field sequence number in ZigBee, it is encoded on 16 bits, if M is the size of the information we want to send then, the number of frames needed to send it, is $M/2312$, where 2312 is the maximum size of the *frame body* field. The number of bits needed to code this number is $\log_2(M/2312)$ and therefore the exploitable space in bits in this field ES_{seq} is expressed by the following equation:

$$ES_{seq} = 16 - \log_2(|M| \cdot 2312^{-1}) \tag{6}$$

– **Source Address and Destination Address:** As in ZigBee we propose local addresses and global addresses which are managed by the gateway, this field is composed of four addresses (address 1, address 2, address 3, and address 4) and each of them is coded on 48 bits, so all is coded on 192 bits, if η is the network size, then we need $\log_2(\eta)$ bits to encode each address. Therefore, the exploitable space on these addresses, ES_{adr} is expressed by the following formula:

$$ES_{adr} = 192 - 4 \cdot \log_2(\eta) \tag{7}$$

Total exploitable bits of WiFi: the addition of the previous fields is given by the following formula:

$$T_{WiFi} = 224 - 5\log_2(\eta) - \log_2(|M| \cdot 2312^{-1}) \tag{8}$$

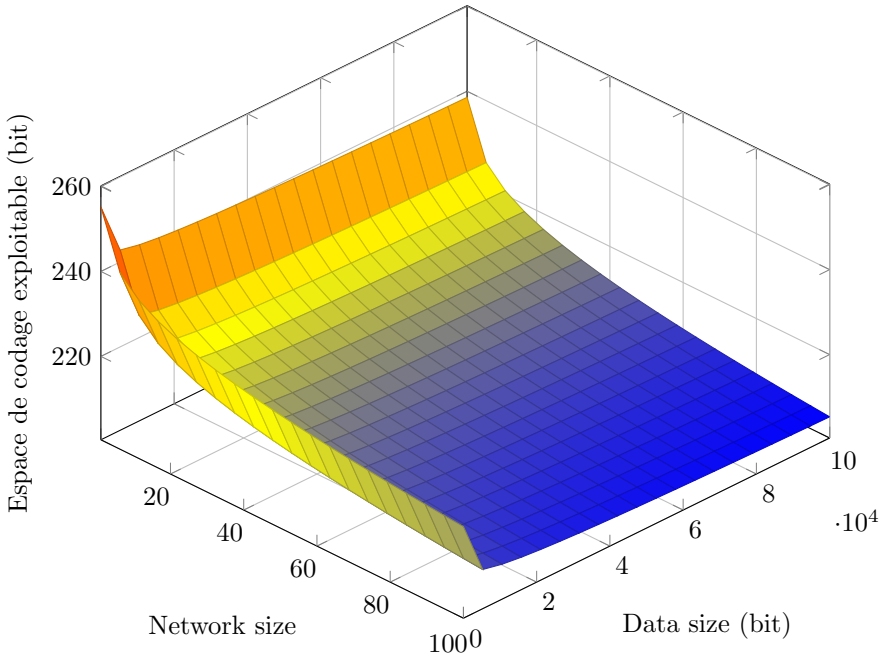

Fig. 3: Exploitable space for concealment in WiFi standard

Fig. 3 represents the number of exploitable bits, for concealment, by the communication scheme WiFi, according to the network size and that of the messages. We notice from Fig. 3 that the exploitable space, sometimes exceeds the 240 bits.

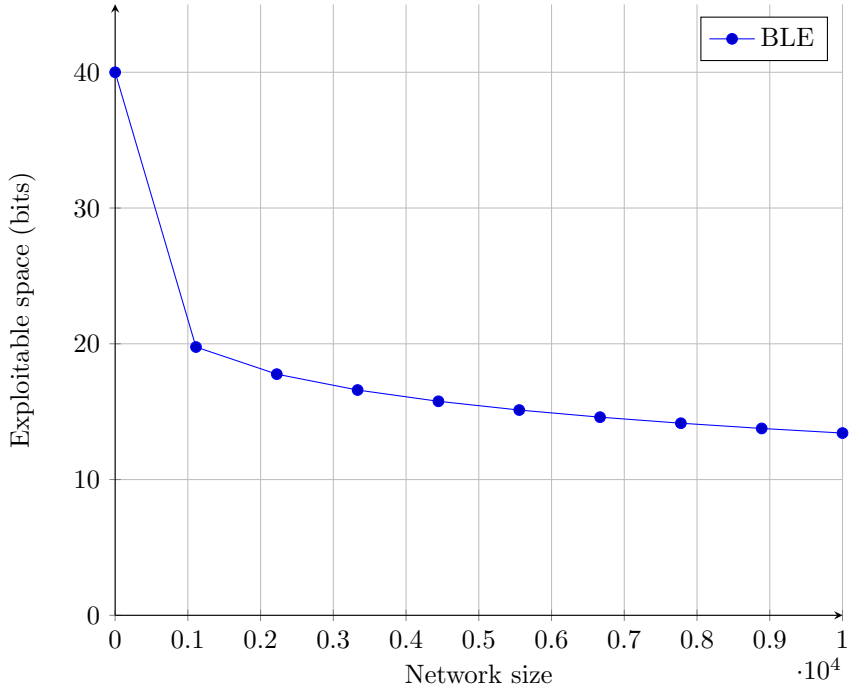

Fig. 4: Exploitable space for concealment in BLE standard

3. According to Bluetooth Low Energy BLE standard The BLE (Bluetooth Low Energy) frame contains two exploitable fields including: Preamble and the Access address.

– **Preamble:** The Preamble is a 1 byte value, it is used for timing estimation at the receiver.If η is the network size, so we need $\log_2(\eta)$ bits to encode it. Consequently, the exploitable space on this $ES_{Preamble}$ field is expressed by the following formula:

$$CS_{Preamble} = 8 - \log_2(\eta) \tag{9}$$

– **Access address:** this field is coded on 32 bits, if η is the network size, then we need $\log_2(\eta)$ bits to encode it. Therefore, the exploitable space on this ES_{adrAcc} field is expressed by the following formula:

$$ES_{adrAcc} = 32 - \log_2(\eta) \tag{10}$$

Total exploitable bits of BLE: the addition of the previous fields expressed by the following formula:

$$T_{BLE} = 40 - 2 \cdot \log_2(\eta) \tag{11}$$

Fig. 4 represents the number of exploitable bits, for concealment, by the BLE communication protocol, depending on the network size. We notice from the Fig. 4 that exploitable space, sometimes exceeds 35 bits.

3.2.1 Negotiation phase

This phase takes place between the base stations of the heterogeneous networks studied (ZigBee, WiFi, and BLE) after computing the number of exploitable bits in each packet of these protocols. To negotiate on the minimum number of exploitable bits, this negotiation operates on the protocol, which contains the minimum number of exploitable bits to ensure that the number of exploitable bits sent is greater than the number of exploitable bits in the receiver protocol.

3.2.2 Addressing and gateway role

The base station (BS) locally maintains a table denoted T, which contains two fields $\langle @\mathtt{Local}, @\mathtt{IP}\rangle$ for each object. The latter provides global addresses for nodes, which have local addresses when transmitting messages, this in order to have more exploitable bits. After the negotiation phase described above, the IoT objects exchange informational messages, where is included a key a parts of the key which is hidden in the exploitable bits, to communicate with the BS. The latter will communicate with the BS of the other networks through a gateway, which serves to convert the norm (the frame) exchanged. Before this conversion we must extract the encryption key parts and then convert the frame and insert it. Once the transaction has taken place, the base station receives the converted frame from the gateway, and sends it to the appropriate objects (see Fig. 5).

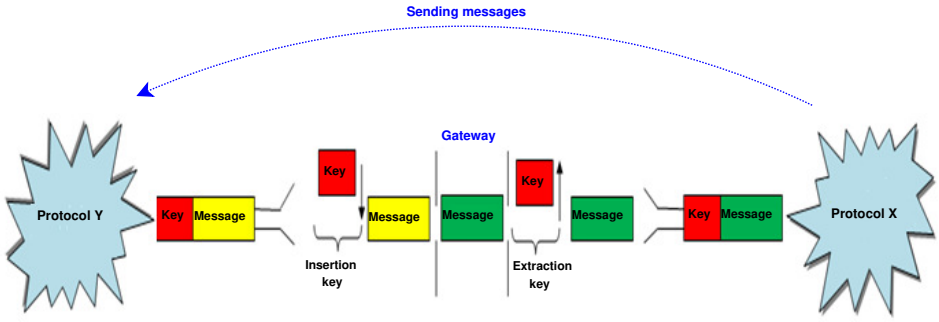

Fig. 5: Message transmission between two different communication standards

3.2.3 Key update phase

This process, in which the symmetric keys of the IoT objects are refreshed, runs periodically. We denote K as the current key of the object $\mathsf{O_i}$, and $\widehat{\mathsf{K}}$ as its new key. We name the waiting time for the re-execution of this process a round. This phase runs between the IoT objects, so we distinguish two possible cases to update a key:

- **Case-1:** the objects are in the same network, to update their keys, it is enough that the object $\mathsf{O_i}$ generates its new key $\widehat{\mathsf{K}}$, concatenates it with a stamp Na, encrypts it with the old key K, and send it to the concerned object $\mathsf{O_j}$. The latter decrypts the received message and retrieves the new key $\widehat{\mathsf{K}}$ to acknowledge the receipt of the key $\widehat{\mathsf{K}}$. Then, it encrypts the received stamp Na with the key $\widehat{\mathsf{K}}$, and sends it in its turn to the concerned object $\mathsf{O_i}$. After the check of the stamp, the object $\mathsf{O_i}$ removes the old key K.
- **Case-2:** when two objects $\mathsf{O_i}$ and $\mathsf{O_j}$ are on two different networks, to update their keys, the object $\mathsf{O_i}$ generates its new key $\widehat{\mathsf{K}}$. Then, concatenate with a stamp Na, encrypt the all with the old key K, and send it to the base station $\mathsf{BS_i}$. Upon the latter receiving the message via the gateway, it transfers the received message from the object $\mathsf{O_i}$ to the base station $\mathsf{BS_j}$ which is in the second network, in its turn the base station $\mathsf{BS_j}$ send it to the appropriate object $\mathsf{O_j}$. The latter decrypts the received message and retrieves the new key $\widehat{\mathsf{K}}$ received by $\mathsf{BS_j}$, to acknowledge the receipt of the key, it encrypts the stamp Na with the key $\widehat{\mathsf{K}}$, then sent it in its turn at the base station $\mathsf{BS_j}$, this last one will send it to the $\mathsf{BS_i}$ which will send it to the sender object $\mathsf{O_i}$.

4 Security analysis

The logic of belief BAN proposed in [12] is used to proof the integrity of the new exchanged key, and the authentication of both the sender object of the new key, and the acknowledgment of the receiver object.

4.1 Formal analysis

A demonstration of a given property in BAN logic involves the following four steps: First, we transform the transmitted messages into formulas of the BAN logic; second, we fix the security objectives to achieve in logical formulas; third, we represent the hypotheses in the form of logical formulas, and fourth and last, we proof the fixed objectives using the deduction rules, the assumptions and the messages.

The notation used through BAN logic are summarized in Table 1.

Table 1: Notation used in BAN logic

Notation	Description
$P \models X$	P believes X
$P \triangleleft X$	P sees X
$P \vdash X$	P says X
$P \overset{K}{\leftrightarrow} Q$	K is only known by P and Q
$P \Rightarrow X$	P control X
$\#(X)$	The data X is fresh (updated)

The rules we use during the validation of our protocol, defined in [12], are as follow:

$$R1 : \frac{P \models Q \overset{K}{\leftrightarrow} P, P \triangleleft \{X\}_K}{P \models Q \vdash X}, \tag{12}$$

$$R2 : \frac{P \models Q \Rightarrow X\,, P \models Q \vdash X}{P \models X}, \tag{13}$$

$$R3 : \frac{P \models \#X\,, P \models V \vdash X}{P \models V \models X} \tag{14}$$

4.2 The rules semantic

- **Rule-R1:** If P believes in the integrity and confidentiality of the key K shared between P and Q, and that P sees the message X encrypted by the key K, then P will believe that Q said X.
- **Rule-R2:** If P believes that Q controls X, and that Q says X, then P will believe in the authenticity of X.
- **Rule-R3:** If P believes in the freshness X and P believes that V said X, then P will believe that V believes X.

proof

- **Step-1: idealization of the scheme** The first message sent by the new key creator is modeled by :

$$I1 : O_j \triangleleft \{O_i \overset{\widehat{K}}{\leftrightarrow} O_j\}_K. \tag{15}$$

O_i represents the sender object and O_j the receiver object, while $\widehat{K}$ denotes the new key. The acknowledgment of the receiver object is modeled by the following formula:

$$I2 : O_i \triangleleft \{O_i \overset{\widehat{K}}{\leftrightarrow} O_j\}_{\widehat{K}}. \tag{16}$$

– **Step-2: security objectives** The objectives to achieve after the key refresh process are the authenticity and confidentiality of the new key $\widehat{K}$. This is expressed by the goal G1 in the following logical formula:

$$G1 : O_j \models O_i \overset{\widehat{K}}{\leftrightarrow} O_j. \tag{17}$$

The sender object O_i have to be sure the receiver object O_j has received the new key $\widehat{K}$. This is the second goal G2, expressed by the following logical formula:

$$G2 : O_i \models O_j \models O_i \overset{\widehat{K}}{\leftrightarrow} O_j. \tag{18}$$

– **Step-3: hypotheses** It is obvious, that before the refreshing process, the object O_j believes in the integrity and authenticity of the shared key with the object O_i. This assumption is expressed by the following logical formula:

$$H1 : O_j \models O_i \overset{K}{\leftrightarrow} O_j. \tag{19}$$

It is also normal to assume that O_i, which is the creator of the new key, believes in the integrity and authenticity of $\widehat{K}$, we modal this by the following formula:

$$H2 : O_i \models O_i \overset{\widehat{K}}{\leftrightarrow} O_j \tag{20}$$

With the same reasoning we can assume that the object O_i believes in the freshness of the new key $\widehat{K}$. This is expressed as follows:

$$H3 : O_i \models \#(O_i \overset{\widehat{K}}{\leftrightarrow} O_j). \tag{21}$$

We also assume that the receiver object O_j knows that it is the object O_i, which has control over the new key $\widehat{K}$, this gives us the following formula:

$$H4 : O_j \models O_i \Rightarrow O_i \overset{\widehat{K}}{\leftrightarrow} O_j. \tag{22}$$

– **Step-4: obtaining objectives** The last step but not the least, in BAN logic proof, is the demonstration of the fixed goals G1 and G2. Indeed, by using the rules (R1, R2 and R3), the hypotheses (H1,H2,H3 and H4) and formulas obtained by the idealization of the scheme (I1 and I2), we have to reach the objectives(G1 and G2).
First, by applying the rule R1, we will have:

$$\frac{O_j \models O_i \overset{K}{\leftrightarrow} O_j, O_j \triangleleft \{O_i \overset{\widehat{K}}{\leftrightarrow} O_j\}_K}{F_1 = O_j \models O_i \vdash O_i \overset{\widehat{K}}{\leftrightarrow} O_j} \quad R1(I_1, H1) \tag{23}$$

Second, applying the rule R2 on the previous result F1 and the hypothesis H4, we will have:

$$\frac{O_j \models O_i \Rightarrow O_i \overset{\widehat{K}}{\leftrightarrow} O_j, O_j \models O_i \vdash O_i \overset{\widehat{K}}{\leftrightarrow} O_j}{F_2 = O_j \models O_i \overset{\widehat{K}}{\leftrightarrow} O_j} \quad R2(F1, H4) \tag{24}$$

SThe obtained result is what we wanted to achieve in the first goal, F2 = G1. We have thus formally demonstrated that with the proposed scheme the IoT objects will believe in the integrity and authenticity of the key $\widehat{K}$. To reach the second objective, we begin by applying the rule R1 on the hypothesis H2 and the obtained formula from the idealization of the second message I2, we obtain then:

$$\frac{O_i \models O_i \overset{\widehat{K}}{\leftrightarrow} O_j, O_i \triangleleft \{O_i \overset{\widehat{K}}{\leftrightarrow} O_j\}_{\widehat{K}}}{F_3 = O_i \models O_j \vdash O_i \overset{\widehat{K}}{\leftrightarrow} O_j} \quad R1(H2, I2) \tag{25}$$

Applying the rule R3 on the previous result (F3) and the hypothesis H3, we will have:

$$\frac{O_i \models \#(O_i \overset{\widehat{K}}{\leftrightarrow} O_j)\ , O_i \models O_j \vdash O_i \overset{\widehat{K}}{\leftrightarrow} O_j}{F_4 = O_i \models O_j \models O_i \overset{\widehat{K}}{\leftrightarrow} O_j} \quad R3(H3, F3) \tag{26}$$

F4 = G2, so this is the second fixed goal, thus the object O_i will know that the object O_j believes in the new key $\widehat{K}$ and is ready to use it, the object O_i can now remove its old key K from its memory.

5 Performance evaluation

In order to demonstrate by experimentation the performance of the proposed scheme, the efficiency of the latter is studied by means of extended simulation tests, developed under Matlab/Simulink environment. To evaluate the proposed scheme, we chose to compare its performances against two other recent and relevant schemes proposed by Kim et al. [2] and Sciancalepore [5]. This, to highlight its efficiency in terms of the network lifetime, the energy consumption, and storage cost.

5.1 Simulation environment

To evaluate the effectiveness of our proposed scheme simulations were performed on three networks by using different communication standards, namely ZigBee, WiFi, and BLE. Each standard contains 150 objects, which are randomly scattered in a square area of $(200 \cdot 200)$ m^2. In the experimentation of our scheme performances described below, we assume the following assumptions:

- All objects are fixed throughout the simulation period.
- We assume that at each 10 second period, the objects send data to the base station, the data size is 1000 bytes.
- The value of the initial energy is the same for all the sensor nodes in the same network.
- We assume that a link exists between two objects if the Cartesian distance between them is smaller than their communication range.
- The communication ranges of objects are 25, 20 and 15 meters in the networks 1, 2 and 3 respectively.
- The following computations: distance, energy consumed, and remaining energy are performed on all the objects during the simulation period.

Although there are several radio models which are used to compute the energy consumed by nodes, but in this paper, we rely on the most common radio model defined by Heinzelman et al. [11]. Based on the latter, the energy consumption cost for both transmission and reception of a k-bit data packet between two nodes on a distance d is valued in joules. Consequently, the energy consumed during the transmission of a k-bit data packet is expressed by the following equation:

$$E_{tr} = k \cdot (E_{elec} + E_{amp} \cdot d^2) \tag{27}$$

Where E_{elec} denotes electrical energy, this is the energy consumed by the electronic transmitter for one bit, E_{amp} denotes the energy of amplification, it is about the energy consumed by the amplifier of transmission. Similarly, when a node receives a k-bit data packet, its energy consumption which is expressed by the following formula:

$$E_{rec} = k \cdot E_{elec} \tag{28}$$

In the performed simulations, we consider the amplification energy equal to 10^{-8}, 10^{-9} and 10^{-10} Joules, and the electrical energy equal to $5 \cdot 10^{-8}$, $4 \cdot 10^{-8}$ and $3 \cdot 10^{-8}$ Joules in the objects of the networks 1, 2 and 3 respectively. The simulation parameters we considered are summarized in the table 2.

Table 2: Simulation parameters

Parameters	**Network 1**	**Network 2**	**Network 3**
Area	$200 \cdot 200$ m^2	$150 \cdot 150$ m^2	$100 \cdot 100$ m^2
Number of nodes	150	150	150
Communication range of the objects (Meters)	25	20	15
Key size (Bits)	80	80	80
Average size of a transmitted message (Bytes)	1000	1000	1000
Initial energy (Joules)	600	500	400
Energy amplification (Joules)	10^{-8}	10^{-9}	10^{-10}
Electric energy (Joules)	$5 \cdot 10^{-8}$	$4 \cdot 10^{-8}$	$3 \cdot 10^{-8}$

5.2 Performance evaluation criteria

In this subsection, we describe the efficiency evaluation criteria, as well as the simulation metrics used.

5.2.1 Criteria for evaluating effectiveness

- **Storage cost:** is the number of memory units used to store the necessary identifiers and keys for all nodes in the network at a given time.
- **Energy consumption:** is the addition of the energies consumed by the nodes to send and receive a certain number of data packets, so it is computed for the whole network at a given time.
- **A life nodes number:** Since the network life time have many definitions according to the application field, we have chosen to study the number of living nodes, which are the nodes which still possess an energy strictly greater than 0.

5.2.2 Simulation metrics

- **Scalability:** it represents increasing the number of nodes in the network and evaluating the criteria mentioned above at the first deployment of the network.
- **Cryptographic algorithm:** this is the fact of varying the encryption key size according to the cryptographic algorithm as cited in the table 3.

5.3 Obtained results

5.3.1 Results following the scalability

Energy consumption Fig.6 shows the total consumed energy as function of the nodes number in the three networks. We observe from the simulated graph that the energy consumption of the AKM scheme [2], and KMP scheme [5] is increasing fast. Because the energy that all nodes spend to establish keys and create messages to send out with a size large enough. However, the energy consumed in the proposed scheme shows a slow increase linked only to the supposed periodic exchanges in the existing messages. According to the energy model described in [11], used in our simulations, the consumed energy during communications is largely higher than the consumed energy in treatments. This explains the good performance of our protocol in energy consumption. Indeed, in our proposal, we do not generate any new message for sending the cryptographic keys, they are hidden in the network operating messages, thanks to the proposed steganography mechanism. The reduction of communication overhead leads to a reduction in energy consumption.

Storage cost Fig.7 shows the storage cost as function of the nodes number in the network. As shown in Fig.7, more the nodes number increases, more the storage cost increases too,

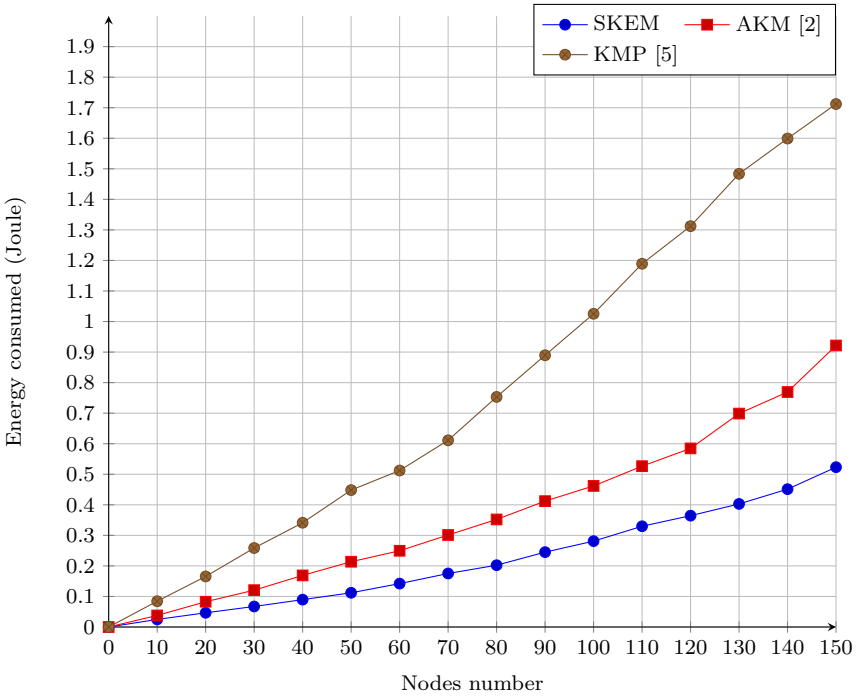

Fig. 6: Energy consumption in function of node number

that for the AKM scheme [2], especially for the KMP scheme[5]. This is consistent with the proposed scheme, which shows the negligible increase. The reason for that, is the fact that our scheme stores only the encryption key, whereas others schemes use a large number of extra keys and parameters including certificates for KMP scheme and the base keys and derivative for AKM scheme which it stores since the first deployment. Therefore, the proposed scheme is adapted even with scaling up.

5.3.2 Results following the cryptography algorithms

As cryptographic algorithms influence on the key size, we mentioned the algorithms used, as well as the key size of each algorithm [1] in Table 3.

Table 3: Key sizes according to the cryptographic algorithm used

Used algorithm	Key Size κ (bit)
2TDEA	80
3DES-112, 3TDEA	112
RC4, RC2, and AES-128	128
3DES-168	168
AES-192	192
AES-256	256

Storage cost Fig.8 shows the storage cost load by the three networks nodes according to the key size of the used cryptographic algorithm.

[1] 2TDEA operates under three keys, in which the first and third ones are identical. The conventional size of each key is 56 bits, and hence, the natural total size would be 112 bits. However, as reported in [16], the 2TDEA crypto-system becomes solid starting from a total key size of 80 bits. That's why we keep the lower configuration of keys in order to square with the requirements imposed by the network, which is quite constrained in terms of resources.

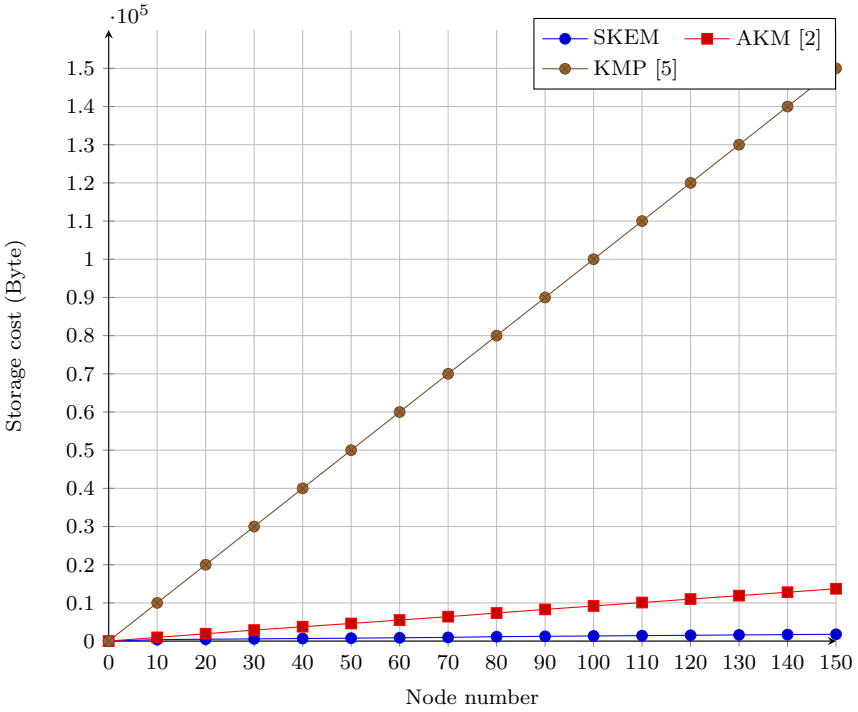

Fig. 7: Storage cost as function of the nodes number

From Fig.8 we can observe that the storage cost of the AKM scheme [2] increases a little faster compared to that of KMP [5]. This is because of the difference of the two parameters, namely the message size and that of the cryptographic algorithm key of each scheme. From the simulated graph we also note that more the size of these two parameters increases more the storage cost increase too.

Energy consumption Fig. 9 shows the energy consumed by the network nodes according to the key size of the used cryptographic algorithm. The graph illustrates that the energy consumption of AKM scheme [2] increases a little bit faster compared to that of KMP [5]. This is because of the difference of the two parameters, namely the size of the messages and those of the keys of the cryptographic algorithms of each scheme. Also, we note that more the size of these two parameters increases, more the consumed energy increases. However, the proposed scheme curve is stable, since the encryption keys are hidden in the content of the transmitted messages, and doesn't influence the communication overhead.

The figure 10 illustrates the variation of a life nodes number as function of time, we note that the nodes start to die in 300 sec for KMP [5] and 500 for AKM [2], and the life nodes number decreases fast and reaches 0 in 1700 seconds for KMP, and in 2000 for AKM. While the first dead in HH-KMS occurs only in 700 seconds, and the networks have always more than 30 a life nodes in 2000 seconds. This is explained by the energy consumption, as HH-KMS is the most performing in conserving energy, it gives the best results in a life nodes.

6 Conclusion

Security in the IoT environment is carried out through key management system, and this poses a very delicate issue, not only to the exchanged message security, but also to the consumed energy and storage cost, which has to be minimized. In this paper, we proposed a key exchange mechanism, which generates no communication overhead. The proposed mechanism is based on a new form of key exchange, which is the key transmission in a hidden way, after having exploited the misused spaces in the header fields of the exchanged

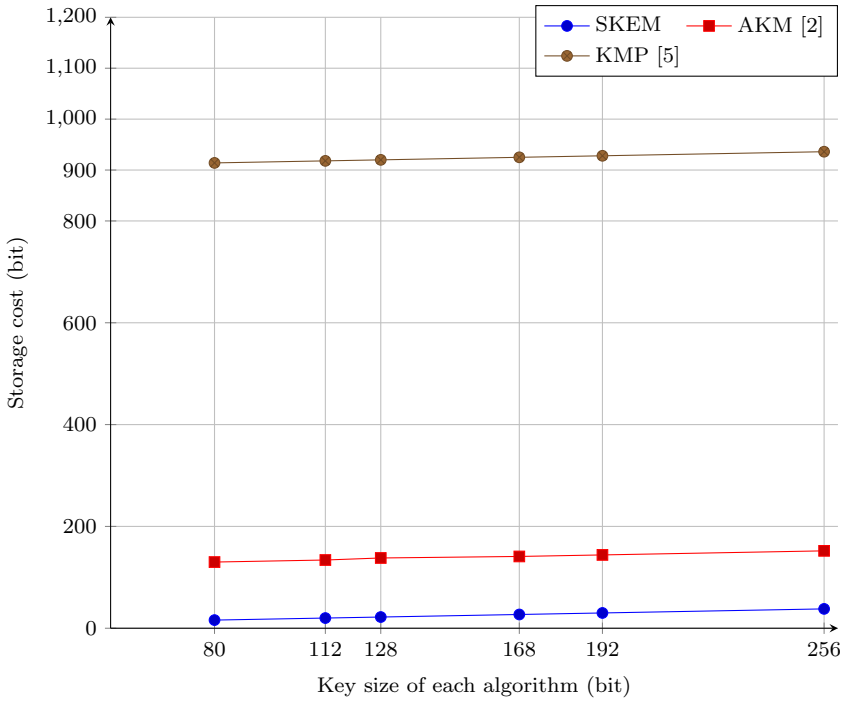

Fig. 8: Storage cost as function of the cryptography algorithm

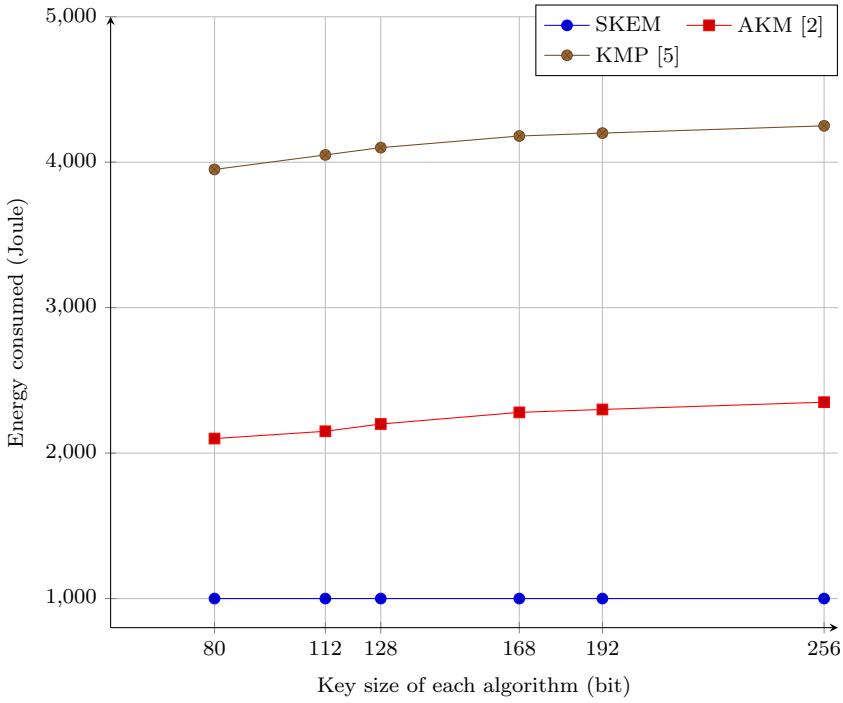

Fig. 9: Energy consumption in function of the cryptography algorithm.

data packets. Through security analysis and performance evaluation, we have been able to validate the proposed scheme. The obtained results are very competitive compared to those of the concurrent schemes, while providing a very high level of security. In our future work, we plan to study other communication standards to expand the use spectrum of the proposed mechanism. Another research area will focus on the implementation of the proposed scheme

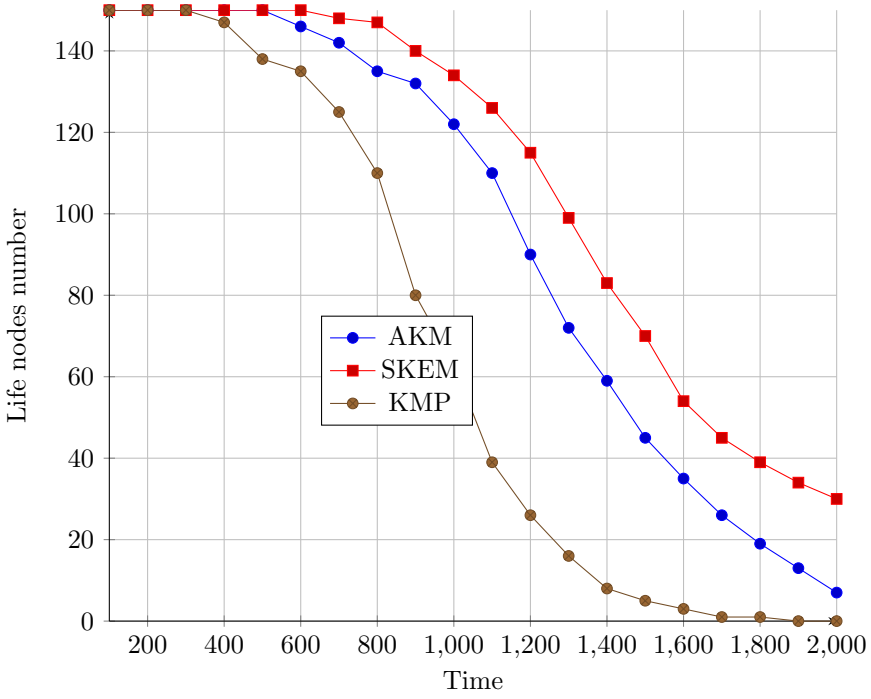

Fig. 10: The living nodes number as function of time

on real IoT objects to test the applicability of the concealment idea. Finally, we plan to extend the proposed scheme within the framework of exploiting the idea of concealment in other issues.

Acknowledgment

This work has been sponsored by General Directorate for Scientific Research and Technological Development, Ministry of Higher Education and Scientific Research (DGRSDT), Algeria.

References

1. Aissani, S., Omar,M., Tari,A., & Bouakkaz, F.(2018). μKMS: Micro key management system for WSNs. IET Wireless Sensor Systems, 8(2), 87−97.
2. Kim,K.W., Han,Y.H. & Min,S.G. (2017). An authentication and key management mechanism for resource constrained devices in IEEE 802.11-based IoT access networks. Sensors (Basel), 17(10), 1−14.
3. Billet,B. (2015). Systéme de gestion de flux pour l'Internet des Objets intelligents. Calcul paralléle, distribué et partagé [cs.DC]. Université de Versailles-Saint Quentin en Yvelines, ⟨NNT : 2015VERS012V⟩. ⟨tel − 01166047⟩, 2015.
4. Seo,S.H., Won,J. S. Sultana, & Bertino,E. Effective key management in dynamic wireless sensor networks. In IEEE Transactions on Information Forensics and Security. 10(2), 371−383.
5. Sciancalepore,S., Capossele,A. Piro,G., Boggia,G. & Bianchi,A.(2015). Key management protocol with implicit certificates for IoT systems. In Proceedings of the 2015 Workshop on IoT Challenges in Mobile and Industrial Systems,(IoT-Sys.'15), 37−42.
6. Porambage,P., Schmitt,C., Kumar,P., Gurtov,A. & Ylianttila,M.(2014). Two-phase authentication protocol for wireless sensor networks in distributed IoT applications. In Proceedings of the IEEE Wireless Communications and Networking Conference, 2728−2733.
7. Suganthi,N. & Sumathy,V. 2014. Energy efficient key management scheme for wireless sensor networks. International Journal of Computers Communications & Control. 9(1), 71−78.
8. Zhang,Y. & Pengfei,J. (2014). An efficient and hybrid key management for heterogeneous wireless sensor networks.In Proceedings of the 26th Chinese Control and Decision Conference. 1881−1885.
9. Bi,J.N. & Xu,E. (2013). An Energy-efficient security node-based key management protocol for WSN. Instruments, Measurement, Electronics and Information Engineering, series: Applied Mechanics and Materials, editor: Trans Tech Publications. 347(350), 2117−2121.

10. Zhang, X., He,J. & Wei,Q. (2010).EDDK: Energy-efficient distributed deterministic key management for wireless sensor networks. EURASIP Journal on Wireless Communications and Networking.1−11.
11. Heinzelman,W., Chandrakasan,A. & Balakrishnan,H.(2000). Energy-efficient communication protocol for wireless sensor networks. In Proceedings of the 33rd Annual Hawaii International Conference on System Sciences. 3005−3014.
12. Burrows,M., Abadi,M. & Needham,R.A.(1990) logic of authentication. Journal ACM Transactions on Computer Systems, 8(1), pp. 18−36.
13. Aissani,S. & Abbache, B.(2020). Secure Key Management System Integrated in Cell−LEACH. Wireless Pers Commun, 112(4) 2109−2129.
14. Aissani,S., Fettioune, T., Maizia, N., Mohammedi,M. & Omar,M.(2018). Overheadless Key Exchange Mechanism for Heterogeneous Connected Objects. The 7th International Symposium ISKO-Maghreb on Knowledge Organization in the perspective of Digital Humanities: Research and Application, 90−97.
15. Présentation de matlab. http ://www.samuelboudet.com/fr/matlab. Accessed on 25th September (2018).
16. Elaine, B. (2016). Recommendation for Key Management. NIST Special Publication 800-57 Part 1, DOI: 10.6028/NIST.SP.800-57pt1r4.