Hindawi
Security and Communication Networks
Volume 2022, Article ID 6218094, 12 pages
https://doi.org/10.1155/2022/6218094

Research Article

E-LPDAE: An Edge-Assisted Lightweight Power Data Aggregation and Encryption Scheme

Junhua Wu,[1] **Zhuqing Xu**,[1] **Guangshun Li**,[1] **Cang Fan**,[1] **Zhenyu Jin**,[1] **and Yuanwang Zheng**[2]

[1]*School of Computer Science, Qufu Normal University, Rizhao 276826, China*
[2]*Shandong Huatong Used Car Information Technology Limited Company, Jining 272000, China*

Correspondence should be addressed to Guangshun Li; guangshunli@qfnu.edu.cn

Received 6 January 2022; Revised 1 March 2022; Accepted 15 March 2022; Published 18 April 2022

Academic Editor: Jinbo Xiong

In smart grid systems, electric utilities require real-time access to customer electricity data; however, these data might reveal users' private information, presenting opportunities for edge computing to encrypt the information while also posing new challenges. In this paper, we propose an Edge-assisted Lightweight Power Data Aggregation Encryption (E-LPDAE) scheme for secure communication in a smart grid. First, in the edge privacy aggregation model, the data of smart meters are rationally divided and stored in a distributed manner using simulated annealing region division, and the edge servers of trusted organizations perform key one-time settings. The model encrypts the data using Paillier homomorphic encryption. It then runs a virtual name-based verification algorithm to achieve identity anonymization and verifiability of the encrypted data. The experimental results indicate that the E-LPDAE scheme reduces overall system power consumption and has significantly lower computation and communication overhead than existing aggregation schemes.

1. Introduction

In recent years, with the rapid development of modern science and technology and urbanization, the combination of power systems and information technology has produced a new concept—Smart Grid [1]. Smart Grid is the intelligence of the power grid. Building a smart grid can optimize resource allocation, reduce consumption, and increase efficiency. In smart grid applications, smart meters are deployed in all households in a residential area, each smart meter can collect the user's electricity consumption data and report it to the control center periodically (for example, every 15 minutes), and the control center can perform actions based on the reported data and real-time data analysis and take corresponding measures to ensure the health of the power system. Therefore, in the process of data transmission, a large number of real-time electricity consumption data of users is interacted with and calculated on the transmission line [2].

By using container technology, edge computing [3] is able to collect heterogeneous data in real time across a wide range of devices and can provide elastic computing resources for deep learning models. The resource configuration of edge computing can satisfy offline processing and analysis of small-area data, thereby ensuring the safe transmission and processing of various data. In addition, edge computing can reduce network latency and improve the utilization of network transmission bandwidth with the help of high-speed communication technology. In the implementation process of smart grid, the introduction of edge computing has a good development prospect, as shown in Figure 1.

Interaction and calculation of real-time electricity consumption provide a great convenience for power companies to fully grasp the electricity consumption of their customers but, at the same time, pose serious security and privacy risks. As pointed out by the National Institute of Standards and Technology (NIST) in the United States, there are more and richer data in smart grid systems. While bringing convenience to services, data leakage will also bring many security threats. Once the real-time electricity consumption information is stolen by the attacker, through the

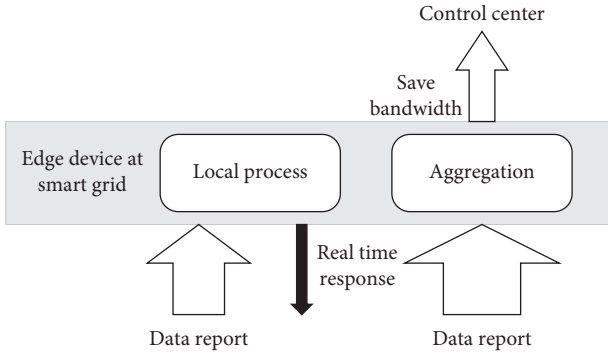

Figure 1: The edge computing paradigm extends cloud computing capabilities to the edge of the network to provide real-time response to local processes, as well as aggregated bandwidth savings.

analysis of the data, the user's detailed family life habits and other information can be obtained. Therefore, how to protect user privacy and data security in smart grids has become a research hotspot in recent years [4].

In order to overcome the above challenges, we propose an edge-assisted lightweight power data aggregation and encryption scheme. The main contributions of this paper are summarized as follows:

(i) An edge privacy aggregation model is proposed. The model uses simulated annealing (SA) to propose a segmentation algorithm for smart meters, Simulated Annealing Region Division (SARD). The algorithm can generate optimal area division according to the energy consumption of electricity meters, which is convenient for data collection and analysis of cluster electricity meters. The realization of distributed data storage is conducive to the privacy protection of smart meter data.

(ii) The Trusted Organization (TO) can set all keys in the system at one time, improve the efficiency of the smart grid, and reduce the power consumption of the system. Since a trusted organization stores a large amount of sensitive information such as keys, if it is stolen by an attacker, it will seriously threaten the data privacy and security of users. Such issues can be resolved by using edge servers, which are relatively trustworthy.

(iii) A virtual name-based authentication algorithm is proposed. The algorithm uses an encryption mechanism combining chameleon signature and Paillier cryptography to encrypt and verify the data to ensure the security of transmitted data while reducing the communication overhead; a selection strategy is developed using an attribute decision tree to improve the value of the data. Finally, the aggregated encrypted data is sent to the Cloud Power Distribution Center (CPDC). The CPDC decrypts the data in order to obtain the final result.

The rest of this paper is organized as follows. Section 2 summarizes the related work. In Section 3, we do some preparatory work. In Section 4, we describe the procedure and algorithm of the scheme. The results of the experimental analysis are reported in Section 5. Finally, the conclusions are discussed in Section 6.

2. Related Works

Although many secure communication schemes to protect the privacy of smart grid users have been introduced over the years, not many privacy-preserving aggregation schemes such as [5–8] have been proposed so far. Electricity consumption data collection is an important process in smart grid communication systems. However, a report from the Netherlands argues that frequent reading of smart meters is problematic from a legal point of view [9], violates the European Convention on Human Rights, and generates many load issues. Fortunately, integrating edge computing into smart grids and designing data aggregation schemes that protect privacy can avoid these problems. First, Pacific Northwest National Laboratory first proposed "edge computing" in an internal report in 2013. With the rapid growth of the Internet of Things, edge computing has received a lot of attention. Shi et al. summarize typical examples of the smart home and collaborative edge and present some of the challenges and opportunities in the area of edge computing [10]. It moves some of the workloads used in the cloud to the edge nodes. The security of sensitive data stored on cloud servers through edge nodes will be of great concern to users. Therefore, consideration should be given to the resource requirements of edge devices, as well as the privacy of smart grid users.

To address these issues, we use data aggregation technology to solve the transmission conflict problem of a large number of data packets for smart grids in edge computing. To improve the security of the data aggregation model, traditional secure data aggregation schemes use hop-by-hop aggregation encryption [11]. However, frequent encryption and decryption operations may affect the aggregation efficiency and increase the corresponding additional energy consumption and the delay of the data aggregation process. An efficient privacy-preserving aggregation scheme (EPPA) for smart grid communication [7] was proposed by Lu et al. They used a super-incremental sequence to construct multidimensional data and encrypted the data with Paillier homomorphic encryption [12]; however, the scheme has security flaws. Shi et al. used an untrusted aggregator to differentially aggregate multiple time slots, which is more costly based on computationally intensive systems [13]. Fan et al. [14] proposed a secure power usage data aggregation scheme for smart grids, but it critically requires a third-party trust mechanism for distribution, adding an additional burden. Li et al. proposed a distributed incremental data aggregation approach where they used homomorphic encryption to solve the repetitive regular data aggregation task [5]. Garcia and Jacobs used homomorphic encryption to ensure the privacy of users and gave a measurement method [6]. Lu et al. proposed a lightweight privacy-preserving data aggregation scheme called Lightweight Privacy-Preserving Data Aggregation (LPDA), but it cannot achieve identity anonymization [15]. Hua et al. proposed an effective smart

grid aggregation scheme against malicious data mining attacks but increased the computational overhead and communication overhead [16].

3. Preparation

This section reviews the main basic concepts related to our work, including Paillier homomorphic encryption, simulated annealing region partition [17], chameleon hash function [18], and Attribute decision tree.

3.1. Paillier Homomorphic Encryption. Paillier cryptography is an additive homomorphic public key cryptography, which has been widely used in the field of encrypted signal processing or third-party data processing. Its homomorphic property is that the corresponding arithmetic operation can be performed on the ciphertext directly after encryption, and the result of the operation is the same as that of the corresponding operation in the plaintext domain. Its probabilistic property is that for the same plaintext, different ciphertexts can be obtained by different encryption processes, thus ensuring the semantic security of the ciphertext. The mechanisms used for encryption and decryption are as follows:

(1) Key generation: randomly select two large prime numbers p and q, calculate their product N and the least common multiple of $p-1$ and $q-1$, and then randomly select an integer that satisfies the following conditions:

$$\gcd\left(L\left(g^{\lambda} \bmod N^2\right), N\right) = 1. \tag{1}$$

Among them, function $L(u) = (u-1)/N$ and function $\gcd(.)$ are used to calculate the greatest common divisor of two numbers. Z_{N^2} is the set of integers less than $x \in Z_p^*$, while $Z_{N^2}^*$ is the set of integers coprime with N^2 in $Z_{N^3}^*$. (N, g) and λ are public key and private key, respectively.

(2) Encryption process: a random integer $r \in Z_i$ is selected. For any plaintext $m \in Z_w$, the corresponding ciphertext c is obtained by using public key (N, g) encryption:

$$\begin{aligned} c &= E[m, r] \\ &= g^m r^N \bmod N^2. \end{aligned} \tag{2}$$

According to the properties of the Paillier encryption system, when ciphertext $c \in Z_{N^2}^*$ is encrypted with the same public key, because the selection of ciphertext r is random, different ciphertext c can be obtained for the same plaintext m, but the same plaintext m can be restored after decryption, thus ensuring the semantic security of ciphertext.

(3) Decryption process: decrypt ciphertext c with private key n to get the corresponding plaintext m.

$$\begin{aligned} m &= D[c] \\ &= \frac{L\left(c^{\lambda} \bmod N^2\right)}{L\left(g^{\lambda} \bmod N^2\right)} \bmod N. \end{aligned} \tag{3}$$

3.2. Simulated Annealing Region Division

3.2.1. Regional Division. For a given smart meter, the division of area Q is expressed as follows:

$$Q \equiv \sum_{s=1}^{s_Q} \left[\frac{l_s}{L} - \left(\frac{d_s}{2L}\right)^2\right]. \tag{4}$$

s_Q is the number of regions, L is the number of links between smart meter nodes in the smart grid, l_s is the number of regions in region Q, L is the number of links between smart meter nodes in the smart grid, l_s is the number of links between smart meter nodes in region Q, and d_s is the sum of degrees of smart meter nodes in region Q. First, we use equation (4) to randomly place smart meters on the device layer into the area. Finally, we use a simulated annealing algorithm to find the optimal partition.

3.2.2. Simulated Annealing Algorithm. It is a general probabilistic algorithm that is used in our scheme to find the optimal solution to the zoning problem, where one can find low-cost smart meter regions, but not local minima for high-cost smart meter regions. We introduce the energy consumption T_e of smart meters to achieve this. Starting from high T_e, it gradually decreases and the system gradually approaches the minimum cost, avoiding the high-cost local minima.

The purpose of identifying modules is to maximize the use of modules, where costs $C = -Q$ and Q are the areas defined in equation (4). We update each energy consumption randomly, and the probability is expressed as

$$p = \begin{cases} 1 C\left(S'\right) \le C\left(T_e\right), \\ \exp\left(-\dfrac{C\left(S'\right) - C\left(T_e\right)}{T}\right) C\left(S'\right) > C\left(T_e\right), \end{cases} \tag{5}$$

where $C(S')$ is the cost after the update and $C(T_e)$ is the cost before the update, $\Delta C = C(S') - C(T_e)$.

3.3. Chameleon Hash Function. Traditional cryptographic hash functions are difficult to find collisions. But the chameleon hash function can artificially set up a "back door": if you master it, you can easily find collisions. This breaks the collision resistance of the hash function, but for most people, these properties remain, and the hash is still secure. Accenture applied the characteristics of the chameleon hash function and applied for a patent on an editable blockchain.

Although the decentralization and irrevocability of the blockchain are damaged to a certain extent, on the other hand, it also expands the application scenario of the blockchain and meets part of the needs of the government's regulatory requirements [19].

Principle description: suppose there exist two prime numbers p, q , and $q = kp + 1$ is large enough. The private key of the chameleon hash function is $x \in Z_p^*$, Z_p^* is the group of order q, and g is its generating element. The public key is $h = g^x \bmod p$. Given an arbitrary message m with random value $r \in Z_p^*$, now tampering the content m to m', it is now desired to find a random number r' such that $H(m\prime) = H(m)$. By the exponential property $g^a * g^b = g^{(a+b)}$, $(g^a)^b = g^{(ab)}$. The solution procedure for $r\prime$ is as follows:

$$H(m) = g^m h^r \bmod p = g^m g^{xr} \bmod p = g^{(m+xr)} \bmod p,$$
$$H(m\prime) = g^{m\prime} h^{r\prime} \bmod p = g^{m\prime} g^{xr\prime} \bmod p = g^{(m\prime+xr\prime)} \bmod p. \tag{6}$$

Therefore, m, $m\prime$, x, and r are known, $r\prime = (m + xr - m\prime)/x \bmod p$.

3.4. Attribute Decision Tree. The attribute decision tree is modeled after the access control tree and is set up according to the needs of the data collector. The leaf nodes of the attribute decision tree represent various attributes, and the intermediate nodes and roots are replaced by AND and OR. When an attribute of the data satisfies the requirements of the attribute decision tree, it is passed and the next calculation is performed; if not, other calculations or steps are performed.

For example, Mr. Li is a professor in the school of computer science of a university, so his attribute set matches the attribute strategy, as shown in Figure 2. Miss Wang is a professor in the school of information security of a university. Her attribute set does not match the attribute policy, as shown in Figure 3.

4. Edge-Assisted Lightweight Power Data Aggregation Encryption Scheme

4.1. Edge Privacy Aggregation Model. The edge privacy aggregation model contains four subjects: the User's Smart Meter (USM), the Marginal Power Services Institutions (MPSI), the Cloud Power Distribution Center, and the trusted organization. First, the USM encrypts data and divides it into optimal regions according to the change of energy consumption at different moments using a simulated annealing region partitioning algorithm, and as the energy consumption of USM changes at different moments, the number and location of clustered meters also change, thus realizing distributed data storage, which is conducive to the privacy protection of user data. Secondly, MPSI aggregates data with user identity anonymized and without affecting the privacy of any party. Finally, CPDC performs secure decryption, and TO performs key generation and distributes the key to the system. The model is shown in Figure 4.

User's Smart Meter. Smart meters use TPM chips to securely store and encrypt data. The SARD algorithm is executed using the handheld unit (including the sensor). Divide the smart meters of all users to meet the power load balance of the meters. The cluster meter regularly sends the collected data to the edge server. Perform data encryption calculation and chameleon signature calculation.

Marginal Power Services Institutions(MPSI). It consists of edge servers. The edge server performs chameleon signature aggregation and verification calculations and data aggregation calculations.

Cloud Power Distribution Center. The cloud server receives the aggregated data and decrypts it.

Trusted Organizations. The real identities of all users are virtualized to form virtual names and distribute system parameters and all private keys, and the distribution channels are all secure channels. The three parties of cloud, edge, and smart meter collaborate with trusted organizations to generate all private keys, as shown in Figure 5. Compared with existing solutions, our private keys require only a one-time setup between the three parties, which is beneficial for resource-limited systems. In addition, the private keys owned by TO are involved in decrypting the ciphertext and verifying the ciphertext, confusing the attacker, and making it impossible to tamper with the ciphertext.

4.2. Scheme Construction. The scheme proposed in this paper realizes the security and integrity of real-time power consumption data transmission between the smart meter and power server. The steps are as follows.

4.2.1. Initialization. TO inputs safety parameter (1^λ) and gets related parameter $(q_1, G_1, G_2, G_r, g_1, g_2, \omega, e)$, where q_1 is a large prime, G_1 and G_2 are two additive cyclic groups, G_r is a multiplicative cyclic group, q_1 is the order of the cyclic group, g_1 and g_2 are the generators of groups G_1 and G_2, respectively, satisfying that $\omega(g_2) = g_1$ and ω is an isomorphic mapping, e: $g_1 \times g_2 \longrightarrow g_r$ is bilinear mapping, and the storage list is established. TO chooses a system master key $s \in Z_p^*$, Z_p^* is a multiplication cycle group, and $y = g_2^s$ is the system public key. Two hash functions $H_1(.)$: $\{0,1\}^* \longrightarrow G_1$ and $H_2(.)$: $\{0,1\}^* \longrightarrow G_2$.

TO publishes system parameters and functions, selects a security parameter for the Paillier encryption algorithm, and sends it to the smart meter for initialization of the Paillier encryption algorithm. TO generates other parameters of the Paillier encryption algorithm: select two large prime numbers p and q, where $|p| = |q| = k$. The smart table computes $n = pq$ and chooses $g \in Z_{n^2}^*$ as the generator to use (n, g) as the public key of the Paillier encryption algorithm. CPDC computes the private key of the Paillier encryption algorithm $\lambda = lcm(p - 1, q - 1)$.

For the initialization of the chameleon signature, TO selects an element g_3 of order q in Z_p^* and an arbitrary index

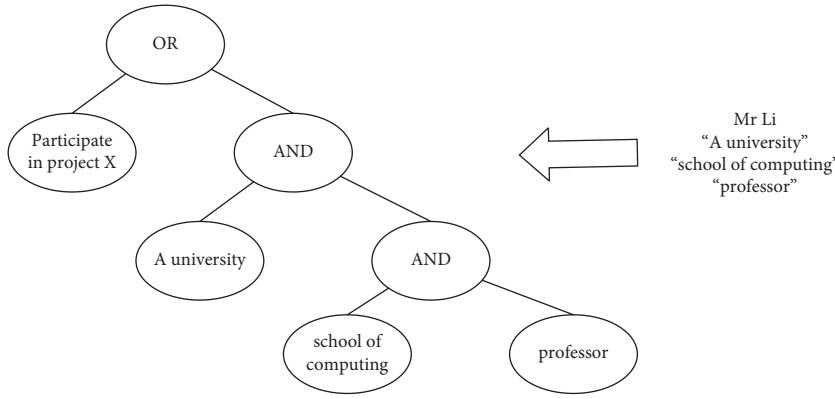

Figure 2: Schematic diagram of successful matching of policy and attribute collection.

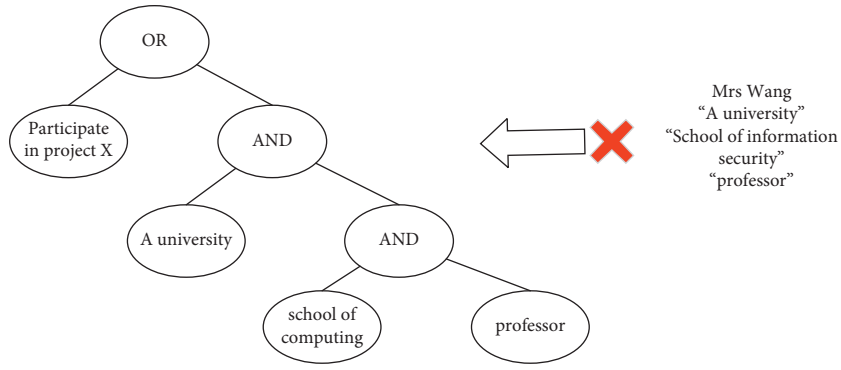

Figure 3: Schematic diagram of policy and attribute collection mismatch.

x , then the private key of the chameleon signature is $CK = x$, and the public key is $HK = g_3^x$.

TO sets the regularized attribute set F as a multiplicative cyclic group; then, any attribute f in the attribute set F is any element in the multiplicative cyclic group. The attribute set F is sent to the smart meter. Similarly, if TO sets the attribute set A of the attribute decision tree as a multiplicative cyclic group, then any attribute a in the attribute set A is any element in the multiplicative cyclic group, and the set attribute set A is sent to CPDC.

4.2.2. User Registration. Assuming a secure channel between TO and the user, in order to complete the user registration, the operation steps between the user and TO are as follows:

User i sends ID, serial number of smart meter to TO.

TO sends a Cert to user i after confirmation.

User i uses the Cert to get permission to request the parameters and key of the algorithm from TO.

TO sends the signature key etc. to the smart meter of user i.

TO calculation:

$$\begin{aligned} DP\ SI\ D &= H(I\ D, t)^{\text{Cert}}, \\ pid_{i,0} &= H(DP\ SI\ D, 0), \\ pid_{i,1} &= H(DP\ SI\ D, 1). \end{aligned} \tag{7}$$

TO calculates the signature key of user i:

$$\begin{aligned} S_{i,0} &= pid_{i,0}^{s}, \\ S_{i,1} &= pid_{i,1}^{s}. \end{aligned} \tag{8}$$

TO sends the signature key $S_i = (S_{i,0}S_{i,1})$, the real-time virtual name $DP\ SI\ D$ to the smart meter of user i.

4.2.3. Data Processing. Within data acquisition time t, the smart meter of user i encrypts the data with the Paillier homomorphic encryption and signs the encrypted data with the chameleon hash function which is referred to as chameleon signature for short. The cluster meter j collects data within the divided area. Finally, the real-time encrypted data and signatures are sent to MPSI. The steps are as follows.

The smart meter of user i selects a random number $a \in Z_{n^2}^*$ and encrypts data m_i.

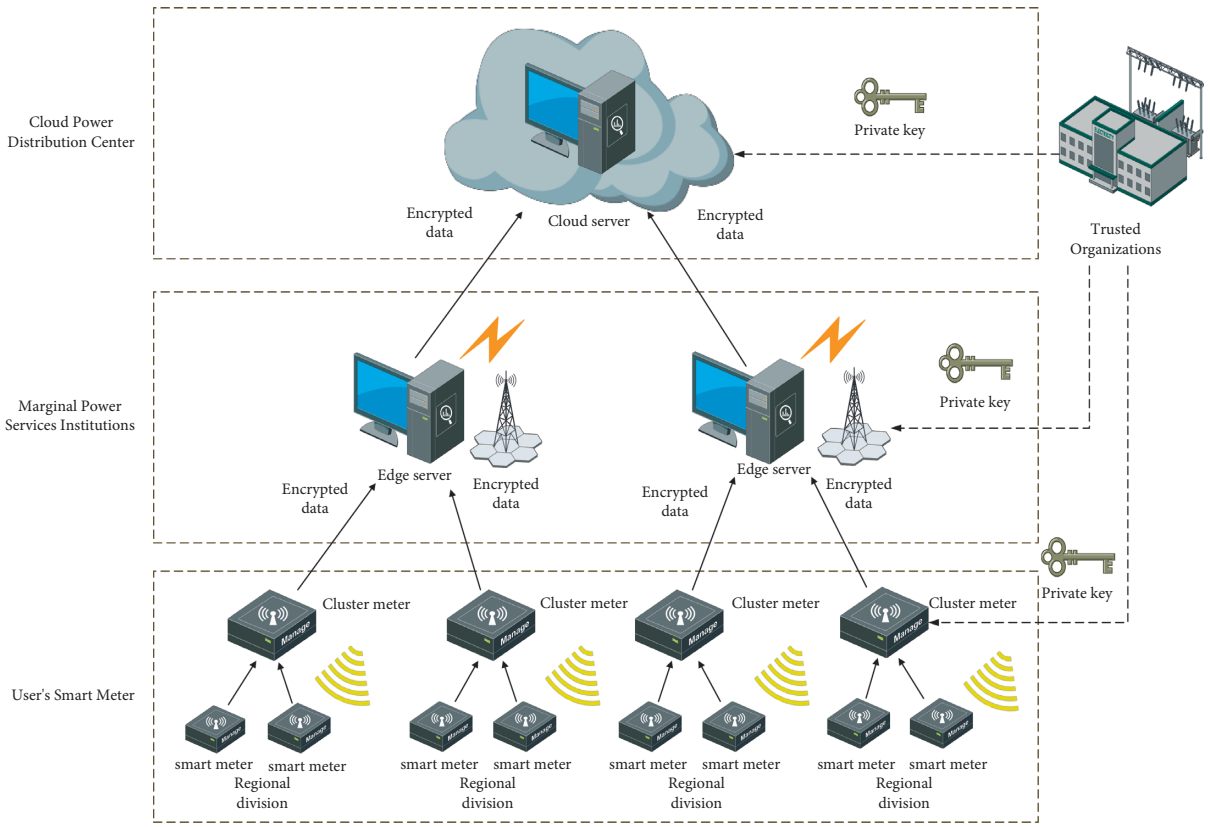

Figure 4: Edge privacy aggregation encryption model.

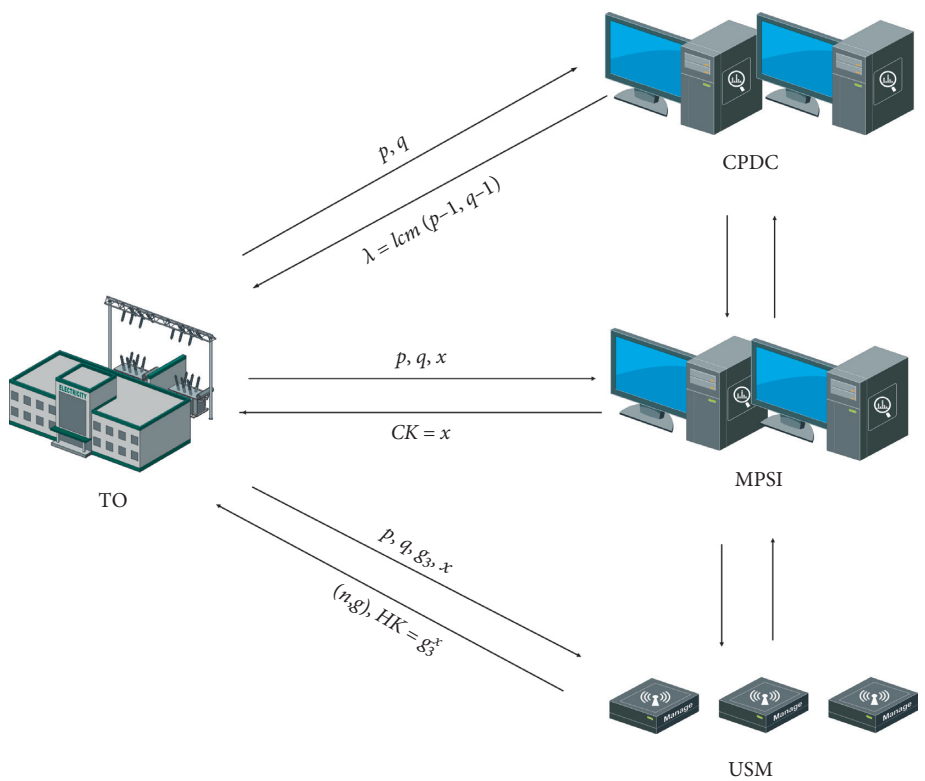

Figure 5: Key generation.

$$c_i = E(m_i) = g^{m_i} a^n \bmod n^2. \quad (9)$$

The smart meter of user i uses signature key $S_i = (S_{i,0}S_{i,1})$, virtual name, and attribute set to sign encrypted data by the chameleon hash function and finally send it to the cluster meter j.

$$\begin{aligned} h_i &= \text{Chamelelon}.H(c_i, HK, DP\ SI\ D, f), \\ \sigma_i &= s_{i,0} s_{i,1}^{h_i}. \end{aligned} \quad (10)$$

Cluster meter j sends $(c_i, \sigma_i, DP\ SI\ D)$ to MPSI.

MPSI receives the information and runs the virtual name-based verification algorithm as shown in Algorithm 1.

The algorithm first aggregates chameleon signatures. After verification, the attribute set f of the data is obtained, and the attribute set A of the data decision tree is matched in turn, and the data satisfying the data decision tree can be data aggregated with other data satisfying that decision tree for the data aggregation operation.

$$\begin{aligned} c &= \prod_{i=1}^{n} c_i \bmod n^2 \\ &= \prod_{i=1}^{n} g^{m_1} \ldots g^{mw} a^n \bmod n^2 \\ &= \prod_{i=1}^{n} g^{m_1 + m_2 + \ldots + m_w} a^n \bmod n^2 \end{aligned} \quad . \quad (11)$$

After aggregation, MPSI sends the aggregated data to the CPDC through the secure channel. Data decryption: CPDC decrypts the encrypted aggregate data.

$$m_i = \frac{L\left(c^{\lambda} \bmod n^2\right)}{L\left(g^{\lambda} \bmod n^2\right)} \bmod n. \quad (12)$$

$m_i = m_1 + m_2 + \ldots\ldots + m_w$, CPDC stores data for power grid operation and puts forward decisions.

4.2.4. Track. While making power consumption analysis and decision-making, CPDC may find that some power consumption values do not meet its predetermined range or abnormal conditions. At this time, CPDC will start the tracking process, and the steps are as follows:

CPDC sends the command to the edge server that submits the relevant abnormal power consumption: let MPSI send the stored power consumption and virtual name at that time to CPDC.

CPDC first decrypts each encrypted data received, detects and finds the abnormal power consumption, and locks its $DP\ SI\ D$.

CPDC sends the virtual name of the abnormal power consumption determined by it to TO and applies for identity tracking.

TO can query the real identity of the users who send out abnormal electricity consumption. TO sends the real identity to CPDC, and CPDC processes the user and his power consumption accordingly.

4.3. Safety Analysis

4.3.1. User Identity Privacy Protection. Before sending data to the CPDC, the USM registers with the TO to obtain a virtual name and signing key. The USM uses the virtual name as the identity of the data transfer in the architecture and performs encryption, signing, and other actions based on it. The USM has a tamper-proof storage device. This storage device can be thought of as a "black box" that can read and write data, but only by the USM; no other device can read or write information. According to the one-way and collision-free characteristics of the hash function, even if the attacker obtains the virtual name, it cannot crack the real identity. This scheme can effectively protect user identity and prevent illegal intrusion.

4.3.2. Security Analysis of Chameleon Signature. The chameleon signature is a preferable designated verifier signature. Compared to other signatures, the chameleon signatures are more suitable for lightweight aggregated encryption schemes due to their ability to transmit data efficiently and reduce computational overhead. Chameleon signatures are also nontransmissible, nonforgeable, and nonrepudiation, which also ensure data security and meet the security requirements of the system.

4.3.3. User Fine-Grained Data Privacy Protection. USM encrypts the electricity consumption data using the Paillier encryption algorithm, sends it to MPSI, which does not have the ciphertext decryption key, and sends the ciphertext to CPDC after successful verification. CPDC mainly receives aggregated numbers of electricity consumption data, so it protects the user's fine-grained data privacy, while CPDC can get the complete electricity consumption data.

5. Experimental Analysis

5.1. Simulated Annealing Region Division. Intraregional connectivity and participation: each region is divided into relatively balanced regions from one or several fully centralized regions based on the energy consumption of smart meters to achieve a balanced electrical load in each region. We define the intraregional connectivity, in order to measure whether the smart meter u is well connected to other smart meters in the region.

$$Z_u = \frac{k_u - \bar{k}_{s_u}}{\sigma_{k_{s_u}}}, \quad (13)$$

where k_u is the number of links from the smart meter u to other smart meters in zone s_u, $\bar{k}_{s_u}$ is the average number of links from all smart meters in the zone s_u, and $\sigma_{k_{s_u}}$ is the standard deviation of all links in the zone.

Of course, we also need to consider unexpected situations. For example, a smart meter u may not be connected to

its own area. Therefore, we define the participation degree p_u of a smart meter u.

$$p_u = 1 - \sum_{s=1}^{s_M} \left(\frac{k_{us}}{k_u}\right)^2, \tag{14}$$

where k_{us} is the number of links from the smart meter u to smart meters in zone s , and k_u is the total number of degrees of the smart meter u. According to equation (14), if the connections of the smart meter u are evenly distributed in all areas, then the participation degree of the smart meter u is close to 1. If all its connections are in its own area, the participation degree is 0.

We use a MATLAB environment with a Dell laptop (i5-6200u, CPU 2.40 GHz, Windows 10 OS) for simulation experiments. Assuming that 100 smart meters are randomly distributed in a $1.0 * 1.0$ km smart grid, and each smart meter has a random electricity consumption $N(T_e)$, a zoning model is established. First, the 100 randomly distributed smart meters are generated as a subset of the neighborhood of electricity consumption $N(T_e)$ Download the open-source dataset from the website Open Energy Data Initiative (OEDI) and randomly select the electricity consumption information from 100 apartments with no missing points and a time granularity of 15 minutes. The average value is calculated based on the electricity load of 100 users at different times of the day, as shown in Figure 6. 14:00–20:00, the user's electricity load continues to grow, with 20:00 reaching the highest peak of the day; 20:00–24:00, the user's electricity load continues to fall to a stable value. After reasonable analysis, we divide the average value of the electricity load of 100 users in different time periods of a day into 6 electricity consumption states. A power consumption state of $S(k)$ is randomly selected for the regional division scheme, and the next power consumption state of S' is randomly selected as the candidate scheme for the next regional division scheme. Calculate $\Delta C = C(S') - C(T_e)$; if $\Delta C < 0$, accept S' for the next region division scheme; otherwise, we judge the random update probability $p = \exp(-\Delta C/cT) > \alpha$, $\alpha \in (0, 1)$; if true, accept S' for the next region division scheme; namely, $S(k+1) = S'$, $k = k+1$. Then, we check whether the connectivity and participation in the region satisfy equations (13) and (14). Finally, we use $S(k+1)$ for the region partition scheme and return the SARD algorithm.

Figures 7(a)–7(f) show the experimental process of the SARD algorithm. We performed six rounds of state calculation, divided the six power consumption states into different regions, and terminated the algorithm. Cluster meters in each area are used to collect data and process the data accordingly to realize power load balancing under different power consumption states.

First, the power consumption of smart meters increases with the increase of users in the smart grid. Since all the data eventually needs to be sent to the cloud server of CPDC for processing, the power consumption of the cloud server also increases with the increase of data, as shown in Figure 8. Then, we introduce edge computing into the smart grid, and the power consumption of MPSI increases with the increase of edge servers. This layer processes a large amount of data and then sends it to the CPDC. Since the CPDC does not need to process a large amount of data, the power consumption of the cloud server in the CPDC does not fluctuate much, as shown in Figure 9. Comparing the experiments in the two figures, the introduction of edge servers to process large amounts of data in the edge privacy aggregation model of the smart grid effectively reduces the power consumption of the CPDC and the total system power consumption.

5.2. Total Computing Overhead. The computational overheads of this scheme and the LPDA scheme mainly involve the following three operations: bilinear pair operation, exponential operation, and Paillier homomorphic encryption and a decryption operation, and other operations are neglected. The bilinear pair operation and the exponential operation are C_b and C_e, respectively, and the encryption and decryption of the Paillier algorithm are C_A and C_B, respectively, and the other computational overheads are neglected. The AMDM scheme is mainly multiple operations, C_{pe} is the multiplication operation in the cyclic group $Z^*_{N^2}$, C_{pm} and C_m are the multiplication operation in $Z^*_{p'}$, C_e is the exponential operation, and C_{gm} is the multiplication operation in the group G_1 because C_{pm} and C_m produce little effect negligible.

The scheme uses the MATLAB environment of a Dell laptop (i5-6200u, CPU 2.40 GHz, Windows 10 OS) for simulation experiments. The simulation measures the amount of time needed by the Dell laptop to perform basic operations in the experimental environment. It takes 1.1 ms to calculate a single C_e, 3.1 ms to calculate C_b, 4.5 ms to calculate C_{pe}, and 2.1 ms to calculate C_{gm}. Since all three scenarios in this paper have only one pair of encryption and decryption operations, we first disregard C_A and C_B .

The scenarios in this paper consider the computational overhead of each of the three participants, Smart Meter, MPSI, and CPDC, and compare them with other scenarios, as shown in Table 1. The total computational overhead of all the solutions is the total computational overhead of the three participants. As can be seen from the table, this paper is significantly more efficient than the other two schemes.

As shown in Figure 10, the computing energy consumption of the scheme in this paper is significantly lower than the aggregated encryption schemes of the remaining two schemes, where the AMDM scheme resists malicious attacks and requires more computing energy and is significantly higher than the LPDA scheme and the scheme in this chapter, while the scheme in this chapter does not cause additional computation while ensuring data security due to the use of the chameleon signature, so the total computation overhead is lower, and it can be said that the scheme in this chapter is better than the LPDA scheme and the AMDM scheme.

5.3. Total Communication Overhead. The total communication overhead of this scheme mainly refers to all the communication data that needs to be transmitted in the system. The output data length of the hash function is

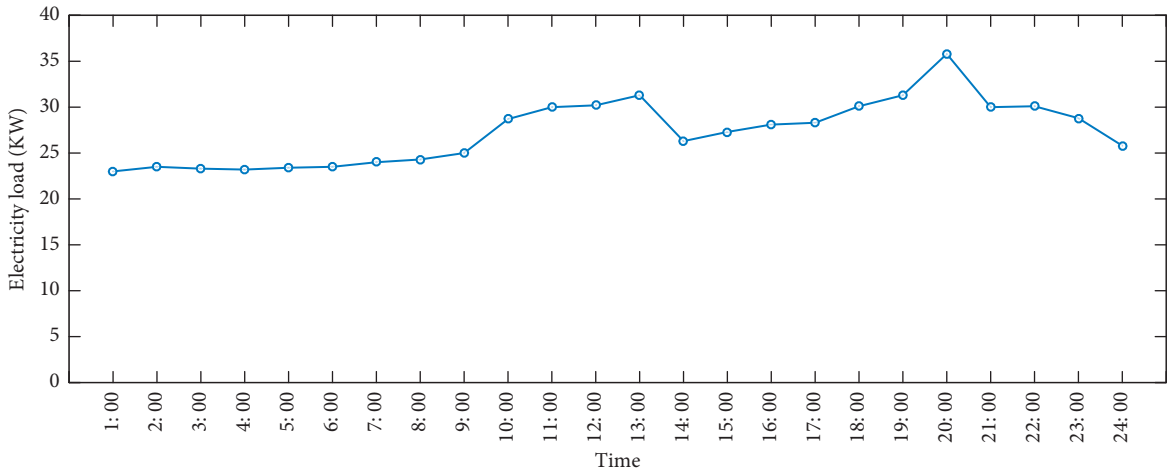

Figure 6: Edge privacy aggregation encryption model.

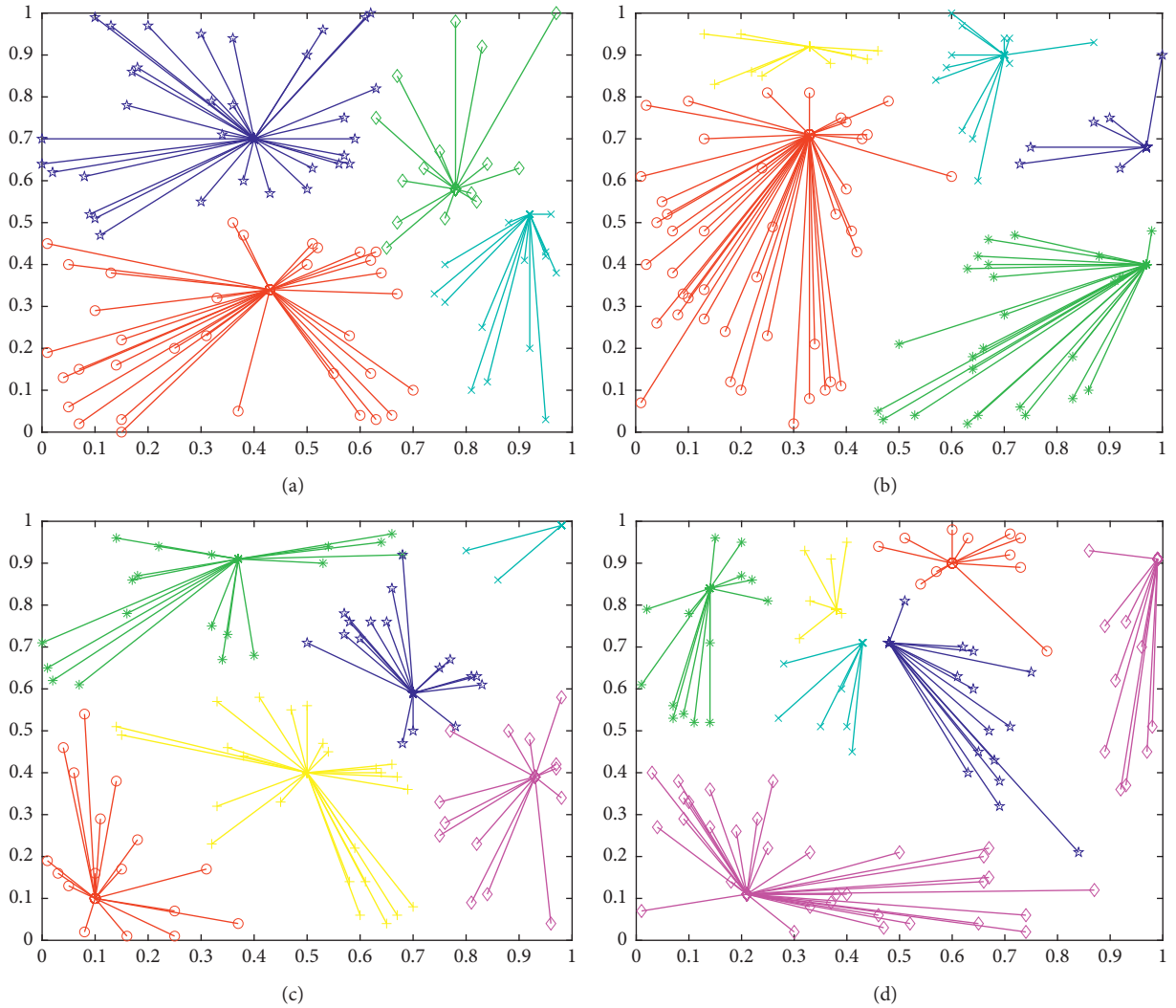

Figure 7: Continued.

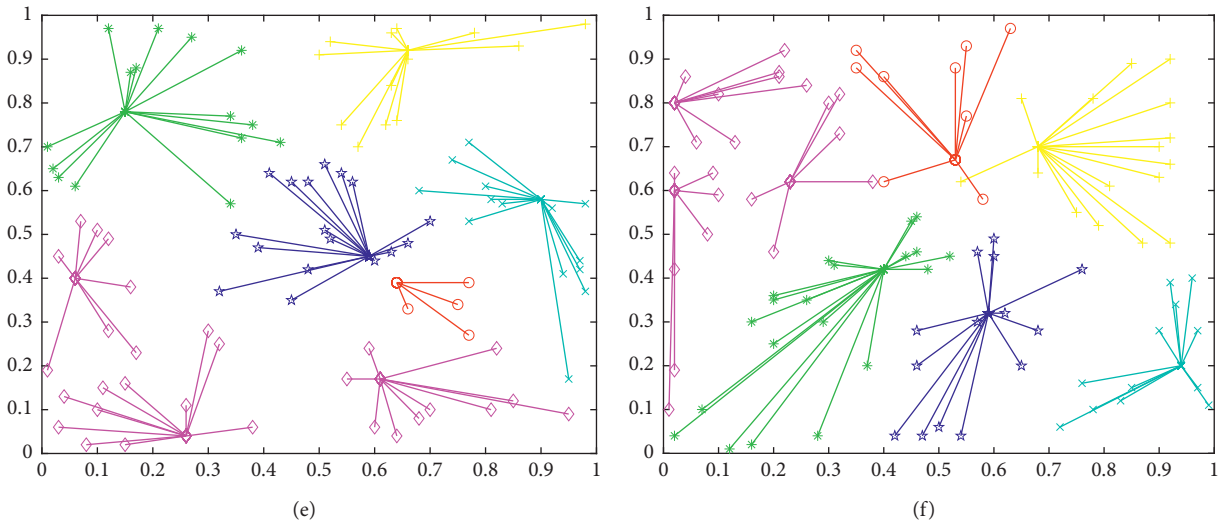

Figure 7: Regional division based on electricity consumption state. (a) Partition of 0:00–7:00. (b) Partition of 13:00-14:00. (c) Partition of 7:00–11:00. (d) Partition of 11:00–13:00. (e) Partition of 14:00–20:00. (f) Partition of 20:00–24:00.

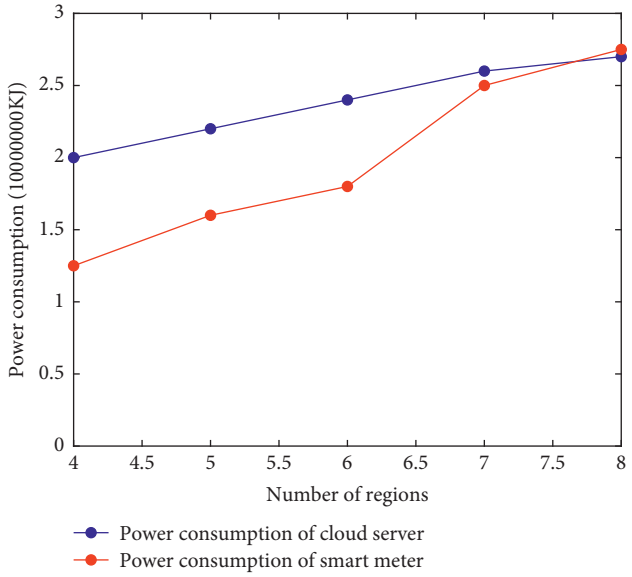

Figure 8: Energy consumption of cloud network.

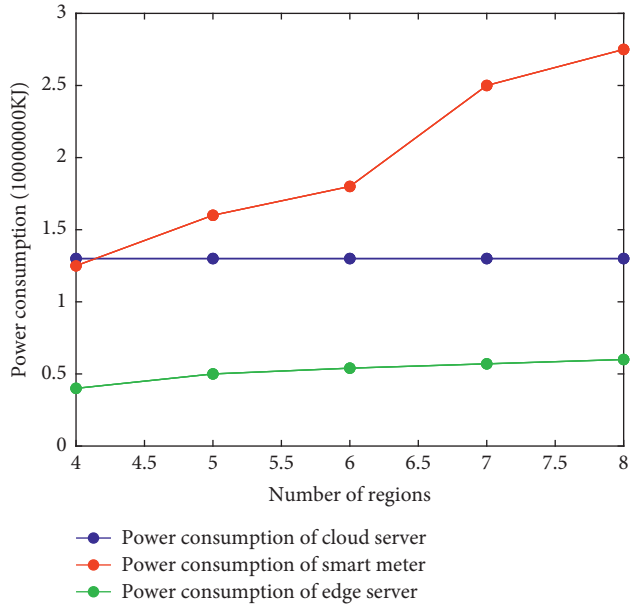

Figure 9: Energy consumption of edge computing network.

160 bits. Suppose the length of n in Paillier encryption algorithm is 512 bits, the length of group G_1 element is 161 bits, the length of $DP\ SI\ D$ is 32 bits, the length of attribute set f is 32 bits, and the length of σ_i is 32 bits. The total communication data volume of this scheme consists of two parts: the first part is from SM to MPSI, and the data transmitted is $(c_i, \sigma_i, DP\ SI\ D)$; the second part is from MPSI to CPDC, and the data transmitted is c . The total traffic of the LPDA scheme consists of two parts. The first part is from SM to ESP, which transmits 2048 bits through calculation, and the second part is from ESP to CC, which transmits 2048 bits through calculation. The total traffic of the AMDM scheme consists of two parts. The first part is SM to GW, which transmits 3264 bits through calculation, and the second part is GW to CC, which transmits 3264 bits through calculation. The comparison between this scheme and other schemes is shown in Table 2.The simulation experiment is carried out using MATLAB, and the results are shown in Figure 11.

We use the smart meter data of a year in London on the Open Energy Data Initiative (OEDI) website to simulate the total communication cost per day. As shown in Figure 12, different colors represent different communication situations; that is, when the number of edge servers and smart meters changes, the communication cost also changes. Based on the actual privacy requirements and cost requirements of the customer, we implement appropriate electricity usage data delivery mechanisms in the actual area.

Input: c_i, σ_i, *DP SI D*
Output: c
(1) **for** $i = 1; i < n; i++$ **do**
(2) $\Omega = \prod_{i=1}^{n} \sigma_i$;
(3) $h_i = \text{Chamelelon}.H(c_i', CK, DP\ SI\ D, f\prime), c_i', f\prime \in Z_p^*$;
(4) $f = f' - c_i - c_i'/x \bmod p$;
(5) f match A;
(6) $pid_{i,0} = H(DP\ SI\ D, 0), pid_{i,1} = H(DP\ SI\ D, 1)$;
(7) $e(\Omega, g) = e(\prod_{i=1}^{n} pid_{i,0} pid_{i,1}^{h_i}, y)$;
(8) $c = \prod_{i=1}^{n} c_i \bmod n^2$;
(9) **end for**
(10) MPSI sends c to CPDC;

Algorithm 1: Verification algorithm based on the virtual name.

Table 1: Analysis of computational complexity.

Scheme	SM	MPSI (ESP, DCP)	CPDC (CC)
Our scheme	$3C_e + C_A$	$NC_e + 2C_b$	C_B
LPDA	$C_e + C_A$	NC_e	$NC_e + C_B$
AMDM	$2C_{pe} + 2C_e + C_{qm} + C_A$	$(N+2)C_b + C_{qm}$	$2C_b + C_{pe} + 2C_e + C_B$

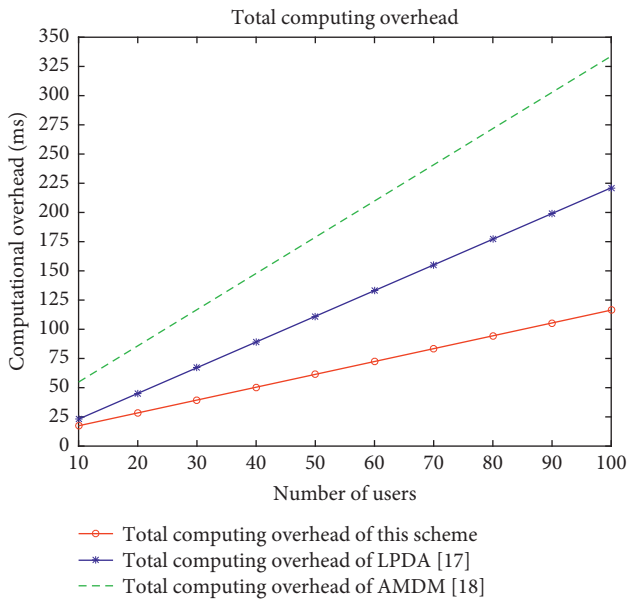

Figure 10: Total computing overhead.

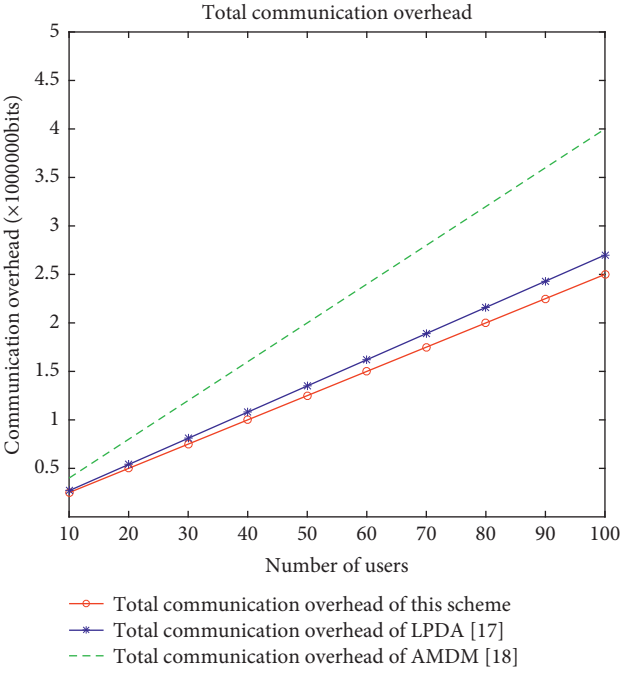

Figure 11: Total communication overhead.

Table 2: Analysis of communication complexity.

Scheme	SM (bit)	MPSI (ESP, DCP) (bit)
Our scheme	1409	1024
LPDA	2048	2048
AMDM	3264	3264

6. Conclusion

In this paper, we consider the actual smart grid, introduce edge computing, and propose an edge-assisted lightweight electricity consumption data aggregation and encryption

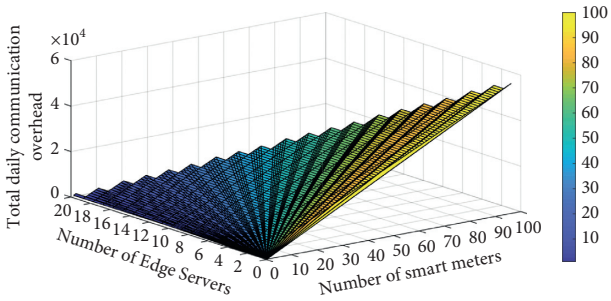

Figure 12: Total daily communication overhead.

scheme, which solves the problem of sending electricity consumption data to the cloud by users securely and efficiently. The scheme uses a simulated annealing zone partitioning algorithm to reasonably partition smart meters according to their electricity consumption energy consumption to achieve load balancing of smart grid systems; at each sending of data, licensed users apply for virtual names from trusted organizations to enable them to communicate with the grid as anonymous, which effectively protects the privacy of user identity security; in encrypting data, CPDC uses a virtual name-based verification algorithm which is used to Paillier encryption technology combined with chameleon signature to ensure authentication, integrity, and nonrepudiation of data, so that CPDC can only obtain encrypted data aggregated by MPSI, protecting the privacy of users' fine-grained data. Performance analysis shows that it is much better than the LPDA scheme and AMDM scheme in terms of communication overhead and computation overhead. In future work, we will evaluate our schemes in realistic smart grid scenarios with stronger adversarial models and study the impact of different signatures on system performance and security.

Data Availability

The research data are obtained from the corresponding author.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant nos. 61771289 and 61832012), Major Basic Research of Natural Science Foundation of Shandong Province (Grant no. ZR2019ZD10), and Key Research and Development Program of Shandong Province (Grant no. 2019GGX101050).

References

[1] A. Saleem, A. Khan, S. U. R. Malik et al., "FESDA: fog-enabled secure data aggregation in smart grid IoT network," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 6132–6142, 2020.

[2] K. Wei, S. Jian, and Pandi, "A practical group blind signature scheme for privacy protection in smart grid," *Journal of Parallel and Distributed Computing*, vol. 136, pp. 29–39, 2020.

[3] J. Xiong, R. Bi, M. Zhao, J. Guo, and Q. Yang, "Edge-assisted privacy-preserving raw data sharing framework for connected autonomous vehicles," *IEEE Wireless Communications*, vol. 27, no. 3, pp. 24–30, 2020.

[4] J. Xiong, J. Ren, L. Chen et al., "Enhancing privacy and availability for data clustering in intelligent electrical service of IoT," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 1530–1540, 2019.

[5] F. Li, B. Luo, and P. Liu, "Secure information aggregation for smart grids using homomorphic encryption," in *Proceedings of the First IEEE International Conference on Smart Grid Communications*, pp. 327–332, IEEE Press, Gaithersburg, MD, USA, October 2010.

[6] F. D. Garcia and B. Jacobs, "Privacy-friendly energy-metering via homomorphic encryption," in *Proceedings of the 6th International Conference on Security and Trust Management*, vol. 67, no. 10, pp. 226–238, Springer-Verlag, Berlin, Germany, 2011.

[7] R. Rongxing Lu, X. Xiaohui Liang, X. Xu Li, X. Xiaodong Lin, and X. Xuemin Shen, "EPPA: an efficient and privacy-preserving aggregation scheme for secure smart grid communications," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 9, pp. 1621–1631, 2012.

[8] R. Petrlic, "A privacy-preserving concept for smart grids," *Sicherheit in Vernetzten Systemen*, pp. B1–B14, 2010.

[9] Q. Zhou, G. Yang, and L. He, "An efficient secure data aggregation based on homomorphic primitives in wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 10, no. 1, pp. 2022–2037, 2014.

[10] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: vision and challenges," *IEEE Internet of Things Journal*, vol. 3, no. 5, pp. 637–646, 2016.

[11] Y. Yang, X. Wang, S. Zhu, and G. Cao, "Sdap," *ACM Transactions on Information and System Security*, vol. 11, no. 4, pp. 1–43, 2008.

[12] P. Paillier, "A public-key cryptosystem based on composite degree residuosity classes," *Advances in Cryptology - EUROCRYPT'99*, vol. 1592, pp. 223–238, Springer-Verlag, Berlin, 1999.

[13] E. Shi, T. H. Chan, E. G. Rieffel, R. Chow, and D. Song, "Privacy preserving aggregation of time-series data," *Network and Distributed System Security (NDSS)*, vol. 2, no. 4, pp. 1–17, 2011.

[14] C.-I. Fan, S.-Y. Huang, and Y.-L. Lai, "Privacy-enhanced data aggregation scheme Against internal attackers in smart grid," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 1, pp. 666–675, 2014.

[15] R. Lu, K. Heung, A. H. Lashkari, and A. A. Ghorbani, "A lightweight privacy-preserving data aggregation scheme for fog computing-enhanced IoT," *IEEE Access*, vol. 5, pp. 3302–3312, 2017.

[16] Y. Dong, J. hen, S. Ji, Q. Rongxin, and L. Shuai, "A novel appliance-based secure data aggregation scheme for bill generation and demand management in smart grids," *Connection Science*, vol. 33, no. 4, pp. 1–22, 2021.

[17] L. Ren and L. Lin, "Simulated annealing algorithm coupled with a deterministic method for parameter extraction of energetic hysteresis model," *IEEE Transactions on Magnetics*, vol. 54, no. 11, pp. 1–5, 2018.

[18] S. A. Hua, A. Yl, X. D. Zhe, and Z. Mingwu, "An efficient aggregation scheme resisting on malicious data mining attacks for smart grid," *Information Sciences*, vol. 526, pp. 289–300, 2020.

[19] Y. Tian, T. Li, J. Xiong, M. Z. A. Bhuiyan, J. Ma, and C. Peng, "A blockchain-based machine learning framework for edge services in IIoT," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 3, pp. 1918–1929, 2022.