Elliptic Curve-Based Secure Multidimensional Aggregation for Smart Grid Communications

Omar Rafik Merad Boudia, Sidi Mohammed Senouci, Mohammed Feham

Elliptic Curve Based Secure Multidimensional Aggregation for Smart Grid Communications

Omar Rafik Merad Boudia, Sidi Mohammed Senouci, *Member, IEEE*, and Mohammed Feham

***Abstract*— In Smart Grid, data aggregation is considered as an essential paradigm in assessing information about current energy usage. To achieve the privacy-preserving goal, several homomorphic-based solutions have been proposed. However, these solutions either consider one-dimensional information or use costly pairing computation in order to ensure source authentication. In fact, smart grid data is likely to be multidimensional (e.g. time, purpose, etc.) for more accurate control. In addition, the aggregation node in smart grid needs to verify data that come from several smart meters in a residential area; hence, the verification must be cost-efficient. In this paper, we propose a scheme that considers multidimensional aggregation with privacy preserving and an efficient verification of smart grid data. The proposal is based on elliptic curve cryptography along with homomorphic encryption and without pairings. The performance analysis shows the efficiency of the scheme for smart grid communications in comparison to existing schemes. For instance, we show that, when an aggregator node is responsible of 600 smart meters, it spends approximately 14s to verify the data in pairing-based schemes, while only 0.3s is needed for verification within the proposed scheme.**

***Index Terms*— Smart grid, privacy-preserving, elliptic curve cryptography, data aggregation, homomorphic encryption.**

NOMENCLATURE

SG	Smart Grid
SM	Smart Meter
HE	Homomorphic Encryption
ECC	Elliptic Curve Cryptography
ECEG	Elliptic Curve El Gamal
ECDSA	Elliptic Curve Digital Signature Algorithm
MRES	Multi-Recipient Encryption Scheme
HAN	Home Area Network
NAN	Neighborhood Area Network
ID_x	The identifier of X
CC	Control Center (Identifier : ID_{cc})
AGG_j	A NAN aggregator j (Identifier : ID_j)
U_{ij}	User i that belong to AGG_j (Identifier : ID_{ij})
Enc_{KA-B}	A symmetric encryption using K_{A-B}
V	A large integer used in *Enc.*
E	A standard elliptic curve
p	The prime that defines the field F_p of E
G	The base point of E
n	The order of E
H	A secure hash function
x(G)	The x coordinate of curve point G

O.R. Merad Boudia and M. Feham are with the STIC Laboratory, University of Tlemcen, 13000, Algeria (e-mail: om_meradboudia, m_feham@mail.univ-tlemcen.dz).

S.M. Senouci is with the DRIVE Laboratory, University of Burgundy, Franche Comté, 49 Rue Mademoiselle Bourgeois, 58000, Nevers, France (e-mail: Sidi-Mohammed.Senouci@u-bourgogne.fr).

I. Introduction

RECENTLY, the Smart Grid (SG) has been proposed as the next-generation approach to making more efficient and more reliable energy service. In fact, the smart grid makes possible to learn about (and respond to) changing electricity demand in real time, which is extensively desirable with the super-increasing numbers of electronic devices and technological capabilities in homes and businesses (electric vehicles, data centers, etc.). Consequently, the inherent problems occurred in traditional grids such as the lack of load balancing, smart consumption, and dynamic pricing can be solved [1].

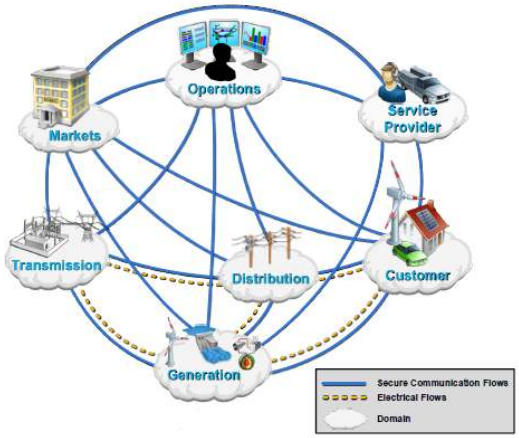

Fig. 1. NIST's SG conceptual model [2].

A conceptual smart grid model from NIST is presented in Fig.1. The SG consists of seven domains: Transmission, Distribution, Operations, Generation, Markets, Customer, and Service Provider. One of the most important components in SG is the Smart Meter (SM) in the customer domain, which is responsible for recording the electricity use and sending the information to the utility company [2]. In order to improve the efficiency and reliability, the SG considers two-way com-

munication between the utility company and its customers, which improves sustainability and security as well [3]. For instance, customers can see, using SM, their electricity use on a real-time basis. As a result, they can reduce their use during peak periods when it is more expensive, which in one hand saves money and in other hand decreases the pressure on the central power grid. In smart grid, data aggregation is an essential paradigm to efficiently manage the energy supply. In fact, for billing, the Control Center (CC) at the operations domain needs individual data that are collected over long intervals (e.g. a week or a month). However, the CC requires data that are collected much more frequently (e.g. every 10 min) for monitoring and control. So, the huge amount of real-time data, in this case, can be aggregated in the network before being processed and used [4]. Consequently, by reducing the traffic, data aggregation can improve the network's efficiency.

The frequent real-time data collection from SM makes data aggregation risky. Indeed, on one hand, the individual data that are aggregated contains sensitive information about customers (e.g. habits and lifestyles), and on the other hand, the data are transmitted through public accessible communication channels. Consequently, the privacy preserving is crucial in order to prevent the disclosure of sensitive information to adversaries, thus avoiding the profiling of customer's personal behavior. Also, the attacks against data integrity can severely impact consistency of the smart grid. In fact, false data injection or alteration could directly decrease the reliability and the quality of the power grid and cause financial impacts as well as annoyance on customers [5]. Therefore, ensuring data integrity and source authentication is essential for smart grid networks.

In order to guarantee the user privacy, the untrustworthy entities, including CC and relay nodes, should not be able to access the sensitive information. For that purpose, Homomorphic Encryption (HE) techniques are considered [6]. HE enables direct calculations on encrypted data. In other words, relay nodes perform the aggregation function directly on received ciphertexts. This enables the CC to compute the total consumption value without compromising the privacy of individual households. Existing solutions either consider one-dimensional information or conduct to a considerable charge in terms of communication and computation. In fact, the CC can utilize other information in addition to the amount of electricity use (e.g. time, purpose, etc.) for more accurate control, so the data in this case should be multidimensional. The challenge here is to aggregate the values in such way that each data type remains private while at the same time the CC only processes the corresponding aggregates.

Furthermore, pairing-based schemes have to work in a running environment with parameters of 1024 bits so as to offer 80 bits security level [7]. Indeed, before performing aggregation, the relay node needs to verify all received data, and given the traffic that traverses that node, the costly pairing operation can severely impact on the reliability of the network, even if batch verification is considered. Elliptic Curve Cryptography (ECC) provides the same level of security with smaller working parameters, namely 160 bits [7]. In this paper, we propose a scheme that considers multidimensional data aggregation with efficient security algorithms for smart grid. Summary of our contributions in this research are:

- First, we employ MRES, a Multi-Recipient Encryption Scheme to secure multidimensional data. Recall that the multi-dimensional data allow CC to use other information than electricity use for more accurate control. The encryption scheme considered is ECEG (Elliptic Curve El-Gamal) [8] for its efficiency in terms of computation and communication. In addition, a reference technique is introduced to overcome the ECEG issues,
- Second, we adopt ECDSA (Elliptic Curve Digital Signature Algorithm) [9] with batch verification to allow intermediate nodes to efficiently verify data integrity and authenticate the senders,
- Finally, the analysis and performance evaluation show that compared with existing secure aggregation schemes for SG communications, the scheme significantly reduces the computation cost and communication overhead.

The rest of this paper is organized as follows. In Section II, we review related work in the area. Section III presents the overview and recalls the foundation works of the proposed scheme. We describe the details of this latter in Section IV. The security analysis and performance evaluation are given in Sections V and VI, respectively. Finally, we draw the conclusions in Section VII.

II. Related Work

A number of works have been developed to secure aggregation in SG. In [10], a spanning tree rooting at the collector device is built to cover all of the SMs. Aggregation is then performed in a distributed manner in accordance to the aggregation tree. The one-dimensional data are encrypted and aggregated using the Paillier cryptosystem [11], so that inputs and intermediate results are not revealed to SMs on the aggregation path. The same cryptosystem is used [12], an Efficient Privacy-Preserving Demand Response (EPPDR) in which an adaptive key evolution technique is used to ensure the users' session keys to be forwarded securely. In [13], Lu et al. propose EPPA, an Efficient and Privacy-Preserving Aggregation. EPPA uses the homomorphic Paillier cryptosystem to achieve privacy-preserving data aggregation. The authors employ a super-increasing sequence to allow multidimensional aggregation. The authors also use a bilinear maps based signature and consider batch verification at aggregator level for efficiency purpose. In [14], the authors propose a privacy enhanced data aggregation. They consider blinding factors to create blinded one-dimensional data in order to prevent internal attackers from learning the electricity consumption of users. Like EPPA, the authors consider batch verification from bilinear maps. Fu et al. [15] propose a fault-tolerance SG communication model, in which the contributed members can be accurately determined to avoid random errors. The authors utilize El-Gamal encryption to achieve privacy-preserving data aggregation and employ a submission information list in order to verify and recover the data. The authors also use a

pairing-based signature with corresponding batch verification.

In the aforesaid schemes, the authors either consider one-dimensional information or use costly pairing computation in order to verify the data. In addition, most of them mainly depend on Paillier's HE, which is of high computational complexity. In fact, two exponentiation operations in group Z_{n^2} are needed to be performed for each smart meter and the size of ciphertext is $2|N|$ bits, where $N = pq$, p and q should be at least 1024 bits. To overcome these drawbacks, we introduce in this paper an ECC-based secure multidimensional aggregation.

III. PRELIMINARIES

In this section, we first briefly introduce some concepts employed in our proposal, and then we formalize the network model, security model, and identify the design goals.

A. Elliptic Curve Cryptography

First introduced in [16], ECC becomes an attractive area of research in the last twenty years. The major benefits of using ECC are the highest strength-per-bit provided and the smallest key size. The security of ECC is based directly on the intractability of ECDLP (the Elliptic Curve Discrete Logarithm Problem). ECC is very useful for wireless communications and low power devices [17]. In fact, compared with traditional cryptosystem like RSA or *mod p* systems, ECC provides the same level of security with reduced key size. For example, an elliptic curve over a 160-bit field currently provides the same level of security as a 1024-bit RSA modulus or Diffie-Hellman prime. The difference becomes even more dramatic as the desired security level increases. In this paper, we use ECEG. More specifically, we use the elliptic curve analog of the Multi-Recipient El-Gamal Encryption Scheme (MRES) [18] in which the sender re-uses the random coin to encrypt different plaintexts under different public keys. MRES approximately halves the computational cost (number of exponentiations) for encryption as compared to the naive method.

MRES is described as follow: Suppose receiver i has secret key $X_i \in Z_q$ and public key g^{Xi}, the naive El-Gamal encryption is the following: Pick $R_1,..., R_n$ independently at random from Z_q and let $C_i = (g^{Ri}, g^{XiRi}. M_i)$ for $1 \le i \le n$. Instead, the authors suggest that one pick just one R at random from Z_q and set $C_i = (g^{R}, g^{XiR}. M_i)$ for $1 \le i \le n$. In our proposal, we use the Elliptic Curve analog of MRES.

In addition, we use the ECDSA [9]. The corresponding algorithm is described as follow:

The ECDSA public parameters include an elliptic curve E over F_q, a base point G of large prime order n in $E(F_q)$ and a one way hash function H.

- *Key-pair generation*: Generate randomly the private key d from $[1, n-1]$ and compute the public key $Q = dG$.
- *Signature generation*: Compute $s = k^{-1}(H(M)+ dr) \bmod n$, and produce the signature $S = (r, s)$, where M is the message, and $r = x(kG) \bmod n$, $x(kG)$ is the x-coordinate of kG.
- *Signature verification*: Compute $R = uG + vQ$, where $u = H(M)w \bmod n$, $v = rw \bmod n$, and $w = s^{-1} \bmod n$. The signature is accepted if and only if $x(R)= r \bmod n$.

B. Homomorphic Encryption

This property is a feature that can be applied to certain cryptosystems and allows calculations on ciphertexts, which have the same effect as performing these calculations on the underlying plaintext data [6]. An encryption algorithm is accepted to be homomorphic if and only if the following equation holds:

$$D(E(x) \Delta E(y)) = D(E(x \Delta y))$$

The operation Δ can support either addition or multiplication or both; it depends on the features of the encryption scheme. The HE that supports any function on ciphertexts is known as Fully Homomorphic Encryption (FHE). In [19], the Gentry's work is the first theoretical representation of FHE based on ideal lattices. It is promising, but the time complexity of its algorithms is still too high for practical use (huge key size and heavy computations). The other class of HE is Partially Homomorphic Encryption (PHE), which includes encryption schemes that have homomorphic property with respect to one operation. The authors of [8] presented several PHE. Their results showed that ECEG is the most promising scheme in terms of efficiency.

C. Castellucia et al's symmetric encryption

For the CC's response, we employ a secure symmetric encryption [20]. In order to generate the keys required for encrypting message responses, we use a Key Derivation Function (KDF). More specifically, we consider the secure KDF that uses Pseudo Random Function (HMAC) recommended by NIST, namely NIST SP800-108 HKDF (HMAC-based KDF) [21]. This is crucial for the security of the encryption as stated in [20] and used in [22]. The encryption of a message m from A to B with the key K_{A-B}, namely $Enc_{KA-B}(m, ID_A|| Nonce)$ is as follow:

- $K = HKDF(K_{A-B}, ID_A|| Nonce)$
- $C_A = K + m \bmod V$, where V is a large enough integer, $m \in [0,V-1], K \in [0,V-1]$

B decrypts C_A as follow:

- $Dec_{KA-B}(C_A, ID_A|| Nonce) = C_A - K \bmod V$

D. Network model

According to the SG architecture proposed in the previous proposals, the system generally consists of three levels. As shown in Fig. 2, the user level known as Home Area Network (HAN) in which there is a smart meter allowing two-way communication between the CC and HAN. The aggregator level known as Neighborhood Area Network (NAN) which consists in a gateway AGG serving as a relay between HAN and CC. The CC level which is responsible for collecting dynamic power usage to efficiently manages electricity generation, distribution, and allocation. We consider that the CC covers F NANs and each NAN consists of N HANs. Each HAN user U_{ij} ($i = 1,2,...,N$, $j = 1,2,...,F$) which belongs to a NAN aggregator AGG_j uses the smart meter to collect, encrypt

and send the data to the corresponding aggregator. AGG_j aggregates the received data and sends them to the CC. With the received data, the CC makes the decisions and sends the control messages to AGG_j and the corresponding SMs.

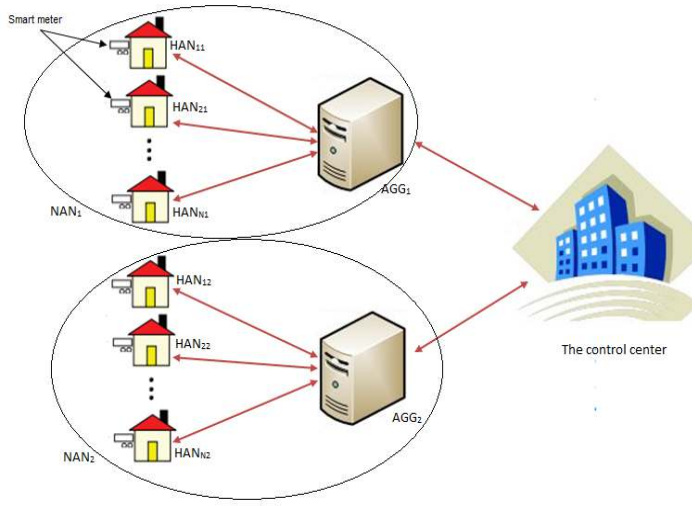

Fig. 2. Our system model.

E. Security Model

We consider an attacker A that can eavesdrop the users' data and try to identify the content of ciphertexts. A can also execute some active attacks, for example, intruding in the database of CC and AGG_j to steal the individual data or injecting fake electricity demand arbitrarily into the aggregation to bias the correct aggregation result, or even replay valid packets already "used" in the SG. Furthermore, the attacker A can capture the response messages from CC and try to identify the encrypted messages.

F. Design goal

Under the aforementioned system models, our design goal is to develop an efficient privacy-preserving demand response scheme for SG communications. More specifically, the following objectives should be achieved.

1) Security

The scheme must be secure under the attacker model mentioned above. More specifically, the following security requirement should be satisfied.

-Confidentiality: refers to preventing unauthorized persons or systems (including AGG and CC) from obtaining the relevant information to users from the transmitted data and thus achieve the privacy-preserving requirement. In fact, consumption data contain detailed information that can be used to gain insights on a customer's behavior. The CC response messages also must be confidential, i.e., only the authorized entities are allowed to read them.

-Integrity and authentication: the former refers to preventing undetected malicious operations by unauthorized persons or systems, whereas the latter means that an encrypted data is issued from a legal user. AGG_j and HAN users should be authenticated by CC and AGG_j, respectively.

2) Efficiency

The scheme must be efficient in terms of communication and computation overhead, so that the real-time high frequency data can be fast collected by CC.

IV. Proposed Scheme

In this section, we propose a scheme, which consists of four phases, as shown in Fig.3: system setup, data generation, data aggregation and response.

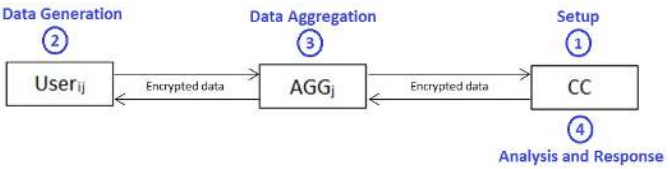

Fig. 3. The proposed scheme's phases.

A. Setup phase

We suppose that the CC bootstraps the whole system. Given the security parameter K, CC first generates the elliptic curve parameters (E, p, G, n) by running $Gen(K)$ and then calculates its public keys. Assume that there are l types of data $(t_1, t_2, \ldots, t_l)$ to be transmitted in the smart grid. Then, CC choose $l+1$ random numbers $(x_1, x_2, \ldots, x_l, x_{cc})$, where $x_1, \ldots, x_l \in [0, n-1]$ and computes the corresponding public keys $(Y_1, Y_2, \ldots, Y_l, Y_{cc})$, where $Y_i = x_iG$, $i=1..l$ and $Y_{cc} = x_{cc}G$. CC also generates l reference values $(f_1, \ldots, f_l)$ for each type of data. CC also chooses a secure hash function H (e.g. SHA-1) and a secure symmetric encryption $Enc()$ [20]. Then, CC publishes all the public information $\{ E, p, G, n, H, Enc(), Y_1, \ldots, Y_l, f_1, \ldots, f_l \}$ and keeps $\{ x_1, x_2, \ldots, x_l, x_{cc} \}$ as secret.

When a user U_{ij} joins the system, it first chooses a random number $a_{ij} \in [0, n-1]$ as the private key and computes the corresponding public key $P_{ij} = a_{ij}G$. Also, when AGG_j (the local gateway) of the residential area registers itself in the system, it generates a random number $a_j \in [0, n-1]$ as the private key and computes the corresponding public key $P_j = a_jG$. In addition, symmetric keys are computed between CC and AGG_j and also AGG_j and U_{ij} as follow:

The CC and AGG_j compute $K_{cc\text{-}AGGj} = x_{cc}P_j = a_jY_{cc} = x_{cc}a_jG$

The AGG_j and U_{ij} compute $K_{AGGj\text{-}Uij} = a_jP_{ij} = a_{ij}P_j = a_ja_{ij}G$

B. Data generation

The user U_{ij} that is in the residential area of AGG_j, first obtains its l types of data $(d_{ij1}, d_{ij2}, \ldots, d_{ijl})$ from the smart meters. Then, it performs the following:

- Compute the points $(M_{ij1}, M_{ij2}, \ldots, M_{ijl})$, where $M_{ijk} = (d_{ijk} - f_k)G$, where $k = 1..l$
- Choose two random numbers $r_{ij1}, r_{ij2} \in [0, n-1]$
- Compute $C_{ij} = (r_{ij1}G, r_{ij1}Y_1 + M_{ij1}, r_{ij1}Y_2 + M_{ij2}, \ldots, r_{ij1}Y_l + M_{ijl})$
- Compute $z_{ij} = r_{ij2}^{-1}(H(D) + a_{ij}r) \bmod n$, and produce $S_{ij} = (r_{ij2}G, z_{ij})$, where $D = C_{ij} \,||\, ID_{ij} \,||\, TS$, TS is the current time stamp, and $r = x(r_{ij2}G) \bmod n$
- Send $C_{ij} \,||\, ID_{ij} \,||\, TS \,||\, S_{ij}$ to AGG_j

C. Data aggregation

After receiving encrypted data $C_{ij} \,||\, ID_{ij} \,||\, TS \,||\, S_{ij}$, AGG_j first verifies the signature S_{ij} i.e. verifies if $r_{ij2}G = u_{ij}G + v_{ij}P_{ij}$, where $u_{ij} = H(D)w \bmod n$, $v_{ij} = rw \bmod n$, and $w = z^{-1} \bmod n$. The correctness is shown as follow:

$$\begin{aligned} u_{ij}G + v_{ij}P_{ij} &= H(D)\, z_{ij}^{-1}\, G + r\, z_{ij}^{-1} a_{ij} G \\ &= (H(D) + r\, a_{ij})\, z_{ij}^{-1}\, G \\ &= (H(D) + r\, a_{ij})\, (H(D) + r\, a_{ij})^{-1}\, (r_{ij2}^{-1})^{-1}\, G \\ &= r_{ij2}\, G \end{aligned}$$

In order to speed up the verification, the aggregator AGG_j can perform the batch verification [23] by checking:

$$\sum_{i=1}^{N} r_{ij2} G = (\sum_{i=1}^{N} u_{ij}) G + \sum_{i=1}^{N} v_{ij} P_{ij}$$

The number of scalar multiplications is then reduced from $2N$ to $N+1$.

After verification, AGG_j performs the following:

- Compute the encrypted aggregate C_j on $C_{1j}, C_{2j}, \ldots, C_{Nj}$ as

$$\begin{aligned} C_j = \sum_{i=1}^{N} C_{ij} = (&\sum_{i=1}^{N} r_{ij1} G, \sum_{i=1}^{N} r_{ij1} Y_1 + \sum_{i=1}^{N} M_{ij1}, \\ &\sum_{i=1}^{N} r_{ij1} Y_2 + \sum_{i=1}^{N} M_{ij2}, \ldots, \sum_{i=1}^{N} r_{ij1} Y_l + \sum_{i=1}^{N} M_{ijl}) \end{aligned}$$

- Choose a random number r_j and compute $z_j = r_j^{-1}(H(D) + a_j r) \bmod n$, and produce $S_j = (r_j G, z_j)$, where $D = C_j \,||\, ID_j \,||\, TS$, TS is the current time stamp, and $r = x(r_j G) \bmod n$.
- Send $C_j \,||\, ID_j \,||\, TS \,||\, S_j$ to CC

D. Analysis and response

After receiving $C_j \,||\, ID_j \,||\, TS \,||\, S_j$, CC first verifies the signature S_j i.e. verifies if $r_j G = u_j G + v_j P_j$, and then retrieves the aggregated data as follow:

$$\begin{aligned} &\sum_{i=1}^{N} M_{ij1}, \sum_{i=1}^{N} M_{ij2}, \ldots, \sum_{i=1}^{N} M_{ijl} = \sum_{i=1}^{N} r_{ij1} Y_1 + \sum_{i=1}^{N} M_{ij1} \\ &- \sum_{i=1}^{N} x_1 r_{ij1} G, \sum_{i=1}^{N} r_{ij1} Y_2 + \sum_{i=1}^{N} M_{ij2} - \sum_{i=1}^{N} x_2 r_{ij1} G, \ldots \\ &, \sum_{i=1}^{N} r_{ij1} Y_l + \sum_{i=1}^{N} M_{ijl} - \sum_{i=1}^{N} x_l r_{ij1} G \\ &\sum_{i=1}^{N} d_{ij1}, \sum_{i=1}^{N} d_{ij2}, \ldots, \sum_{i=1}^{N} d_{ijl} = RM(\sum_{i=1}^{N} M_{ij1}) + f_1, \\ &RM(\sum_{i=1}^{N} M_{ij2}) + f_2, \ldots, RM(\sum_{i=1}^{N} M_{ijl}) + f_l, \end{aligned}$$

where RM is the reverse mapping function which transforms a curve point to an integer using Pollard's lambda method [24].

After analyzing the data, the CC responds by a message m to the users U_{ij}.

- CC sends $C_{cc} \,||\, ID_{cc} \,||\, TS$, where $C_{cc} = Enc_{Kcc\text{-}AGGj}\ (m, ID_{cc} \,||\, TS)$
- AGG_j retrieves $m = Dec_{Kcc\text{-}AGGj}\ (C_{cc}, ID_{cc} \,||\, TS)$ and forwards $C_m \,||\, ID_j \,||\, TS$ to the user U_{ij}, where $C_m = Enc_{KAGGj\text{-}Uij}\ (m, ID_j \,||\, TS)$

U_{ij} retrieves $m = Dec_{KAGGj\text{-}Uij}\ (C_m, ID_j \,||\, TS)$ and uses m to manage the cost efficiency.

V. Security Analysis

In this section, we analyze the security properties of the proposed scheme.

- *The proposed scheme provides the Privacy-Preserving of User's data.*

The encryption in this scheme is an ECEG encryption. The security of ECEG is based on ECDLP, which makes the encryption secure if ECDLP is intractable. In this work, we consider a security level of 160 bits, the same provided by RSA with 1024 bits key, see Section VI. Also, the ECEG is IND-CPA secure, so the encryption is secure against any form of *ciphertext analysis*. The reference technique used in the proposal is only considered in order to speed-up encryption and decryption operations. Roughly speaking, this technique does not negate the security of ECEG. In fact, recall that the encrypted data is the difference d_i - f_i, so even if the attacker has f_i, he cannot deduce the difference d_i - f_i nor d_i. The privacy-preserving is guaranteed for all types of data because for El-Gamal, it is safe (w.r.t IND-CPA) to encrypt data under different public keys with the same randomness [18]. Furthermore, AGG_j performs aggregation directly on ciphertexts, so even if an adversary intrudes the AGG_j's database, it cannot deduce the individual user's data. Also, CC performs decryption on aggregated ciphertexts, and retrieves aggregated plaintexts, so even if an adversary intrudes the CC's database, it still cannot deduce the individual user's data. Therefore, the Privacy-Preserving is ensured in our scheme.

- *The proposed scheme ensures data integrity and source authentication of User's data, and data confidentiality for CC's response*

The ciphertext in the proposed scheme is signed using ECDSA, which is provably secure. In fact, an adversary who wants to forge a signature should either crack the hash function; or solve the ECDLP. In this scheme, we use a secure hash function (SHA-1) and a recommended elliptic curve, thus the above adversary tasks are infeasible. Therefore, data integrity and source authentication of user's data are provided. Also, when CC responds the users, CC uses *Enc()* to encrypt the response message m. Since m is encrypted using [20], which is provably secure, the confidentiality of response messages is provided.

VI. Performance Evaluation

In this section, we evaluate the computation and communication overheads in the proposed scheme.

A. Computation overhead

Before presenting the scheme's overhead in terms of

computation, we calculate using MIRACL library [25] the costs of various operations involved in the proposal. We use the standard curve SECP160R1 and SHA1 for ECC and hash operations, respectively. The experiments are conducted on a computer with Intel Core i5-2430 2.4-GHz CPU, and 2-GB RAM. The results are presented in Table I.

TABLE I
OPERATIONS COSTS

Symbol	OPERATION	Time (ms)
PM	Point Multiplication	0.40
PMp	PM with precomputation	0.17
PD	Point Decompression	0.09
PA	Point Addition	0.003
MI	Modular Inversion	0.002
HF	Hash Function	0.001
SE	Sym.Enc	0.001

From Table I, we can see that the operation cost of PA, MI, HF, and SE can be neglected with respect to elliptic curve scalar point multiplication. Consequently, in what follow we only consider PM in the analysis.

For our scheme, to generate the data C_{ij} $||ID_{ij}$ $||TS$ $||S_{ij}$, a residential user U_i needs to perform $2(l+1)$ Point Multiplication, $(2l+1)$ PM for encryption and one PM for S_{ij}'s generation. After receiving the ciphertext from N users, the local AGG_j first verifies the data integrity by performing batch verification which includes $(N+1)$ PM. After that, the AGG aggregates these authenticated data, a calculation that involves only PA operation. Next, the AGG needs to perform one PM for signature generation. After receiving the data from AGG_j, the CC first checks the integrity by performing *2* PM. Then, the decryption involves N PM and l RM to retrieve the data. Finally, CC uses symmetric encryption for response. Then, the total cost of our proposal is $(2N+l+7)$ PM + l RM. Recall that RM is the reverse mapping function and can be efficiently computed using Pollard's lambda method [24].

In the proposed scheme, the user can send a multi-dimensional data to the CC for more precise control. It could be possible for example to embed all data types l into one piece of data and encrypt it using ECEG. However, in this case, the reverse mapping function at CC becomes very difficult to perform. In our proposal, we use different public keys with a reference value to enhance the execution time. Also, the same random number is used to encrypt each type of data, indeed, this avoids one PM for each encryption at users and CC. In ECC, the points of an elliptic curve, with x and y coordinates each represented by N bits, can be compressed to only the x coordinate and a compression bit, requiring only half the space. The Point Compression (PC) helps to increase the efficiency in terms of storage and bandwidth. However, the decompression operation requires the computation of one square root in prime fields. The PD operation takes 0.09 *ms* according to Table I.

For benchmarking, we compare our approach to EPPA [13]. In fact, among related work, EPPA is the most efficient approach to secure multidimensional data, in terms of computation and communication. In Table II, we compare the scheme with EPPA in terms of computation complexity. C_e, C_m and C_p refer to computation cost of exponentiation in Z_{n^2}, multiplication in the group of pairing system and a pairing operation, respectively.

TABLE II
COMPUTATION COMPLEXITY

	USER	AGG
Our scheme	$2(l+1)$ PM	$(N+1)$ PM
Our scheme (PC)	$2(l+1)$ PM	$(N+1)$ PM $+(N+l+1)$ PD
EPPA [13]	$(l+1)$ C_e+C_m+4 C_p	$(N+3)$ C_p+ C_m

To give a sense to this and to make the comparison results more accurate and valuable, we consider the pairing-based cryptography library [26] in order to measure the EPPA's operation costs. We conduct the experiment with the same computer with a base field size of 160 bits for pairing and 1024 bits n for exponentiation, the results for C_e, C_m and C_p are 9.78 *ms*, 1.18 *ms* and 22,84 *ms*, respectively.

In Fig. 4, we depict the variation of computation costs (encryption and signature) in terms of data type number (l) for users (HAN). The figure shows that our scheme hugely reduces the computation overhead for the user. This is due to, in one hand EPPA uses a costly operation for encryption, namely, an exponentiation in Z_{n^2}, and on the other hand, it uses another costly operation in order to decrypt the message response from CC. In our proposal, the user operations involve ECC scalar multiplication, which is very efficient compared to exponent-tiation in Z_{n^2}. Also, a lightweight decryption is considered to decrypt the CC's message. Furthermore, the execution time in our scheme can be reduced even more. In fact, the point used in our system, namely $(G, Y_1, ..., Y_l)$, are fixed and known a priori. A PM with precomputation (*window = 4*) takes about 0.17 *ms*, according to Table I.

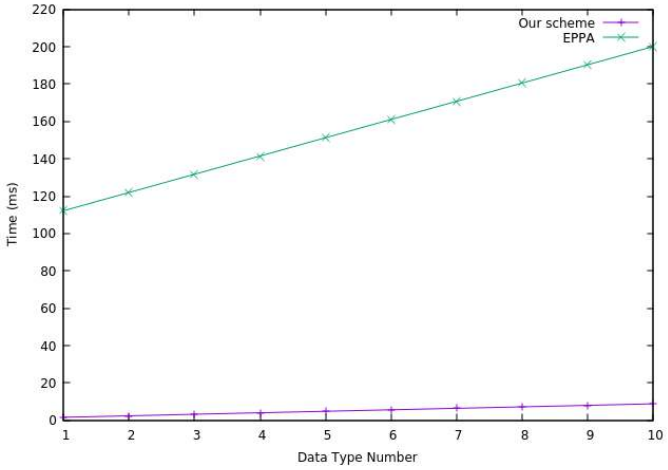

Fig. 4. Computation cost for users.

In Fig. 5, we depict the variation of computation costs in terms of users number (N) for the AGG. It is shown that there is a great difference between our proposal and EPPA's work. In fact, the pairing operation is much slower than PM. Furthermore, the two graphs (our scheme with and without PC) are almost identical. So, even if we use the PC technique and the additional overhead generated by the corresponding PD, our scheme is still efficient.

In [27], the authors investigate the overhead due to using

homomorphic encryption in SG in terms of bandwidth and end-to-end data delay when providing data privacy. They do not consider the time spent for source authentication in the latency calculation. The authors also state that the computation overhead impacts severely the end-to-end delay and must be considered. In this paper, we show that using pairing algorithms for verification leads to a non-negligible amount of time. For instance, in a NAN with 600 HANs, the corresponding AGG spends approximately 14s to verify the data received from HAN's smart meters, while only 0.3s is needed for verification in our proposal.

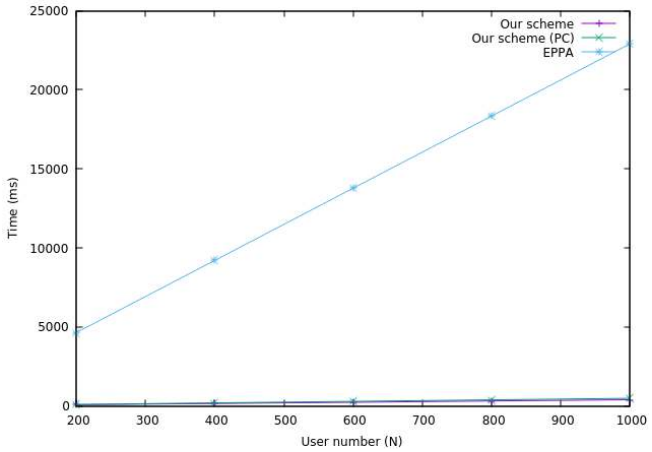

Fig. 5. Computation cost for AGG.

B. Computation overhead

In what follow, we analyze the communication overhead of our scheme according to the data report transmitted at each level of the hierarchy i.e. the user data reports for AGG and the AGG data report for CC. The data report generated by the user U_i and submitted to the AGG_j is in the form of $C_{ij}||ID_{ij}|\ TS||\ S_{ij}$. The length of the ciphertext C_{ij} is *(320+320*l)* bits without PC and *(161+161*l)* bits with PC. The length of the signature is *480* bits without PC and *321* with PC. Thus, its size can be represented as $Sz = 800 + 320*l + |\ ID_{ij}\ | + |\ TS\ |$ without PC and $Sz_c = 482 + 161*l + |\ ID_{ij}\ | + |\ TS\ |$ considering PC. So, if we have N users, the overall communication overhead will be $N*Sz$ without PC and $N*Sz_c$ considering PC.

Next, we analyze the AGG to CC communication. In that level of the hierarchy, the same amount of data i.e. Sz and Sz_c is transmitted. So, if we have K AGG, the overall communication overhead will be $K*Sz$ without PC and $K*Sz_c$ considering PC.

TABLE III
COMMUNICATION OVERHEAD

	USER -> AGG	AGG -> CC
Our scheme	$800 + 320*l + \|ID_{ij}\| + \|TS\|$	$800 + 320*l + \|ID_j\| + \|TS\|$
Our scheme (PC)	$482 + 161*l + \|ID_{ij}\| + \|TS\|$	$482 + 161*l + \|ID_j\| + \|TS\|$
EPPA [13]	$2208 + \|RA\| + \|U_j\| + \|TS\|$	$2208 + \|RA\| + \|GW\| + \|TS\|$

The impact on the communication overhead during each aggregation is shown in Table III, where we compare our scheme with our scheme with PC and EPPA. Let the size of an entity identifier and the timestamp be all 4 bytes. In Fig. 6, we plot the communication cost during each aggregation in terms of data type l. The figure shows that in our scheme, larger is the number l, and larger is the overhead at each aggregation. This is obvious because, for each new data type, a new elliptic curve point is added to the ciphertext. Besides, it is shown that the PC technique reduces significantly the communication overhead. In comparison with EPPA, our scheme with PC is more efficient in terms of communication. In fact, for a reduced number of data types our scheme preserves the network's resources e.g. for *l=5* only *1319* bits is transmitted. Note that in the EPPA's work, the authors considered only *l=10* in their experiments. For a larger number l, another larger modulus should be considered for the Paillier's encryption in order to support the corresponding super-increasing sequence. In Fig. 7, we also plot the communication overhead in terms of user number. It can be seen that the communication overhead is reduced in our proposal for large scale networks.

The aforementioned analysis concerns the user to AGG communication. Note that the same analysis can be carried out for AGG to CC communication since we have the same amount of data to be transmitted. Consequently, the proposed scheme effectively reduces the communication overhead compared to EPPA.

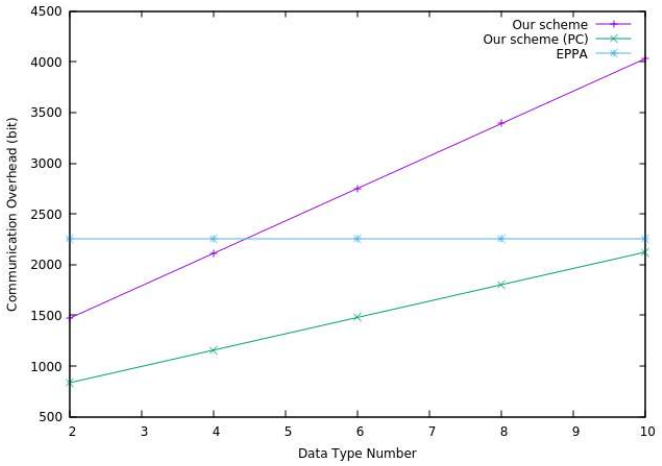

Fig. 6. Communication overhead in terms of l.

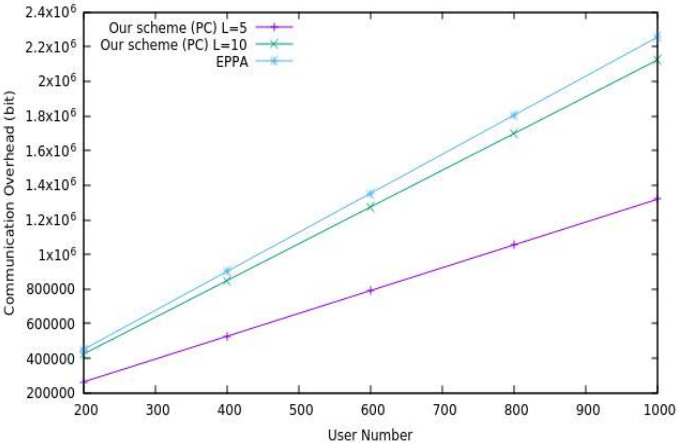

Fig. 7. Communicatin overhead in terms of user number N.

VII. CONCLUSION AND FUTURE WORK

In this paper, we proposed an efficient and secure aggregation scheme for smart grid communications based on elliptic curve cryptography. Compared with previous solutions, the scheme can achieve a comparable security level with a significant efficiency in terms of computation and communication overheads. The multidimensional data are very important for more accurate control; the analysis shows that our proposal can achieve privacy-preservation of each data type. The paper also shows the scheme's efficiency. In fact, the results show that smaller inputting parameters in pairing cryptography systems cannot bring them a good speed, and for real-time high frequency data collection, the speed efficiency is crucial. In future work, we aim to extend our work to consider new attacks such as selective forwarding.

REFERENCES

[1] Heydt, G. T. The next generation of power distribution systems. *IEEE Trans. Smart Grid*, vol. 1, no 3, p. 225-235. 2010.

[2] NIST Framework and Roadmap for Smart Grid Interoperability Standards, Release 3.0. September 2014.

[3] TAN, Song, DE, Debraj, SONG, Wen-Zhan, et al. Survey of Security Advances in Smart Grid: A Data Driven Approach. *IEEE Commun. Surveys Tuts.*, vol. 19, no 1, p. 397-422. 2017.

[4] R. Ma, H.-H. Chen, Y.-R. Huang, and W. Meng, "Smart grid communication: Its challenges and opportunities," *IEEE Trans. Smart Grid*, vol. 4, no. 1, pp. 36–46, Mar. 2013.

[5] Ruilong. Deng, Gaoxi. Xiao, Rongxing. Lu, Hao. Liang, Athanasios V. Vasilakos, "False Data Injection on State Estimation in Power Systems - Attacks Impacts and Defense: A Survey", *IEEE Trans. Ind. Inform.*, vol. 13, no 2, p. 411-423. 2017.

[6] Fontaine, C., and Galand, F. A survey of homomorphic encryption for nonspecialists. *EURASIP J. Inform. Security*, no 1, p. 1-10. 2007.

[7] Cao, Z., and Liu, L. On the Disadvantages of Pairing-based Cryptography. *IACR Cryptology ePrint Archive*, 84. 2015.

[8] E. Mykletun, J. Girao, D. Westhoff, "Public Key Based Cryptoschemes for Data Concealment in Wireless Sensor Networks", *Proc. IEEE Int'l Conf. Comm. (ICC '06)*, vol. 5, 2006.

[9] D. Johnson, A. Menezes, and S. Vanstone, "The elliptic curve digital signature algorithm (ECDSA)," *Int. J. Inf. Security*, vol. 1, no. 1, pp. 36–63, 2001.

[10] Li, F., Luo, B., and Liu, P. Secure information aggregation for smart grids using homomorphic encryption. In. *IEEE SmartGridComm'10*, p. 327–332, 2010.

[11] Paillier, P. Public-key cryptosystems based on composite degree residuosity classes. In *Advances in Cryptology – EUROCRYPT 99*, p. 223–238, 1999.

[12] H. Li et al., "EPPDR: An efficient privacy-preserving demand response scheme with adaptive key evolution in smart grid," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 8, pp. 2053–2064, Aug. 2014.

[13] 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 Trans. Parallel Distrib. Syst.*, vol. 23, no. 9, pp. 1621–1631, Sep. 2012.

[14] C.-I. Fan, S.-Y. Huang and Y.-L. Lai, "Privacy-enhanced data aggregation scheme against internal attackers in smart grid", *IEEE Trans. Ind. Inform.*, vol. 10, no. 1, pp. 666-675, 2014.

[15] Fu, S., Ma, J., Li, H., and Jiang, Q. A robust and privacy-preserving aggregation scheme for secure smart grid communications in digital communities. *Sec. Comm. Networks*, vol. 15, no 9, p. 2779-2788. 2016.

[16] V. S. Miller, "Use of Elliptic Curves in Cryptography," H. C.Williams, Ed., Advances in Cryptology — *CRYPTO, LNCS,vol. 218, Springer-Verlag*, pp. 417–26, 1986.

[17] Rafik, M. B. O., and Mohammed, F. The impact of ECC's scalar multiplication on wireless sensor networks. In *IEEE 11th Int'l Symposium on Programming and Systems (ISPS),* p. 17-23. 2013.

[18] Bellare, M., Boldyreva, A., Kurosawa, K., and Staddon, J. Multi-recipient encryption schemes: How to save on bandwidth and computation without sacrificing security. *IEEE Trans. Inf. Theory*, *53*(11), 3927-3943, 2007.

[19] C. Gentry, Fully homomorphic encryption using ideal lattices. In *Symposium on the Theory of Computing (STOC)*, p. 169–178. 2009.

[20] Castelluccia, C., Chan, A. C., Mykletun, E., and Tsudik, G. Efficient and provably secure aggregation of encrypted data in wireless sensor networks. *ACM Trans. Sensor Networks*, vol. 5, no 3, p. 20. 2009.

[21] Chen, Lily. Recommendation for Key Derivation Using Pseudorandom Functions, *NIST Special Publication*, vol. 800, p. 108, 2008.

[22] Boudia, O. R. M., Senouci, S. M., and Feham, M. A novel secure aggregation scheme for wireless sensor networks using stateful public key cryptography. *Ad Hoc Networks*, vol. 32, p. 98-113. 2015.

[23] A. Antipa et al., Accelerated verification of ECDSA signatures. In *Selected Areas in Cryptography, Springer,* p. 307-318. August 2005.

[24] Pollard, J. M. Monte Carlo methods for index computation ($mod\backslash\ p$). *Mathematics of computation*, vol. 32, no. 143, p. 918-924. 1978.

[25] Certivox. Multiprecision integer and rational arithmetic c/c++ library (MIRACL), 2014, https://github.com/miracl/MIRACL

[26] B. Lynn, PBC Library http://crypto.stanford.edu/pbc/; pbc-0.5.14 (Released on Jun 14, 2013).

[27] Saputro, N., and Akkaya, K. Performance evaluation of smart grid data aggregation via homomorphic encryption. In *IEEE Wireless Communications and Networking Conference (WCNC)*, p. 2945-2950. 2012.