SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

March 2019, Vol. 62 032103:1–032103:14
https://doi.org/10.1007/s11432-018-9451-y

EFFECT: an efficient flexible privacy-preserving data aggregation scheme with authentication in smart grid

Zhitao GUAN[1], Yue ZHANG[1], Liehuang ZHU[2*], Longfei WU[3] & Shui YU[4]

[1]*School of Control and Computer Engineering, North China Electric Power University, Beijing* 102206*, China;*
[2]*School of Computer, Beijing Institute of Technology, Beijing* 100081*, China;*
[3]*Department of Mathematics and Computer Science, Fayetteville State University, Fayetteville* 28301*, USA;*
[4]*School of Information Technology, Deakin University, Burwood* 3125*, Australia*

Received 27 February 2018/Accepted 14 May 2018/Published online 11 January 2019

Abstract Smart grid is considered as a promising approach to solve the problems of carbon emission and energy crisis. In smart grid, the power consumption data are collected to optimize the energy utilization. However, security issues in communications still present practical concerns. To cope with these challenges, we propose EFFECT, an efficient flexible privacy-preserving aggregation scheme with authentication in smart grid. Specifically, in the proposed scheme, we achieve both data source authentication and data aggregation in high efficiency. Besides, in order to adapt to the dynamic smart grid system, the threshold for aggregation is adjusted according to the energy consumption information of each particular residential area and the time period, which can support fault-tolerance while ensuring individual data privacy during aggregation. Detailed security analysis shows that our scheme can satisfy the desired security requirements of smart grid. In addition, we compare our scheme with existing schemes to demonstrate the effectiveness of our proposed scheme in terms of low computational complexity and communication overhead.

Keywords privacy-preserving, authentication, batch verification, smart grid

Citation Guan Z T, Zhang Y, Zhu L H, et al. EFFECT: an efficient flexible privacy-preserving data aggregation scheme with authentication in smart grid. Sci China Inf Sci, 2019, 62(3): 032103, https://doi.org/10.1007/s11432-018-9451-y

1 Introduction

Smart grid has emerged as one of the most important trend in the next-generation smart technologies. The smart grid integrates the traditional power grid with nearly real-time communication system and intelligent control system [1, 2]. As shown in Figure 1, smart grid realizes energy optimization through bidirectional control of information flow and energy flow. Smart meters (SMs) are the essential components in smart grid, which are deployed at consumers side to record energy consumption periodically. Control center (CC) is able to devise a better power generation plan to intelligently balance the consumption between peak and off-peak periods.

However, the collection of electricity usage data may cause the leakage of user privacy-sensitive information that threaten the user's privacy [3, 4]. Attackers could eavesdrop the communication messages between SMs and CC, so as to predict the users' living habits from the respective electricity demand. What's more, an attacker could modify the in-transit messages, forge a fake message, or launch a replay attack to induce wrong and even disastrous actions. Therefore, it is of great significance to ensure user privacy and data integrity, as well as conducting source authentication in smart grid communications.

* Corresponding author (email: liehuangz@bit.edu.cn)

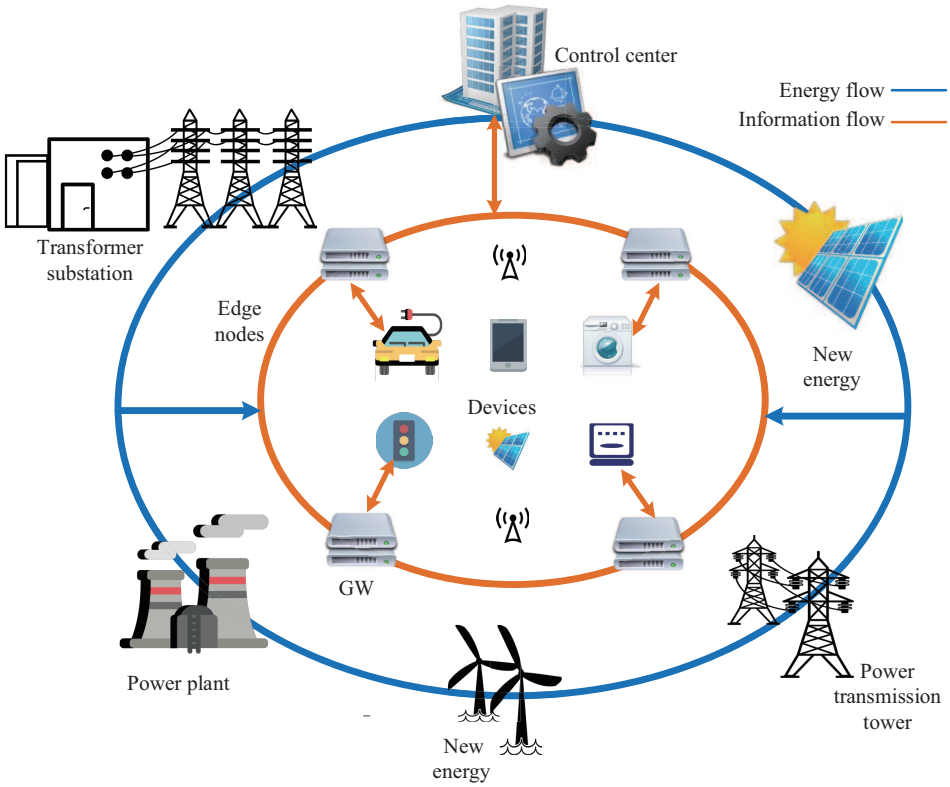

Figure 1 (Color online) The conceptual architecture of smart grid.

To protect the user privacy while still being able to timely provide power generation planning and dynamic pricing, the data aggregation scheme must be efficient and privacy-preserving in smart grid. The users' data are usually encrypted by the homomorphic encryption technique and the encrypted data are sent to the local gateway for data aggregation [5]. Using homomorphic encryption, gateway (GW) can perform aggregation on the encrypted data without decrypting them. In [5], data encrypted by SMs then send encrypted data to the local GW, which aggregates the encrypted data and then uploads the aggregated encrypted data to CC. Only CC can decrypt the aggregated residential users' data. This can guarantee the privacy of individual residential user.

However, the design of existing privacy-preserving data aggregation schemes in the smart grid system is not ideal, because it subjects to the following challenges. First, nearly real-time communication system is crucial to smart grid. Any delay could result in the loss of consumers and power providers [6], thus ensuring the efficiency of communication is very important. Second, data source authentication is necessary [7], to prevent an adversary fraudulently forge the messages of an authorized user. In addition, the ability to detect any data manipulation during transmission cannot be ignored. Some existing studies attempt to solve these challenges [8–10], however, their solutions are limited in terms of efficiency. Hence, we propose a high efficiency privacy-preserving aggregation scheme, which can resist various attacks (e.g., injection, modification, forgery, replay and/or delay (delay seems to be the same thing as replay)). Besides, we implement an enhanced and flexible data aggregation scheme. Smart grid is a highly dynamic system. The differences in time periods and residential areas (RA) will result in large inconsistency in communication cost and computation overhead, which may reduce the system stability. In the meanwhile, SMs may suffer from malfunction, hence some previous aggregation schemes can provide fault-tolerance feature. However, one major drawback of these studies is that they only focus on the reliability of the aggregation process, but not on individual user privacy. Particularly when the number of malfunctioning SMs in a RA is very large, the anonymization of individual data in aggregation cannot be achieved. Hence, we set a flexible threshold for the secret sharing scheme in data aggregation (limiting the number of users in the aggregation), which is adjusted according to the energy consumption information in a given RA and

time period. The SM's historical malfunctioning probability in RA is also taken into account, so as to simultaneously provide fault-tolerance for malfunctioning SMs and preservation of user data privacy.

In sum, we propose EFFECT, an efficient flexible privacy-preserving aggregation scheme with authentication for smart grid, which supports also data integrity verification and source authentication, hence is more suitable for the high-frequency real-time data gathering system. The main contributions of this paper are summarized as follows:

(1) We introduce a flexible threshold for data aggregation based on the secret sharing scheme, to enhance the privacy of individual user. The threshold is adjusted flexibly according to the specific residential area, time period, and the malfunctioning probability of SM in RA.

(2) We realize the data integrity verification and sender authentication in a highly efficient way, each receiver can verify if the received packets do come from the claimed sender and have not been tempered. Besides, our scheme can resist various attacks during transmission (e.g., injection, modification, forgery, replay).

(3) We show the efficiency and security of the proposed scheme with performance evaluation and security analysis.

The rest of this paper is organized as follows. The related work is introduced in Section 2. The preliminaries are given in Section 3. In Section 4, the system model and design goals are stated. In Section 5, the proposed EFFECT scheme is described in detail. The security analysis is given in Section 6. The performance of EFFECT is evaluated in Section 7. At last, the paper is concluded in Section 8.

2 Related work

Data obfuscation [11] and homomorphic encryption [12–14] are widely used in data aggregation in smart grid to prevent the users' privacy-sensitive information from being exposed. However, it is a difficult task to select the obfuscation parameters in practice. In contrast, homomorphic encryption is not only practical but also in [15], Przydatek proposed a scheme to aggregate data securely in smart grid. Homomorphic encryption is used by SMs to protect users' privacy-sensitive information, and GWs can aggregate all users' encrypted data. Although this framework was able to realize efficient data aggregation, the protection of data privacy is still not enough. To solve the data privacy issue caused by frequent data collection, Shi et al. [16] proposed an effective method to aggregate data in time-series that supports high frequency data collection by a set of collectors, and uploads the enciphered data to the GW periodically to ensure the data privacy.

With the development of smart grid, forgery attack and other potential threats have been taken into account. The homomorphic Hash function [17] is adopted for the authentication of CC and SM. Lu et al. [18] presented a privacy-preserving and effective data aggregation method that can achieve the data aggregation in multi-dimension and support sender authentication. Based on Lu's work, Chen et al. [19] proposed a scheme to realize fault tolerance in aggregating data using third authorities. Shi et al. [20] proposed a fault-tolerant scheme named DG-APED to deal with the problems caused by the damaged SMs. Specifically, it aggregates the data in groups, and drops the data group that contains the malfunctioning SMs. However, this approach is not accurate since the error rate of SMs is not a fixed value. Additionally, it also needs to spend extra computation overhead in searching the damaged member. Some other studies [21–23] are proposed to realize efficient and fault tolerant data aggregation, however, the additional overhead is needed. In addition, many approaches have been proposed to reduce the computational and communication overheads for authentication, such as the short signature scheme [24], lightweight data aggregation approach [25] and batch verification scheme [26]. Fouda et al. [27] presented a lightweight scheme for message authentication in smart grid, which provides inspiration for our scheme.

3 Preliminaries

3.1 Paillier cryptosystem

Firstly, the Paillier cryptosystem and some important assumptions will be reviewed, which are the basis of the proposed EFFECT scheme.

Paillier [28] cryptosystem uses an asymmetric encryption algorithm, which can realize the homomorphic properties more efficiently than other homomorphic encryption algorithms. It is widely used in many privacy-preserving applications because of its additive homomorphism properties. It includes three algorithms for key generation, encryption, and decryption, respectively.

(1) Key generation. Given the security parameters κ, choose two large prime numbers p, q, where $|p| = |q|$, and compute $\lambda = \mathrm{lcm}(p-1, q-1)$. Define a function $L(u) = \frac{u-1}{n}$, where $n = pq$. After choosing a generator $g \in Z^*_{N^2}$, $\mu = (L(g^\lambda \bmod n^2))^{-1} \bmod n$ is calculated. Then, the public key is $\mathrm{pk} = (n, g)$, and the private key is $\mathrm{sk} = (\lambda, \mu)$.

(2) Encryption. Given a message $m \in Z_N$, selects a random number $r \in Z^*_N$, then the ciphertext can be calculated as $C = E(m) = g^m \cdot r^n \bmod n^2$.

(3) Decryption. Given a ciphertext, the original message can be recovered with the secret key $m = D(c) = L(c^{\lambda \bmod n^2}) \cdot \mu \bmod n$. In [28], Paillier cryptosystem has been proved to be provably secure against the chosen plaintext attack.

3.2 Secret sharing scheme

Secret sharing scheme [29] is a typical secret protection scheme, which splits a secret into pieces distributed in different users. Only if the number of known secret pieces reaches a given threshold, can the whole secret be retrieved.

The secret is split by a polynomial:

$$G(x) = \theta + a_1 x + a_2 x + \cdots + a_{k-1} x^{k-1},$$

where θ is the shared secret, and k is the threshold. $(x_i, G(x_i))$ is the corresponding share. According to the Lagrange interpolation polynomial, we calculate the secret by

$$l_j(x) := \prod_{i=1, i\neq j}^{k} \frac{x - x_i}{x_j - x_i}. \tag{1}$$

Then θ can be calculated as

$$\sum_{i=1}^{k} G(x_i) l(x_i) = G(0) = \theta. \tag{2}$$

Remarkably, because of the homomorphic feature of Shamir secret sharing scheme, it can be used in the utility data aggregation.

Definition 1 (Computational Diffie Hellman assumption)**.** This assumption claims that given g^a, g^b, it is computationally hard to compute g^{ab}.

Definition 2 (RSA assumption)**.** Given $x, y \in Z_N$ and $a, b \in Z$ such that $x^a = y^b$, if $\gcd(a, b) = 1$, it is hard to find out an $x' \in Z_N$ such that $(x')^a = y$.

Definition 3 (Subgroup decision problem)**.** Given a tuple (e, G, G_T, n, h), where the element h is randomly drawn from either G or subgroup G_q, it is difficult to decide whether or not $h \in G_q$.

4 System model and design goals

4.1 System model

The system model of EFFECT shown in Figure 2 consists of CC, trusted certification authority (TCA), edge devices like GW and users in the RA. In the meanwhile, three kinds of information flow are involved

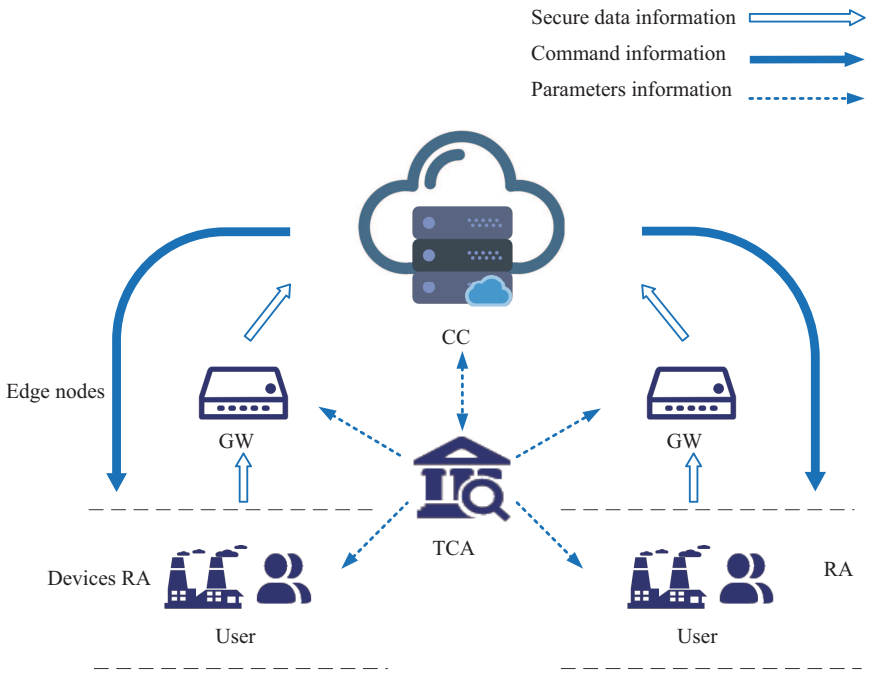

Figure 2 (Color online) System model of EFFECT scheme.

in the system. Parameters information contains all the parameters involved in the operations. Configured with these parameters, users can encrypt their power-usage data as the secure utility information transmitted from SM to CC. From the utility information, CC can analyze the power consumption trend before planning power generation. Additionally, CC issues dynamic commands to users. The entities are described in detail as follows.

User. Including home users and industrial users. The power consumption data are collected by SMs, and uploaded to the CC. To protect user's privacy, before being sent to CC, the data are enciphered.

GW. GW is responsible for aggregating the enciphered data from SMs, running the homomorphic algorithm over the aggregated data, and sending the ready-processed data to CC. To enhance the system efficiency, the nearest GW available in a RA will be selected by CC.

TCA. TCA sets up the whole system. It generates the keys and system parameters for the system.

CC. CC can obtain the trend of power consumption and conduct power generation planning as well as dynamic pricing, by analyzing the aggregated real-time utility data from GWs. To improve the efficiency, each region will set up its own CCs.

4.2 EFFECT scheme procedure

The EFFECT scheme includes the following three steps.

Step1. User data encryption. (1) Before user's utility data is sent out (either periodically or per CC's request), the TCA initializes the system by generating secret keys and the related parameters for respective entities, and the CCs set flexible aggregation thresholds based on different RAs, time periods, as well as the malfunction probability of the SMs. (2) GW and SM are able to authenticate mutually and generate authentication parameters for the following steps. (3) SM encrypts the current electricity usage data by Paillier encryption, and sends the encrypted packet to the local GW in RA.

Step2. Efficient data authentication and flexible data aggregation. (1) When RA users' reports have arrived at the local GW, the packets will be authenticated and verified in a highly efficient way. (2) The valid usage data will be aggregated by GW, and then GW checks whether the flexible threshold can be satisfied. If the number of usage data in the aggregation meets the threshold, GW forwards the encrypted aggregation data to the CC. Otherwise, GW aborts the aggregation, and resends data request to users.

Step3. Secure report reading. (1) Upon receiving the data packets from GW, CC authenticates the packet source and verifies the validity of the data, to ensure that it is indeed coming from the claimed GW and the data integrity. (2) CC decrypts the ciphetext and recover the aggregated usage data of RA. Then, it can obtain the trend of power consumption based on the nearly real-time aggregation data without disclosing the individual user data, and consequently make the generation schedule and dynamic pricing accordingly.

4.3 Adversarial model

In our attack model, the CC is considered to be trustful. The users $(U_1, U_2, \ldots, U_n)$ in RA and the SMs installed on the user side are considered to be honest. However, the GW may be semi-trusted (honest-but-curious). In other words, it does not manipulate user's data, but maybe try to snoop the user's privacy-sensitive data passing through. Due to the large number of GWs deployed, the leak of user privacy could be disastrous.

Besides, both external and internal threats may exist in smart grid. For instance, the adversary $\mathcal{A}$ in the system could eavesdrop the communication data or hack into the servers in CC and GW to steal user's privacy information. By intercepting the message, the adversary could also forge the identity of authorized users to inject false data into the system. What's more, the adversary $\mathcal{A}$ could as well make active attacks to compromise the data integrity in smart grid.

4.4 Design goals

To solve the issues mentioned above, the design goals include five aspects.

(1) Privacy-preservation. user's data should not be available to the unauthorized users. The adversary, CC or GW cannot access an individual user's data even in case that they conclude with each other.

(2) Enhanced flexible aggregation. the threshold of aggregation will be adjusted according to the energy consumption information of each particular residential area, time period, and the potential malfunction rate of SMs. Data cannot be successfully aggregated if the number of users is less than the threshold, thus preventing small cluster users privacy being disclosed during data aggregation. This can also realize fault-tolerance for malfunctioning SMs.

(3) Fault-tolerance. setting threshold according to SMs' malfunctioning probability in RA allows GW still being able to aggregate data from normal-functioning SMs even though there are malfunctioning ones.

(4) Data source authentication and data integrity verification. the user's communication packet and the GW's packet need be authenticated, to validate that they are indeed sent by the legal residential user and target gateway, respectively. The valid communications between entities cannot be modified during the transmission, and can resist different attacks during the transmission (e.g., injection, modify, forge, reply and/or delay).

(5) Computation efficiency. To collect the power consumption data of thousands of users in smart grid efficiently, the high computation efficiency of the proposed EFFECT scheme is expected.

5 EFFECT

5.1 System initialization

When CC sends a data collection request to residential users, TCA and CC initialize the system by generating the related parameters for the operations. The initialization process is shown in Figure 3, the detailed steps are as follows.

Step1. Given the security parameters κ, TCA chooses two large prime numbers p_0, q_0, where $|p_0| = |q_0| = \kappa$, and generates keys by running $\mathrm{Gen}_{\mathrm{Pai}}(\kappa)$ for the Paillier cryptosystem, which public key is $\mathrm{pk}_{\mathrm{Pai}} : (N = p_0 q_0, g)$, and the corresponding private key is $\mathrm{sk}_{\mathrm{Pai}} : (\lambda, \mu)$.

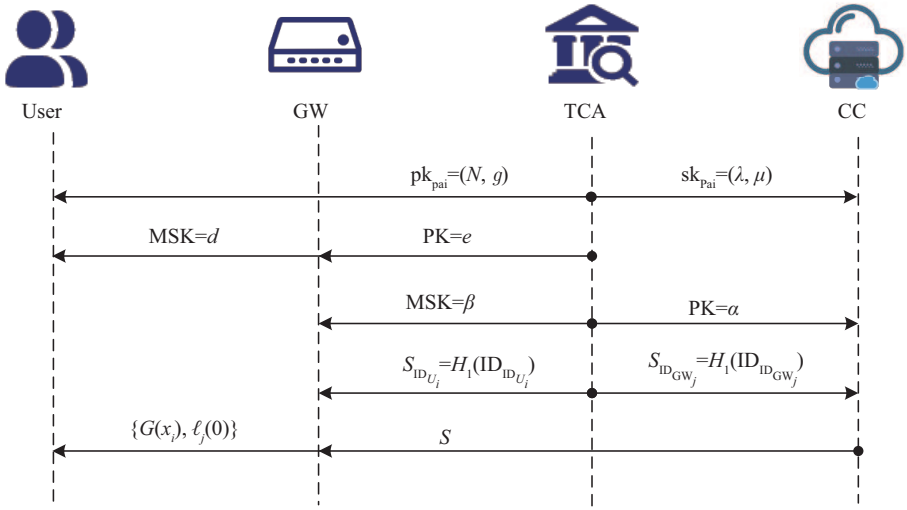

Figure 3 (Color online) Initialization process.

Step2. TCA randomly chooses two large primes p_1, q_1 by running $\mathrm{Gen}_{\mathrm{RSA}_1}$ then the RSA modulus $N_1 = p_1 q_1$ is calculated. Next, a prime number e is chosen, and $d \equiv e^{-1} \bmod \varphi(N_1)$ is computed. d is the master private key of user to generate the signature in the user report generation phase, and then GW uses the public key e to verify the validity of the signature. Similarly, TCA runs $\mathrm{Gen}_{\mathrm{RSA}_2}$, and generates the RSA modulus $N_2 = p_2 q_2$ after choosing two large primes p_2, q_2. Then, a prime number α is chosen, and $\beta \equiv \alpha^{-1} \bmod \varphi(N_2)$ is calculated, β is used as GW's master private key to generate signature in aggregation phase, and the public key α is held by CC.

Step3. Then TCA defines four one-way Hash functions: $H : \{0,1\}^* \to Z_N^*$, $H_1 : \{0,1\}^* \to G$, $H_2 : \{0,1\}^* \to Z_N^*$, $H_3 : \{0,1\}^* \to Z_N^*$, and assumes that they are random oracle mappings. Given the use $U_i's$ identity ID_{U_i}, $\mathrm{ID}_{U_i} \in \{0,1\}^*$, users compute $U_i's$ secret identity $S_{\mathrm{ID}_{U_i}} = H_1(\mathrm{ID}_{U_i})$, then TCA computes the GW's secret identity based on the GW's identity $\mathrm{ID}_{\mathrm{GW}_j} \in \{0,1\}^*$, $S_{\mathrm{ID}_{\mathrm{GW}_j}} = H_1(\mathrm{ID}_{\mathrm{GW}_j})$, residential user's secret identity is held by GW, and GW's secret identity is held by CC.

Step4. TCA publishes the system parameters as $\mathrm{pubs} = \{N, g, e, \alpha, H, H_1, H_2, H_3\}$, keeps the master private keys (λ, μ, d, β) in secret.

Step5. CC sets the threshold of aggregation according to the actual electricity consumption and the SMs' malfunctioning probability. Then, it chooses a random number $s \in Z_N$, calculate $s + s_0 = 0 \bmod N$ and forward $s \in Z_N$ to the target GW. Next, the secret parameter s_0 is divided into n (the number of users in residential area) pieces, when messages are held by k $(k \leqslant n)$ or more participants, s_0 can be calculated. $G(x) = s_0 + a_1 x + a_2 x + \cdots + a_{k-1} x^{k-1}$. CC sends $\{G(x_i), l_j(0)\}$ to U_i in RA via security channel, which $l_j(x) := \prod_{i=1, i \neq j}^{k} \frac{x - x_i}{x_j - x_i}$.

5.2 Data aggregation request

In order to get more precise power consumption trend, the nearly real-time residential user's electricity usage data are collected every η minutes (suppose $\eta = 15$ min), at timestamp $t \in \mathbb{T}$, the CC sends a usage data collection request to users, and performs as follows.

Step1. The CC first selects a random number $l \in Z_N$ and calculates $g' = g^l$. Then it sends g' to user $U_i \in U$, and sends the usage data collection request to GWs in the target RA.

Step2. The user $U_i \in U$ in RA selects a random number $a_i \in Z_N$, and calculates $S_{\mathrm{ID}_{U_i}}^{a_i}$ based on U_i' secret identity, it then sends the request packet to the target GW in RA. And the request is encrypted with the key $\mathrm{Pub}_{\mathrm{User} \to \mathrm{GW}}$, which is pre-generated and used for data transmission between the user and GW.

$$\mathrm{User}_i \to \mathrm{GW}_j : \left\{ S_{\mathrm{ID}_{U_i}}^{a_i} || S_{\mathrm{ID}_{U_i}} \right\}_{\{\mathrm{enc}\}\mathrm{Pub}_{\mathrm{User} \to \mathrm{GW}}}. \tag{3}$$

Step3. After receiving the packets from users, GW_j decrypts it and selects a random number $b_i \in Z_N$, then it computes $S_{\mathrm{ID}_{U_i}}^{b_j}$ and sends it to user U_i.

$$\mathrm{GW}_j \to \mathrm{User}_i : \left\{ S_{\mathrm{ID}_{U_i}}^{a_i} || S_{\mathrm{ID}_{U_i}} || S_{\mathrm{ID}_{U_i}}^{b_j} \right\}_{\{\mathrm{enc}\}\mathrm{Pub}_{\mathrm{GW}\to\mathrm{User}}}. \tag{4}$$

Step4. When user U_i receives the response packet, U_i recovers $S_{\mathrm{ID}_{U_i}}^{a_i}$, if $S_{\mathrm{ID}_{U_i}}^{a_i}$ is correct, then it performs the following phases.

5.3 User report generation (SM)

Each user U_i in RA collects data $(d_1, d_2, \ldots, d_n)$ by SM_j at timestamp $t \in \mathbb{T}$, and performs the following steps to generate U_i' s report.

Step1. U_i first computes the Hash value $H(t)$, then chooses a random number $r_i \in Z_n^*$, and calculates the ciphertext after Paillier encryption:

$$C_i = (g')^{d_i} \cdot H(t)^{G(x_i)\cdot l_j(0)} \cdot r_i^N \bmod N^2 = g^{l\cdot d_i} \cdot H(t)^{G(x_i)\cdot l_j(0)} \cdot r_i^N \bmod N^2. \tag{5}$$

Step2. U_i uses the master private key d to generate a signature δ_i with $S_{\mathrm{ID}_{U_i}}^{b_j}$, $\delta_i = H_2(C_i || S_{\mathrm{ID}_{U_i}}^{b_j a_i})^d \bmod N_2$.

Step3. The signature and encrypted usage data are formed as $(C_i||\delta_i||S_{\mathrm{ID}_{U_i}}^{a_i}||l_j(0)||G(x_i))$. Then, U_i sends it to the local GW in the RA (residential area).

5.4 Privacy-preserving report aggregation (GW)

After GW_j receives users' reports $(C_i||\delta_i||S_{\mathrm{ID}_{U_i}}^{a_i}||l_j(0)||G(x_i))$, $i = 1, 2, \ldots, n$, it performs the following steps for privacy-preserving report aggregation.

Step1. GW receives n' $(k \leqslant n' \leqslant n)$ reports from the RA, GW verifies all signatures by checking if $\delta_i^e = H_2(C_i||S_{\mathrm{ID}_{U_i}}^{a_i b_j}) \bmod N^2$ holds with the public key e. In order to make the verification efficiently, the GW adopts the batch verification by checking

$$\begin{aligned}
\prod_{i=1}^{n'} \delta_i^e &= \prod_{i=1}^{n'} H_2\big(C_i||S_{\mathrm{ID}_{U_i}}^{a_i b_j}\big) \bmod N_2, \\
\prod_{i=1}^{n'} \delta_i^e &= \prod_{i=1}^{n'} H_2\big(C_i||S_{\mathrm{ID}_{U_i}}^{b_j a_i}\big)^{d\cdot e} \bmod N_2 \\
&= \prod_{i=1}^{n'} H_2\big(C_i||S_{\mathrm{ID}_{U_i}}^{a_i b_j}\big) \bmod N_2.
\end{aligned} \tag{6}$$

Step2. GW computes the enhanced aggregation of encrypted data:

$$\begin{aligned}
C_a &= \prod_{i=1}^{n'} C_i \cdot H(t)^s \\
&= \prod_{i=1}^{n'} g^{l\cdot d_i} \cdot H(t)^{G(x_i)\cdot l_j(0)} \cdot r_i^N \bmod N^2 \quad (k \leqslant n' \leqslant n) \\
&= g^{l\cdot\sum_{i=1}^{n'} d_i} \cdot H(t)^{\sum_{i=1}^{n'} G(x_i) l_j(0)+s} \cdot (r_1 \cdots r_{n'})^N \bmod N^2.
\end{aligned} \tag{7}$$

Step3. GW_j verifies whether the aggregation is valid, by checking if $\sum_{i=1}^{n'} G(x_i)\cdot l_j(0) + s = 0 \bmod N$ holds based on Lagrange, and then uses the master secret key β to generate a signature σ_j, $\sigma_j = (H_3(C_a||S_{\mathrm{ID}_{\mathrm{GW}_j}}))^\beta \bmod N^2$.

Step4. GW_j reports the packet $(C_a||\sigma_j||\mathrm{TS})$ to CC, where TS is the current timestamp, used to defeat the replay attack.

5.5 Secure report reading (CC)

Upon receiving $(C_a||\sigma_j||\text{TS})$, the CC first verifies the validity by checking the timestamp TS, then checks the source and integrity of the aggregated data with the public key α as follows:

$$\sigma_j^{\alpha} = H_3(C_a||S_{\text{ID}_{\text{GW}_j}})^{\beta\cdot\alpha} \bmod N_2 = H_3(C_a||S_{\text{ID}_{\text{GW}_j}}) \bmod N_2. \tag{8}$$

Then CC performs the following steps to decrypt the aggregation data.

Step1.

$$C_a \cdot g^{-l} = \prod_{i=1}^{n'} C_i \cdot g^{-l} = g^{\sum_{i=1}^{n'} d_i} \cdot (r_1 \cdots r_{n'})^N \bmod N^2. \tag{9}$$

Step2. The master key $\text{sk}_{\text{pai}} = (\lambda, \mu)$ is used to recover the aggregated data in RA_z based on Paillier decryption [28] as

$$D_z = \text{Dec}(C_a) = L(C_a^{\lambda} \bmod N^2) \cdot \mu \bmod N = \frac{L(C_a^{\lambda} \bmod N^2)}{L(g^{\lambda} \bmod N^2)} \bmod N = \sum_{i=1}^{n'} d_i. \tag{10}$$

6 Security analysis

The security properties of the EFFECT scheme is analyzed in the following. Especially, the analysis will concentrate on how EFFECT scheme makes the smart grid resilient against various passive and active attacks, including the prevention of individual information leakage, as well as data authentication and integrity verification, which can defend against active attacks like data modification of and injection.

(1) The individual user's report is secured in the proposed EFFECT scheme. In the proposed EFFECT scheme, user U_i''s data in RA, $(d_1, d_2, \ldots, d_n)$ are encrypted as $C_i = (g')^{d_i} \cdot H(t)^{G(x_i)\cdot l_j(0)} \cdot r_i^N \bmod N^2$, which are still valid ciphertexts of Paillier cryptosystem. Since Paillier cryptosystem is provably secure against the chosen plaintext attack based on the Definition 3 (subgroup decision assumption), the data $(d_1, d_2, \ldots, d_n)$ in C_i are also and privacy-preserving and secure. Hence, even if the adversary $\mathcal{A}$ intercepts C_i, he cannot none the less recover the individual data. After gathering all data $(C_1, C_2, \ldots, C_n)$ from the RA, the GW also cannot recover the plaintext of individual user's data. It only can get $C_a = \prod_{i=1}^{n'} C_i$ to execute data aggregation. Hence, even if an adversary $\mathcal{A}$ invades in the the database of GW, he cannot obtain the individual data $(d_1, d_2, \ldots, d_n)$. Finally, after receiving $C_a = \prod_{i=1}^{n'} C_i$ from GW in RA, the CC recovers C_a as $D_a = \sum_{i=1}^{n'} d_i$. Besides, we add a threshold for aggregation under the secret sharing scheme to ensure the security of individual data. This threshold is adjusted according to the energy consumption information of each particular residential area and the time period. When the number of valid data in aggregation is below the threshold, the aggregation is invalid, preventing adversary from inferring the individual information in a small cluster of users. Since D_a is an enhanced aggregation result, even if an adversary $\mathcal{A}$ steals the data, he cannot identify the user U_i''s data $(d_1, d_2, \ldots, d_n)$. Hence, from the three aspects mentioned, the user's data is privacy-preserving in the proposed EFFECT scheme.

(2) Data integrity and source can be guaranteed in EFFECT. In the proposed EFFECT scheme, and report integrity verification and the sender authentication are both implemented. In the meanwhile, we prove that our scheme is secure against active attacks such as message forgery and reply attacks.

• **Source authentication.** The communications from user to GW are considered firstly. When GW_j receives a report $(C_i||\delta_i||S_{\text{ID}_{U_i}}^{a_i}||l_j(0)||G(x_i))$, $\delta_i = H_2(C_i||S_{\text{ID}_{U_i}}^{b_j a_i})^d \bmod N^2$ from U_i, GW_j verifies if $\delta_i^e = H_2(C_i||S_{\text{ID}_{U_i}}^{b_j a_i}) \bmod N^2$ holds. In the data aggregation request phase, we should note that $S_{\text{ID}_{U_i}}^{a_i}$ has been encrypted with U_i''s public key, which means that only GW_j can recover $S_{\text{ID}_{U_i}}^{a_i}$ with the corresponding private key. Therefore, when U_i receives the correct $S_{\text{ID}_{U_i}}^{a_i}$ from GW_j, U_i can ensure that its counterpart is GW_j. Similarly, because $S_{\text{ID}_{U_i}}^{b_j}$ is encrypted with U_i''s public key, GW_j can also authenticate U_i. Besides,

our authentication is under Definition 1 (CDH assumption). Given $S_{\mathrm{ID}_{U_i}}^{a_i}$, $S_{\mathrm{ID}_{U_i}}^{b_j}$, it is computationally hard to compute $S_{\mathrm{ID}_{U_i}}^{a_i b_j}$. In addition, H_2 is a random oracle mapping $H_2 : \{0,1\}^* \to Z_N^*$ (The collision intractability of the Hash function means that collisions in the Hash values correspond to equal messages. δ_i is calculated by RSA encryption with the public key e, only the GW_j can recover the message with the corresponding master private key. Additionally, based on Definition 2 (RSA assumption) with large public exponents in the random oracle model (which is provably secure as in work [30], even if an adversary $\mathcal{A}$ holds e, it is hard to find out an $x' \in Z_N$ such that $(x')^e = \delta$. Therefore when GW_j receives the correct $S_{\mathrm{ID}_{U_i}}^{a_i b_j}$, GW_j can ensure that its counterpart is U_i in the target RA. Therefore, our scheme can provide effective mutual authentication between GW and user. Similarly, under the RSA assumption, the source authentication of the report from GW_j to CC can also be achieved. Hence, our scheme can ensure that the received messages indeed come from the claimed senders, and prevent message forgery attacks.

• **Data integrity.** The communications from user to GW are considered firstly. Upon receiving $(C_i||\delta_i||S_{\mathrm{ID}_{U_i}}^{a_i}||l_j(0)||G(x_i))$, GW_j''s public key e is used to verify if $\delta_i = H_2(C_i||S_{\mathrm{ID}_{U_i}}^{a_i b_j})^d \bmod N^2$ holds. Only users can generate valid signatures with $S_{\mathrm{ID}_{U_i}}^{b_j}$, which can only be verified with U_i''s private key. This means that an external adversary $\mathcal{A}$ cannot alter the original data encrypted and reported by U_i. If the report is tempered by $\mathcal{A}$ during the transmission, GW_j is able to find that the report has been modified through batch verification. In [31], it has been proved that exponentiation batch verification can provide data verification efficiently. Due to the same reason, the data integrity of thereport from GW_j to CC can be ensured as well.

In addition, U_i encrypts data by Paillier encryption, $C_i = (g')^{d_i} \cdot H(t)^{G(x_i)\cdot l_j(0)} \cdot r_i^N \bmod N^2$. $H(t)$ is calculated according to the current timestamp t, only if all users in RA generate reports at timestamp t, the reports could be aggregated successfully. Besides, GW_j sends $(C_a||\sigma_j||\mathrm{TS})$ to CC, where TS is the timestamp of the data aggregation process. After receiving the report, CC checks whether the report is valid using TS. Only the valid report corresponding to the current timestamp or TS can pass the verification, hence our proposed scheme can resist the message replay attack.

(3) Secure and reliable fault tolerance in EFFECT. To realize fault tolerance of SM's malfunction, a flexible threshold is set according to the historical malfunctioning probability of SMs in RA, which can ensure that the data aggregation is valid while there are malfunctioning SMs in the system. Only if the number of valid data sent by SMs exceeds the threshold, can the aggregated data be recovered successfully by CC using g and the secret key $\mathrm{sk}_{\mathrm{Pai}} = (\lambda, \mu)$. Since $\mathrm{sk}_{\mathrm{Pai}}$ is kept in secret by CC, no one else can recover the sum of valid usage data. Besides, in the previous section, the authentication between SM and CC is proved to be secure against forgery attack, the adversary cannot launch forgery or other attacks through malfunctioning SM to the system, which is of great significance in the smart grid.

7 Performance evaluation

The performance of EFFECT is evaluated in this section, including the computation complexity of SM, GW, and CC, and the communication overhead.

7.1 Computation complexity

Compared with exponentiation operation and multiplication operation, Hash operation can be regarded as negligible[1)]. In the EFFECT scheme, we assume that RA contains n users and all user data can be collected successfully. The computations in the data aggregation process mainly include three phases, data encryption, batch verification and aggregation, authentication and decryption. We first look into the generation of the report $(C_i||\delta_i||S_{\mathrm{ID}_{U_i}}^{a_i}||l_j(0)||G(x_i))$. The generation of C_i requires three multiplication operations in $Z_{N^2}^*$ and three exponentiation operations in Z_N^*; the generation of δ_i requires three times multiplication operations in $Z_{N^2}^*$ and four exponentiation operations in Z_N^*. After GW_j receives the ciphertext from n users, GW_j first authenticates the source and integrity of the received data by batch

1) Crypto++ 6.0.0 Benchmarks. 2017. https://www.cryptopp.com/benchmarks.html.

Table 1 Computation complexity

Entity name	Involving operations	Computation complexity
SM	(1) User's electricity usage data collection (2) Data encryption (3) Signature δ_i generation	$4 \times C_e + 3 \times C_m$
GW	(1) User's data integrity verification and sender authentication (2) User's data aggregation (3) Signature σ_j generation	$(2n+1) \times C_m + 3 \times C_e$
CC	(1) Aggregated data integrity verification and sender authentication (2) Data decryption	$3 \times C_e + 2 \times C_m$

Table 2 Comparison of computation complexity in authentication phrase

Scheme	EPPA	EPPDA	Shen's scheme	EFFECT
Complexity	$(n+1) \times C_p$	$2n \times C_p$	$(n+2) \times C_p$	$n \times C_m + C_e$

verification, which includes n multiplication operations and one exponentiation operation in Z_N^*. The user's data aggregation computation complexity is $n+1$ multiplication operations and one exponentiation operation. Besides, generating the signature σ_j needs one exponentiation operation in Z_N^*. Then GW_j sends the report $(C_a||\sigma_j||\mathrm{TS})$ to CC. Upon receiving the report from GW_j, the CC authenticates the sender identity and verifies the integrity of report, which needs one exponentiation operation in $Z_{N^2}^*$. After successful authentication, CC decrypts the aggregated report, the Paillier decryption includes one exponentiation operation and one multiplication operation in $Z_{N^2}^*$, hence the total decryption computation complexity includes two exponentiation operations and two multiplication operations in $Z_{N^2}^*$. We denote the computational cost of an exponentiation operation and a multiplication operation, by C_e, C_m, respectively. The computation complexities of the major entities in the system are as show in Table 1. It can be seen that the computation complexities in SM and CC is less then that in GW, so the efficiency of data aggregation in smart grid can be increased by making full use of computing power of local GW.

To prove the efficiency of the EFFECT scheme, we compare the computation complexity in authentication phase with EPPA [18], EPPDA [32] and Shen's scheme [26]. Both EPPA and Shen's scheme use BLS (proposed by Boneh et al.) short signature [33], while EPPDA uses identity-based signature, to realize batch verification. The comparison is shown in Table 2, we suppose the GW receives n users' report and denote the computational complexity of a pairing operation by C_p.

Because we adopt the batch verification in RSA, the verification relation is modular exponentiation, which has been proved to be very efficient compared [30] with the pairing operation. That is to say, our scheme is more suitable for the system with high real-time requirements such as smart grid.

We further analyze the computational cost when there exists malfunctioning SMs in RA, compared with DG-APED [20] and PDAFT [19] schemes, which can also realize fault tolerance of malfunctioning SMs. Then, we observe the relationship between the computational cost and malfunctioning SMs. It is assumed that the highest rate of malfunctioning SMs in RA is seventy percent. We conduct the experiments with MIRALC [34] and LiDIA [35] libraries running on a 3.2 GHz-processor and 8 GB-memory PC. As shown in Figure 4, our scheme has a significant advantage in computional cost compared with previous fault-tolerance schemes when the number of SM varies from 50 to 250.

7.2 Communication overhead

The communication overhead in the proposed EFFECT scheme can be mainly divided into two parts: one part is from user's devices to GW in RA communication, and the other part is from GW to CC communication, abbreviated as SM-to-GW and GW-to-CC, respectively. In the first phase, the data collected by SM is used to generate the report $(C_i||\delta_i||S_{\mathrm{ID}_{U_i}}^{a_i}||l_j(0)||G(x_i))$ and is sent to the target GW. The size of the user's report is $S_z = |C_i| + |\delta_i| + |S_{\mathrm{ID}_{U_i}}^{a_i}| + |l_j(0)| + |G(x_i)|$. Considering that some SMs could be malfunctioned, for a residential area with n users, the maximum communication overhead in

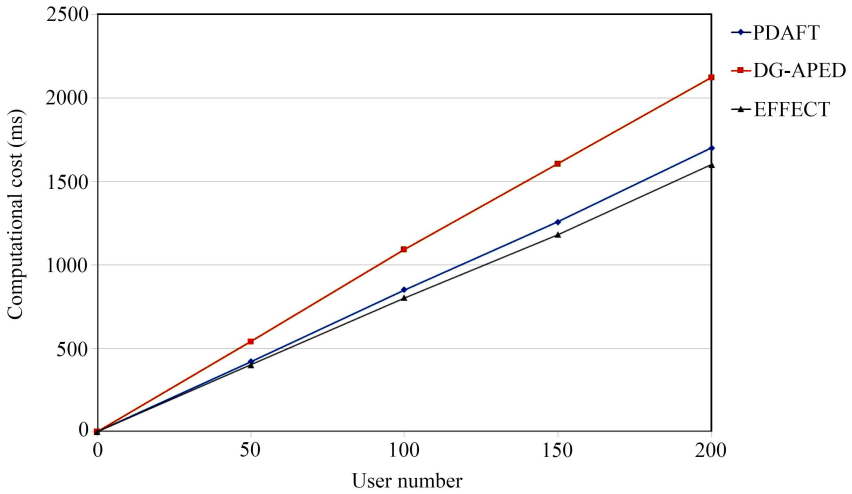

Figure 4 (Color online) Computational cost considering fault tolerance.

this phase is $S_{1\max} = n \times S_z = n \times (|C_i| + |\delta_i| + |S^{a_i}_{\mathrm{ID}_{U_i}}| + |l_j(0)| + |G(x_i)|)$. In the next phase, the GW collects the reports from n' $(k \leqslant n' \leqslant n)$ users, and delivers the aggregated report to the CC, the report is generated as $(C_a||\sigma_j||\mathrm{TS})$, so the size of this report is $S_{2\max} = |C_a| + |\sigma_j| + |\mathrm{TS}|$.

Next, we compare the communication overheads in the SM-to-GW and GW-to-CC phases in our scheme with EPPA, EPPDA, which also use the Paillier encryption algorithm to encrypt user data. Since the EPPA has multidimensional usage data, for the consistency of the variables, we consider the usage data as one-dimensional data.

(1) Communication overhead in EPPA.

SM-to-GW:

$$n \times (|C_i| + |\delta_i| + |\mathrm{RA}| + |U_i| + |\mathrm{TS}|). \tag{11}$$

GW-to-CC (OA):

$$|C_a| + |\mathrm{RA}| + |U_i| + |\mathrm{TS}| + |\sigma_j|. \tag{12}$$

(2) Communication overhead in EPPDA.

SM-to-GW(BG):

$$n \times (|C_i| + |\mathrm{ID_{GW}}| + |\mathrm{ID}_{U_i}| + |\mathrm{TS}| + |\delta_i|). \tag{13}$$

GW(BG)-to-CC:

$$|C_a| + |\mathrm{ID_{GW}}| + |\mathrm{ID_{CC}}| + |\mathrm{TS}| + |\sigma_j|. \tag{14}$$

And we chooses the same length parameter as in [18, 32]. Suppose that each user generates a 2048-bit ciphertext C_i and chooses 160-bit Z_N^*, 160-bit G. We set $|S^{a_i}_{\mathrm{ID}_{U_i}}| + |l_j(0)| + |G(x_i)|$, $|\mathrm{RA}| + |U_i| + |\mathrm{TS}|$ and $|\mathrm{ID_{GW}}| + |\mathrm{ID}_{U_i}| + |\mathrm{TS}|$ as 100-bit length as in [18]. In Figure 5, we plot the communication overhead in EPPA, EPPDA and our scheme versus the user number n and the data size. (In Figure 5(b) we set $n = 400$).

It is shown that our scheme realizes the sender authentication and data integrity verification in an efficient way, and the flexible threshold of aggregation does not bring too much communication overhead.

8 Conclusion

In this paper, we propose the EFFECT scheme, an efficient flexible privacy-preserving aggregation and authentication Scheme in smart grid. We realize an enhanced aggregation based on secret sharing, which can guarantee individual privacy and support fault-tolerance in a flexible way. Compared with the previous schemes of this kind, EFFECT provides a high level of controllability in data collection and processing phase. Additionally, EFFECT realizes data integrity verification and source authentication

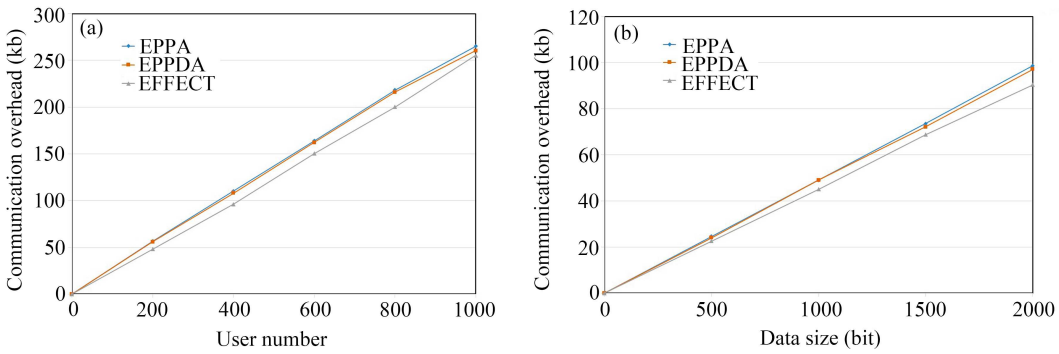

Figure 5 (Color online) Comparison of communication overhead with (a) different user numbers, and (b) different data size.

efficiently, which is highly desired for a real-time high-frequency data collection system. We provide security analysis to demonstrate the level of security achieved in our scheme. Finally, by comparing the computation complexity and the communication overhead with existing schemes, we prove the efficiency of our scheme. In future, we will work on resolving the privacy of multidimensional power use data, and further develop the EFFECT scheme.

Acknowledgements This work was partially supported by Beijing Natural Science Foundation (Grant No. 4182060), and Fundamental Research Funds for the Central Universities (Grant No. 2018ZD06).

References

1 Wang K, Du M, Maharjan S, et al. Strategic honeypot game model for distributed denial of service attacks in the smart grid. IEEE Trans Smart Grid, 2017, 8: 2474–2482
2 Guan Z T, Si G L, Zhang X S, et al. Privacy-preserving and efficient aggregation based on blockchain for power grid communications in smart communities. IEEE Commun Mag, 2018, 56: 82–88
3 Xue K P, Li S H, Hong J N, et al. Two-cloud secure database for numeric-related sql range queries with privacy preserving. IEEE Trans Inf Foren Sec, 2017, 12: 1596–1608
4 Wu J, Dong M X, Ota K, et al. Securing distributed storage for social internet of things using regenerating code and blom key agreement. Peer-to-Peer Netw Appl, 2015, 8: 1133–1142
5 Erkin Z, Troncoso-Pastoriza J, Lagendijk R, et al. Privacy-preserving data aggregation in smart metering systems: an overview. IEEE Signal Proc Mag, 2013, 30: 75–86
6 Yan Y, Qian Y, Sharif H, et al. A survey on smart grid communication infrastructures: motivations, requirements and challenges. IEEE Commun Surv Tut, 2013, 15: 5–20
7 Cho S, Li H, Choi B J. Palda: efficient privacy-preserving authentication for lossless data aggregation in smart grids. In: Proceedings of IEEE International Conference on Smart Grid Communications, 2014. 914–919
8 Guan Z T, Li J, Zhu L H, et al. Toward delay-tolerant flexible data access control for smart grid with renewable energy resources. IEEE Trans Ind Inform, 2017, 13: 3216–3225
9 Zheng J M, Tan Y A, Zhang Q K, et al. Cross-cluster asymmetric group key agreement for wireless sensor networks. Sci China Inf Sci, 2018, 61: 048103
10 Guan Z T, Li J, Wu L F, et al. Achieving efficient and secure data acquisition for cloud-supported internet of things in smart grid. IEEE Int Thing J, 2017, 4: 1934–1944
11 Zhang Z J, Qin Z, Zhu L H, et al. Cost-friendly differential privacy for smart meters: exploiting the dual roles of the noise. IEEE Trans Smart Grid, 2016, 8: 619–626
12 Li S H, Xue K P, Yang Q Y, et al. PPMA: privacy-preserving multisubset data aggregation in smart grid. IEEE Trans Ind Inf, 2018, 14: 462–471
13 Li S H, Zhang X, Xue K P, et al. Privacy-preserving prepayment based power request and trading in smart grid. China Commun, 2018, 15: 14–27
14 Xiao Y, Tan Y A, Sun Z Z, et al. A fault-tolerant and energy-efficient continuous data protection system. J Amb Intel Hum Comp, 2018. doi: 10.1007/s12652-018-0726-2
15 Przydatek B, Song D, Perrig A. Sia: secure information aggregation in sensor networks. In: Proceedings of the 1st International Conference on Embedded Networked Sensor Systems, 2003. 255–265
16 Shi E, Chan T H, Rieffel E, et al. Privacy-preserving aggregation of time-series data. In: Proceedings of the 18th Annual Network and Distributed System Security Conference, 2011
17 Kim Y S, Heo J. Device authentication protocol for smart grid systems using homomorphic Hash. J Commun Netw, 2012, 14: 606–613
18 Lu R X, Liang X H, Li X, et al. EPPA: an efficient and privacy-preserving aggregation scheme for secure smart grid communications. IEEE Trans Paral Distrib Syst, 2012, 23: 1621–1631

19 Chen L, Lu R X, Cao Z F. Pdaft: a privacy-preserving data aggregation scheme with fault tolerance for smart grid communications. Peer Peer Netw Appl, 2015, 8: 1122–1132

20 Shi Z G, Sun R X, Lu R X, et al. Diverse grouping-based aggregation protocol with error detection for smart grid communications. IEEE Trans Smart Grid, 2015, 6: 2856–2868

21 Wu J, Dong M X, Ota K, et al. Big data analysis-based secure cluster management for optimized control plane in software-defined networks. IEEE Trans Netw Serv Manage, 2018, 15: 27–38

22 Zhang X S, Tan Y A, Xue Y, et al. Cryptographic key protection against FROST for mobile devices. Cluster Comput, 2017, 20: 2393–2402

23 Gao S, Ma X D, Zhu J M, et al. APRS: a privacy-preserving location-aware recommender system based on differentially private histogram. Sci China Inf Sci, 2017, 60: 119103

24 Mustafa M A, Zhang N, Kalogridis G, et al. Dep2sa: a decentralized efficient privacy-preserving and selective aggregation scheme in advanced metering infrastructure. IEEE Access, 2016, 3: 2828–2846

25 Wang T, Zeng J D, Bhuiyan M Z A, et al. Trajectory privacy preservation based on a fog structure for cloud location services. IEEE Access, 2017, 5: 7692–7701

26 Shen H, Zhang M W, Shen J. Efficient privacy-preserving cube-data aggregation scheme for smart grids. IEEE Trans Inf Foren Secur, 2017, 12: 1369–1381

27 Fouda M M, Fadlullah Z M, Kato N, et al. A lightweight message authentication scheme for smart grid communications. IEEE Trans Smart Grid, 2011, 2: 675–685

28 Paillier P. Public-key cryptosystems based on composite degree residuosity classes. In: Proceedings of International Conference on Theory and Application of Cryptographic Techniques, 1999. 223–238

29 Blakley G R. Safeguarding cryptographic keys. In: Proceeding of International Workshop on Managing Requirements Knowledge, 1979. 313–317

30 Yu Y, Xue L, Au M H, et al. Cloud data integrity checking with an identity-based auditing mechanism from RSA. Future Gener Comput Syst, 2016, 62: 85–91

31 Bellare M, Garay J A, Rabin T. Fast batch verification for modular exponentiation and digital signatures. In: Proceeding of International Conference on the Theory and Applications of Cryptographic Techniques, 1998. 236–250

32 Li H W, Lin X D, Yang H M, et al. EPPDR: an efficient privacy-preserving demand response scheme with adaptive key evolution in smart grid. IEEE Trans Paral Distrib Syst, 2014, 25: 2053–2064

33 Dan B, Lynn B, Shacham H. Short signatures from the weil pairing. In: Proceeding of International Conference on the Theory and Application of Cryptology and Information Security, 2001. 514–532

34 Failla P. Privacy preserving processing of biometric templates by homomorphic encryption. Dissertation for Ph.D. Degree. Siena: University of Siena, 2011

35 Lynn B. PBC: the pairing-based cryptography library. Version 0.5.14, 2013. http://crypto.stanford.edu/pbc/