Wireless Pers Commun (2016) 87:971–993
DOI 10.1007/s11277-015-2633-6

Malleability Resilient Concealed Data Aggregation in Wireless Sensor Networks

Keyur Parmar[1] · **Devesh C. Jinwala**[1]

Published online: 6 May 2015

Abstract The objective of concealed data aggregation is to achieve the privacy preservation at intermediate nodes while supporting in-network data aggregation. The need for privacy preservation at intermediate nodes and the need for data aggregation at intermediate nodes can be simultaneously realized using privacy homomorphism. Privacy homomorphism processes the encrypted data without decrypting them at intermediate nodes. However, privacy homomorphism is inherently malleable. Although malicious adversaries cannot view transmitted sensor readings, they can manipulate them. Hence, it is a formidable challenge to realize conflicting requirements, such as end-to-end privacy and end-to-end integrity, while performing en route aggregation. In this paper, we propose a malleability resilient concealed data aggregation protocol for protecting the network against active and passive adversaries. In addition, the proposed protocol protects the network against insider and outsider adversaries. The proposed protocol simultaneously realizes the conflicting objectives like privacy at intermediate nodes, end-to-end integrity, replay protection, and en route aggregation. As per our knowledge, the proposed solution is the first that achieves end-to-end security and en route aggregation of reverse multicast traffic in the presence of insider, as well as outsider adversaries.

Keywords Wireless sensor networks · Secure data aggregation · Concealed data aggregation · Privacy homomorphism · Non-malleable

Preliminary version of this paper is accepted at the 20th EUNICE/IFIP WG 6.2, 6.6 Workshop on Advances in Communication Networking, France (EUNICE 2014), for publication in Springer LNCS series. The present article is a reworked and extended version that includes major novel features such as replay protection, comprehensive analysis of models and assumptions, overhead analysis and security analysis.

✉ Keyur Parmar
keyur.mtech@gmail.com

[1] S. V. National Institute of Technology, Surat, India

1 Introduction

Wireless sensor network (WSN) contains a collection of tiny sensor devices [3, 52]. These devices [40, 41] are composed of very limited resources like battery power, memory, processor, bandwidth, etc. [3]. Among these resources, energy is the most limiting factor that has a profound effect on the lifetime of WSNs [4, 15]. Due to non-replenishable energy supply, numerous solutions [4, 53] have been proposed to improve energy efficiency across WSNs. Moreover, radio frequency operations consume far more energy than the CPU instructions [22]. Hence, the need to reduce communication traffic becomes indispensable. One of the techniques used for reducing communication traffic is "in-network processing", also known as "data aggregation" [15]. In-network processing aggregates sensor readings at intermediate nodes and forwards the cumulative result towards the base station. Such aggregation process helps in a scenario where instead of raw sensor readings, only derivatives such as sum, average, minimum, maximum etc., are required [15].

Security becomes another important design parameter for WSNs [11, 48, 61]. Hostile and unattended deployments, unreliable communication channel and lack of physical protection make WSNs vulnerable to a wide range of attacks [29, 48]. Moreover, the goal of data aggregation is to reduce communication traffic while security features add extra communication traffic. Hence, the need for simultaneous realization of these conflicting goals has led the development of secure data aggregation protocols. Secure data aggregation protocols achieve the following objectives together; namely, (1) data aggregation and (2) security. Secure data aggregation is often being classified as either hop-by-hop secure data aggregation or end-to-end secure data aggregation [44]. Hop-by-hop secure data aggregation considers that intermediate nodes are trustworthy. Intermediate nodes can decrypt raw sensor readings; aggregate them and forward encrypted results toward the base station. Though viable, such hop-by-hop aggregation becomes problematic if intermediate nodes are compromised. Compromised intermediate nodes can reveal aggregated data to adversaries that may have a catastrophic effect on the viability of WSNs in hostile environments. Hence, the need to ensure the privacy of sensor readings at intermediate nodes becomes an important security objective in data-centric networks [20, 62].

End-to-end secure data aggregation, also known as concealed data aggregation [18, 20, 62], protects the privacy of sensor readings while performing en route data aggregation. Privacy homomorphism [54] realizes these objectives by means of encrypted data processing. It can process the encrypted data without decrypting them at intermediate nodes. Although privacy homomorphism helps to protect sensor readings from passive attackers, it makes them susceptible to active attackers. Algorithms that support privacy homomorphism are inherently malleable [16]. As aggregator nodes do not require any secret information to aggregate data packets, any malicious node can inject fake data packets to falsify genuinely aggregated data. Traditional mechanisms used to provide message authentication/integrity, ensure that data cannot be altered on the way from leaf nodes to the base station. However in data-centric networks, data are supposed to be altered at every hop. There exist numerous solutions [45, 57] that combine end-to-end privacy with hop-by-hop message authentication. However, hop-by-hop message authentication [24, 45] only considers outsider adversaries, whereas, in WSNs, there exist malicious intermediate nodes that can successfully falsify sensor readings without being detected.

As data aggregation changes the original data en route, verifying the correctness of aggregated data becomes challenging [9]. Westhoff et al. [63] proposed a solution (malleability resilient (premium) concealed data aggregation—MR(P)CDA) that verifies integrity of sensor readings at the base station while ensuring privacy and data aggregation. They used a homomorphic MAC [2] for aggregating the message authentication codes (MACs) of corresponding data packets. Although it provides end-to-end integrity verification, it can only protect against outsider adversaries. When there exist malicious insider adversaries, it fails to detect tempered data packets. In addition, their protocol provides integrity verification only at the base station. Hence, maliciously injected data packets have to be forwarded up to the base station for integrity verification. Such redundant communication depletes sensor nodes' precious energy. Existing algorithms in concealed data aggregation provide integrity verification at intermediate nodes [9, 44] or at the base station [36, 58, 63]. However, there exists a need to verify the integrity at intermediate nodes as well as at the base station.

A replay protection is another vital security attribute for sensor networks. As sensor readings are aggregated en route, the verification of data freshness at each intermediate node becomes imperative. Hop-by-hop secure data aggregation protocols use a counter or a nonce to provide the replay protection [49]. However, end-to-end secure data aggregation protocols where sensor readings remain encrypted at intermediate nodes, cannot adapt such mechanisms directly. In addition, malicious intermediate nodes can be a threat against hop-by-hop replay protection techniques [49]. Thus, the need for verifying the data freshness before processing encrypted data, becomes important security objective.

In this paper, we propose a malleability resilient concealed data aggregation protocol for protecting privacy and integrity of sensor readings. As single authentication mechanism cannot provide integrity verification at intermediate nodes as well as at the base station, we use separate primitives for verifying the integrity at both these levels. The proposed protocol uses a symmetric-key based MAC for layer-wise integrity protection against outsider adversaries. In addition, it uses a homomorphic MAC for protecting the network against active insider adversaries. The major contribution of this work is the protection against insider and outsider adversaries as well as active and passive adversaries. As per our knowledge, the proposed solution is the first to achieve the end-to-end privacy and the end-to-end integrity of reverse multicast traffic when there exist insider and outsider adversaries. Moreover, the proposed solution verifies the data freshness before performing encrypted data processing at intermediate nodes.

The rest of the paper is organized as follows. Section 2 briefly describes the relevant literature. In Sect. 3, we provide an overview of homomorphic primitives used by the proposed protocol. In addition, we present a comprehensive analysis of models and assumptions. Section 4 presents the proposed protocol for malleability resilient concealed data aggregation in WSNs. The security analysis is discussed in Sect. 5. Section 6 analyzes the resource overhead. In Sect. 7, we conclude with emphasizing our contributions.

2 Related Work

Secure data aggregation is one of the well-researched topics in WSNs [44]. Initially, secure data aggregation protocols achieve security (confidentiality, message authentication (data integrity), replay protection etc.) in hop-by-hop manner [28, 35, 44, 55]. This approach for security is perfectly valid for traditional networks where machines are physically secure. However due to the hostile deployments, such approaches are not viable in WSNs. Compromised intermediate nodes can have a catastrophic effect on the security of WSNs

[20]. Hence, the focus of secure data aggregation has been shifted to concealed data aggregation, also known as, end-to-end secure data aggregation. Concealed data aggregation protocols [8, 20, 42, 59] protect the privacy of sensor readings at intermediate nodes and allow en route aggregation of reverse multicast traffic.

Rivest et al. [54] proposed a way to perform encrypted data processing and referred to it as "privacy homomorphism". Algorithms that support privacy homomorphism provide additive and multiplicative operations over encrypted data [16]. Moreover, privacy homomorphism has been used to perform homomorphic encryption [13, 31, 43, 46], homomorphic hash functions [17, 33], homomorphic MACs [2, 25] and homomorphic digital signatures [6, 27].

Concealed data aggregation protocols use privacy homomorphism to process encrypted sensor readings at intermediate nodes. Girao et al. [18, 20], first coined the term "concealed data aggregation" (CDA) with the intent of privacy protection at intermediate nodes. They used Domingo-Ferrer's privacy homomorphism [13] for computing over encrypted data at intermediate nodes. Chan et al. [10] formally defined concealed data aggregation and presented formal security proofs of CDA algorithms. The work related to concealed data aggregation is divided into two categories: (1) CDA using symmetric-key based privacy homomorphism techniques and (2) CDA using asymmetric-key based privacy homomorphism techniques.

Symmetric key based privacy homomorphism techniques [8, 13, 50] for concealed data aggregation are scarce compared to the public key based techniques. The major drawback of such techniques is the key management. Castelluccia et al. [7, 8] presented a technique based on the one time pad that handles a key management issue. It supports a separate key for each node in the network, unlike other symmetric-key based techniques [20, 50]. However, it requires nodes' identity related information in order to perform decryption. Hence, costly identity transfer limits its usage for resource-constrained environments.

Asymmetric-key based techniques are initially considered as an expensive alternative for resource constrained devices like sensor nodes [40, 41]. However, Gura et al. [21] and Wander et al. [60], implemented the asymmetric-key based techniques including those based on elliptic curve cryptography (ECC), on 8-bit micro-controllers, generally used for sensor nodes [40]. Performance results presented by them indicate that public key cryptography is viable on resource constrained devices even if it is implemented on software. Authors in [39, 42], comparatively evaluated the applicability of public key based homomorphic cryptosystems for WSNs. They pointed out that the elliptic curve ElGamal (EC-ElGamal) cryptosystem [31] is a good choice in a situation where costly decryption is carried out at the base station. Elliptic curve cryptosystems provide the same level of security as asymmetric-key based cryptosystems with reduced key size. As shown by Koblitz et al. [32], elliptic curve cryptosystems with 160 bit key size offers approximately the same level of security as RSA with 1024 bit key size.

Any wireless network requires an authentication mechanism. The use of an encryption algorithm without having an authentication mechanism is proven to be insecure [28]. There have been numerous solutions [5, 9, 24, 25, 63] that provide message authentication in WSNs. They are generally based on hash functions, MACs or digital signatures. Although traditional message authentication mechanisms protect sensor readings from outsider adversaries, they fail to protect them against compromised aggregator nodes. Hence, the need for end-to-end integrity verification becomes obvious due to encrypted data processing and en route aggregation. The end-to-end integrity, in the presence of insider adversaries, is considered to be an open problem. Chan et al. [9] discussed challenges that provide end-to-end integrity along with end-to-end privacy in secure data aggregation scenarios. Agrawal et al. [2], proposed a homomorphic MAC to provide end-to-end integrity verification in network coding systems. Recently, the homomorphic MAC based solutions [25, 63] are proposed to provide integrity protection. Westhoff et al. used [63] the homomorphic MAC to provide end-to-end integrity verification in sensor networks. They used the homomorphic MAC [2] for aggregating the MACs of corresponding data packets.

In sensor networks, MAC size is considerably higher than raw sensor readings. The raw sensor readings generally require less than three bytes for representation (e.g., temperature sensor requires only 1 byte). In traditional networks, 8 or 16 byte MAC is preferred for a reasonable amount of security. In addition, the security strength of a MAC algorithm is dependent on its length and underling encryption algorithm. However, the first security architecture for WSNs, TinySec [28], uses a 4 byte MAC. The reasons to reduce MAC length are as follows: (1) As the default packet size in TinyOS [34] is 36 bytes, using an 8 or a 16 byte MAC is not affordable due to high communication cost. (2) Moreover, the Mica2 mote with a CC1000 radio takes twenty months to forge a 4-byte MAC [28]. Motes with a radio CC2420 (MicaZ [40] or TelosB [41]) can forge the same MAC tag in 3 months. However, if the key used to produce the MAC is changed within a period of 3 months, such attacks can be mitigated.

3 Preliminaries

In this section, we describe privacy homomorphism and two different homomorphic primitives, namely, homomorphic encryption and homomorphic MAC. An EC-ElGamal cryptosystem [31] is used for processing encrypted data while a homomorphic MAC algorithm [2] is used for aggregating the MACs of corresponding data packets.

3.1 Homomorphic Primitives

Rivest et al., in their seminal paper [54], introduce the need for encrypted data processing and a way to realize it using cryptography. They referred to it as "privacy homomorphism". Privacy homomorphism processes the encrypted data in the same way as otherwise it is being processed in a raw form. In addition, it requires a special encryption function that supports encrypted data processing. Encryption functions can perform an additive and multiplicative operations over encrypted data. Formally, we define privacy homomorphism as follows:

Formal definition $\mathcal{E}_{\mathcal{K}'}(\cdot)$ is an encryption function and $\mathcal{D}_{\mathcal{K}''}(\cdot)$ is a corresponding decryption function. In symmetric-key based cryptosystems, keys are identical, $\mathcal{K}' = \mathcal{K}''$. However, public key based cryptosystems use different keys, $\mathcal{K}' \neq \mathcal{K}''$, for encryption $\mathcal{E}(\cdot)$ and corresponding decryption $\mathcal{D}(\cdot)$.

Given an encryption of m_1, $\mathcal{E}(m_1)$, and an encryption of m_2, $\mathcal{E}(m_2)$, we can efficiently compute[1]:

$$\mathcal{E}_{\mathcal{K}'}(m_1) \oplus \mathcal{E}_{\mathcal{K}'}(m_2) = \mathcal{E}_{\mathcal{K}'}(m_1 \otimes m_2) \tag{1}$$

[1] For probabilistic cryptosystems, decryption must be performed before comparison.

As shown in Eq. 1, the order of operands used in the encryption process will not have any impact on the decrypted result. In the above-mentioned cases, the decryption process yields the same result. This property is known as “privacy homomorphism”. The operators $\oplus$ and $\otimes$ can remain same as in the case of Domingo-Ferrer’s cryptosystem [13] (Eq. 2) or they can be different as in the case of Paillier et al.’s [46] cryptosystem (Eq. 3).

$$\mathcal{D}_{\mathcal{K}'}(\mathcal{E}_{\mathcal{K}'}(m_1) + \mathcal{E}_{\mathcal{K}'}(m_2)) \bmod \mathtt{n} = \mathcal{D}_{\mathcal{K}'}(\mathcal{E}_{\mathcal{K}'}(\mathtt{m}_1 + \mathtt{m}_2)) \bmod \mathtt{n} \tag{2}$$

$$\mathcal{D}_{\mathcal{K}''}(\mathcal{E}_{\mathcal{K}'}(m_1) \times \mathcal{E}_{\mathcal{K}'}(m_2)) \bmod \mathtt{n}^2 = \mathcal{D}_{\mathcal{K}''}(\mathcal{E}_{\mathcal{K}'}(\mathtt{m}_1 + \mathtt{m}_2)) \bmod \mathtt{n} \tag{3}$$

As shown in Eq. 2, Domingo-Ferrer’s symmetric-key based cryptosystem [13] requires the same key $\mathcal{K}^{'}$ for encryption and corresponding decryption. However, as shown in Eq. 3, Paillier’s [46] asymmetric-key based cryptosystem uses different keys, $\mathcal{K}^{'}$ and $\mathcal{K}^{''}$ for encryption and its corresponding decryption.

Majority of WSNs applications require computing functions like MIN, MAX, SUM, AVG, movement detection, over sensor readings. Only additive privacy homomorphism can support these functions [7, 62]. Although numerous cryptosystems support additive privacy homomorphism [8, 13, 31, 42], comparative evaluation by Mykletun et al. [42] suggests that EC-ElGamal requires fewer resources than other asymmetric-key and ECC based cryptosystems.

3.1.1 Homomorphic Encryption

We describe the homomorphic primitives used to provide the encrypted data processing and end-to-end message authentication. The EC-ElGamal [31] is an elliptic curve based public key cryptosystem with support for additive privacy homomorphism. The EC-ElGamal is used to protect the privacy, while the Homomorphic MAC [2] is used to verify the authenticity of aggregated ciphertext.

Elliptic Curve ElGamal Cryptosystem (EC-ElGamal)

Key Generation $\mathcal{K}$:

1. Select a large prime p
2. Choose an elliptic curve E over $\mathbb{F}_p$
3. Choose a point P on $E(\mathbb{F}_p)$
4. Randomly choose a secret s_k and compute, $p_k = s_k \cdot P$ over $E(\mathbb{F}_p)$

Encryption $\mathcal{E}$:

1. A plaintext, $m \in E(\mathbb{F}_p)$ and a random integer r
2. Compute ciphertexts $c_1 = r \cdot P$ and $c_2 = m + r \cdot p_k$

Decryption $\mathcal{D}$:

1. Decrypt the ciphertexts, $\mathcal{D}(c) = c_2 - s_k \cdot c_1 = m$

Ciphertexts Aggregation $\mathcal{A}$:

1. Given an encryption of m_1, $\mathcal{E}(m_1)$ as ciphertexts, c_{11} and c_{12}
2. Given an encryption of m_2, $\mathcal{E}(m_2)$ as ciphertexts, c_{21} and c_{22}
3. Compute aggregated ciphertexts, $c_1 = c_{11} + c_{21}$ and $c_2 = c_{12} + c_{22}$
4. Decryption of c_1 and c_2 gives an aggregated plaintext $m_1 + m_2$ on $E(\mathbb{F}_p)$

As shown by Mykletun et al. [42], EC-ElGamal consumes fewer resources compared to other asymmetric-key based or ECC-based techniques. A message expansion ratio of EC-ElGamal is fewer than the existing public key based additive homomorphic cryptosystems [42]. However, there are some practical difficulties in implementing EC-ElGamal for resource-constrained devices. They are as follows:

- In EC-ElGamal cryptosystem [31], homomorphic operations are performed over elliptic curve points. Therefore, a plaintext m needs to be represented as a point $P \in E(\mathbb{F}_p)$. However, it is not easy to map a plaintext value to a point on the elliptic curve $E(\mathbb{F}_p)$. In addition, such mapping function needs to be deterministic in order to ensure the correctness of deciphered values. Although numerous mapping function can transform a plaintext m to the corresponding elliptic curve point $P \in E(\mathbb{F}_p)$, and vice versa, we require a mapping function that is additively homomorphic, i.e. $\mathtt{map}(\mathtt{m}_1 + \mathtt{m}_2) = \mathtt{map}(\mathtt{m}_1) + \mathtt{map}(\mathtt{m}_2)$. Adler et al. [1], Ugus et al. [59] and, Mykletun et al. [42] describe a homomorphic mapping function that transform a plaintext m to an elliptic curve point P and vice versa.
- The ElGamal cryptosystem [14] has a 2-to-1 message expansion ratio, while the EC-ElGamal [31] has a 4-to-1 message expansion ratio. The increase in message expansion ratio is due to the transformation of a plaintext $m \in \mathbb{F}_p$ into a point $P \in E(\mathbb{F}_p)$. As each point on the elliptic curve $E(\mathbb{F}_p)$, has two coordinates, $(x, y) \in \mathbb{F}_p$, it increases the message expansion ratio by a factor of 2. In addition, a plaintext value m is converted to the ciphertexts (c_1, c_2) in the ElGamal and EC-ElGamal cryptosystem that increases a message expansion ratio by a factor of 2. As shown in Hoffstein et al. [23], the point compression techniques can reduce the message expansion of EC-ElGamal. The point compression techniques transfer only a single coordinate value $x \in \mathbb{F}_p$ of a point $P \in E(\mathbb{F}_p)$ and a single bit representing y coordinate. Hence, the ciphertext size of EC-ElGamal cryptosystem remains nearly equal as compared to the ciphertext size of the ElGamal cryptosystem.

3.1.2 Homomorphic Message Authentication Code

In sensor networks, data aggregation has been used to reduce the communication overhead. However, security attributes such as MACs cannot be aggregated using data aggregation techniques. Moreover, the size of MAC is comparatively much larger than the size of original sensor readings (e.g., temperature). Hence, there exists a need to aggregate MAC en route for reducing the communication overhead. Aggregate MAC based techniques [9, 12, 30] can aggregate MACs of corresponding data packets. However, they require the original data packets to verify the integrity. Hence, they cannot be viable for scenarios where data are aggregated en route. Therefore, we need a mechanism that helps in verifying the integrity of an aggregated data with minimal communication overhead. Homomorphic MAC [2] helps in verifying the integrity of aggregated data in data-centric networks. Although there exist homomorphic digital signatures [6, 27], they are not viable for resource-constrained devices due to their excessive computation and communication cost.

Homomorphic MAC [2] consists of three probabilistic, polynomial-time algorithms, namely, $\mathcal{Generation}$, $\mathcal{Aggregation}$, and $\mathcal{Verification}$. Initially, the message m is divided into a sequence of vectors $v_1, v_2, \ldots, v_m$. The $\mathcal{Generation}$ algorithm generates a tag $\mathcal{T}$ for each vector $v_1, v_2, \ldots, v_m \in \mathbb{F}_q^{n+m}$ of vector space V in an n-dimensional linear space $\mathbb{F}_q^n$. Here, n, m, q are fixed and known before the sensor nodes' deployment. The $\mathcal{Aggregation}$

algorithm aggregates the MAC tags. Here, the $\mathcal{A}$*ggregation* algorithm needs to be homomorphic in order to verify the integrity of aggregated messages. The $\mathcal{V}$*erification* algorithm verifies the integrity of the aggregated messages using the vector(v)-tag($\mathcal{T}$) pairs. In this section, we briefly present a homomorphic MAC algorithm [2]. The relevant security proofs of the algorithm can be found in [2].

Homomorphic MAC

- Let a pseudo random generator $G : \mathcal{K}_G \rightarrow \mathbb{F}_q^{n+m}$.
- Let a pseudo random function $F : \mathcal{K}_F \times (\mathcal{I} \times [m]) \rightarrow \mathbb{F}_q$.
- `id` is an identifier of the vector space V
- Let $k_1 \in \mathcal{K}_G$ and $k_2 \in \mathcal{K}_F$ are the keys used for the MAC construction.
- Let $\alpha_1, \alpha_2, \cdots, \alpha_m \in \mathbb{F}_q$ are the coefficients that produce v as a linear combination of the original message vectors.

$\mathcal{G}$**eneration**: To generate a MAC tag $\mathcal{T}$, for an i^{th} basis vector from the vectors $v_1, v_2, \cdots, v_m \in \mathbb{F}_q^{n+m}$ of a vector space V, using a key pair $k = (k_1, k_2)$, do:

1. $u \leftarrow G(k_1) \in \mathbb{F}_q^{n+m}$
2. $b \leftarrow F(k_2, (id, i)) \in \mathbb{F}_q$
3. $\mathcal{T} \leftarrow (u \cdot v) + b \in \mathbb{F}_q$

Here, $\mathcal{T} \in \mathbb{F}_q$ is a MAC tag computed over the vector $v_i \in \mathbb{F}_q^{n+m}$.

$\mathcal{A}$**ggregation**: Given (v_1, t_1, α_1), ..., (v_m, t_m, α_m), compute,

$$\mathcal{T} \leftarrow \sum_{j=1}^{m} \alpha_j \mathcal{T}_j \in \mathbb{F}_q$$

$\mathcal{V}$**erification**: Given a key pair $k = (k_1, k_2)$ and $y = (y_1, \cdots, y_{n+m}) \in \mathbb{F}_q^{n+m}$, do:

- $u \leftarrow G(k_1) \in \mathbb{F}_q^{n+m}$ and $a \leftarrow (u \cdot y) \in \mathbb{F}_q$
- $b \leftarrow \sum_{i=1}^{m} [y_{n+i} \cdot F(k_2, (id, i))] \in \mathbb{F}_q$
- If $a + b = \mathcal{T}$ then output 1; otherwise output 0

The correctness of homomorphic MAC can be verified as follows: Given $y = \sum_{i=1}^{m} \alpha_i v_i$, where $v_1, v_2, \ldots, v_m$ are the augmented basis vectors and $t_1, t_2, \ldots, t_m$ are the corresponding MAC tags. The coordinates of y and coefficients $(\alpha_1, \alpha_2, \ldots, \alpha_m)$ are equal. Hence, as per the $\mathcal{V}$erification algorithm,

$$a + b = u \cdot y + b = \sum_{i=1}^{m} \alpha_i \cdot ((u \cdot v_i) + F(k_2, (id, i))) = \sum_{i=1}^{m} \alpha_i \cdot t_i = \mathcal{T}$$

4 The Proposed Protocol

Concealed data aggregation using inherently malleable privacy homomorphism, makes sensor readings vulnerable against active attackers. The privacy homomorphism not only helps genuine aggregator nodes in aggregating the ciphertexts, but it also helps the malicious adversaries in modifying the ciphertexts. Therefore, there exists a need to verify the integrity of sensor readings before processing them at intermediate nodes. In concealed data aggregation scenarios, the end-to-end integrity verification requires the fulfillment of the following conditions.

- Each intermediate node has to verify the integrity of the original sensor readings.
- Each intermediate node has to verify the integrity of aggregated sensor readings, forwarded by its child nodes.
- The base station has to verify the integrity of the received aggregated data.
- The base station has to verify the correctness of aggregated data in order to ensure that the aggregated data is the correct representation of the original sensor readings.

The fulfillment of the above-mentioned conditions not only preserves the integrity of sensor readings, it helps in reducing the extra communication traffic by detecting the maliciously aggregated packets nearer to their sources. Existing secure data aggregation algorithms either provide integrity verification at intermediate nodes [37, 49] or at the base station [63, 64]. However, there exists a need to verify the integrity at intermediate nodes and second at the base station. A single authentication mechanism cannot provide integrity verification at both the levels. Hence, there exists a need to use separate mechanisms for providing integrity verification at the base station and at the intermediate nodes. In this paper, we propose the use of separate primitives for integrity verification at intermediate nodes as well as at the base station. In the proposed protocol, we use a pairwise symmetric-key for providing the integrity protection at intermediate nodes. In addition, the use of homomorphic MAC helps in verifying the integrity of the processed sensor readings at the base station.

Westhoff et al. [63] proposed a way for verifying the integrity of the sensor readings at the base station when there exists only outsider adversaries. They have used the homomorphic MAC [2] for verifying the integrity of aggregated sensor readings. However, it provides the integrity verification at the base station only. Hence, a late detection of maliciously aggregated data packets can have a serious impact on the precious sensor node's energy when there exist malicious adversaries. In addition, if there exist malicious insider adversaries, it cannot correctly verify the aggregated sensor readings.

The data freshness also becomes an important security primitive for wireless sensor networks. In traditional networks, the replay protection is only being required against outsider adversaries. However, when the data are aggregated en route, there exists a need to provide the replay protection against insider and outsider adversaries. A detection of the unauthorized aggregation of reused data packets by an active insider adversary becomes a formidable challenge for concealed data aggregation scenarios.

In this section, we present the proposed protocol for malleability resilient concealed data aggregation in WSNs. The proposed protocol uses an EC-ElGamal cryptosystem [31] and a homomorphic MAC algorithm [2]. The proposed protocol protects the confidentiality against outsider adversaries, privacy of sensor readings against insider adversaries, data integrity against insider and outsider adversaries as well as at intermediate nodes and at the base station, and the data freshness against insider and outsider adversaries. Table 1 describes the notations used in the proposed protocol.

Table 1 Notations used in the proposed protocol

Symbol	Description
i	Sensor node ID
PRF	Pseudo-random function
$Cr_{i,j}$	Counter generated at each node i using a PRF and a pair-wise secret key shared with a node j
n	Total number of nodes in the network
S_i	Plaintext value sensed by a sensor node i
C_i	Ciphertext value computed by a sensor node i
E	Encryption algorithm
D	Decryption algorithm
p_k	Public key of the base station, generated using EC-ElGamal cryptosystem
s_k	Private key of the base station, generated using EC-ElGamal cryptosystem
$k = \{k_1, k_2\}$	A symmetric-key generated using Homomorphic MAC algorithm
$H - MAC$	Homomorphic MAC generated using a key k, shared between the base station and sensor node(s)
$\mathcal{T}_i$	A homomorphic MAC tag produced by a sensor node i
m	Distance representing the number of hops from a sensor node to its parent node(s)
j_m	A parent node of a node at a distance of m hops
t	Distance representing the number of hops between a node and the base station
E_{i,j_m}	Encryption using a symmetric-key cipher and a key shared between a node i and its parent node j_m
D_{i,j_m}	Decryption using a symmetric-key cipher and a key shared between a node i and its parent node j_m
x	Number of child nodes of a node
$\oplus$	Operator representing additive homomorphic operation (Encryption and MAC)

Malleability resilient concealed data aggregation protocol

- *Bootstrapping phase*

1. The base station generates a key pair (p_k, s_k), using the EC-ElGamal cryptosystem.
2. The base station shares a symmetric-key-pair $k = \{k_1, k_2\}$ with the leaf nodes.
3. Each sensor node i shares a pair-wise symmetric-key, $\{i,j\}$ with its neighboring nodes j, as well as a pair-wise symmetric-key, $\{i, j_m\}$ with the node(s) at m hops away, j_m.

- *At leaf nodes*

4. Each leaf (sensor) node i senses a value S_i and encrypts it using the public key of EC-ElGamal cryptosystem.

$$C_i = E_{p_k}(S_i) \quad \forall i \in \{1, 2, \ldots, n\}$$

5. A node i computes a homomorphic MAC tag: $\mathcal{T}_i \leftarrow MAC(C_i)$. Here, the homomorphic MAC tag is generated over a ciphertext unlike traditional networks.
6. A node i generates a counter Cr using a PRF and a pair-wise symmetric-key(s) shared with its neighboring nodes.

7. A node i encrypts a homomorphic MAC tag $\mathcal{T}_i$ and a counter Cr using a key shared with a node j_m which is mth-hop parent for node i. Here, an encryption algorithm can be any symmetric-key-based algorithm like the advanced encryption standard (AES), RC5, etc. The privacy homomorphism property is not required for this encryption.

   - $E_{i,j_m}(\mathcal{T}_i \quad || \quad Cr_{i,j_m})$
     $\mathtt{Here}, \forall \mathtt{i}, \mathtt{j_m} \in \{1, 2, \ldots, \mathtt{n}\}, \ \mathtt{i} \neq \mathtt{j_m}, \ \forall \mathtt{m} \in \{1, 2, \ldots, \mathtt{t}\}$

   Here, t is a number of hops distance between a leaf node and the base station.

8. A node i transmits,

   - $C_i \quad \forall i \in \{1, 2, \ldots, n\}$
   - $E_{i,j_m}(\mathcal{T}_i \quad || \quad Cr_{i,j_m})$
     $\mathtt{Here}, \forall \mathtt{i}, \mathtt{j_m} \in \{1, 2, \ldots, \mathtt{n}\}, \ \mathtt{i} \neq \mathtt{j_m}, \ \forall \mathtt{m} \in \{1, 2, \ldots, \mathtt{t}\}$

- *At intermediate nodes*

9. Each intermediate node j receives,

   - $C_i \quad \forall i \in \{1, 2, \ldots, n\}$
   - $E_{i,j_m}(\mathcal{T}_i \quad || \quad Cr_{i,j_m})$
     $\mathtt{Here}, \forall \mathtt{i}, \mathtt{j_m} \in \{1, 2, \ldots, \mathtt{n}\}, \ \mathtt{i} \neq \mathtt{j_m}, \ \forall \mathtt{m} \in \{1, 2, \ldots, \mathtt{t}\}$

10. A node j_m, m-hops away from node i, decrypts $E_{i,j_m}(\mathcal{T}_i \quad || \quad Cr_{i,j_m})$ using its pairwise secret key with a node i. Number of child nodes of a node j is denoted by x, where $1 \leq x \leq n-1$.

$$D_{i,j_m}(\oplus_{i=1}^{x} \mathcal{T}_i \quad || \quad Cr_{i,j_m})$$

    Here, $\forall i, j_m \in \{1, 2, \ldots, n\}$, $i \neq j_m$, $\forall m \in \{1, 2, \ldots, t\}$, $1 \leq x \leq n-1$

11. A node j_m generates a counter using a PRF and a secret key shared with a node i. It compares it with the received counter Cr_{i,j_m}. If both counters are equal, it proves data freshness, else a node j_m drops the packet.
12. A node j_m generates a MAC tag over a ciphertext C_i using its shared secret key with a node i and compares it with the decrypted MAC tag computed in the previous step.

$$D_{i,j_m}(\mathcal{T}_i) \stackrel{?}{=} MAC(C_i)$$

    Here, $\forall i, j_m \in \{1, 2, \ldots, n\}$, $i \neq j_m$, $\forall m \in \{1, 2, \ldots, t\}$, $1 \leq x \leq n-1$. If it holds, it accepts the ciphertext C_i
13. If all the ciphertexts, $(1, 2, \ldots, x)$, coming from child nodes are valid, a node j_m uses the EC-ElGamal cryptosystem and the Homomorphic MAC algorithm to aggregate the ciphertexts of its child nodes and their corresponding MACs.

$$C_{j_m} = \oplus_{i=1}^{x} C_i \quad 1 \leq x \leq n-1$$
$$\mathcal{T}_{j_m} = \oplus_{i=1}^{x} \mathcal{T}_i \quad 1 \leq x \leq n-1$$

14. A node j_m encrypts a newly generated MAC with a key shared with its parent node(s) which is/are up to m-hops away and repeats the steps 6–12.

- *At the base station*

15. The base station follows the steps 9, 10 and 11 to verify a counter and a homomorphic MAC tags. The base station also verifies a homomorphic MAC tag and its corresponding ciphertext using its private keys.

$$D_{s_k}(\oplus_{i=1}^{n-1} C_i) = \oplus_{i=1}^{n-1} S_i \tag{4}$$

4.1 Example

As shown in Fig. 1, we consider the reverse multi-cast communication from the leaf nodes towards the base station. For $m = 2$, each leaf node shares a unique pair-wise symmetric-key with its parent node as well as its grand-parent node. If we consider $m = 3$, a node shares a pair-wise key with its first hop parent, second hop parent and third hop parent. Here, the proposed approach requires the number of hops m to be greater than or equal to two. The value of m depends on the chances of collaborated and active insider adversaries. However, if there aren't any collaborated adversaries, a small value of m can provide the adequate security.

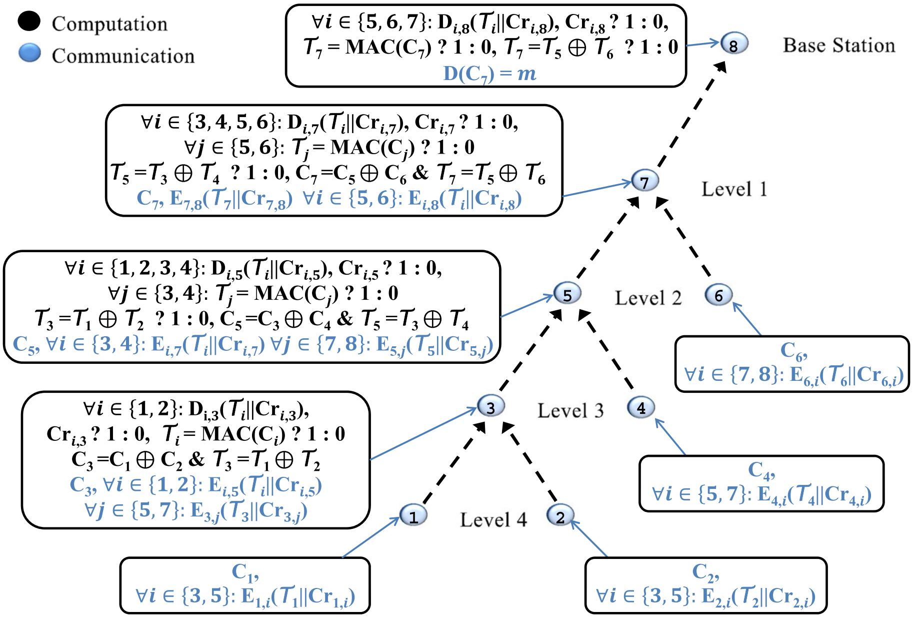

● Computation
● Communication
$\forall i \in \{5,6,7\}$: $D_{i,8}(\mathcal{T}_i \| Cr_{i,8})$, $Cr_{i,8}$? 1 : 0,
$\mathcal{T}_7 = MAC(C_7)$? 1 : 0, $\mathcal{T}_7 = \mathcal{T}_5 \oplus \mathcal{T}_6$? 1 : 0
$D(C_7) = m$
8
Base Station
$\forall i \in \{3,4,5,6\}$: $D_{i,7}(\mathcal{T}_i \| Cr_{i,7})$, $Cr_{i,7}$? 1 : 0,
$\forall j \in \{5,6\}$: $\mathcal{T}_j = MAC(C_j)$? 1 : 0
$\mathcal{T}_5 = \mathcal{T}_3 \oplus \mathcal{T}_4$? 1 : 0, $C_7 = C_5 \oplus C_6$ & $\mathcal{T}_7 = \mathcal{T}_5 \oplus \mathcal{T}_6$
C_7, $E_{7,8}(\mathcal{T}_7 \| Cr_{7,8})$ $\forall i \in \{5,6\}$: $E_{i,8}(\mathcal{T}_i \| Cr_{i,8})$
7
Level 1
$\forall i \in \{1,2,3,4\}$: $D_{i,5}(\mathcal{T}_i \| Cr_{i,5})$, $Cr_{i,5}$? 1 : 0,
$\forall j \in \{3,4\}$: $\mathcal{T}_j = MAC(C_j)$? 1 : 0
$\mathcal{T}_3 = \mathcal{T}_1 \oplus \mathcal{T}_2$? 1 : 0, $C_5 = C_3 \oplus C_4$ & $\mathcal{T}_5 = \mathcal{T}_3 \oplus \mathcal{T}_4$
C_5, $\forall i \in \{3,4\}$: $E_{i,7}(\mathcal{T}_i \| Cr_{i,7})$ $\forall j \in \{7,8\}$: $E_{5,j}(\mathcal{T}_5 \| Cr_{5,j})$
5
Level 2
6
C_6,
$\forall i \in \{7,8\}$: $E_{6,i}(\mathcal{T}_6 \| Cr_{6,i})$
$\forall i \in \{1,2\}$: $D_{i,3}(\mathcal{T}_i \| Cr_{i,3})$,
$Cr_{i,3}$? 1 : 0, $\mathcal{T}_i = MAC(C_i)$? 1 : 0
$C_3 = C_1 \oplus C_2$ & $\mathcal{T}_3 = \mathcal{T}_1 \oplus \mathcal{T}_2$
C_3, $\forall i \in \{1,2\}$: $E_{i,5}(\mathcal{T}_i \| Cr_{i,5})$
$\forall j \in \{5,7\}$: $E_{3,j}(\mathcal{T}_3 \| Cr_{3,j})$
3
Level 3
4
C_4,
$\forall i \in \{5,7\}$: $E_{4,i}(\mathcal{T}_4 \| Cr_{4,i})$
1
Level 4
2
C_1,
$\forall i \in \{3,5\}$: $E_{1,i}(\mathcal{T}_1 \| Cr_{1,i})$
C_2,
$\forall i \in \{3,5\}$: $E_{2,i}(\mathcal{T}_2 \| Cr_{2,i})$

Fig. 1 Malleability resilient concealed data aggregation (m = 2)

5 Overhead Analysis

In this section, we compare the communication overhead of the proposed protocol with five different approaches: (1) No Aggregation (2) Concatenation (3) Hop-by-hop secure data aggregation (SDA) (4) End-to-end secure data aggregation (CMT cryptosystem) [7, 8] (5) MR(P)CDA approach [63]. For the ease of comparison, we consider the same network model as described by Castelluccia et al. [7, 8].

5.1 Network Model

In this section, we consider a ternary tree based topology as discussed by Castelluccia et al. [7]. Although we consider the same ternary tree topology for ease of a comparison, we can seamlessly apply the proposed protocol to any n-ary tree topology or cluster based topology.

We consider a balanced ternary tree topology with a single base station and multitude of sensor nodes. Moreover, we assume that leaf nodes only perform the sensing operations while the intermediate nodes only perform the aggregation and forwarding operations. Here, we assume that the sensor readings require 7 bits for representation (e.g., Temperature ranges from 9 to 127 F). Moreover, we consider the bandwidth consumption of the nodes at different levels in the hierarchy. We consider a standard packet format of the TinyOS operating system [34], where a packet header (HDR) includes 56 bits, and the payload includes 232 bits.

5.2 Communication Overhead

In a no-aggregation scenario, an HDR well as a payload are forwarded without performing aggregation at intermediate nodes. Hence, as shown in Fig. 2, the communication overhead increases when data move upward in the hierarchy. For concatenation based scenarios, payloads are concatenated together to reduce the extra overhead of individual packet headers. However, the reduction achieved through the concatenation of payload is negligible when compared to the data aggregation based scenarios.

In a hop-by-hop secure data aggregation scenario (Fig. 3), the total number of bits transmitted by a node is HDR $+ \, log_2(t)$. Here, t is a range of possible sensor measurements. In this case, we use $t = 7$ bits to represent 127 different temperature values. Here, each leaf node in a hop-by-hop secure data aggregation approach has to transmit $56 + 7 = 63$ bits toward the next hop. At aggregator nodes, the number of bits required to be

Fig. 2 Communication overhead

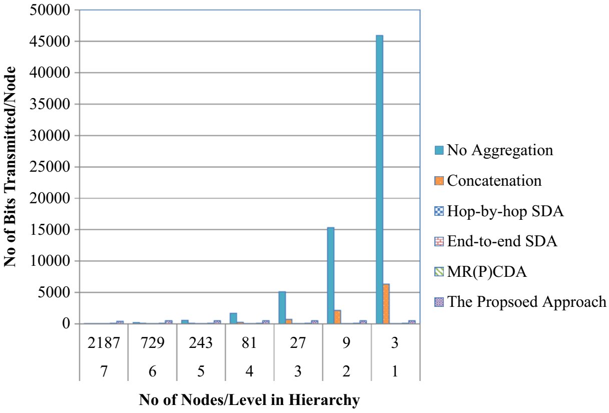
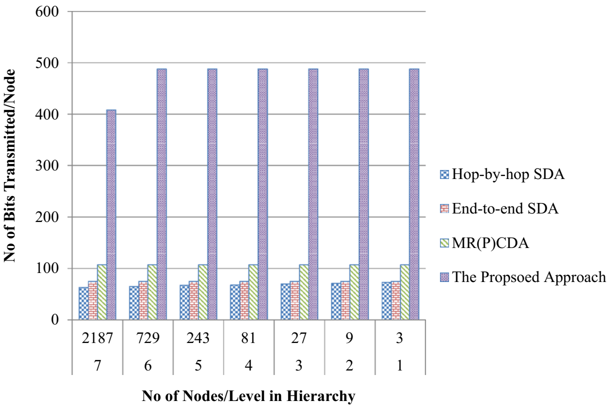


Fig. 3 Communication overhead (data aggregation)

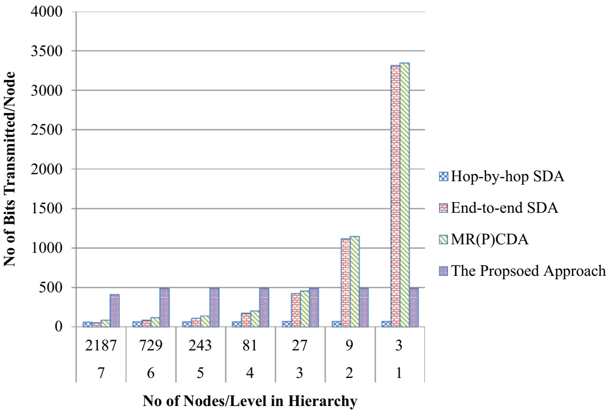


transmitted is $log_2(n' \cdot t)$. Here, n' represents the aggregation of child nodes' readings. Hop-by-hop secure data aggregation reduces the communication overhead drastically. However, it increases the security vulnerabilities. The aggregated sensor readings are comparatively more vulnerable and have a wider impact on the performance of WSNs. End-to-end secure data aggregation protocols ensure privacy while performing the en route aggregation. However, as shown in Fig. 3, addition of a security feature increases the communication overhead compared to the hop-by-hop secure data aggregation approach. However, the total number of bits transmitted by the CMT cryptosystem [7, 8] (end-to-end secure data aggregation) remains constant. The number of bits transmitted by the CMT cryptosystem depends on the modulus M. The total number of bits transmitted by the CMT cryptosystem is HDR $+ \; log_2(n) + log_2(t)$. Here, n represents the total number of nodes in the network, and t represents a range of sensor measurements.

In an MR(P)CDA approach [63], the total number of bits increases due to the added security feature for integrity protection at the base station. Here, we consider a 4-byte homomorphic MAC for calculation. The similar MAC size have been used by TinySec [28], the first security architecture for sensor networks. The authors of TinySec have validated the use of 4-byte MAC and proved that it provides the adequate security strength for WSNs scenarios. In addition, the authors in an MR(P)CDA approach [63] consider the same CMT cryptosystem [7, 8] for encryption. Hence, the total number of bits transmitted by an MR(P)CDA approach is HDR $+ \; log_2(n) + log_2(t) + 32$.

As shown in Fig. 3, the communication overhead of the proposed protocol is considerably high compared to the other secure data aggregation protocols. However, the proposed protocol achieves much stronger security strength compared to these protocols. The increase in communication overhead is due to the added security features. In the proposed protocol, we use EC-ElGamal cryptosystem [31] for performing the encryption of sensor readings. Moreover, elliptic curve cryptosystems require 160 bit parameter size for achieving the same level of security as provided by 1024 bit RSA. Although the EC-ElGamal cryptosystem has a message expansion ratio of 4-to-1, the total number of bits required to represent the ciphertext is fewer than asymmetric-key based cryptosystems. In addition, the ciphertext size of EC-ElGamal can be reduced using the point compression techniques [23]. To measure the bandwidth consumption of the proposed protocol, we consider an elliptic curve with the 163 bits of parameter size. In addition, we use the same 4 byte homomorphic MAC to provide the integrity protection. The proposed protocol requires 2 additional bytes to store the counter value. Hence, the total number of bits

Fig. 4 Communication overhead (30 % non-responding nodes)



required by the proposed protocol becomes nearly 408 bits at leaf nodes and 488 bits at intermediate nodes.

Although the communication overhead of the proposed protocol is higher in Fig. 3, when we consider the 30 %—non-responding nodes like Castelluccia et al. [7, 8], the constant overhead incurred by the proposed protocol outperforms the other secure data aggregation protocols (Fig. 4). The performance of the CMT cryptosystem [7, 8] as well as the Westhoff et al.'s cryptosytem [63] degrades significantly when there exist non-responding nodes. The reason for the degradation in performance is due to the requirement of transferring the identity information of non-responding nodes. Here, the constant overhead of the proposed protocol achieves the desired security objectives along with reduced communication overhead. Moreover, the proposed protocol's early verification of maliciously injected data packets nearer to their sources can also help to reduce the communication overhead.

Along with communication cost, computation cost also affects the performance of resource-constrained WSNs. The sensor nodes [40, 41] not only have limited bandwidth and energy, but they also have limited memory and processing capabilities. Therefore, the implementation of asymmetric key based techniques as mentioned in the proposed protocol are always under scrutiny. However, as shown in Malan et al. [38] and Gura et al. [21], the efficient implementation of asymmetric key based techniques are viable for resource-constrained sensor nodes. In addition, as shown by Hill et al. [22], transmission of a single bit requires the same amount of energy as computing 1000 CPU instructions. Therefore, we assume that the implementation of the proposed protocol is feasible, and the impact of computation on the energy is negligible compared to the impact of communication.

6 Security Analysis

In this section, we discuss an adversary model and the assumptions regarding the adversary's ability to launch various attacks, and the security strength of the proposed protocol against well-known cryptographic attacks.

6.1 Adversary Model

In this paper, we consider two types of adversaries, the passive adversary and the active adversary.

- *Passive adversary* The goal of a passive adversary is to eavesdrop the communication for deducing a key or any meaningful information. In WSNs, a wireless communication medium increases the risk of passive adversaries. In addition, an adversary may analyze the traffic patterns to disrupt the routing or to identify the nodes for launching active attacks. In this paper, we assume an adversary whose goal is to breach the confidentiality and privacy of sensor readings. An adversary can launch different attacks like a ciphertext only attack, a known plaintext attack, a chosen ciphertext/plaintext attack, etc. Encryption algorithms can help in securing the network against passive adversaries.
- *Active adversary* Besides having the equal abilities as a passive adversary, an active adversary can alter the contents of the communication. It may add, modify, replay, and delete the packets communicated within the network. In sensor networks, active adversaries are classified as either outsider adversaries or insider adversaries.
    - An outsider adversary does not have access to the secret information (e.g., a secret key) stored within the sensor nodes. An active outsider adversary can launch different attacks like a malleability and an unauthorized ciphertext aggregation. Authentication mechanisms (e.g., MAC, hash function, digital signature) help to protect the network against active outsider adversaries.
    - In sensor networks, node capture attacks are common due to the hostile deployments and the lack of temper-proof hardware for sensor nodes. Therefore, the captured sensor node can work as an insider adversary, where it has a complete access to the information stored within the node. It is generally referred to as a byzantine adversary [26]. An active insider adversary is considered as the strongest of all types of adversaries. The end-to-end privacy and the end-to-end authentication mechanisms ensure the security against an active insider adversary.

6.2 Cryptographic Attacks and Countermeasures

In this section, we discuss the security strength of the proposed protocol with respect to some well-known cryptographic attacks [45, 51] against concealed data aggregation protocols. As discussed in Sect. 6.1, we analyze the security strength of the proposed protocol against active and passive adversaries. Due to space constraints, the security proofs of EC-ElGamal and Homomorphic MAC are not discussed in this paper. However, interested readers can refer the paper on EC-ElGamal cryptosystem [31] and homomorphic MAC algorithm [2] for relevant security analysis and proofs.

6.2.1 Ciphertext Analysis

In a ciphertext-only attack, the aim of an adversary is to deduce a key or to derive a plaintext information from the gathered ciphertexts. However, the probabilistic cryptosystems can produce different ciphertexts for the same plaintext. Hence, analyzing the ciphertexts will not reveal any information about the corresponding plaintexts. In the proposed protocol, we used two different encryption mechanisms, namely, an asymmetric-key based and a symmetric-key based encryption mechanism. The asymmetric-key based EC-ElGamal is used to encrypt the plaintexts while the symmetric-key based encryption algorithm (e.g., AES) is used to encrypt a pair of the homomorphic MAC tag and a counter value. As the EC-ElGamal is a probabilistic cryptosystem, the randomness it uses for producing the ciphertexts ensures the protection against ciphertext analysis. In addition, a symmetric-key based encryption algorithm uses a MAC tag and a counter value as an input to the algorithm. Hence, the PRFs used to generate a MAC tag, as well as a counter, will generate a different MAC tag and a counter for the same ciphertext. Moreover, the encryption of this pair will always remain the probabilistic and secure against a ciphertext-only attack.

6.2.2 Known Plaintext Attack

In a known-plaintext attack, an adversary has collected the plaintext–ciphertext pairs from which it tries to deduce a key or a plaintext information. In WSNs, nodes are not physically secure and hence obtaining the plaintext–ciphertext pairs is not a hypothetical scenario. In addition, for any public key based cryptosystems, a public key is openly available throughout the network. Hence, any node can generate the plaintext–ciphertext pairs in order to cryptanalyze the system. All the well-known public-key cryptosystems provide the security against a known-plaintext attack. As we are using the EC-ElGamal cryptosystem, it inherently provides the security against a known-plaintext attack.

6.2.3 Malleability

Any cryptosystem that supports privacy homomorphism is inherently malleable. Such cryptosystems allow the processing of encrypted data without decrypting them or without having any secret information. As the proposed protocol uses an additively homomorphic EC-ElGamal cryptosystem, it can perform the ciphertext additions without a need for the decryption. In addition, the encrypted data processing does not require the secret information in order to process the data. Hence, any malicious node can also perform the same operations as performed by the genuine sensor nodes. Hence, it becomes crucial to prevent such malicious aggregation of encrypted data while allowing the genuine sensor nodes to process the encrypted data. In this paper, we used a dual authentication mechanism for ensuring protection against malicious adversaries. Although the proposed protocol cannot stop the malicious adversary in aggregating the data, the proposed protocol ensures the detection and the removal of maliciously aggregated data at the immediate next hop. The homomorphic property of a MAC algorithm verifies the validity of the aggregated ciphertext at the base station. In addition, the encryption using a pairwise-key ensure the protection against an unauthorized aggregation by the outsider as well as the insider adversaries. Hence, the proposed protocol ensures the protection against the active insider and outsider adversaries and adversaries cannot successfully violate the integrity of an aggregated data without compromising m consecutive nodes.

6.2.4 Node Capture Attack

Although node capture attacks cannot be completely mitigated in WSNs, resilience against such attacks can be achieved through strong authentication mechanisms. The proposed protocol encrypts the sensor readings using the EC-ElGamal cryptosystem and decrypts them only at the base station. Hence, privacy of the sensor readings at intermediate nodes is ensured when there exists node capture attacks. Moreover in the proposed protocol, we use two different authentication mechanisms for protecting the integrity of sensor readings against insider and outsider adversaries. First, the use of a pairwise secret key(s) to encrypt a homomorphic MAC tag(s) ensures that no outsider adversaries can violate integrity of the encrypted sensor readings without being detected. In addition, the use of a homomorphic

MAC ensures the protection against an active insider adversary. In the case of a node capture attack, a compromised intermediate node may perform malicious aggregation using the genuine keys. However, such attacks can easily be detected at the immediate next hop. A parameter *m* should be chosen carefully for achieving the adequate resilience against the node capture attacks launched by the collaborated active insider adversaries. Moreover, the compromised intermediate nodes can be isolated after detection through the intrusion prevention and key revocation techniques for making the network resilient against such attacks.

6.2.5 Replay Attack

In traditional networks, the replay protection considers only outsider adversaries. However in a concealed data aggregation scenario, the replay attacks can also be launched by the active insider adversaries. In addition, such insider adversaries can easily aggregate the replayed or previously stored data packets using the genuine keys. In the proposed protocol, we use a pair-wise key(s) for encrypting a ciphertext-counter pair. The pair-wise key used for encrypting a ciphertext-counter pair ensures that not only the outsider adversaries but also the active insider adversaries cannot launch replay attacks. A parameter *m* should be chosen carefully to thwart the replay attacks against active insider adversaries. A counter value generated at each end helps an aggregator node to verify the freshness of the packets without decrypting the ciphertext.

6.2.6 Denial of Service Attacks

Due to the inherent resource constraints, sensor nodes are more vulnerable to the denial of service attacks. Among the various different denial of service attacks that exists in WSNs, one of the famous attack is against the non-replenishable and scarce energy supply. In this type of attack, an adversary tries to waste the sensor node's precious energy. The radio frequency (RF) operations, the verification of incoming packets, etc. consume the precious energy. In addition, an adversary may target nodes that are nearer to the base station. Such nodes have a catastrophic impact on the performance of the sensor network. Although there aren't any cryptographic solutions that can protect against the denial of service attacks, it's impact can be reduce by using the aggregation for load balancing and using the symmetric-key based cryptosystems with fewer computation overhead. We can achieve the reduction in communication overhead through the identification of the malicious packets nearer to their sources. Such overhead reduction ensures resilience against the denial of service attacks.

6.3 Comparison of Secure Data Aggregation Protocols

In this section, we compare the proposed protocol with existing secure data aggregation protocols in order to measure its security strength. As shown in Table 2, the comparison is based on the well-known security requirements of the WSN's applications. In addition, as in-network processing has a strong impact on the security characteristics of various protocols, we consider in-network processing for the comparison.

- *Agg.*—En route aggregation
- *Conf.*—Confidentiality
- *Privacy*—Privacy at intermediate nodes
- *H-MA*—Hop-by-hop message authentication

Table 2 Comparison of secure data aggregation protocols

Protocol	Agg.	Conf.	Privacy	H-MA	E-MA	Replay
Perrig et al. [49]	×	✔	×	✔	×	✔
Girao et al. [20]	✔	✔	✔	×	×	×
Castelluccia et al. [8]	✔	✔	✔	×	×	×
Westhoff et al. [62]	✔	✔	✔	×	×	×
Mykletun et al. [42]	✔	✔	✔	×	×	×
Luk et al. [37]	×	✔	×	✔	×	✔
Ugus et al. [59]	✔	✔	✔	×	×	×
Girao et al. [19]	✔	✔	✔	×	×	×
Sun et al. [58]	✔	✔	✔	×	✔	×
Castelluccia et al. [7]	✔	✔	✔	✔	×	×
Li et al. [36]	✔	×	×	×	✔	×
Ozdemir et al. [45]	✔	✔	✔	×	✔	×
Sicari et al. [56]	✔	✔	✔	✔	×	×
Izawa et al. [25]	✔	×	×	×	✔	×
Westhoff et al. [63]	✔	✔	✔	×	✔	×
Zhou et al. [64]	✔	✔	✔	×	✔	×
Parmar et al. [47]	✔	✔	✔	✔	✔	×
The proposed protocol	✔	✔	✔	✔	✔	✔

- *E-MA*—End-to-end message authentication
- *Replay*—Replay protection

7 Conclusions

In this paper, we proposed a malleability resilient concealed data aggregation protocol for ensuring the integrity, privacy, data freshness and en route aggregation of converge-cast traffic in wireless sensor networks. Due to en route aggregation and encrypted data processing, the protection against these adversaries (insider and outsider) becomes extremely important in concealed data aggregation. The proposed protocol is the first that achieves these conflicting objectives against insider and outsider adversaries. We used different homomorphic primitives namely, the homomorphic encryption and the homomorphic MAC for performing the encrypted data processing. A comparison of the communication overhead with respect to the existing protocols exhibits the viability and efficiency of the proposed protocol on resource-constrained devices. We believe that the proposed protocol helps in improving the resource utilization in resource-constrained environments while achieving the desired security requirements. Moreover, we compare the performance of the proposal protocol to show the resilience against the well-known cryptographic attacks such as the known plaintext/ciphertext attacks, malleability, node capture attacks, replay attacks and denial of service attacks. We compare the security strength of the proposed protocol with respect to the existing secure data aggregation protocols. As per our knowledge, the proposed protocol is the first that ensures conflicting security objectives while performing en route aggregation of reverse multicast traffic.

Acknowledgments This research was a part of the project “A Secure Data Aggregation System and An Intrusion Detection System for Wireless Sensor Networks”. It was supported by the Department of Electronics and Information Technology, Ministry of Communications and Information Technology, Government of India.

References

1. Adler, J. M., Dai, W., Green, R. L., & Neff, A. C. (2000). *Computational details of the votehere homomorphic election system*. Bellevue: VoteHere Inc.
2. Agrawal, S., & Boneh, D. (2009). Homomorphic MACs: MAC-based integrity for network coding. In *Proceedings of the 7th international conference on applied cryptography and network security, ACNS ’09, Lecture Notes in Computer Science* (Vol. 5536, pp. 292–305). Paris-Rocquencourt: Springer. doi:10.1007/978-3-642-01957-9_18.
3. Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., & Cayirci, E. (2002). Wireless sensor networks: A survey. *Computer Networks: The International Journal of Computer and Telecommunications Networking*, *38*(4), 393–422. doi:10.1016/S1389-1286(01)00302-4.
4. Anastasi, G., Conti, M., Di Francesco, M., & Passarella, A. (2009). Energy conservation in wireless sensor networks: A survey. *Ad Hoc Networks*, *7*(3), 537–568. doi:10.1016/j.adhoc.2008.06.003.
5. Apavatjrut, A., Znaidi, W., Fraboulet, A., Goursaud, C., Lauradoux, C., & Minier, M. (2010). Energy friendly integrity for network coding in wireless sensor networks. In *Proceedings of the 4th international conference on network and system security, NSS’10* (pp. 223–230). Melbourne: IEEE. doi:10.1109/NSS.2010.32.
6. Boneh, D., Freeman, D., Katz, J., & Waters, B. (2009). Signing a linear subspace: Signature schemes for network coding. In *Proceedings of the 12th international conference on practice and theory in public key cryptography, PKC’09, Lecture Notes in Computer Science* (Vol. 5443, pp. 68–87). Irvine: Springer. doi:10.1007/978-3-642-00468-1_5.
7. Castelluccia, C., Chan, A. C. F., Mykletun, E., & Tsudik, G. (2009). Efficient and provably secure aggregation of encrypted data in wireless sensor networks. *ACM Transactions on Sensor Networks (TOSN)*, *5*(3), 20:1–20:36. doi:10.1145/1525856.1525858.
8. Castelluccia, C., Mykletun, E., & Tsudik, G. (2005). Efficient aggregation of encrypted data in wireless sensor networks. In *Proceedings of the 2nd annual international conference on mobile and ubiquitous systems: Networking and services, MOBIQUITOUS’05* (pp. 109–117). Washington, DC: IEEE. doi:10.1109/MOBIQUITOUS.2005.25.
9. Chan, A. C. F., & Castelluccia, C. (2008). On the (Im)possibility of aggregate message authentication codes. In *Proceedings of the IEEE international symposium on information theory, ISIT’08* (pp. 235–239). Toronto: IEEE. doi:10.1109/ISIT.2008.4594983.
10. Chan, A. C. F., & Castelluccia, C. (2011). A security framework for privacy-preserving data aggregation in wireless sensor networks. *ACM Transactions on Sensor Networks (TOSN)*, *7*(4), 29:1–29:45. doi:10.1145/1921621.1921623.
11. Chan, H., & Perrig, A. (2003). Security and privacy in sensor networks. *Computer*, *36*(10), 103–105. doi:10.1109/MC.2003.1236475.
12. Chen, Y. S., & Lei, C. L. (2013). Aggregate message authentication codes (amacs) with on-the-fly verification. *International Journal of Information Security*, *12*(6), 495–504. doi:10.1007/s10207-013-0202-0.
13. Domingo-Ferrer, J. (2002). A provably secure additive and multiplicative privacy homomorphism. In *Proceedings of the 5th international conference on information security, ISC’02, Lecture Notes in Computer Science* (Vol. 2433, pp. 471–483). Sao Paulo: Springer. doi:10.1007/3-540-45811-5_37
14. El Gamal, T. (1985). A public key cryptosystem and a signature scheme based on discrete logarithms. In *Proceedings of the advances in cryptology, CRYPTO’ 84, Lecture Notes in Computer Science* (Vol. 196, pp. 10–18). California: Springer. doi:10.1007/3-540-39568-7_2.
15. Fasolo, E., Rossi, M., Widmer, J., & Zorzi, M. (2007). In-network aggregation techniques for wireless sensor networks: A survey. *Wireless Communications*, *14*(2), 70–87. doi:10.1109/MWC.2007.358967.
16. Fontaine, C., & Galand, F. (2007). A survey of homomorphic encryption for nonspecialists. *EURASIP Journal on Information Security*, *2007*(15), 1–15. doi:10.1155/2007/13801.
17. Gennaro, R., Katz, J., Krawczyk, H., & Rabin, T. (2010). Secure network coding over the integers. In *Proceedings of the 13th international conference on practice and theory in public key cryptography, PKC’10, Lecture Notes in Computer Science* (Vol. 6056, pp. 142–160). Paris: Springer. doi:10.1007/978-3-642-13013-7_9.

18. Girao, J., Schneider, M., & Westhoff, D. (2004). CDA: Concealed data aggregation in wireless sensor networks. In *Proceedings of the ACM workshop on wireless security, WiSe'04* (pp. 1–2). Philadelphia: ACM. Poster presentation.
19. Girao, J., Westhoff, D., Mykletun, E., & Araki, T. (2007). TinyPEDS: Tiny persistent encrypted data storage in asynchronous wireless sensor networks. *Ad Hoc Networks*, *5*(7), 1073–1089. doi:10.1016/j.adhoc.2006.05.004.
20. Girao, J., Westhoff, D., & Schneider, M. (2005). CDA: Concealed data aggregation for reverse multicast traffic in wireless sensor networks. In *Proceedings of the 40th international conference on communications, ICC'05* (pp. 3044–3049). Seoul: IEEE. doi:10.1109/ICC.2005.1494953.
21. Gura, N., Pate, A., Wander, A., Eberle, H., & Shantz, S. C. (2004). Comparing elliptic curve cryptography and RSA on 8-bit CPUs. In *Proceedings of the 6th international workshop on cryptographic hardware and embedded systems—CHES'04, Lecture Notes in Computer Science*(Vol. 3156, pp. 119–132). Cambridge: Springer. doi:10.1007/978-3-540-28632-5_9.
22. Hill, J., Szewczyk, R., Woo, A., Hollar, S., Culler, D., & Pister, K. (2000). System architecture directions for networked sensors. *ACM SIGPLAN Notices*, *35*(11), 93–104. doi:10.1145/356989.356998.
23. Hoffstein, J., Pipher, J., & Silverman, J. (2008). *An introduction to mathematical cryptography* (1st ed.). Berlin: Springer. doi:10.1007/978-1-4939-1711-2.
24. Hu, L., & Evans, D. (2003). Secure aggregation for wireless networks. In *Proceedings of the symposium on applications and the internet workshops, SAINT'03* (pp. 384–391). Washington, DC: IEEE. doi:10.1109/SAINTW.2003.1210191.
25. Izawa, K., Miyaji, A., & Omote, K. (2012). Lightweight integrity for XOR network coding in wireless sensor networks. In *Proceedings of the 8th international conference on information security practice and experience, ISPEC'12, Lecture Notes in Computer Science* (Vol. 7232, pp. 245–258). Hangzhou: Springer. doi:10.1007/978-3-642-29101-2_17.
26. Jaggi, S., Langberg, M., Katti, S., Ho, T., Katabi, D., & Medard, M. (2007). Resilient network coding in the presence of byzantine adversaries. In *Proceedings of the 26th IEEE international conference on computer communications, IEEE INFOCOM'07* (pp. 616–624). Barcelona: IEEE. doi:10.1109/INFCOM.2007.78.
27. Johnson, R., Molnar, D., Song, D. X., & Wagner, D. (2002). Homomorphic signature schemes. In *Proceedings of the cryptographer's track at the RSA conference on topics in cryptology, CT-RSA'02, Lecture Notes in Computer Science* (Vol. 2271, pp. 244–262). London: Springer. doi:10.1007/3-540-45760-7_17.
28. Karlof, C., Sastry, N., & Wagner, D. (2004). TinySec: A link layer security architecture for wireless sensor networks. In *Proceedings of the 2nd international conference on embedded networked sensor systems, SenSys'04* (pp. 162–175). Baltimore: ACM. doi:10.1145/1031495.1031515.
29. Karlof, C., & Wagner, D. (2003). Secure routing in wireless sensor networks: Attacks and countermeasures. *AdHoc Networks*, *1*(2–3), 293–315. doi:10.1016/S1570-8705(03)00008-8.
30. Katz, J., & Lindell, A. Y. (2008). Aggregate message authentication codes. In *Proceedings of the 2008 the cryptopgraphers' track at the RSA conference on topics in cryptology, CT-RSA'08, Lecture Notes in Computer Science* (Vol. 4964, pp. 155–169). San Francisco: Springer. doi:10.1007/978-3-540-79263-5_10.
31. Koblitz, N. (1987). Elliptic curve cryptosystems. *Mathematics of Computation*, *48*(177), 203–209. doi:10.1090/S0025-5718-1987-0866109-5.
32. Koblitz, N., Menezes, A., & Vanstone, S. (2000). The state of elliptic curve cryptography. *Designs, Codes and Cryptography*, *19*(2–3), 173–193. doi:10.1023/A:1008354106356.
33. Krohn, M. N., Freedman, M. J., & Mazières, D. (2004). On-the-fly verification of rateless erasure codes for efficient content distribution. In *Proceedings of the IEEE symposium on security and privacy* (pp. 226–240). California: IEEE. doi:10.1109/SECPRI.2004.1301326.
34. Levis, P., Madden, S., Polastre, J., Szewczyk, R., Whitehouse, K., Woo, A., Gay, D., Hill, J., Welsh, M., Brewer, E., & Culler, D. (2005). TinyOS): An operating system for sensor networks. In *Ambient intelligence* (pp. 115–148). Berlin: Springer. doi:10.1007/3-540-27139-2_7.
35. Li, H., Li, K., Qu, W., & Stojmenovic, I. (2011). Secure and energy-efficient data aggregation with malicious aggregator identification in wireless sensor networks. In *Proceedings of the 11th international conference on algorithms and architectures for parallel processing—volume part I, ICA3PP'11, Lecture Notes in Computer Science* (Vol. 7016, pp. 2–13). Melbourne: Springer. doi:10.1007/978-3-642-24650-0_2.
36. Li, Z., & Gong, G. (2010). Data aggregation integrity based on homomorphic primitives in sensor networks. In *Proceedings of the 9th international conference on ad-hoc, mobile and wireless networks, ADHOC-NOW'10, Lecture Notes in Computer Science* (Vol. 6288, pp. 149–162). Edmonton: Springer. doi:10.1007/978-3-642-14785-2_12.
37. Luk, M., Mezzour, G., Perrig, A., & Gligor, V. (2007). MiniSec: A secure sensor network communication architecture. In *Proceedings of the 6th international conference on information processing in sensor networks, IPSN'07* (pp. 479–488). Cambridge: ACM. doi:10.1145/1236360.1236421.

38. Malan, D. J., Welsh, M., & Smith, M. D. (2004). A public-key infrastructure for key distribution in TinyOS based on elliptic curve cryptography. In *Proceedings of the 1st IEEE international conference on sensor and ad hoc communications and network, SECON'04* (pp. 71–80). Santa Clara: IEEE. doi:10.1109/SAHCN.2004.1381904.
39. Malan, D. J., Welsh, M., & Smith, M. D. (2008). Implementing public-key infrastructure for sensor networks. *ACM Transactions on Sensor Networks (TOSN)*, *4*(4), 22:1–22:23. doi:10.1145/1387663.1387668.
40. MEMSIC. (2015). MICAz mote platform. Datasheet. http://www.memsic.com/userfiles/files/Datasheets/WSN/6020-0060-04-B_MICAz.pdf. Accessed 12 March 2015
41. MEMSIC. (2015). TelosB mote platform. http://www.memsic.com/userfiles/files/Datasheets/WSN/6020-0094-02_B_TELOSB.pdf. Accessed 12 March 2015
42. Mykletun, E., Girao, J., & Westhoff, D. (2006). Public key based cryptoschemes for data concealment in wireless sensor networks. In *Proceedings of the IEEE international conference on communications, ICC'06* (pp. 2288–2295). Istanbul: IEEE. doi:10.1109/ICC.2006.255111.
43. Okamoto, T., & Uchiyama, S. (1998). A new public-key cryptosystem as secure as factoring. In *Proceedings of the international conference on the theory and application of cryptographic techniques, advances in cryptology, EUROCRYPT'98, Lecture Notes in Computer Science* (Vol. 1403, pp. 303–318). Espoo: Springer. doi:10.1007/BFb0054135.
44. Ozdemir, S., & Xiao, Y. (2009). Secure data aggregation in wireless sensor networks: A comprehensive overview. *Computer Networks: The International Journal of Computer and Telecommunications Networking*, *53*(12), 2022–2037. doi:10.1016/j.comnet.2009.02.023.
45. Ozdemir, S., & Xiao, Y. (2011). Integrity protecting hierarchical concealed data aggregation for wireless sensor networks. *Computer Networks: The International Journal of Computer and Telecommunications Networking*, *55*(8), 1735–1746. doi:10.1016/j.comnet.2011.01.006.
46. Paillier, P. (2000). Trapdooring discrete logarithms on elliptic curves over rings. In *Proceedings of the 6th international conference on the theory and application of cryptology and information security: advances in cryptology, ASIACRYPT'00, Lecture Notes in Computer Science* (Vol. 1976, pp. 573–584). Kyoto: Springer. doi:10.1007/3-540-44448-3_44.
47. Parmar, K., & Jinwala, D. C. (2014). Malleability resilient concealed data aggregation. In *Proceedings of the 20th EUNICE/IFIP WG 6.2, 6.6 workshop on advances in communication networking, EUNICE'14, Lecture Notes in Computer Science* (Vol. 8846, pp. 160–172). Rennes: Springer. doi:10.1007/978-3-319-13488-8_15.
48. Perrig, A., Stankovic, J., & Wagner, D. (2004). Security in wireless sensor networks. *Communications of the ACM*, *47*(6), 53–57. doi:10.1145/990680.990707.
49. Perrig, A., Szewczyk, R., Tygar, J. D., Wen, V., & Culler, D. E. (2002). SPINS: Security protocols for sensor networks. *Wireless Networks*, *8*(5), 521–534. doi:10.1145/990680.990707.
50. Peter, S., Piotrowski, K., & Langendoerfer, P. (2007). On concealed data aggregation for WSNs. In *Proceedings of the 4th IEEE consumer communications networking conference, CCNC'07* (pp. 192–196). Las Vegas: IEEE. doi:10.1109/CCNC.2007.45.
51. Peter, S., Westhoff, D., & Castelluccia, C. (2010). A survey on the encryption of convergecast traffic with in-network processing. *IEEE Transactions on Dependable and Secure Computing*, *7*(1), 20–34. doi:10.1109/TDSC.2008.23.
52. Pottie, G. J., & Kaiser, W. J. (2000). Wireless integrated network sensors. *Communications of the ACM*, *43*(5), 51–58. doi:10.1145/332833.332838.
53. Rault, T., Bouabdallah, A., & Challal, Y. (2014). Energy efficiency in wireless sensor networks: A top-down survey. *Computer Networks*, *67*, 104–122. doi:10.1016/j.comnet.2014.03.027.
54. Rivest, R. L., Adleman, L., & Dertouzos, M. L. (1978). On data banks and privacy homomorphisms. *Foundations of Secure Computation*, *4*(11), 169–180.
55. Sang, Y., Shen, H., Inoguchi, Y., Tan, Y., & Xiong, N. (2006). Secure data aggregation in wireless sensor networks: A survey. In *Proceedings of the 7th international conference on parallel and distributed computing, applications and technologies, PDCAT'06* (pp. 315–320). Taipei: IEEE. doi:10.1109/PDCAT.2006.96.
56. Sicari, S., Grieco, L. A., Boggia, G., & Coen-Porisini, A. (2012). DyDAP: A dynamic data aggregation scheme for privacy aware wireless sensor networks. *Journal of Systems and Software*, *85*(1), 152–166. doi:10.1016/j.jss.2011.07.043.
57. Simplicio, M. A, Jr, De Oliveira, B. T., Margi, C. B., Barreto, P. S. L. M., Carvalho, T. C. M. B., & NäSlund, M. (2013). Survey and comparison of message authentication solutions on wireless sensor networks. *Ad Hoc Networks*, *11*(3), 1221–1236. doi:10.1016/j.adhoc.2012.08.011.
58. Sun, H. M., Hsiao, Y. C., Lin, Y. H., & Chen, C. M. (2008). An efficient and verifiable concealed data aggregation scheme in wireless sensor networks. In *Proceedings of the international conference on embedded software and systems, ICESS'08* (pp. 19–26). Sichuan: IEEE. doi:10.1109/ICESS.2008.9.

59. Ugus, O. (2007). Asymmetric homomorphic encryption transformation for securing distributed data storage in wireless sensor networks. Master's thesis, Technische Universität Darmstadt, Germany. http://www.ist-ubisecsens.org/publications/diplarb_ugus.pdf. Accessed 20 Nov 2014

60. Wander, A. S., Gura, N., Eberle, H., Gupta, V., & Shantz, S. C. (2005). Energy analysis of public-key cryptography for wireless sensor networks. In *Proceedings of the 3rd IEEE international conference on pervasive computing and communications, PerCom'05* (pp. 324–328). Kauai: IEEE. doi:10.1109/PERCOM.2005.18.

61. Wang, Y., Attebury, G., & Ramamurthy, B. (2006). A survey of security issues in wireless sensor networks. *IEEE Communications Surveys & Tutorials*, *8*(2), 2–23. doi:10.1109/COMST.2006.315852.

62. Westhoff, D., Girao, J., & Acharya, M. (2006). Concealed data aggregation for reverse multicast traffic in sensor networks: Encryption, key distribution, and routing adaptation. *IEEE Transactions on Mobile Computing*, *5*(10), 1417–1431. doi:10.1109/TMC.2006.144.

63. Westhoff, D., & Ugus, O. (2013). Malleability resilient (premium) concealed data aggregation. In *Proceedings of the 4th IEEE international workshop on data security and privacy in wireless networks, D-SPAN'13* (pp. 1–6). Madrid: IEEE. doi:10.1109/WoWMoM.2013.6583470.

64. Zhou, Q., Yang, G., & Liwen, H. (2014). An efficient secure data aggregation based on homomorphic primitives in wireless sensor networks. *International Journal of Distributed Sensor Networks*, *2014*(962925), 1–11. doi:10.1155/2014/962925.

Keyur Parmar is a senior research fellow and a Ph.D. scholar at the Department of Computer Engineering, S. V. National Institute of Technology, Surat. He has received his M.Tech. degree in Computer Engineering from SVNIT, Surat (2010). He worked as a project scientist at BISAG, Gandhinagar and an Assistant Professor at GIT, Gandhinagar, before joining SVNIT, Surat as a Ph.D. Scholar. His research interests broadly include Information Security, Wireless Sensor Networks and Internet of Things.

Devesh C. Jinwala has been working as a Professor in Computer Engineering at the Department of Computer Engineering, S. V. National Institute of Technology, Surat, India since 1991. His principal research areas of interest are broadly Security, Cryptography, Algorithms and Software Engineering. Specifically his work focuses on Security and Privacy Issues in Resource-constrained environments (Wireless Sensor Networks) and Data Mining, Attribute-based Encryption techniques, Requirements Specification, and Ontologies in Software Engineering. He has been/is the Principal Investigator of several sponsored research projects funded by ISRO, GUJCOST, Govt of Gujarat and DiETY-MCIT-Govt of India.