Access Control Protocol with Node Privacy in Wireless Sensor Networks

Pardeep Kumar, *Member, IEEE,* Andrei Gurtov, *Senior Member, IEEE,* Jari Iinatti, *Senior Member, IEEE,* Mangal Sain, and Phuong H. Ha

***Abstract*—For preventing malicious nodes joining wireless sensor networks (WSNs), an access control mechanism is necessary for the trustworthy cooperation between the nodes. In addition to access control, recently, privacy has been an important topic regarding how to achieve privacy without disclosing the real identity of communicating entities in the WSNs. Based on elliptic curve cryptography (ECC), in this paper we present an access control protocol with node privacy (called ACP) for the WSN. The proposed scheme not only accomplishes the node authentication but also provides the identity privacy (*i.e.,* source to destination and vice-versa) for the communicating entities. Compared with the current state of the art, the proposed solution can defend actively against attacks. The efficacy and efficiency of the proposed ACP are confirmed through the test-bed analysis and performance evaluations.**

***Index Terms*—Access control, authentication, security and privacy, wireless sensor networks.**

I. Introduction

NOWADAYS, the advancement in electronics, communications, information technologies and the Internet have led to the rapid proliferation of wireless sensor networks (WSN). WSNs are envisioned as the future technology, and are emerging as an interesting research area among the research organizations, academia, and industries [1]. Sensor-enabled products and their networks are becoming commonplace, and central to everyday life, *e.g.,* healthcare, smart homes, object tracking and monitoring, and so on. Moreover, according to the latest finding "wireless sensor networks 2012-2021" [2] - WSN businesses will grow rapidly to over two billion US dollars for the future systems in 2022.

WSNs consist of a large number of inexpensive sensors that have quite limited resources (*e.g.,* low processing units, low bandwidth, limited battery power, and low memory) [3]. Sensors are small in size, and are integrated with a sensing unit and wireless communication capabilities. These nodes are being deployed in a wide terrain to perform their intended tasks efficiently. Typically, heterogeneous sensor networks are more practical, having better network performance (*i.e.,* multi-hop communication, delay tolerant, *etc.*), provide scalability and efficient load-balancing [4]–[6]. However, with the increasing ubiquity of WSNs in real applications (*e.g.,* hospitals, military, wildlife monitoring), WSNs data will be available almost everywhere, anytime. Inevitably, much of WSNs information is highly sensitive and critical, and thus it is possible that an adversary may introduce malicious nodes in a network to leak the sensors information (and/or insert false reports) without the consent of the network owner. In addition, an adversary can purposefully interrupt the network smooth functionality by deploying the malicious nodes into the network. Therefore, to protect such a information leakage from the global adversaries and malicious nodes, access control mechanisms are to be enforced to real WSNs from the beginning of a WSN deployment [7]–[13].

Another big forthcoming issue for actually deploying WSNs is how to achieve (access control with) network privacy without disclosing the sensors (real) identity (ID), which is considered as one of the imperative security concerns for critical and real WSNs. For instance, consider a WSN (*e.g.,* wildlife monitoring) scenario, where the endangered animals [14], [15] are being monitored and detected using a low-cost wireless sensor nodes. A deployed sensor node sends (animal) report (including source and destination IDs) to the base station. However, in such a use-case, the hunter may directly deploy a malicious node to generate the false tracking report for the endangered animal species. In addition to that an adversary can eavesdrop the insecure wireless communication to trace the animal location by analyzing the message header that contains both the source and destination identities (IDs). This kind of privacy breach would cause severe life-threatening consequences to the endangered animals. Thus, how to achieve efficient access control without disclosing the nodes identities (*i.e.,* source and destination and vice-versa) has become an important requirement in real sensor networks.

In recent years, many access control protocols have been proposed to protect WSNs [7]–[13], as shown in Section II-A. Most of the schemes focus on authentication and key establishment to address access control, neglecting other relevant but sometimes paramount aspects such as privacy. Generally, privacy includes two types of concerns, *data-centric privacy* and *context-aware privacy* [14]. *Data-centric privacy* includes secure integrity of the data gathered that is transmitted to the sink. While, in a *context-aware privacy*, how to prevent adversaries from gaining access to the context information, for instance, identity, physical location and so on. Data-centric privacy has been addressed significantly (*e.g.,* Zhang et al [16], Sicari et al [17], Yang et al [18] and Yao et al [19]

P. Kumar, and P. H. Ha are with the Department of Computer Science, UiT The Arctic University of Norway, N-9037, Tromso, Norway (e-mails: pkumar@cs.uit.no, phuong.hoai.ha@uit.no)

A. Gurtov is with IDA, Linkoping University, Sweden and ITMO University (e-mail gurtov@acm.org)

J. Iinatti is with Centre for Wireless Communications, University of Oulu, P. O. Box 4500, FI-90014, Finland (e-mails: jari.iinatti@ee.oulu.fi)

M. Sain is with Dongseo University, San 69-1, Jurye-2-Dong, Sasang-Gu, 617-716, Busan, Republic of Korea (e-mail: mangalsain1@gmail.com)

schemes), but ignoring the node identity privacy. Similarly, in the context-aware privacy, recently, Debnath et al [14], Li et al [15] and Pnogaliur-Xiao [20] have paid a significant attention to the source node privacy and leaving out the destination node privacy. In these protocols, the source node identity (*i.e.,* sensor) is either hashed or encrypted, whereas, the destination node identity is being used a plain-text. However, in the real WSN most of the queries are generally requested and/or issued at the point of base stations or gateway nodes. In such scenarios, the existing solutions can provide the guaranteed source nodes (*e.g.,* base station/gateway) identity privacy, but they cannot provide the destination nodes (such as, a sensor node) identity privacy. Therefore, an attacker can easily monitor the base station/gateway initiated wireless packets, and can intercept the sensor nodes identity (*i.e.,* destination node). Thus, it is needless to say, the identity privacy of the involved nodes (source to destination and vice-versa) not been properly addressed in real WSNs.

The aim of this work is to design an access control protocol with node privacy (ACP) that would take care of the node (identity) privacy (*i.e.,* from source to destination and vice-versa) in WSNs. The proposed ACP utilizes elliptic curve cryptography (ECC), and provides explicit mutual authentication between the (transmitter and receiver) nodes, and also establishes a shared secret session key. At the same time, the scheme ensures nodes privacy, *i.e.,* without disclosing their real identities. Similar to Pnogaliur-Xiaos protocol [20], the scheme proposed in this paper also seamlessly integrates two underlying cryptographic primitives (*e.g.,* encryption and hash value) to achieve the access control (*i.e.,* authentication and key establishment) with node privacy. The ACP scheme possesses identity privacy while providing robust security against passive and active attacks. We have demonstrated a test-bed on Tmote Sky platform for evaluating the ACP. In this paper, we analyse the ability of proposed scheme (*e.g.,* efficiency) in terms of overhead when compared to the existing literatures, and believe that the proposed protocol can be used in many practical WSNs where identity privacy is highly required.

The remainder of this article is organized as follows. Section II details the existing work on access control and privacy-preserving schemes. Section III presents the proposed scheme and Section IV presents security analysis. Section V details implementation and performance evaluations of the proposed ACP. Finally, Section VI concludes the ACP scheme.

II. Related work

The section is divided into two: access control and privacy preserving protocols.

A. Access Control Protocols

Zhou et al proposed an access control protocol, which is based on ECC [7]. The scheme is more efficient than the RSA-based public key cryptography schemes. Authors claim that a new node (using the timestamp) could join in the network any time and support key establishment. However, to authenticate a sensor node, Zhou et al's scheme incurred substantially high computational and communicational costs. In real WSNs, the high consumption rates, therefore, may be a severe bottleneck. Thereby, based on ECC and hash-chain, Huang's proposed a novel access control protocol (NACP) [8]. NACP is quite adequate for low-powered sensor nodes. Huang's showed that the NACP can be easily implemented as a dynamic access control system because all the secrets and broadcasting information in existing nodes should not be updated once a new node is added into the network. In 2009, Kim-Lee pointed out that NACP scheme is susceptible to a message replay attack, and has lack of the hash-chain renewability problem, for details the readers may refer to [9]. In addition, Kim-Lee proposed an enhanced novel access control protocol (ENACP) that uses a hash-chain approach to perform authentication and key establishment between two nodes [9].

Later, Zeng et al [10] and Shen et al [11] showed the ENACP also suffers from inherent design flaws and is vulnerable to many attacks (*e.g.,* masquerade attack). Extending the same idea of NACP, Huang's proposed a new design of access control protocol that exploiting the clock of a node and provided security at higher computational cost than the previous proposed schemes [12].

In 2012, Lee et al [13] demonstrated that ENACP is not practical for real environments and is susceptible to message forgery attack and a new node masquerade attack. To solve ENACP problems, Lee et al proposed practical access control protocols (PACPs) for WSNs and claimed that PACPs are secure against many attacks [13]. PACPs comprised of two schemes, namely, secure PACP (secPACP) and memory-efficient PACP (ePACP). However, Chen et al [21] pointed out that the large number of pre-stored keys (in PACPs) are subjected to the adversary attacks and required unnecessarily huge keys storage overhead at a resource-hungry sensor node.

Note that in the aforementioned works, the authors have given considerable efforts to the access control protocols, leaving out other relevant but sometimes paramount aspects like node anonymity (*i.e.,* identity privacy).

B. Privacy-preserving Protocols

During the last decade, a significant amount of research papers have been published, addressing mainly two privacy concerns in WSNs: (i) *data-centric privacy*, and (ii) *context-aware privacy*.

Data-centric privacy focuses on proving protection for the data items. Zhang et al proposed two privacy-preserving data aggregation protocols, namely, PASKOS (privacy-preserving based on anonymously shared keys and omniscient sink) and PASKIS (privacy-preserving based on anonymously shared keys and ignorant sink) [16]. Authors exploited the concept of data perturbation, where each node computes a perturbed data, *i.e.,* adding the secret keyed value to the sensed data and transmits this perturbed data to the sink node. In PASKOS, the secret keyed values are computed based on pre-distributed key rings, which are randomly chosen from a key pool, offline. Authors assumed that the sink possesses the whole key pool in PASKOS; whereas, in PASKIS the sink node doesn't have

knowledge of the key pool. Sicari et al proposed DyDAP (dynamic data aggregation scheme for privacy aware protocol) that provides an end-to-end secure perturbation based data aggregation by employing a privacy function [17].

Yang et al proposed a precision-enhanced and encryption-mixed privacy-preserving data aggregation (PEPDA) [18]. The main focus of PEPDA is to reduce collision during data transmission and energy consumption, and how to compensate losses that are caused by the collision. Based on the similar idea that proposed by Yang et al [18], Yao et al proposed two privacy-preserving data aggregation protocols: PDAAS (privacy-preserving data aggregation against non-colluded aggregator and sink), and PDACAS (privacy-preserving data aggregation against colluded aggregator and sink) for two-tiered WSNs with mobile nodes [19]. The PDAAS and PDACAS protocols also exploited perturbation concept to secure the sensor data, and thus provide the data privacy.

Context-aware privacy ensures the privacy of context-related information, such as, the location from which the data being transmitted (including source node identity) and the location where the data being received (including destination node identity). To deal with the node privacy, Pongaliur-Xiao proposed a source node privacy and packet recovery under eavesdropping and node compromised attacks (SPENA) [20]. SPENA has employed on the encryption-based cryptosystem that increases the source node privacy. In addition, SPENA uses a one-way hash-chain based keying system to hide the source node information from the adversary. Debnath et al proposed privacy in WSNs using ring signature, utilizing the ID-based public key cryptography [14]. Source-location privacy (SLP) based routing scheme is proposed by Li et al [15]. The SLP uses two-phase routing: routing to a single randomly selected intermediate node (SRIN) and routing through the network mixing ring (NMR).

Notice that, Debnath et al [14], Li et al [15] and Pnogaliur-Xiao [20] have paid the attention to the source node privacy but ignoring the destination node privacy.

III. PROPOSED SCHEME

This section presents the proposed ACP, *i.e.,* access control with node (identity) privacy. As shown in Fig. 1, consider the practical (heterogeneous) WSN that maintains the network load-balancing, energy-efficiency, and lifetime [4]–[6]. The network architecture consists of three types of entities: the low-cost sensor node (SN), the coordinator node (C), and the base station (BS). Typically, a SN is small in size and low cost; thus, it is restricted on computation, memory storage, and communication capability. Generally SNs are deployed to sense and collect the environmental reading continuously or event/time based. As shown in Fig. 1, the WSN is divided into the cells and each cell has a coordinator (C) node, where the deployed SNs securely forward their sensory data to the base station through the C node and the other way around. In addition, the C node can also collect and aggregate the environment data securely from the SNs within a cell [22], [23]. The C node has powerful resources with different capabilities and it can directly contact to the BS over a long transmission range [21].

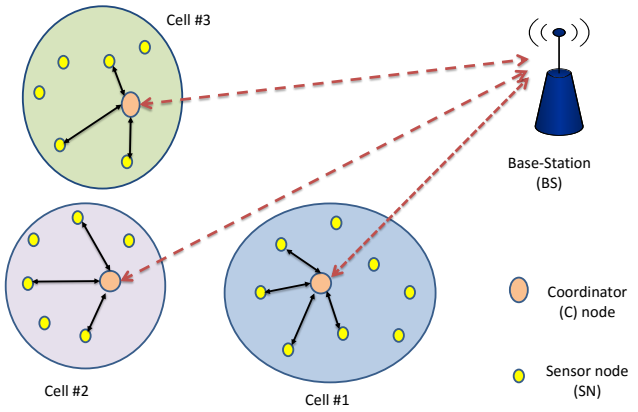

Fig. 1. Wireless sensor network model.

TABLE I
NOTATIONS AND DESCRIPTIONS

Notations	Descriptions
ID_U	Identity of sensor (U)
ID_C	Identity of Cell (C)
LKS	Large key space
$E_K[M]$	Encryption using secret key *(K)*
$D_K[M]$	Decryption using secret key *(K)*
F_q	A finite field
E	Elliptic curve defined on finite field F_q with prime order n
G	Group of elliptic curve points on E
P	A point on elliptic curve E with order n
r	large prime number
β	a salt
π	a long-term secret
$h(.)$	Secure one-way hash function, SHA1/SHA2/SHA3/MD5
$\|\|$	Concatenation operation

Finally, the whole WSN is controlled by the base station (BS). The BS has large bandwidth, strong computing capability, large memory, and high power to support the cryptographic, routing and other requirements for the whole network.

The proposed scheme includes three phases: (a) system initialization; (b) authentication and key establishment; and (c) new node addition phase.

Before starting the WSN deployment, consider the elliptic curve discrete logarithm problem (ECDLP), to find an integer r, given an elliptic curve E defined over Fq, a point $P \in E(Fq)$ of order n, and a point $Q = rP$ where $0 \leq r \leq n-1$, as shown in [13]. The notation and description are shown in Table I.

A. System Initialization Phase

The BS, first, performs off-line tasks and distributes the required security parameters to the sensor and coordinator nodes, as follows.

1) The BS generates a large key space, $LKS, (e.g., K_1, K_2, K_3, \cdots K_N)$ and key identifiers (K_{idi}), identities for sensors (ID_s) and cell coordinators (ID_{Ci}).

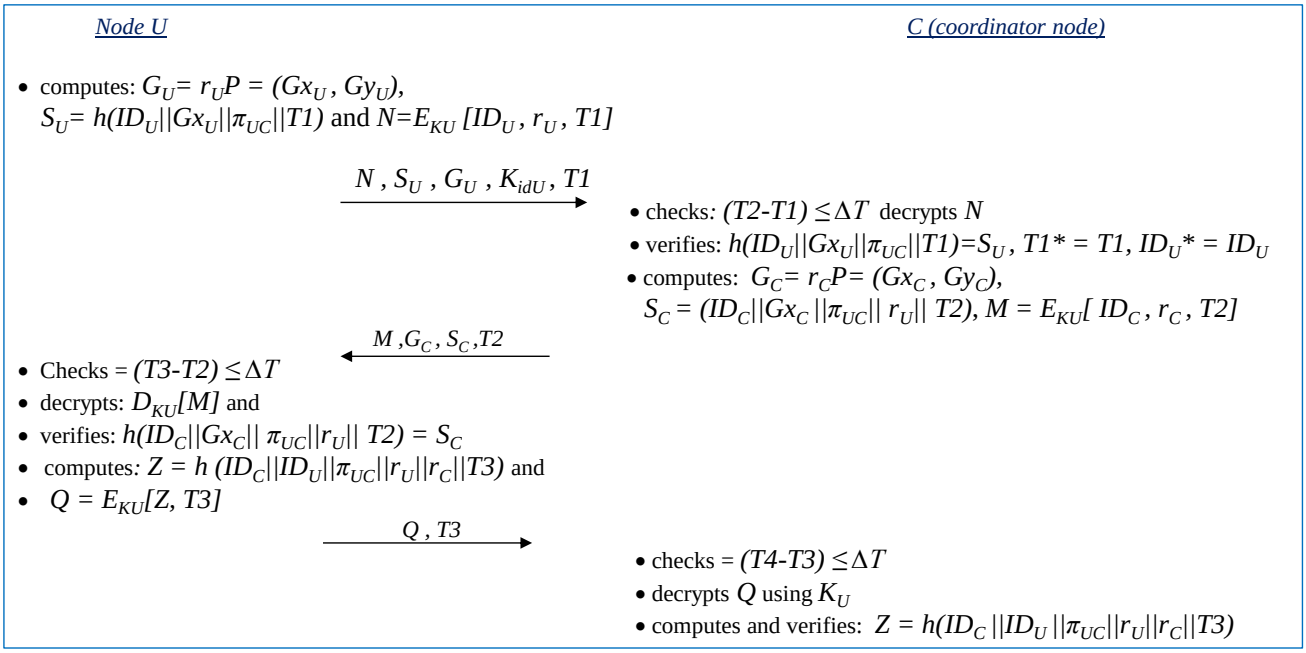

Fig. 2. Flow of authentication and key establishment.

2) The BS chooses a sensor node (*e.g.,* U) for the cell deployment and assigns the cell identity (ID_{Ci}) to the U, where U will be deployed. For each sensor, U, the BS generates a salt (β), and computes an unique secret, *i.e.,* $\pi_{UC} = h(ID_U||\beta||ID_{Ci}||ID_{BS})$.
3) Now, the BS installs a secret key (K_U) and its identifier (K_{idU}), π_{UCi} and ID_{Ci} to the dedicated node (*i.e.,* node U), which is selected for the dedicated cell (*e.g.,* C_i). Finally, all parameters of the assigned sensors (*i.e.*, IDs, secret keys and corresponding key identifiers, and unique secret, $\pi_{UC} = h(ID_U||\beta||ID_{Ci}||ID_{BS})$) are stored to their corresponding C node.

Further, to execute the scheme, this paper has made the following assumptions: (i) in a cell, each sensor synchronizes the clock with its coordinator node (*i.e.,* C) using Kim et al [1] and/or Du et al [24]; and (ii) The BS and the C node are trusted and secured entities.

B. Authentication and Key Establishment Phase

To establish a pairwise key with the C node, node U initiates the following procedure:

1) The node U generates a random number r_U and computes the point $G_U = r_U P = (Gx_U, Gy_U)$ over the elliptic curve, $S_U = h(ID_U||\ Gx_U||\pi_{UC})||T1)$ and $N = E_{KU}\ [ID_U, r_U, T1]$. Here, E is the encryption algorithm (*e.g.*, AES (advanced encryption standard) [25]), and $T1$ is the current time-stamp of node U. The U then sends $N, S_U,\ G_U, T1, K_{idU}$ to the C node.
2) Upon receiving message, the C node checks if $(T2 - T1) \leq \Delta T$; if that is not true, then aborts the ACP. Here, ΔT is the expected transmission delay. Otherwise, the C node gets the corresponding secret key of K_{idU} (*i.e.,* K_U) from own database and decrypts N and obtains $ID_U{}^*, r_U{}^*, T1^*$. It computes $S_U{}^* (= h(ID_U||\ Gx_U||\pi_{UC}||T1))$ and verifies $S_U{}^* = S_U$, $ID_U{}^* = ID_U$ and $T1^* = T1$. If this holds, then the C node is assured that G_U is generated by a legal node. Now the C node generates a random number r_C and computes the point $G_C = r_C P = (Gx_C,\ Gy_C)$, $S_C = h(ID_C||Gx_C\ ||\pi_{UC}||r_U||T2)$, and $M = E_{KU}[ID_C, r_C,\ T2]$. Here, E denotes the identical encryption scheme (AES [25]) and $T2$ is the current time-stamp of the C node. The C node sends $M, G_C, S_C, T2$ to the node U.
3) Now, the node U verifies $(T3 - T2) \leq \Delta T$; if it holds, then further steps are executed; otherwise, the node U aborts the system. The node U decrypts message M and obtains $ID_C{}^*, r_C{}^*, T2^*$. To verify SC, the node U computes $S_C{}^* (= h(ID_C||Gx_C||\pi_{UC}||r_U||T2))$ and verifies $S_C{}^* = SC$, $ID_C{}^* = ID_C$ and $T2^* = T2$. If conditions hold, the node U assured that G_C is generated by a legal C node, and it computes $Z = h(ID_C||ID_U||\pi_{UC}||r_U\ ||r_C||T3)$, and $Q = E_{KU}[Z, T3]$. It then sends $Q, T3$ to the C node.
4) After receiving a message, the C node verifies $(T4 - T3) \leq \Delta T$; if it is true, then further steps are executed. Otherwise, C aborts the system. Now, the C node decrypts Q using K_U and obtains Z^* and $T3^*$. Thereafter, it computes $Z = h(ID_C||ID_U||\pi_{UC}||r_U||r_C||T3)$ and verifies $Z^* = Z$ and $T3^* = T3$. If this holds, then the C node confirms that the node U is a legitimate node and Z will be used for further secure communication between

the node U and the C node. The flow of authentication and key establishment phase is shown in Fig. 2.

C. New Sensor Addition Phase

It is very practical to replace the sensor nodes with exhausted batteries with new sensors during the network operation. In the proposed scheme, replacing the exhausted sensors with the new nodes in a cell is easy.

The BS simply preloads the required parameters, K_X, K_{idX}, π_{XCi}, and ID_{Ci} (refer initialization phase, Section III-A) to the newly added sensor node (*e.g.,* node X) and assigns the dedicated cell where it needs replacing. In addition, the BS securely passes the information of the new sensor to the assigned cell, as shown in Fig. 3. After deploying the new sensor node into the cell, it will perform authentication and key establishment (as shown in Fig. 2) and maintain a trustworthy connection with the network.

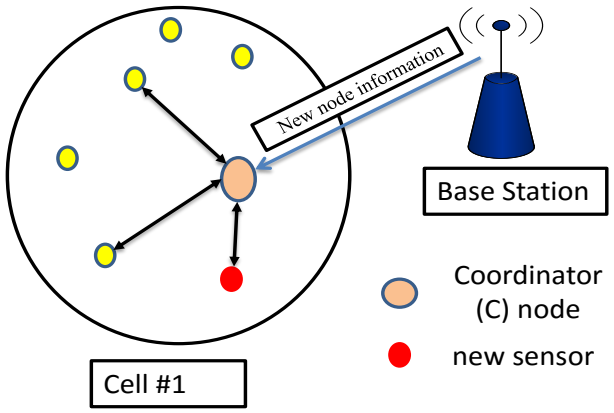

Fig. 3. New node addition.

IV. Security Analysis

A. Security Analysis

The security of the proposed framework is based on secrecy of the one-way hash function, the elliptic curve discrete logarithm problem (ECDLP), and the encryption algorithm.

To facilitate the security discussion, consider that an attacker has full control over the wireless channels *e.g.,* he/she can insert, drop, modify, replay the wireless messages, and/or steal the nodes identities. Based on the attack model, the security analysis includes the security against a message replay attack, legal node masquerading attack, message forgery attack, identity threat, Sybil attack, and node capture and fabrication threat [12], [13].

Resistance to the message replay attack: In this attack, an attacker wants to perform a message replay attack using previously broadcasted messages. For instance, an adversary sends the node U's captured message, $N, S_U, G_U, K_{idU}, T1$ to the C node, at time Ta (*i.e.,* adversary time-stamp). The adversary must then provide r_U and real-time $T1$ to verify the sub-message S_U at the C node. However, it is not feasible for the adversary to obtain r_U under a practical assumption that the ECDLP is hard to breach. Moreover, the replayed message will be detected at an early stage because of the time interval $(T2-T1) < \Delta T$ and the attackers time stamp (Ta) would not be verified at the C node. Here, ΔT is the mutually agreed transmission delay between the node U and the coordinator (C) node. Therefore, an attacker cannot succeed in replaying old messages to the node C.

Similarly, assuming that an adversary sends the C's captured message $(M, G_C, S_C, T2)$ to the node U at time Ta. The attacker then also would have to provide r_C and real-time $T2$ to verify the sub-message S_C at the node U. However, to obtain a real r_C is hard because of the complexity of the ECDLP. In addition, the attacker attempts will be detected because of the mutually agreed time interval condition, *i.e.,* $(T3 - T2) < \Delta T$. Thus, the adversary cannot succeed in the replay message attack at the node U. □

Resistance to legal node masquerade attack: In the proposed scheme, the node U computes S_U with the secret of π_{UC} and the random number r_U for the current session. Similarly, for the identical session, the C node computes S_C with the secret π_{UC} and r_C. Assuming that even if the adversary could monitor a message flow between the node U and the C node, and obtains the S_U, G_U and S_C, G_C, respectively, as shown in Fig. 2, it is hard for the adversary to derive the correct S_U $(= h(ID_U||Gx_U||\pi_{UC})||T1))$ and S_C $(= h(ID_C||Gx_C||\pi_{UC}||r_U||T2))$ due to the one-way hash operation. In addition, the adversary could not derive the legalized session key $Z = h(ID_C||ID_U||\pi_{UC}||r_U||r_C||T3)$, it is computed over the r_U and r_C that are difficult to derive due to the ECDLP hardness. Therefore, the our scheme can resistance to a legal node masquerade attack. □

Resistance to forgery attack: Assume that an attacker has intercepted values (*e.g.,* $G_U = r_U P$ and $G_C = r_C P$) by the monitoring on the wireless communication channel between the node U and the C node. Now the attacker may attempt to establish a legitimate message, for instance, N, Sa, Ga, K_{idU}, Ta. However, the attacker would not be able established a legal Sa because he/she does not have knowledge of the secret π_{UC}. More importantly, under the assumption of ECDLP, it is very hard for the attacker to derive the random number r_U and r_C from G_U and G_C, respectively. Therefore, the attacker message N, Sa, Ga, K_{idU}, Ta will be captured by the C node. Thus, the proposed ACP is resistant to a message forgery attack. □

Protection from privacy threat: Recall context-centric privacy (*e.g.,* identity) and data-centric privacy from Section II-B.

Node identity privacy (i.e., context-centric): In the proposed scheme, the node identity privacy can be guaranteed by the security of cryptosystem $(i.e., E_K[m])$. If the ciphertext is secure, so does the node identity [20]. Consider an adversary, he/she monitors the broadcasts, tries to intercept the source and destination nodes identities to learn about the WSN by recreating the context from the flow of wireless messages, which are

either time-based or event-based temporal information. The mapping of node identities data can breach the privacy of WSNs. However, in the proposed scheme, an attacker can compare the two packets (*i.e.,* broadcasted messages) as string length of bits, but cannot derive the nodes identities from the source and destination (and vice-versa), since IDs are shielded, *i.e.,* encrypted, $N = E_{KU}[ID_U, r_U, T1]$ and $M = E_{KU}[ID_C, r_C, T2]$, using the secret key K_U that possessed by the legal parties (*such as,* the node U and the C node).

Data-centric privacy: An attacker cannot illegally breach the data privacy in the ACP scheme. Since, two legal communicating entities being mutually authenticated each other, establishing a hashed key ($Z = h(ID_C||ID_U||\pi_{UC}||r_U||r_C||T3)$), which is an unique key for every session, and will be used to provide the data protection for the current session. Hence the attacker cannot breach the data privacy over wireless channels. □

Protection from Sybil attack: In this attack, a malicious sensor poses multiple fake identities to other non-compromised nodes. It is widely accepted that the Sybil attacks are unavoidable but they can be detected. However, in the proposed protocol, the secret $S_U(= h(ID_U||Gx_U||\pi_{UC}||T1))$ of node U is computed with the real identity of node U (ID_U), secret (π_{UC}), and timestamp (*cf.,* Section III-B: Step 1). Thus, a malicious node U cannot claim a new identity $ID_U{}'$ in the vicinity of C node, and therefore, the proposed scheme can withstand the Sybil attack [11]. Furthermore without knowing the secret key (K_U) of the node U (within a cell), it is difficult for the Sybil node to decrypt $N = E_{KU}[ID_U, r_U, T1]$. Moreover, to construct communication with other nodes through authentication, the attacker must obtain the real identity (ID_U) and time-stamp $(T1)$ of a legal node U, otherwise it would be difficult for an attacker to establish a pair-wise key. On the other hand, if an attacker somehow compromised the node U, then he/she cannot communicate with the other non-compromised nodes because the attacker does not possess the secrets (K_U, π_{UC}) of non-compromised nodes.

Furthermore, an intrusion detection techniques based on mutual guarding have been proposed by Bhuse et al [26], [27] means if an attacker succeeds in sending a fake identity to a legal node, then it is practical to detect the Sybil attack using a mutual guarding mechanism [26], [27]. For this mechanism, when two or more nodes (*e.g.,* nodes A and node B) are in direct broadcasting range where the broadcasted information sent by both nodes can be received by them, this is said to be mutually guarded.

For a simple illustration, consider an example of how two nodes can mutually guard each other, as shown in Fig. 4 [26]. Suppose that an attacker is located in a common area of the node A and node B. The attacker sends a message to the node B by using the source information (*e.g.,* ID, location, *etc.*) of the node A. Note that in this case, node A also receives the information that is broadcasted by the attacker. Therefore, node A can detect the presence of a Sybil node (to some magnitude) that being masquerading as node A. □

Security against node capture and fabrication attack:

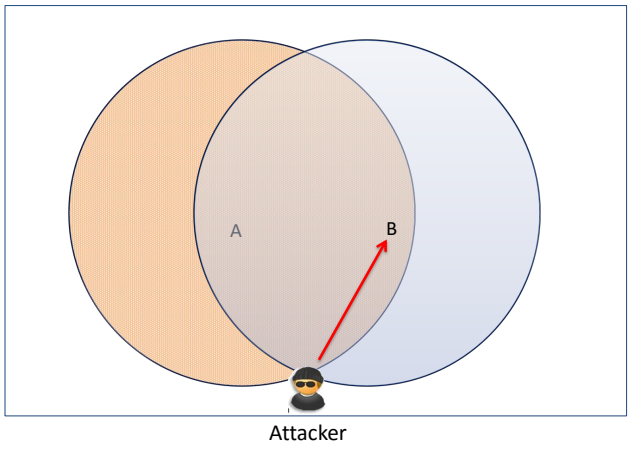

Fig. 4. Mutual guarding within a cell.

TABLE II
SECURITY COMPARISON

	ENACP [9]	secPACP [13]	ePACP [13]	PASKOS [16]	ACP
R1	Yes	Yes	Yes	Yes	Yes
R2	No	Yes	Yes	ND	Yes
R3	No	Yes	Yes	No	Yes
R4	No	No	No	No	Yes
R5	ND	Partial	Partial	ND	Yes
R6	ND	Yes	Yes	ND	Yes

R1-Resist replay attack; R2-Resist to sensor masquerade attack; R3-Resist forgery attack; R4-Protection from identity threat; R5-Protection from Sybil threat; R6-Protection against node capture and fabrication threat; ND - no discussion.

In practice, sensors are deployed in unattended and hostile environments. Therefore, one could physically capture a node and extract all the stored keys and other information (such as ID, clocks, and other parameters) about the network. To disturb the smooth functionality of the network, an attacker may fabricate a clone (node fabrication) with a compromised node. However, the proposed scheme is a variants of pairwise key pre-distribution protocols, which provide perfect network resilience against a compromised node attack – for more details the reader may refer to [13], [28], [29]. Moreover, in the proposed scheme, the C node can periodically monitor the misbehavior of SNs using [30] within its cell and informs the BS upon a detection because the proposed scheme is divided into a number of cells. Thus, compromising a node and node fabrication attack in the cell does not affect the security of non-compromised cells. □

Table II summarizes the security comparisons of the proposed ACP with [9], [13], [16], in terms of the security of access control method with node privacy. It can be seen from Table II that the ENACP is vulnerable to many attacks, *cf.,* [10], [11], [13]. Similarly, the PACPs (secPACP and ePACP) may be subjected to various adversary attacks [21]. In addition, none of these protocols take care for the source and destination

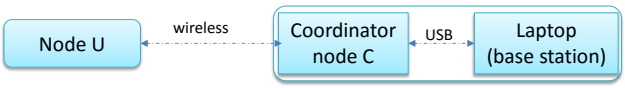

Fig. 5. Experimental setup.

node identities privacy. Whereas, the ACP scheme achieved the node identity privacy (from source to destination and vice-versa) than the ENACP [9], PACPs [13] and PASKOS [16].

V. Implementation and Performance Evaluation

We evaluate the proposed ACP by implementing the cryptographic components on the experimental test-bed using the TelosB platform.

A. Implementation and experimental setup

Our implementation environment is TinyOS 2 [31] running on the TelosB platform, having a 16-bit, 8 MHz CPU (MSP430) with 48 KB of program ROM and 10 KB of RAM [32]. Networked embedded systems C (nesC) being used as the development environment. Our test-bed has the coordinator node (C) that is serially connected to the laptop (base-station), and the sensor node (U), *cf.* Fig. 5. Both the nodes' clocks are synchronized using [25]. Following the Section III-B, the node U initiates the broadcast as a sender, while the node C acts as a cell coordinator, as shown in Fig. 2.

In experimental settings, we choose to use the AES (Advanced Encryption Standard) symmetric-key algorithm for the encryption, which is integrated in CC2420 radios [32]. For the ECC operations, we used TinyECC library [33]. TinyECC supports all elliptic curve operations over Fp, including point addition, point doubling and scalar point multiplication for TelosB platform. It supports SECG (standards for efficient cryptography group) recommended 128-bit, 160-bit and 192-bit elliptic curve domain parameters. For hashing, note that we recommend implementers to use high security hashing algorithms. We choose to use SHA-1 function for the sake of experiment purpose that provide one-way hashing as a base hash function. In ACP, we choose to use secp160r1 defined over a 160-bit prime field. The key size (K_U) for encryption is 128-bit, the length of time-stamp is 32-bit, 128-bit SHA-1 [34], and the key identity (K_{id}) is 8-bit. Therefore, the length of messages *i.e.,* $\{N, S_U, G_U, T1, K_{idU}\}$, $\{M, G_C, S_C, T2\}$, and $\{Q, T3\}$ are 57, 56, 20 bytes, respectively.

B. Evaluation

It is accepted that any inclusion of security principles will incur additional overhead. Nevertheless, due to the sensor nodes scarcity nature, this paper evaluates the price of security (in terms of program size (*i.e.,*memory), execution time and energy overhead for each operation of the proposed scheme) at the sensor node. Table III shows security prices at the sensor node including the following operations: ECC (the time for one point multiplication computation over an elliptic curve); AES (the time for executing an encryption); and SHA-1 (the time for executing the one-way hash function).

TABLE III
SECURITY PRICES FOR THE PROPOSED ACP

	RAM (KB)	ROM(KB)	time *(in* ms*)*	Energy *(in* µJ*)*
ECC	1.25	13.6	2,765	14,931
AES	0.76	10.1	3.9	21.06
SHA-1	1.9	9.7	36.3	196.02

TABLE IV
COMPUTATION COST COMPARISONS

	ENACP [9]	[12]	SecPACP [13]	ePACP [13]	**ACP**
T_{pm}	$2T_{pm}$	$5T_{pm}$	$2T_{pm}$	$2T_{pm}$	$1T_{pm}$
T_{hc}	$2T_{hc}$	–	–	–	–
T_h	$4T_h$	$2T_h$	$5T_h$	$4T_h$	$3T_h$
T_C	–	–	–	–	$3T_C$

T_{pm} - the time for executing point-multiplication; T_{hc} - the time for executing hash-chain; T_h - the time for executing one-way hash function; T_C - the time for performing cryptosystem (*i.e.,* encryption and decryption).

All together, our program code required 3.91 KB of RAM and 33.4 KB of ROM storage at the node U. The node U took 2805 ms execution time, which is due to the fact of (expensive) scalar point multiplication operations (*i.e.,* computing the point $Q = rP$, required 2,765 ms).

The energy consumption of each operation can be estimated using $W = V \times I \times t$, here V is the voltage, I is the current draw in active mode with radio off, and t is execution time. The voltage (*i.e.,* 3 V(volt)) and the current (*i.e.,* 1.8 µA (micro-amp)) values are directly obtained from TelosB data-sheet [32]. By multiplying the values of V and I with the execution time (t), we can determine the amount of energy required to execute the ECC, AES, and SHA-1 operations [25], [35], [36]. As we can seen in Table II, the total amount of energy consumed in ECC, AES and SHA-1 is $(\approx)$ 15,148 µJ. Notice that the impact of energy consumption from the AES and SHA-1 computation is low, *i.e.,* 21.06 µJ, and 196.02 µJ, respectively. Whereas, with the fact of the time complexity of the point multiplications, it consumes significant amount of energy, *i.e.,* 14,931 µJ.

Additionally, Table IV illustrates the computational overhead comparison of the proposed scheme with ENACP [9], Huangs [12] and PACPs [13]. ENACP requires two point multiplications $(2T_{pm})$, two hash chain operations $(2T_{hc})$ and four hash computations $(4T_h)$; Huangs scheme requires five point multiplications $(5T_{pm})$ and two hash computations $(2T_h)$, and secPACP and ePACP (in PACPs) requires $(2T_{pm}+5T_{hc})$ and $(2T_{pm}+4T_h)$, respectively. However in the proposed scheme (*cf.,* Fig. 2), a sensor node computes a single point multiplication operation $(1T_{pm})$, and three one-way hash operations $(3T_h)$. In order to achieve context and data privacy, a sensor node needs to perform three cryptosystem operations (*i.e.,* two encryption and one decryption) that does not require high computational complexity.

Considering our implementations (*i.e.,* the time to perform a point multiplication operation on a sensor node is 2,765

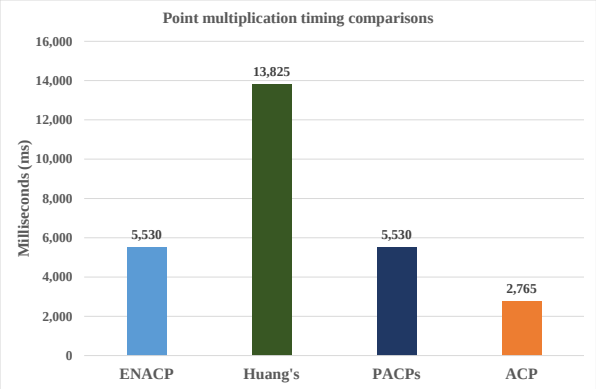

Fig. 6. Point multiplication comparisons in the terms of execution time on a sensor node.

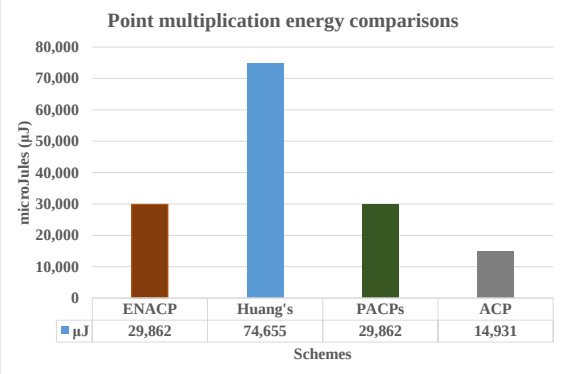

Fig. 7. Point multiplication comparisons in the terms of energy consumptions on a sensor node.

ms) – ENACP [9] and PCAPs [13] takes 5,530 ms and 5,530 ms, respectively, as shown in Fig 6. Huangs scheme [12] requires 13,825 ms, whereas the proposed ACP incurred less time than the protocols presented in [9], [12], [13]. Moreover, Fig. 7 shows the point multiplication energy consumption comparisons on a sensor node – ENACP [9] and PCAPs [13] consumes 29,862 µJ and 29,862 µJ, respectively. Huangs scheme [12] consumes 74,655 µJ of energy. While on the contrary our proposed ACP required 14,931 µJ of energy for the point multiplication operation. From Fig. 6 and Fig. 7, it can be observed the cost of point multiplication operation of the proposed ACP required the half of time and energy.

As shown in Table III, a hash operation consumes 196.02 µJ of energy on a sensor node in our proposed scheme. Considering the similar settings, the ENACP scheme [9] requires 784.08 µJ of energy for executing $4T_h$ operations. Huangs scheme [12] consumes 392.04 µJ for $2T_h$ operations; and SecPACP [13] and ePACP [13] requires 980 µJ for $5T_h$ operations and 784.08 µJ for $4T_h$ operations, respectively. Whereas, the proposed scheme consumes 588.06 µJ for $3T_h$

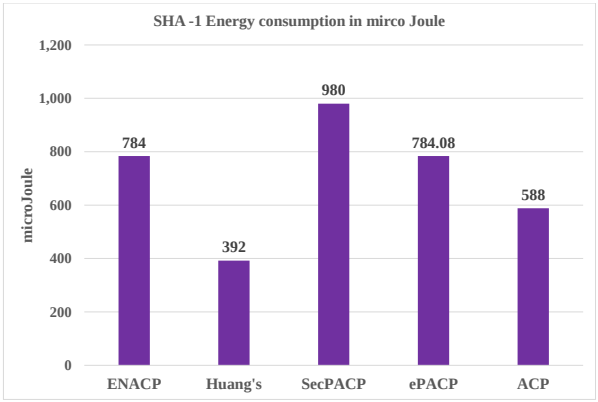

Fig. 8. SHA-1 energy consumption comparisons.

operations while providing the source to destination (and vice-versa) node identity privacy. Fig. 8 roughly summarizing the total amount energy consumption for SHA-1 operations in the [9], [12], [13] and the proposed ACP.

VI. Conclusion

In real WSN, an access control mechanism is necessary for the trustworthy cooperation between the nodes, where sensors send/receive request to/from the base-station. In such two-way wireless networks, however, compromising identity privacy (of the nodes) can inadvertently leak event privacy and it can give away the event occurrence without the consent of the WSN owner. This paper has proposed an access control scheme with node identity privacy for WSN using ECC, hash function, and cryptosystem. The proposed scheme achieves the access control while taking care of the identity privacy (source to destination and vice-versa) of a node and provides robust security. We have evaluated the proposed ACP using a test-bed on the TelosB platform. We have analyzed the ability of ACP (*e.g.,* efficiency) in terms of overhead and energy, and compared to the existing literatures, *i.e.,*, ENACP, Huangs scheme, and PACPs. We believe that the proposed ACP can be feasible in many practical WSN applications where the access control and identity privacy are highly required.

Acknowledgment

This work has received funding from the European Union Seventh Framework Programme (EXCESS project, grant no. 611183) and from the Research Council of Norway (PREAPP project, grant no. 231746/F20). The part of this work is supported by RFBR research project (14-07-00252).

References

[1] B.-K. Kim, S.-H. Hong, K. Hur, and D.-S. Eom, "Energy-efficient and rapid time synchronization for wireless sensor networks," *Consumer Electronics, IEEE Transactions on*, vol. 56, no. 4, pp. 2258–2266, November 2010.

[2] K.-J. Kim and S.-P. Hong, "Privacy care architecture in wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2013, 2013.

[3] M. I. Salam, P. Kumar, and H. Lee, "An efficient key pre-distribution scheme for wireless sensor network using public key cryptography," in *Networked Computing and Advanced Information Management (NCM), 2010 Sixth International Conference on*. IEEE, 2010, pp. 402–407.

[4] Z. Zhang, M. Ma, and Y. Yang, "Energy-efficient multihop polling in clusters of two-layered heterogeneous sensor networks," *Computers, IEEE Transactions on*, vol. 57, no. 2, pp. 231–245, Feb 2008.

[5] J. Peng, T. Liu, H. Li, and B. Guo, "Energy-efficient prediction clustering algorithm for multilevel heterogeneous wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2013, 2013.

[6] A. Koubaa and M. Alves, "A Two-Tiered Architecture for Real-Time Communications in Large-Scale Wireless Sensor Networks: Research Challenges," in *17th Euromicro Conference on Real-Time System (ECRTS 05)*, July 2005, pp. 1–4.

[7] Y. Zhou, Y. Zhang, and Y. Fang, "Access control in wireless sensor networks," *Ad Hoc Networks*, vol. 5, no. 1, pp. 3–13, 2007.

[8] H.-F. Huang, "A novel access control protocol for secure sensor networks," *Computer Standards & Interfaces*, vol. 31, no. 2, pp. 272–276, 2009.

[9] H.-S. Kim and S.-W. Lee, "Enhanced novel access control protocol over wireless sensor networks," *Consumer Electronics, IEEE Transactions on*, vol. 55, no. 2, pp. 492–498, 2009.

[10] P. Zeng, K.-K. R. Choo, and D.-Z. Sun, "On the security of an enhanced novel access control protocol for wireless sensor networks," *Consumer Electronics, IEEE Transactions on*, vol. 56, no. 2, pp. 566–569, 2010.

[11] J. Shen, S. Moh, and I. Chung, "Comment:enhanced novel access control protocol over wireless sensor networks," *IEEE Transactions on Consumer Electronics*, vol. 3, no. 56, pp. 2019–2021, 2010.

[12] H.-F. Huang, "A new design of access control in wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2011, 2011.

[13] H. Lee, K. Shin, and D. H. Lee, "PACPs: practical access control protocols for wireless sensor networks," *Consumer Electronics, IEEE Transactions on*, vol. 58, no. 2, pp. 491–499, 2012.

[14] A. Debnath, P. Singaravelu, and S. Verma, "Privacy in wireless sensor networks using ring signature," *Journal of King Saud University-Computer and Information Sciences*, vol. 26, no. 2, pp. 228–236, 2014.

[15] Y. Li, J. Ren, and J. Wu, "Quantitative measurement and design of source-location privacy schemes for wireless sensor networks," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 23, no. 7, pp. 1302–1311, 2012.

[16] L. Zhang, H. Zhang, M. Conti, R. Di Pietro, S. Jajodia, and L. V. Mancini, "Preserving privacy against external and internal threats in wsn data aggregation," *Telecommunication Systems*, vol. 52, no. 4, pp. 2163–2176, 2013.

[17] S. Sicari, L. A. Grieco, G. Boggia, and A. Coen-Porisini, "DyDAP: a dynamic data aggregation scheme for privacy aware wireless sensor networks," *Journal of Systems and Software*, vol. 85, no. 1, pp. 152–166, 2012.

[18] G. Yang, S. Li, X. Xu, H. Dai, and Z. Yang, "Precision-enhanced and encryption-mixed privacy-preserving data aggregation in wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2013, 2013.

[19] Y. Yao, J. Liu, and N. N. Xiong, "Privacy-preserving data aggregation in two-tiered wireless sensor networks with mobile nodes," *Sensors*, vol. 14, no. 11, pp. 21 174–21 194, 2014.

[20] K. Pongaliur and L. Xiao, "Sensor node source privacy and packet recovery under eavesdropping and node compromise attacks," *ACM Transactions on Sensor Networks (TOSN)*, vol. 9, no. 4, p. 50, 2013.

[21] C.-Y. Chen, A. D. Yein, T.-C. Hsu, J. Y. Chiang, and W.-S. Hsieh, "Secure access control method for wireless sensor networks," *Int. J. Distrib. Sen. Netw.*, vol. 2015, 2015.

[22] Y. Cheng and D. P. Agrawal, "An improved key distribution mechanism for large-scale hierarchical wireless sensor networks," *Ad Hoc Networks*, vol. 5, no. 1, pp. 35–48, 2007.

[23] X. Liu, "A survey on clustering routing protocols in wireless sensor networks," *Sensors*, vol. 12, no. 8, pp. 11 113–11 153, 2012.

[24] X. Du, M. Guizani, Y. Xiao, and H.-H. Chen, "Secure and efficient time synchronization in heterogeneous sensor networks," *Vehicular Technology, IEEE Transactions on*, vol. 57, no. 4, pp. 2387–2394, 2008.

[25] P. Kumar, A. Gurtov, J. Iinatti, M. Ylianttila, and M. Sain, "Lightweight and secure session-key establishment scheme in smart home environments," *Sensors Journal, IEEE*, vol. 16, no. 1, pp. 254–264, Jan 2016.

[26] V. Bhuse, A. Gupta, and A. Al-Fuqaha, "Detection of masquerade attacks on wireless sensor networks," in *Communications, 2007. ICC'07. IEEE International Conference on*. IEEE, 2007, pp. 1142–1147.

[27] V. S. Bhuse, "Lightweight intrusion detection: A second line of defense for unguarded wireless sensor networks," Ph.D. dissertation, Kalamazoo, MI, USA, 2007, aAI3263334.

[28] D. Huang, M. Mehta, D. Medhi, and L. Harn, "Location-aware key management scheme for wireless sensor networks," in *Proceedings of the 2nd ACM workshop on Security of ad hoc and sensor networks*. ACM, 2004, pp. 29–42.

[29] H. Chan, A. Perrig, and D. Song, "Random key predistribution schemes for sensor networks," in *Security and Privacy, 2003. Proceedings. 2003 Symposium on*. IEEE, 2003, pp. 197–213.

[30] M. Drozda, S. Schaust, and H. Szczerbicka, "Ais for misbehavior detection in wireless sensor networks: Performance and design principles," in *Evolutionary Computation, 2007. CEC 2007. IEEE Congress on*. IEEE, 2007, pp. 3719–3726.

[31] P. Levis, "TinyOS programming," 2006.

[32] "Telos Ultra low power IEEE 802.15.4 compliant wireless sensor module." [Online]. Available: http://www4.ncsu.edu/ kkolla/CSC714/datasheet.pdf

[33] "TinyECC: A Configurable Library for Elliptic Curve Cryptography in Wireless Sensor Networks," 2010 (accessed August 06, 2016). [Online]. Available: http://discovery.csc.ncsu.edu/software/TinyECC/

[34] P. E. Jones *et al.*, "US secure hash algorithm 1 (SHA1)," 2001. [Online]. Available: https://tools.ietf.org/html/rfc3174

[35] D. He, S. Chan, and M. Guizani, "Accountable and privacy-enhanced access control in wireless sensor networks," *Wireless Communications, IEEE Transactions on*, vol. 14, no. 1, pp. 389–398, Jan 2015.

[36] P. Nie, J. Vähä-Herttua, T. Aura, and A. Gurtov, "Performance analysis of hip diet exchange for wsn security establishment," in *Proceedings of the 7th ACM Symposium on QoS and Security for Wireless and Mobile Networks*, ser. Q2SWinet '11. New York, NY, USA: ACM, 2011, pp. 51–56.