Ann. Telecommun. (2017) 72:551–561
DOI 10.1007/s12243-017-0602-7

# Towards a set aggregation-based data integrity scheme for smart grids

**Mouzna Tahir[1] · Abid Khan[1] · Abdul Hameed[2] · Masoom Alam[1] · Muhammad Khurram Khan[3] · Farhana Jabeen[1]**

Received: 15 January 2017 / Accepted: 31 July 2017 / Published online: 16 August 2017


**Abstract** Data aggregation (DA) is the process of combining smart metering data so that it can be sent to a control center in package form rather than as individual data points. Smart metering data represents sensitive information that must be protected during the aggregation process. Traditional data aggregation schemes have addressed privacy issues based primarily on computationally expensive homomorphic encryption. In contrast, this paper presents a novel method based on hash chaining to verify the integrity of a set of aggregated data. This scheme divides the user's data into two diverse groups. It also enables the control center to collect more fine-grained data aggregation results at a reduced cost. In addition, the proposed scheme ensures data integrity by maintaining a hash chain and assigning new values in the hash chain by XORing previous hash values with the current hash value. The proposed scheme is evaluated in terms of computational cost and communication overhead. A comparative analysis of our proposed methodology with existing aggregation schemes regarding computational cost and communication overhead illustrates the optimality of our proposed scheme.

**Keywords** Smart grid · Privacy-preserving · Set aggregation · Data integrity · Hashing

✉ Abid Khan
abidkhan@comsats.edu.pk

Mouzna Tahir
muzna9227@gmail.com

Abdul Hameed
hameed@iqraisb.edu.pk

Masoom Alam
masoom.alam@comsats.edu.pk

Muhammad Khurram Khan
mkhurram@ksu.edu.sa

Farhana Jabeen
farhanakhan@comsats.edu.pk

[1] Department of Computer Science, COMSATS Institute of Information Technology, Islamabad, Pakistan

[2] Department of Computing and Technology, Iqra University, Islamabad Campus, Islamabad, Pakistan

[3] Center of Excellence in Information Assurance(COEIA), King Saud University, Riyadh, Saudi Arabia

## 1 Introduction

The introduction of the smart grid (SG) has revolutionized the way electricity is consumed and managed. More than 50% of all anticipated users belong to seven nations: Denmark, Germany, Italy, Austria, UK, France, and Spain [11]. A SG is an electrical grid that controls power usage by acting on information collected from the power grid by monitoring. It supports a bi-directional flow of power information between providers and consumers of electric power [37]. Compared to a traditional electrical grid, a smart grid has a more advanced structure, features two-way communication, is self-observing, and can be remotely monitored [14]. There are four main components in an SG involved in metering communications [20]:

1. A smart meter (SM)
2. An aggregator (also known as a gateway (GW))
3. A control center (CC)
4. Appliances

Smart meters are a key component in smart grids. Smart meters are devices that support two-way communication. They collect information about a household's electricity

consumption, which is sensitive, because it may reveal private information about the inhabitants of a dwelling, such as which appliances are being used at what time, when are the inhabitants of the house are home, and so on. This metering information is reported to a neighboring GW, e.g., a workstation. The GW gathers and "pre-forms" the data (e.g., it validates values and calculates totals) and then forwards the metering information to the CC for further investigation.

In this study, data aggregation (DA) involves combining smart metering data so it can be transferred to the control center in the form of packages rather than as individual values [21]. Immediate aggregation of smart meter data is an important feature of smart grids. For example, power usage data at multiple levels is aggregated periodically because the aggregated information is useful not only for monitoring and predicting power consumption but also for allocating and balancing loads and resources. DA schemes in smart grids can be distributed or centralized, depending on the communication architecture being used. DA schemes can adopt either a secure hop-by-hop DA strategy or an end-to-end DA strategy, or a combination of the two [3].

Over the past few years, many privacy-preserving DA schemes have been proposed to ensure data integrity in smart grids [6, 23]. However, much of the existing work is based on computationally expensive homomorphic encryption schemes (such as the Paillier cryptosystem [12, 31]). However, such schemes [10, 29] are impractical in smart grids due to bandwidth issues and their high computational costs. Therefore, it is essential to find a better way to provide efficient and secure data aggregation in smart grids. The concept of secure data aggregation (SDA) was introduced to protect user privacy. SDA can be used to collect information about the electricity consumption of a set of users without disclosing specific information about any individual user. Therefore, it plays an important role in preserving user privacy.

Security and privacy issues for user data aggregation are the most important challenges faced by smart grids. Security issues emerge due to various types of misrepresentation and attacks on user's energy utilization estimations. The aforementioned attacks include eavesdropping, data alteration, and injecting false information. Moreover, data protection issues involve ensuring that information does not become available to unauthorized entities. Confidentiality, authenticity, integrity, protection, and adaptation to internal failure are the principle security prerequisites that a privacy-preserving data aggregation scheme must satisfy. Research on SDA has traditionally focused on applying homomorphic encryption schemes, such as Paillier-based cryptosystems [29], BGN [10], and LWE [9]. However, the operations supported by such schemes all suffer from computational complexity issues. Recently, a set-based aggregation scheme was proposed by Lu et al. [22] that divides a user's data into two diverse groups. This scheme enables a control center to obtain more fine-grained data aggregation results at less expense. However, one problem with this approach is its inability to ensure data integrity. Our goal is to achieve set aggregation-based data integrity by applying a hash-chaining scheme for smart grids.

Specifically, this paper has the following contributions:

1. First, we extend the set aggregation approach by adding data integrity (a feature missing from the existing set aggregation scheme). To accomplish this, we propose a hash-chaining technique.
2. Second, we present a comparison with the Paillier cryptosystem-based data aggregation scheme proposed in [29]. In this comparison, the schemes' encryption, decryption, and chain verification times are measured and the overhead introduced by the proposed scheme is quantified.

The rest of the paper is organized as follows: In Section 2, we present related works. The system model, attacker model, and design goals are described in Section 3. Our proposed scheme is introduced in Section 4. Sections 5 and 6 provide security and performance analyses, respectively, and Section 7 concludes the paper.

## 2 Related work

In recent years, many researchers have investigated the SDA problem in the context of smart grids and numerous protocols have been proposed to secure smart grid networks and their devices. In this section, we briefly review some of the proposed SDA schemes. Saputro et al. [32] proposed an end-to-end (ETE) and hop-by-hop (HBH) Paillier-based homomorphic encryption scheme. However, HBH homomorphic encryption has a considerably higher computational overhead than does end-to-end encryption. Garcia et al. [17] proposed a secure and privacy preserving communication protocol for smart grid communications. The protocol combines Paillier homomorphic encryption and a "secret sharing" technique to detect data leakage. Chen et al. [25] achieved fault tolerance and privacy preserving data aggregation by using a hybrid approach. The proposed scheme was extended to support dynamic users and temporal aggregation, but it is not secure against internal attacks. However, the scheme provides protection against external attacks and has significantly less communication overhead than previous schemes. Doh et al. [13] also used a homomorphic encryption scheme that did not require significant communication or computational overhead and provided efficient data verification and attack detection. In addition, denial of service (DoS) attacks have been prevented using homomorphic encryption with an integrity check function.

Although, some efficient homomorphic encryption schemes that are secure against differential attacks [4, 7, 10,

25, 34] have been proposed, the aforementioned schemes support only whole set aggregation; they lack the ability of partial aggregation (subset). Lu et al. [26] proposed a scheme called bolstered set aggregation and derived a group check strategy with reduced verification costs. The scheme used a super-expanding arrangement to organize multidimensional information; then, they scrambled the organized information using the Paillier cryptosystem. However, in [26], it is possible for semi-honest attacker to read the individual users' metering data through the control center.

Contemporary studies include the works of [2, 5, 8, 15, 16, 18, 19, 24, 27, 30, 33, 35, 36] all of which address security issues in data aggregation for smart grids, as described below.

Asmaa et al. [2] proposed a lattice-based homomorphic encryption scheme for privacy-preserving data aggregation in smart grids. In this scheme, smart appliances aggregate their data themselves, without involving a smart meter. Due to the lattice-based homomorphic encryption scheme, the aggregated consumption data can then be verified by the smart meter and by the gateway node without decrypting the aggregated data. Ambrosin et al. [5] proposed a scheme that allowed fine-grained and anonymous collection of smart metering data in a collaborative multipath protocol. The proposed collaborative protocol is both random and verifiable, which allows a trusted verification authority to verify that a smart meter is working correctly by accessing its internal logs. He et al. [19] proposed a data aggregation scheme can withstand internal attacks. The proposed scheme also overcomes key leakage problems. Furthermore, because it uses a reduced number of bilinear pairing operations, this scheme has comparatively better performance than do other schemes. Shen et al. [33] proposed an efficient privacy-preserving scheme to aggregate smart grid cube data. Furthermore, the authors provided a batch verification technique to reduce the cost of authentication. However, as with previous schemes, this scheme is also based on the computationally expensive Pailler cryptosystem. Tonyali et al. [35] proposed a secure and reliable data aggregation scheme for IoT-enabled smart metering systems using fully homomorphic encryption (FHE) and secure multiparty computation (SMC). To prevent replay attacks, all the messages in this protocol are time-stamped. Jianbin et al. [30] proposed a scheme that used an extended version of the Lifted ElGamal encryption scheme to simultaneously support data aggregation, differential privacy, fault tolerance, and range-based filtering. Furthermore, the scheme can withstand false data injection attacks without revealing individual consumption data.

Bao et al. [8] proposed a scheme that achieved privacy preservation, fault tolerance, and data integrity. The authors adopted an enhanced version of AES to achieve the aforementioned properties. Experiment results suggested that the proposed scheme is efficient in terms of computation and communication costs. Amin et al. [15] used a bilinear pairing identity-based encryption scheme to update certificates in smart grid communications. The proposed scheme can achieve both data and gateway privacy. Additionally, the scheme is robust to data replay, modification, man-in-the-middle and sybil attacks. Wang et al. [36] proposed a scheme based on bilinear pairing and the Castagnos–Laguillaumie cryptosystem that features anonymous and secure data aggregation for a fog-based public cloud. The authors provided a concrete instantiation of their scheme, and a performance evaluation suggested that the scheme is both efficient and secure. Recently, Lu et al. [22], overcome the inefficiency problem in a set aggregation-based scheme for smart grids. Although the proposed set-based aggregation scheme is efficient and can provide more accurate smart meter readings, the data integrity issue has not been addressed. In contrast, this paper addresses the data integrity issue in the set aggregation-based scheme using a hash-chaining-based approach.

## 3 Models and design goals

This section describes the system and attacker model used in the rest of the paper as well as the desirable security properties and design goals. We discuss why the security properties and goals are important to smart grid security.

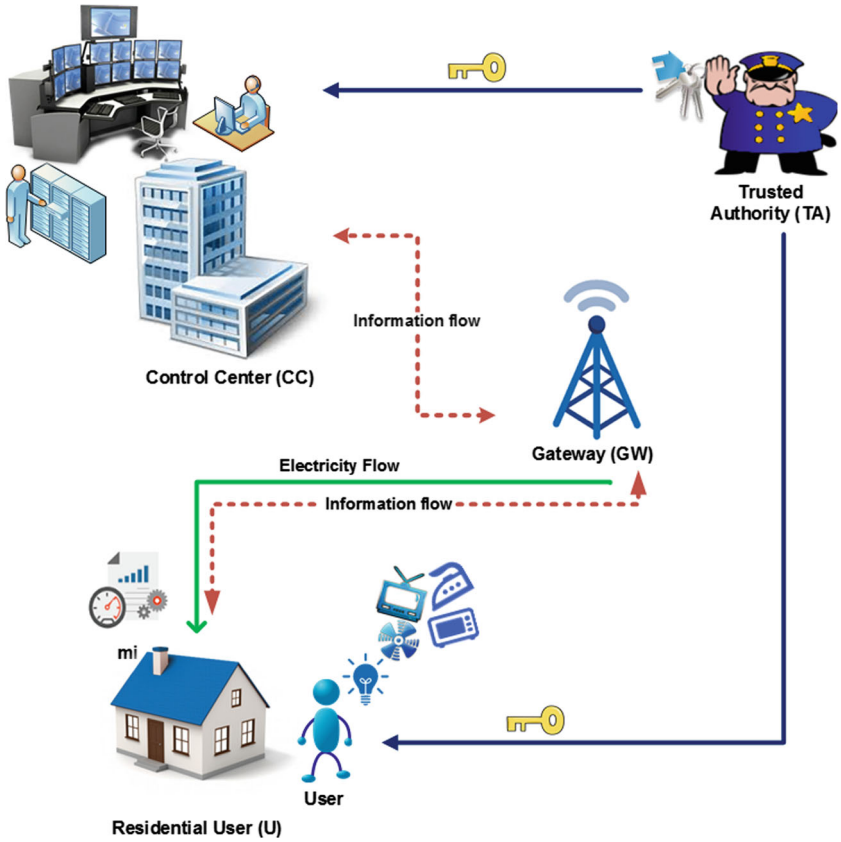
### 3.1 System model

In this subsection, we describe the entities and their roles. The four entities constituting our system model are shown in Fig. 1.

- Trusted authority (TA)
- Control center (CC)
- Residential gateway (GW) or data concentrator
- Residential users $U = \{U_1, U_2, ..., U_N\}$.

- Trusted authority: The TA [28] is a fully trusted entity capable of overseeing and conveying keys to different entities throughout the entire framework. After the keys have been distributed, the TA is no longer required for the data aggregation process.
- Control center: The CC is the most trusted entity in the smart grid. The CC is responsible for collecting, processing, and de-aggregating the aggregated information received from the private GW.
- Residential gateway: The GW is a powerful component that plays an aggregator role in the system. The main responsibilities of the GW are as follows:
  - To connect the CC and residential users (U) by enabling communication between these parties in the system.
  - To collect and aggregate the metering data from $U = \{U_1, U_2, ..., U_N\}$ in a residential area.

**Fig. 1** System model




  - To forward the aggregated electricity usage information to the CC.

- Residential users: A residential area (RA) is a typical area in which residential users $U = \{U_1, U_2, ..., U_N\}$ live. Each user $U_i \in U$ is equipped with an SM. The SM continually measures power usage and sends a record every 15 min, $m_i$, to the CC via the GW.

### 3.2 Attacker model

We assume that each residential user and the GW communicate using inexpensive WiFi technology. However, we consider an "honest-but-curious" model for both the CC and GW in which all the residential users are considered as honest in our set aggregation protocol. In a real smart grid system, attackers often launch attacks such as false data injection and DDoS. Therefore, our attacker model considers communication attacks, which is still a challenging issue. In a communication attack, an attacker eavesdrops on the communication package between a client and the aggregator and tries to compromise the client's privacy. We make the following assumptions about the attacker's capabilities:

1. Both the CC and GW are honest-but-curious.
2. All the residential users $U = \{U_1, U_2, ..., U_N\}$ are honest.
3. An adversary can compromise the gateway.
4. An adversary can also compromise household measurements by capturing keys and data during transmission.

### 3.3 Design goals

For smart grid communication, our goal is to develop a set aggregation-based data integrity scheme under the previously described system and attacker model. In particular, our scheme should meet the following desirable objectives.

1. Data integrity: In the context of smart grids, the term "data integrity" means to protect user consumption reports from modification by an unauthorized party. However, if a report is modified, the system must be able to identify that it has been altered. However, data integrity is the highest priority of all the requirements; when data integrity is not ensured, the entire power system may be abused. Serious damage can occur when power usage measurements are altered, when they are disclosed, or when they are not delivered to the destination during end to end communication.
2. Efficiency: The data aggregation process must be efficient. The proposed set aggregation-based data integrity scheme should consider both computational and communication overhead. However, it must also consider

the overhead between the users and the GW as well as between the GW and the CC. The computational cost of our scheme is expected to be significantly less than that of traditional homomorphic encryption schemes such as those proposed in [10, 29]. However, it is expected to be comparatively higher compared to [22] due to the computational costs of hash chaining. In addition to efficient communications, our proposed set aggregation-based data integrity scheme should also use a single aggregated data block for transmission, similar to the existing set aggregation scheme [22].

## 4 Proposed set aggregation-based data integrity scheme

This work presents a novel method to verify the integrity of a set of aggregated data based on a hash-chaining scheme. In hash chaining, each new value of the hash chain is calculated by XORing the hash of the previous value with the current value to verify the integrity of data. The abbreviations and cryptographic functions used in this paper are shown in Table 1.

We first review the hard problems involved in groups with composite order to provide a theoretical foundation for our proposed scheme as in [22].

**Table 1** Notations and cryptographic functions

| Symbol | Definition |
|---|---|
| U | User |
| GW | Gateway |
| CC | Control center |
| $k$ | Security parameter |
| $\triangle$ | Maximum data consumption in time t |
| $p, q$ | Large primes chosen by the CC |
| $N_{max}$ | Maximum number of users in set $U$ |
| $\mathbb{G}$ | Multiplicative group |
| $g$ | Group generator |
| $H$ | Cryptographic hash function chosen by the CC |
| $h_0$ | Public parameter I |
| $h_1$ | Public parameter II |
| $\oplus$ | Exclusive-OR operations |
| $p_k$ | The CC's public key |
| $S_k$ | The CC's private key |
| $U_i$ | $i^{th} user$ |
| $m_i$ | $U_i$'s electricity consumption |
| th | Threshold value chosen by the CC |
| $C_i$ | Encrypted version of $m_i$ |
| $x_i$ | $U_i$'s secret key |
| $x_0$ | The CC's secret key |

– *Hard problems in groups with composite order:* Given a security parameter $k$ and a cyclic multiplicative subgroup $\mathbb{G}(g, \times)$, let $g$ be the primitive element with composite order $n$, where $n = pq$ and $p$ and $q$ are large prime numbers such that $|p| = |q| = k$. We use the same definitions of the Decisional Diffie-Hellman (DDH) Problem and the Subgroup Decision (SD) Problem in $\mathbb{G}$ that were described in [22].

Our proposed approach consists of the following steps:

### 4.1 System initialization

Given the security parameter $k$, a number $N_{max}$ indicating the maximum number of users in $U$, and a small number $\triangle$, a random IV is also generated and used as an initial hash value.

**Algorithm 1** Key Generation Algorithm

**procedure** KEY GENERATION ALGORITHM
**Input**: $k$, $\Delta T$ , $N_{max}$, IV
**Output**: $K_{pub}$= $(n, g, h_0, h_1, H)$, $S_k = p$
  Choose two primes $p$ and $q$ randomly
  Select a security parameter such that $|p| = |q|$
  Compute $n = p.q$ ▷ such that $2q > p$ and $p - q > N_{max}$ ▷ $\Delta$ represents power usage data in every time '$t$' ▷ where $N_{max}$ represent maximal number of users in a set $U$
  Choose a cyclic multiplicative group $\bullet$ with generator $g$ and $|\bullet| = n$
  Choose $H : \{0, 1\}^* \rightarrow \bullet$
  Compute $h_o = g^q \in \bullet$
  Compute $h_1 = g^p \in \bullet$
  Return $K_{pub} = (n, g, h_0, h_1, H)$, $S_k = p$
secret key $S_k = p$ ▷ $S_k$ is the private key
**end procedure**

### 4.2 Encryption and hash-chain construction at the user end

$U_i$ compares consumption data with 'th'. When $m_i \geq th$, $u_i \in u_1$ uses his secret key $x_i$ to compute $c_i = g^m.h_1.H(t)^{x_i}$. Otherwise, when $m_i < th$, $u_i \in u_0$ computes $c_i = h_0^{m_i}.H(t)^{x_i}$. Next $U_i$ computes the hash values and starts building the hash chain as $H_1 = H(c_i) \oplus H_0$, where $H_0 = H(IV)$, $H_i = H(c_i) \oplus H_{i-1}$. The computational complexity is never more than a constant $O(1)$. Next, $c_i$ and $H_i$ are sent to the aggregator. Here, the initial value of the hash, $H_0$ is initialized using an initialization vector $(IV)$. A pseudorandom value is generated for $IV$, which is then hashed using an MD5 digest and assigned to $H_0$, i.e., $H_0 = H(IV)$. This $H_0$ acts as the first hash value. Note

that using MD5 is not a requirement in our scheme; SHA-1 could also be used here, which produces a 160-bit (20- byte) digest value [28].

**Algorithm 2** Encryption and Chain construction

**procedure** ENCRYPTION AND CHAIN CONSTRUCTION
**Input**: $m_i, \bullet = \{U_1, U_2....U_N\}, t, th, x_i$
**Output**: $C_i, Hi$
  **for** $(i = 0; i \leq N; i++)$ **do**
    **If** $(m_i \geq th, U_i \in \bullet_{\bullet})$
    $C_i = g^{m_i}.h_1.H(t)^{x_i}$ ▷ The encryption when $m_i \geq th$
    **Else If** $(m_i < th, U_i \epsilon \bullet_{o})$
    $C_i = h_o^{m_i}.H(t)^{x_i}$ ▷ The encryption when $m_i < th$
    $U_i$ Computes Hash values
    $H_i = H(C_i) \bigoplus H_{i-1}$ ▷ The hash value of the current $C_i$ is XORED with that of the previous $C_i$ and $H_0 = H(IV)$,
  **end for**
    Return $H_i, C_i$
  **end procedure**

### 4.3 Aggregation and hash-chain verification at the gateway

After receiving $c_i$ and $H_i$, where $i = 1, 2, 3...N$ from the residential users $u$, the gateway performs the following aggregation.

$$C = \prod_{i=1}^{N} c_i$$
$$= g^{\Sigma U_i \in U_1 m_i}.g^{\Sigma U_j \in U_o m_j}.H(t)^{\Sigma_{i=1}^{N} x_i}$$
$$= g^{\Sigma U_i \in U_1 m_i}.h_0^{\Sigma U_j \in U_o m_j}.h_1^{|U_1|} H(t)^{\Sigma_{i=1}^{N} x_i}$$
.

Figure 2 depicts two receiving ends at the gateway, where the aggregation and verification processes are performed. Moreover, $H_i$ is stored for integrity verification purposes:

$\widehat{H} = H' \oplus H_{i-1}$ and $\widehat{H}$ is compared to $H_i$. If it matches, then the integrity has been preserved; otherwise, some tampering has occurred during the metering data transmission. In contrast, $c_i$ is used during the aggregation process. In the aggregation process, the hash value of the aggregated sum $C$ is calculated and stored in $\widehat{H}$. Moreover, the GW forwards both the resulting $\widehat{H}$ and $C$ to the CC. The aggregation and hash-chain verification details are illustrated in Fig. 3.

$$\widehat{H} = H(C)$$

### 4.4 Decryption and hash-chain verification at the control center

At the CC, the decryption process is the same as that described in [22] because we are extending that scheme to provide data integrity. The CC receives the values $C$ and $\hat{H}$ and then performs verification by recalculating the hash and comparing the calculated hash with the received hash using its secret key $x_0$ .

$$H' = H(C)$$

The CC tests whether $\widehat{H} = H'$. When it matches, then the aggregated sum has been preserved; otherwise, some tampering has occurred at the GW. The CC also saves $C$ for future records. The $CC$ can recover the aggregated data by using its secret key to compute

$$D = C.H(t)^{x_0}$$
$$D' = D^p$$
$$D = D'.\widehat{D}$$
.

Because this work adopts the encryption and decryption processes from Lu et al. [22], we do not provide the details of the decryption process here; instead, we refer interested readers to Lu et al. [22].

**Algorithm 3** Decryption and verification

**procedure** $Ver\_Dec \sum_{U_i \in U1} m_i$ AND $|U_1|$
**INPUT**:$\overline{D} = h_1^{\sum_{U_i \in U1} m_i + p.|U_1|}, x_o$, C, $\hat{H}$
**Output**: Verified / not verified , $\sum_{U_i \in U1}$
Calculate $H'$ = H(C)
**If** ( $\hat{H} = H'$)
**Print** "VERIFIED"
**Else**
**Print** "Not- VERIFIED"
**End If**
  **for** (i = 0; i ≤ N ; i++) **do**
    **for** (j = 0; j ≤ N. Δ ; j++) **do**
**If** $(h_1^j.(h_1^p)^i == \overline{D})$ **Then**
set $\sum_{U_i \in U1} m_i = j, |U_1| = i$ ▷ For recovering the aggregated sum at CC. For details see Ref. [22]
    **end for**
  **end for**
    Return $\sum_{U_i \in U1} m_i, |U_1|$
  **end procedure**

## 5 Security analysis

This section contains a discussion of the security and privacy properties supported by the proposed scheme. Our

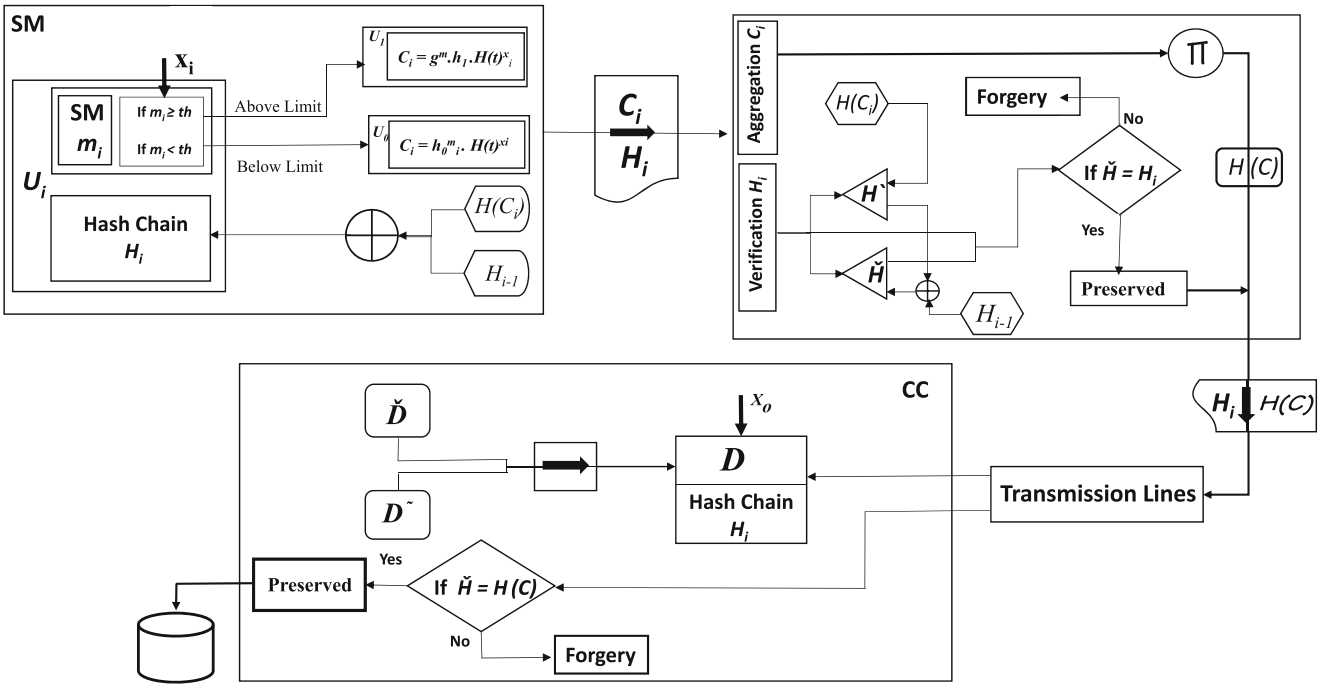



**Fig. 2** Set aggregation-based data integrity scheme with hash chaining

proposed scheme uses the RSA encryption algorithm to provide data privacy. To maintain data integrity, our scheme introduces the concept of hash chaining with a set aggregation-based approach, resulting in an extension of the work presented in [22]. The proposed scheme provides the following security properties:

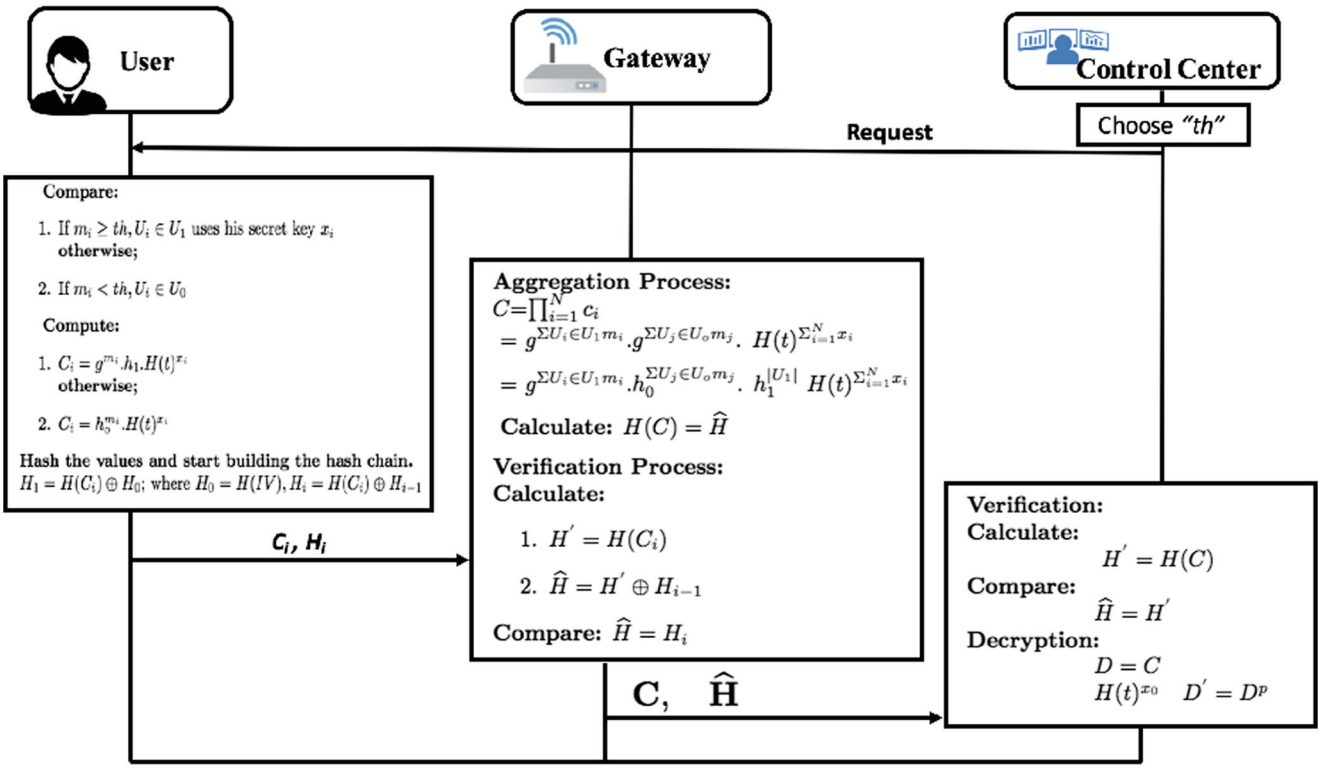



**Fig. 3** Scheme block diagram

- Privacy preservation.
- Authentication and data integrity.
  - Source authentication
  - Data integrity

### 5.1 Privacy preservation

Our scheme borrows the privacy preservation properties from [29]. All the communication from the user side is encrypted using the following two RSA encryption schemes. To encrypt data, $U_i$ compares the consumption data with "th." When $m_i \geq th$, $u_i \in u_1$ uses its secret key $x_i$ to compute $c_i = g^m.h_1.H(t)^{x_i}$. Otherwise, when $m_i < th$, $u_i \in u_0$ computes $c_i = h_0^{m_i}.H(t)^{x_i}$. In our proposed scheme, communication involves two transmissions: from the client to the GW and from the GW to the CC. Here, we consider the communication between user and the GW, where each individual user $U_i$ transmits one and only one encrypted data (cipher-text) "$C_i$" item, which is then appended to the hash chain "$H_i$" of such encoded data items at every time interval "$t$" (e.g., every 15 min) at the GW.

The proposed scheme not only preserves the privacy of each individual user's data, but also allows the CC to read the set aggregation results.

### 5.2 Authentication and data integrity

Next, we consider the communication between the GW and CC, where a verification of the hash chain $H(C_i)$ is performed at the gateway. After verification, the GW first aggregates the received $C_i$ values. Then it calculates the hash of the aggregated data, $(C)$, and finally, forwards $C$ and $H(C)$ to the CC. After receiving the transmission, the CC computes the hash of the aggregated information, $(C)$, to perform verification. After verification, the CC decrypts the aggregated sum.

We chose a security key length of $k = 512$. At a given time point the size of the residential users $U_i$ report is $|C_i| =$ 1,024 bits, and the length of $P = 2pq + 1$ is 1,025 bits when $|p| = |q| = k$, which is equal to 512. In subgroup G of $\mathbb{Z}_P^*$, any ciphertext (including either $C_i$ or the aggregated $C$) is less than or equal to 1,025 bits due to the additional transmission of hash chain $H_i$ along with the $C_i$ from the GW.

From this discussion, it is clear that our proposed scheme simultaneously supports both data integrity and privacy preservation. Because both security properties are provided at the same time, the proposed scheme incurs some communication and computational overhead. From a communication perspective, the hash chain must be transmitted along with the encrypted data. From a computational perspective, the hash chain must be computed, which incurs computational overhead. Our proposed scheme uses an MD5 algorithm that generates a 16-byte hash value. In our proposed scheme, the 16-byte hash is consistently delivered; this hash value acts as communication overhead and marginally increases the transmission time during communication compared with the existing set aggregation scheme [22]—particularly when the number of users is substantial. Later, we show that our proposed scheme performs one hash chaining $H_i$ and two hash verification operations to ensure the trustworthiness of user information during transmission, which causes an additional communication overhead compared with the existing set aggregation scheme [22].

## 6 Performance evaluation

In this section, we study the performance of the proposed data integrity scheme by evaluating the computational and communication overhead from the users to the GW and from the GW to the CC. We implemented our set aggregation-based data integrity scheme using Java. For the cryptographic primitives, we used the Java cryptographic architecture (JCA) [1]. We executed our experiments 24 times on a Windows 10 Professional platform eqiupped with an Intel(R) Core(TM) i3 with a 1.70 GHz processor and 4 GB RAM. These results represent an average of 24 runs. Table 2 shows the parameters used in our experimental setup.

The experimental results show that our proposed scheme is computationally more efficient during encryption compared to BGN [10] and the Paillier cryptosystem scheme [29]. However, our encryption results are similar to the results of the existing set aggregation scheme [22]. Moreover, our contribution does not involve proposing a new encryption scheme but in providing data integrity using hash chaining.

### 6.1 Computational overhead

The experiments were executed between 20–24 times by increasing the number of users from 50 to 400 with a step size of 10 with a settled limit estimation threshold of 5.

**Table 2** Experimental setup

| Parameter | Setting value |
|---|---|
| $k$ | Security parameter k = 512 |
| $\mathbb{G}$ | Subgroup of $\mathbb{Z}_p^*$ of order $n = p.q$, |
| $N_{max}$ | $N_{max} = 400$ |
| $N$ | $N$ = 50,100,150,200,250,300,350,400 |
| $\triangle$ | $\triangle = 5$ |
| $th$ | Randomly chosen threshold value from[1, $\triangle$] |

We expected that our proposed scheme would involve some additional computational costs compared with the existing scheme [22], because our scheme requires constructing the hash chain at the user side as well as generating encrypted measurements. However, our scheme is more efficient than other homomorphic encryption schemes such as [29] and [10]. We evaluated four aspects of the computational overhead: encryption time, aggregation time, decryption time, and verification time for both the hash chain and the aggregated sum "$C$." Hash chain construction is a continuous process that consists of just one hash operation and one XOR operation.

From Table 3, it is evident that the encryption time (in milliseconds) of the set aggregation scheme is significantly less than the encryption time required by the Paillier- and BGN-based aggregation schemes, which both depend on homomorphic encryption. The cost of such encryption increases steadily as the number of users ($N$) increases.

The BGN homomorphic encryption scheme [10] which supports an arbitrary numbers of addition and, notably, one multiplication operation, requires more encryption time when compared with the proposed set aggregation-based scheme, the existing set aggregation-based scheme [22], and the Paillier additive homomorphic encryption scheme [29] as shown in Table 3. Our proposed scheme substantially reduces the computational overhead and results in a lower encryption time.

Figure 4 depicts the average computational cost of aggregating the encrypted measurements ($c_i$) and Fig. 5 shows the decryption time required by our proposed scheme as compared to the existing set aggregation-based scheme. The aggregation time of our proposed scheme is slightly higher than that of the existing set aggregation-based scheme [22]. It varies as the number of residential users $N$ increases from 50 to 400 with an increment of 50. This result occurs because of the incorporation of the hash chaining mechanism introduced in the existing scheme [22]. The number of users $N$ has little effect on the aggregation and decryption phases after the hash-chaining algorithm is used in advance, as is evident from Figs. 5, 6 and 7.

**Table 3** Comparison of encryption time(ms)

| No. of users | Lu et al. [22] | Proposed scheme | Mustafa et al. [29] | Chen et al. [10] |
|---|---|---|---|---|
| 50 | 16.394 | 17.57 | 922 | 92000 |
| 100 | 17.869 | 17.93 | 1816 | 193000 |
| 150 | 21.008 | 22.46 | 2733 | 287000 |
| 200 | 24.12 | 25.46 | 3895 | 406000 |
| 250 | 25.81 | 26.11 | 4324 | 513000 |
| 300 | 27.514 | 28.21 | 5471 | 597000 |

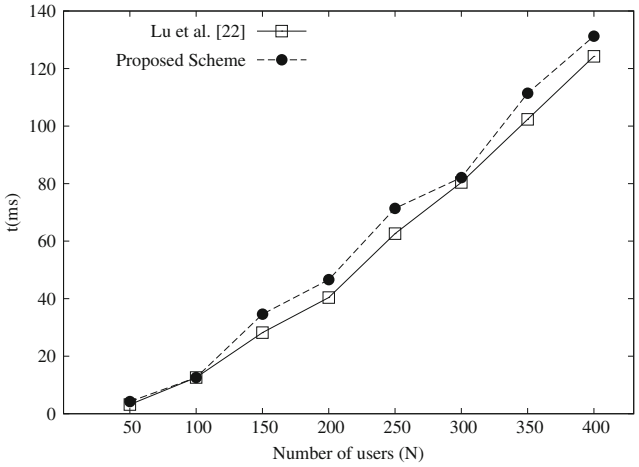



**Fig. 4** Aggregation time

To verify the integrity of users' encrypted measurements ($c_i$), hash verification is performed at the GW and CC. Moreover, a message digest (hash) value is stored at the GW that can be used later for hash chain verification or aggregated sum ($C$) verification at the CC. Figure 6 depicts the time required by our proposed scheme to verify the aggregated sum ($C$), and Fig. 7 shows the time required to verify the integrity of the hash chain constructed from the users' encrypted measurements ($c_i$). Note that the average verification time taken for 50 to 400 users at the GW is less than the aggregated sum verification time at the CC.

### 6.2 Communication overhead

In our proposed scheme, communication overhead occurs during two transmissions: (i) from the user side to the GW and (ii) from the GW to the CC. We first consider the communication between user and GW, where every individual user $U_i$ transmits an encrypted $C_i$ with the hash chain $H_i$ of

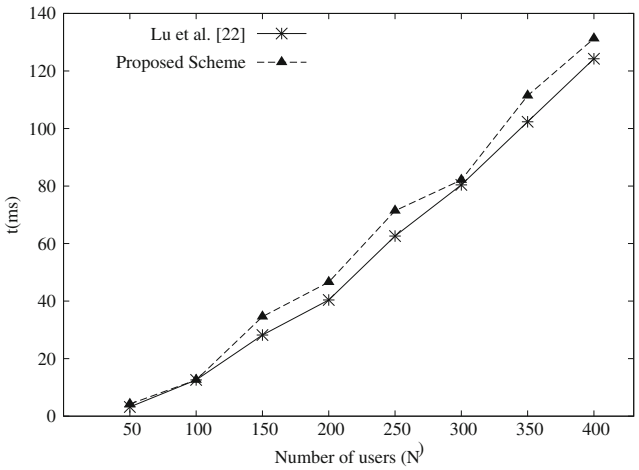



**Fig. 5** Decryption time

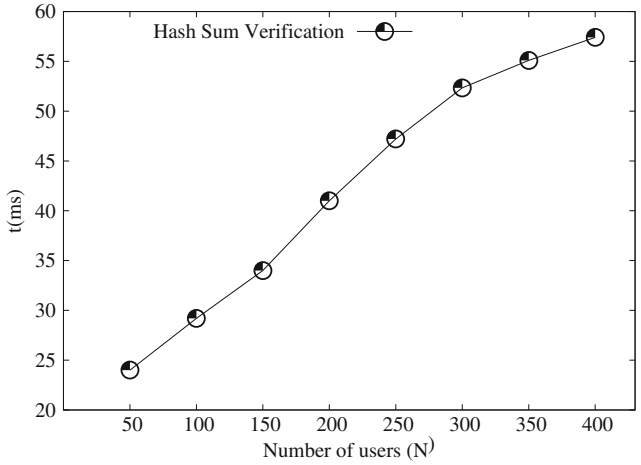



**Fig. 6** Time to verify aggregated sum (C)

the encrypted data items at every time interval $t$ (e.g. every 15 minutes) to the GW. Then, we consider the communication between the GW and CC, when hash chain $H(C_i)$ verification is performed at the GW. After verification, the aggregator at GW aggregates the received $C_i$ and calculates the hash of the aggregated data ($C$). Finally, it forwards $C$ and $H(C)$ to the CC. The CC additionally computes the hash of the aggregated information ($C$) to perform verification. After the verification process, the CC decrypts the aggregated sum.

We chose a key length of $k = 512$. At a given time point, the size of the residential users $U_i$ report is $|C_i| = 1{,}024$ bits, and the length of $P = 2pq+1$ is 1,025 bits when $|p| = |q| = k$, which is equal to 512. In subgroup G of $\mathbb{Z}_P^*$, any ciphertext (including either $C_i$ or the aggregated $C$) is less than or equal to 1,025 bits due to the additional transmission of hash chain $H_i$ along with the $C_i$ from the GW.

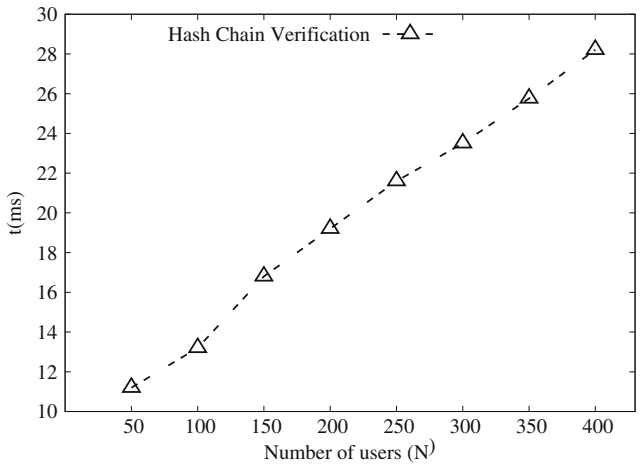



**Fig. 7** Time to verify hash chain

The experimental results illustrate that although our proposed approach ensures data integrity, it lags behind in communication cost because our use of the MD5 algorithm incurs a consistent 16-bytes communication overhead. Our proposed scheme performs one hash chaining $H_i$ and two hash verification operations to ensure the privacy of user information during transmission, which results in an additional communication overhead compared to the existing set aggregation-based scheme [22]. Using hash chaining does require some computation and communication overhead; however, we argue that such overhead is unavoidable if we want to achieve data integrity for smart metering data communication. The hash chaining is used to provide such integrity; however, our proposed scheme, which ensures information integrity while also safeguarding security is not as effective from a communication cost point of view because the hash-chaining involves extra costs. In our scheme, hash chaining uses a hashing calculation (MD5) that delivers a 16-byte (constant) hash length. In our proposed scheme, this consistent 16-byte hash functions as extra communication overhead; thus, it marginally increases the communication transmission time compared to the existing scheme, particularly when the number of users is large.

## 7 Conclusion and future work

The deployment of advanced SGs has introduced a number of problems involving data privacy. These security concerns revolve around gathering data to estimate data consumption based on users' energy utilization. In this context, a set aggregation-based data integrity scheme for smart grids was proposed. In this scheme, a hash-chaining based mechanism is adopted to achieve data integrity in smart grids. Hash-chain construction is a computationally inexpensive operation in which the hash of a current value is XORed with the hash of previous values. We compared the proposed scheme with an existing set aggregation-based scheme to investigate the overhead introduced by the hash-chaining scheme. We also compared the proposed scheme with Paillier and BGN cryptosystem-based schemes. The encryption and decryption costs of the set aggregation-based scheme are considerably less than those of the Paillier and BGN cryptosystem-based schemes. The overhead introduced by hash chaining is outweighed by the security it provides. In future work, we intend to formally evaluate our proposed scheme.

**Acknowledgments** The authors would like to thank the COMSATS Institute of Information Technology and the Higher Education Commission of Pakistan for their support and encouragement.

## References


1. Java Cryptography Architecture (JCA). http://docs.oracle.com/javase/7/docs/technotes/guides/security/crypto/CryptoSpec.html#KeyGeneratorvisitedon11-03-2016
2. Abdallah A, Shen X (2016) A lightweight lattice-based homomorphic privacy-preserving data aggregation scheme for smart grid. IEEE Trans Smart Grid PP(99). https://doi.org/10.1109/TSG.2016.2553647
3. Abdullah MDH, Welch I, Seah WK (2013) Efficient and secure data aggregation for smart metering networks. In: 2013 IEEE eight international conference on intelligent sensors, sensor networks and information processing. IEEE, pp 71–76
4. Alharbi K, Lin X (2012) LPDA: A lightweight privacy-preserving data aggregation scheme for smart grid. In: 2012 international conference on wireless communications & signal processing (WCSP). IEEE, pp 1–6
5. Ambrosin M, Hosseini H, Mandal K, Conti M, Poovendran R (2016) Despicable me(ter): anonymous and fine-grained metering data reporting with dishonest meters. In: 2016 IEEE conference on communications and network security (CNS), pp 163–171
6. Bao H, Chen L (2015) A lightweight privacy-preserving scheme with data integrity for smart grid communications. Concurrency and computation: practice and experience
7. Bao H, Lu R (2015) A new differentially private data aggregation with fault tolerance for smart grid communications. J IEEE Internet Things 2(3):248–258
8. Bao H, Lu R (2017) A lightweight data aggregation scheme achieving privacy preservation and data integrity with differential privacy and fault tolerance. Peer-to-Peer Netw Appl 10(1):106–121
9. Bertino E, Yi X, Paulet R (2014) Homomorphic encryption and applications
10. Chen L, Lu R, Cao Z, AlHarbi K, Lin X (2015) Muda: multi-functional data aggregation in privacy-preserving smart grid communications. Peer-to-Peer Netw Appl 8.5:777–792
11. Colak I, Sagiroglu S, Fulli G, Yesilbudak M, Covrig CF (2016) A survey on the critical issues in smart grid technologies. Renew Sust Energ Rev 54:396–405
12. Damgard IJ (2001) A generalisation, a simplification and some applications of Paillier's probabilistic public-key system. In: Proceedings of the 4th international workshopx on practice and theory in public key cryptosystems
13. Doh I, Lim J, Chae K (2013) Secure aggregation and attack detection for smart grid system. In: 2013 16th international conference on network-based information systems (NBis). IEEE, pp 270–275
14. Farhangi H (2010) The path of the smart grid. IEEE Power and Energy Magazine 8(1):18–28
15. Ferrag MA (2017) EPEC: an efficient privacy-preserving energy consumption scheme for smart grid communications. Telecommun. Syst:1–18. https://doi.org/10.1007/s11235-017-0315-2
16. Fu Z, Wu X, Guan C, Sun X, Ren K (2016) Toward efficient multi-keyword fuzzy search over encrypted outsourced data with accuracy improvement. IEEE Trans Inf Forensics Secur 11(12):2706–2716
17. Garcia FD, Jacobs B (2011) Privacy-friendly energy-metering via homomorphic encryption. In: Security and trust management. Springer, pp 226–238
18. Gupta B, Agrawal DP, Yamaguchi S (2016) Handbook of research on modern crypto-graphic solutions for computer and cyber security IGI Global
19. He D, Kumar N, Lee JH (2016) Privacy-preserving data aggregation scheme against internal attackers in smart grids. Wirel Netw 22(2):491–502
20. Hoglund R, Tiloca M (2015) Current state of the art in smart metering security
21. Kumar V, Madria S (2012) Secure hierarchical data aggregation in wireless sensor networks: performance evaluation and analysis. In: 2012 IEEE 13th international conference on mobile data management (MDM). IEEE, pp 196–201
22. Lu R, Alharbi K, Lin X, Huang C (2015) A novel privacy-preserving set aggregation scheme for smart grid communications. In: 2015 IEEE global communications conference (GLOBECOM). IEEE, pp 1–6
23. Li F, Luo B (2012) Preserving data integrity for smart grid data aggregation. In: 2012 IEEE third international conference on smart grid communications (smartgridcomm). IEEE, pp 366–371
24. Li J, Li J, Chen X, Jia C, Lou W (2015) Identity-based encryption with outsourced revocation in cloud computing. IEEE Trans Comput 64(2):425–437
25. Li C, Lu R, Li H, Chen L, Chen J (2015) PDA: a privacy-preserving dual-functional aggregation scheme for smart grid communications security and communication networks
26. Lu R, Liang X, Li X, Lin X, Shen X (2012) EPPA: An efficient and privacy-preserving aggregation scheme for secure smart grid communications. IEEE Trans Parallel Distrib Syst 23(9):1621–1631
27. Memos VA, Psannis KE, Ishibashi Y, Kim BG, Gupta B (2017) An efficient algorithm for media-based surveillance system (EAM-Sus) in IoT Smart City framework Future Generation Computer Systems
28. Menezes AJ, Van Oorschot PC, Vanstone SA (1996) Handbook of applied cryptography. CRC Press
29. Mustafa MA, Zhang N, Kalogridis G, Fan Z (2015) MUSP: multi-service, user self- controllable and privacy-preserving system for smart metering. In: 2015 IEEE international conference on communications (ICC). IEEE, pp 788–794
30. Ni J, Zhang K, Alharbi K, Lin X, Zhang N, Shen X (2017) Differentially private smart metering with fault tolerance and range-based ltering. IEEE Trans Smart Grid pp(99):1–1
31. Paillier P (1999) Public-key cryptosystems based on composite degree residuosity classes. In: Advances in Cryptology EUROCRYPT'99. Springer, pp 223–238
32. Saputro N, Akkaya K (2012) Performance evaluation of smart grid data aggregation via homomorphic encryption. In: 2012 IEEE on wireless communications and networking conference (WCNC). IEEE, pp 2945–2950
33. Shen H, Zhang M, Shen J (2017) Efficient privacy-preserving cube-data aggregation scheme for smart grids. IEEE Trans Inf Forensics Secur 12(6):1369–1381
34. Shi E, Chan THH, Rieffel EG, Chow R, Song D (2011) Privacy-preserving aggregation of time-series data
35. Tonyali S, Akkaya K, Saputro N, Uluagac AS, Nojoumian M (2017) Privacy-preserving protocols for secure and reliable data aggregation in IoT-enabled smart metering systems. Futur Gener Comput Syst. https://doi.org/10.1016/j.future.2017.04.031
36. Wang H, Wang Z, Domingo-Ferrer J (2017) Anonymous and secure aggregation scheme in fog-based public cloud computing. Futur Gener Comput Syst pp
37. Zhang J, Liu L, Cui Y, Chen Z (2013) SP2DAS: self-certified PKC-based privacy-preserving data aggregation scheme in smart grid. Int J Distrib Sens Netw 9(1):457325