BIG DATA MINING AND ANALYTICS
ISSN 2096-0654 09/15 pp131–141
DOI: 10.26599/BDMA.2023.9020008
Volume 7, Number 1, March 2024

Multi-Smart Meter Data Encryption Scheme Based on Distributed Differential Privacy

Renwu Yan*, Yang Zheng, Ning Yu, and Cen Liang

Abstract: Under the general trend of the rapid development of smart grids, data security and privacy are facing serious challenges; protecting the privacy data of single users under the premise of obtaining user-aggregated data has attracted widespread attention. In this study, we propose an encryption scheme on the basis of differential privacy for the problem of user privacy leakage when aggregating data from multiple smart meters. First, we use an improved homomorphic encryption method to realize the encryption aggregation of users' data. Second, we propose a double-blind noise addition protocol to generate distributed noise through interaction between users and a cloud platform to prevent semi-honest participants from stealing data by colluding with one another. Finally, the simulation results show that the proposed scheme can encrypt the transmission of multi-intelligent meter data under the premise of satisfying the differential privacy mechanism. Even if an attacker has enough background knowledge, the security of the electricity information of one another can be ensured.

Key words: smart grid; homomorphic encryption; data aggregation; differential privacy; cloud computing

1 Introduction

With the popularization of smart grids, the two-way communication technology between smart meters and power systems has developed rapidly[1]. The massive user electricity data collected by smart meters provide a reliable basis for power systems to achieve short-term load forecasting, price regulation, and differentiated and personalized electricity service[2]. However, due to the close connection between smart grids and daily lives, power data may be intercepted during transmission by illegal attackers, resulting in user privacy information leakage[3]. Attackers can infer users' work and rest habits by collecting several power data of users in each period, which may expose their privacy. Despite huge electricity data, power systems also face problems such as insufficient data computing and storage capacity, insufficient resource utilization, large investment in information technology facilities, and complex system management. These factors are essentially hindering smart grid development. Thus, the problem of massive user data aggregation and privacy problems in the analysis process must be solved as soon as possible.

Cloud computing platforms have been widely used in the research on data aggregation in recent years due to their strong computing power, scalable storage capacity, and substantial economic benefit[4]. However, using cloud storage for confidential data causes huge security risks. Suppose that users store confidential information in cloud services without privacy

• Renwu Yan, Yang Zheng, and Cen Liang are with School of Electronic, Electrical Engineering and Physics, Fujian University of Technology, Fuzhou 350118, China. E-mail: yrw2010@fjut.edu.cn; 369818737@qq.com; 1019311783@qq.com.
• Ning Yu is with Department of Computing Sciences at The College at Brockport, State University of New York, New York, NY 14420, USA. E-mail: nyu@brockport.edu.
* To whom correspondence should be addressed.
Manuscript received: 2022-10-31; revised: 2023-04-25; accepted: 2023-04-26

protection, their information can be copied or peeked at, and the value of confidential information is greatly reduced or even completely lost. Even if data can be encrypted, leaking decrypted data to cloud service providers in the decryption stage is still a risk. Moreover, if attackers obtain some background data, then only data encryption and aggregation are insufficient to protect the privacy of honest users. Nevertheless, adopting homomorphic encryption in a cloud platform is a feasible solution to handle this problem. The breakthrough of homomorphic encryption technology can solve the critical security problems of cloud services according to the latest literature[5].

Although many privacy protection schemes have emerged, they have a common drawback, which is dependent on the attacker's background knowledge and does not make reasonable assumptions on attack models[6]. In some specific cases, if attackers have the private data of some users, then the sum of data minus the background data of attackers is used to differentially attack the aggregated data, and the risk of data privacy disclosure of the remaining users increases[7]. The differential privacy model proposed by Dwork et al.[8] in 2006 can solve this problem. The differential privacy protection model can ensure that inserting or deleting a record in a dataset does not affect the output of any calculation. Compared with the traditional privacy protection model, the differential privacy protection model does not need to rely on how much background knowledge an attacker has and provides high-level semantic security for privacy information. Therefore, it is widely used as a new privacy protection model.

In this research, our contributions are summarized as follows:

(1) We describe an improved homomorphic encryption method that aggregates multiple smart meter data in an upload message. Even if certain encryption data are cracked, subsequent smart meter data can still be transmitted securely.

(2) We propose a double-blind noise addition protocol that satisfies differential privacy. This protocol ensures that each participant cannot know the amount of noise added to real data. Even if data attackers have sufficient background knowledge or dishonest participants collude with one another, they cannot obtain the smart meter data of certain users.

2 Related Work

The privacy protection of electricity data in smart grids is mainly carried out from two aspects. One is user identity privacy protection, and the other is user electricity data protection.

Identity anonymity technology is one of the key technologies to achieve identity privacy protection. The basic idea is to use K-anonymity, blind signature, ring signature, and other anonymous technologies to protect user privacy so that attackers cannot associate data with user identity even if they steal user smart meter data. Sweeney[9, 10] proposed a K-anonymous privacy protection technology that provides a good property for a dataset. This technology can make every individual information contained in the anonymous dataset not be distinguished from other $K-1$ individual information. However, K-anonymity has no random attributes, so attackers can still infer individual-related privacy information from the dataset satisfying the K-anonymity property. Yu et al.[11] proposed a ring signature-based scheme to protect user electricity data. However, the computational cost increases with the loop size and is inapplicable in the actual scenario. Zhang et al.[12] proposed a certificateless ring signature scheme based on a trusted third party. Electricity data are associated with users at a trusted third party to achieve user billing while protecting privacy. However, if the trusted third party is compromised, then it may still cause user electricity data leakage.

Aggregation technology is a relatively practical method for electrical data protection at this stage. The basic idea is to hide electrical data through an aggregation method. Even if an attacker recognizes a user identity, he is obstructed from acquiring the users' accurate electrical data to achieve privacy protection. The most commonly used method is homomorphic encryption, which aggregates and encrypts the data sent by users. However, simply using encryption to protect user privacy can still disclose users' privacy in some specific cases. Chen et al.[13] proposed a privacy-preserving data aggregation scheme with fault tolerance. Their scheme supports customer data protection against an adversary that can compromise servers. Smart meter data are encrypted through Paillier encryption. Garcia and Jacobs[14] proposed a privacy-preserving aggregation protocol. Here, users' data are first divided into several parts, and then the corresponding public key is used to encrypt the users' data, and then sent them to the aggregator. After

receiving the encrypted value, the aggregator decrypts the encrypted value using homomorphic encryption and decryption key properties to obtain the sum of users' data. Ni et al.[15] used homomorphic encryption to achieve privacy protection. This method has less computational overhead than some existing aggregation methods and does not introduce a third-party platform. However, it cannot guarantee that honest users' data are satisfied with differential privacy. Similarly, He et al.[16] only used simple encryption aggregation without considering differential privacy. In their scheme, although the aggregator can only obtain the sum of, if the attacker subtracts the background data from the aggregate data to perform differential attacks on the aggregate data, then users' data security is unguaranteed.

Considering the above problems, Ács and Castelluccia[17] used differential privacy protection and encryption aggregation to achieve secure multiparty communication for ensuring differential user privacy. The authors of the scheme in Ref. [18] claimed that they achieved differential privacy by introducing an O (1) error in aggregation activity accuracy when failed smart meters exist. Chan et al.[19] used a geometric distribution to add noise to perturb metering data. During the decryption process, if the noise can be eliminated from each other, then the final estimate contains a noise of roughly O (log n). In the scheme in Ref. [20], to achieve differential privacy, noise is added from a geometric distribution to aggregate data at the gateway level. The authors calculated the root mean square errors for all smart meters, and the malfunctioning smart meter and claimed that their proposed scheme achieves better utility with lower errors. The binomial distribution is used in the scheme presented in Ref. [21] to achieve differential privacy. Every smart meter perturbs its data with generated noise and encrypts them with its private key. Given that the aggregator's motivation is to obtain total power consumption as accurately as possible, smart meters generate distributed noise that satisfies the differential privacy mechanism. However, their scenarios assume that smart meters are honest, no collusion exists among smart meter users, and attackers do not subtract added noise from aggregated data. Unfortunately, if semi-honest participants collude with each other, then they can remove noise from the total power consumption, thereby reducing the privacy protection levels of honest participants.

3 Preliminary

3.1 Paillier homomorphic encryption

The Paillier encryption system was invented by Pascal Paillier and is a probabilistic asymmetric algorithm based on a decision compound residual problem. The Paillier cryptosystem is widely used in privacy protection because it can realize additive homomorphic encryption, and its security is based on the problem of determining the remaining compounds.

The homomorphic characteristics of Paillier cryptosystem are as follows: After encryption, the corresponding arithmetic operation can be directly performed on the ciphertext. The calculation results are consistent with the corresponding calculation results in the plaintext domain after decryption[22]. Therefore, it can be widely used in data aggregation schemes.

The encryption system comprises key generation, encryption, and decryption algorithms. The process is as follows:

(1) Key generation: Choose two large prime numbers p and q randomly, which are independently of each other, such that

$$\gcd(p \cdot q,\ (p-1)\cdot(q-1)) = 1 \tag{1}$$

where $\gcd(\ )$ stands for the greatest common divisor. This property is assured if both primes are of equal lengths.

Compute least common multiple λ as follows:

$$\lambda = \operatorname{lcm}(p-1, q-1) \tag{2}$$

where $\operatorname{lcm}(\ )$ stands for the least common multiple.

Select a random integer g, and define the variable μ as follows:

$$\mu = \left(L\left(g^{\lambda} \bmod n^2\right)\right)^{-1} \bmod n \tag{3}$$

where $n = p \cdot q$, $g \in \mathbf{Z}^*_{n^2}$, The order of n divided by g is determined by the inverse of modular multiplcation, and function $L(\ )$ is defined as follows:

$$L(u) = \frac{u-1}{n} \tag{4}$$

Finally, the public key is (n, g), and the secret key is (λ, μ).

(2) Encryption: Use M as a plaintext to be encrypted, where $M \in \mathbf{Z}^*_n$. Use c as the ciphertext to decrypt, where $c \in \mathbf{Z}^*_{n^2}$. Select a random r, where $r \in \mathbf{Z}^*_n$. Compute ciphertext as follows:

$$c = E(M) = g^M \cdot r^n \bmod n^2 \tag{5}$$

where $E(\cdot)$ is the encryption function.

(3) Decryption: Compute the plaintext message as follows:

$$M = D(c) = L\left(c^{\lambda \bmod n^2}\right) \cdot \mu \bmod n \tag{6}$$

where $D(\,)$ is the decryption function.

Homomorphic addition of plaintexts: The following two ciphertexts M_1 and M_2 are given:

$$E(M_1) = g^{M_1} \cdot r_1^n \bmod n^2 \tag{7}$$

$$E(M_2) = g^{M_2} \cdot r_2^n \bmod n^2 \tag{8}$$

where r_1 and r_2 are randomly chosen from $\mathbf{Z}_n^*$, and $M_1, M_2 \in \mathbf{Z}_n^*$.

The product of two ciphertexts is decrypted as the sum of their corresponding plaintexts,

$$D(E(M_1) \times E(M_2)) \bmod n^2 = (M_1 + M_2) \bmod n \tag{9}$$

because

$$\begin{aligned} E(M_1) \times E(M_2) &= \left(g^{M_1} \cdot r_1^n\right)\left(g^{M_2} \cdot r_2^n\right) \bmod n^2 = \\ & g^{M_1+M_2} (r_1 \cdot r_2)^n \bmod n^2 = \\ & E(M_1 + M_2) \end{aligned} \tag{10}$$

3.2 Differential privacy

The essence of the differential privacy protection model is to ensure that the operation of inserting or deleting a record in a dataset does not affect the output of any calculation. Compared with the traditional privacy protection model, the differential privacy protection model is independent of the attacker's background knowledge. At the same time, it has a rigorous statistical model that can provide quantifiable privacy guarantees, making the privacy protection levels provided by datasets under different parameters processing comparable.

Differential privacy can be formally defined as follows:

Definition 1 (ε-differential privacy) The difference between any two adjacent datasets D_1 and D_2 is at most one record. If random function A satisfies ε-differential privacy protection, Range (A) represents the range of random function A, then for all variables $S \subseteq \text{Range}(A)$,

$$\Pr[A(D_1) \in S] \leqslant e^{\varepsilon} \times \Pr[A(D_2) \in S] \tag{11}$$

where $\Pr[\,]$ represents the disclosure risk of the event, and privacy budget ε is the parameter that controls the privacy protection level. When ε is smaller, the probability distribution of the query results returned by the differential privacy algorithm acting on a pair of adjacent datasets is more similar. The attacker has a more difficult time to distinguish this pair of adjoining datasets, and the degree of protection is relatively high.

The output probability of adjacent datasets is show in Fig. 1. Adding a specific distribution of noise to the output of the original dataset, the probability of the statistical output of the original dataset D and adjacent dataset D', which differs from a single protected record, is limited to the set range. Even if attackers know all the background knowledge about D and D', according to Definition 1, the privacy disclosure probability does not exceed e^{ε}, which mathematically defines the upper limit of privacy and the possibility of privacy disclosure.

Definition 2 (Global sensitivity) The form of query function $f: D \to R$, where R is the return result of the query function. The global sensitivity on a pair of arbitrary adjacent datasets D_1 and D_2 is defined as follows:

$$\Delta f = \max_{D_1, D_2} \|f(D_1) - f(D_2)\| \tag{12}$$

where $\|f(D_1) - f(D_2)\|$ is the Manhattan distance (first-order norm distance) between $f(D_1)$ and $f(D_2)$.

Global sensitivity reflects the maximum range of changes when a query function queries on a pair of adjacent dataset, it is independent of the datasets and is determined only by the query function.

Definition 3 (Laplace mechanism) Given a query function f, the input D satisfies

$$A(D) = f(D) + \xi \tag{13}$$

Then A satisfies ε differential privacy protection, where ξ is the random noise subject to Laplace distribution, x is the random variable, namely

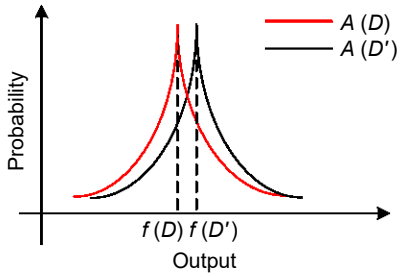

Fig. 1 Output probability of the differential privacy algorithm on adjacent datasets.

$\xi \sim \mathrm{Lap}(\lambda) = \frac{1}{2\lambda} e^{-\frac{|x-\mu|}{\lambda}}$, $\lambda = \Delta f/\varepsilon$, and Δf is the global sensitivity of the dataset.

Lemma 1 (Infinite separability of Laplace distribution[23]) For any random variable, $\mathrm{Lap}(\lambda)$ is subject to the probability density distribution of Laplace distribution, its distribution is infinitely separable, for any integer $m \geqslant 1$, satisfying

$$\mathrm{Lap}(\lambda) = \sum_{i=1}^{m} [G_1(m,\lambda) - G_2(m,\lambda)] \tag{14}$$

where $G_1(m,\lambda)$ and $G_2(m,\lambda)$ are random variables of independent, and identically distribute gamma distribution.

In summary, the Laplacian mechanism is a simple and widely used privacy protection mechanism for numerical queries. For numeric query results, the Laplace mechanism implements differential privacy protection by returning a query result with noise satisfying the $\mathrm{Lap}(\lambda)$ distribution.

4 Our Scheme

4.1 Data aggregation model

In the smart grid data aggregation model, we default that each user is equipped with a smart meter, which sends the measured electricity data to the cloud platform at the end of each electricity cycle. A typical transmission network model of the star network is adopted in this scheme, as shown in Fig. 2. Our model comprises a Control Center (CC), a cloud platform, and m users.

We use SM_i $(i = 1, 2, \ldots, m)$ to represent smart meters. SM_i can perform simple encryption operations, consistent with most literature. We use d_i to represent the power data of smart meter i and $d = \sum_{i=1}^{m} d_i$ to represent the total data of all users.

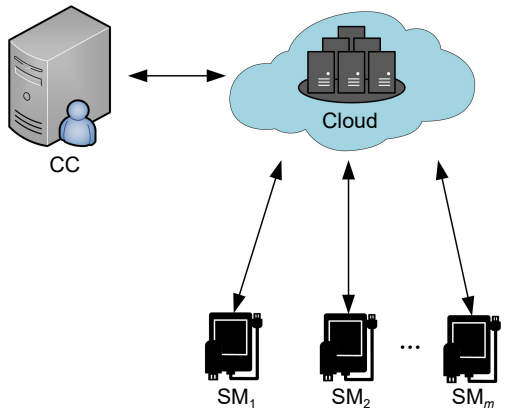

Fig. 2 Star network transmission model.

Considering the limited computing power of smart meters in smart grid scenarios, each smart meter contributes only one noise during the noise generation process. Smart meters send noise-added data to cloud platforms, which aggregates this data through homomorphic encryption.

We assume that each participant (smart meter and cloud platform) is semi-honest. We set semi-honest participants to abide by the following rules: They will follow the rules of the agreement, but they try to obtain sensitive information from other participants based on the information they see in each step of the agreement. Therefore, semi-honest participants have the incentive to add less noise (or even no noise) to the data, leading to a low security level of differential privacy for users. Thus we cannot trust noise addition from the semi-honest participants to satisfy differential privacy. We choose to use a double-blind protocol approach to add distributed noise and ensure that the aggregated data satisfy differential privacy.

4.2 Specific scheme

We design a data aggregation scheme on the basis of homomorphic encryption to meet differential privacy protection and use the double-blind protocol approach to realize the differential privacy mechanism. Users and cloud platform cooperate to participate in the double-blind protocol. A binary sequence is generated in the cloud platform and sent to users. These users generate the corresponding random noise sample, and then a noise item is randomly selected from the noise sample by the binary sequence to the data. Therefore, neither the cloud platform nor users can know the contribution of each user. Thus, even if there may be a collusion between users and the cloud platform, the protection level of differential privacy cannot be reduced.

4.2.1 Initialization phase

CC generates parameters $\{n, p, q, \lambda, \mu, g\}$ for the cryptographic system. The public key is $\mathrm{pk} = (n, g)$, and the secret key is $\mathrm{sk} = (\lambda, \mu)$. CC publishes the public key to all other entities and keeps the secret key.

4.2.2 Registration phase

Registered messages are sent through dedicated and secure channels. SM_i obtains current timestamp t_i and calculates hash value $h_{(i)} = h(\mathrm{id}_i \| t_i)$, where id_i is the user number. SM_i sends the registration request

$(\mathrm{id}_i, t_i, h_i)$ to the CC.

CC receives this registration request. First, it verifies whether equation $h_{(i)} = h(\mathrm{id}_i \| t_i)$ is valid. If so, then it is verified and registered successfully. The cloud platform also registers with CC in the same way. Meanwhile, user hash value $h_{(i)}$ is sent to the cloud platform to provide SM_i authentication result to the cloud platform.

4.2.3 Binary sequence generation

After the cloud platform receives $h_{(i)}$ and authenticates the legal identity of SM_i, binary sequence $b_{i,j}$ is randomly generated, and the sequence satisfies

$$\sum_{i,j} b_{i,j} = m \tag{15}$$

$$\sum_{j=1}^{\gamma} b_{i,j} = 1 \tag{16}$$

where $i = 1, 2, \ldots, m$ and $j = 1, 2, \ldots, \gamma$.

The above two formulas show that for a binary sequence received by a single smart meter user, only one $b_{i,j} = 1$ exists, and the remaining $b_{i,j}$ is 0. Here, γ is a security parameter, and the greater the n, the higher the security degree. CC encrypts the binary sequence to obtain $E_{vA}(b_{i,j})$ and sends it to SM_i, $E_{vA}(\,)$ is used to encrypt binary sequences.

4.2.4 Encryption process

Random number $k_{(i)}$ is generated and sent to SM_i by a trusted third party. For m smart meter users, $k_{(i)}$ must satisfy

$$k_{(1)} + k_{(2)} + \cdots + k_{(m)} = 0 \tag{17}$$

After SM_i receives $k_{(i)}$, $R_{(i)} = \gamma + k_{(i)}$ is calculated by combining the security parameter γ. At the same time, the blind term x_i used to hide data is randomly generated to satisfy $\sum_i^m x_i = x$. The public key is used to encrypt and obtain $E_{pk}(d_i - x_i)$,

$$E_{pk}(d_i - x_i) = g^{d_i - x_i} \cdot h_{(i)}^{R} \cdot R_{(i)} = g^{d_i - x_i} \cdot h_{(i)}^{\gamma + k_{ij}} \tag{18}$$

where $E_{pk}(\,)$ is used to encrypt data, and then SM_i sends the encrypted data to the cloud platform,

$$\prod_{i=1}^{m} E_{pk}(d_i - x_i) = \prod_{i=1}^{m} g^{d_i - x_i} \cdot h_{(i)}^{\sum_i^m R(i)} = g^{\sum_i^m d_i - x_i} \cdot h_{(i)}^{\sum_i^m R(i)} \tag{19}$$

According to Eq. (17), $\sum_i^m R_{(i)} = \sum_i^m \gamma + \sum_i^m k_{(i)} = m\gamma$, the encrypted data (including blind items) of all m users can be obtained as follows:

$$\prod_{i=1}^{m} E_{pk}(d_i - x_i) = g^{\sum_i^m (d_i - x_i)} \cdot \left(h_{(i)}^{m}\right)^{\gamma} = E_{pk}\left(\sum_i^m d_i - x_i\right) = E_{pk}(d - x) \tag{20}$$

4.2.5 Distributed Laplace noise generation

In our scheme, noise is generated by a double-blind protocol, which makes the scheme satisfy the differential privacy mechanism. This ensures that each participant cannot know the amount of noise added to the real data.

According to Lemma 1, Laplace distribution can be constructed as the sum of multiple independent and identically distributed gamma distributions. Given the infinite separability of Laplace distribution, it applies to any number of users and has high scalability. Therefore, our scheme requires that the distributed noise satisfying differential privacy be added to the original data independently on each SM_i. The specific process is as follows:

Firstly, each smart meter user generates n noise samples $\xi_{i,j}$, as illustrated in Fig. 3.

Furthermore, after SM_i receives $E_{vA}(b_{i,j})$ from CC, n sub-blind items $x_{i,j}$ are generated randomly to satisfy

$$\sum_{j=1}^{\gamma} x_{i,j} = x_i \tag{21}$$

Similarly, SM_i uses the additive homomorphic characteristics of the Paillier homomorphic encryption algorithm to calculate the encryption variables $e_{i,j}$ as follows:

$$e_{i,j} = E_{vA}\left(b_{i,j}\right)^{\xi_{i,j}} \cdot E_{vA}\left(x_{i,j}\right) = E_{vA}\left(\xi_{i,j} \cdot b_{i,j} + x_{i,j}\right) \tag{22}$$

Subsequently, SM_i sends $e_{i,j}$ to the cloud platform.

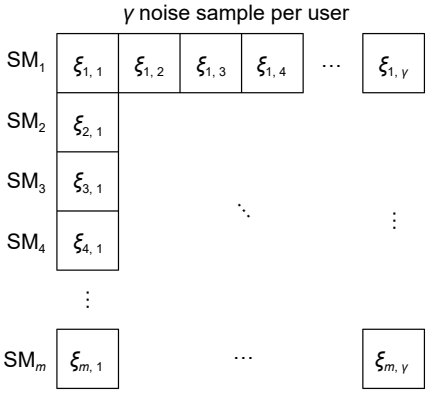

Fig. 3 Schematic of noise samples generated by users.

According to the characteristics of a binary sequence, when $b_{i,j}=1$, the encrypted data of noise contributed by SM_i can be obtained as follows:

$$\prod_{j=1}^{\gamma} e_{i,j}=\prod_{j=1}^{\gamma} E_{vA}\left(\xi_{i,j}\cdot b_{i,j}+x_{i,j}\right)=E_{vA}\left(\xi_i\cdot 1+x_i\right) \quad (23)$$

where ξ_i is the Laplace distributed noise satisfying Lap (λ), $\lambda=\Delta f/\varepsilon$, and Δf is the global sensitivity of the dataset[24]. According to the infinite separability of ξ, when $m\geqslant 1$, it satisfies the Laplace distribution,

$$\sum_{i=1}^{m}\xi_i=\text{Lap}\left(\Delta f/\varepsilon\right)=\xi \quad (24)$$

Finally, the encrypted data for all m users can be obtained as follows:

$$\prod_{i=1}^{m} E_{vA}\left(\xi_i+x_i\right)=E_{vA}\left(\sum_{i=1}^{m}\left(\xi_i+x_i\right)\right)=E_{vA}(\xi+x) \quad (25)$$

4.2.6 Decryption process

The cloud platform sends encrypted data $E_{vA}(\xi+x)$ and $E_{pk}(d-x)$ to CC. The decryption process is as follows:

Let $E_{vA}\left(\xi+x\right)=C_1$ and $E_{pk}\left(d-x\right)=C_2$, CC decrypts C_1 and C_2 with the secret key to obtain $\xi+x$ and $d-x$, respectively,

$$\xi+x=D(C_1)=L\left(C_1^{\lambda \bmod n^2}\right)\cdot\mu \bmod n \quad (26)$$

$$d-x=D(C_2)=L\left(C_2^{\lambda \bmod n^2}\right)\cdot\mu \bmod n \quad (27)$$

Add the following two plaintext data:

$$d-x+\xi+x=d+\xi \quad (28)$$

where $d+\xi$ is the aggregated data of m users, which satisfies differential privacy requirements.

5 Privacy and Security Analysis

5.1 Security analysis

In our scheme, the data d_i of SM_i are encrypted using the improved Paillier homomorphic encryption algorithm. The original Paillier algorithm uses a random number $r\in\mathbf{Z}_n^*$ to achieve semantic security. In our scheme, we randomize the form of $R_{(i)}$. We generate the hash value of timestamp $h\left(\text{id}_i\|t_i\right)$ instead of r.

The improved Paillier homomorphic encryption algorithm makes the encrypted transmission further secure because $R_{(i)}$ of each SM_i is different. Even if the encryption is cracked at a certain time, the subsequent data of SM_i can still be transmitted safely.

5.2 Privacy analysis

In the multirole security data aggregation problem model study, a typical assumption is that the third-party platform is semi-honest. In our model scenario, the smart meter is also semi-honest. We assume that semi-honest participants abide by the following rules: They all follow the rules of the protocol, but they can try to obtain sensitive information from other participants based on the information they obtain in each step of the protocol. Note that users and cloud platforms will not provide wrong or false input to the protocol, so the input information is true.

As shown in Fig. 4, we divide the collusion of semi-honest participants into two scenarios for discussion:

(1) Only semi-honest smart meters collude, as illustrated in Fig. 4a. The cloud platform follows the protocol, and l smart meters are honest; that is, the remaining $m-l$ smart meters collude with one another to form a collusion. In this scenario, semi-honest users may use their own electricity consumption and noise to obtain the real electricity consumption of honest users. However, the amount of noise added to honest smart meters may not guarantee sufficient differential privacy.

(2) We also consider the scenario of collusion between smart meters and cloud platforms, as displayed in Fig. 4b.

5.2.1 Single (no collusion) semi-honest user

For smart meter, without collusion, SM_i only obtains the encrypted $b_{i,j}$, $j=1, 2, \ldots, \gamma$, so the single semi-honest user neither knows which $b_{i,j}$ is 1, nor how much noise he contributes to the total differential privacy noise. All he knows is his data d_i.

5.2.2 Semi-honest users collusion

Although smart meter users cannot communicate directly with each other, it cannot be ignored that some

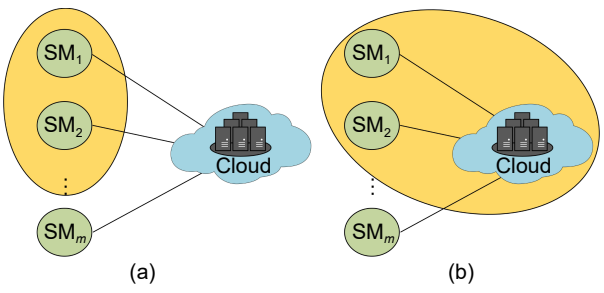

Fig. 4 Two scenarios of collusion between semi-honest participants. (a) Some semi-honest users collude with one another and (b) semi-honest users collude with the cloud platform.

semi-honest users collude by other means. For example, semi-honest users in collusion send their data to an external server, which can subtract users' data of smart meter in collusion from all users' data (with noise) published by CC, to obtain honest users' data. However, this scheme uses a double-blind noise protocol, which makes it impossible for collusive users to know the specific noise they are adding. Therefore, the noise-added data obtained after subtracting colluder data still ensure the privacy of honest users.

We can use S to represent the collection of colluding smart meter users and $\tilde{S}$ to represent the collection of honest smart meter users. Through data collusion, S can obtain the following information:

$$\sum_{i=1}^{m}(d_i+\xi_i)-\sum_{j\in S}d_j=\sum_{i\in\tilde{S}}d_i+\sum_{i=1}^{m}\xi_i \tag{29}$$

Evidently, colluding users cannot offset their own noise samples from the total data, so the result of Eq. (29) contains more noise than originally needed. Therefore, for $1\leqslant|S|\leqslant m-1$, semi-honest colluders cannot reduce the differential privacy protection level but make honest users' data more secure.

5.2.3 Semi-honest cloud platform

Although the semi-honest cloud platform knows $b_{i,j}$, it cannot know the exact noise added by each smart meter. The cloud platform receives the privacy data of a single user (including d_i-x_i and $e_{i,j}$). The analysis is as follows:

For d_i-x_i, due to the existence of randomized blind terms, the semi-honest cloud platform is unable to obtain a specific value of d_i.

For $e_{i,j}$, according to Eq. (22), When $b_{i,j}=1$, then $e_{i,j}=\xi_{i,j}+x_{i,j}$. Given that $x_{i,j}$ is a blind item generated randomly, the semi-honest cloud platform cannot obtain information about $\xi_{i,j}$. When $b_{i,j}=0$, $e_{i,j}=x_{i,j}$, which has no information about $\xi_{i,j}$.

Therefore, the semi-honest cloud platform cannot obtain the specific data of a single user.

5.2.4 Semi-honest cloud platform and users collusion

In this case, due to collusion between the semi-honest cloud platform and users, the colluding smart meter knows the value of $b_{i,j}$ and the value of the distributed noise that it has added. Through collusion, colluders obtain the following information:

$$\sum_{i=1}^{m}(d_i+\xi_i)-\sum_{j\in S}\left(d_j+\xi_j\right)=\sum_{i\in\tilde{S}}(d_i+\xi_i) \tag{30}$$

The amount of noise finally added to the honest smart meter is reduced and cannot satisfy the requirement of differential privacy. Theoretically, if we know which smart meters collude with the cloud platform, then we can generate enough distributed noise from $\tilde{S}$ to make the final noise accumulate and satisfy ε-differential privacy. However, implementing this defense strategy in practice is difficult because honest users often do not know how many semi-honest users collude with the cloud platform. Therefore, they cannot correctly choose the amount of noise they should contribute.

6 Simulation Analysis

We implement the above scheme in Java using the Paired Cryptography Library. This section conducts an evaluation using a 3.2 GHz Intel Pentium PC with 16 GB RAM.

6.1 Computational complexity analysis

We analyze the computational complexity of smart meters and cloud platforms from two aspects: the encryption and aggregation of electricity consumption and the aggregation of noise. For encrypted data aggregation, only one encryption is required per period to obtain the hash value of each cycle for smart meters. For the cloud platform, the encrypted measurement value of each collected smart meter must be multiplied $m-1$ times and decrypted once. Therefore, in terms of encryption and aggregation, the computational complexity of the cloud platform is $O(m)$, and that of the smart meter is $O(1)$. For the cloud platform, the binary sequence is generated and then encrypted and sent to each smart meter. Its computational complexity is $O(m\cdot\gamma)$, whereas the computational complexity of $e_{i,j}$ generated by the smart meter is $O(\gamma)$. Evidently, the greater the security parameter γ, the greater the security of the noise-adding protocol. Therefore, the selection of appropriate security parameters should be considered between security and protocol complexity. By contrast, in the data aggregation process of literature[25], the computational complexity of the aggregator is $O(m^2)$ and that of the smart meter is $O(m)$. The computational complexity of data aggregation in this solution is relatively low.

6.2 Time complexity simulation

We simulate the time complexity of the double-blind noise addition protocol of this scheme. Figures 5 and 6

show the influences of m and γ on the calculation time, respectively. In Fig. 5, γ is set to 4, and the number of users m is in the interval [100, 1000]. In Fig. 6, m is set to 500, and the safety parameter γ is in the interval [2, 20].

Figure 6 shows that the larger m and γ, the longer the encryption time of the smart meter. The cloud platform encryption operation aims to encrypt binary sequence $e_{i,j}$ ($i = 1, 2, \ldots, m$ and $j = 1, 2, \ldots, \gamma$). The larger m and γ, the longer the encryption time. The calculation overhead of smart meters is relatively small. When security parameter γ is set to 20, the calculation time is about 42 ms. Even if the number of users increases, the encryption calculation overhead of smart meters is still lightweight. For the cloud platform, when security parameter γ is set to 20, and the number of users is set to 500, the time of the encrypted binary sequence of the aggregator is about 4 s. The differential privacy aggregation mechanism designed in this chapter is lightweight for smart meters.

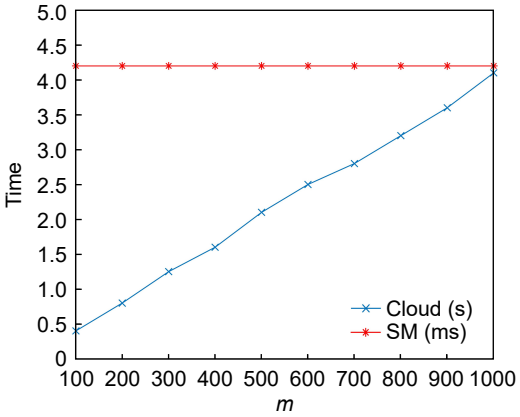

Fig. 5 Influences of different values of m on calculation time.

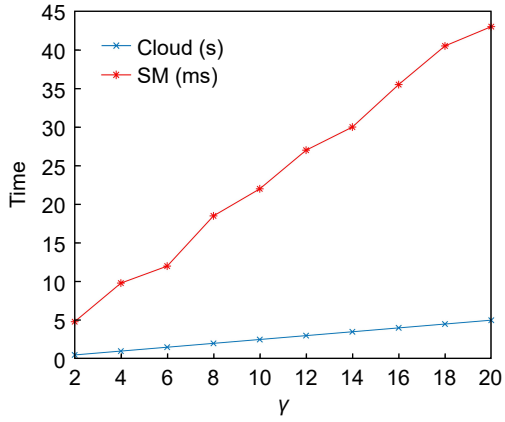

Fig. 6 Influences of different values of γ on calculation time.

6.3 Scheme comparison

We verify the efficiency of the solution in this article. Table 1 provides a comparison of time and Standard Deviation (SD) with the privacy protection methods for electricity data proposed in other documents. As presented in Table 1, compared with other schemes, our proposed scheme has advantages, especially in terms of computational performance.

Figure 7 shows that when the number of users is small, the efficiency of the three schemes is close. When the number of users exceeds 1200, our proposed scheme consumes less time than the two other schemes. Therefore, our proposed scheme has low computational overhead when the number of users is large and has obvious advantages.

Figure 8 compares the SD of the proposed scheme with those of WAV and DG-APED. In the case of various user numbers, the proposed scheme achieves the minimum SD, especially when the number of users reaches more than 800. Compared with the other two schemes, our proposed scheme has obvious advantages in SD, which indicates that the error rate of this scheme is lower than that of the other two schemes.

7 Conclusion

In this study, we propose a data encryption scheme for smart meters based on differential privacy. We improve the homomorphic encryption algorithm to realize the encryption aggregation of smart meter data.

Table 1 Comparisions of average execution time and SD.

Method	Time (ms)	SD
WAV[26]	0.3092	0.0356
DG-APED[27]	0.2679	0.0541
Our scheme	0.2198	0.0468

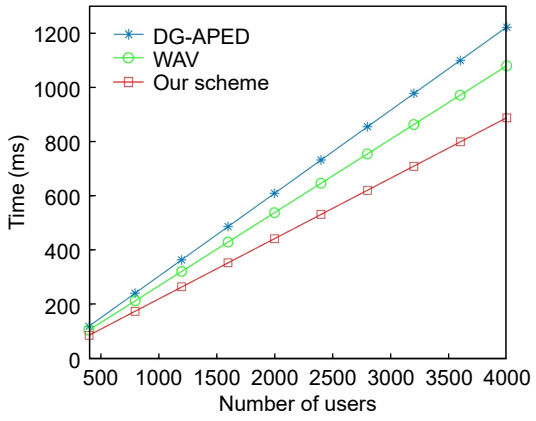

Fig. 7 Efficiency curves of different methods.

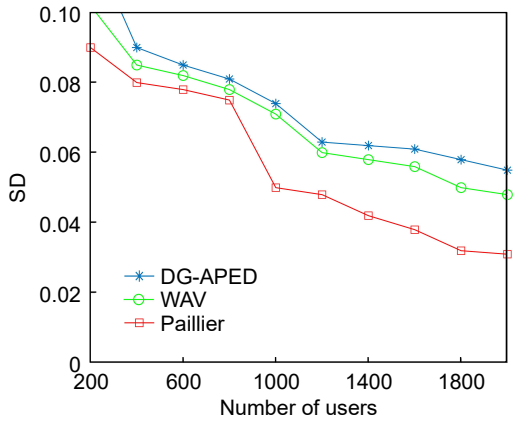

Fig. 8 SD comparison of different methods.

When certain encrypted data are cracked, subsequent smart meter data can still be transmitted securely. We also propose a double-blind noise addition protocol. We use a distributed noise addition approach to ensure that each participant cannot know the specific noise added to the original data. Even if collusion occurs among semi-honest participants, the protection level of the differential privacy mechanism cannot be reduced. Finally, the simulation analysis demonstrates that the proposed scheme has the following advantages:

(1) Higher computational efficiency. With the same number of users, the proposed scheme consumes less time than other schemes. Thus, the scheme has a significant advantage in computational performance.

(2) Better privacy protection. Compared with other schemes, the proposed scheme can encrypt and transmit multiple users' data. Even if an attacker has the maximum background knowledge, the privacy protection of each user's data can be ensured.

Acknowledgment

This work was supported by the National Natural Science Foundation of China (No. 51677059) and the Fujian Provincial University Engineering Research Center Open Fund (No. KF-D21009).

References

[1] M. Jusup, P. Holme, K. Kanazawa, M. Takayasu, I. Romić, Z. Wang, S. Gecek, T. Lipic, B. Podobnik, L. Wang, et al., Social physics, *Phys. Rep.*, vol. 948, pp. 1–148, 2022.

[2] D. Helbing, D. Brockmann, T. Chadefaux, K. Donnay, U. Blanke, O. Woolley-Meza, M. Moussaid, A. Johansson, J. Krause, S. Schutte, et al., Saving human lives: What complexity science and information systems can contribute, *J. Stat. Phys.*, vol. 158, no. 3, pp. 735–781, 2015.

[3] P. Kumar, Y. Lin, G. D. Bai, A. Paverd, J. S. Dong, and A. Martin, Smart grid metering networks: A survey on security, privacy and open research issues, *IEEE Commun. Surv. Tut.*, vol. 21, no. 3, pp. 2886–2927, 2019.

[4] Z. Tari, Security and privacy in cloud computing, *IEEE Cloud Comput.*, vol. 1, no. 1, pp. 54–57, 2014.

[5] J. Zhou, Z. F. Cao, X. L. Dong, and A. V. Vasilakos, Security and privacy for cloud-based IoT: Challenges, *IEEE Commun. Mag.*, vol. 55, no. 1, pp. 26–33, 2017.

[6] Z. Tari, X. Yi, U. S. Premarathne, P. Bertok, and I. Khalil, Security and privacy in cloud computing: Vision, trends, and challenges, *IEEE Cloud Comput.*, vol. 2, no. 2, pp. 30–38, 2015.

[7] S. J. Mohammed and D. B. Taha, Performance evaluation of RSA, ElGamal, and Paillier partial homomorphic encryption algorithms, in *2022 Int. Conf. Computer Science and Software Engineering* (*CSASE*), Duhok, Iraq, 2022, pp. 89–94.

[8] C. Dwork, Differential privacy, in *Proc. 33rd Int. Colloquium on Automata, Languages, and Programming,* Berlin, Germany, 2006, pp. 1–12.

[9] L. Sweeney, *K*-anonymity: A model for protecting privacy, *Int. J. Uncertain. Fuzz. Knowl. Based Syst.*, vol. 10, no. 5, pp. 557–570, 2002.

[10] L. Sweeney, Achieving *K*-anonymity privacy protection using generalization and suppression, *Int. J. Uncertain. Fuzz. Knowl. Based Syst.*, vol. 10, no. 5, pp. 571–588, 2002.

[11] C. M. Yu, C. Y. Chen, S. Y. Kuo, and H. C. Chao, Privacy-preserving power request in smart grid networks, *IEEE Syst. J.*, vol. 8, no. 2, pp. 441–449, 2014.

[12] S. Zhang, Y. Zhao, and B. Wang, Certificateless ring signcryption scheme for preserving user privacy in smart grid, *Automat. Electr. Power Syst.*, vol. 42, no. 3, pp. 118–135, 2018.

[13] L. Chen, R. Lu, and Z. Cao, PDAFT: A privacy-preserving data aggregation scheme with fault tolerance for Smart Grid communications, *Peer-to-Peer Netw. Appl.*, vol. 8, no. 6, pp. 1122–1132, 2015.

[14] F. D. Garcia and B. Jacobs, Privacy-friendly energy-metering via homomorphic encryption, in *Proc. 6th Int. Workshop on Security and Trust Management,* J. Cuellar, J. Lopez, G. Barthe, and A. Pretschner, eds. Athens, Greece: Springer, 2010, pp. 226–238.

[15] J. Ni, K. Zhang, X. Lin, and X. S. Shen, EDAT: Efficient data aggregation without TTP for privacy-assured smart metering, in *Proc. 2016 IEEE Int. Conf. Communications* (*ICC*), Kuala Lumpur, Malaysia, 2016, pp. 1–6.

[16] D. He, N. Kumar, S. Zeadally, A. Vinel, and L. T. Yang, Efficient and privacy-preserving data aggregation scheme for smart grid against internal adversaries, *IEEE Trans. Smart Grid*, vol. 8, no. 5, pp. 2411–2419, 2017.

[17] G. Ács and C. Castelluccia, I have a dream! (differentially private smart metering), in *Proc. 13th Int. Workshop on Information Hiding*, T. Filler, T. Pevný, S. Craver, and A. Ker, eds. Prague, Czech Republic: Springer, 2011, pp. 118–132.

[18] M. Jawurek and F. Kerschbaum, Fault-tolerant privacy-preserving statistics, in *Proc. 12th Int. Symp. Privacy*

Enhancing Technologies Symp., S. Fischer-Hübner and M. Wright, eds. Vigo, Spain: Springer, 2012, pp. 221–238.
[19] T. H. H. Chan, E. Shi, and D. Song, Privacy-preserving stream aggregation with fault tolerance, in *Proc. 16th Int. Conf. Financial Cryptography and Data Security*, A. D. Keromytis, ed. Kralendijk, the Netherlands: Springer, 2012, pp. 200–214.
[20] H. Bao and R. Lu, A lightweight data aggregation scheme achieving privacy preservation and data integrity with differential privacy and fault tolerance, *Peer-to-Peer Netw. Appl.*, vol. 10, no. 1, pp. 106–121, 2017.
[21] H. Liu, J. Chen, L. Lin, A. Ye, and C. Huang, An efficient and privacy-preserving data aggregation scheme supporting arbitrary statistical functions in IoT, *China Commun.*, vol. 19, no. 6, pp. 91–104, 2022.
[22] P. Paillier, Public-key cryptosystems based on composite degree residuosity classes, in *Proc. Int. Conf. Theory and Application of Cryptographic Techniques on Advances in Cryptology-EUROCRYPT'99*, Prague, Czech Republic, 1999, pp. 223–238.
[23] C, Dwork, The promise of differential privacy: A tutorial on algorithmic techniques, in *Proc. 52nd Annual Symp. Foundations of Computer Science,* Palm Springs, CA, USA, 2011, pp. 1–2.
[24] F. Kemp, The Laplace distribution and generalizations: A revisit with applications to communications, economics, engineering, and finance, *J. Roy. Stat. Soc. Ser. D Stat.*, vol. 52, no. 4, pp. 698–699, 2003.
[25] V. Bindschaedler, S. Rane, A. E. Brito, R. Alejandro, V. Rao, and E. Uzun, Achieving differential privacy in secure multiparty data aggregation protocols on star networks, in *Proc. Seventh ACM on Conf. Data and Application Security and Privacy*, Scottsdale, AZ, USA, 2017, pp. 115–125.
[26] D. Engel, Wavelet-based load profile representation for smart meter privacy, in *Proc. 2013 IEEE PES Innovative Smart Grid Technologies*, Washington, DC, USA, 2013, pp. 1–6.
[27] Z. Shi, R. Sun, R. Lu, L. Chen, J. Chen, and X. S. Shen, Diverse grouping-based aggregation protocol with error detection for smart grid communications, *IEEE Trans. Smart Grid*, vol. 6, no. 6, pp. 2856–2868, 2015.

Renwu Yan received the PhD degree in electrical machinery and electrical appliance from Fuzhou university, China in 2010. He is an associate professor at Fujian University of Technology, China. His main research interest is artificial intelligence applications in power system.

Ning Yu received the PhD degree in computer science from Georgia State University, USA in 2016. He is currently working as an associate professor at Department of Computing Sciences at The College at Brockport, State University of New York, USA. He has published more than 20 papers in prestigious conferences and journals. His research areas include artificial intelligence, network and information security, big data and data analytics, deep learning, and high performance computing.

Yang Zheng received the BEng degree from Henan University of Technology, China in 2018. He is currently a master student at Fujian University of Technology, China. His main research interest is cyber security in smart grid.

Cen Liang received the BEng degree from Fujian University of Technology, China in 2020. He is currently a master student at Fujian University of Technology, China. His main research interest is privacy protection of smart grid.