Hindawi
Wireless Communications and Mobile Computing
Volume 2020, Article ID 8811175, 12 pages
https://doi.org/10.1155/2020/8811175

Research Article

Study on Delay Optimization of Fog Computing Edge Nodes Based on the CPSO-LB Algorithm

J. Y. Wu,[1] **R. Xin**,[1] **J. B. Zhao**,[2] **T. Zheng**,[1] **D. Jiang**,[1] **and P. F. Zhang**[1]

[1]*State Grid Hebei Information and Telecommunication Branch, Shijiazhuang 050021, China*
[2]*State Grid Hebei Electric Power Co. Ltd., Shijiazhuang 050021, China*

Correspondence should be addressed to J. Y. Wu; wujy@he.sgcc.com.cn

Received 24 April 2020; Revised 11 June 2020; Accepted 7 July 2020; Published 2 December 2020

Academic Editor: Fuhong Lin

With the development of modern science and technology as well as the steady advancement of urbanization, intelligent networks have emerged and are replacing traditional networks with the identity of next-generation networks. And information security is one of the most important research directions in the intelligent network construction. In order to resist the threat of privacy leakage during the data transmission of intelligent terminal, an original four-layer fog computing system which is suitable for intelligent network data collection, transmission, and processing structure is established in the paper. With the help of the Paillier algorithm for encryption and fine-grained aggregation, the fine-grained aggregated data as coefficients are embed in the cloud node, and Horner's rule is conformed for unary polynomials, which further aggregates to reduce the amount of transmitted data, so that communication overhead is reduced as well. Meanwhile, the resolvability of Horner's rules allows EPSI to finally obtain the subregional information plain text, and it is summed up to obtain cloud-level information data. Therefore, the comparative analysis of simulation experiments with other algorithms proves that the rational optimization of the research content in this paper plays a higher security role.

1. Introduction

The intelligent network is designed to combine traditional network and information network technologies to encourage family users to actively manage daily energy consumption and efficiently provide reference information needed by power supply companies for planning and regulation. Although the deployment of intelligent networks can bring huge socioeconomic benefits, severe information security risks also follow. During the transmission of information data, illegal attackers can master user's life habits by maliciously eavesdropping on the data, which can also cause huge property losses to users or suppliers by maliciously tampering with the data [1, 2].

The network privacy protection research is usually dedicated to solve two kinds of security risks: intelligent terminal identity security risks and intelligent terminal data security risks. Identity security needs to consider the problem of identity distribution of each entity in the intelligent network and the problem of mutual authentication among different domains. Data security needs to ensure the confidentiality and integrity of the data to avoid data loss or leakage.

The work in the paper makes full use of decentralized computing and storage resources to achieve a better user experience, and its specific contributions are as follows [3–5]:

(1) Lightweight Key Agreement Identity Authentication to Achieve Privacy Data Integrity. A key agreement scheme based on elliptic curve is applied to the identity authentication between layers, which can avoid bilinear pairing, and effectively reduces the calculation overhead. In addition, a certificateless mode is adopted in the paper, which effectively avoids the case of dishonest key generation center eavesdropping and forging user signatures.

(2) Achieving Fine-Grained Aggregation of Data Privacy and Confidentiality. The Paillier encryption algorithm is used to process private data, and its additive homomorphism is applied to sum the intelligent terminal data in an encrypted state, so that data aggregation on the premise of protecting

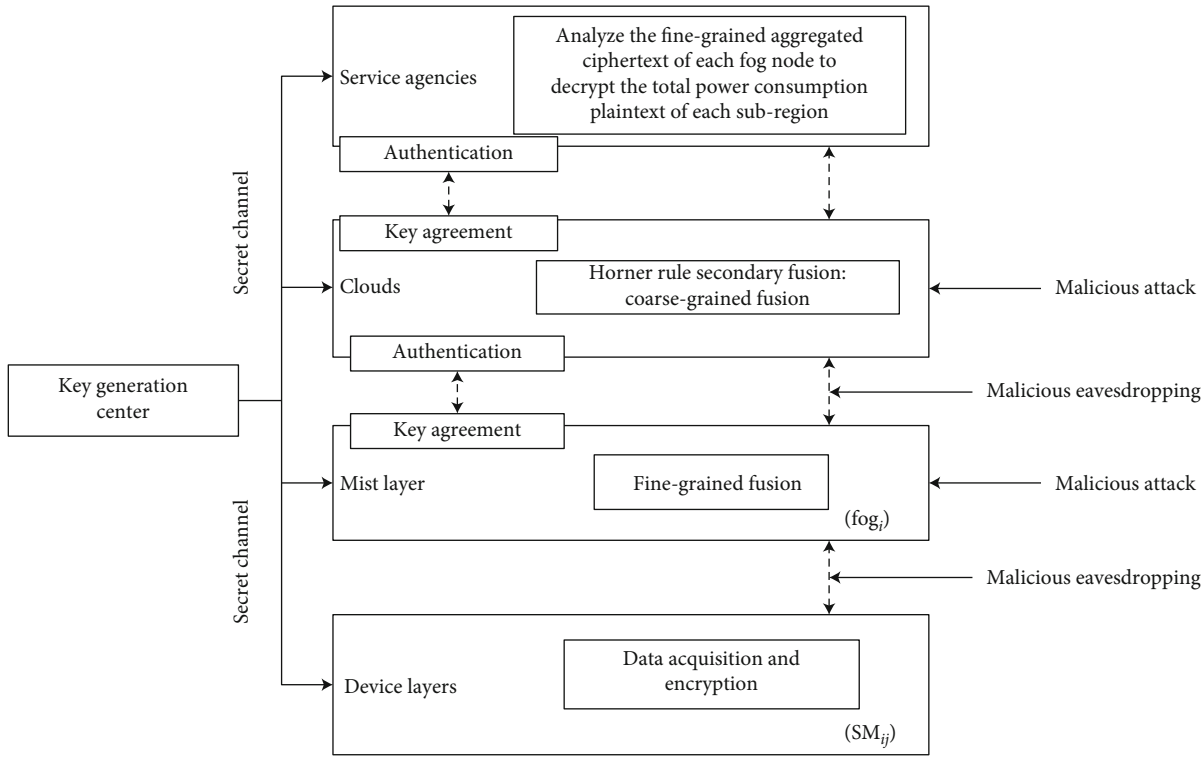

Figure 1: Four-layer fog computing network architecture.

the privacy of personal information can be completed. Therefore, the data aggregation is completed under the premise of protecting the privacy of personal information, which effectively resists the eavesdropping attack brought by the curiosity of fog node.

(3) Realizing Multigranular Security Aggregation of Cloud-Fog Collaboration. The Horner rules are used for further coarse-grained aggregation of fog node data, and the least multiplication operation strategy is applied to speed up the operation. Meanwhile, the final data results can be accurate to the range of a single fog node area and also ensure that the fog cloud cannot obtain personal information data, which is enough to provide differentiated user data services.

2. Materials and Methods

A fog computing network architecture suitable for intelligent network data collection, processing, and transmission is built in the paper, as shown in Figure 1. It consists of 4 layers: equipment layer, fog layer, cloud layer, and EPSI. In the constructed system model, cloud node coverage is divided into f subregions, and each region is assigned 1 jog node fog_j, which corresponds to fog_1, fog_2, …, fog_f, and there are the numbers of intelligent terminal SM_{ij} (it indicates the i^{th} intelligent terminal device under the j^{th} fog node, $i \in [1, n]$) of n in the coverage of each fog node [6, 7].

As can be seen from Figure 1, the system model mainly includes the following five entities: KGC (key generation center), intelligent terminal, fog node, cloud node, and EPSI (electric power service institutions).

(1) KGC. It is a third party that is not completely trusted and is mainly responsible for generating various keys and sending them to various entities, which has relatively strong computing power [8, 9].

(2) Intelligent Terminal. The user information data are collected in real time, and it is encrypted. After negotiating with the corresponding fog node key, the encrypted data are signed and uploaded to the corresponding fog node periodically to wait for aggregation [10, 11, 15]. In addition, the user can send a request to the corresponding fog node to view the real-time data of the total amount of information in its coverage area so as to understand the regional information.

(3) Fog Node. It locates in the middle layer of intelligent terminal and cloud node, which dedicates to fully tap local computing power. There is a fog node in each subarea, and the fog node interacts with the intelligent terminal within its coverage area, which can effectively resist malicious injection attacks through identity authentication technology, perform fine-grained aggregation on the authenticated data, and forward the aggregated data to the corresponding cloud node.

(4) Cloud Node. The cloud receives the aggregated cipher text from each fog node in its coverage area, avoids malicious injection through identity authentication, and uses Horner rules for the second aggregation to obtain a coarse-grained aggregation

result [12–14]. In addition, the cloud node sends coarse-grained aggregated data to EPSI through a secure channel.

(5) EPSI. EPSI receives the coarse-grained aggregated data from the cloud node, and it first parses at a high speed to obtain the fine-grained aggregated cipher text of each fog node. Then, decryption is performed to obtain the plain text of the information amount of each subregion, and plain texts are summed. Thus, the total amount of information covered by the cloud is obtained, and multigranular real-time data basis is provided for scheduling. Meanwhile, EPSI can also aggregate real-time data of subregions and send them back to each fog node through the "cloud-fog" communication link so that the user can query the amount of information. This operation not only saves the computing resources of the cloud but also enables the user to query the information in real time with low latency.

The communication link between KGC and intelligent terminal, fog node, cloud node, and EPSI and the communication link between the cloud node and EPSI are credible, while the communication links between other layers are not safe. The paper mainly considers the following three threats [15–18]:

(1) There are threats on the fog and cloud nodes. The fog and cloud nodes are generally considered to be honest and trustworthy, which will follow the protocol and are trustworthy in most cases. However, it cannot be ignored that the fog node and the cloud node also have the possibility of being captured. Therefore, the system must ensure that the fog node and the cloud node cannot obtain the private user's private data in plain text; that is, the data cannot appear in plain text in the fog node and the cloud node. Meanwhile, the fog node and the cloud node cannot have the decryption key, which can ensure the security of the system

(2) The eavesdropper threatens to eavesdrop on the communication link. The eavesdroppers may obtain user privacy data by eavesdropping on the communication link. Therefore, the system must ensure that the privacy data of a single user does not appear in each communication link; that is, the data exists in the form of cipher text during the transmission of each communication link. At the same time, the key generation center only sends the decryption key to the intelligent terminal and EPSI through the trusted secret channel, and the communication links from intelligent terminal to the fog node, the fog node to the cloud node and the cloud node to EPSI will not transmit the decryption key, and eavesdroppers will not be able to eavesdrop on the decryption key. It ensures that even if eavesdroppers eavesdrop on the information from the communication link, they will not be able to crack user's private data

(3) The threat of an attacker actively attacking. In addition to launching passive attacks through eavesdropping, attackers can also maliciously inject through camouflage and other methods, thereby destroying the authenticity and integrity of private data. Therefore, before receiving the data and performing the protocol operation, the fog node or the cloud node must authenticate the identity through key agreement to ensure that the data comes from the legal entity, and the data is sent to the legal entity

3. Results and Discussion

EPSI's scheduling analysis depends on the real-time information volume of each area, so the data will be read to the intelligent terminal of each area at a fixed time interval. However, there is a certain risk of privacy leakage in the process of reading and transmitting data, and there is a problem that the communication overhead of traditional data transmission is relatively large. Therefore, homomorphic encryption is used in the paper to ensure the privacy and confidentiality of data during transmission. What is more, the multigranular aggregation of the fog layer and the cloud layer can effectively reduce the amount of data transmitted, thereby reducing transmission consumption [19]. In particular, the data results of multigranularity aggregation can also improve the flexibility of scheduling. With the help of a lightweight identity authentication scheme with low computing overhead, it can save fog and cloud computing resources while resisting camouflage attacks.

The privacy protection data aggregation scheme proposed in the paper consists of the key generation and distribution, the intelligent terminal data report, the fog node fine-grained report, the cloud node coarse-grained aggregation report, and EPSI aggregation report reading 5 parts.

3.1. Key Generation and Distribution

3.1.1. Paillier Key Generation and Distribution. KGC first randomly selects two large prime numbers p and q to satisfy $\gcd[pq, (p-1)(q-1)] = 1$ and calculates $N = pq$ as the public key for homomorphic encryption. Assuming $L(u) = (u-1)/n$, $\lambda = \text{lcm}(p-1, q-1)$ is calculated [20–22], a random integer $g (g < N^2)$ is chosen to ensure the existence of $\mu = [L(g^{\lambda} \bmod n^2)]^{-1} \bmod N$. The public key is (N, g), and the private key is (λ, μ). The key generation center sends the same set of public and private keys to each intelligent terminal within the coverage of the same fog node and sends the corresponding public key to the fog node, which provides the public and private keys to EPSI.

3.1.2. Key Generation and Distribution in the Key Agreement Part. KGC randomly selects large prime numbers m_P, m_q, and $F(m_P)$ to generate a pseudorandom elliptic curve $E(m_P)$ and determines the generator P. If P is a base point of order m_q on the elliptic curve, and the cyclic group generated by the base point P will be G. Then, a secure hash function is selected. The construction method of H_1 is to first perform the point multiplication operation on the elliptic curve to obtain the point X, add the horizontal and vertical coordinate values of X, and then modulo m_q to complete the hash operation. The structure of H_2 is direct modulo. The construction method of H is to first

add the points of 3 points on the elliptic curve, then do the point multiplication, and add the two coordinate values to do a hash operation [23].

KGC randomly generates $x \in Z_q^*$, calculates $Y = x_P$, publicizes the parameters (m_P, m_q, P, H_1, H_2, and H), and keeps it confidential x. Each intelligent terminal SM_{ij}, fog node fog_j, and cloud node select account $ID_{SM_{ij}}$, ID_{fog_j}, and ID_{cloud} to, respectively, register. After successful registration, KGC provides the intelligent terminal with a partial private key $d_{SM_{ij}}$, intelligent terminal public key $R_{SM_{ij}}$, and fog node public key R_{fog_j} and x. Next, KGC provides the fog node with some private keys $d_{fog_{ij}}$, intelligent terminal public key $R_{SM_{ij}}$, fog node public key R_{fog_j}, cloud node public key R_{cloud} and x. Finally, KGC provides the cloud node with some private key d_{cloud}, cloud node public key R_{cloud}, fog node public key R_{fog_j} and x [24, 25].

Among them, for the account $ID_{SM_{ij}}$, KGC selects $r_{SM_{ij}} \in Z_q^*$, generates a public key $R_{SM_{ij}} = r_{SM_{ij}}P$, and produces a partial private key $d_{SM_{ij}} = [r_{SM_{ij}} + xH(ID_{SM_{ij}}, R_{SM_{ij}})] \bmod m_p$, thereby computing $P_{SM_{ij}} = d_{SM_{ij}}P$ for calculating the final K_2. For the account identity ID_{fog_j}, KGC generates the public key R_{fog_j} and part of the private key d_{fog_j} and also computes $P_{fog_j} = d_{fog_j}P$ to calculate the final K_2. For the account ID_{cloud}, KGC generates the public key R_{cloud} and part of the private key d_{cloud} and computes $P_{cloud} = d_{cloud}P$ to calculate the final K_2. KGC sends these public keys and some private keys to intelligent terminal SM_{ij}, fog node fog_j, and the cloud node through secure channels, respectively [26, 27].

After obtaining the public and private keys, user SM_{ij} can determine whether some of the private keys given by KGC are valid by calculating whether $H_1(ID_{SM_{ij}}, R_{SM_{ij}})Y = d_{SM_{ij}}P$ is established. In addition, fog node fog_j and the cloud node are the same.

3.2. Intelligent Terminal Data Report. In order to prevent the user's private data from being exposed to eavesdroppers in the "intelligent terminal-fog" communication link, the private data in the paper are chosen to encrypt in the intelligent terminal. Moreover, the data generated by the intelligent terminal is generally uploaded to the fog node periodically, assuming that the time gap is 15 min. Then, the intelligent terminal encrypts the real-time information data every 15 minutes, generates a signature on the encrypted data after the two parties of the transmission complete the key agreement, and uploads the data report to the corresponding fog node, and finally, waits for the fog node to aggregate it [28, 29].

Assuming that there are n intelligent terminals in a subregion, the information stored in the ith intelligent terminal SM_{ij} in the subregion is x_{ij} ($0 \le i \le n$, $0 \le j \le f$), and intelligent terminal SM_{ij} will perform the following operations.

3.2.1. Key Negotiation between the Fog Node and Intelligent Terminal. In order to prevent attackers from impersonating intelligent terminal and injecting false data or impersonating the fog node to eavesdrop on the data, this scheme builds a lightweight identity authentication based on the elliptic curve to confirm the identity of the operation user.

In order to prevent an attacker from eavesdropping on the key from KGC and pretending to be an intelligent terminal or node, the public key and some private keys are generated by KGC during system initialization. The long-term private key $x_{SM_{ij}}$ and temporary private key a_{ij} are generated by the intelligent terminal node itself, and the long-term private key x_{fog_j} and temporary private key b_j are generated by the fog node itself.

Given the user SM_{ij} identity $ID_{SM_{ij}}$, it calculates

$$\begin{aligned} T_{SM_{ij}} &= a_{ij}P, \\ h_1 &= H_2\left(T_{SM_{ij}} + ID_{SM_{ij}} + \text{nonce}\right), \\ s &= \left[a_{ij}{}^*\left(x_{SM_{ij}} + d_{SM_{ij}} + h_1\right) - 1\right] \bmod m_p, \end{aligned} \tag{1}$$

and sends the message $(ID_{SM_{ij}}, h_1, s, \text{nonce})$ to fog node fog_j, where nonce is the current time stamp. Next, wait for fog node fog_j's reply report. If the reply report is a retransmission command, then rekey negotiation will be performed. If the response report is $(ID_{fog_j}, h_3, s, \text{nonce})$, then determine whether the nonce is the time stamp sent by the intelligent terminal before. If it is, then calculate the $T_{fog_j}{}'$ according to the formula to determine whether $H_2(T_{fog_j}{}' + ID_{fog_j} + \text{nonce}) = h_3$ is established. If it is true, according to formula (2), calculate K_1, K_2, and K_3.

$$\begin{cases} K_{1_{ij}} = \left(X_{SM_{ij}} + d_{SM_{ij}} + a_{ij}\right)X_{fog_j}, \\ K_{2_{ij}} = \left(X_{SM_{ij}} + d_{SM_{ij}} + a_{ij}\right)P_{fog_j}, \\ K_{3_{ij}} = \left(X_{SM_{ij}} + d_{SM_{ij}} + a_{ij}\right)T_{fog_j}{}', \end{cases} \tag{2}$$

if it is not true, the negotiation fails, and fog node fog_j will be required to resend the verification message. Finally, user SM_{ij} calculates the K value according to

$$K_{ij} = H\left(ID_{SM_{ij}} \| ID_{fog_j} \| K_{1_{ij}} \| K_{2_{ij}} \| K_{3_{ij}}\right). \tag{3}$$

3.2.2. Raw Data Perception. The intelligent terminal uploads data every 15 minutes and generally consists of one integer and several decimals. In order to ensure the normal operation of the Paillier algorithm, the original data x_{ij} is multiplied by 10^n before encryption and a rounding operation is performed to retain n digits after the decimal point. Three digits after the decimal point are retained in the simulation

verification, but the proposed scheme can be generalized to more than one decimal point. It is calculated as follows:

$$x_{t_{ij}} = \left\lfloor xi_j \times 10^n \right\rfloor. \tag{4}$$

The more the number of reserved data bits, the greater the data calculation and transmission consumption will be, but the accuracy of the same data will be higher.

3.2.3. Original Data Encryption. In order to ensure the confidentiality of private data, this section uses the Paillier algorithm to encrypt the intelligent terminal to protect it from the threat of malicious attacks. In this encryption scheme, it is assumed that each intelligent terminal and EPSI shares a private key and a public key, but the private key is completely hidden from the fog node and the cloud node. In particular, the public key and the private key have been generated and distributed by KGC in the generation of system parameters. The encryption process of private data is as follows: random number $r_{ij} \in Z_q^*$ is selected, and for any plain text $x_{t_{ij}}$, the public key (N, g) is used to encrypt the cipher text CCCC obtained as

$$c_{t_{ij}} = E\left[x_{t_{ij}}, r_{ij}\right] = g^{x_{t_{ij}}} \cdot r_{ij}^N \bmod N^2. \tag{5}$$

Each intelligent terminal packages the encrypted data $c_{t_{ij}}$ and session key K_{ij} into an intelligent terminal data report, which is uploaded to the corresponding fog node fog_j every 15 minutes.

3.3. Fog Node Fine-Grained Aggregation Report. The operation of directly uploading explosively increased intelligent terminal data to the cloud will generate a large amount of transmission energy consumption and increase the bandwidth burden, which can make it difficult to meet the needs of low-latency transmission. Therefore, this section reduces the data traffic at the core network by introducing a fog node and further reduces the amount of data by performing relevant calculations at the fog node, which can reduce data transmission energy consumption. What is more, the Paillier algorithm used by intelligent terminal encryption has good addition homomorphism, which can support the addition calculation of data in the encrypted state, and obtain the correct data result after decryption. In addition, this homomorphic encryption feature ensures the privacy of the data on the fog side, even if an attacker maliciously eavesdrops, who cannot obtain the private data plain text, thereby effectively protecting the data security.

3.3.1. Identity Negotiation between the Fog Node and Intelligent Terminal. Given the fog node fog_j identity ID_{fog_j}, it checks whether the nonce sent from the intelligent terminal SM_{ij} is time-sensitive; that is, the current time stamp nonce is obtained and verified whether $\text{nonce}' - \text{nonce} \le \Delta\text{nonce}$ is established. If it is not established, the key negotiation fails, and user SM_{ij} will be required to resend the authentication message. If it is true, $T_{\text{SM}_{ij}}{}' = s(X_{\text{SM}_{ij}} + R_{\text{SM}_{\text{ij}}} + h_2 Y + h_1 P)$ will be calculated according to the formula $T_{\text{fog}_j}{}'$ to determine whether $H_2(T_{\text{SM}_{ij}}{}' + \text{ID}_{\text{SM}_{ij}} + \text{nonce}) = h_1$ is true. If it is true, the formulas (6), (7), and (8) will calculate and $(\text{ID}_{\text{fog}_j}, h_3, s, \text{nonce})$ will be returned to user SM_{ij}

$$T_{\text{fog}_j} = b_j P, \tag{6}$$

$$h_3 = H_2\left(T_{\text{fog}_j} + \text{ID}_{\text{fog}_j} + \text{nonce}\right), \tag{7}$$

$$\text{s} = \left[\text{b}_*^{\text{j}}\left(\text{x}_{\text{fog}_j} + \text{d}_{\text{fog}_j} + \text{h}_3\right)\right]^{-1} \bmod \text{m}_\text{p} \tag{8}$$

Then, according to formula (9), K_1, K_2, and K_3 are calculated.

$$\begin{cases} K_{1_{ij}} = x_{\text{fog}_j}\left(X_{\text{SM}_{ij}} + P_{\text{SM}_{ij}} + T_{\text{SM}_{ij}}{}'\right), \\ K_{2_{ij}} = d_{\text{fog}_j}\left(X_{\text{SM}_{ij}} + P_{\text{SM}_{ij}} + T_{\text{SM}_{ij}}{}'\right), \\ K_{3_{ij}} = b_j\left(X_{\text{SM}_{ij}} + P_{\text{SM}_{ij}} + T_{\text{SM}_{ij}}{}'\right). \end{cases} \tag{9}$$

Finally, fog node fog_j calculates the K value according to

$$K_{ij} = H\left(\text{ID}_{\text{SM}_{ij}} \left\| \text{ID}_{\text{fog}_j} \right\| K_{1_{ij}} \left\| K_{2_{ij}} \right\| K_{3_{ij}}\right). \tag{10}$$

3.3.2. Fog Node Identity Authentication. The session key K_{ij} is extracted from the data report sent from the intelligent terminal and compared with the corresponding session key K_{ij} in the fog node. If they are consistent, the encrypted data in the data report will be received and will wait for the next aggregation. If they are inconsistent, they will be discarded. So far, the key agreement and identity authentication between the device layer and the fog layer are completed. The complete process is shown in Figure 2

3.3.3. Fog Node Fine-Grained Aggregation. At the fog node, for the data $C_j = \{ct_{0_j}, c_{t1_j}, \cdots, c_{tn_j}\}$, it is encrypted in the user report sent by the coverage intelligent terminal and added aggregation; that is, the data is multiplied in C_j:

$$\begin{aligned} \text{Sum}_j &= c_{t0_j} \cdot c_{t1_j} \cdot \cdots \cdot c_{tn_j} \\ &= E\left[x_{t0_j}, r_{0_j}\right] \cdot E\left[x_{t1_j}, r_{1_j}\right] \cdot \cdots \cdot E\left[x_{tn_j}, r_{n_j}\right] \\ &= g^{x_{t0_j} + x_{t1_j} + \cdots + x_{tn_j}} \cdot \left(r_{0_j} \cdot r_{1_j} \cdot \cdots \cdot r_{n_j}\right)^N \bmod N^2. \end{aligned} \tag{11}$$

3.3.4. Fog Node and Cloud Node Key Agreement. Before uploading, a key agreement is performed on the fog node fog_j and cloud node again to calculate the session key K_j

$$K_j = H\left(ID_{fog_j} \| ID_{cloud} \| K_{1_j} \left\| K_{2_j} \right\| K_{3_j}\right) \tag{12}$$

3.3.5. Fog Node Fine-Grained Aggregation Report Generation. The aggregated cipher text Sum_j and session key K_j of fog

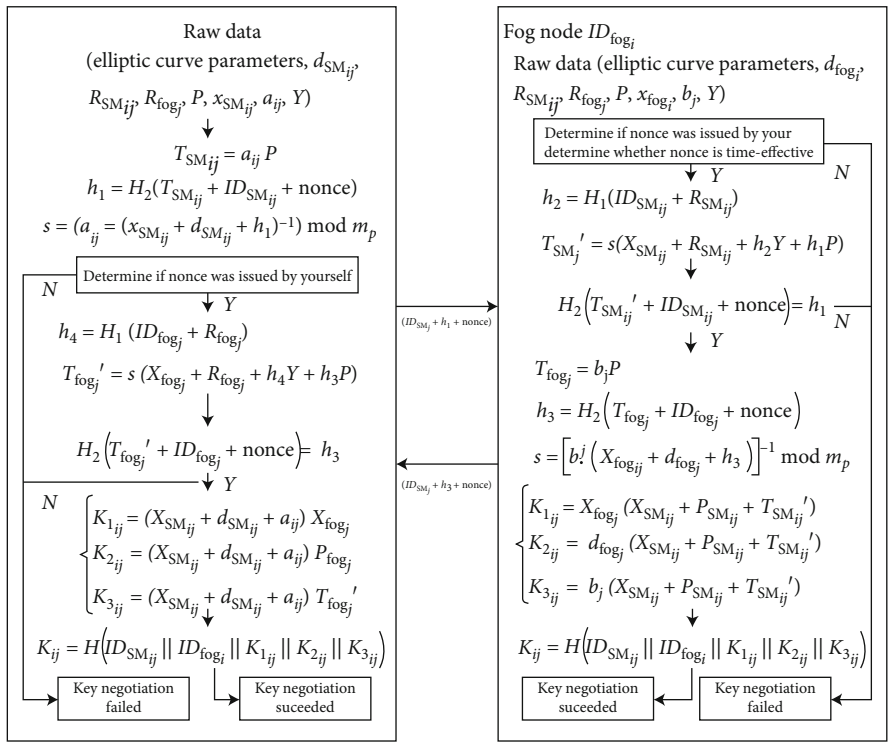

Figure 2: Key agreement authentication process.

node fog_j are packaged into a fine-grained aggregated report of the fog node and sent to the corresponding cloud node

3.4. Cloud Node Coarse-Grained Aggregation Report. The cloud received f encrypted aggregate data from f fog nodes within the coverage of the cloud node. In order to perform multigranular aggregation on the data in this area, the data obtained after the final EPSI decryption can be accurate to the fog node layer, and Horner rules are introduced in this section to complete the coarse-grained aggregation of the data. Horner rules can not only provide aggregation and parsing operations, since they use the least multiplication strategy, but also reduce the energy consumption caused by the calculation.

3.4.1. Cloud Node and Fog Node Key Agreement. Given the cloud node identity ID_{cloud}, key agreement is performed with fog node fog_j and the session key K_j is calculated.

$$K_j = H\left(\mathrm{ID}_{\mathrm{fog}_j} \| \mathrm{ID}_{\mathrm{cloud}} \| K_{1_j} \left\| K_{2_j} \right\| K_{3_j}\right). \tag{13}$$

3.4.2. Cloud Node Identity Authentication. The session key K_j from the fog node fine-grained aggregation report is compared with the corresponding session key K_j in the cloud node. If they are consistent, the fine-grained aggregated data will be received in the fog node report and the next aggregation will be waited for. If they are inconsistent, they will be discarded.

3.4.3. Cloud Node Coarse-Grained Aggregation. For the fog-level fine-grained aggregated data set $\mathrm{Sum}_y = \{\mathrm{Sum}_0, \mathrm{Sum}_1, \cdots, \mathrm{Sum}_f\}$ from the fog node, x_h is selected to satisfy $x_h > \mathrm{Sum}_j (j \in \{0, 1, 2, \cdots, f\})$ as a parameter for Horner aggregation.

$$\mathrm{Sum}_c = \left\{\cdots\left[\left(\mathrm{Sum}_f x_h + \mathrm{Sum}_{f-1}\right) x_h + \mathrm{Sum}_{f-2}\right]\right\} \cdots \mathrm{Sum}_1 x_h + \mathrm{Sum}_0. \tag{14}$$

3.4.4. Cloud Node Coarse-Grained Aggregation Report Generation. The $n \times f$ intelligent terminal data within the coverage of a cloud node is aggregated into a data Sum_c in the cloud and transmitted to EPSI through a secure channel.

3.5. EPSI Aggregation Report Reading. EPSI receives coarse-grained aggregated data from the cloud Sum_c. Due to the resolvability of Horner's rule, the aggregated data can be parsed into fog-level fine-grained aggregated data of f fog nodes to provide differentiated data services for users.

3.5.1. Horner Analysis. The coarse aggregated data Sum_c is analyzed in the cloud.

$$\begin{aligned} &\text{for } i = 0 : f, \\ &\mathrm{Sum}_i = \bmod\left(\mathrm{Sum}_c, x_h\right), \\ &\mathrm{Sum}_c = \frac{\mathrm{Sum}_c - \mathrm{Sum}_i}{x_h}. \end{aligned} \tag{15}$$

Through the Horner rule analytical formula, the fine-grained aggregated data of each fog node $\mathrm{Sum}_j (j \in \{0, 1, 2, \cdots, f\})$ is obtained.

3.5.2. Decryption of Fog-Level Fine-Grained Aggregated Data. The fog-level fine-grained aggregated data Sum_j is decrypted to obtain the plaintext m_{Sum_j} of the fine-grained aggregated data in each fog node

$$D\left[c_{t0_j} \cdot c_{t1_j} \cdot \cdots \cdot c_{fj}\right] = D = \left[E\left(x_{\mathrm{to}_j}, r_{0j}\right)\right] \cdot E\left(E_{t1_j}, r_{0j}\right) \cdot \cdots \cdot E\left(x_{tn_j}, r_{nj}\right) \bmod N^2 \cdot \left(x_{t_{0j}} + x_{t1_j} + \cdots + x_{tn_j}\right) \bmod N = m_{\mathrm{Sum}_j}. \tag{16}$$

Since the original data is multiplied by 10^n before, the original data is uploaded and the rounding operation is performed to retain the n digits after the decimal point. After EPSI decrypted the privacy protection data to obtain the plain text m_{Sum_j}, it is necessary to divide the data by 10^n to restore the data.

$$M_{\mathrm{Sum}_j} = m_{\mathrm{Sum}_j} \times 10^{-n}. \tag{17}$$

EPSI performs data mining on the fine-grained aggregated data of these fog nodes, and the cloud-level aggregated data are obtained by adding them together to provide differentiated real-time data support for scheduling.

In addition, EPSI can also package and send the fine-grained aggregated data of each subregion in plain text back to the fog node of each subregion, so that users can query with low latency and save EPSI computing processing resources.

4. Discussion

4.1. Security Analysis. This part mainly analyzes the security of this scheme from the aspects of privacy, confidentiality, and integrity and compares the security with the existing privacy protection data aggregation PPADA scheme.

4.1.1. Privacy. Privacy data is always encrypted when it is uploaded to the fog node and the cloud node which do not have permission to obtain the decryption key. Therefore, even if the fog node or the cloud node tries to eavesdrop on the private data, it can only obtain the private data cipher text instead of the plain text. Finally, EPSI sends the total real-time information of each sub-region to the fog node in plain text. At this time, the fog node and the cloud node receive the total information volume of the subregion instead of the information volume of a single user, which can guarantee the privacy of data and effectively respond to threats on the fog node and the cloud node.

4.1.2. Confidentiality. Privacy data is encrypted when it is transmitted in each unsecured communication link of the system model, and even if the eavesdropper eavesdrops on the private data cipher text, it cannot obtain valid data plain text without the decryption key. Finally, when EPSI sends the real-time data of each subregion back to the fog node, the transmission data is the total information volume of the subregion instead of the information volume of a single user, which does not expose user privacy. Therefore, it guarantees the confidentiality of the data and can effectively deal with the threat of eavesdropper eavesdropping on the communication link.

4.1.3. Integrity. The solution in this paper uses a lightweight key agreement identity authentication. Before each data is uploaded, the two parties of the session conduct a key agreement to facilitate identity authentication when the data is uploaded. Once the session keys of the two parties are inconsistent, if one of the two parties in the session or both parties are not legal entities but the attacker is disguised, the identity authentication will fail, and the data will be discarded and reissued. Moreover, the identity authentication scheme effectively avoids malicious injections caused by the identity masquerading of the data sender and node eavesdropping attacks caused by the identity masquerading of the data recipient, which can ensure the integrity of the data.

4.1.4. Security Comparison. The PPADA scheme, respectively, uses the Paillier encryption scheme and blind signature to ensure the confidentiality and integrity of the data. However, since the private data is decrypted in the fog node and compiled into the database, the privacy of the data cannot be guaranteed at this time. In particular, the data aggregation scheme combining the Horner rule and the Paillier encryption algorithm proposed in the paper can meet this challenge. While ensuring data privacy, the electricity bill is directly generated by the control center and passed back to each user through the fog node.

In summary, the scheme in the paper can guarantee the privacy, confidentiality, and integrity of data during transmission, which has more advantages in terms of security.

4.2. Identity Authentication Performance. The performance of the identity authentication scheme is mainly compared in three aspects: the number of operations, the number of dot multiplications, and the number of communications between both parties in the key agreement. Security starts with four aspects: antieavesdropping on session keys, antieavesdropping on long-term public keys, antispoofing attacks, and two-way authentication. Compared with the scheme, the performance and safety are compared as shown in Table 1.

The number indicates the number of times, "√" indicates that this aspect of security, and "×" indicates that it does not have this aspect of security. In particular, "Φ1" stands for antieavesdropping of session keys, "Φ1" refers to antieavesdropping of long-term public keys, and "Φ1" indicates anti-identity fraud attacks. "Φ1" stands for two-way authentication.

As shown in Table 1, the identity authentication scheme in the paper is lower than other operations, point multiplication, and communication times, which reflects its lightness

Table 1: Comparison of identity authentication performance and security.

Program	Pair operation	Point multiplication	Communication times	$\Phi 1$	$\Phi 2$	$\Phi 3$	$\Phi 4$
Literature [16]	0	5	2	√	√	√	×
Literature [17]	0	2	4	√	√	√	×
Literature [18]	0	5	3	√	√	√	×
Literature [19]	1	5	2	√	√	×	×
Literature [20]	0	5	2	√	√	√	√
Literature [21]	0	5	2	√	√	√	√
This article	0	4	2	√	√	√	√

and also has the characteristics of antieavesdropping of session keys, antieavesdropping of long-term public keys, antiidentity fraud attacks, and two-way authentication. Through the comparison of the seven performance indicators in Table 1, the solution proposed in the paper is more secure than that of the literature.

4.3. Performance Comparison Test. The performance of the proposed solution is evaluated in terms of the computational cost of each entity, communication overhead between entities, and resource distribution and compared with the existing PPUAC scheme, PADF scheme, and the constructed one-time aggregation scheme (called SIG-ADD), where the SIG-ADD achieves the same result as this solution; that is, EPSI can obtain the fine-grained aggregated data of each fog node in plain text. Supposing that the fog node performs fine-grained aggregation on the data from the intelligent terminal in the coverage area, then it is up loaded to the cloud node. What is more, the cloud node no longer aggregates the data but directly forwards it to EPSI, and EPSI uses the Paillier decryption algorithm to solve each fog node granular aggregated data plain text. The simulation data in this part comes from the real data of residents provided by the Energy Control Committee of the Irish Social Science Data Archive, which is shown in Table 2.

A large prime number $m_p = 3701$ and $m_q = 37$ are taken to generate the curve used for key negotiation, and Paillier-encrypted large prime numbers p and q are randomly generated by a big integer class.

4.3.1. Calculation Overhead. Assuming that the EPSI management area is divided into y sub-areas, there is one cloud node under each subarea, there are f fog nodes under each cloud node, and there are n intelligent terminals in each fog node area. In the simulation, it is assumed that there is 1 cloud node, i.e., $y = 1$; there are 3 fog nodes, i.e., $f = 3$.

The symbol T_{eZ} represents the computational cost of an exponential operation on Z_N^{*2}, the symbol T_{eZ} refers to the computational cost of an exponential operation on G, the symbol T_{mG} represents the computational cost of a multiplication operation on G, and the computational cost of bilinear pair operation is T_p.

(i) *Intelligent Terminal* SM_{ij}. Both the scheme in the paper and the SIG-ADD scheme use the Paillier algorithm for encryption, which requires a total of $2n$ times exponential modulus finger operations T_{eZ}, and the key negotiation part requires a $2n$ times dot product algorithm. In particular, the PPUAC scheme requires $2n$ times exponential modulus multiplications T_{eZ} and n times G multiplication T_{mG}, while the PDAF scheme requires $2n$ exponential modulus multiplications T_{eZ}, n times G multiplications T_{mG}, and twice times bilinear pair calculation T_p.

(ii) *Fog Node* fog_j. In order to complete fine-grained aggregation, the scheme of this paper and the SIG-ADD scheme need to perform n times of multiplication operations on Z_N^{*2} and perform the key agreement $(2n + 2f)$ times multiplication algorithm. In addition, the PPUAC scheme requires $(n + 2)$ times bilinear pair calculation T_p and n times G to multiply T_{mG}, while the PDAF scheme requires $(n + f + 2)$ times bilinear pair calculation T_p and $(n + 1)$ times exponential modular multiplication operation T_{eZ}.

(iii) *Cloud Node*. In the scheme of this paper, when performing coarse-grained aggregation, f times upper multiplication and key agreement $2f$ times the dot product algorithm on G are performed. The SIG-ADD scheme only performs $2f$ times the dot product algorithm for key agreement, and the PPUAC scheme performed $(f + 2)$ times bilinear pair calculation T_p and the f times the multiplication of T_{mG} on the cloud node.

(iv) *EPSI*. The decryption of this program and the SIG-ADD program requires $2f$ times exponential modulus finger operation T_{eZ} and $3f$ times G upper modulus operation T_{eG}. In addition to these calculation

Table 2: Simulation test environment configuration.

Configuration	Model
Processor	Intel Core i5 9600k six-core 3.70 GHz
RAM	16 GB DDR4
Operating system	Win 10 Professional Edition
Programming environment	Eclipse

Use the big integer class in JAVA to complete the calculation of large numbers in the Paillier encryption algorithm.

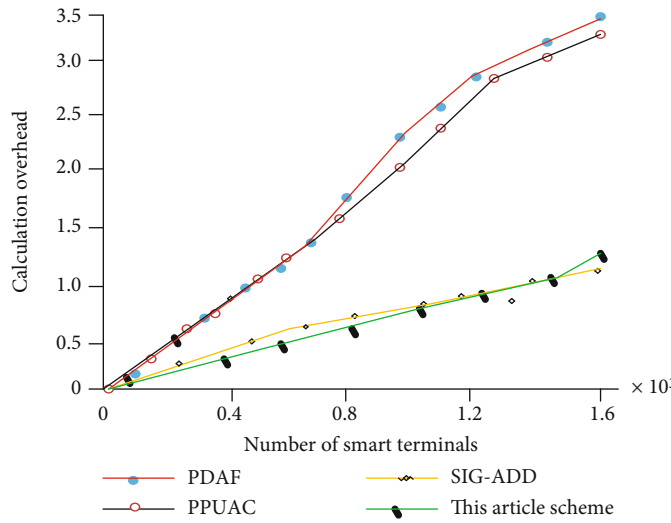

Figure 3: Comparison of calculation costs of the four schemes.

overheads, the PPUAC scheme also needs to perform $2f$ times bilinear pair calculation T_p. In addition to the above calculation overhead, the PDAF scheme also needs to perform $3f$ times bilinear pair calculation T_p.

Among them, the upper multiplication operation Z_N^{*2} is negligible relative to T_{eZ} and T_p, and the point multiplication operation on the elliptic curve is generally replaced by cumulative point addition, and it can also be ignored.

The comparison of the computational costs of the four schemes in the entire system model is shown in Figure 3.

It can be seen that the calculation overhead generated by the scheme in this paper and the SIG-ADD scheme in the entire system model is almost the same and is much smaller than the PPUAC scheme and PDAF scheme. As the number of intelligent terminals increases, this advantage is more obvious. Since the scheme in this paper adopts a lightweight key agreement identity authentication scheme and compares with the complicated and cumbersome bilinear pairing authentication, the calculation overhead generated is smaller.

4.3.2. Communication Overhead. Supposing that the symbol $C_{\text{usertofog}}$ indicates the length of data sent by the intelligent terminal user to the corresponding fog node, the symbol $C_{\text{fogtocloud}}$ indicates the length of data sent by the fog node to the corresponding cloud node, the symbol $C_{\text{cloudtoEPSI}}$ indicates the length of data sent by the cloud node to EPSI. If the parameter N is 64 bits, then the number of Paillier cipher text data bits will be 128 bits. So, it is

$$\begin{aligned} C_{\text{usertofog}} &= 128 + \left|\text{ID}_{\text{SM}_{ij}}\right| + \left|\text{ID}_{\text{fog}_j}\right| + |K|, \\ C_{\text{fogtocloud}} &= 128 + \left|\text{ID}_{\text{fog}_j}\right| + |\text{ID}_{\text{cloud}}| + |K|, \\ C_{\text{cloudtoEPSI}} &= 128 + |\text{ID}_{\text{cloud}}| + |\text{ID}_{\text{EPSI}}| + |K|. \end{aligned} \tag{18}$$

Then, the total communication cost C_{all} of this solution is

$$C_{\text{all}} = \text{yfn}C_{\text{usertofog}} + \text{yf}C_{\text{fogtocloud}} + \text{y}C_{\text{cloutoEPSI}}. \tag{19}$$

Among them, the $C_{\text{usertofog}}$ and $C_{\text{fogtocloud}}$ of the SIG-ADD scheme are consistent with the calculation formula of this scheme, while the calculation formula of $C_{\text{cloudtoEPSI}}$ is

$$C_{\text{cloudtoEPSI}}{}' = 128 + \left|\text{ID}_{\text{fog}_j}\right| + |\text{ID}_{\text{cloud}}| + |\text{ID}_{\text{EPSI}}| + |K|. \tag{20}$$

Therefore, the total communication cost $C_{\text{SIG-ADD}}$ of the SIG-ADD scheme is

$$C_{\text{SIG-ADD}} = \text{yfn}C_{\text{usertofog}} + \text{yf}C_{\text{fogtocloud}} + \text{yf}C_{\text{cloudtoESI}}{}'. \tag{21}$$

Supposing there are 3 cloud nodes, the length of each ID is 160 bits and the length of each session key K is 256 bits, the communication overhead of the scheme in this paper, the SIG-ADD scheme, the PPUAC scheme, and the PDAF scheme are analyzed in the intelligent terminal to the cloud node, cloud node to EPSI communication link under different fog node numbers. Figure 4 shows a comparison of the communication overhead of the four schemes for transmitting data on the link between the intelligent terminal and cloud node.

Figure 5 shows the comparison of the communication overhead of the four schemes for transmitting data on the link between the cloud node and EPSI.

It can be seen from Figures 4 and 5 that in the communication link from the intelligent terminal to cloud node, the communication overhead of the PPUAC scheme and the PDAF scheme continues to increase with the increase of the f5og node. When the number of the fog node is 5, the communication overhead has, respectively, reached 13000 bits and 12000 bits. Meanwhile, the communication overhead of this scheme and the SIG-ADD scheme is almost the same

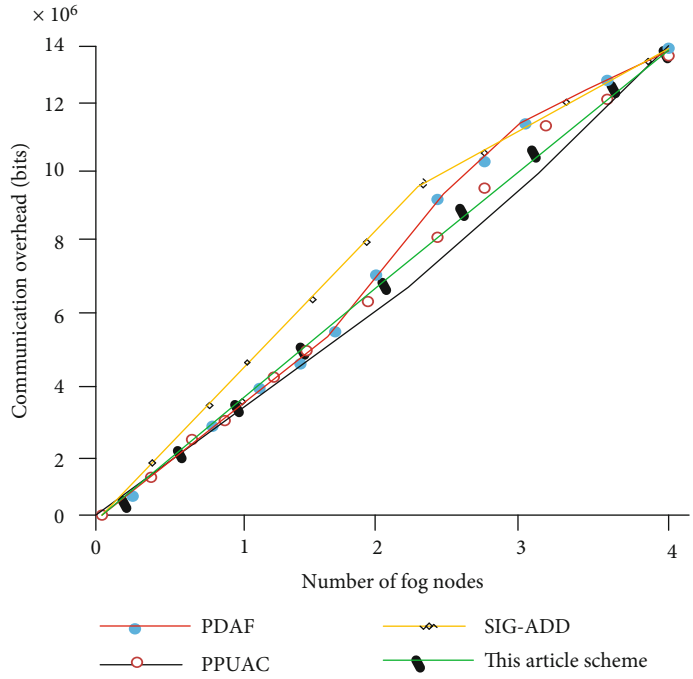

Figure 4: Comparison of communication overhead from the smart terminal to the cloud node.

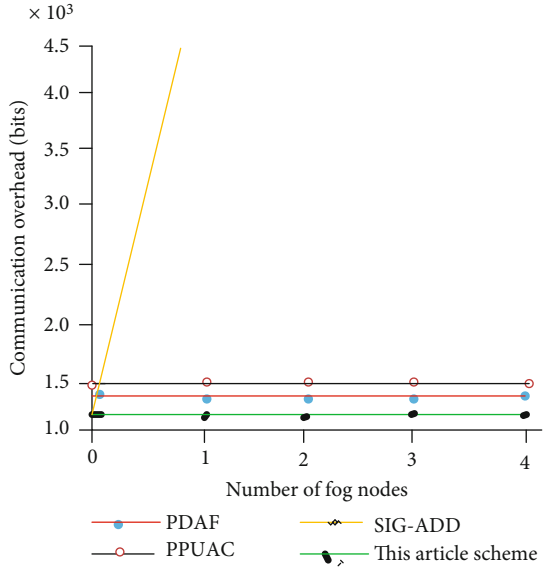

Figure 5: Comparison of communication costs between cloud nodes and service agencies.

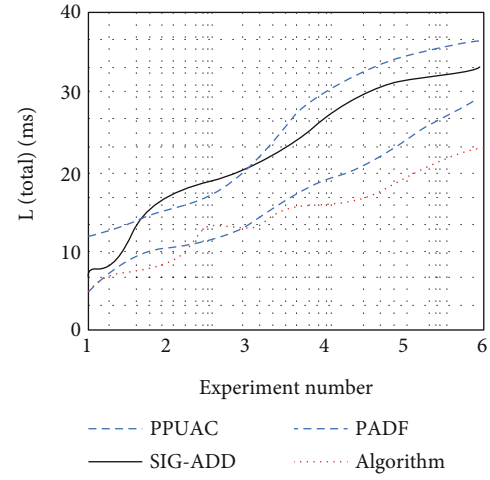

Figure 6: Comparison of L (total) distribution.

at this stage. In the communication link from the cloud node to EPSI, the communication overhead of the SIG-ADD scheme increases linearly with the increase of the number of fog nodes, and when the number of fog nodes exceeds 3, the communication overhead has reached more than 8000 bits. The communication overhead of the scheme of this paper, PPUAC scheme, and PDAF scheme is maintained at the same level and the communication overhead of this scheme is lower. In summary, the total communication overhead of the solution in this paper on the entire communication link is much lower than that of the other three solutions. Additionally, as the fog node increases, the advantages of this solution are more obvious. It shows that the scheme in this paper is a lightweight privacy data multilevel aggregation scheme. Based on lightweight identity authentication and multilevel aggregation considerations, the data transmission overhead is effectively reduced.

4.3.3. Resource Distribution Assessment. Different intelligent environments have different intelligent node networks and different rule sets. In order to verify the effectiveness of the distribution algorithm in the paper, the distribution mechanism in the paper is compared with the centralized distribution and ordinary distribution mechanisms. The two sets of data are designed as follows.

Table 3: Stdsen under different number of rules.

Experiment number	1	2	3	4	5	6
Algorithm	3.85	3.97	9.09	13.31	17.58	22.65
PPUAC	7.39	12.67	19.76	26.34	33.75	39.44
PADF	4.27	6.44	8.28	12.68	18.57	26.76
SIG-ADD	6.34	12.38	17.68	21.88	30.69	38.28

Under the same simulation environment of the intelligent node network, six groups of experiments with an increasing number of rules are set. The experiment numbers and corresponding rules are shown in Table 2, and the L (total) obtained by the above three allocation mechanisms in six sets of experiments is shown in Figure 6.

As can be seen from Figure 6, as the rules grow, the inference network becomes more complicated, and the delay of the centralized distribution increases sharply. Meanwhile, the real-time performance is very poor. Compared with the other three distributed real-time performances, the real-time performance has been significantly improved. Moreover, the real-time performance has been further improved on the basis of ordinary distribution with the algorithm proposed in the paper.

For the centralized distribution, it is not necessary to evaluate the resource balance. The Stdsen obtained by the distributed mechanism in six sets of experiments is shown in Table 3.

From Table 3, it can be obtained that the resource utilization of the algorithm in the paper is obviously better than that of the other three distributed types, and the optimization degree is more obvious when the number of rules is large.

5. Conclusions

A data aggregation scheme for intelligent network security and privacy protection in the paper is proposed based on fog computing in view of the hidden security risks faced by intelligent network data collection and transmission. Moreover, the key generation center in the solution is not completely trusted. In particular, by means of the point-plus-add feature of the elliptic curve, the authentication speed can be sped up. Meanwhile, with the advantage of data aggregation, the amount of data transmission can be lowered, which further reduces communication overhead. Therefore, simulation experiments have further confirmed the performance advantages of the proposed scheme in terms of security, practicality, calculation, and communication overhead. In the future, the theory of data space-time compression and network resource optimization will be considered integrating to further improve network system performance.

Data Availability

All data included in this study are available upon request by contact with the corresponding author.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

References

[1] Z. Wang, Z. F. Ma, and S. S. Luo, "Efficient authentication key agreement protocol for mobile internet based on identity," *Journal of Communications*, vol. 17, no. 8, pp. 213–222, 2017.

[2] Y. W. Zhou, B. Yang, and W. Z. Zhang, "An improved certificateless two-party authentication key agreement protocol," *Chinese Journal of Computers*, vol. 234, no. 5, pp. 512–525, 2017.

[3] H. Shen and M. W. Zhang, "A privacy-protected multi-level user power aggregation control scheme for smart grids," *Journal of Cryptography*, vol. 643, no. 2, pp. 356–370, 2016.

[4] H. W. Hui, C. C. Zhou, S. G. Xu, and F. Lin, "A novel secure data transmission scheme in industrial internet of things," *China Communications*, vol. 17, no. 1, pp. 73–88, 2020.

[5] S. G. Chen, S. J. Zhang, X. Y. Zheng, and X. Ruan, "Layered adaptive compression design for efficient data collection in industrial wireless sensor networks," *Journal of Network and Computer Applications*, vol. 129, pp. 37–45, 2019.

[6] E. Vahedi, M. Bayat, M. R. Pakravan, and M. R. Aref, "A secure ECC-based privacy preserving data aggregation scheme for smart grids," *Computer Networks*, vol. 129, pp. 28–36, 2017.

[7] Y. Wang, "Architecture selection of laboratory information management system," *Chemical Engineering and Equipment*, vol. 35, no. 1, pp. 367–380, 2020.

[8] L. Xu and H. C. Huang, "Discussion on the impact of 5G wireless network architecture on transmission networks," *Information and Computer*, vol. 90, no. 3, pp. 23–40, 2019.

[9] J. Xing, "Analysis of library's wireless network architecture," *Digital Communications World*, vol. 17, no. 11, pp. 567–579, 2017.

[10] J. L. Yu, Z. X. Zhu, and C. Y. Li, "A comparative study of the old and new Hadoop MapReduce architectures," *Computer and Digital Engineering*, vol. 150, no. 8, pp. 211–232, 2017.

[11] R. T. Chen, "Cloud computing architecture and its key technologies," *Electronic Technology and Software Engineering*, vol. 271, no. 1, pp. 341–354, 2017.

[12] D. L. Lu and S. B. Zhu, "A review of big data and its architecture and key technologies," *Journal of the Academy of Equipment*, vol. 45, no. 2, pp. 3212–3230, 2017.

[13] X. T. Li, H. J. Liang, and M. Zhang, "Construction of power data center based on cloud computing architecture," *China New Telecommunications*, vol. 21, no. 12, pp. 45–61, 2017.

[14] L. Wang and Z. G. Wang, "Research and application of improved three-tier architecture," *Computer Engineering and Design*, vol. 11, no. 7, pp. 2320–2341, 2017.

[15] D. L. Lu and S. B. Zhu, "Overview of big data and its architecture and key technologies," *Journal of the Academy of Equipment*, vol. 17, no. 8, pp. 300–322, 2017.

[16] F. H. Lin, Y. T. Zhou, X. S. An, I. You, and K.-K. R. Choo, "Fair resource allocation in an intrusion-detection system for edge computing: ensuring the security of internet of things devices," *IEEE Consumer Electronics Magazine*, vol. 7, no. 6, pp. 45–50, 2018.

[17] W. S. Shi, H. Sun, J. Cao et al., "Edge computing: a new computing model in the era of the internet of everything," *Journal*

of Computer Research and Development, vol. 17, no. 8, pp. 27–50, 2017.

[18] T. Cui, "Innovative and optimized infrastructure to ensure production and operation safety," *Financial Electronics*, vol. 12, no. 2, pp. 210–222, 2017.

[19] H. Zhang, J. B. Zhang, and Y. Zhou, "Research on physical layer security technology of large-scale distributed systems," *Radio Engineering*, vol. 28, no. 1, pp. 113–122, 2019.

[20] X. M. Liao, S. L. Feng, and W. J. Tang, "Physical layer security strategy of wireless mesh network combining full duplex and cooperative interference," *Telecommunications Technology*, vol. 10, no. 7, pp. 200–212, 2018.

[21] H. Y. Song, Y. Y. Gao, and N. Sha, "A method to effectively improve the physical layer security performance of RFID system," *Computer Engineering*, vol. 17, no. 5, pp. 112–122, 2018.

[22] Y. Lu, Y. Chen, T. Li et al., "Construction method of edge FPGA-oriented embedded FPGA convolutional neural network," *Computer Research and Development*, vol. 134, no. 3, pp. 2212–2222, 2018.

[23] Y. Z. Fu and D. S. Li, "Application-driven network delay measurement and optimization technology in edge computing environment," *Journal of Computer Research and Development*, vol. 17, no. 8, pp. 213–222, 2018.

[24] Z. M. Zhao, F. Liu, Z. P. Cai, and N. Xiao, "Edge computing: platforms, applications and challenges," *Journal of Computer Research and Development*, vol. 13, no. 8, pp. 116–122, 2018.

[25] J. T. Su, F. H. Lin, X. W. Zhou, and X. Lu, "Steiner tree based optimal resource caching scheme in fog computing," *China Communications*, vol. 12, no. 8, pp. 161–168, 2015.

[26] T. Wang, L. Qiu, A. K. Sangaiah, A. F. Liu, M. Z. A. Bhuiyan, and Y. Ma, "Edge-computing-based trustworthy data collection model in the internet of things," *IEEE Internet of Things Journal*, vol. 7, no. 5, pp. 4218–4227, 2020.

[27] T. Wang, P. Wang, S. B. Cai, Y. Ma, A. Liu, and M. Xie, "A unified trustworthy environment Establishment based on edge computing in industrial IoT," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 9, pp. 6083–6091, 2020.

[28] Y. Ma, Y. Sun, Y. J. Lei, N. Qin, and J. Lu, "A survey of blockchain technology on security, privacy, and trust in crowdsourcing services," *World Wide Web*, vol. 23, no. 1, pp. 393–419, 2020.

[29] C. J. Xiao, C. Liu, Y. Ma, Z. Li, and X. Luo, "Time sensitivity-based popularity prediction for online promotion on twitter," *Information Sciences*, vol. 52, no. 5, pp. 82–92, 2020.