

Research Article

Lightweight Data Aggregation Scheme against Internal Attackers in Smart Grid Using Elliptic Curve Cryptography

Debiao He,^{1,2} Sherali Zeadally,³ Huaqun Wang,⁴ and Qin Liu¹

¹State Key Lab of Software Engineering, Computer School, Wuhan University, Wuhan, China

²Co-Innovation Center for Information Supply & Assurance Technology, Anhui University, Hefei, China

³College of Communication and Information, University of Kentucky, Lexington, KY, USA

⁴Jiangsu Key Laboratory of Big Data Security & Intelligent Processing, Nanjing University of Posts and Telecommunications, Nanjing, China

Correspondence should be addressed to Qin Liu; qinliu@whu.edu.cn

Received 17 February 2017; Accepted 30 March 2017; Published 9 May 2017

Academic Editor: Jaime Lloret

Copyright © 2017 Debiao He et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Recent advances of Internet and microelectronics technologies have led to the concept of smart grid which has been a widespread concern for industry, governments, and academia. The openness of communications in the smart grid environment makes the system vulnerable to different types of attacks. The implementation of secure communication and the protection of consumers' privacy have become challenging issues. The data aggregation scheme is an important technique for preserving consumers' privacy because it can stop the leakage of a specific consumer's data. To satisfy the security requirements of practical applications, a lot of data aggregation schemes were presented over the last several years. However, most of them suffer from security weaknesses or have poor performances. To reduce computation cost and achieve better security, we construct a lightweight data aggregation scheme against internal attackers in the smart grid environment using Elliptic Curve Cryptography (ECC). Security analysis of our proposed approach shows that it is provably secure and can provide confidentiality, authentication, and integrity. Performance analysis of the proposed scheme demonstrates that both computation and communication costs of the proposed scheme are much lower than the three previous schemes. As a result of these aforementioned benefits, the proposed lightweight data aggregation scheme is more practical for deployment in the smart grid environment.

1. Introduction

By providing bidirectional communications of electricity and information, the smart grid performs real-time monitoring of power usage [1]. Based on the real-time information, the providers can monitor the power generation and consumption and get immediate power demand of each area. Then, they can take prompt action to optimize the power supply. The consumer can also get the current power price and adjust his/her behavior to lower expenses. Therefore, the smart grid can achieve efficient, economical, and reliable power services. Due to such advantages, the smart grid was a widespread concern for governments, industry, and academia in the last decade and is considered as the most promising candidate of the next generation power system [2].

The National Institute of Standards and Technology (NIST) presents a model and describes seven important domains of the smart grid [3]. As shown in Figure 1 [4], a smart grid consists of seven important domains, that is, the power generation (PG) domain, the power transmission (PT) domain, the power distribution (PD) domain, the power customer (PC) domain, the power operation (PO) domain, the power market (PM) domain, and the power service provider (PSP) domain [5, 6]. After being generated, transmitted, and distributed in the PG domain, the PT domain, and the PD domain, respectively, the customers in the PC domain can enjoy wonderful life based on the power. The PO domain, the PM domain, and the PSP domain manage the power flow, the participants, and all third-party operations, respectively [7, 8].

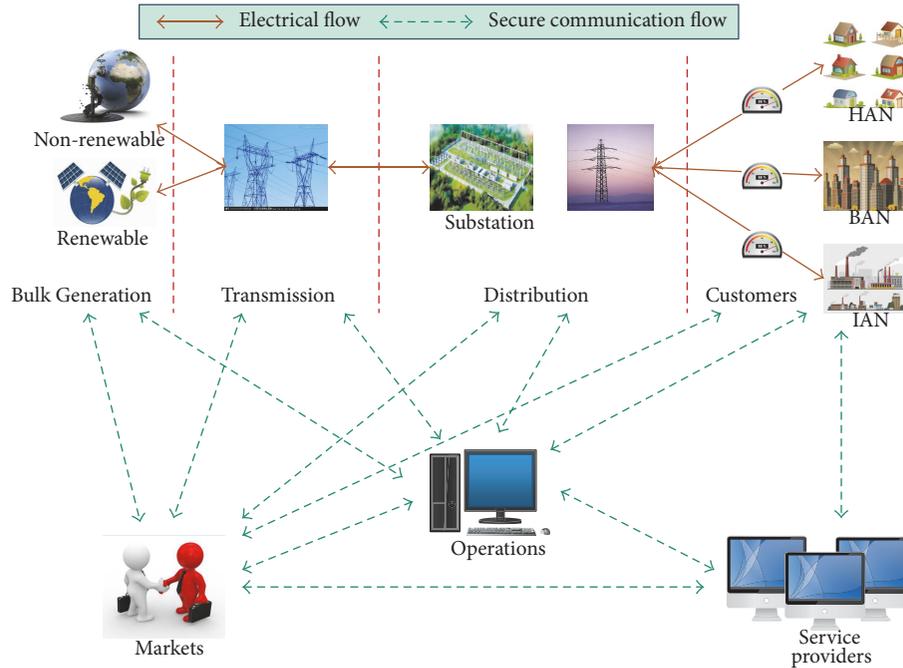


FIGURE 1: The model of the smart grid.

The smart meters in the smart grid collect the consumers' power consumption data and other information and send them to the remote control center. Generally speaking, the smart meter is installed outside the door of a consumer and an attacker is in charge of the communication channel easily due to its openness. The attacker may maliciously modify the power consumption data to increase/decrease the consumer's power expense. He/she also can get the daily routine of the consumer in order to commit crimes. For example, he/she knows that the consumer goes out when there is no power consumption and sneaks into the house to steal expensive things.

To address the above problems, how to achieve secure communications in the smart grid becomes an issue that needs to be addressed. In particular, ensuring the data's integrity and confidentiality is even more important. Several cryptographic schemes can be applied for secure communications in the smart grid. Many key management schemes [9–11], key distribution schemes [12–14], and key agreement schemes [15–17] were presented in recent years. However, many of these schemes cannot implement the integrity and confidentiality simultaneously. To address this challenge, data aggregation schemes have been proposed by several researchers and applied in the smart grid. However, most of them are vulnerable to attacks from internal attackers. Although several data aggregation schemes against internal attackers were proposed to enhance security, their computation or communication costs are too high for practical smart grid applications. In addition, the smart meter has very limited computation and communication capabilities. It is therefore necessary to design lightweight data aggregation schemes for practical deployment.

1.1. Our Contributions. To reduce both computation and communication costs, we propose a lightweight data aggregation scheme based on the Elliptic Curve Cryptography (ECC) [18, 19], which can obtain the same security level but with a much shorter key size. The main contributions of our paper are demonstrated as follows:

- (i) First, we propose a lightweight data aggregation scheme based on Schnorr's signature scheme [18].
- (ii) Second, we prove that the proposed lightweight data aggregation scheme is secure and is able to satisfy security requirements.
- (iii) Finally, we analyze the performance of the proposed lightweight data aggregation scheme to demonstrate its high performance.

1.2. Organization of the Paper. In Section 2, we briefly review related papers about data aggregation schemes. In Section 3, we give some preliminaries, including backgrounds of ECC, network model, and security requirements of the data aggregation scheme. In Section 4, we present our lightweight data aggregation scheme based on ECC. In Section 5, we describe a security model for the data aggregation scheme and present the security analyses of our scheme. In Section 6, we present the computation and communication analyses of our data aggregation scheme.

2. Related Works

To guarantee secure communication in open environments, a lot of authentication schemes [20–22], encryption schemes [23–26], and secure outsourcing schemes [25, 27, 28] have

been constructed in last several years. Li et al. [29] and Garcia and Jacobs [30] designed two data aggregation schemes using Paillier's encryption scheme [31]. To improve performance, Lu et al. [32] designed an improved data aggregation scheme using Paillier's encryption scheme and the super-increasing sequence. However, the above three schemes [29, 30, 32] cannot protect consumers' privacy because none of them can provide anonymity. To protect consumers' privacy, Zhang et al. [33] designed a security-enhanced data aggregation scheme based on the Chinese Remainder Theorem and Paillier's encryption scheme. Chen et al. [34] also designed a security-enhanced data aggregation scheme with fault tolerance based on Paillier's encryption scheme.

Unfortunately, internal attacks are not considered in the above data aggregation schemes [29, 30, 32–34] thereby allowing internal attackers to access the consumers smart grid data. To address this weakness, Fan et al. [35] designed the first data aggregation scheme that can withstand attacks from internal attackers by using blinding technology. Unfortunately, Bao and Lu [36] demonstrated that Fan et al.'s data aggregation scheme cannot guarantee the integrity of transmitted data. To enhance security, He et al. [4] designed an improved data aggregation scheme based on Boneh et al.'s encryption scheme [37]. The performance of Fan et al.'s data aggregation scheme [35] and He et al.'s data aggregation scheme [4] is not good enough because they use bilinear pairing operations.

3. Preliminaries

3.1. Elliptic Curve. Given a prime number p , we say that the equation $y^2 = x^3 + a \cdot x + b \pmod{p}$ defines an elliptic curve $E(F_p)$, where $a, b \in F_p$ and $\Delta = 4a^3 + 27b^2 \neq 0 \pmod{p}$ [38]. It is well known that all points on $E(F_p)$ and the infinite point \mathcal{O} make an additive group \mathcal{G} . Given a generator point P with a prime order q , the scale multiplication operation is defined as $n \cdot P = P + P + \dots + P_n \text{ times}$, where n is a positive integer.

Previous researches have showed that the following problems in the group \mathcal{G} are suitable for the design of public key cryptography because no probabilistic polynomial time algorithm can solve them efficiently [38].

Discrete Logarithm (DL) Problem. Given an element $Q \in \mathcal{G}$, the DL problem is to extract an element $x \in Z_q^*$ such that $Q = x \cdot P$.

Computational Diffie-Hellman (CDH) Problem. Given two elements $x \cdot P, y \cdot P \in \mathcal{G}$ with two unknown elements $x, y \in Z_q^*$, the CDH problem is to extract the element $Q = x \cdot y \cdot P$.

3.2. Network Model. As shown in Figure 2 [4], there are three participants in the system of a data aggregation scheme, namely, a trusted third party (TTP), an aggregator (Agg), and a smart meter (SM_i) [4, 35]. The functions of the above three participants are presented as below.

- (i) TTP: it is a trusted third party and its function is to generate blinding factors to withstand the internal attackers.

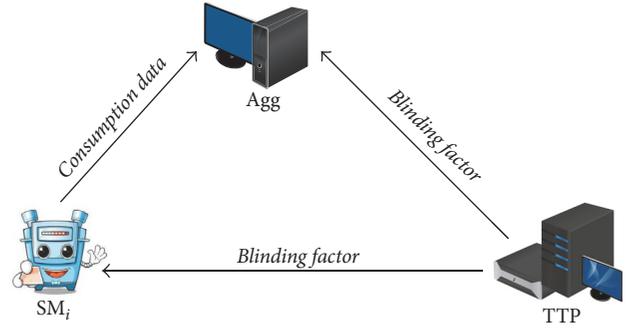


FIGURE 2: The registration phase.

- (ii) Agg: it is the manager of the smart grid and its function is to generate the system parameters and the private keys of smart meters.
- (iii) SM_i : it is a smart meter and its function is to collect consumers' electricity consumption data and send it to Agg.

The workflow of the system is presented as follows. (1) Agg produces the system parameters and the master private key; (2) SM_i registers in Agg and gets its private key; (3) TTP generates the blinding factors for Agg and SM_i ; (4) SM_i collects the electricity consumption, produces a ciphertext, and sends it to Agg; (5) after collecting all ciphertexts, Agg checks their validity and extracts the sum of all electricity consumption data.

3.3. Security Requirements. Based on recently works, we know that a data aggregation scheme for the smart grid should meet the below security requirements [4, 35].

(i) **Confidentiality.** The consumer's power consumption data indicates his/her habit and its leakage may be used by an attacker to commit a crime. To ensure the consumer's safety, a data aggregation scheme should provide confidentiality; that is, both the external attackers and the internal attackers cannot extract the electricity consumption data from intercepted messages.

(ii) **Authentication.** The malicious attacker may forge a message and impersonate the consumer. To ensure if the received message is transmitted by a legal SM_i , a data aggregation scheme should provide authentication; that is, Agg can check the legality of the received message.

(iii) **Integrity.** All messages are transmitted over open communication channels and the malicious attacker may modify them to break regular transactions. To protect the rights and interests of all participants in the smart grid, a data aggregation scheme should provide integrity; that is, Agg can detect any modification of the received data.

(iv) **Resistance against Attacks.** Due to the openness of communication channels in the smart grid, the system is vulnerable to many types of attacks. To obtain secure communications in the smart grid, a data aggregation scheme should

supply resistance against attacks; that is, it can withstand the replay attack, the modification attack, the man-in-the-middle attack, and the impersonation attack.

4. The Proposed Data Scheme

We describe our proposed lightweight data aggregation scheme, which consists of three phases, namely, the initialization phase, the registration phase, and the aggregation phase.

Initialization Phase. In this phase, Agg executes some steps to produce the system parameters. TTP and Agg execute some other steps to produce the blind factors against internal attackers.

Agg runs the following steps to produce the system parameters.

- (1) Agg selects an elliptic curve $E(F_p)$ determined by the equation $y^2 = x^3 + a \cdot x + b \pmod p$, where p is a prime and $a, b \in Z_q$.
- (2) Agg selects an element P with the order q existing on $E(F_p)$, where q is a prime.
- (3) Agg selects an element $s \in Z_q^*$ and calculates $P_{\text{pub}} = s \cdot P$.
- (4) Agg selects three cryptographic hash functions $h_i : \{0, 1\}^* \rightarrow Z_q^*$ ($i = 1, 2, 3$).
- (5) Agg publishes params = $\{p, a, b, q, P, P_{\text{pub}}, h_1, h_2, h_3\}$ and saves s secretly.

TTP and Agg execute the following steps to produce the blinding factors.

- (1) TTP randomly selects a group of elements $\theta_1, \theta_2, \dots, \theta_n \in Z_q^*$ and computes $\theta = \sum_{i=1}^n \theta_i \pmod q$. At last, TTP sends θ to Agg and also sends θ_i to SM_i , where $i = 1, 2, \dots, n$.
- (2) Agg computes $\theta_0 = -\theta \pmod q$ and keeps it secretly.

Registration Phase. In this phase, SM_i registers in Agg. After registration, SM_i receives its private key and becomes a legal smart meter. As demonstrated in Table 1, SM_i and Agg run the following processes to finish the registration.

- (1) SM_i randomly chooses an element $x'_i \in Z_q^*$, computes $X'_i = x'_i \cdot P$, and transmits $\{\text{id}_i, X'_i\}$ to Agg secretly.
- (2) Agg randomly chooses an element $x''_i \in Z_q^*$ and computes $X''_i = x''_i \cdot P$, $X_i = X'_i + X''_i$, $\alpha_i = h_1(\text{id}_i, X_i)$, and $s''_i = s + \alpha_i \cdot x''_i \pmod q$. At last, Agg sends $\{s''_i, X''_i\}$ to SM_i secretly.
- (3) SM_i computes $X_i = X'_i + X''_i$, $\alpha_i = h_1(\text{id}_i, X_i)$, $s_i = s''_i + x'_i \cdot \alpha_i \pmod q$ and checks if the equation $s_i \cdot P = P_{\text{pub}} + \alpha_i \cdot X_i$ holds. If not, SM_i rejects the session; otherwise, SM_i stores $\{s_i, X_i\}$ and finishes the registration.

TABLE 1: The registration phase of our scheme.

SM_i	Agg
Generate $x'_i \in Z_q^*$; $X'_i = x'_i \cdot P$	
	$\xrightarrow{\{\text{id}_i, X'_i\}}$
	Generate $x''_i \in Z_q^*$; $X''_i = x''_i \cdot P$; $X_i = X'_i + X''_i$; $\alpha_i = h_1(\text{id}_i, X_i)$; $s''_i = s + \alpha_i \cdot x''_i \pmod q$
	$\xleftarrow{\{s''_i, X''_i\}}$
$X_i = X'_i + X''_i$; $\alpha_i = h_1(\text{id}_i, X_i)$; $s_i = s''_i + \alpha_i \cdot x'_i \pmod q$; check $s_i \cdot P \stackrel{?}{=} P_{\text{pub}} + \alpha_i \cdot X_i$; store $\{s_i, X_i\}$	

Due to the fact that $X'_i = x'_i \cdot P$, $X''_i = x''_i \cdot P$, $X_i = X'_i + X''_i$, $s''_i = s + \alpha_i \cdot x''_i \pmod q$, and $s_i = s''_i + x'_i \cdot \alpha_i \pmod q$, then we have

$$\begin{aligned}
s_i \cdot P &= (s''_i + x'_i \cdot \alpha_i) \cdot P = (s + \alpha_i \cdot x''_i + \alpha_i \cdot x'_i) \cdot P \\
&= (s + \alpha_i \cdot (x'_i + x''_i)) \cdot P \\
&= s \cdot P + \alpha_i \cdot (x'_i + x''_i) \cdot P \\
&= P_{\text{pub}} + \alpha_i \cdot (x'_i \cdot P + x''_i \cdot P) \\
&= P_{\text{pub}} + \alpha_i \cdot (X'_i + X''_i) = P_{\text{pub}} + \alpha_i \cdot X_i.
\end{aligned} \tag{1}$$

Therefore, the correctness of the registration phase is demonstrated.

Aggregation Phase. In this phase, SM_i extracts the power consumption data and sends it to Agg. Agg checks the validity of the received messages and aggregates all the received data. As demonstrated in Table 1, the steps below are executed by SM_i and Agg.

- (1) SM_i gets the power consumption data m_i , randomly chooses an element $y_i \in Z_q^*$, and computes $Y_i = y_i \cdot P$, $\hat{Y}_i = y_i \cdot P_{\text{pub}}$, $c_i = m_i + \theta_i + h_2(\hat{R}_i) \pmod q$, $\beta_i = h_3(\text{id}_i, X_i, Y_i, c_i, t)$, and $d_i = s_i + \beta_i \cdot y_i \pmod q$. At last, SM_i transmits $\{X_i, Y_i, c_i, d_i, t\}$ to Agg.
- (2) Agg checks if $d_i \cdot P = P_{\text{pub}} + \alpha_i \cdot X_i + \beta_i \cdot Y_i$, where $\alpha_i = h_1(\text{id}_i, X_i)$ and $\beta_i = h_3(\text{id}_i, X_i, Y_i, c_i, t)$. To improve performance, we use the small exponent test technology [39] to achieve the batch verification. Agg randomly chooses a group of integers $z_1, z_2, \dots, z_n \in [1, 2^w]$ and checks if the equation $(\sum_{i=1}^n z_i \cdot d_i) \cdot P = (\sum_{i=1}^n z_i)P_{\text{pub}} + \sum_{i=1}^n (z_i \cdot \alpha_i \cdot X_i + z_i \cdot \beta_i \cdot Y_i)$ holds. Agg computes $c = \sum_{i=1}^n (c_i - h_2(s \cdot Y_i))$ and extracts the sum of the power consumption data by computing $m = c + \theta_0 \pmod q$.

Due to the fact that $s_i \cdot P = P_{\text{pub}} + \alpha_i \cdot X_i$, $Y_i = y_i \cdot P$, $\hat{Y}_i = y_i \cdot P_{\text{pub}}$, $c_i = m_i + \theta_i + h_2(\hat{Y}_i) \bmod q$ and $d_i = s_i + \beta_i \cdot y_i \bmod q$, we can derive

$$\begin{aligned}
d_i \cdot P &= (s_i + \beta_i \cdot y_i) \cdot P = s_i \cdot P + \beta_i \cdot y_i \cdot P \\
&= P_{\text{pub}} + \alpha_i \cdot X_i + \beta_i \cdot Y_i \\
\left(\sum_{i=1}^n z_i \cdot d_i \right) \cdot P &= \left(\sum_{i=1}^n z_i \cdot (s_i + \beta_i \cdot y_i) \right) \cdot P \\
&= \left(\sum_{i=1}^n z_i \cdot (s_i + \beta_i \cdot y_i) \right) \cdot P \\
&= \sum_{i=1}^n (z_i \cdot s_i \cdot P + z_i \cdot \beta_i \cdot y_i \cdot P) \\
&= \sum_{i=1}^n (z_i \cdot (P_{\text{pub}} + \alpha_i \cdot X_i) + z_i \cdot \beta_i \cdot Y_i) \\
&= \left(\sum_{i=1}^n z_i \right) \cdot P_{\text{pub}} + \sum_{i=1}^n (z_i \cdot \alpha_i \cdot X_i + z_i \cdot \beta_i \cdot Y_i), \\
c + \theta_0 &= \sum_{i=1}^n (c_i - h_2(s \cdot Y_i)) + \theta_0 \\
&= \sum_{i=1}^n (m_i + \theta_i + h_2(\hat{R}_i) - h_2(s \cdot y_i \cdot P)) - \theta \\
&= \sum_{i=1}^n (m_i + \theta_i + h_2(\hat{R}_i) - h_2(y_i \cdot s \cdot P)) - \theta \\
&= \sum_{i=1}^n (m_i + \theta_i + h_2(\hat{R}_i) - h_2(y_i \cdot P_{\text{pub}})) - \theta \\
&= \sum_{i=1}^n (m_i + \theta_i + h_2(\hat{R}_i) - h_2(\hat{R}_i)) - \theta \\
&= \sum_{i=1}^n m_i + \sum_{i=1}^n \theta_i - \sum_{i=1}^n \theta_i = \sum_{i=1}^n m_i.
\end{aligned} \tag{2}$$

According to the above equations, the correctness of the aggregation phase of our scheme is demonstrated.

5. Security Analysis

The security of the proposed lightweight data aggregation scheme is analyzed in this section. First, we present a security model for the data aggregation scheme. Second, we demonstrate that the proposed lightweight data aggregation scheme is provably secure in the security model. Finally, we demonstrate that the proposed lightweight data aggregation scheme can meet the security requirements presented in Section 3.

5.1. Security Model. Based on security models [40] for sign-cryption schemes, we presented a security model for data

aggregation schemes. The security of confidentiality and unforgeability is formally defined by two games executed by an attacker \mathcal{A} and a challenger \mathcal{C} . \mathcal{A} is allowed to make the following queries.

- (i) $h_i(m)$: for such a query made by \mathcal{A} , \mathcal{C} randomly selects $r \in Z_q^*$, sends r to \mathcal{A} , and stores (m, r) in the table L_{h_i} , where $i = 1, 2, 3$.
- (ii) CreateSM(id_i): for such a query made by \mathcal{A} , \mathcal{C} generates SM_i 's secret key and blinding factor and stores them in the table L_{SM} .
- (iii) CorruptSM(id_i): for such a query made by \mathcal{A} , \mathcal{C} sends SM_i 's private key and blinding factor to \mathcal{A} .
- (iv) Signcrypt(id_i, m_i): for such a query made by \mathcal{A} , \mathcal{C} generates a ciphertext $\{X_i, Y_i, c_i, d_i, t\}$ corresponding to the message m_i .
- (v) Designcrypt($\text{id}_i, X_i, Y_i, c_i, d_i, t$): for the query made by \mathcal{A} , \mathcal{C} checks the validity of the ciphertext and decrypts it to get the plaintext.

Definition 1. A data aggregation scheme is able to provide confidentiality [indistinguishability against adaptive chosen ciphertext attacks (IND – CCA)] if no attacker can win the following game with a nonnegligible advantage.

Setup. \mathcal{C} produces system parameters and transmits them to \mathcal{A} .

Phase 1. \mathcal{A} is able to adaptively make h_i , CreateSM, CorruptSM, Signcrypt, and Designcrypt queries.

Challenge. \mathcal{A} picks a challenging identity id_i^* , chooses two messages m_0 and m_1 , and sends them to \mathcal{C} . \mathcal{C} picks a random element $b \in \{0, 1\}$, produces a signcrypted ciphertext $\{X_i, Y_i, c_i, d_i, t\}$, and sends it to \mathcal{A} .

Phase 2. In this phase, \mathcal{A} can adaptively make h_i , CreateSM, CorruptSM, and Signcrypt queries except that it cannot make a CorruptSM query with id_i^* or a Designcrypt query with $\{X_i, Y_i, c_i, d_i, t\}$.

Finally, \mathcal{A} gives its guess $b' \in \{0, 1\}$ about the value of b selected by \mathcal{C} .

The advantage of \mathcal{A} is defined by the equation $\text{Adv}_{\mathcal{A}}^{\text{IND-CCA}} = |2 \cdot \Pr[b' = b] - 1|$. \mathcal{A} wins in the above game if it guesses the value of b correctly.

Definition 2. A data aggregation scheme is able to provide unforgeability [existential unforgeability against adaptive chosen messages attacks (EUF CMA)] if no attacker wins the following game with a nonnegligible advantage.

Setup. \mathcal{C} produces the system parameters and sends them to \mathcal{A} .

Query. In this phase, \mathcal{A} picks a challenging identity id_i^* and is able to adaptively make h_i , CreateSM, CorruptSM, Signcrypt, and Designcrypt queries except that it cannot make a CorruptSM query with id_i^* .

Forgery. In this phase, \mathcal{A} outputs a ciphertext $\{X_i, Y_i, c_i, d_i, t\}$ corresponding to the challenging identity id_i^* .

We say \mathcal{A} wins in the above game if $\{X_i, Y_i, c_i, d_i, t\}$ is valid and it is not generated by executing a Signcrypt query.

5.2. Security Analysis

Theorem 3. *The proposed data aggregation scheme is able to provide confidentiality if the CDH problem is hard.*

Proof. Assume that an attacker \mathcal{A} wins the game defined in Definition 1 with a nonnegligible advantage ϵ . Based on \mathcal{A} 's capability, we can construct a challenger to solve the CDH problem with a nonnegligible advantage. Given an instance $(P, Q_1 = a \cdot P, Q_2 = b \cdot P)$ of the CDH problem, \mathcal{C} sets $P_{\text{pub}} \leftarrow a \cdot P$ and sends $\text{params} = \{p, a, b, q, P, P_{\text{pub}}, h_1, h_2, h_3\}$ to \mathcal{A} . \mathcal{C} randomly picks up an identity id_I as the challenging identity and answers queries from \mathcal{A} according to the rules below.

- (i) $h_i(m)$: \mathcal{C} keeps a table L_{h_i} of the form (m, r) , where $i \in \{1, 2, 3\}$. Upon receiving such a query, \mathcal{C} checks if L_{h_i} contains a tuple (m, r) . If so, \mathcal{C} sends r to \mathcal{A} ; otherwise, \mathcal{C} randomly selects an element $r \in Z_q^*$, stores (m, r) into L_{h_i} , and sends r to \mathcal{A} .
- (ii) $\text{CreateSM}(id_i)$: \mathcal{C} keeps a table L_{SM} of the form $(id_i, \theta_i, s_i, X_i)$. Upon receiving such a query, \mathcal{C} checks if L_{SM} contains a tuple $(id_i, \theta_i, s_i, X_i)$. If so, \mathcal{C} sends X_i to \mathcal{A} ; otherwise, \mathcal{C} randomly selects three integers $\theta_i, \alpha_i, s_i \in Z_q^*$ and sets $X_i \leftarrow \alpha_i^{-1} \cdot (s_i \cdot P - P_{\text{pub}})$. \mathcal{C} stores (id_i, X_i, α_i) and $(id_i, \theta_i, s_i, X_i)$ into L_{SM} , respectively.
- (iii) $\text{CorruptSM}(id_i)$: \mathcal{C} checks if L_{SM} contains a tuple $(id_i, \theta_i, s_i, X_i)$. If not, \mathcal{C} makes CreateSM -query with the identity id_i . After that, \mathcal{C} returns $(id_i, \theta_i, s_i, X_i)$ to \mathcal{C} .
- (iv) $\text{Signcrypt}(id_i, m_i)$: \mathcal{C} checks if L_{SM} contains a tuple $(id_i, \theta_i, s_i, X_i)$. If not, \mathcal{C} makes CreateSM -query with the identity id_i . After that, \mathcal{C} gets the tuple $(id_i, \theta_i, s_i, X_i)$ from L_{SM} and uses it to produce a ciphertext $\{X_i, Y_i, c_i, d_i, t\}$. At last, \mathcal{C} sends $\{X_i, Y_i, c_i, d_i, t\}$ to \mathcal{A} .

Given the power consumption data m_0 and m_1 , \mathcal{C} extracts $(id_I, \theta_I, s_I, X_I)$ from L_{SM} and selects a random element $b \in \{0, 1\}$. \mathcal{C} sets $Y_I \leftarrow b \cdot P$, randomly selects three elements $\beta_I, c_I, d_I \in Z_q^*$, stores $(id_I, X_I, Y_I, c_I, t, \beta_I)$ into L_{h_3} , and sends $\{X_I, Y_I, c_I, d_I, t\}$ to \mathcal{A} .

After that, \mathcal{A} can make h_i , CreateSM , CorruptSM , and Signcrypt queries and get the corresponding responses. Then, \mathcal{A} outputs b' as his/her guess against the confidentiality. \mathcal{C} selects a random tuple (R, r) from L_{h_2} and outputs R as the solution of the given CDH problem.

Let q_{h_2} denote the number of h_2 -query involved in the game. The probability that \mathcal{C} can solve the given CDH problem is $\eta = \epsilon/q_{h_2}$. Because of the nonnegligibility of ϵ , we know that η is nonnegligible. This contradicts with the hardness of the CDH problem. Thus, the proposed data aggregation scheme is able to provide confidentiality. \square

Theorem 4. *The proposed data aggregation scheme is able to provide unforgeability if the DL problem is hard.*

Proof. Assume that an attacker \mathcal{A} wins the game defined in Definition 1 with a nonnegligible advantage ϵ . Based on \mathcal{A} 's capability, we can construct a challenger to solve the DL problem with a nonnegligible advantage. Given an instance $(P, Q_1 = a \cdot P)$ of the DL problem, \mathcal{C} picks a random integer $s \in Z_q^*$, computes $P_{\text{pub}} = s \cdot P$, and sends $\text{params} = \{p, a, b, q, P, P_{\text{pub}}, h_1, h_2, h_3\}$ to \mathcal{A} . \mathcal{C} randomly selects an identity id_I as the challenging identity and answers queries from \mathcal{A} according to the rules below.

- (i) $h_i(m)$: \mathcal{C} keeps a table L_{h_i} of the form (m, r) , where $i \in \{1, 2, 3\}$. Upon receiving such a query, \mathcal{C} checks if L_{h_i} contains a tuple (m, r) . If so, \mathcal{C} sends r to \mathcal{A} ; otherwise, \mathcal{C} randomly picks up an element $r \in Z_q^*$, stores (m, r) into L_{h_i} , and sends r to \mathcal{A} .
- (ii) $\text{CreateSM}(id_i)$: \mathcal{C} keeps a table L_{SM} of the form $(id_i, \theta_i, s_i, X_i)$. Upon receiving such a query, \mathcal{C} checks if L_{SM} contains a tuple $(id_i, \theta_i, s_i, X_i)$. If so, \mathcal{C} sends X_i to \mathcal{A} ; otherwise, \mathcal{C} answers the query through the rules below:
 - (1) If $id_i = id_I$, \mathcal{C} randomly picks two integers $\theta_i, \alpha_i \in Z_q^*$ and sets $X_i \leftarrow a \cdot P$. \mathcal{C} stores (id_i, X_i, α_i) and $(id_i, \theta_i, \perp, X_i)$ into L_{SM} , respectively.
 - (2) Otherwise ($id_i \neq id_I$), \mathcal{C} randomly selects three integers $\theta_i, \alpha_i, s_i \in Z_q^*$ and sets $X_i \leftarrow \alpha_i^{-1} \cdot (s_i \cdot P - P_{\text{pub}})$. \mathcal{C} stores (id_i, X_i, α_i) and $(id_i, \theta_i, s_i, X_i)$ into L_{SM} , respectively.
- (iii) $\text{CorruptSM}(id_i)$: \mathcal{C} checks if L_{SM} contains a tuple $(id_i, \theta_i, s_i, X_i)$. If not, \mathcal{C} makes CreateSM -query with the identity id_i . After that, \mathcal{C} returns $(id_i, \theta_i, s_i, X_i)$ to \mathcal{A} .
- (iv) $\text{Signcrypt}(id_i, m_i)$: \mathcal{C} checks if id_i and id_I are equal. If they are not, \mathcal{C} extracts the tuple $(id_i, \theta_i, s_i, X_i)$ from L_{SM} and uses it to produce a ciphertext $\{X_i, Y_i, c_i, d_i, t\}$ according to the description of the proposed data aggregation; otherwise, \mathcal{C} randomly selects two integers $d_i, \beta_i \in Z_q^*$ and computes $Y_i = \beta_i^{-1} \cdot (d_i \cdot P - P_{\text{pub}} - \alpha_i \cdot X_i)$ and $c_i = m_i + \theta_i + h_2(s \cdot Y_i)$. \mathcal{C} stores (id_i, X_i, Y_i, c_i, t) into L_{h_2} and sends $\{X_i, Y_i, c_i, d_i, t\}$ to \mathcal{A} .
- (v) $\text{Designcrypt}(id_i, X_i, Y_i, c_i, d_i, t)$: for the query made by \mathcal{A} , \mathcal{C} checks the validity of the ciphertext and decrypts it to get the plaintext using the systems secret key s . \square

At last, \mathcal{A} outputs a forged ciphertext $(id_i, X_i, Y_i, c_i, d_i, t)$. \mathcal{C} stops the game if the equation $id_i = id_I$ holds. Based on the forking lemma [41], \mathcal{C} can output another valid ciphertext $(id_i, X_i, Y_i, c_i, d_i', t)$ by choosing a different hash function h_1 . Since both ciphertexts are valid, we can derive the following two equation:

$$\begin{aligned} d_i \cdot P &= P_{\text{pub}} + \alpha_i \cdot X_i + \beta_i \cdot Y_i, \\ d_i' \cdot P &= P_{\text{pub}} + \alpha_i' \cdot X_i + \beta_i \cdot Y_i. \end{aligned} \quad (4)$$

Based on the above two equations, we can derive the equation below:

$$\begin{aligned}
(d_i - d'_i) \cdot P &= d_i \cdot P - d'_i \cdot P \\
&= (P_{\text{pub}} + \alpha_i \cdot X_i + \beta_i \cdot Y_i) \\
&\quad - (P_{\text{pub}} + \alpha'_i \cdot X_i + \beta_i \cdot Y_i) \\
&= (\alpha_i - \alpha'_i) \cdot X_i = (\alpha_i - \alpha'_i) \cdot a \cdot P.
\end{aligned} \tag{5}$$

\mathcal{C} outputs $(d_i - d'_i) \cdot (\alpha_i - \alpha'_i)^{-1}$ as the solution of the given DL problem. To compute the probability that \mathcal{C} solves the DL problem, three related events are listed below.

- (i) E_1 : id_i equals id_i .
- (ii) E_2 : \mathcal{C} is able to forge a legal ciphertext.

Let q_{h_1} denote the number of h_1 involved in the game. It is easy to get that $\Pr[E_1] = 1/q_{h_1}$ and $\Pr[E_2|E_1] = \epsilon$. Then, the probability that \mathcal{C} solves the DL problem is $\eta = \Pr[E_1 \wedge E_2] = \Pr[E_2|E_1] \cdot \Pr[E_1] = \epsilon/q_{h_1}$. Because of the nonnegligibility of ϵ , we know that η is nonnegligible. This is in contradiction with the hardness of the DL problem. Thus, the proposed data aggregation scheme is able to provide unforgeability.

5.3. Analysis of Security Requirements. We will show that the proposed lightweight data aggregation scheme is able to meet security requirements presented in Section 3.

(i) *Confidentiality.* The internal attacker against the proposed data aggregation scheme can compute $c = \sum_{i=1}^n (c_i - h_2(s \cdot Y_i))$. Without the blinding factor c , he/she cannot extract the sum of the power consumption data by computing $m = c + \theta_0 \bmod q$. Besides, Theorem 4 shows that the proposed lightweight data aggregation scheme can supply confidentiality against any external attacker. Thus, our lightweight data aggregation scheme can supply confidentiality.

(ii) *Authentication.* Theorem 3 shows that any attacker cannot forge a legal ciphertext. Then, Agg can verify the legality of received messages by verifying if the equation $(\sum_{i=1}^n z_i \cdot d_i) \cdot P = (\sum_{i=1}^n z_i)P_{\text{pub}} + \sum_{i=1}^n (z_i \cdot \alpha_i \cdot X_i + z_i \cdot \beta_i \cdot Y_i)$ holds. Therefore, the proposed data aggregation scheme can provide authentication.

(iii) *Integrity.* Theorem 3 demonstrates that any attacker against the proposed data aggregation scheme cannot forge a legal ciphertext. Agg can detect any modification of the received data by verifying if the equation $(\sum_{i=1}^n z_i \cdot d_i) \cdot P = (\sum_{i=1}^n z_i)P_{\text{pub}} + \sum_{i=1}^n (z_i \cdot \alpha_i \cdot X_i + z_i \cdot \beta_i \cdot Y_i)$ holds. Therefore, the proposed data aggregation scheme can provide integrity.

(iv) *Resistance against Attacks.* The proposed lightweight data aggregation scheme can resist the replay attack, the modification attack, the man-in-the-middle attack, and the impersonation attack. The reason is analyzed below.

(1) *Replay Attack.* The timestamp t is involved in the ciphertext. Agg can find any reply of previous message by verifying

t 's freshness. Thus, the proposed lightweight data aggregation scheme can resist the replay attack.

(2) *Modification Attack.* Theorem 3 demonstrates that any attacker against the proposed data aggregation scheme cannot forge a legal ciphertext. Agg can detect any modification of the received data by verifying if $(\sum_{i=1}^n z_i \cdot d_i) \cdot P = (\sum_{i=1}^n z_i)P_{\text{pub}} + \sum_{i=1}^n (z_i \cdot \alpha_i \cdot X_i + z_i \cdot \beta_i \cdot Y_i)$ holds. Thus, the proposed lightweight data aggregation scheme can resist the modification attack.

(3) *Man-in-the-Middle Attack.* The above analysis demonstrates that the proposed lightweight data aggregation scheme can supply authentication; that is, Agg can authenticate SM_i by checking if $d_i \cdot P = P_{\text{pub}} + \alpha_i \cdot X_i + \beta_i \cdot Y_i$ holds. Thus, the proposed lightweight data aggregation scheme can resist the man-in-the-middle attack.

(4) *Impersonation Attack.* Theorem 4 shows that any attacker cannot forge a legal ciphertext without SM_i 's secret key. Then, Agg can detect any impersonation by verifying the validity of the received ciphertext. Therefore, the proposed lightweight data aggregation scheme can resist the impersonation attack.

6. Performance Analysis

We analyze both computation and communication costs of our lightweight data aggregation scheme in this section. We also compare its performance with two of the most recently proposed data aggregation schemes to show its lightweight costs.

To achieve a fair comparison, we compare recently proposed aggregation schemes under the same security level. In the BGN encryption scheme [37], two 512-bit prime numbers $p = 2 \cdot p' + 1$ and $q = 2 \cdot q' + 1$ are applied in our experiments, where p' and q' are also large prime numbers. In schemes based on bilinear pairing, a Tate pairing based on a Type A elliptic curve $\bar{E} : y^2 = x^3 + 1 \bmod \bar{p}$ with a prime order \bar{q} is applied in our experiments, where the lengths of \bar{p} and \bar{q} are 512 bits and 160 bits, respectively. In schemes based on ECC, an elliptic curve $\bar{E} : y^2 = x^3 + a \cdot x + b \bmod \bar{p}$ with a prime order \bar{q} is applied in our experiments, where the lengths of \bar{p} and \bar{q} are 160 bits.

6.1. Analysis of Computation Costs. Based on the well-known cryptographic library MIRACL [42], we have implemented all related operations on a personal computer with an Intel I5-3210M 2.50 GHz Center Processor Unit (CPU), an 8 Gbyte Random Access Memory (RAM), and the Windows 7 operation system. Table 3 presents the operations' notations and runtime results.

Each SM_i in the Fan et al.'s scheme [35] runs one BGN encryption operation, one exponentiation in BGN algorithm, two multiplications related to BGN algorithm, one HTP_{G_1} operation, one PM_{G_1} operation, and one general hash function. Therefore, SM_i 's runtime is $T_{\text{ENC}_{\text{BGN}}} + T_{\text{EXP}_{\text{BGN}}} + 2 \times T_{\text{MUL}_{\text{BGN}}} + T_{\text{HTP}_{\text{BP}}} + T_{\text{PM}_{\text{BP}}} + T_{\text{GH}} = 8.315 + 8.096 + 2 \times 0.032 + 14.293 + 5.485 + 0.001 = 36.254$ ms. Agg in Fan et al.'s scheme [35] runs one BGN decryption, one exponentiation

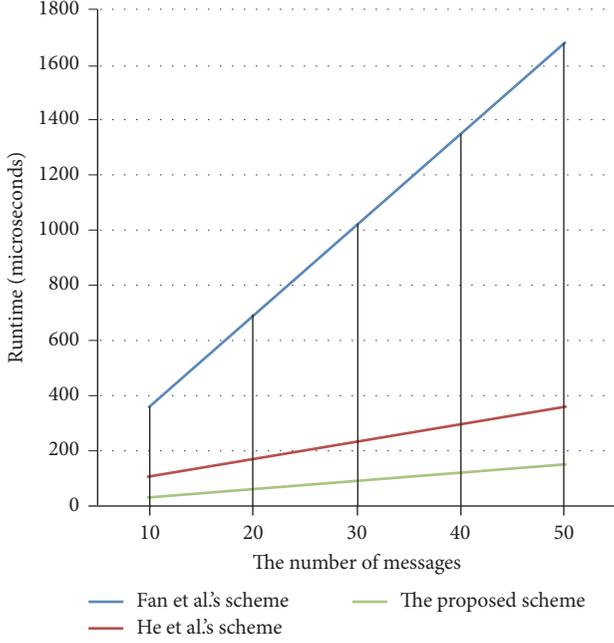


FIGURE 3: Runtime comparisons of related schemes.

related to the BGN algorithm, $n - 1$ multiplication related to BGN algorithm, n hash-to-point, $n + 1$ bilinear pairing, $2n$ point multiplication related to the bilinear pairing, $n - 1$ point multiplication with a short exponent related to the bilinear pairing, $n - 1$ exponentiation related to the bilinear pairing, and one general hash function. Therefore, Agg's runtime is $T_{\text{DEC}_{\text{BNG}}} + T_{\text{exp-BNG}} + (n - 1) \cdot T_{\text{MUL}_{\text{BNG}}} + n \cdot T_{\text{HTP}_{\text{BP}}} + (n + 1) \cdot T_{\text{BP}} + (2n) \cdot T_{\text{PM}_{\text{BP-s}}} + (n - 1) \cdot T_{\text{PA}_{\text{BP}}} + (n - 1) \cdot T_{\text{EXP}_{\text{BP}}} + T_{\text{GH}} = 4.056 + 8.096 + (n - 1) \times 0.032 + n \times 14.293 + (n + 1) \times 17.001 + (2n) \times 0.343 + (n - 1) \times 0.023 + (n - 1) \times 0.874 + 0.001 = (32.909 \cdot n + 28.225)$ microseconds.

Each SM_i in the proposed scheme executes two point multiplication operations related to ECC and two general hash functions. Therefore, SM_i 's runtime is $2 \times T_{\text{PM}_{\text{ECC}}} + 2 \times T_{\text{GH}} = 2 \times 0.986 + 2 \times 0.001 = 1.974$ microseconds. Agg in the proposed scheme executes $3 \times n + 2$ point multiplication related to ECC, $2 \times n$ point addition related to ECC, and $3 \times n$ general hash functions. Therefore, Agg's runtime is $(3 \times n + 2) \times T_{\text{PM}_{\text{ECC}}} + 2 \times n \times T_{\text{PA}_{\text{ECC}}} + 3 \times n \times T_{\text{GH}} = (3 \times n + 2) \times 0.986 + 2 \times n \times 0.004 + 3 \times n \times 0.001 = 2.969 \cdot n + 1.972$.

Table 4 and Figure 3 show the runtime comparisons among Fan et al.'s data aggregation scheme [35], He et al.'s scheme [4], and the proposed scheme. From Tables 4 and 2, the proposed scheme incurs a lower computation cost as compared to Fan et al.'s scheme and He et al.'s scheme at both sides of SM_i and Agg.

6.2. Analysis of Communication Costs. Since the sizes of p_1 , q_1 , p' , q' , \hat{p} , and \hat{q} are 512 bits, 512 bits, 512 bits, 160 bits, 1024 bits, and 160 bits, respectively, we can determine that the sizes of elements in Z_n^* , G_1 , G_2 , $Z_{q'}^*$, $Z_{\hat{p}}^*$, and $Z_{\hat{q}}^*$ are 1024 bits, 1024 bits, 1024 bits, 160 bits, 1024 bits, and 160 bits, respectively. We assume that the size of both the timestamp and the identity

are each 32 bits. The communication costs of the related data aggregation schemes are shown below.

In Fan et al.'s data aggregation scheme [35] SM_i sends the message $(\delta_i, \text{CT}_i, t)$ to Agg, where $\delta_i \in G_1$, $\text{CT}_i \in Z_n^*$, and t is the timestamp. Therefore, the communication cost of Fan et al.'s data aggregation scheme is $1024 + 1024 + 32 = 2080$ bits. In He et al.'s data aggregation scheme [4] SM_i sends the message $(\text{ID}_i, R_i, \delta_i, \text{CT}_i, t)$ to Agg, where $R_i \in G_1$, $\delta_i \in Z_{q'}^*$, $\text{CT}_i \in G_1$, ID_i is SM_i 's identity, and t is the timestamp. Therefore, the communication cost of He et al.'s data aggregation scheme is $32 + 1024 + 160 + 1024 + 32 = 2272$ bits. In the proposed data aggregation scheme, SM_i sends the message (c_i, d_i, e_i, t) to Agg, where $c_i \in Z_n^*$, $d_i \in Z_{\hat{p}}^*$, $e_i \in Z_{\hat{q}}^*$, and t is the timestamp. Therefore, the communication cost of the proposed data aggregation scheme is $1024 + 1024 + 160 + 32 = 2240$ bits.

Based on the above evaluation, we note that the proposed data aggregation scheme incurs lower communication cost than He et al.'s data aggregation scheme. The proposed data aggregation scheme incurs a higher communication cost than Fan et al.'s data aggregation scheme. Security is most important for a data aggregation scheme. Therefore, it is reasonable to address serious security weaknesses in Fan et al.'s data aggregation scheme at the cost of increasing the communication cost slightly.

7. Conclusion

To ensure security and protect privacy in the smart grid environment, several data aggregation schemes have been proposed in recent years. However, most of them are not secure against internal attackers. To address the problem, Fan et al. [35] proposed a data aggregation scheme to mitigate internal attacks. Unfortunately, their data aggregation scheme suffers from serious security weaknesses. To enhance security, He et al. [4] proposed an improved data aggregation scheme using bilinear pairing. However, the performance of He et al.'s scheme is not very suitable for the smart grid environment because the smart meter has limited computation capability. In this paper, we have proposed a novel data aggregation scheme that can thwart internal attacks for the smart grid environment. The security analysis shows that the proposed scheme is provably secure and can meet the security requirements. Besides, performance evaluation results show that the proposed scheme incurs lower communication costs. The stronger security and better performance of the proposed scheme demonstrate that it is more suitable for smart grids.

With the fast development of quantum computing, the traditional mathematical problems (such as the DL problem and the CDH problem) are likely to be solved in polynomial time by quantum computers. Subsequently, all the above data aggregation schemes for the smart grid will not be secure at all. The lattice has been widely used to construct many cryptographic schemes that can provide resistance against the strong capabilities of quantum computers. However, no data aggregation scheme based on the lattice has been proposed yet. To improve security, it is worthwhile to consider the design of a data aggregation scheme for the smart grid based on the lattice approach.

TABLE 2: The aggregation phase.

SM _i	Agg
Extract m_i ; generate $y_i \in Z_q^*$; $Y_i = y_i \cdot P$; $\hat{Y}_i = y_i \cdot P_{\text{pub}}$; $c_i = m_i + \theta_i + h_2(\hat{Y}_i) \bmod q$; $\beta_i = h_3(\text{id}_i, X_i, Y_i, c_i, t)$; $d_i = s_i + \beta_i \cdot y_i \bmod q$	
$\xrightarrow{\{X_i, Y_i, c_i, d_i, t\}}$	
	$\alpha_i = h_1(\text{id}_i, X_i)$; $\beta_i = h_3(\text{id}_i, X_i, Y_i, c_i, t)$; generate $z_1, \dots, z_n \in [1, 2^w]$; check $\left(\sum_{i=1}^n z_i \cdot d_i \right) \cdot P \stackrel{?}{=} \left(\sum_{i=1}^n z_i \right) \cdot P_{\text{pub}} + \sum_{i=1}^n (z_i \cdot \alpha_i \cdot X_i + z_i \cdot \beta_i \cdot Y_i)$; $c = \sum_{i=1}^n (c_i - h_2(s \cdot Y_i))$; $m = c - \theta_0$

TABLE 3: Notations about related operations and runtime results (microseconds).

Notation	Description	Runtime
ENC _{BGN}	BGN encryption	8.315
DEC _{BGN}	BGN decryption	4.056
EXP _{BGN}	Exponentiation related to BGN algorithm	8.096
MUL _{BGN}	Multiplication related to BGN algorithm	0.032
BP	Bilinear pairing	17.001
HTP	Hash-to-point	14.293
PM _{BP}	Point multiplication related to the bilinear pairing	5.485
PM _{BP-s}	Point multiplication with a short exponent related to the bilinear pairing	0.343
PA _{BP}	Point addition related to the bilinear pairing	0.023
EXP _{BP}	Exponentiation related to the bilinear pairing	0.874
MUL _{BP}	Multiplication related to the bilinear pairing	0.005
EXP _{DLP}	Exponentiation related to the DL problem	1.295
EXP _{DLP-s}	Exponentiation with a short exponent related to the DL problem	0.081
MUL _{DLP}	Multiplication related to the DL problem	0.012
PM _{ECC}	Point multiplication related to ECC	0.986
PM _{ECC-s}	Point multiplication with a short exponent related to ECC	0.061
PA _{ECC}	Point addition related to ECC	0.004
GH	General hash function	0.001

TABLE 4: Runtime comparisons (microseconds).

	Fan et al.'s scheme	He et al.'s scheme	The proposed scheme
SM _i	36.254	20.145	17.751
Agg	32.909n + 28.225	6.264n + 48.249	2.969n + 1.972

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The work was supported by the National Natural Science Foundation of China (nos. 61572370, 61501333, 61572379, and U1536204), the National High-Tech Research and Development Program of China (863 Program) (no. 2015AA016004), and the Natural Science Foundation of Hubei Province of China (no. 2015CFB257). Sherali Zeadally's work has been supported by a University Research Professorship Award from the University of Kentucky.

References

- [1] Z. M. Fadlullah, M. M. Fouda, N. Kato, A. Takeuchi, N. Iwasaki, and Y. Nozaki, "Toward intelligent machine-to-machine communications in smart grid," *IEEE Communications Magazine*, vol. 49, no. 4, pp. 60–65, 2011.
- [2] Y. Zhang, R. Yu, M. Nekovee, Y. Liu, S. Xie, and S. Gjessing, "Cognitive machine-to-machine communications: visions and potentials for the smart grid," *IEEE Network*, vol. 26, no. 3, pp. 6–13, 2012.
- [3] C. Greer, D. A. Wollman, D. E. Prochaska et al., "Nist framework and roadmap for smart grid interoperability standards, release 3.0".

- [4] D. He, N. Kumar, and J.-H. Lee, "Privacy-preserving data aggregation scheme against internal attackers in smart grids," *Wireless Networks*, vol. 22, no. 2, pp. 491–502, 2016.
- [5] J. Gao, Y. Xiao, J. Liu, W. Liang, and C. L. P. Chen, "A survey of communication/networking in smart grids," *Future Generation Computer Systems*, vol. 28, no. 2, pp. 391–404, 2012.
- [6] N. Saputro, K. Akkaya, and S. Uludag, "A survey of routing protocols for smart grid communications," *Computer Networks*, vol. 56, no. 11, pp. 2741–2771, 2012.
- [7] V. C. Güngör, D. Sahin, T. Kocak et al., "Smart grid technologies: communication technologies and standards," *IEEE Transactions on Industrial Informatics*, vol. 7, no. 4, pp. 529–539, 2011.
- [8] W. Su, H. Eichi, W. Zeng, and M.-Y. Chow, "A survey on the electrification of transportation in a smart grid environment," *IEEE Transactions on Industrial Informatics*, vol. 8, no. 1, pp. 1–10, 2012.
- [9] D. Wu and C. Zhou, "Fault-tolerant and scalable key management for smart grid," *IEEE Transactions on Smart Grid*, vol. 2, no. 2, pp. 375–381, 2011.
- [10] Z. Wan, G. Wang, Y. Yang, and S. Shi, "SKM: scalable key management for advanced metering infrastructure in smart grids," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 12, pp. 7055–7066, 2014.
- [11] K. Yu, M. Arifuzzaman, Z. Wen, D. Zhang, and T. Sato, "A key management scheme for secure communications of information centric advanced metering infrastructure in smart grid," *IEEE Transactions on Instrumentation and Measurement*, vol. 64, no. 8, pp. 2072–2085, 2015.
- [12] J. H. Park, M. Kim, and D. Kwon, "Security weakness in the smart grid key distribution scheme proposed by xia and wang," *IEEE Transactions on Smart Grid*, vol. 4, no. 3, pp. 1613–1614, 2013.
- [13] J.-L. Tsai and N.-W. Lo, "Secure Anonymous Key Distribution Scheme for Smart Grid," *IEEE Transactions on Smart Grid*, vol. 7, no. 2, pp. 906–914, 2016.
- [14] D. He, H. Wang, M. K. Khan, and L. Wang, "Lightweight anonymous key distribution scheme for smart grid using elliptic curve cryptography," *IET Communications*, vol. 10, no. 14, pp. 1795–1802, 2016.
- [15] H. Liu, H. Ning, Y. Zhang, and M. Guizani, "Battery status-aware authentication scheme for V2G networks in smart grid," *IEEE Transactions on Smart Grid*, vol. 4, no. 1, pp. 99–110, 2013.
- [16] H. Li, R. Lu, L. Zhou, B. Yang, and X. Shen, "An efficient Merkle-tree-based authentication scheme for smart grid," *IEEE Systems Journal*, vol. 8, no. 2, pp. 655–663, 2014.
- [17] N. Saxena, B. J. Choi, and R. Lu, "Authentication and Authorization Scheme for Various User Roles and Devices in Smart Grid," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 5, pp. 907–921, 2016.
- [18] N. Koblitz, "Elliptic curve cryptosystems," *Mathematics of Computation*, vol. 48, no. 177, pp. 203–209, 1987.
- [19] V. S. Miller, "Use of elliptic curves in cryptography," in *Proceedings of the Conference on the Theory and Application of Cryptographic Techniques*, pp. 417–426, Springer, 1985.
- [20] X. Huang, Y. Xiang, E. Bertino, J. Zhou, and L. Xu, "Robust multi-factor authentication for fragile communications," *IEEE Transactions on Dependable and Secure Computing*, vol. 11, no. 6, pp. 568–581, 2014.
- [21] X. Huang, Y. Xiang, A. Chonka, J. Zhou, and R. H. Deng, "A generic framework for three-factor authentication: Preserving security and privacy in distributed systems," *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 8, pp. 1390–1397, 2011.
- [22] D. He, S. Zeadally, N. Kumar, and J.-H. Lee, "Anonymous authentication for wireless body area networks with provable security," *IEEE Systems Journal*, 2016.
- [23] Z. Fu, X. Sun, Q. Liu, L. Zhou, and J. Shu, "Achieving efficient cloud search services: multi-keyword ranked search over encrypted cloud data supporting parallel computing," *IEICE Transactions on Communications*, vol. E98B, no. 1, pp. 190–200, 2015.
- [24] Z. Fu, X. Wu, C. Guan, X. Sun, and K. Ren, "Toward efficient multi-keyword fuzzy search over encrypted outsourced data with accuracy improvement," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 12, pp. 2706–2716, 2016.
- [25] Z. Fu, K. Ren, J. Shu, X. Sun, and F. Huang, "Enabling personalized search over encrypted outsourced data with efficiency improvement," *IEEE Transactions on Parallel and Distributed Systems*, vol. 9, no. 27, pp. 2546–2559, 2015.
- [26] Z. Xia, X. Wang, X. Sun, Q. Liu, and Q. Wang, "A secure and dynamic multi-keyword ranked search scheme over encrypted cloud data," *IEEE Transactions on Parallel and Distributed Systems*, vol. 2, no. 27, pp. 340–352, 2015.
- [27] X. F. Chen, J. Li, J. Ma, Q. Tang, and W. Lou, "New algorithms for secure outsourcing of modular exponentiations," *IEEE Transactions On Parallel And Distributed Systems*, vol. 25, no. 9, pp. 2386–2396, 2014.
- [28] X. Chen, J. Li, X. Huang, J. Ma, and W. Lou, "New Publicly Verifiable Databases with Efficient Updates," *IEEE Transactions on Dependable and Secure Computing*, vol. 12, no. 5, pp. 546–556, 2015.
- [29] F. Li, B. Luo, and P. Liu, "Secure information aggregation for smart grids using homomorphic encryption," in *Proceedings of the 1st IEEE International Conference on Smart Grid Communications (SmartGridComm '10)*, pp. 327–332, 2010.
- [30] F. D. Garcia and B. Jacobs, "Privacy-friendly energy-metering via homomorphic encryption," in *Proceedings of the International Workshop on Security and Trust Management*, pp. 226–238, Springer, 2010.
- [31] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," vol. 1592, pp. 223–238, Springer, Berlin, Germany, Proceedings of the International Conference on the Theory and Applications of Cryptographic Techniques.
- [32] R. Lu, X. Liang, X. Li, X. Lin, and X. Shen, "EPPA: an efficient and privacy-preserving aggregation scheme for secure smart grid communications," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 9, pp. 1621–1632, 2012.
- [33] J. Zhang, L. Liu, Y. Cui, and Z. Chen, "SP²DAS: self-certified PKC-based privacy-preserving data aggregation scheme in smart grid," *International Journal of Distributed Sensor Networks*, vol. 2013, Article ID 457325, 11 pages, 2013.
- [34] L. Chen, R. Lu, and Z. Cao, "PDAFT: a privacy-preserving data aggregation scheme with fault tolerance for smart grid communications," *Peer-to-Peer Networking and Applications*, vol. 8, no. 6, pp. 1122–1132, 2014.
- [35] C.-I. Fan, S.-Y. Huang, and Y.-L. Lai, "Privacy-enhanced data aggregation scheme against internal attackers in smart grid," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 1, pp. 666–675, 2014.
- [36] H. Bao and R. Lu, "Comment on 'Privacy-Enhanced Data Aggregation Scheme Against Internal Attackers in Smart Grid,'" *IEEE Transactions on Industrial Informatics*, vol. 12, no. 1, pp. 2–5, 2016.

- [37] D. Boneh, E.-J. Goh, and K. Nissim, "Evaluating 2-dnf formulas on ciphertexts," in *Proceedings of the Theory of Cryptography Conference*, vol. 3378 of *Lecture Notes in Comput. Sci.*, pp. 325–341, Springer, Berlin, Germany, 2005.
- [38] H. Cohen, G. Frey, R. Avanzi et al., *Handbook of Elliptic and Hyperelliptic Curve Cryptography*, CRC Press, 2005.
- [39] J. K. Liu, T. H. Yuen, M. H. Au, and W. Susilo, "Improvements on an authentication scheme for vehicular sensor networks," *Expert Systems with Applications*, vol. 41, no. 5, pp. 2559–2564, 2014.
- [40] M. Barbosa and P. Farshim, "Certificateless signcryption," in *Proceedings of the ACM Symposium on Information, Computer and Communications Security (ASIACCS '08)*, pp. 369–372, ACM, March 2008.
- [41] D. Pointcheval and J. Stern, "Security arguments for digital signatures and blind signatures," *Journal of Cryptology*, vol. 13, no. 3, pp. 361–396, 2000.
- [42] M. Scott, Miracl library, 2011, <http://www.shamus.ie>.



Hindawi

Submit your manuscripts at
<https://www.hindawi.com>

