Data Security in Unattended Wireless Sensor Networks through Aggregate Signcryption

Faezeh Sadat Babamir and Ziba Eslami
Shahid Beheshti university of Tehran, G. C.
Tehran - Iran
[e-mail: f.babamir@mail.sbu.ac.ir, z_eslami@sbu.ac.ir]
*Corresponding author: Faezeh Sadat Babamir

Received June 7, 2012; revised September 24, 2012; revised October 27, 2012; accepted October 30, 2012;published November 30, 2012

Abstract

In this paper, we propose aggregate signcryption for achieving data security in UWSNs. The main challenge of these networks established in sensitive environments is offline sink visiting. Moreover, the sensors must retain collected data for long enough time to offload them onto the itinerant sink. Thus, the unattended nature of data collection intervals might offer the adversary the opportunity to apply various attacks without detection. In this paper, employing low order operations (in time and space), we propose a new secure scheme in which various security goals such as confidentiality (through encrypting), authentication and integrity (through signing) are achieved. In addition, the aggregation process of our scheme reduces the space and communication overheads both for sensors and sink, i.e. the proposed technique efficiently enables the sensors and sink to protect, verify and recover all the related data. We further compare our scheme with the best alternative work in the literature.

Keywords: Aggregate signcryption, data security, confidentiality, authentication, integrity

This work is supported by research institute for Information Communication Technology (ICT) (No. ت/6981/500. The authors would like to thank the anonymous reviewers for their valuable comments which improved the quality of this paper and clarified many important points to the reader.

http://dx.doi.org/10.3837/tiis.2012.10.011

1. Introduction

In recent years, sensor and sensor networks have been extremely popular in the research community. Wireless Sensor Networks (WSNs) deployed in the hostile environment are referred to as disconnected or Unattended WSNs (UWSNs). In these networks, one of the fundamental issues is how to securely collect and maintain sensitive data. More exactly, a trusted entity or sink can not be always present and collect data online. It roams around the sensing region and collects data at the end of each collection interval. This periodic visiting of network causes sink to collect data in non-real time or disconnected manner. On the other hands, considering the constrained sensor resources, the security issue becomes very challenging. Especially, due to high computation cost of conventional intensive cryptographic primitives, complicated cryptosystems can not be utilized to guarantee the system security.

The expensive computation and data sending will exhaust sender nodes' energy quickly. To enhance the total performance, we should resolve the following problems. (1) Reducing broadcast energy consumption, (2) Reducing the amount of node information and (3) Reducing the computational overheads on sender nodes. Data privacy, integrity and source authenticity are the main tasks to strengthen information systems. To achieve confidentiality and authenticity simultaneously, encryption and signature are often combined in sequence. This traditional method is infeasible due to heavy overheads and lack of security. In 2000, Zheng proposed a novel concept named *signcryption* to perform the encryption and signature in a single primitive [1] to fix the above problems. Recently, Elliptic Curve (EC) based signcryption scheme is a new technique to fulfil both the functions of secure encryption and digital signature with a significant smaller cost.

Unattended WSNs have diverse applications : monitoring potential nuclear activity, detecting underground sound and vibration in order to be aware of troop movement (or border crossings) and detecting enemy aircrafts for air-bone sensor network tracking fluctuation in air turbulence and pressure. Trident systems [2] provide reliable communication links. There are often used for transmitting timely messages back to command and control centers. These sensors can be used in battlefield applications including perimeter defense, border patrol and surveillance, target acquisition and situation awareness. Another well-known project deployed by U.S. Defense Advanced Research Project Agency (DARPA) is the so-called LANdroids [3]: smart robotic radio relay nodes for battlefield deployment. LANdroid is an ad-hoc network and provides connectivity as well as valuable information for soldiers. LANdroids might retain valuable information for long time, until soldiers move close to the network. In the interval, the adversary might attempt to delete or modify that information, without disrupting network operations, while he remains undetected.

Contribution: To the best of our knowledge, this paper is the first to deal with the problem of data security in UWSNs using the *aggregate signcryption technique*. This new, effective, and efficient countermeasure achieves confidentiality, integrity, and authentication of data. Furthermore, by using the aggregation concept, communication and memory overheads are significantly reduced. We use unknown receiver identification to hide the nature of the receiver. Finally, our research opens up new directions and identifies challenges in the context of UWSN security.

Organization: Section 2 gives an overview of the related work and Section 3 covers preliminary materials. In Section 4, we provide description of the system assumptions and the proposed scheme. Section 5 is devoted to proof of system security and Section 6 covers performance analysis. Finally, conclusions are made in Section 7.

2. Related Work

Di Pietro et al. in [4] studied super-encryption and re-encryption techniques to defend mobile adversary. They do not evaluate the cost of time, memory and energy consumption. In addition, due to dependency on symmetric encryption, their proposed solution has some limitations. Symmetric setting prevents sensors from using data aggregation techniques. Another solution is asymmetric based scheme. Although it is resource consuming compared to symmetric solution, the sensors can decrypt the ciphertexts, aggregate data, eliminate redundancy to minimize memory and communication overheads. Consequently, extra efficiency through data aggregation is obtained at the cost of energy and memory consumptions. Data aggregation is more considered than energy and memory consumptions, since 1 bit transmitted may require the power equivalent to execute 800-1000 instructions [5].

D. Ma et al. in [6] proposed 2 approaches: First, FssAgg-BLS (a kind of signature) as ideal cryptographic tools for achieving data integrity and authentication for UWSNs. Second approach is FssAgg-Mac based on symmetric key cryptography, hash chains and Message Authentication Codes (MACs) that requires full symmetric key distribution and does not allow to be public verifiable. This makes it unscable and impractical for large distributed UWSNs. Later D. Ma et al. developed FssAgg-AR and FssAgg-BM in [7][8] that are more computational and storage efficient than FssAgg-BLS. However, all these schemes are still not efficient enough for UWSNs and are effective only against the adversary that is relatively weak and easy to overcome.

R.D. Pietro et al. in [9] proposed two collaborative authentication schemes, CoMac and ExCo to defend against the strong adversary. However, some flaws reported in these schemes which make them vulnerable to attacks such as Path-Based DoS (PDoS) [10] and False-Endorsement DoS (FEDoS) [11] attacks. It is claimed in [9] that ExCo is stronger than CoMAC in terms of sensor compromising but C. M. Yu et al. in [12] have proved that their resilience against sensor compromising is actually the same in practice. To address these problems, Yu et al. have proposed Acquire Authentication Data (AAD) which has three characteristics: 1- there is acceptable communication-efficiency, due to the proper use of sensors' position information, 2- in addition to acquiring authentic data, AAD is resilient against both PDoS and FEDoS attacks and 3- the resilience of AAD against sensor compromising is superior to [9].Generally, ADD is a superior scheme in terms of resilience and communication overhead for now but it heavily relies on the invariant position information of each sensor. Hence it has weakness in the application of mobile sensors. Moreover, according to the authors, ADD is robust against PDoS and FEDoS attacks but many attacks such as radio jamming attack should be considered in the improvement of ADD scheme. Different attacks in WSNs are studied in SMTR [13]. This model achieves authentication and data integrity using either Message Authentication Code (MAC) and Digital Signature (DS) techniques. However SMTR is costly to apply in UWSNs.

3. Preliminaries

In this section, we briefly cover necessary background such as elliptic curve, computational assumptions and id based system models along with its security model.

3.1 Elliptic Curve Cryptography

Elliptic curves are algebraic structures first mentioned in [14]. The properties of the elliptic curve cryptography (ECC) allow the setup of a problem to operate similarly to the well-known discrete logarithm problem of finite (Galois) fields. An elliptic curve *E* is a set of points over a field that satisfies a certain equation. Curve *E* is defined over the finite binary field GF_t and all of its points *(x, y)* with $x, y \in GF_t$ satisfying the so called Weierstraß equation. Here a_i are the parameters of the curve.

$$y^2 + a_1xy + a_3y \equiv x^3 + a_2x^2 + a_4x + a_6, s.t. a_1, a_2, a_3, a_4, a_6 \in GF_t$$

3.2 Elliptic Curve Discrete Logarithm Problem (EC-DLP)

Suppose there is a curve *E* over a finite binary field GF_t, with two points $P \in E$ and $Q \in E$.The problem is to find a $k \in N$ such that $Q = kP$ holds. A good starting point in cryptography is to consider what the minimum *secure* key size is; that is, what secure key size cannot feasibly be broken by modern hardware. There are a lot of theoretical discussions on this topic [15]. The largest Elliptic Curve Discrete Logarithm Problem (EC-DLP) to be solved so far had a key size of 109 bits, that is, over the finite field GF_{109} it took 17 months to break [14]. Although 17 months with huge computing power (2600 workstations) is a lot of time, the security of the 109-bit key size is now debatable for today's hardware. To ensure higher security for critical data, one can select even larger keys in sensor networks. The next possible highly memory-saving key size is 113bits, i.e., a curve over GF_{113}. This curve offers about $2^4 = 16$ times more security than 109 bits. Blab et al. in [16] proposed 113-bit ECC keys to offer high enough security with the smallest possible memory consumption in the sensor network.

3.3 Related Computational Assumptions

In this section, we review the assumptions of the computational diffie-hellman and elliptic curve discerete logarithm problems,which are relevant to the protocol.

3.3.1 Computational Diffie Hellman Problem (CDHP) [17]

Let G be an additive cyclic group with prime order *q* and generator *g*. Given $(g, ag, bg) \in G^3$ for unknown $a, b \in Z_q^*$, The CDHP in G is to compute *abg*.

3.3.2 Elliptic Curve-Discrete Logarithm Problem (EC-DLP) [17]

Let G be an additive cyclic group with prime order *q* and generator *g*. Given $xg \in G$ for unknown $x \in Z_q^*$, the EC-DLP in G is to find *x*.

3.4 Identity Based Aggregate Signcryption Model

In this section, we define the general model for an identity-based aggregate signcryption scheme [18][19]. Using our scheme, every sensor with identity *ID* signcrypts every message and finally aggregates all signcryptions to make one packet to send. The scheme consists of the following algorithms:

Setup(*d*): Given a security parameter *d*, the Private Key Generator (PKG) generates the public parameters *params* and Master Secret Key (*MSK*) of the system using this algorithm.

KeyGen(*ID*): The PKG inputs the public parameters *params*, the *MSK x* and identity *ID*, and then computes the partial private key *(R, s)* corresponding to *ID*.

Signcrypt(*m_i, ID, (R, s), int*): A sender with identity *ID* and partial private key *(R, s)* runs this algorithm to signcrypt a message $m_i \in M$ (M is message space) of round *i* at interval *int*. This algorithm outputs the signcryption σ_i corresponding to *ID*.

Aggregate signcrypt($\sigma_i \forall i$, *ID*): After collecting all σ_i corresponding to *ID*, the sensor *ID* runs this algorithm to create an aggregate signcryption σ_{agg} to send to the receiver.

Aggregate-unsigncrypt(σ_{agg} *, ID, (R, s), int)*: Upon receiving the aggregate signcryption σ_{agg} of identity *ID* with partial private key *(R, s)* and interval number *int*, the receiver runs this algorithm to obtain either the plain text $m_i \forall i$ or the error symbol $'\varepsilon'$ according to whether σ_i was a valid signcryption for identity *ID* or not. For consistency, we require that if $\sigma = Signcrypt(m, ID, (R, s))$, then $m = Unsigncrypt(\sigma, ID, (R, s))$.

3.5 Formal Security Models for Identity Based Aggregate Signcryption

Since our problem is different from the usual ones in the signcryption field, we state the proof of system with a slight difference. This difference refers to the new operation of a user considered as a node. Let **C** be a challenger who utilizes adversary **A** to break some hard problems. In this case, **C** trains the adversary **A** with a number of nodes such that every node signcrypts many messages. More precisely, each node gathers *$data_i$* and simultaneously signcrypts them as long as the station appears. At the transmission time, each node aggregate-signcrypts $m_i \forall i$ σ_{agg} (the aggregation of signcryptions of $m_i \forall i$) *by itself* to send. Unlike usual proofs in which every user with a unique partial key signcrypts *one* message, in our proof each node aggregate-signcrypts *a number of messages* $m_i \forall i$ by generating a special key k_i for every message. Thus we consider variables $k_1,\ldots,k_n$ for $m_1, m_2,\ldots,m_n$. Considering these assumptions, the two security properties that are desired from any ID-based aggregate signcryption scheme, called *message conffidentiality* and *unforgeability*,will be disscused. The following are the security models for our scheme.

3.5.1 Confidentiality

Below, the adversary **A** wants to attack node *ID**. The main challenge to confidentiality is security of σ_j signcrypted by *ID**.

Definition: An ID-based aggregate signcryption scheme is said to be secure against adaptive chosen ciphertext attacks (IND-IBAS_CCA) [18], if no probabilistic polynomial time adversary **A** has a non-negligible advantage in the following game where the advantage of **A** is defined as:

$$Adv_A = [\prod_{i=1}^{n} \Pr[b_i = b'_i] - \frac{1}{2^n}]$$

Start: The challenger **C** runs *Setup(d)*, sends the system public parameters *params* to **A** and keeps the *MSK* secret.

Query phase: The adversary **A** makes a polynomially bounded number of queries to **C** and has access to the following oracles. The only restriction is that **A** should not have queried the second part of the private key corresponding to the target identity *ID**.

– ***Keygen queries:*** **A** produces an identity *ID* and obtains the corresponding partial secret key of *ID*.

–***Signcrypt query***: **A** makes a query with the message *m* and the sender identity *ID* as input. **C** outputs the signcryption σ on *m* corresponding to *ID*.

The aggregate signcryption is not involved because it does not input a partial private key.

– ***Unsigncrypt query***: **A** submits the signcryption σ (from *ID*) as input. **C** outputs the corresponding message *m* if σ is a valid signcryption of *m* corresponding to *ID*.

– ***Aggregate-unsigncrypt query***: **A** submits the aggregate signcryption σ_{agg} corresponding to sender *ID*. If σ_{agg} is a valid aggregate signcryption on *ID*, the challenger returns all the corresponding messages $\{m_i\}_{i=1}^{n}$.

Selection phase: **A** produces *2n* messages $m_{0i}, m_{1i} \in [1,n]$ where the i^{th} messages are of equal length from the message space M with sender identity *ID**. **A** sends $\{m_{0i}, m_{1i}\}_{i=1}^{n}$ to the challenger **C**. The adversary **A** must not have queried these messages before and there is a restriction against **A** accessing the master private key *x* in the start phase and the second part of the patial private key *(s)*.

– ***Challenge*:**The challenger **C** selects a random $j \in [1,n]$ and considers a random C_j instead of the real $Enc_{y_j}(m_j)$ to make a signcryption $\sigma_j^* =< C_j, Y_j, Z_j >$. **C** flips a coin to sample a bit *b*← *{0, 1}* for every message $m_{bi} \forall i \neq j$ to signcrypt them. Then, **C** aggregates σ_i for all *i*, including invalid σ_j^*, and sends σ_{agg}^* to **A**.

Response: **A** outputs a guess bit $b' \in \{1,0\}$ for every message *i*. **A** wins the game if *b'=b* for every message. That **A** can distinguish *j* and the invalidity of σ_j^*.

3.5.2 Unforgeability

The signature of our scheme is inspired by [20]. This signature which is very efficient and specialized for WSNs, consist of low power and low storage sensors. In the following, we specify a brief explanation of the unforgeability proof adapted from [20].

Definition: An ID-based aggregate signcryption scheme is said to be secure in the sense of existential unforgeability against chosen message attack (EUF-IBAS-CMA)[19, 20] in the random oracle model, if no probabilistic polynomial time adversary **A** has a non-negligible advantage in the following game.

Start: The challenger **C** runs *Setup(d)*, sends the system public parameters *params* to **A** and keeps *MSK* secret.

Query: The adversary **A** makes a polynomially bounded number of queries to the challenger **C**. The attack may be conducted adaptively, and allows the same queries as in the IND-IBAS-CCA game, namely *Keygen* and *Signcrypt* queries.

Forgery: At the end of the game, **A** issues a new signcryption σ_{agg}^* for node *ID**, where the second part of the patial private key *(s)* and the signcryption oracle must not have been queried on *ID** and $m_i^* \forall i$. **A** wins the game if σ_{agg}^* is a valid aggregate signcryption of $m_i^*, i \geq 1$ (σ_{agg}^* contains at least one forged message).

4. Identity Based Aggregate Signcryption

In this section, we describe the assumptions about the network and the adversary. Also, we consider several well-known attacks in these kinds of networks.

4.1 System Assumptions

Suppose there are some UWMNS which consist of *N* sensors and a station. The station has to visit the network periodically. sensors collect data during *collection intervals,* each of which is divided into *v rounds*. At the end of each round, the collected data will be signcrypted by a sensor, and then at the end of each interval all signcryptions will be aggregated into one unit of data to be sent. These signcryptions are threatened by an adversary denoted as **A** during the interval. **A** is either curious, or aims to prevent the receipt of data by the station, or even change the data to deceive the station. In this paper, we propose a new scheme to defend against reading attack (through encryption), changing the measured data (through signing) and deleting attack (through alerting). The alert informs the station to supply the deleted critical data via other neighbouring sensors. Below, we describe the condition of the sensors and the adversary:

Data collection policy: Each sensor collects some unit of data for each round, or equivalently, a sensor collects n data $(m_i\ i=1$ to $n)$ for each interval. By applying the proposed algorithm, these n units of data will be aggregated into one unit for sending.

Cryptographic capabilities: Each sensor can perform 160-bit elliptic curve cryptography. Also, any sensor is equipped with a True-Random Number Generator (TRNG).

Re-initialization: The network is unattended; that is, as soon as each sensor offloads its accumulated data to the station, every sensor immediately and securely erases the previous values and receives the new interval key and partial private key as seed; also the round counter is reset to zero.

***Portrait of the adversary*:** We now focus on the description of the anticipated adversary.

Compromise power: We envision a powerful mobile adversary. We assume that **A** is capable of compromising at most k out of N sensors within a particular time interval ($0 <k <N/2$). This subset of compromised sensors is not clustered or contiguous; that is, concurrently compromised sensors can be spread through the entire network. Note that once **A** compromises a sensor and as long as it remains compromised, **A** is able to read from, write to, and delete all or some of the stored data.

Compromise and collection rounds: For ease of exposition, we assume that all of the rounds have the same duration. Also, both types of compromising and collection rounds start and end at the same time.

Limited erasure capacity: Between any two successive station visits, **A** can erase no more than a given number of measurements from the network. Otherwise, this raises an alarm at the station and contradicts **A**'s goal of remaining undetected.

Stealthy operation: **A**'s movements between intervals are unpredictable and untraceable; that is, once **A** moves from one set of k sensors to the next, it leaves no trace behind.

Network knowledge: **A** knows the composition and topology of the network. During the time when **A** compromises a given sensor, it can read from, write to, and delete the compromised sensor's stored data as desired. Thus, it can learn all of the sensor's secrets durimg compromising.

Defence awareness: **A** is fully aware of any scheme or algorithm that any sensor uses for defence. Our assumptions are similar to [21], while our adversary goals are inspired by [22].

In this setting, we consider a proactive adversary that can compromise sensors before identifying the target; that is, it essentially starts compromising sensors at round 1 before receiving any information about the target sensor and the target critical data collection round. It chooses and compromises different sensors in a geographic area even before such a signal is received. This powerful adversary who is usually referred to as a mobile adversary, can even roam around the network and change from one set of compromised nodes to another. We consider several attacks (explained two of them), each with different goals, inspired by [21].

Curious: **A** aims to learn as many measured data as possible. Generally, if no *countermeasures* are taken, it is not difficult to read data from the RAM and ROM of a commodity sensor, as demonstrated in [23]; that is, **A** can simply compromise sensors and learn the data directly. This data might be specific sensor measurements that disclose critical or high-value information. The encryption technique, which is part of our scheme, protects the entire data set against this attack.

Search-and-replace: Consider a network that aims to monitor nuclear activity. The station raises an alarm once any sensor reports a value above a certain threshold. In this case, **A**'s goal is to find that value and replace it with a concocted one before the value reaches the station. In this case, the station compares the data with the data of neighbouring sensors. This kind of network can tolerate lost data (and **A** knows this). However, in our integrated model, any data are encrypted by an incomputable true-random key. This key is concatenated to the both previous and current signcryptions. Consequently, all of the signcryptions in one interval are connected like a chain. This trick prevents the adversary from erasing some random or target signcryptions from many sensors.

Mostly, adversaries aim to be stealthy and undetected. Moreover, **A** strives not only to attack but also, to remain undetected. If it succeeds in doing so, its movements become not only unpredictable but also untraceable.

4.2 The Proposed Scheme

The new identity based aggregate signcryption scheme for unattended WSNs consists of the algorithms *Setup, KeyGen, Signcrypt, Aggregate-Signcrypt, Unsigncrypt,* and *Aggregate-Unsigncrypt*, which are explained below. Suppose that identity *ID* signcrypts messages m_i, $i \in [1,n]$ and finally aggregates them. Note that the signature of our scheme is inspired by [20].

- **Setup:** Let d be a security parameter of the system. We define an elliptic curve E on a finite field GF_v where v is a prime power number discussed in Section 3.2. Let G_1 be an additive cyclic subgroup of the group of EC points (including infinity point O_E) with g and q as the generator and prime order of G_1, respectively. Let G_2 be a multiplicative group with prime order q. Let H be a set of hash functions: $H_1 : G_1 \times \{0,1\}^* \to Z_q^*$, $H_2 : G_1 \times G_1 \times \{0,1\}^* \to Z_q^*$, $H_3 : G_1 \times \{0,1\}^* \to \{0,1\}^l , l = q + |m| + |G_1|$, where $|G_1|$ and $|m|$ are the size of $G_1$and the length of message m, respectively. Let the master private key *"Msk"* be $x \in Z_q^*$ and the master public key $X = xg$. Therefore, the public parameter is:

  $$\text{“}Params\text{”} = <G_1, G_2, X, g, H>, Msk = x$$

- **KeyGen*(ID)*:** To generate a partial secret key for identity *ID*, the *KeyGen* selects $t \in Z_q^*$ at random, and then computes:

  $$\frac{t}{x} = r , \qquad R \leftarrow rg \quad , \qquad s \leftarrow r + xH_1(R, ID) \bmod q ,$$

  We call $H' = H_1(R, ID)$. The sensor partial private key is *(R, s).* A correctly generated partial secret key should fulfill $sg = R + XH'$ (1). This technique enables any server to receive data, with parameter *(r)* corresponding to the *ID*.

- **Signcrypt(*m_i*, *ID*, *(R, s)*, *int*):** The signcrypt algorithm inputs a message ***m_i***, sender identity ***ID***, sender partial private key ***(R, s)***, and interval number ***int***. This function uses the key $t_1 \in Z_q^*$ which is shared with the station at the beginning of the interval. Setting t_1 between the sensor and the station prevents the adversary from replacing the whole signcryptions with forged ones (a kind of replacing attack). Note that every t_i and y_i will be securely erased at the end of the i^{th} round. Note that every y_i and k_i will be securely erased at the end of the i^{th} round. For every message m_i, the sensor performs the following steps:

  Computes t_{i+1}=*TRNG()* mod q where *TRNG()* is a true random number generator function; Obtains signature Z_i by computing $y_i + sH_2(Y_i, R, m_i) \bmod q$ where $Y_i = gy_i$; $y_i \in Z_q^*$ at random. Finally computes $C_i = [i \| m_i \| K_{i+1}] XOR [H_3(Rk_i, ID)]$ to make ciphertext where $k_i = y_i + t_i$. The signcryption of message m_i is $\sigma_i =< C_i, Y_i, Z_i >$.

- **Aggregate-signcrypt($\{\sigma_i\}_{i=1}^n$, *ID*):** On receiving n individual signcryptions $\sigma_i =< C_i, Y_i, Z_i >$, where *i=1* to n and identity ***ID*** as sender, the aggregation is as follows. The output is the aggregation $\sigma_{agg} =< \{C_i, Y_i\}_{i=1}^n, Z_{agg} >$ where $Z_{agg} = \sum_{i=1}^n Z_i$. Every σ_{agg} is a unique packet and is composed of all ciphertexts and keys in addition to the total signature Z_{agg} . This packet will be sent to the station. Due to every K_i can be obtained once by computing $Y_i + T_i$ and one more time from $C_{i-1} = Enc_{y_{i-1}}(i-1 \| m_{i-1} \| K_i)$, the station can investigate all K_i, *i=1* to n. Therefore, the station can decide on the integrity of σ_{agg} . Moreover, this technique prevents adversary from inserting fake signcryption σ'_i in σ_{agg} . In **Fig. 1**, we show the aggregate signcryption (σ_{agg}). Note that, in order to prevent a replay attack, i can be considered as a time stamp. This trick can alarm professionals with regard to the replaying of false, dangerous data. This attack can threaten the patient's life and is very dangerous.

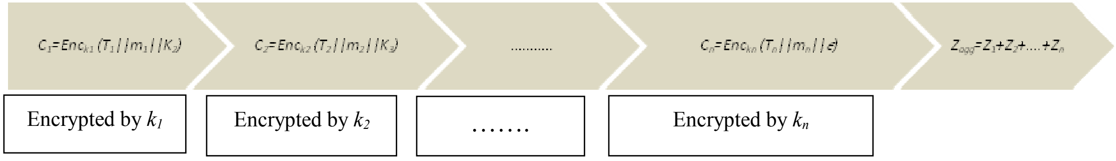

Fig. 1. Aggregate signcryption of m_i, *i=1,..,n*

- **Unsigncrypt** $(\sigma_i, ID, (R, s), \text{int})$ **:** The receiver (which maybe the station) executes this algorithm with σ_i, sender identity ***ID***, its partial private key ***(R, s)***, and interval number ***int***. This algorithm outputs ***m_i*** if it is valid, otherwise it outputs *"invalid"*. Note that, as explained in *Aggregate-signcrypt()*, the key of round *i* (i.e. Y_i) from $\sigma_i =< C_i, Y_i, Z_i >$ should be equal to the Y_i of the previous Unsigncrypt $(\sigma_{i-1}, ID, (R, s), \text{int})$ = ($i-1 \| m_{i-1} \| K_i$), otherwise the integrity of message m_i is doubtable.

To unsigncrypt every message, the sensor computes $[C_i]XOR[H_3(rK_i,ID)]$ $=[C_i]XOR[H_3(r(Y_i+T_i),ID)]$ to obtain $i\,||\,m_i\,||\,K_{i+1}$.

The receiver verifies $gZ_i=Y_i+Rh_i+XH'h_i$ (where $h_i=H_2(Y_i,R,m_i)$). If this check passes, outputs the message *(m*$_i$*, ID)*, else outputs *"Invalid"*.

- **Aggregate-Unsigncrypt(**σ_{agg} ***, ID, (R, s), int*****):** The receiver executes the algorithm with $\sigma_{agg}=<\{C_i,Y_i\}_{i=1}^{n},Z_{agg}>$, the sender identity ***ID***, its partial private key ***(R, s)***, and the interval number ***int***. This algorithm outputs $m_i\forall i$ for every valid message, otherwise it outputs *"invalid"*.

For every message, the sensor computes $[C_i]XOR[H_3(rK_i,ID)]=[C_i]XOR[H_3(r(Y_i+T_i),ID)]$ to obtain $i\,||\,m_i\,||\,K_{i+1}$.

To verify the aggregate signcryption σ_{agg} for messages $m_i\forall i$ and identity *ID*, the verifier should compute $h_i=H_2(Y_i,R,m_i)$ for $m_i\forall i$ and verifies $gZ_{agg}=[\sum_{i=1}^{n}Y_i]+[(R+XH')\sum_{i=1}^{n}h_i]$. If this check passes, outputs $(m_i\forall i)$ corresponding to *ID*, else outputs *"Invalid"*.

4.2.1 Correctness of The Proposed Scheme

$$gZ_{agg}=g\sum_{i=1}^{n}(y_i+sh_i)=\sum_{i=1}^{n}(gy_i+gsh_i)=\sum_{i=1}^{n}(Y_i+gsh_i)=$$
$$\sum_{i=1}^{n}(Y_i+g\left(r+xH'\right)h_i)=\sum_{i=1}^{n}(Y_i+\left(R+XH'\right)h_i)=\left[\sum_{i=1}^{n}Y_i\right]+\left[\left(R+XH'\right)\sum_{i=1}^{n}h_i\right]$$

5. Proof of Security

In this section, we give the formal proof of confidentiality and unforgeability. Confidentiality of signcryption ensures that nobody can discover the main message. We show that the adversary **A** should solve the *Computational Diffie Hellman Problem (CDHP)* [17] to detect the message. As for unforgeability, we prove that **A** should solve *Elliptic Curve Discrete Logarithm Problem (ECDLP)* [17] to forge a message.

5.1 Confidentiality

Theorem: The proposed identity based aggregate signcryption scheme is secure against adversary **A** under an adaptive chosen ciphertext attack in the random oracle model if the CDHP assumption holds in G_1.

Proof: To prove, we show that if **A** can threat the security of the scheme, then a challenger **C** can use **A** as a sub-routine to solve an instance of CDHP, i.e. **C** can find the solution of *rcg* with the given instance $(g,rg,cg)\in G_1^3$, where $r,c\in Z_q^*$ are unknown. To do so, **C** sets the oracles $H_2\in Z_q^*ID=ID^*(R^*,\perp)(|m_{bi}|=|m_{1i}|)i\neq j$ at random. **C** then considers *KeyGen*, *signcrypt*, H_1 (in the *KeyGen* oracle), and H_2 oracles as follows. In this proof, the adversary has the restriction that it cannot access the partial key *(s)* of *(R,s)* corresponding to the target identity *ID** and *MSK=a*. **A** is given additive cyclic group G_1 from EC points with generator *g* and prime order *q*.

First the adversary **A** outputs the identity *ID**which it aims to attack. Then, the challenger

C gives **A** the system parameters *params*, consisting of *g* and $X = ag$. **C** will also simulate all oracles required during the game. It controls $H_3 \grave{o} Z_q^*$ at random and answers **A**'s other queries as follows:

KeyGen (ID) oracle: **C** randomly chooses $r_1, r_2, r_3 \in Z_q^*$ and sets $s \leftarrow r_2$, $H_1 \leftarrow -r_1$ and $R \leftarrow r_1 X + r_2 g$. Note that *(R, s)* generated in this way satisfies Equation (1) in the *KeyGen* algorithm. It is a valid partial private key. **C** outputs *(R, s)* as the secret key of *ID*. If *ID=ID**, this algorithm outputs private key *(R*, ε)*.

***Signcrypt (m_i, ID, R, s) oracle*:** When **A** makes a signcryption query for the signcryption m_i with *ID* corresponding to *(R, s)*, **C** returns $\sigma_i =< C_i, Y_i, Z_i >$ to **A,** where the partial private key *(R, s)* is generated by querying the *KeyGen* oracle.

Aggregate-signcrypt $(\sigma_i \forall i)$ ***oracle*:** After receiving $\sigma_i =< C_i, Y_i, Z_i >$ *(i=1* to *n)*, the challenger aggregates them as $Z_{agg} = \sum_{i=1}^{n} Z_i$. The final aggregate signcryption is $\sigma_{agg} =< \{C_i, Y_i\}_{i=1}^{n}, Z_{agg} >$.

Unsigncrypt $(\sigma_i, ID, (R, s))$ ***oracle*:** **A** submits the signcryption σ_i from *(R,s)* belonged to the *ID* as input. Using Unsigncryption algorithm, **C** outputs the corresponding message *m* if σ is a valid signcryption of *m* corresponding to *ID*. If **A** queries σ^* then invalid value will be returened.

Aggregate-unsigncrypt $(\sigma_{agg}, ID, (R, s))$ ***oracle*:** **A** submits the aggregate signcryption σ_{agg} corresponding of sender ID with private key *(R, s)*. If σ_{agg} is a valid aggregate signcryption on *ID*, the challenger using Aggregate-unsigncryption algorithm, returns all the corresponding messages $\{m_i\}_{i=1}^{n}$. If **A** queries σ_{agg}^* then invalid value will be returened.

***Challenge*:** In this phase, **A** cannot ask for an aggregate unsigncrypt on the challenge aggregate signcryption σ_{agg}^* . After getting sufficient training, **A** submits the tuple (m_{0i}, m_{1i}) where $|m_{0i}|=|m_{1i}|$ *i=1 ...n,* with sender *ID** to **C**. The challenger performs the following steps:

- Chooses a random $j \in [1, n]$.
- Sets $y_j^* = d \in Z_q^* => Y_j^* = dg$ and $T_j^* = cg \in G_1$;
- Computes $H_2 = d$;
- Updates list H_2;
- Computes $Z_j^* = d + (r_2)(d) = (r_2 + 1)d$;
- Chooses a random C_j^* and sets the signcryption of message m_{bj} as $\sigma_j^* =< C_j^*, Y_j^*, Z_j^* >$.

For each $i \neq j$, **C** randomly chooses $b \in \{0,1\}$ and signcrypts m_{bi} as the normal *signcrypt* oracle using the sender's private key. Finally, **C** aggregates all the signcryptions $\sigma_i^* \forall i$ and gives the challenge aggregate signcryption σ_{agg}^* to **A**.

***Output*:** After **A** has carried out sufficient trainings, **A** outputs the guess b'_i for *i=1* to *j – 1.* For the j^{th} output, if the adversary aborts then the adversary has found out that it is not a valid signcryption of either of the messages (m_{0j}, m_{1j}) (we assume that the adversary is capable of doing this). If so, **C** gets $H_3(rkg, ID^*)$ to decrypt C_j^*. Such a tuple exists because **A** must have queried the *H₃* oracle to aggregate unsigncrypt the challenge ciphertext successfully and found out the error. Note that the probability of **A** guessing the hash value is negligible). More exactly, **C** receives a random instance $(g, K_j^*, R^*) = (g, Y_j^* + T_j^*, R^*) = (g, cg + dg, rg) \in G_1^3$ in which *d* is disclosed to **A** by quering *H₂* oracle. Here **A** wants to compute $rkg = r(c+d)g = rcg + rdg \in G_1$ where *rdg=dR* is easily computable but obtaining *rcg* is the computational Diffie Hellman problem. In other words, given $(g, cg, rg) \in G_1^3$ from $(g, K_j^*, R^*) = (g, Y_j^* + T_j^*, R^*) = (g, cg + dg, rg) \in G_1^3$, computing *rcg* is the hard computational problem. **C** runs **A** as a subroutine and acts as **A**'s challenger in the IND-IBAS-CCA game.

5.2 Unforgeability

We now consider proof of unforgeability; that is, nobody can forge an arbitrary message such that the verification algorithm accepts the validity of the forged message. The security of unforgeability is based on the hardness of the EC-DLP [17, 20]. More exactly, if adversary **A** aims to forge a message, it should break EC-DLP. This proof is adapted from [20].

Theorem: The proposed identity based aggregate signcryption scheme is secure against EFU-IBAS-CMA adversary **A** in the random oracle model if the EC-DLP is hard in G_1.

***Proof*:** We show that if **A** is capable of forging the proposed scheme then on getting an EC-DLP instance *ag* as challenge, the challenger **C** can use **A** to solve the EC-DLP and get *a*. To do so, the adversary **A** outputs the identity *ID** which it intends to attack. **A** outputs a forged aggregate signcryption $\sigma_{agg}^* =< \{C_i, Y_i\}_{i=1}^n, Z_{agg}^* >$ in which at least one forged message exists. The main challenge in unforgeability is security and validation of forged signature(s) in σ_{agg}^* signcrypted by *ID**. This proof is similar to the confidentiality, and hence only changes are presented.

***Signcrypt (mᵢ, ID, R, s) Oracle*:** The adversary **A** queries the signcryption oracle for message *m* and an identity *ID*. The challenger **C** first checks whether *ID* has been queried for the random oracle *H₁* or *KeyGen* oracle before. If so, it just retrieves *(R, s, H₁(ID, R))* from the table and uses these values to sign for the message, according to the scheme. It outputs the signcryption *<C, Y, Z>* for the message *m* and stores the value *H₂(Y, R, m)* in the hash table for consistency. If *ID* has not been queried for the *KeyGen* oracle, **C** executes the simulation of the *KeyGen* and uses the corresponding partial private key to sign the message and to add the value of *(R, s, H₁(ID, R))* to the table.

Forgery: Finally **A** outputs a forged aggregate signcryption including $\sigma_{(1)}^* =< C^*, Y^*, Z_{(1)}^* >$ on message m^* and identity ID^*. **C** rewinds **A** to the point where it queries $H_2(Y^*, R^*, m^*)$ and supplies a different value. **A** outputs another pair of signcryption $\sigma_{(2)}^* =< C^*, Y^*, Z_{(2)}^* >$. This is achieved by running the Turing machine again with the same random tape but with a different hash value. **C** repeats and obtains $\sigma_{(3)}^* =< C^*, Y^*, Z_{(3)}^* >$. Note that Y^* and R^* should be the same every time. We let *c₁*, *c₂*, and *c₃* be the output of the random oracle queries $H_2(Y^*, R^*, m^*)$ for the first, second, and third times. By $r, x, y \in Z_q^*$, we now denote the EC-discrete logarithm of *R*, *X*, and *Y* respectively, that is, *rg=R*, *xg=X*, and *yg=Y*. From the signature equation, we have:

$$Z_{(i)}^* = y + rc_i + xc_i H(R^*, ID) \bmod q \quad for\ ? = \ ,?,?$$

In these equations, only *r*, *x*, and *y* are unknown to **C**. **C** solves for these values from the above three linear independent equations and outputs *x* as the solution of the EC-DLP.

6. Performance Analysis

In this section, we compare our scheme with those of Selvi et al. [24], Kushwah et al. [25] and Barreto et al. [26] with regard to the communication and computation costs. Note that to the best of our knowledge, no work on signcryption is applied in UWSNs. Since user storage is very important in wireless networks; comparison on the user side is presented. To make the EC-discrete logarithm intractable, we assume, as in [27][28], that the size of both the key and secure one-way hash function are 160 bits.

Table 1 lists the computation cost of the schemes on the user side. 'BM', '$\hat{e}$'and 'EXP' denote bilinear map, bilinear pairing evaluation and exponentiation operation in multiplicative group, respectively. 'M' and 'A' are scalar multiplication and addition in additive groups respectively. Finally, 'H' represents hash operation. The scheme of Selvi et al. has the highest computation cost, because of the bilinear pairing which is a very costly operation. The main advantage of our proposal is that there is no need to carry out pairing on either the user or the receiver side. The time costs of bilinear pairing and multiplication calculated by Oliveira et al. [29] are 5.45 and 0.00402 seconds, respectively, using the binary field and MIRACL library [30]. Szczechowiak et al. [31] were able to compute the pairing in 1.71 seconds on TelosB.

Table 1. Comparison of signcryption computation times.

Operation	Proposal	Selvi et al. [24]	Kushwah et al. [25]	Barreto et al. [26]
Bilinear Map (BM)	0	1	0	0
Exponentiation (EXP)	0	0	0	1
Scalar Mul in G_1 (M)	2	3	2	2
Add in G_1 (A)	0	1	1	1
Hash operation (H)	2	2	3	3
Total	2M+2H	1BM+3M+1A+2H	2M+1A+3H	1EXP+2M+1A +3H

*Note that the bilinear pairing, multiplication, and addition are based on ECC in the proposed scheme.

The unsigncryption computation times are compared in **Table 2**. **Table 1** and **Table 2** show that our scheme is the most efficient with regard to the computation. **Table 3** and **Table 4** represent the communication overhead and user storage order. In the case of the communication overhead, our proposal and the ones by Selvi et al. and Barreto et al. are close and slightly better than that of Kushwah et al.

But according to **Table 4**, using our scheme, the sensors have to save one more key of size $|Z_q^*|$. With the additional $|Z_q^*|$, first, any authorized server is able to receive and aggregate unsigncrypted data, second, the partial private key can be refreshed at the beginning of every interval, and third, the adversary cannot detect any sensor private parameters of the next rounds by compromising the current interval. Thus the data collected in future will be secure. Consequently, our scheme is the most efficient signcryption in terms of computation and communication and is also efficient for user storage of the master key and public parameters. An insignificant disadvantage of the proposal is that the size of partial private key of each user has $|Z_q^*|$ more bits.

Table 2. Comparison of unsigncryption computation times.

Operation	Proposal	Selvi et al. [24]	Kushwah et al. [25]	Barreto et al. [26]
Bilinear Map (BM)	0	1	2	1
$\hat{e}$ Evaluation ($\hat{e}$)	0	3	1	1
Exponentiation (EXP)	0	0	0	1
Scalar Mul in G_1 (M)	4	0	1	1
Add in G_1 (A)	2	0	1	0
Hash operation (H)	2	2	3	3
Total	4M+2A+2H	1BM+3 $\hat{e}$ +2H	2BM+1 $\hat{e}$ +1M+1A+3H	1BM+1 $\hat{e}$ +1EXP+1M+3H

Note that the bilinear pairing, multiplication, and addition are based on ECC in proposal scheme.

Table 3. Comparison of communication overheads.

Proposal	Selvi et al. [24]	Kushwah et al. [25]	Barreto et al. [26]
$\lvert M\rvert+\lvert G\rvert+\lvert Z_q^*\rvert$	$\lvert M\rvert+\lvert ID\rvert+2\lvert G\rvert$	$\lvert M\rvert+2\lvert G\rvert+\lvert\lvert Z_q^*\rvert\rvert+\lvert ID\rvert$	$\lvert M\rvert+2\lvert G\rvert$

|M|: the length of message, |G|: size of additive cyclic group, $\lvert Z_q^*\rvert$: size of multiplicative group

Table 4. Comparison of user storage.

Scheme	Master public key	Public parameters	Partial private key
Proposal	$\lvert G\rvert$	$2\lvert G\rvert$	$\lvert G\rvert+\lvert Z_q^*\rvert$
Selvi et al. [24]	$\lvert G\rvert$	$2\lvert G\rvert$	$\lvert G\rvert$
Kushwah et al. [25]	$\lvert G\rvert$	$2\lvert G\rvert+\lvert Z_q^*\rvert$	$\lvert G\rvert$
Barreto et al. [26]	$\lvert G\rvert$	$3\lvert G\rvert+\lvert Z_q^*\rvert$	$\lvert G\rvert$

7. Conclusion

In this paper, we provide a secure measuring system in which data will be confidential and authorized. We further study the security of collected data through an identity-based signcryption scheme. Our scheme is formally proven to be secure against chosen cipher text and existential unforgeability against chosen message attacks in the random oracle model. These are the strongest security notions for message confidentiality and authentication, respectively. In addition, in comparison with similar work [24][25], our scheme is the most efficient in terms of time and space orders; that is, both sender and receiver need the lowest time and space overheads to make the system secure. The presented scheme also naturally provides integrity.

In future work, we aim to improve our work by applying the homomorphic property. By applying this property, the sensors are able to make a secure connection through the network. On the other hand, by studying other difficult problems, we will improve our work to gain linear time efficiency. These new properties help networks to transmit data securely and efficiently.

References

[1]. Y. Zheng, "Digital signcryption or how to achieve cost (signature & encryption) << cost (signature) + cost (encryption)", *Advances in Cryptology,* LNCS 1294, pp.165-179, Springer-Verlag, 1997. Article (CrossRef Link)

[2]. Trident Systems, Trident Family of Unattended Ground Sensors. Available online http://www.tridsys.com/white-unattended-ground-sensors.htm (accessed on 21 December 2009).

[3]. Information Processing Technology Office (IPTO) Defense Advanced Research Projects Agency (DARPA), BAA 07-46 LANdroids broad agency announcement, http://www.darpa.mil/IPTO/solicit/open/BAA-07-46_PIP.pdf , 2007.

[4]. R. D. Pietro, L. V. Mancini, C. Soriente, A. Spognaedi, G. Tsudik, "Playing hide-and-seek with a focused mobile adversary in unattended wireless sensor networks"*, Ad Hoc network*, vol. 7, no. 8, pp. 1463-1475, 2009. Article (CrossRef Link)

[5]. C. Karlof, N. Sastry, D. Wagner, "TinySec: A link layer security architecture for wireless sensor networks", *in proceeding of the 2nd ACM conference on embedded networked sensor systems,* pp. 162-175, USA, 2004. Article (CrossRef Link)

[6]. D. Ma, G. Tsudik "Extended abstract: Forward-secure sequential aggregate authentication", In *proc. of IEEE symposium on security and privacy*, pp. 86-91, USA, 2007. Article (CrossRef Link)

[7]. D. Ma, "Practical forward secure sequential aggregate signatures", In *proc. of the 3rd ACM symposium on information, computer and communications security*, pp. 341-352, USA, 2008. Article (CrossRef Link)

[8]. D. Ma, G. Tsudik, "A new approach to secure logging", *ACM transaction on storage,* vol. 5, no. 1, pp. 1-21, 2009. Article (CrossRef Link)

[9]. R.D. Pietro, A. Spognardi, C. Soriente, G. Tsudik, "Collaborative authentication in unattended WSNs", In *proc. of ACM conference on Wireless Network Security* pp. 237-244, Switzerland, 2009. Article (CrossRef Link)

[10]. J. Deng, R. Han, S. Mishra, "Defending against Path-based DoS attacks in wireless sensor networks", In *proc. of ACM workshop on Security of Ad-hoc and Sensor Networks,* pp. 89-96, USA, 2005. Article (CrossRef Link)

[11]. C. Kraub, M. Schneider, C. Eckert, "Defending against False-endorsement-based DoS attacks in wireless sensor networks", In *proc. of ACM conference on Wireless senor network Security,* pp. 13-23, USA, 2008. Article (CrossRef Link)

[12]. C. M. Yu, C. Y. Chen, C. S. Lu, S. Y. Kuo, H. C. Chao, "Acquiring authentic data in unattended wireless sensor networks", *Joural of sensors,* pp. 2770-2792, 2010. Article (CrossRef Link)

[13]. I. Almomani, M. Saadeh, "Security model for tree-based routing in wireless sensor networks, structure and evaluation", *KSII transactions on internet and information systems*, vol. 6, NO. 4, pp. 1223-47, 2012. Article (CrossRef Link)

[14]. Certicom,"Certicom announces elliptic curve cryptosystem (ECC) challenge," *press release,*Winter1997.http://www.certicom.com/index.php?action=company,press-archive&view=121

[15]. A. K. Lenstra and E. R. Verheul, "Selecting cryptographic key size,"Journal of Cryptography: *Journal of the International Association for Cryptographic Research,* 2001. Article (CrossRef Link)

[16]. E. Blab and M. Zitterbart, "Efficient implementation of elliptic curve cryptography for wireless sensor network," *Technical report TM-2005-1,* Institute of Telematics, University of Karlsruhe Zirkel 2, D-76128 Karlsruhe, Germany, 2005.

[17]. A. W. Dent and Y. Zheng, "Practical signcrypiton" *Springer-verlag,* 2010.

[18]. F. Liu and M. K. Khan, " A survey of identity based signcryption," *IETE technical review journal,* vol. 28, issue. 34, pp. 256–72, 2011. Article (CrossRef Link)

[19]. Y. Dodis "Signcryption (short survey)," *Encyclopedia of Cryptography and Security*, Springer-Verlag, 2005.

[20]. J. K. Liu, J. Beak, J.Zhou, Y. Yang, and J. W. Wong, "Efficient online/offline identity based signature for wireless sensor network," *International Journal of Security,* vol. 9, no. 4, pp.

287–296, 2010. Article (CrossRef Link)

[21]. D. Ma, C. Soriente, and G. Tsudik, "A new adversary and threats: security in unattended wireless sensor networks," *IEEE Network Conference,* vol. 23, no. 2, pp. 43–48, 2009. Article (CrossRef Link)

[22]. A. Herzberg, S. Jarecki, H. Krawczyk, and M. Yung, "Proactive secret sharing or: How to cope with perpetual leakage," *in D. Coppersmith (Ed.), CRYPTO,* vol. 963, LNCS, pp. 339–352, 1995. Article (CrossRef Link)

[23]. J. Deng, C. Hartung, R. Han, and S. Mishra, “A practical study of transitory master key establishment for wireless sensor networks," *IEEE Secure Communication,* 2005. Article (CrossRef Link)

[24]. S. Selvi, S. Vivek, R. Srinivasan, C. Rangan, “An efficient identity-based signcryption scheme for multiple receivers,” *IWSEC, LNCS,* vol. 5824, pp. 71–88, 2009. Article (CrossRef Link)

[25]. P. Kushwah and S. Lal, “Efficient generalized signcryption schemes,” *Journal of Theoretical Computer Science,* vol. 412, no. 45, pp. 6382–6389, 2011. Article (CrossRef Link)

[26]. P. S. L. M. Barreto, B. Libert, N. McCullagh and J. J. Quisquater, “Efficient and provably-secure identity based signcryption from bilinear maps,” In *Proc. of advances in cryptography,* LNCS 3788, Springer,-Verlag, pp. 515-32, 2005.

[27]. A. Lenstra, E. Tromer, A. Shamir, W. Kortsmit, B. Dodson, J. Hughes, and P. Leyland, “Factoring estimates for a 1024-bit rsa modulus,” *In C. Laih (ed.), Advances in Cryptology*, Lecture Notes in Computer Science, Vol. 2894,pp. 55–74, 2003. Article (CrossRef Link)

[28]. FIPS PUB 180-2, 2004. Secure Hash Standard, National Institute of Standards and Technology, US Department of Commerce.

[29]. L. Oliveira, M. Scott, J. Lopez, and R. Dahab, “TinyPBC: Pairings for authenticated identity-based non-interactive key distribution in sensor networks,” In *Proc. of Networked Sensing Systems*, pp. 173–180, 2008. Article (CrossRef Link)

[30]. MIRACL Big Integer Library, http://www.shamus.ie/, 2009.

[31]. P. Szczechowiak, A. Kargl, M. Scott, and M. Collier, “On the application of pairing based cryptography to wireless sensor networks,” In *Proc. of ACM Wireless Network Security conference,* USA, pp. 1–12, 2009. Article (CrossRef Link)

Faezeh Sadat Babamir received her B.Sc. degree of Computer Science (Hardware) from Shahid Bahonar university of Kerman and M.Sc. degree of Computer Science (Cryptography) from Shahid Beheshti university of Tehran in 2009 and 2012, respectively. From 2008 to 2010, she worked on application of genetic algorithm in software testing engineering as well as optimization in wireless sensor networks. From 2010 to 2012 she worked on signcryption and evaluation of different security techniques in wireless sensor networks. Now she is working toward her Ph.D. degree. Her research interests include signcryption, coding theory, security in healthcare aware wireless sensor networks and genetic algorithms.

Ziba Eslami received her B.S., M.S., and Ph.D. in Applied Mathematics from Tehran University in Iran. She received her Ph.D. in 2000. From 1991to 2000, she was a resident researcher in the Institute for Studies in Theoretical Physics and Mathematics (IPM), Iran. During the academic years 2000-2003, she was a post doctoral fellow in IPM. She served as a non-resident researcher at IPM during 2003-2005. Currently, she is the professor in the department of Computer Sciences at Shahid Beheshti University in Iran. Her research interests include design theory, combinatorial algorithms, cryptographic protocols and steganography.