A Complete Security Framework for Wireless Sensor Networks: Theory and Practice

Christophe Guyeux, Abdallah Makhoul, Ibrahim Atoui, Samar Tawbi, Jacques Bahi

A Complete Security Framework for Wireless Sensor Networks:

Article · January 2015

DOI: 10.4018/ijitwe.2015010103

5 authors, including:

Christophe Guyeux
University of Franche-Comté
144 PUBLICATIONS **527** CITATIONS
SEE PROFILE

Abdallah Makhoul
University of Franche-Comté
52 PUBLICATIONS **257** CITATIONS
SEE PROFILE

Samar Tawbi
Lebanese University
17 PUBLICATIONS **17** CITATIONS
SEE PROFILE

Jacques M. Bahi
University of Franche-Comté
236 PUBLICATIONS **1,342** CITATIONS
SEE PROFILE

A Complete Security Framework for Wireless Sensor Networks: Theory and Practice

Christophe Guyeux [1], Abdallah Makhoul [1], Ibrahim Atoui [2], Samar Tawbe [2], Jacques M. Bahi [1]

[1] University of Franche-Comté (FEMTO-ST)

17 Av du Maréchal Juin, 90000

Belfort Cedex, France

E-mails: {jacques.bahi, christophe.guyeux, abdallah.makhoul}@univ-fcomte.fr

[2] Lebanese University

Faculty of sciences

Hadath Lebanon

E-mails: ibrahim.atoui@hotmail.com, stawbi@ul.edu.lb

Abstract

Wireless sensor networks are often deployed in public or otherwise untrusted and even hostile environments, which prompt a number of security issues. Although security is a necessity in other types of networks, it is much more so in sensor networks due to the resource-constraint, susceptibility to physical capture, and wireless nature. Till now, most of the security approaches proposed for sensor networks present single solution for particular and single problem. Therefore, to address the special security needs of sensor networks as a whole we introduce a security framework. In our framework, we emphasize the following areas: (1) secure communication infrastructure, (2) secure scheduling, and (3) a secure data aggregation algorithm. Due to resource constraints, specific strategies are often necessary to preserve the network's lifetime and its quality of service. For instance, to reduce communication costs, data can be aggregated through the network, or nodes can go to sleep mode periodically (nodes scheduling). These strategies must be proven as secure, but protocols used to guarantee this security must be compatible with the resource preservation requirement. To achieve this goal, secure communications in such networks will be defined, together with the notions of secure scheduling and secure aggregation. The concepts of indistinguability, nonmalleability, and message detection resistance will thus be adapted to communications in wireless sensor networks. Finally, some of these security properties will be evaluated in concrete case studies.

1 Introduction

In the last few years, wireless sensor networks (WSN) have gained increasing attention from both the research community and actual users. As sensor nodes are generally battery-powered devices, the critical aspects to face concern how to reduce the energy consumption of nodes, so that the network lifetime can be extended to reasonable times. Therefore, energy conservation is a key issue in the design of systems based on wireless sensor networks. In the literature, we can find different techniques to extend the sensor network lifetime [1]. For example, energy efficient protocols are aimed at minimizing the energy consumption during network activities. However, a large amount of energy is consumed by node components (CPU, radio, etc.) even if they are idle. Power management schemes are thus used for switching off node components that are temporarily not needed [2, 3, 4, 5]. Other techniques suitable to reduce the energy consumption of sensors are data acquisition (i.e. sampling or transmitting) reduction as data fusion and aggregation [6, 7, 8, and 9].

On the other hand, sensor networks are often deployed in unattended even hostile environments, thus leaving these networks vulnerable to passive and active attacks by the adversary. The communication between sensor nodes can be eavesdropped by the adversary and can forge the data. Sensor nodes should be resilient to these attacks. Therefore, one of the major challenges in such networks is how to provide connection between sensors and the base station and how to exchange the data while maintaining the security requirements and taking into consideration their limited resources. Until now, most of the security and saving energy approaches proposed for sensor networks propose single solution for particular and single problem. Therefore, to address the special security needs of sensor networks as a whole we introduce a security framework. In our framework, we emphasize the following areas: (1) secure communication infrastructure, (2) secure scheduling, and (3) a secure data aggregation algorithm.

Secure communication infrastructure: In wireless sensor networks, a sensor node generally senses the data and sends to its neighbor nodes or to the sink (base station). Stationary adversaries equipped with powerful computers and communication devices may access whole WSN from a remote location. For instance, an intrusion detection system detects the different types of attacks and sends the report to base station. It uses all nodes or some special nodes to detect these types of attacks. These nodes co-operate with each other to take the decision and finally send the report to the base station. It requires lots of communication between the nodes. If adversary can trap the message exchanging between the nodes then he can easily tamper the messages and send the false information to the other nodes. Secure communication is a necessary condition in order to make the network smooth so that nodes can send data or exchange the message securely. In our paper, we provide the definition of a communication system for WSNs, and define some of the required security properties (indistinguability, nonmalleability, and message detection resistance) dedicated to sensor networks.

Secure scheduling: The main objective of a secure scheduling is to prolong the whole network lifetime while fulfilling the surveillance application needs. In other words, a common approach is to define a subset of the deployed nodes to be active while the other nodes can sleep. In this paper, we present a novel

scheduling algorithm where only a subset of nodes contribute significantly to detect intruders and prevent malicious attacker to predict the behavior of the network prior to intrusion. We present a random scheduling to solve this issue, by guaranteeing a uniform coverage while preventing attackers to predict the list of awaken nodes.

Secure data aggregation algorithm: Since the deployed nodes are separated, they need to cooperatively communicate sensed data to the base station. To reduce the amount of sending data, an aggregation approach is often applied along the path from sensors to the sink. However, usually the carried information contains confidential data. Therefore, an end-to-end secure aggregation approach is required to ensure a healthy data reception. End-to-end encryption schemes that support operations over cypher-text have thus to be found for private party sensor network implementations. These schemes offer two main advantages: end-to-end concealment of data and ability to operate on cipher text, then no more decryption is required for aggregation. Unfortunately, nowadays these methods are very complex and not suitable for sensor nodes having limited resources. This example illustrates the necessity to adapt and rethink cryptographic definitions and protocols to the WSN context. In this paper, we propose a secure end-to-end encrypted-data aggregation scheme. It is based on elliptic curve cryptography that exploits a smaller key size. Additionally, it allows the use of higher number of operations on cypher-texts and prevents the distinction between two identical texts from their cryptograms.

As discussed above, two strategies are in common used to extend the network lifetime while preserving its performance, namely the use of scheduling processes reducing the number of awakened nodes, or the data aggregation through the network to reduce the communication costs (transmissions are very energy-consuming operations). These strategies take particular forms in hostile contexts, taking into account the fact that adversaries can benefit from potential weakness that can result from these strategies. The problem is that, currently, there is no formal and rigorous framework to evaluate these strategies and, generally speaking, to study the security of wireless sensor networks.

Cryptographic tools are often relaxed for performances, and embedded into schemes or protocols, hoping that the resulting processes maintain a certain level of security. The underlying idea is that, due to resource constraints and other specific requirements of wireless sensor networks, conventional cryptography is not relevant to bring solutions to security issues that concretely occur in WSNs. The objective of this article is thus to provide a rigorous framework inherited from cryptography, that satisfies the level of exigency usually attained in that domain. More precisely, we will give definitions of security in wireless sensor networks compatible with the ones commonly formulated in cryptology.

The main contributions of this document can be summarized as follows. A complete security framework for wireless sensor networks, investigating all the aspects of security related to WSNs, is proposed. This rigorous theoretical framework encompasses: (1) secure communication (communication systems, indistinguability, nonmalleability, and message detection), (2) cryptographically secure scheduling, and (3) secure aggregation of data. The formalism is as

rigorous as possible; it is inspired by equivalent formulations in cryptography, but is compliant with all the particularities of a WSN. A first scheduling algorithm proven as cryptographically secure is then given as another contribution of this paper, and a discussion of the efficiency vs. security compromise is then outlined. Finally, a method for secure data aggregation is proposed. Compared with up-to-date state of the art in homomorphic encryption for data aggregation in WSN, the number of allowed aggregation functions is increased here.

The remainder of this research work is organized as follows. In the next section we provide a general presentation for security in WSN. A rigorous formalism for secure communications in wireless sensor networks is presented in section 3, in which the notions of communication systems, indistinguability, nonmalleability, and message detection resistance are formalized rigorously in the WSN framework. In Section 4, the notion of secure scheduling is defined and applied on a given example. The security of data aggregation is studied in Section 5. This section also contains a complete case study of a wireless sensor network that supports secure aggregations. This research work ends by a conclusion section, where our contribution is summarized and intended future work is presented.

2 Security in WSN: General presentation

Wireless nature of communication, lack of infrastructure and uncontrolled environment improve capabilities of adversaries in WSN. Stationary adversaries equipped with powerful computers and communication devices may access whole WSN from a remote location. They can gain mobility by using powerful laptops, batteries and antennas, and move around or within the WSN. In this section, we consider a WSN where nodes communicate together by sending data publicly. These *transmitted data* contain a *message* whose confidentiality must be preserved. For instance, transmitted data is the cryptogram of a message, modulated in an electromagnetic radiation, or the message is dissimulated into the electromagnetic radiation by using a spread spectrum information hiding technique. Wireless communication helps adversaries to perform variety of attacks. A secure communication can be used to provide the following general security goals:

One-wayness (OW) The adversary who sees transmitted data is not able to compute the corresponding message.

Indistinguability (IND) Observing transmitted data, the adversary learns nothing about the contained message.

Non-malleability (NM) The adversary, observing data for a message m, cannot derive another data for a meaningful message m' related to m.

The OW and IND goals relate to the confidentiality of messages sending through the WSN. The IND goal is, however, much more difficult to achieve than the one-wayness. Non-malleability guarantees that any attempt to manipulate the observed data to obtain a valid data will be unsuccessful (with a high probability).

The power of a polynomial attacker (with polynomial computing resources) very much depends on his/her knowledge about the system used to transform *information* in *data*. The weakest attacker is an outsider who knows the public embedding algorithm together with other public information about the setup of the system. The strongest attacker seems to be an insider (he/she is inside the network) who can access the extraction device (recovering information from data) in regular interval. The access to the extraction key is not possible as the extraction device is assumed to be tamperproof.

An *extraction oracle* is a formalism that mimics an attacker's access to the extraction device. The attacker can experiment with it providing *data* and collecting corresponding *information* from the oracle (the attacker cannot access to the decryption key). In general, the public-key WSN can be subjected to the following attacks (ordered in increasing strength):

$$\begin{array}{ccccc} (NM, CDA2) & \rightarrow & (NM, CDA1) & \rightarrow & (NM, CIA) \\ \updownarrow & & \downarrow & & \downarrow \\ (IND, CDA2) & \rightarrow & (IND, CDA1) & \rightarrow & (IND, CIA) \end{array}$$

Figure 1: Relations among security notions

Chosen information attack (CIA) The attacker knows the embedding algorithm and the public elements including the public key (the embedding oracle is publicly accessible).

Nonadaptative chosen data attack (CDA1) the attacker has access to the extraction oracle before he sees a data that he wishes to manipulate.

Adaptative chosen data attack (CDA2) the attacker has access to the extraction oracle before and after he observes a data s that he wishes to manipulate (assuming that he is not allowed to query the oracle about the data s).

The security level that a public-key WSN achieves can be specified by the pair (goal, attack), where the goal can be either OW, IND, or NM, and the attack can be either CIA, CDA1, or CDA2. For example, the level (NM, CIA) assigned to a public-key network says that the system is nonmalleable under the chosen message attack. There are two sequences of trivial implications

- (NM, CDA2) $\Rightarrow$ (NM, CDA1) $\Rightarrow$ (NM, CIA),
- (IND, CDA2) $\Rightarrow$ (IND, CDA1) $\Rightarrow$ (IND, CIA),

which are true because the amount of information available to the attacker in CIA, CDA1, and CDA2 grows. Figure 1 shows the interrelation among different security notions. Consequently, we can identify the hierarchy of security levels. The top level is occupied by (NM, CDA2) and (IND, CDA2). The bottom level contains (IND, CIA) only as the weakest level of security. If we are after the strongest security level, it is enough to prove that our network attains the (IND, CDA2) level of security.

$[\![1; N]\!]$	$\rightarrow \{1, 2, \ldots, N\}$
f^K	$\rightarrow k^{th}$ composition of a function f
$\mathcal{X} \times \mathcal{Y}$	$\rightarrow$ the Cartesian product of the two sets
$\mathcal{X}^n$	$\rightarrow$ the Cartesian product $\mathcal{X} \times \mathcal{X} \times \ldots \times \mathcal{X}$ (n times)
maxA	$\rightarrow$ the greatest element in the set N
ln	$\rightarrow$ Neperian logarithm
$log_a(x)$	$\rightarrow$ the logarithm of x to base a
$[a,b[$	$\rightarrow$ the real interval $\{x \in \mathbb{N} \mid a \leq x < b\}$
S_n	$\rightarrow$ the n^{th} term of a sequence $S = (S_1, S_2 \ldots)$
$gcd(a,b)$	$\rightarrow$ the greatest common divisor of a and b
OW	$\rightarrow$ One-wayness
IND	$\rightarrow$ Indistinguability
NM	$\rightarrow$ Non-malleability
CIA	$\rightarrow$ Chosen information attack
$CDA1$	$\rightarrow$ Nonadaptative chosen data attack
$CDA2$	$\rightarrow$ Adaptative chosen data attack
Pr	$\rightarrow$ Probability of
Adv	$\rightarrow$ advantage of an adversary
$InSec$	$\rightarrow$ insecurity

Table 1: Notations

3 Rigorous Formalism for Secure Communications in WSNs

In this section, we present new principles formalism for secure communication in wireless sensor networks. We start by introducing some notations in table 1 in use in the remainder of this document, before defining the communication system in WSN and other security concerns.

3.1 Communication System in a WSN

Definition 1 (Communication system) Let $\mathcal{S}$, $\mathcal{M}$, and $\mathcal{K} = \{0,1\}^{\ell}$ three sets of words on $\{0,1\}$ called respectively the sets of transmission supports, of messages, and of keys (of size ℓ).

A *communication system* on $(\mathcal{S}, \mathcal{M}, \mathcal{K})$ is a tuple $(\tau, \mathcal{E}, inv)$ such that:

- $\tau : \mathcal{S} \times \mathcal{M} \times \mathcal{K} \to \mathcal{S}$, $(s,m,k) \mapsto \tau(s,m,k) = s'$, is the *insertion function*, which put the message m into the support of transmission s according to the key k, leading to the transmitted data s'.
- $\tau : \mathcal{S} \times \mathcal{K} \to \mathcal{M}$, $(s,k) \mapsto \mathcal{E}(s,k) = m'$, defined as the *extraction function*, which extract a message m' from a transmitted data s, depending on a key k.
- $inv : \mathcal{K} \to \mathcal{K}$, s.t. $\forall k \in \mathcal{K}, \forall (s,m) \in \mathcal{S} \times \mathcal{M}, \mathcal{E}(\tau(s,m,k), inv(k)) = m$, which is the function that can "invert" the effects of the key k, producing the message m that has been embedded into s using k.
- τ and $\mathcal{E}$ can be computed in polynomial time, and τ is a probabilistic algorithm (the same values inputted twice produce two different transmitted data).

k is called the embedding key and $k' = inv(k)$ the extraction key. If $\forall k \in \mathcal{K}$, $k = inv(k)$, the communication system through the WSN is said *symmetric* (private-key), otherwise it is *asymmetric* (public-key).

3.2 Indistinguability

Suppose that the adversary has two messages m_1, m_2 and a transmitted data s in his/her possession. He/she knows that s contains either m_1 or m_2. Our intention is to define the fact that, having all these materials, the key, and the insertion function (we take place into the (IND, CIA) context), he cannot determine with a non-negligible probability the message that has been embedded into s.

The difficulty of the challenge comes, for a large extends, from the fact that the insertion algorithm τ is a probabilistic one, which is a common-sense assumption usually required in cryptography.

Definition 2 An Indistinguability I-adversary is a couple (A_1, A_2) of nonuniform algorithms, each with access to an oracle $\mathcal{O}$.

Definition 3 (Indistinguability) for a public communication system in WSN $(\tau, \mathcal{E}, inv)$ on $(S, M, \{0,1\}^{\ell})$, define the advantage of an I-adversary A by

$$Adv_{\mathrm{A}}^{I-O} = Pr\left[\begin{array}{lll} k \xleftarrow{\$} \{0,1\}^{\ell} & & \\ (m_0, m_1, s) \leftarrow \mathrm{A}_1(k) & : & \mathrm{A}_2(k, s, m_1, m_2, \alpha) = b \\ b \leftarrow \{0,1\} & & \\ \alpha = \tau(s, m_b, k) & & \end{array}\right]$$

We define the insecurity of $S = (\tau, \mathcal{E}, inv)$ with respect to the Indistinguability as

$$InSec_S^{I-O}(t) = \max_{\mathrm{A}} \{Adv_{\mathrm{A}}^{I-O}\}$$

where the maximum is taken over all adversaries A with total running time *t*.

We distinguish three kinds of oracles:

- The Non-adaptative oracle, denoted $\mathcal{NA}$, where A_1 and A_2 can only access to the elements of the communication system.
- The Adaptative oracle, denoted $\mathcal{AD}1$, where A_1 has access the communication system and to an oracle that can in a constant time provide a message m' from any transmitted data $\tau\ (M', m', k')$, without knowing neither M' nor k' nor $inv(k')$. In this context, A_2 has no access to this oracle.
- The Strong adaptative oracle, denoted $\mathcal{AD}2$, where A_1 has access to the communication system and to an oracle that can in a constant time provide a message m' from any transmitted data $\tau\ (M', m', k')$, without knowing neither M nor k' nor $inv(k')$. In this context, A_2 has also access to this oracle but for the message $\tau\ (M, m_b, k)$.

3.3 Relation Based Non-malleability

In some scenarios, malicious nodes can integrate the WSN, hoping by doing so to communicate false information to the other nodes. We naturally suppose that communications are secured. The problem can be formulated as follows: is it possible for the attacker to take benefits from his/her observations, in order to forge transmitted data either by embedding erroneous messages, or sending data that appear to be similar with what a node is supposed to produce.

As wireless sensor networks have usually a dynamical architecture, the (dis)appearance of nodes is not necessarily suspected. Authentication protocols can be deployed into the WSN, but in some cases such authentication is irrelevant, because of its energy consumption, communication cost, or rigidity. We focus in this section on the possibility to propose a secured communication scheme in WSN that prevents an attacker to forge such malicious transmitted data. Such non-malleability property can be formulated as follows.

Definition 4 A Relation Based NM-adversary is a nonuniform algorithm A having access to an oracle $\mathcal{O}$.

Definition 5 (Relation Based Non-malleability) For a public communication system $(\tau, \mathcal{E}, inv)$ on $(\mathcal{S}, \mathcal{M}, \{0,1\}^{\ell})$, define the advantage of a NM-adversary A by

$$Adv_{\mathrm{A}}^{NM-O}(m) = Pr\begin{bmatrix} s \leftarrow S \\ k \xleftarrow{\$} \{0,1\}^{\ell} \quad : \quad \mathrm{m}' \in R(m) \\ s' \leftarrow \mathrm{A}(\tau(s,m,k)) \\ m' = \mathcal{E}(s',k) \end{bmatrix}$$

where $R : \mathcal{M} \to \mathcal{P}(\mathcal{M})$ is a function that map any message m to a subset of $\mathcal{M}$ containing messages related to m (for a given property). For instance, if we suppose that an attacker has inserted or corrupted some nodes in a network that measures temperature, he can make these nodes send wrong temperatures values fixed *a priori*.

We can now define the insecurity of $\mathcal{S} = (\tau, \mathcal{E}, inv)$ with respect to the relation Based Non-malleability as

$$InSec_{S}^{NM-O}(t) = \max_{\mathrm{A}} \left\{ \max_{m \in \mathrm{M}} \{Adv_{\mathrm{A}}^{NM-O}(m)\} \right\}$$

where the maximum is taken over all adversaries A with total running time t. Similar kinds of oracles than previously can be defined in that context.

3.4 Message Detection Resistance

We now address the particular case where transmitted data can contain or not an embedded message. For security reasons, it is sometimes required that an attacker cannot determine when information are transmitted through the network. For instance, in a video surveillance context, suppose that an attacker can determine when an intrusion is detected, or when something considered as suspicious is forwarded through the nodes to the sink. Then he/she can use this knowledge to deduce what kind of behavior is suspicious for the network, adapting so his/her attacks. Decoys are often proposed to make such attacks impossible: transmitted data do not always contain information; some of the communications are only realized to mislead the attacker. The quantity and frequency of these decoys must naturally respect the energy consumption requirement, and a compromise must be found on the message/decoy rate to face such attacks while preserving the WSN lifetime. However, such an approach supposes that the attacker is unable to make the distinction between decoys and meaningful communications. Such a supposition leads to the following definitions.

Definition 6 (Detection Resistance) A Detection Resistance DR-adversary is a couple (A_1, A_2) of nonuniform algorithms, each with access to an oracle $\mathcal{O}$.

Definition 7 (Message Detection Resistance) For a public communication system $(\tau, \mathcal{E}, inv)$ on $(\mathcal{S}, \mathcal{M}, \{0,1\}^{\ell})$, define the advantage of a DR-adversary A by

$$Adv_{\mathrm{A}}^{DR-O} = Pr\left[\begin{array}{l} M_0, M_1 \leftarrow S \\ k \xleftarrow{\$} \{0,1\}^{\ell} \\ m \leftarrow \mathrm{A}_1(k) \\ b \leftarrow [0,1] \\ \alpha = \{M_b, \tau(M_{\bar{b}}, m, k))\} \end{array} : \quad \mathrm{A}_2(m,k,\alpha) = M_b \right]$$

where the set defining α is a non-ordered one.

We define the insecurity of S = ($\tau, \mathcal{E}, inv$) with respect to the Message Detection Resistance as

$$InSec_S^{DR-O}(t) = \max_{\mathrm{A}} \left\{ Adv_{\mathrm{A}}^{DR-O} \right\}$$

where the maximum is taken over all adversaries A with total running time *t*. Similar kinds of oracles than previously can be defined in that context.

4 Secure Scheduling

4.1 Motivations

A common way to enlarge lifetime of a wireless sensor network is to consider that not all of the nodes have to be awakened: a subset of well-chosen nodes participates temporarily to the task devoted to the network [10, 11] (video surveillance of an area of interest, sensing environmental values ...), whereas the other nodes sleep in order to preserve their batteries. Obviously, the scheduling process determining the nodes that have to be awakened at each time must be defined carefully, both for guaranteeing a certain level of quality in the assigned task and to preserve the network capability over time. Problems that are of importance in that approach are often related to coverage, ratio of awaken vs sleeping nodes, efficient transmission of wake up orders, and capability for the partial network to satisfy, with a sufficient quality, the objectives it has been designed for. Existing surveillance application works focus on finding an efficient deployment pattern so that the average overlapping area of each sensor is bounded. The authors in [12] analyze new deployment strategies for satisfying some given coverage probability requirements with directional sensing models. A model of directed communications is introduced to ensure and repair the network connectivity. Based on a rotatable directional sensing model, the authors in [13] present a method to deterministically estimate the amount of directional nodes for a given coverage rate. A sensing connected sub-graph accompanied with a convex hull method is introduced to model a directional sensor network into several parts in a distributed manner. With adjustable sensing directions, the coverage algorithm tries to minimize the overlapping sensing area of directional sensors only with local topology information. Lastly, in [14], the authors present a distributed algorithm that ensures both coverage of the deployment area and

network connectivity, by providing multiple cover sets to manage Field of View redundancies and reduce objects disambiguation.

All the above algorithms depend on the geographical location information of sensor nodes. These algorithms aim to provide a complete-coverage network so that any point in the target area would be covered by at least one sensor node. However, this strategy is not as energy-efficient as what we expect because of the following two reasons. Firstly, the energy cost and system complexity involved in obtaining geometric information may compromise the effect of those algorithms. Secondly, sensor nodes located at the edge of the area of interest must be always in an active state as long as the region is required to be completely covered. These nodes will die after some time and their coverage area will be left without surveillance. Thus, the network coverage area will shrink gradually from outside to inside. This condition is unacceptable in surveillance applications and (intelligent) intrusion detection, because the major goal here is to detect intruders as they cross a border or as they penetrate a protected area.

In case of hostile environments, security plays an important role in the written of the scheduling program. Indeed an attacker, observing the manner nodes are waken up, should not be able to determine the scheduling program. For instance, in a video surveillance context, if the attacker is able to determine at some time the list of the sleeping nodes, then he can possibly achieve an intrusion without being detected [15].

Obviously, a random scheduling can solve the issues raised above, by guaranteeing a uniform coverage while preventing attackers to predict the list of awaken nodes. However, this approach needs random generators into nodes, which cannot be obtained by deterministic algorithms embedded into the network. Even if truly random generators (TRG) can be approximated by physical devices, they need a certain quantity of resources, suppose that the environment under observation has a sufficient variability of a given set of physical properties (to produce the physical noise source required in that TRG), and are less flexible or adaptable on demand than pseudorandom number generators (PRNGs). Furthermore, neither their randomness nor their security can be mathematically proven: these generators can be biased or wrongly designed.

Being able to guarantee a certain level of security in scheduling leads to the notion of *secure scheduling* proposed below.

4.2 Secure Scheduling in Wireless Sensor Networks

Two kinds of scheduling processes can be defined: each node can embed its own program, determining when it has to sleep (local approach), or the sink or some specific nodes can be responsible of the scheduling process, sending sleep or wake up orders to the nodes that have to change their states (global approach).

We consider that a deterministic scheduling algorithm is a function $\mathcal{S}$: $\{0,1\}^n \rightarrow \{0,1\}^M$, where $M > n$. This definition can be understood as follows:

• The value inputted in $\mathcal{S}$ is the secret key launching the scheduling process. It can be shown as the seed of a PRNG.

• In case of a local approach, the binary sequence produced by this function corresponds to the moments where the node must be awaken: if the *k*-th term of this sequence is 0, then the node can go to sleep mode between t_k and t_{k+1}.

• In case of a global approach, the binary sequence returned by $\mathcal{S}$ can be divided by blocs, such that each bloc contains the id of the node to which an order of state change will be send.

Loosely speaking, $\mathcal{S}$ is called a secure scheduling if it maps uniformly distributed input (the secret key or seed of the scheduling process) into an output which is computationally indistinguishable from uniform. The precise definition is given below.

Definition 8 A *T*-time algorithm $\mathcal{D} : \{0, 1\}^M \longrightarrow 0,1$ is said to be a (T, ε) −distinguisher for $\mathcal{S}$ if

$$\left| Pr[\mathcal{D}(S(\mathcal{U}_2^n)) = 1] - Pr[\mathcal{D}(\mathcal{U}_2^M) = 1] \right| \geq \varepsilon$$

where $\mathcal{U}_2$ is the uniform distribution on $\{0, 1\}$.

Definition 9 (Secure scheduling) Algorithm $\mathcal{S}$ is called a (T, ε) − secure scheduling if no (T, ε) −distinguisher exists for $\mathcal{S}$.

Adapting the proofs of [16, 17], it is possible to show that a (T, ε) −distinguisher exists if and only if a *T*-time algorithm can, knowing the first l bits of a scheduling *s*, predict the $(l + 1)$−st bit of *s* with probability significantly greater than 0.5. This comes from the fact that a PRNG passes the next-bit test if and only if it passes all polynomial-time statistical tests [16, 17].

An important question is what level of security (T, ε) suffices for practical applications in scheduled wireless sensor networks. Unfortunately, the level of security is often chosen arbitrarily. It is reasonable to require that a scheduling process is secure for all pairs (T, ε) such that the time-success ratio T/ε is below a certain bound. In the next section we present an illustration of this notion.

4.3 Practical Study

Suppose that a wireless sensor node has been scheduled by a Blum-Blum-Shub BBS pseudorandom generator. This generator produces bits $y_0, y_1, ...$, and the node is awaken during the time interval $[t_i; t_{i+1}[$ if and only if $y_i = 1$.

Let us recall that the Blum Blum Shum generator [18] (usually denoted by BBS) is defined by the following process:

1. Generate two large secret random and distinct primes p and q, each congruent to 3 modulo 4, and compute $N = pq$.

2. Select a random and secret seed $s \in [\![1, N - 1]\!]$ such that $gcd(s, N) = 1$, and compute $x_0 = s^2 (\mathrm{mod}\, N)$.

3. For $1 \leq i \leq l$ do the following:

(a) $x_i = x_{i-1}^2 (\text{mod } N)$.

(b) y_i = the least significant bit of x_i.

4. The output sequence is $y_1, y_1, \ldots, y_l$

Suppose now that the network will work during M = 100 time units, and that during this period, an attacker can realize 10^{12} clock cycles. We thus wonder whether, during the network's lifetime, the attacker can distinguish this sequence from truly random one, with a probability greater than ε = 0.2. We consider that N has 900 bits.

The scheduling process is the BBS generator, which is cryptographically secure. More precisely, it is (T, ε)-secure: no (T, ε)-distinguishing attack can be successfully realized on this PRNG, if [19]

$$T \leq \frac{L(N)}{6N\left(log_2(N)\right)\varepsilon^{-2}M^2} - 2^7 N\varepsilon^{-2}M^2 log_2(8N\varepsilon^{-1}M)$$

where M is the length of the output ($M = 100$ in our example), and

$$L(N) = 2.8 \times 10^{-3} exp\left(1.9229 \times (Nln(2)\ln(2)^{\frac{1}{3}}) \times \ln(N\ ln2)^{\frac{2}{3}}\right)$$

is the number of clock cycles to factor a $N-$bit integer.

A direct numerical application shows that this attacker cannot achieve its $(10^{12}, 0.2)$ distinguishing attack in that context.

4.4 Results study

This section presents simulation results on comparing our approach to the standard C++ rand()-based approach with random intrusions. We use the OMNET++ simulation environment and the next node selection will either use our approach or the C++ rand() function (rand() % 2^n) to produce a random number between 0 and 2^n. For these set of simulations, 128 sensor nodes are randomly deployed in a $75m * 75m$ area. First, we noted the percentage of active nodes, which is the average number of nodes involved in the active set over initial number of deployed sensors. This metric reflects, to a certain extent, the effectiveness of the proposed scheduling approach. The requirement of minimizing power consumption boils down to that of minimizing the number of working sensors. Figure 2 shows the percentage of active nodes. Both our approach and the standard rand() function have similar behavior: the percentage of active nodes progressively decreases due to battery shortage. From this result we can notice that these approaches improve the network lifetime.

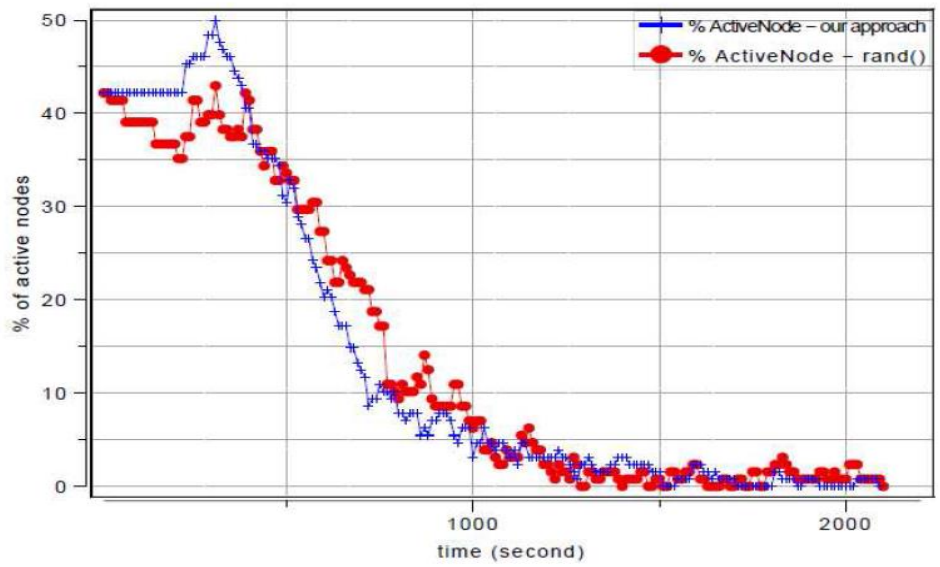

Figure 2: Percentage of active nodes

Another result we want to show is the energy consumption distribution. We recorded every 10s the energy level of each sensor node in the field and computed the mean and the standard deviation. Figure 3 shows the evolution of the standard deviation during the network lifetime. We can see that our approach selection provides a slightly better distribution of activity than the standard rand() function. This better distribution of the node's activity has the beneficial effect to increase the detection quality: as nodes are used more equally, there are less "holes" in the surveillance network due to some nodes having battery shortage earlier.

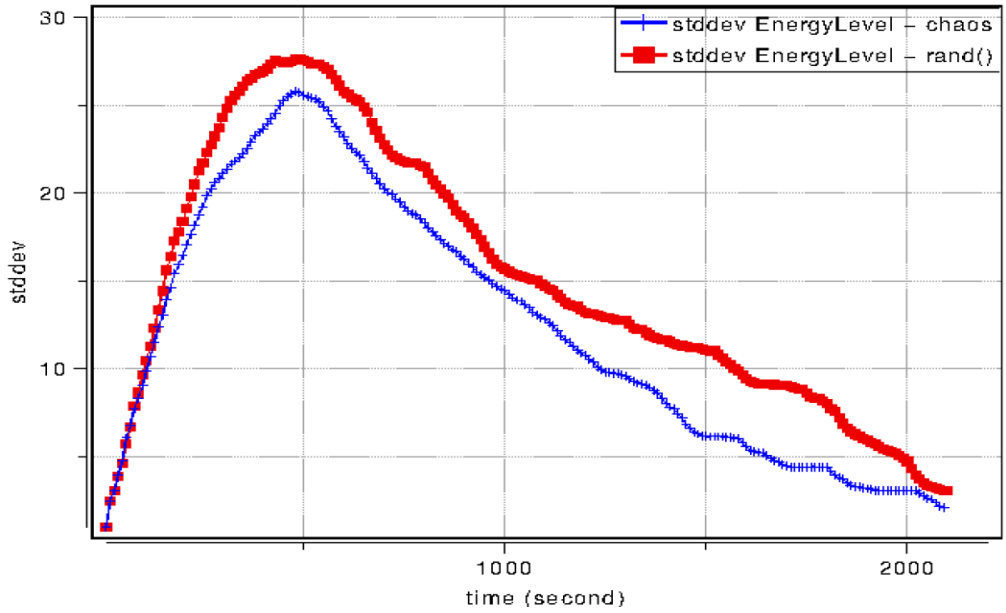

Figure 3: Evolution of the energy consumption's standard deviation.

5 Cryptographically Secure Data Aggregation in WSN

5.1 Theoretical Framework

To finalize the definition of a cryptographically secure wireless sensor, we need to introduce rigorously the notion of secure data aggregation in such networks.

Definition 10 (Aggregator) Let $\mathcal{S} = (\mathcal{T}, \mathcal{E}, inv)$ a public communication system on $(\mathcal{S}, \mathcal{M}, \{0,1\}^{\ell})$, and $c : \{0,1\}^{n} \rightarrow \{0,1\}^{m}$, $m < n$, a compression function.

A c-aggregator function on $\mathcal{S}$ is a couple of algorithms $Agg : \mathcal{S}^{p} \rightarrow \mathcal{S}$, $inv : \mathcal{K}^{p} \rightarrow \mathcal{K}$, with $p > 1$, such that Agg is a probabilistic algorithm, Agg and Inv can be computed polynomially, and $\forall\, s_1, \ldots, s_p \in \mathcal{S}$, $\forall m_1, \ldots, m_p \in \mathcal{M}$, $\forall k_1, \ldots, k_p \in \mathcal{K}$,

$$Agg(\mathcal{T}(s_1, m_1, k_1), \ldots, \mathcal{T}((s_p, m_p, k_p)) = s'$$

satisfies $\mathcal{E}(s', Inv(k_1, \ldots, k_p)) = c(m_1, \ldots, m_p)$.

The idea is that, as for hash functions, the security of the aggregator lays on the security of the compression function.

We now show how these materials for security can be applied in concrete wireless sensor networks.

5.2 Practical study of secure aggregation in WSN

5.2.1 The Model

Let us consider the tree-based wireless sensor network as shown in Fig. 4. When security properties are not taken into account, this protocol is the well-known tree-based aggregation model for WSN, which has already been well-studied several times in the literature (see, *e.g.*, [20, 21, 22, 23]). For the correctness of this protocol, reader is referred to [24, 25] for instance. Our goal is to prevent attackers from gaining any information about sensed data, while preserving the network lifetime. For the proposed context, suppose that scheduling is not possible, but that the intermediate nodes have the ability to make simple arithmetic operations of aggregation to reduce the global simulation cost. We suppose that this WSN is deployed in a hostile environment. Therefore, ensuring an end-to-end privacy between sensor nodes and the sink becomes problematic. This is largely because popular and existing cyphers are not additively homomorphic. In other words, the summation of encrypted data does not allow for the retrieval of the sum of the plain text values. Moreover, privacy existing homomorphisms have usually exponential bound in computation.

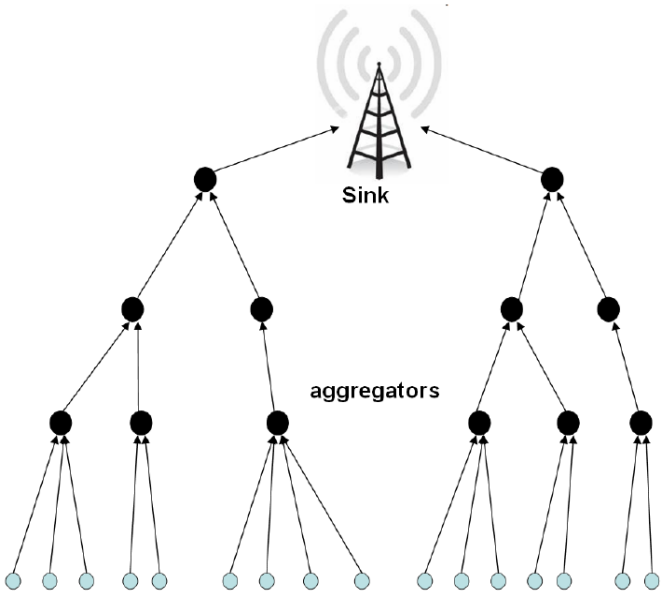

Figure 4: Tree-based data aggregation in sensor networks

5.2.2 Related Work

Some privacy homomorphism based researches have been proposed recently [26, 27, 28] that, without participating in checking, the aggregators can directly aggregate the encrypted data. The problem of aggregating encrypted data in sensor networks was introduced in [27] and further refined in [26]. The authors propose to use homomorphic encryption schemes to enable arithmetic operations over cypher-texts that need to be transmitted in a multi-hop manner. However, these approaches provide a higher level of system security, since nodes would not be equipped with private keys, which would limit the advantage gained by an attacker compromising some of the nodes. Unfortunately, existing privacy homomorphisms used for data aggregation in sensor networks have exponential bound in computation. For instance, Rivest Shamir Adleman (RSA) based cryptosystems [29, 30] are used, which require high CPU and memory capabilities to perform exponential operations. It is too computationally expensive to implement in sensor nodes. Moreover, the expansion in bit size during the transformation of plain text to cypher-text introduces costly communication overhead, which directly translates to a faster depletion of the sensors energy. On the other hand and from security viewpoint, the cryptosystems [31] used in these approaches were cryptanalyzed [32, 33], which means they cannot guarantee anymore high security levels. Some other methods are also reported for security in pervasive sensor networks [34, 35, 36, 37]. In [34], the authors proposed a pervasive, secure access scheme for a hierarchical-based architecture that closely matches that of MEDiSN. A QoS architecture to provide a level of quality assurance for applications in heterogeneous environments is discussed in [37]. The proposal implements a schedule-based approach that draws on information

about delay, loss and current network resources, and adjusts the scheduler to improve the video quality of delivery.

Finally, in SecureDAV [38], authors develop a key establishment protocol that generates a secret cluster key for each cluster. The share will be used to generate partial signatures on the readings, ensuring node authentication. Then a secure data aggregation and verification (SecureDAV) protocol is proposed, that ensures that the base station does not accept faulty readings. However, only one kind of aggregation is possible with such an approach.

In this paper we try to relax the statements above by investigating elliptic curve cryptography that allows feasible and suitable data aggregation in sensor networks beside the security of homomorphisms schemes. Elliptic Curve Cryptography has been the top choice among various public key cryptography options due to its fast computation, compact signatures, and small key size [39, 40, 41]. The authors in [41] propose digital signature algorithm that is based on the Elliptic Curve Digital Signature Algorithm to reduce energy consumption. In [42] a cooperative secure data aggregation in sensor networks is proposed which exploits Elliptic Curve Diffie-Hellman (ECDH) based security methods to achieve cooperative secure information integration. In [40], a concealed data aggregation scheme extended from Boneh et al.'s [43] homomorphic public encryption system is proposed. It is designed for a multi-application environment. The base station extracts application-specific data from aggregated ciphertexts. Despite the recent progress on elliptic curve cryptography for secure data aggregation in sensor networks, all the previous attempts have limitations. A common limitation of all these efforts is that all these attempts were developed as independent study without seriously considering the resource demands of sensor network aggregation and arithmetic operations.

First of all, our proposed scheme for secure data aggregation in sensor networks is based on a cryptosystem that has been proven safe and has not been cryptanalyzed. Indeed, it is known to be the sole secure and almost fully homomorphic cryptosystem usable now. Another property that enforces the security level of such an approach is coming from the fact that, as it is the case in ElGamal cryptosystem, for two identical messages it generates two different cryptograms. This property suggested fundamental for security in sensor networks [43, 44, 45], to the best of our knowledge, was not addressed in previous homomorphism-based security data aggregation researches. Beside all these properties and due to the use of elliptic curves, our approach saves energy by allowing nodes to encrypt and aggregate data without the need of high computations. Lastly, the scheme we use allows more aggregations types over cypher data than the homomorphic cryptosystems used until now.

5.2.3 The Elliptic Curves

Let us recall that [39] elliptic curves used in cryptography are typically defined over two types of finite fields: prime fields $\mathbb{F}_p$, where p is a large prime number, and binary extension fields $\mathbb{F}_{2^m}$ [46]. In our paper, as in [47, 48], we focus on elliptic curves over $\mathbb{F}_p$. Let $p > 3$, then an elliptic curve over $\mathbb{F}_p$ is defined by a cubic equation $y^2 = x^3 + ax + b$ as the set

$$\mathcal{E} = \left\{ (x, y) \in \mathbb{F}_p \times \mathbb{F}_p, y^2 \equiv x^3 + ax + b (\text{mod } p) \right\}$$

where $a, b \in \mathbb{F}_p$ are constants such that $4a^3 + 27b^2 \not\equiv 0 \,(\text{mod } p)$. An elliptic curve over $\mathbb{F}_p$ consists of the set of all pairs of affine coordinates (x, y) for $x, y \in \mathbb{F}_p$ that satisfy an equation of the above form and an infinity point $\mathcal{O}$.

The point addition and its special case, point doubling over $\mathcal{E}$ is defined as follows (the arithmetic operations are defined in $\mathbb{F}_p$) [39] :

Let $P = (x_1, y_1)$ and $Q = (x_2, y_2)$ be two points of $\mathcal{E}$. Then:

$$P + Q = \begin{cases} \mathcal{O} & if \quad x_2 = x_1 \text{ and } y_2 = -y_1, \\ (x_3, y_3) & \text{otherwise} \end{cases} \tag{1}$$

where:

- $x_3 = \lambda^2 - x_1 - x_2$,
- $y_3 = \lambda \times (x_1 - x_3) - y_1$,

$$\lambda = \begin{cases} (y_2 - y_1) \times (x_2 - x_1)^{-1} & if \quad P \neq Q, \\ (3x_1^2 + a) \times (2y_1)^{-1} & \text{if} \quad P = Q. \end{cases} \tag{2}$$

Finally, we define $P + \mathcal{O} = \mathcal{O} + P =, \forall P \in \mathcal{E}$, which leads to a commutative group $(\mathcal{E}, +)$. On the other hand the multiplication $n \times P$ means $P + P + \ldots + P$ n times and $-P$ is the symmetric of P for the group law $+$ defined above for all $P \in \mathcal{E}$.

5.2.4 Public/Private Keys Generation with ECC

In this section we recall how to generate public and private keys for encryption, following the cryptosystem proposed by Boneh *et al.* [43]. The analysis of the complexity will be treated in a later section.

Let $\tau > 0$ be an integer called "security parameter". To generate public and private keys, first of all, two τ-bits prime numbers must be computed. Therefore, a cryptographic pseudo-random generator can be used to obtain two vectors of τ bits, q_1 and q_2. Then, a Miller-Rabin test can be applied for testing the primality or not of q_1 and q_2. We denote by n the product of q_1 and q_2, $n = q_1 q_2$, and by l the smallest positive integer such that $p = l \times n - 1$. l is a prime number while $p = 2 \,(\text{mod } 3)$.

In order to find the private and public keys, we define a group H, which presents the points of the super-singular elliptic curve $y^2 = x^3 + 1$ defined over $\mathbb{F}_p$. It consists of $p + 1 = n \times l$ points, and thus has a subgroup of order n, we call it G. In another step, we compute g and u as two generators of G and $h =$

$q^2 \times u$. Then, following [43], the public key will be presented by (n, G, g, h) and the private key by q_1. Finally, it is supposed that each node embeds a cryptographically secure PRNG like the BBS one, which has been seeded randomly. Thus, at any time a random integer is required, a call to this primitive is realized (its cost is negligible compared to the other operations detailed in what follows).

5.2.5 Encryption and Decryption

After the private/public keys generation, we proceed now to the two encryption and decryption phases:

- **Encryption**: Assuming that our messages space consists of integers in the set $\{0, 1, \ldots, T\}$, where $T < q_2$, and m the (integer) message to encrypt. Firstly, a random positive integer is picked from the interval $[0, n - 1]$. Then, the cypher-text is defined by

  $$C = m \times g + r \times h \in G,$$

  in which $+$ and $\times$ refer to the addition and multiplication laws defined previously.

- **Decryption**: Once the message C arrived to destination, to decrypt it, we use the private key q_1 and the discrete logarithm of $(q_1 \times C)$ base $q_1 \times g$ as follows:

  $$m = log_{q1 \times g}\, q_1 \times C.$$

  This takes expected time $\sqrt{T}$ using Pollard's lambda method. Moreover, this decryption can be speed-up by precomputing a table of powers of $q_1 \times g$.

The encryption stage can be rewritten according to the notations introduced in Def 1. Let s be a carrier wave, m the message to secure (an integer), and (n, G, g, h) be the public key as defined previously. Then $\tau\ (s, m, (n, G, g, h)) = i_s(mg + rh)$, where r is a random integer as described above, and i_s is a frequency modulation function that conveys the $mg + rh$ information over the carrier wave s by varying its instantaneous frequency. More precisely, the information is here a point of the elliptic curve, that is, two integers lower than p, or two binary vectors of size lower than $log_2(p)$. So any frequency modulation technique for binary sequences can be used as i_s.

5.2.6 Homomorphic Properties

The proposed approach supports homomorphic properties, which gives us the ability to execute operations on values even though they have been encrypted. Indeed, it allows *N* additions and one multiplication directly on cryptograms, which prevents the decryption phase at the aggregators level and saves nodes energy, which is crucial in sensor networks.

Additions over cypher-texts are done as follows: let m_1 and m_2 be two messages and C_1, C_2 their cypher-texts respectively. Then the sum of C_1 and C_2, let's call it C, is represented by

$$C = C_1 + C_2 + r \times h,$$

where r is an integer randomly chosen in $[0, n-1]$ and $h = q_2 \times u$ as presented in the previous section. This sum operation guarantees that the decryption value of C is the sum $m_1 + m_2$. The addition operation can be done several times, which means we can do sums of encrypted sums.

The multiplication of two encrypted values and its decryption are done as follows: let e be the modified Weil pairing on the curve and g, h the points of G as defined previously. Let us recall that this modified Weil pairing e is obtained from the Weil pairing E [43], [49] by the formula: $e(P, Q) = E(x \times P, Q)$, where x is a root of $X^3 - 1$ on $\mathbb{F}_{P^2}$. Then, the result of the multiplication of two encrypted messages C_1, C_2 is given by

$$C_m = e(C_1, C_2) + r \times h_1,$$

where $h_1 = e(g, h)$ and r is a random integer pick in $[1, n]$.

The decryption of C_m is equal to the discrete logarithm of $q_1 \times C_m$ to the base $q_1 \times g_1$:

$$m_1 m_2 = log_{q_1 \times g_1} (q_1 \times C_m.)$$

where $g_1 = e(g, g)$.

5.2.7 Encryption for Sensor Networks

Our contribution compared to existing secure data aggregation In previous secure data aggregation protocols, security and data aggregation are often achieved together in a hop-by-hop manner. That is, data aggregators must decrypt every message they receive, aggregate the messages according to the corresponding aggregation function, and encrypt the aggregation result before forwarding it. Therefore, these techniques cannot provide data confidentiality at data aggregators and result in latency because of the decryption/encryption process.

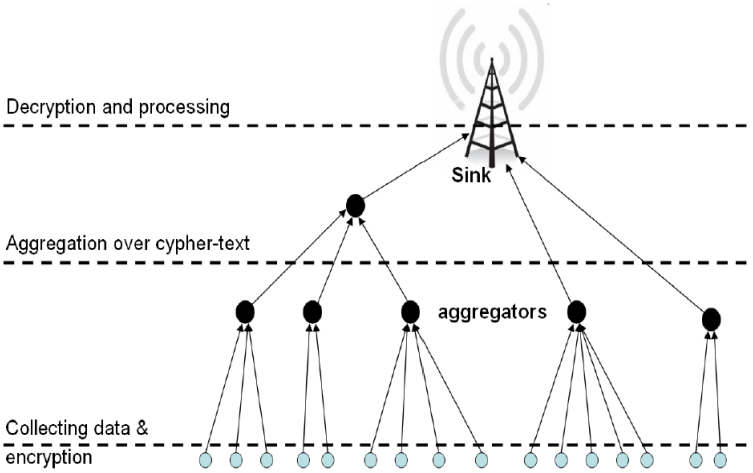

Figure 5: Secure data aggregation in sensor networks

In our previous work, we have proposed an encryption protocol that performs data aggregation without requiring the decryption of the sensor data at data aggregators. We adopt the following scenario as shown in Figure 5: after collecting information, each sensor node encrypts its data according to elliptic curve encryption (*c.f.* Section 5.2.3) and sends it to the nearest aggregator. Then, aggregators aggregate the received encrypted data (without decryption) and send it to the base station, which in its turn decrypts the data and aggregates it. We notice that all aggregators can do *N* additions and the final layer of aggregators can do one multiplication on encrypted data.

Secure Aggregation: Computing the Arithmetic Mean To compute the average of nodes measurements, aggregators can calculate the sum of the encrypted measurements and the number of nodes take these measurements and send it to the base station. More precisely, when using our scheme, each sensor encrypts its data x_i to obtain cx_i. The sensor then forwards cx_i to its parent, who aggregates all the cx_j's of its k children by simply adding them up. The resulting value and the encryption of k are then forwarded. The sink can thus compute the average value with all of these data. A same approach can be followed to compute the variance or the weighted mean.

According to notations introduced in Def. 10, the Agg function is defined by

$$Agg(C_1, C_2, \ldots, C_n) = (n, C_1 + C_2 + \ldots + C_n + rh) \in G,$$

and so $c(C_1, \ldots, C_n) = C_1 + \ldots + C_n + rh$, with r a random integer.

5.2.8 Security study

Due to hostile environments and unique characteristics of sensor networks, it is a challenging task to protect sensitive information transmitted by nodes to the end user. In addition, this type of networks has security problems that traditional

networks do not face. In this section, we outline a security study dedicated to wireless sensor networks.

Table 2: Encryption policies and vulnerabilities

Encryption policy	**Possible attacks**
Sensors transmit readings without encryption	Man-in-the middle
Sensors transmit encrypted readings with permanent keys	Known-plain text attack Chosen-plain text attack Man-in-the-middle
Sensors transmit encrypted readings with dynamic keys	None of above
Our scheme	None of above

In a sensor network environment adversaries can commonly use the following attacks:

Table 3: Encryption policies and aggregation

Encryption policy	**Data aggregation**
Sensors transmit readings without encryption	Generating wrong aggregated results
Sensors transmit encrypted readings with permanent keys	Data aggregation is impossible, unless the aggregator has encryption keys
Sensors transmit encrypted readings with dynamic keys	Data aggregation cannot be achieved, unless the aggregator has encryption keys
Our scheme	Data aggregation can be achieved

Known-plain text attack: They can use common key encryption to see when two readings are identical. By using nearby sensors under control, attackers can conduct a known-plain text attack.

Chosen-plain text attack: Attackers can tamper with sensors to force them to predetermined values.

Man-in-the-middle: They can inject false readings or resend logged readings from legitimate sensor motes to manipulate the data aggregation process.

In Tables 2, 3 and similar to [46], we present a comparison between different encryption policies and possible attacks. In our method, as data are encrypted by public keys, and these public keys are sent by the sink to the sole authenticated motes, the wireless sensor network is then not vulnerable to a Man-in-the-middle attacks. On the other hand, our approach guarantees that for two similar texts gives two different cryptograms, which prevents the Chosen-plain text attacks and the Man-in-the-middle attacks. Finally, as the proposed scheme possesses the

homomorphic property, data aggregation is done without decryption, and no private key is used in the network.

5.3 Experimental Results

5.3.1 Evaluation of the homomorphic approach

In this section we present some practical issues to our data encryption model. Firstly, we study the sizes of the encryption keys and we compare it to existing approaches. Then, we show how we can optimize the sizes of cryptograms in order to save more sensors energy.

Sizes of the Keys Cryptograms are points of the elliptic curve $\mathcal{E}$. They are constituted by couples of integer coordinates lesser than or equal to $p = lq_1q_2 - 1$.

It is commonly accepted [50], [51] that for being secure until 2020, a cryptosystem:

- must have $p \approx 2^{161}$, for EC systems over $\mathbb{F}_p$,
- must satisfy $p \approx 2^{1881}$ for classical asymmetric systems, such as RSA or ElGamal on $\mathbb{F}_p$.

Thus, for the same level of security, using elliptic curve cryptography does not demand high keys sizes, contrary to the case of RSA or ElGamal on $\mathbb{F}_p$. The use of small keys leads to small cryptograms and fast operations for encryption.

Reducing the Size of Cryptograms In this section we show how we can reduce the size of cryptograms while using ECC. This is benefit for sensor nodes in terms of reducing energy consumption by sending data with smaller size. The messages are encrypted with q_2 bits, which leads to cryptograms with a mean of 160 bits long.

Let us suppose that $p \equiv 3 \pmod 4$. As the cryptogram is an element (x, y) of $\mathcal{E}$, which is defined by $y^2 = x^3 + 1$, we can compress this cryptogram (x, y) to $(x, y \bmod 2))$ before sending it to the aggregator (as the value of y^2 is known). In this situation, we obtain cryptograms with a mean of 81 bits long for messages between 20 and 40 bits long.

To decompress the cryptogram (x, i), the aggregator must compute $z = x^3 + 1 \bmod p$ and $y = \sqrt{z} \bmod p$, which can be written as $y = z^{(p+1)/4} \bmod p$, then:

- if $y \equiv i(mod\ 2)$, then the decompression of (x, i) is (x, y).
- else the decompression point is $(x, p - y)$.

We compared our approach to the well-known RSA cryptosystem, concerning the average energy consumption of an aggregating wireless sensor network. In this comparison, we considered a network formed of 500 sensor nodes, each on is equipped by a battery of 100 units capacity. We consider that the energy consumption *"E"* of a node is proportional to the computational time *t, i.e.*, $E = kt$. The same coefficient of proportionality k is taken while comparing the two

encryption scenarios. Sensor nodes are then connected to 50 aggregators chosen randomly. Each sensor node chooses the nearest aggregator. The running of each simulation is as follows: each sensor node takes a random value, encrypts it using one of the encryption methods then sends it to its aggregator. Aggregators compute the sum of the encrypted received data and send it to the sink.

Figure 6 gives the comparison between RSA and elliptic curve based encryption. We can notice that our approach saves the energy largely greater than the case of RSA, where its depletion is so fast (time in seconds). Finally let us notice that, in addition of reducing the amount of energy units (of the battery) needed for encryption and aggregation, the sink receives many more values per second in EC-based networks than in RSA-based one.

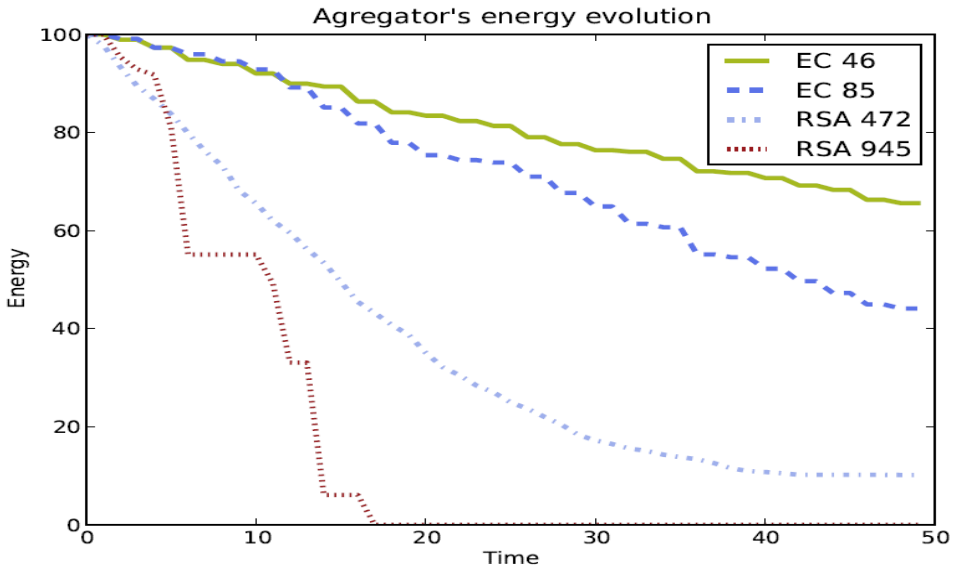

Figure 6: Comparison of energy consumption

6 Conclusion

In this document, a rigorous framework for security in wireless sensor networks has been formalized. The definition of a communication system in WSNs has been introduced, and security properties (indistinguability, nonmalleability, and message detection resistance) have been formalized in that context. Furthermore, the definitions of secure scheduling and secure aggregation, specific to such networks, have been given too. With this theoretical framework, it has been possible to evaluate the security of a scheduling scheme based on the BBS cryptographically secure PRNG, and of a secure communication protocol on WSN compatible with aggregation. This proposed scheme helps sensor networks beside the security saving energy. It permits the generation of shorter encryption asymmetric keys, which is so important in the case of sensor networks. Moreover, it allows nodes to encrypt and aggregate data without the need of high

computations. Then, the proposed approach provides a scheduling method preventing all the nodes to be active at the same time. It defines a subset of the deployed nodes to be active while the other nodes can sleep. Thus, nodes in sleep mode reduce their energy consumption. The experimental results show that our method significantly reduces computation and communication overhead compared to other works, and can be practically implemented in on-the-shelf sensor platforms.

Although this paper presents a complete security framework, in the future work we plan to make a classification of attacks that the adversary can achieve depending on the data he has access to. Our desire is to distinguish between several levels of security into each category of malicious attacks, from the weakest one to the strongest one. Then, we intend to enlarge the number of security properties that can be established for a WSN. Existing scheduling, aggregation, or communication protocols on wireless sensor networks will have their security evaluated. The scheduling process given in example will be integrated to our proposed network architecture based on elliptic curve, and the resulted network will be completely evaluated, both for security and energy consumption. Finally, new original solutions being both efficient and secure will be investigated thanks to the theoretical framework introduced here.

Disclosure Policy: The authors declare that there is no conflict of interests regarding the publication of this article.

References

[1] Giuseppe Anastasi, Marco Conti, Mario Di Francesco, and Andrea Passarella. Energy conservation in wireless sensor networks: A survey. *Ad Hoc Netw.*, 7(3):537–568, May 2009.

[2] M.Cardei and J.Wu. Energy-efficient coverage problems in wireless ad-hoc sensor networks. *Computer Communications*, 29(4), 2006.

[3] J.Bahi, A.Makhoul, and A.Mostefaoui. Localization and coverage for high density sensor networks. *Computer Communications journal*, 31(4):770–781, 2008.

[4] Hai Liu, Pengjun Wan, and Xiaohua Jia. Maximal lifetime scheduling for sensor surveillance systems with k sensors to one target. *IEEE Transactions on Parallel and Distributed Systems*, 17(12):1526–1536, 2006.

[5] T. Zhao and Q. Zhao. Coverage based information retrieval for lifetime maximization in sensor networks. *In the 41st Conference on Information Sciences and Systems*, 2007.

[6] X. Li. A survey on data aggregation in wireless sensor networks. *Project Report for CMPT 765*, 2006.

[7] Jacques Bahi, Abdallah Makhoul, and Maguy Medlej. Data aggregation for periodic sensor networks using sets similarity functions. *IWCMC 2011, 7th IEEE Int. Wireless Communications and Mobile Computing Conference*, pages 559–564, July 2011.

[8] Mehdi Esnaashari and M. R. Meybodi. Data aggregation in sensor networks using learning automata. *Wireless Networks*, 16(3):687–699, 2010.

[9] Jacques Bahi, Abdallah Makhoul, and Maguy Medlej. A two tiers data aggregation scheme for periodic sensor networks. *AdHoc & Sensor Wireless Networks*. Accepted manuscript. To appear.

[10] Congduc Pham, Abdallah Makhoul, and Rachid Saadi. Risk-based adaptive scheduling in randomly deployed video sensor networks for critical surveillance applications. *Journal of Network and Computer Applications*, 34(2):783–795, 2011.

[11] Abdallah Makhoul and Congduc Pham. Dynamic scheduling of cover sets in randomly deployed wireless video sensor networks for surveillance applications. *2nd IFIP Wireless Days Conference, WD'09*, pages 73–78, December 2009.

[12] Huadong Ma and Yonghe Liu. Some problems of directional sensor networks. *International Journal of Sensor Networks*, 2(1-2):44–52, 2007.

[13] Dan Tao, HuadongMa, and Liang Liu. Coverage-enhancing algorithm for directional sensor networks. *Lecture Notes in Computer Science - Springer*, pages 256–267, November 2006.

[14] Congduc Pham and Abdallah Makhoul. Performance study of multiple cover-set strategies for mission-critical video surveillance with wireless video sensors. In *6th IEEE Int. Conf. on Wireless and Mobile Computing, Networking and Communications, wimob'10*, pages 208–216, 2010.

[15] Jacques Bahi, Christophe Guyeux, Abdallah Makhoul, and Congduc Pham. Secure scheduling of wireless video sensor nodes for surveillance applications. In *ADHOCNETS 11, 3rd Int. ICST Conference on Ad Hoc Net- works*, volume 89 of *LNICST*, pages 1–15, Paris, France, September 2011. Springer.

[16] Andrew C. Yao. Theory and application of trapdoor functions. In *Proceedings of the 23rd Annual Symposium on Foundations of Computer Science*, SFCS '82, pages 80–91, Washington, DC, USA, 1982. IEEE Computer Society.

[17] Oded Goldreich, Shafi Goldwasser, and Silvio Micali. How to construct random functions. *J. ACM*, 33:792–807, August 1986.

[18] Manuel Blum and Shafi Goldwasser. An efficient probabilistic public key encryption scheme which hides all partial information. In *Proceedings of CRYPTO 84 on Advances in cryptology*, pages 289–302, New York, NY, USA, 1985. Springer-Verlag New York, Inc.

[19] R. Fischlin and C. P. Schnorr. Stronger security proofs for rsa and rabin bits. In *Proceedings of the 16th annual international conference on Theory and*

application of cryptographic techniques, EUROCRYPT'97, pages 267–279, Berlin, Heidelberg, 1997. Springer-Verlag.

[20] Miloud Bagaa, Noureddine Lasla, Abdelraouf Ouadjaout, and Yacine Challal. Sedan: Secure and efficient protocol for data aggregation in wireless sensor networks. In *Proceedings of the 32nd IEEE Conference on Local Computer Networks*, LCN '07, pages 1053–1060, Washington, DC, USA, 2007. IEEE Computer Society.

[21] Claude Castelluccia. Efficient aggregation of encrypted data in wireless sensor networks. In *MobiQuitous*, pages 109–117. IEEE Computer Society, 2005.

[22] Bhaskar Krishnamachari, Deborah Estrin, and Stephen B. Wicker. The impact of data aggregation in wireless sensor networks. In *Proceedings of the 22nd International Conference on Distributed Computing Systems*, ICDCSW '02, pages 575–578, Washington, DC, USA, 2002. IEEE Computer Society.

[23] J. Girao,M. Schneider, and D.Westhoff. Cda: Concealed data aggregation in wireless sensor networks. In *Proceedings of the ACMWorkshop onWireless Security*, 2004.

[24] Hani Alzaid, Ernest Foo, and Juan Gonzalez Nieto. Secure data aggregation in wireless sensor network: a survey. In *Proceedings of the sixth Australasian conference on Information security - Volume 81*, AISC '08, pages 93–105, Darlinghurst, Australia, Australia, 2008. Australian Computer Society, Inc.

[25] *Secure Data Aggregation in Wireless Sensor Networks: A Survey*, 2006.

[26] C. Castelluccia, E. Mykletun, and G. Tsudik. Efficient aggregation of encrypted data in wireless sensor networks. *Proc. of the 2nd Annual MobiQ-uitous*, pages 119–117, 2005.

[27] Joao Girao, Markus Schneider, and Dirk Westhoff. Cda: Concealed data aggregation in wireless sensor networks. October 2004. Poster presentation.

[28] M. Acharya, J.Girao, and D. Westhoff. Secure comparison of encrypted data in wireless sensor networks. *Third international symposium WiOpt'05*, pages 47–53, 2005.

[29] W. Haodong, S. Bo, and L. Qun. Elliptic curve cryptography-based access control in sensor networks. *International Journal of Security and Networks*, 1(3-4):127–137, 2006.

[30] A. Liu and P. Ning. Tinyecc: A configurable library for elliptic curve cryptography in wireless sensor networks. *Proceedings of IPSN'08*, pages 245–256, 2008.

[31] J. Domingo-Ferrer. A provably secure additive and multiplicative privacy homomorphism. *6th ISC conference*, pages 471–483, 2003.

[32] J. Cheon, W.-H. Kim, and H. Nam. Known-plaintext cryptanalysis of the domingo ferrer algebraic privacy homomorphism scheme. *Inf. Processing*

Letters, 97(3):118–123, 2006.

[33] D.Wagner. Cryptanalysis of an algebraic privacy homomorphism. *6th ISC conference*, 2851, 2003.

[34] Y. M. Huang, M. Y. Hsieh, H. C. Chao, S. H. Hung, and J. H. Park. Pervasive, secure access to a hierarchical sensor-based healthcare monitoring architecture in wireless heterogeneous networks. *IEEE J.Sel. A. Commun.*, 27(4):400–411, May 2009.

[35] Yue-Hsun Lin, Shih-Ying Chang, and Hung-Min Sun. Cdama: Concealed data aggregation scheme for multiple applications in wireless sensor networks. *IEEE Transactions on Knowledge and Data Engineering*, 99(PrePrints), 2012.

[36] Liang Zhou and Han-Chieh Chao. Multimedia traffic security architecture for the internet of things. *IEEE Network*, pages 35–40, 2011.

[37] LiangZhou, Han-ChiehChao, andAthanasiosV.Vasilakos. Joint forensics-scheduling strategy for delay-sensitive multimedia applications over heterogeneous networks. *IEEE Journal on Selected Areas in Communications* pages 1358–1367, 2011.

[38] Ajay Mahimkar. Securedav: A secure data aggregation and verification protocol for sensor networks. In *In Proceedings of the IEEE Global Telecommunications Conference*, pages 2175–2179, 2004.

[39] D. Hankerson, A. Menezes, and S. Vanstone. Guide to elliptic curve cryptography. *Springer*, 2004.

[40] Y. Lin, S. Chang, and H. Sun. Cdama: Concealed data aggregation scheme for multiple applications in wireless sensor networks. *IEEE Transactions on Knowledge and Data Engineering*, PP(99):1, 2012.

[41] V. Bhoopathy and R.M.S. Parvathi. Cdama: Concealed data aggregation scheme for multiple applications in wireless sensor networks. *American Journal of Applied Sciences*, 9(6):858–864, 2012.

[42] Hua-Yi Lin and Tzu-Chiang Chiang. Cooperative secure data aggregation in sensor networks using elliptic curve based cryptosystems. *Lecture Notes in Computer Science*, 5738:384–387, 2009.

[43] D. Boneh, E.-J. Goh, and K. Nissim. Evaluating 2-dnf formulas on ciphertexts. *Theory of Cryptography, LNCS*, pages 325–341, 2005.

[44] Shih-I Huang, Shiuhpyng Shieh, and J. D. Tygar. Secure encrypted-data aggregation for wireless sensor networks. *Wirel. Netw.*, 16:915–927, May 2010.

[45] Hua-Yi Lin and Tzu-Chiang Chiang. Cooperative secure data aggregation in sensor networks using elliptic curve based cryptosystems. In *Proceedings of the 6th international conference on Cooperative design, visualization, and engineering*, CDVE'09, pages 384–387, Berlin, Heidelberg, 2009. Springer-Verlag.

[46] R.C.C. Cheung, N.J. Telle, W. Luk, and P.Y.K. Cheung. Secure encrypteddata aggregation for wireless sensor networks. *IEEE Trans. on Very Large Scale Integration Systems*, 13(9):1048–1059, 2005.

[47] Jacques Bahi, Christophe Guyeux, and Abdallah Makhoul. Secure data aggregation in wireless sensor networks. homomorphism versus watermarking approach. In *ADHOCNETS 2010, 2nd Int. Conf. on Ad Hoc Networks*, volume 49 of *Lecture Notes in ICST*, pages 344–358, Victoria, Canada, August 2010.

[48] Jacques Bahi, Christophe Guyeux, and Abdallah Makhoul. Two security layers for hierarchical data aggregation in sensor networks. *IJAACS*, 7(3): 239-270 (2014)

[49] D. Boneh and M. Franklin. Identity-based encryption from the weil pairing. *Crypto'2001, LNCS*, 2139:213–229, 2001.

[50] E. Barker and A. Roginsky. Draft nist special publication 800-131 recommendation for the transitioning of cryptographic algorithms and key sizes. 2010.

[51] A.K. Lenstra and E.R. Verheul. Selecting cryptographic key sizes. *Jour. Of the International Association for Cryptologic Research*, 14(4):255–293, 2001.