IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 4, 2019, accepted February 6, 2019, date of publication February 21, 2019, date of current version March 13, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2900934

EPDA: Efficient and Privacy-Preserving Data Collection and Access Control Scheme for Multi-Recipient AMI Networks

AHMAD ALSHARIF[1,2], (Member, IEEE), MAHMOUD NABIL[2], MOHAMED M. E. A. MAHMOUD[2], (Member, IEEE), AND MOHAMED ABDALLAH[3], (Senior Member, IEEE)

[1]Department of Computer Science, University of Central Arkansas, Conway, AR 72035, USA
[2]Department of Electrical and Computer Engineering, Tennessee Tech University, Cookeville, TN 38505, USA
[3]Information and Computing Technology Division, College of Science and Engineering, Hamad Bin Khalifa University, Doha, Qatar

Corresponding author: Mohamed Abdallah (moabdallah@hbku.edu.qa)

This work was supported by the Qatar National Research Fund (a member of Qatar Foundation) through NPRP under Grant NPRP10-1223-160045.

ABSTRACT Advanced metering infrastructure (AMI) networks allow the data collection of consumers' fine-grained power consumption data (PCD) to perform real-time monitoring and energy management. However, PCD can leak sensitive information about consumers' activities. Various privacy-preserving data collection schemes have been proposed for AMI networks to allow the collection of an aggregated PCD to preserve consumers' privacy. However, most of these schemes are designed for single-recipient AMI networks and cannot be used efficiently for multi-recipient AMI networks in which several entities should have access to the aggregated PCD of different sets of users for legitimate uses. In this paper, we propose an efficient and privacy-preserving data collection and access control scheme for multi-recipient AMI networks named EPDA. We developed a novel proxy re-encryption scheme that allows data aggregation before re-encryption and can allow either full or partial access to the aggregated data after re-encryption as needed. The proposed scheme can be used for fine-grained access control for multi-recipient AMI networks in which each recipient can access only the data intended to it. The EPDA uses lightweight operations in encryption, aggregation, and decryption which result in low computation and communication overheads. Our security analysis demonstrates that the EPDA is secure, can resist collusion attacks and hide customers' distribution which is needed for a fair electricity trade market. Our experimental results confirm that the EPDA has improved performance for the computational cost at each entity in the AMI network and low communication overhead.

INDEX TERMS Smart grid, AMI networks, privacy preservation, data aggregation, proxy re-encryption, fine-grained access control.

I. INTRODUCTION

Traditional power grids are obsolete and vulnerable to blackouts. Recent reports indicated that the power outages cost the United States (U.S.) at least 150 billion dollars each year [1]. Also, the north-east blackout in August 2003, which lasted for a week, affected over 100 power plants and about 55 million people [2]. Investigations showed that this blackout could be avoided if the grid could provide effective real-time diagnostic support [3].

The associate editor coordinating the review of this manuscript and approving it for publication was Ayaz Ahmad.

The smart grid (SG) is the next generation of the traditional power grid and the way for innovations in the electric sector [4], [5]. It uses information and communication technologies to provide two-way communications between the grid's entities to ensure the efficient and reliable operation of the grid [6]. Figure 1 shows the conceptual architecture of the SG. The figure shows several processes through which electricity is generated and transferred to electricity consumers, or simply ''users''. Electricity generation is the process of generating electric power from several source. Transmission is the bulk movement of electrical energy from generation sites to distribution substations through

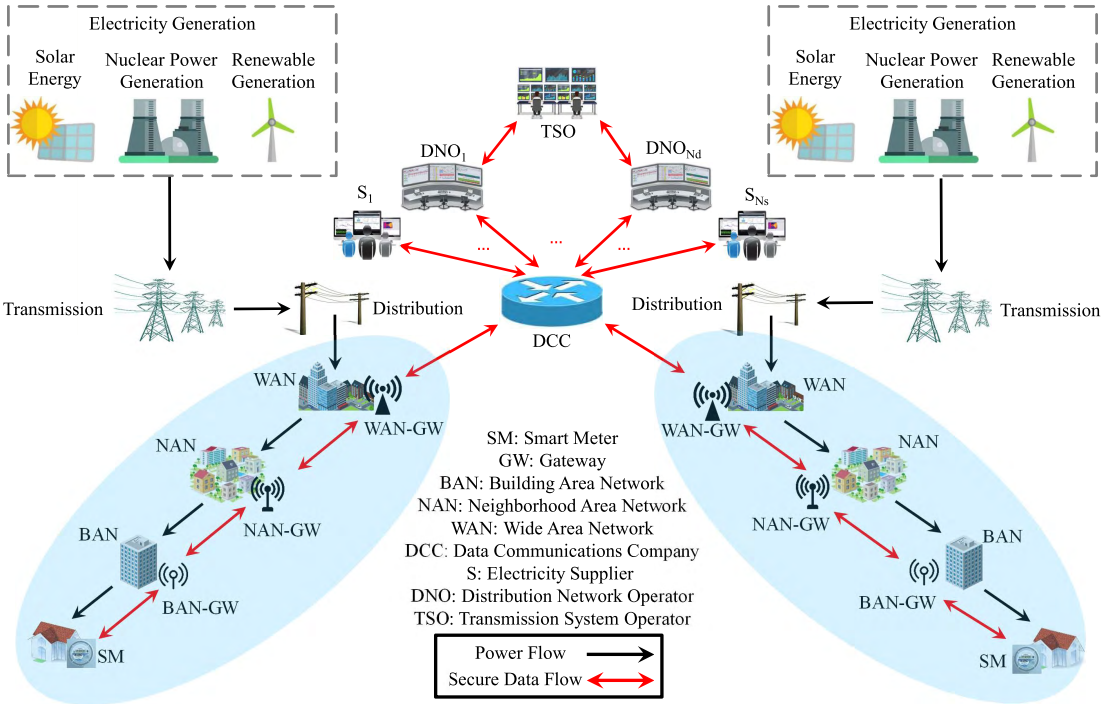

FIGURE 1. The smart grid conceptual architecture with multi-recipient AMI network [28].

transmission networks which are maintained by a transmission system operator (TSO). Distribution is the final stage in the delivery of electric power through distribution networks that carry electricity from distribution substations to individual users. These networks are operated and maintained by distribution network operators (DNOs).

One of the main components of the SG is the advanced metering infrastructure (AMI) networks that allow secure information flow and enable the automated collection of metering data [7]–[9]. Figure 1 shows an AMI network model in which smart meters (SMs) installed at users' sides are connected to grid operators through a hierarchical network structure that consists of building area networks (BANs), neighborhood area networks (NANs), wide area networks (WANs) and a data communications company (DCC). AMI networks allows the collection of fine-grained power consumption data (PCD) of electricity consumers at high rates, e.g., few minutes. Then, multi entities, e.g. grid operators and electricity suppliers, can analyze the collected PCD for real-time grid monitoring and energy management [10]–[13]. For example, fine-grained data analysis can be used for the reduction of the peak-to-average ratio, which can help in preventing brownouts, an electricity reduction in a particular area, and blackouts, a failure to supply electricity [14]. Also, fine-grained PCD are needed for real-time price-based demand/response programs in which electricity prices depends on the supply-to-demand ratio especially during peak hours [15], [16].

Despite the aforementioned benefits of fine-grained PCD collection, it creates serious privacy issues to users as these PCD can reveal users' daily activities. For example, non-intrusive load monitoring for PCD patterns can reveal sensitive information about users such as the times at which they leave/return homes, as well as, the appliances they use since each appliance has a unique power consumption signature [17]–[19]. According to the Electronic Privacy Information Center (EPIC), determining users' personal activities is a serious privacy concern in smart grid and thus fine-grained PCD should be protected from unauthorized access [20].

In order to preserve users' privacy, data aggregation has been widely used for AMI networks [21]–[30]. Specifically, users' PCD should be aggregated such that grid operators can only obtain an aggregated PCD (APCD) of a group of users in order to monitor and maintain the grid while preserving users' privacy. In addition, in order to prevent the intermediate nodes between users and the grid entities from accessing the individual PCD, several techniques can be used such as PCD masking [21]–[24] and homomorphic encryption [25]–[29].

One main limitation in most of the data collection schemes is that they are designed for single-recipient AMI network in which only a single entity should receive the APCD. A competitive electricity market, which is deployed in most European countries [31] and several states in the U.S. [32], allows energy deregulation, i.e., it allows electricity retailing through different electricity suppliers. Therefore, multiple entities, e.g. TSO, DNOs and electricity suppliers, should have access to the APCD of different sets of users for legitimate uses as will be explained in subsection III-C, and thus a data collection and access control scheme is needed for multi-recipient AMI network. As will be explained in section II,

if existing schemes, such as [21]–[27], when applied to the multi-recipient AMI networks are inefficient and not scalable. Moreover, they allow aggregators to learn the customers' distribution of each supplier within their areas, i.e., the number of users of each suppliers in each area. Aggregators can share customers' distribution of a supplier with other competitors such that they can modify their plans to attract competitors' customers. Such information should be hidden from competitors to ensure fair electricity trade market [33], [34].

In order to address the aforementioned limitations, we propose in this paper an **E**fficient and **P**rivacy-preserving **D**ata collection and **A**ccess control scheme for multi-recipient AMI network named ''EPDA''. In EPDA, each SM encrypts its PCD such that it can be aggregated only with other encrypted PCD (EPCD) intended for the same recipient. Then, to reduce communication overhead, all the ciphertexts are aggregated together into a single ciphertext instead of sending one aggregated ciphertext for each recipient in the network. Since all the APCD intended to all recipients are contained in a single aggregated ciphertext, we propose a novel data re-encryption scheme to allow fine-grained access control, i.e., each recipient can access only the APCD intended to it and cannot access the APCD intended for other recipients.

The novelty and contributions of this work can be summarized as follows.

1) We developed a novel proxy re-encryption scheme that allows data aggregation before re-encryption and can allow full or partial access to the aggregated data after re-encryption as needed. This scheme can be used to ensure fine-grained access control for multi-recipient AMI networks in which each recipient can access only the data intended to it.
2) EPDA provides higher privacy protection than the similar existing scheme [28], [29]. Specifically, EPDA not only protects the individual users' privacy by hiding their individual PCD, but also hides the customers' distribution of each supplier which cannot be achieved in [28] and [29].
3) In [28], electricity suppliers cannot ensure the correctness of the APCD they receive if the DNO colludes with the DCC. EPDA can resist this type of collusion attack.
4) Compared to [28] and [29], EPDA uses lightweight operations in encryption, aggregation and decryption which results in better performance than the existing schemes in terms of communication and computation overheads.

The remainder of this paper is organized as follows. Related works are discussed in Section II. The considered system models and the design requirements are presented in Section III. Preliminaries are given in Section IV. The proposed data collection and access control scheme is explained in Section V The security analysis and performance evaluation are given in Sections VI and VII, respectively. Conclusions are drawn in Section VIII. For better readability,

TABLE 1. Acronyms.

SG	Smart Grid	AMI	AMI Network
TA	Trusted Authority	TSO	Transmission System Operator
SM	Smart Meter	DNO	Distribution Network Operator
DCC	Data Communication Company	PCD	Power Consumption Data
GW	Gateway	EPCD	Encrypted PCD
BAN	Building Area Network	APCD	Aggregated PCD
NAN	Neighborhood Area Network	AEPCD	Aggregated Encrypted PCD
WAN	Wide Area Network		

Table 1 lists all the acronyms used in the paper. We used some acronyms similar to those used in [28].

II. RELATED WORKS

Several schemes have been proposed to ensure privacy-preserving data collection for AMI networks through data aggregation [21]–[30]. The schemes proposed in [21]–[24] uses one-time masking to mask PCD such that when the masked PCD are aggregated, masks cancel each other and an APCD can be obtained. These schemes differ in the way the secret masks are generated. The schemes proposed in [25]–[29] uses the additive homomorphic property of the Paillier cryptosystem [35] to preserve users' privacy. In these schemes, each user should encrypt his PCD using the Paillier cryptosystem, then an AEPCD can be computed in the ciphertext domain. Finally, the APCD can be recovered through a decryption process.

Most of the above data collection schemes are designed for single-recipient AMI network [21]–[27]. In order to apply them for the multi-recipient network, each meter needs to encrypt its PCD twice, one with the public key of its electricity supplier and the other with the public key of its DNO. Then, aggregators should aggregate the ciphertext intended to the same supplier/DNO together. However, this naive idea requires high communication overhead because the size of the aggregated ciphertexts increases linearly with the number of recipients in the network. Also, it imposes double the computation overhead for encryption. Therefore, they are inefficient and unscalable. Furthermore, the aggregators must learn the customers distribution of each supplier to achieve the functional requirement of the multi-recipient AMI network. This information can be misused if shared with other competitor suppliers as mentioned earlier.

Secure multi-party computation (SMC) based aggregation schemes were developed to achieve privacy through secret sharing and data aggregation [36]–[38]. The idea of SMC was introduced first in [39] where each node splits its data into k blocks such that the sum of all k blocks is equal to the node's data value. Then, it randomly selects $k-1$ other nodes and sends to each of them a distinctive block. The receiving nodes should aggregate the blocks they receive and transmit the result to the next node and so on. Conventionally, SMC-based schemes incur extremely high communication overhead and limit the network scalability due to transmitting shares to all other network nodes.

Few schemes addressed the data collection problem for multi-recipient [28], [29]. The scheme in [28] uses Paillier cryptosystem to encrypt users PCD only once with the public key of the intended DNO. In order to enable the DNO to distribute the aggregated data to the intended suppliers, SMs must send the ID of its supplier to the aggregators. For N_s suppliers in the network, aggregators must classify the received ciphertexts into N_s groups based on the suppliers' IDs, aggregate each group of ciphertext together, and then send N_s aggregated ciphertexts to each DNO. Finally, each DNO should decrypt the received N_s ciphertexts to obtain the APCD and then re-distribute these APCD to the suppliers. Similar idea can be used by [29] using attribute based encryption (ABE) [40] along with Paillier cryptosystem. Specifically, each SM encrypts its PCD using the Paillier cryptosystem with the public key of a trusted central node and creates an access policy to identify which entities can access its PCD. Then, EPCD with the same attributes in an access policy are aggregated together. Then central trusted node decrypts all the received Paillier ciphertexts and re-encrypts each APCD using ABE [40] based on the associated access policy to ensure access control. Finally, any entity that can satisfy that access policy of an ABE ciphertext can access the aggregated data.

These schemes suffer from the following limitations. First, since they allow aggregators to classify the ciphertexts, BAN-GWs, controlled by the DCC, can learn the customers' distribution of each supplier in each area. This information can be misused if shared with other competitor suppliers as mentioned earlier. On contrary, EPDA allows the aggregators to aggregate all the ciphertexts intended to all recipients into a single ciphertext without allowing aggregators to learn the customers' distribution of any supplier. In addition, sending several aggregated ciphertexts, one ciphertext for each supplier as in [28] or one for each attribute set as in [29], increases the communication overhead dramatically. Furthermore, in [28], if the DCC is an active adversary, it can collude with the DNO to deceive an electricity supplier such that the supplier pays the DNO higher distribution network fees. On contrary, this type of collusion is not possible in EPDA as will be explained in subsection VI-D.

In [30], Mustafa *et al.* proposed a privacy preserving protocol for PCD collection in multi-recipient AMI networks based on the combination of both multi party computation (MPC) and the linear secret sharing LSS. In [30], each SM splits its PCD into three shares using a linear secret sharing scheme and sends the three shares to three non-colluding DCC servers to carry out the aggregation process. Then, each server reports the aggregation result on the received shares to grid operators and suppliers. Finally, grid operators and suppliers combine the aggregated shares to obtain correct aggregation results. There are main differences between [30] and EPDA. First, in [30], since BAN-GWs relay the three shares to next gateways, they must be considered as trusted entities with respect to SMs, otherwise, they can learn the PCD of SMs by combining the three shares. Unlike that, EPDA considers the BAN-GWs as honest-but-curious entities. Second, EPDA allows decentralized aggregation as [28] and [29], i.e., PCD is being aggregated in a bottom-up manner by the different gateways as the data is transmitted to the DCC. On contrary in [30], the gateways only relay the users' shares to the DCC servers and then aggregation is done centrally at the DCC servers. Therefore, due to these differences between EPDA and [30], we will evaluate the performance of EPDA against [28] and [29].

III. SYSTEM MODELS

In this section, we describe the considered network and threat models. Also, we define the functional and security requirements. The main notations used in this paper are given Table 2.

TABLE 2. Main notations.

Notation	Description
$\mathbb{U}$	Set of users $\mathbb{U} = \{u_i, 1 \leq i \leq N_u\}$
$\mathbb{D}$	Set of DNOs $\mathbb{D} = \{D_j, 1 \leq j \leq N_d\}$
$\mathbb{S}$	Set of suppliers $\mathbb{S} = \{S_k, 1 \leq k \leq N_s\}$
u_i, D_j, S_k	ith user / jth DNO / kth supplier
$bg_\alpha / ng_\beta / wg_\gamma$	αth BAN-GW / βth NAN-GW / γth WAN-GW
$r_i^{j,k}$	PCD of u_i operated by D_j and supplied by S_k
$R^{j,k}$	APCD of set of users operated by D_j and supplied by S_k $\quad R^{j,k} = \sum_i r_i^{j,k}$
$R^{j,\mathbb{S}}$	APCD of set of users operated by D_j $R^{j,\mathbb{S}} = \sum_{k=1}^{N_s} R^{j,k}$
$R^{\mathbb{D},k}$	APCD of set of users supplied by S_k $R^{\mathbb{D},k} = \sum_{j=1}^{N_d} R^{j,k}$
$R^{\mathbb{D},\mathbb{S}}$	APCD of set of all users in the system $R^{\mathbb{D},\mathbb{S}} = \sum_{i=1}^{N_u} r_i^{j,k}$
$q, \mathbb{G}, \mathbb{G}_T, g, \hat{e}$	Bilinear pairing parameters
H	Hash Function $H : \{0,1\}^* \rightarrow \mathbb{G}$
x_ℓ	Private key of entity ℓ, used for signing
Y_ℓ	Public keys of entity ℓ, used for sig. verification
σ_ℓ	Signature generated by entity ℓ
ID_ℓ	Identity of entity ℓ
$\mathcal{MK}_1$	First master key set known only to the TA. $\mathcal{MK}_1 = \{M_1, M_2, N_1, N_2, N_3, N_4\}$
$\mathcal{MK}_2$	Second master key set known only to the TA. $\mathcal{MK}_2 = \{X_1, X_2, Y_1, Y_2, Y_3, Y_4\}$
$\mathcal{EK}_i$	Encryption key of user u_i derived from $\mathcal{MK}_1$
$\mathcal{RK}_k$	Re-encryption key given to the DCC for S_k derived from $\mathcal{MK}_1$ and $\mathcal{MK}_2$
$\mathcal{DK}_k$	Decryption key of S_k derived from $\mathcal{MK}_2$
P_i / C_i	Plaintext data vector / Ciphertext generated by u_i
$C_\alpha / C_\beta / C_\gamma$	Aggregated ciphertext of users under $bg_\alpha / ng_\beta / wg_\gamma$
C_{agg}	Aggregated ciphertext of all users in the system
$C_{agg}^{\mathbb{D},k}$	Re-encrypted aggregated ciphertext for S_k
$P_{agg}^{\mathbb{D},k}$	Decrypted ciphertext data vector for S_k
Q_k	Access control vector created by the TA for S_k
R_k	Decryption vector created by the TA for S_k

A. NETWORK MODEL

As shown in Figure 1, our network model considers a multi-recipients AMI network. In specific, the network model consists of the following entities.

- Transmission System Operator (TSO). It has the responsibility of balancing the entire grid. For example, National Grid Electricity Transmission plc (NGET) is the TSO for the Great Britain grid [41].
- Distribution Network Operators (DNOs). We consider a set of DNO companies, $\mathbb{D} = \{D_j, 1 \leq j \leq N_d\}$. Each D_j is licensed to distribute electricity in a particular geographic area j. Also, DNOs charge electricity suppliers distribution fees to transport electricity to users.
- Electricity Suppliers. We consider a set of electricity supplier companies, $\mathbb{S} = \{S_k, 1 \leq k \leq N_s\}$. Each S_k is responsible for supplying electricity to its users who may be located at different DNOs coverage areas.
- Users. We consider a set of users $\mathbb{U} = \{u_i, 1 \leq i \leq N_u\}$. Users can choose only one electricity supplier from the suppliers' set $\mathbb{S}$ and can change from one supplier to another at any time. Each user is equipped with a SM to encrypt its PCD, $r_i^{j,k}$, and report the EPCD to the DCC through networking facilities.
- Data Communication Company (DCC). It has the responsibility of delivering users' AEPCD to each DNO, and each supplier.
- Networking Facilities. They form a hierarchical network structure to connect SMs at users' side to the DCC through a BAN-GW, a NAN-GW, and a WAN-GW as shown in Fig. Figure 1.

B. THREAT MODEL

Users are considered honest-but-curious. They will correctly report their PCD to their intended recipients, however, they may try to learn individual's PCD of other users, e.g., their neighbors. DNOs and suppliers are also considered honest-but-curious. They may try to learn individual's PCD. In additions, DNOs may try to learn the APCD of any group of users located at other DNOs areas, whereas suppliers may try to learn the APCD of any group of users supplied by other competitor suppliers. Moreover, a supplier may try to learn the customer distribution of other suppliers in any area so that it can customize offers to attract competitors' customers. The DCC and the gateways are honest-but-curious. They follow the proposed scheme, but they may try to learn the PCD of any individual user, the APCD of any group of users within a DNO area, and/or the APCD of any group of users supplied by a specific supplier. Most importantly, they may try to learn the customers' distribution of each supplier within their areas. In addition, suppliers and DNOs may collude with the DCC to launch successful attacks.

C. DESIGN REQUIREMENTS

Based on the multi-recipient AMI network objectives and the aforementioned threat model, the proposed scheme should achieve the following functional and security requirements.

1) FUNCTIONAL REQUIREMENTS

For multi-recipient AMI network, the following functional requirements should be met at each reporting period [28].

(F1) Each DNO D_j should have access to
  (a) $R^{j,k} = \sum_i r_i^{j,k}$ for $1 \leq k \leq N_s$, which is the APCD of the users supplied by supplier S_k in its area j. This is needed so that D_j can split the distribution network fees fairly between the set of suppliers.
  (b) $R^{j,\mathbb{S}} = \sum_{k=1}^{N_s} R^{j,k}$ which is the APCD of all users in the jth area in order that D_j can monitor and manage its distribution network.

(F2) Each supplier S_k should have access to
  (a) $R^{j,k} = \sum_i r_i^{j,k}$ for $1 \leq j \leq N_d$, which is the APCD of the users in each area j so that S_k can be assured that it pays the correct distribution network fees to each DNO D_j, i.e., it is not over/under charged.
  (b) $R^{\mathbb{D},k} = \sum_{j=1}^{N_d} R^{j,k}$ which is the APCD of the users supplied by S_k in all areas in order that S_k can accurately predict customers' demands to avoid any potential imbalance penalties.

(F3) In order to balance the grid efficiently, the TSO should have access to
  (a) $R^{j,\mathbb{S}}$ for each area j.
  (b) $R^{\mathbb{D},\mathbb{S}} = \sum_{i=1}^{N_u} r_i^{j,k}$ which is the APCD of all users in the system.

2) SECURITY REQUIREMENTS

At each reporting period, the following security requirements should be met.

(S1) User privacy preservation. No entity should be able to access the PCD, $r_i^{j,k}$, of any individual user u_i.

(S2) Confidentiality of aggregated data and access control. Each recipient should be able to access only the APCD intended to it and should not be able to access APCD of other recipients.

(S3) Confidentiality of customer distribution of each supplier. External entities, DNOs, electricity suppliers, should not be able to learn the customers' distribution of other suppliers for fair electricity market.

IV. PRELIMINARIES

A. k-NEAREST NEIGHBOR ENCRYPTION TECHNIQUE

Secure computation over encrypted data using the k-nearest neighbor (kNN) similarity measurement has been widely used in several applications such as keyword searching [42]–[49], and location-based applications [50]–[54]. Based on, but not limited to, the kNN similarity measurement, we develop EPDA that allows data re-encryption to allow fine-grained access control. The schemes proposed in [42]–[53] allows only secure dot product computation between two data vectors without revealing the content of the two vectors to ensure data confidentiality. Different from these schemes, we allow in EPDA secure computation of element-wise multiplication of more than two vectors.

This can be used as a proxy re-encryption scheme that can allow full or partial data access after re-encryption as need.

B. BILINEAR PAIRING BASED AGGREGATE SIGNATURE

Let $\mathbb{G}$ be a multiplicative cyclic group of prime order q, g be a generator of $\mathbb{G}$, and $\mathbb{G}_T$ be a multiplicative cyclic group of the same prime order q. Suppose a computable bilinear pairing $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$. with the following properties:

- *Bilinearity*: $\hat{e}(g^a, g^b) = \hat{e}(g, g^{ab}) = \hat{e}(g^{ab}, g) = \hat{e}(g, g)^{ab} \in \mathbb{G}_T \; \forall \; a, b \in \mathbb{Z}_q^*$.
- *Non-degeneracy*: $\hat{e}(g, g) \neq 1_{\mathbb{G}_T}$.

Based on this bilinear pairing, an aggregate signature scheme [55] can be constructed such that multiple signatures computed on different messages by different users, an aggregate signature can be easily computed and verified in a batch way. The scheme employs a hash function H defined as $H : \{0, 1\}^* \rightarrow \mathbb{G}$

1) *Key Generation*. For a particular user u_i, pick a random number $x_i \in \mathbb{Z}_q^*$ as a user's secret key, and compute $v_i = g^{x_i} \in \mathbb{G}$ as the corresponding public key.
2) *Signing*. Given a message $m_i \in \{0, 1\}^*$, a user u_i uses his secret key x_i to compute a signature $\sigma_i \in \mathbb{G}$ on m_i as $\sigma_i = (H(m_i))^{x_i}$
3) *Verification*. Given a message m_i, a signature σ_i, and a userâĂŹs public key v_i, a verifier accepts m_i if the signature is verified iff $e(\sigma_i, g) = e(H(m_i), v_i)$ holds.
4) *Aggregation*. For a group of users $u_i \in U$ and i from 1 to $|U|$ providing individual signatures, an aggregate signature can be computed as $\sigma_{agg} = \prod \sigma_i$.
5) *Aggregate Verification*. For a group of users, given individual distinct messages m_i, users' signatures σ_i, and the users' public keys v_i, the signatures can be batch verified to accept all messages if $e(\prod \sigma_i, g) = \prod e(H(m_i), v_i)$ holds.

V. THE PROPOSED SCHEME

A. OVERVIEW

Figure 2 shows an overview of the information flow in EPDA. First, users encrypt their PCD and send EPCD to their BAN-GWs as shown in the figure. Then, all the gateways perform data aggregation process until the AEPCD reaches the DCC. The DCC performs the re-encryption process to ensure fine-grained access control and then distribute the re-encrypted ciphertexts, one to each DNO and supplier. Finally, each DNO and supplier performs a single decryption process to recover the APCD intended to them without revealing APCD intended to other recipients. The details of these phases are explained in subsection V-C, subsection V-D, subsection V-E, and subsection V-F, respectively.

Figure 3, shows an example of the data reported and the access control process. As shown in the figure, a plaintext vector of size $v = N_d \times N_s$ for the multi-recipient AMI network should be used by all users. Each element in the plaintext vector is assigned to a specific DNO and a specific supplier at the same time. As shown in the figure, user u_i

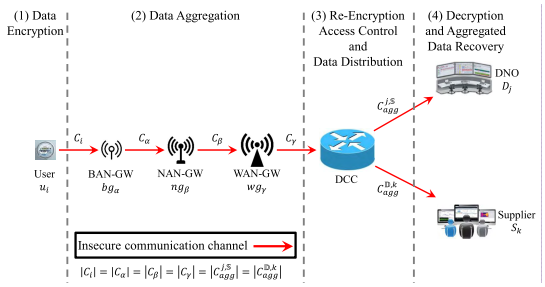

FIGURE 2. An overview of the information flow in EPDA.

in the area of D_j and supplied by S_k builds its plaintext data vector, P_i, by reporting its PCD, $r_i^{j,k}$, in the location intended for both D_j and S_k, and sets all other locations in the data vector to zeros. Then, P_i is encrypted and aggregated with other users' encrypted vectors to produce an aggregated encrypted vector, C_{agg}, which is the encryption of P_{agg} as will be explained later in this section. No entity should be able to decrypt C_{agg} to obtain P_{agg} since P_{agg} contains all the APCD intended for all entities in the network. Therefore, access control process is required such that each recipient should be able to access only the data intended to it. Access control is achieved by the proposed secure element-wise multiplication between P_{agg} and the access control vector Q_k as will be explained later in the re-encryption process give in subsection V-E. For instance, the figure shows the access control process for S_k in which all the APCD should be hidden except the data intended to S_k. As shown in the figure, the element-wise product between P_{agg} and Q_k contains only the data intended to S_k. Therefore, no entity except S_k should be able to get this result. This can be achieved by another element-wise multiplication process by a decryption vector R_k, owned by a recipient S_k, such that S_k is the only entity that can obtain P_{agg}^k as will be explained in the decryption process given in subsection V-F. It should be noted that, the access control vector Q_k and the decryption vector R_k are the same for the same recipient S_k. Also, each other supplier and DNO will have a unique access control vector and a decryption vector to achieve the functional and security requirements defined earlier.

Adding or removing recipients after system initialization can be done as follows. In case a recipient leaves the system, the consumers of that recipient will use the same vector structure but report their PCD in the element corresponding to their new recipient. In addition, during system initialization, several redundant elements can be added to the data vector for future use, e.g., in case a new recipient is to be added to the system. Moreover, like any secure communication protocol, the keys should be frequently updated for better security. Therefore, during key updates, the unused elements corresponding to the removed recipients can be removed, or if the number of recipients to be added is greater than the redundant elements, the keys should be updated. It should be noted that, adding or removing a recipient is not a very frequent event

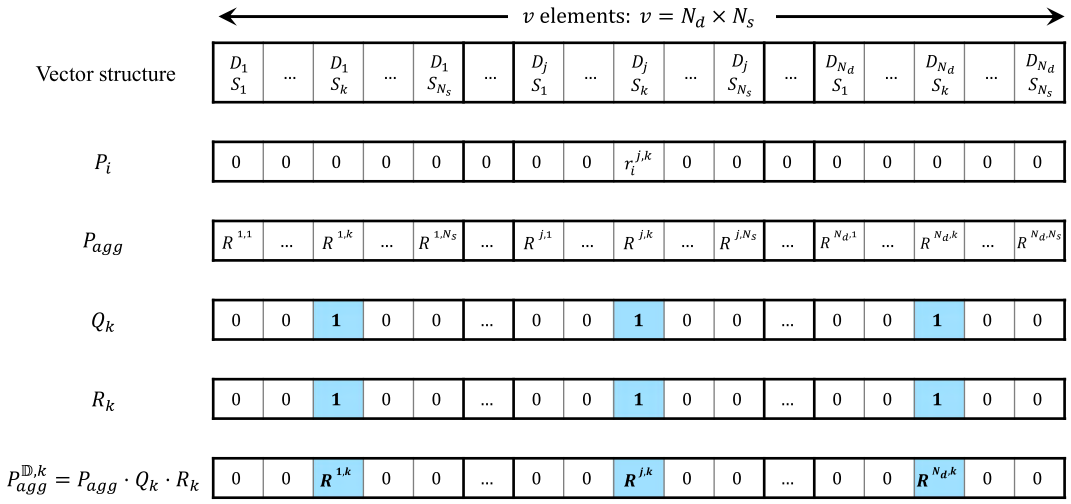

FIGURE 3. Aggregation and access control process using element-wise vectors product.

and thus it will be rare that the keys are updated to add a new recipient after consuming all the elements allocated for future use.

B. SYSTEM SETUP

An offline trusted authority (TA) is needed to setup the system. System setup consists of the following phases (1) generation of public system parameters, (2) generation of private/public pairs to be used in signing/verifying the exchanged messages and (3) generation of the kNN encryption, re-encryption, and decryption keys to be used by users, DCC, and recipients, respectively.

1) GENERATION OF PUBLIC SYSTEM PARAMETER

The TA should generate the bilinear pairing parameters $(q, \mathbb{G}, \mathbb{G}_T, g, \hat{e})$ and chooses a hash function $H : \{0,1\}^* \rightarrow \mathbb{G}$. Then, it publishes the system public parameters as $pubs = \{q, \mathbb{G}, \mathbb{G}_T, g, \hat{e}, H\}$.

2) GENERATION OF PUBLIC/PRIVATE KEY PAIRS

Each user u_i chooses a secret key $x_i \in \mathbb{Z}_q^*$ and computes the corresponding public key $Y_i = g^{x_i} \in \mathbb{G}$. Similarly, each BAN-GW bg_α, each NAN-GW ng_β, each WAN-GW wg_γ, and the DCC generate public/private key pairs (x_α/Y_α), (x_β/Y_β), (x_γ/Y_γ), and (x_{dcc}/Y_{dcc}), respectively. Each user and the DCC should obtain a digital certificate from the TA to certify their public keys, while the DCC should generate a certificates for each GW.

3) GENERATION OF KNN ENCRYPTION, RE-ENCRYPTION AND DECRYPTION KEYS

The TA generates a random vector S to be used as a splitting indicator for the kNN encryption technique. The size of S is $v = N_d \times N_s$. Each element in S is either 0, 1, or 2. Then, the TA generates two master key sets, $\mathcal{MK}_1 = \{M_1, M_2, N_1, N_2, N_3, N_4\}$ and $\mathcal{MK}_2 = \{X_1, X_2, Y_1, Y_2, Y_3, Y_4\}$, where each element in the key sets is a $v \times v$ invertible random matrix. $\mathcal{MK}_1$ is used to derive a unique encryption key for each user, both sets are used to generate a unique re-encryption and access control key for each DNO/supplier, and $\mathcal{MK}_2$ is used to derive a unique decryption key for each DNO/supplier.

a: GENERATION OF USERS' ENCRYPTION KEYS

For each user u_i, the TA uses $\mathcal{MK}_1$ to generate a unique encryption key $\mathcal{EK}_i$ which consists of 4 parts $\{\mathcal{EK}_{i1}, \ldots, \mathcal{EK}_{i4}\}$ as $\mathcal{EK}_i = \{a_i N_1, b_i N_2, c_i N_3, d_i N_4\}$, where a_i, b_i, c_i, and d_i are $v \times v$ invertible random matrices such that $a_i + b_i = M_1$ and $c_i + d_i = M_2$. Finally, the TA should send $\mathcal{EK}_i$ to user u_i via a secure channel. It should be noted that whenever a user joins/leaves the system or even changes his electricity supplier, he can use the same encryption key and does not need to receive a new one.

b: GENERATION OF RE-ENCRYPTION KEYS

Using $\mathcal{MK}_1$ and $\mathcal{MK}_2$, the TA generates $(N_d + N_s)$ re-encryption keys, one for each DNO and for each supplier, and send them to the DCC. Re-encryption key for supplier S_k is computed through the following steps. First, the access control binary vector Q_k is created by setting the bit locations for S_k to ones, i.e., setting the locations for D_j, S_k for all $1 \leq j \leq N_d$, and then, all other bits are set to zeros. An example of Q_k is shown in Figure 3. Then, Q_k is split into two vectors q'_k and q''_k as follows. For each element z, $1 \leq z \leq v$ in the splitting indicator S, if $S(z)$ is 1, then $q'_k(z)$ and $q''_k(z)$ are set to two random numbers such that $q'_k(z) + q''_k(z) = Q_k(z)$, while if $S(z)$ is zero or two, then $q'_k(z)$ and $q''_k(z)$ are set equal to $Q_k(z)$. Then, q'_k and q''_k are extended to two symmetric square diagonal matrices $\hat{q}_k$ and $\ddot{q}_k$ respectively. This extension will allow the secure element-wise multiplication of the data vectors instead of obtaining the dot product as in previous kNN-based schemes [42]–[53]. Finally, the re-encryption and access control key $\mathcal{RK}_k$ for S_k is

computed as

$$\mathcal{RK}_k = \begin{Bmatrix} N_1^{-1}M_1^{-1}\hat{q}_k X_1 Y_1 \\ N_2^{-1}M_1^{-1}\hat{q}_k X_1 Y_1 \\ N_1^{-1}M_1^{-1}\hat{q}_k X_1 Y_2 \\ N_2^{-1}M_1^{-1}\hat{q}_k X_1 Y_2 \\ N_3^{-1}M_2^{-1}\ddot{q}_k X_2 Y_3 \\ N_4^{-1}M_2^{-1}\ddot{q}_k X_2 Y_3 \\ N_3^{-1}M_2^{-1}\ddot{q}_k X_2 Y_4 \\ N_4^{-1}M_2^{-1}\ddot{q}_k X_2 Y_4 \end{Bmatrix} \equiv \begin{Bmatrix} \mathcal{RK}_{k1} \\ \mathcal{RK}_{k2} \\ \mathcal{RK}_{k3} \\ \mathcal{RK}_{k4} \\ \mathcal{RK}_{k5} \\ \mathcal{RK}_{k6} \\ \mathcal{RK}_{k7} \\ \mathcal{RK}_{k8} \end{Bmatrix}$$

where $\mathcal{RK}_k$ consists of 8 parts $\{\mathcal{RK}_{k1}, \ldots, \mathcal{RK}_{k8}\}$, and each part is a square matrix of size $v \times v$.

In a similar process, a re-encryption and access control key $\mathcal{RK}_j$ should be computed for each DNO D_j. The only difference is that the access control binary vector Q_j is created by setting the bit locations intended for D_j to ones, i.e., locations for D_j, S_k for $1 \leq k \leq N_s$ are set to ones, and all other bits are set to zeros. Finally, the TA should send all the re-encryption keys to the DCC via a secure channel.

c: GENERATION OF DECRYPTION KEYS

Using $\mathcal{MK}_2$, the TA generates a decryption key for each DNO and supplier in the system. Decryption key for S_k is computed through the following steps. First, a decryption binary vector R_k is created exactly as the access control vector Q_k. Then, R_k is split into two vectors r'_k and r''_k as follows. For each element z in S, if $S(z)$ is two, then $r'_k(z)$ and $r''_k(z)$ are set to two random numbers such $r'_k(z) + r''_k(z) = R_k(z)$, while if $S(z)$ is zero or one, then $r'_k(z)$ and $r''_k(z)$ are set similar to $R_k(z)$. Then, r'_k and r''_k are extended into two symmetric square diagonal matrices $\hat{r}_k$ and $\ddot{r}_k$ respectively. Finally, the decryption key $\mathcal{DK}_k$ for S_k is computed as

$$\mathcal{DK}_k = \begin{Bmatrix} Y_1^{-1}e_k\hat{r}_k \\ Y_2^{-1}f_k\hat{r}_k \\ Y_3^{-1}g_k\ddot{r}_k \\ Y_4^{-1}h_k\ddot{r}_k \end{Bmatrix} \equiv \begin{Bmatrix} \mathcal{DK}_{k1} \\ \mathcal{DK}_{k2} \\ \mathcal{DK}_{k3} \\ \mathcal{DK}_{k4} \end{Bmatrix}$$

$\mathcal{DK}_k$ consists of 4 parts $\{\mathcal{DK}_{k1}, \ldots, \mathcal{DK}_{k4}\}$, and each part is a square matrix of size $v \times v$. Also, e_k, f_k, g_k, and h_k are $v \times v$ invertible random matrices such that $e_k + f_k = X_1^{-1}$ and $g_k + h_k = X_2^{-1}$. In a similar process, decryption key $\mathcal{DK}_j$ for each DNO D_j should be created using a decryption vector R_j. Finally, the TA should send each decryption key to its corresponding recipient via a secure channel.

C. USERS: DATA ENCRYPTION

At each reporting period, each user u_i should report its EPCD to its local BAN-GW bg_α by performing the following steps.

- Step 1: Build a plaintext data vector P_i by placing the PCD, $r_i^{j,k}$, at the location in the data vector for D_j and S_k and set all other elements to zeros.
- Step 2: Split P_i into two random vectors p'_i and p''_i as follows. For each element z in S, if $S(z)$ is zero, then $p'_i(z)$ and $p''_i(z)$ are set to two random numbers such that $p'_i(z) + p''_i(z) = P_i(z)$, while if $S(z)$ is one or two, then $p'_i(z)$ and $p''_i(z)$ are set equal to $P_i(z)$.
- Step 3: Generate a ciphertext C_i using p'_i, p''_i and the encryption key $\mathcal{EK}_i$ as

$$C_i = \{p'_i a_i N_1,\ p'_i b_i N_2,\ p''_i c_i N_3,\ p''_i d_i N_4\}$$

where C_i is a row vector of size $1 \times 4v$.
- Step 4: Use its private key x_i to generate a signature σ_i on C_i

$$\sigma_i = \Big(H(C_i \parallel \mathrm{ID}_{u_i} \parallel \mathrm{ID}_{bg_\alpha} \parallel \mathrm{TS})\Big)^{x_i}$$

where TS is a timestamp.
- Step 5: Report to bg_α the following message

$$C_i \parallel \mathrm{ID}_{u_i} \parallel \mathrm{ID}_{bg_\alpha} \parallel \mathrm{TS} \parallel \sigma_i$$

D. GATEWAYS: EFFICIENT AND DECENTRALIZED AGGREGATION

After collecting N_α reports from N_α users, each BAN-GW bg_α should verify the received signatures, aggregate the received ciphertexts into a single one, and send a message to its NAN-GW ng_β by performing the following steps

- Step 1: Check the freshness of the timestamps to thwart replay attacks.
- Step 2: Verify the received signatures to ensure reports' integrity and the authenticity of reports' senders. Efficient batch verification process can be done by checking

$$\hat{e}\Big(\prod_{i=1}^{N_\alpha}\sigma_i, g\Big) \stackrel{?}{=} \prod_{i=1}^{N_\alpha}\hat{e}\Big(H(C_i \parallel \mathrm{ID}_{u_i} \parallel \mathrm{ID}_{bg_\alpha} \parallel \mathrm{TS}), Y_i\Big)$$

- Step 3: Compute the aggregated ciphertext C_α as

$$C_\alpha = \sum_{i=1}^{N_\alpha} C_i = \begin{Bmatrix} \sum_{i=1}^{N_\alpha} p'_i a_i N_1, & \sum_{i=1}^{N_\alpha} p'_i b_i N_2, \\ \sum_{i=1}^{N_\alpha} p''_i c_i N_3, & \sum_{i=1}^{N_\alpha} p''_i d_i N_4 \end{Bmatrix}$$

where C_α is a row vector of size $1 \times 4v$.
- Step 4: Use its private key x_α to compute the signature

$$\sigma_\alpha = \Big(H(C_\alpha \parallel \mathrm{ID}_{bg_\alpha} \parallel \mathrm{ID}_{ng_\beta} \parallel \mathrm{TS})\Big)^{x_\alpha}$$

- Step 5: Report to its NAN-GW ng_β the following message

$$C_\alpha \parallel \mathrm{ID}_{bg_\alpha} \parallel \mathrm{ID}_{ng_\beta} \parallel \mathrm{TS} \parallel \sigma_\alpha$$

The operations done by each NAN-GW ng_β and each WAN-GW wg_γ are exactly the same as those done by each BAN-GW bg_α. Finally, each WAN-GW wg_γ should send to the DCC the following message

$$C_\gamma \parallel \mathrm{ID}_{wg_\gamma} \parallel \mathrm{ID}_{dcc} \parallel \mathrm{TS} \parallel \sigma_\gamma$$

E. DCC: AGGREGATION, RE-ENCRYPTION AND ACCESS CONTROL

After collecting N reports from N WAN-GWs, the DCC should verify the received signatures, aggregate the received ciphertexts, re-encrypt the aggregated ciphertext such that each DNO and each supplier should access only the data intended to it, i.e., re-encryption is done to achieve access control, and then forward each re-encrypted ciphertext to its intended DNO/supplier. The DCC should carry out the following steps

- Step 1: Check the freshness of the timestamps.
- Step 2: Perform batch signature verification process to verify the received signatures by checking

$$\hat{e}\Big(\prod_{\gamma=1}^{N}\sigma_\gamma, g\Big) \stackrel{?}{=} \prod_{\gamma=1}^{N}\hat{e}\Big(H(C_\gamma \parallel \mathrm{ID}_{wg_\gamma} \parallel \mathrm{ID}_{dcc} \parallel \mathrm{TS}), Y_\gamma\Big)$$

- Step 3: Compute the encrypted aggregated ciphertext C_{agg} for all the N_u users as

$$\begin{aligned} C_{agg} &= \sum_{\gamma=1}^{N} C_\gamma = \{C_{a1}, C_{a2}, C_{a3}, C_{a4}\} \\ &= \begin{Bmatrix} \sum_{i=1}^{N_u} p'_i a_i N_1, & \sum_{i=1}^{N_u} p'_i b_i N_2, \\ \sum_{i=1}^{N_u} p''_i c_i N_3, & \sum_{i=1}^{N_u} p''_i d_i N_4 \end{Bmatrix} \end{aligned}$$

  where C_{agg} is a row vector of size $1 \times 4v$.
- Step 4: Re-encrypt C_{agg} to generate $C^{\mathbb{D},k}$, where $C^{\mathbb{D},k}$ is the AEPCD intended to supplier S_k for the set of all DNOs $\mathbb{D}$ in all areas. Re-encryption and access control are done using the re-encryption and access control key $\mathcal{RK}_k$ through the following operation

$$\begin{aligned} C^{\mathbb{D},k} &= \begin{Bmatrix} C_{a1}\mathcal{RK}_{k1} + C_{a2}\mathcal{RK}_{k2} \\ C_{a1}\mathcal{RK}_{k3} + C_{a2}\mathcal{RK}_{k4} \\ C_{a3}\mathcal{RK}_{k5} + C_{a4}\mathcal{RK}_{k6} \\ C_{a3}\mathcal{RK}_{k7} + C_{a4}\mathcal{RK}_{k8} \end{Bmatrix}^T \\ &= \begin{Bmatrix} p'_a\hat{q}_k X_1 Y_1 \\ p'_a\hat{q}_k X_1 Y_2 \\ p''_a\ddot{q}_k X_2 Y_3 \\ p''_a\ddot{q}_k X_2 Y_4 \end{Bmatrix}^T \equiv \begin{Bmatrix} C_1^{\mathbb{D},k} \\ C_2^{\mathbb{D},k} \\ C_3^{\mathbb{D},k} \\ C_4^{\mathbb{D},k} \end{Bmatrix}^T \end{aligned} \quad (1)$$

  where $C^{\mathbb{D},k}$ is a row vector of size $1 \times 4v$ and $p_a = \sum_{i=1}^{N_u} p'_i$. The correctness proof of the first component of $C^{\mathbb{D},k}$ is as follows

$$\begin{aligned} C_1^{\mathbb{D},k} &= C_{a1}\mathcal{RK}_{k1} + C_{a2}\mathcal{RK}_{k2} \\ &= \sum_{i=1}^{N_u} p'_i a_i N_1 N_1^{-1} M_1^{-1}\hat{q}_k X_1 Y_1 \\ &\quad + \sum_{i=1}^{N_u} p'_i b_i N_2\, N_2^{-1} M_1^{-1}\hat{q}_k X_1 Y_1 \\ &= \sum_{i=1}^{N_u} p'_i a_i M_1^{-1}\hat{q}_k X_1 Y_1 + \sum_{i=1}^{N_u} p'_i b_i M_1^{-1}\hat{q}_k X_1 Y_1 \\ &= \sum_{i=1}^{N_u} p'_i (a_i + b_i) M_1^{-1}\hat{q}_k X_1 Y_1 \\ &= \sum_{i=1}^{N_u} p'_i M_1 M_1^{-1}\hat{q}_k X_1 Y_1 = \sum_{i=1}^{N_u} p'_i \hat{q}_k X_1\ Y_1 \\ &= p'_a\hat{q}_k X_1\ Y_1 \end{aligned}$$

  Similarly, we can prove the correctness of the other components of $C^{\mathbb{D},k}$. As shown in Equation 1, the result of the re-encryption process is a new ciphertext encrypted under the key set $\mathcal{MK}_2$ instead of $\mathcal{MK}_1$. In addition, $C^{\mathbb{D},k}$ represents the encryption of the element-wise product between the access control vector Q_k and $P_{agg} = \sum_{i=1}^{N_u} P_i$, which is the aggregation of all plaintext data vectors, i.e. S_k can access only the aggregated data in the locations specified by the TA while creating the access control vector Q_k.
- Step 5: Use its private key x_{dcc} to compute the signature

$$\sigma_{dcc} = \Big(H\Big(C^{\mathbb{D},k} \parallel \mathrm{ID}_{dcc} \parallel \mathrm{ID}_{S_k} \parallel \mathrm{TS}\Big)\Big)^{x_{dcc}}$$

- Step 6: Forward to S_k the following message

$$C^{\mathbb{D},k} \parallel \mathrm{ID}_{dcc} \parallel \mathrm{ID}_{S_k} \parallel \mathrm{TS} \parallel \sigma_{dcc}$$

  The DCC should repeat steps 4, 5, and 6 for each DNO D_j and each supplier S_k by utilizing the corresponding re-encryption keys $\mathcal{RK}_j$ and $\mathcal{RK}_k$ respectively.

F. DNO AND SUPPLIER: DECRYPTION AND AGGREGATED DATA RECOVERY

Supplier S_k should verify the received signature from the DCC, and decrypt the received ciphertext $C^{\mathbb{D},k}$ to obtain $P_{agg}^{\mathbb{D},k}$ by performing the following steps.

- Step 1: Check the freshness of the timestamp.
- Step 2: Verify the DCC signature by checking

$$\hat{e}\,(\sigma_{dcc}, g) \stackrel{?}{=} \hat{e}\Big(H\Big(C^{\mathbb{D},k} \parallel \mathrm{ID}_{dcc} \parallel \mathrm{ID}_{S_k} \parallel \mathrm{TS}\Big), Y_{dcc}\Big)$$

- Step 3: Decrypt $C^{\mathbb{D},k}$ to obtain $P_{agg}^{\mathbb{D},k}$ using the decryption key $\mathcal{DK}_k$ through the following operation

$$\begin{aligned} P_{agg}^{\mathbb{D},k} &= C_1^{\mathbb{D},k}\mathcal{DK}_{k1} + C_2^{\mathbb{D},k}\mathcal{DK}_{k2} \\ &\quad + C_3^{\mathbb{D},k}\mathcal{DK}_{k3} + C_4^{\mathbb{D},k}\mathcal{DK}_{k4} \\ &\equiv P_{agg} \cdot Q_k \cdot R_k \end{aligned} \quad (2)$$

  where $\cdot$ is the element-wise product of the three vectors P_{agg}, Q_k, R_k. The correctness proof of Equation 2 is as follows

$$\begin{aligned} P_{agg}^{\mathbb{D},k} &= C_1^{\mathbb{D},k}\mathcal{DK}_{k1} + C_2^{\mathbb{D},k}\mathcal{DK}_{k2} \\ &\quad + C_3^{\mathbb{D},k}\mathcal{DK}_{k3} + C_4^{\mathbb{D},k}\mathcal{DK}_{k4} \\ &= p'_a\hat{q}_k X_1 Y_1 Y_1^{-1} e_k \hat{r}_k + p'_a\hat{q}_k X_1 Y_2 Y_2^{-1} f_k \hat{r}_k \\ &\quad + p''_a\ddot{q}_k X_2 Y_3 Y_3^{-1} g_k \ddot{r}_k + p''_a\ddot{q}_k X_2 Y_4 Y_4^{-1} h_k \ddot{r}_k \end{aligned}$$

$$
\begin{aligned}
&= p'_a \hat{q}_k X_1 e_k \hat{r}_k + p'_a \hat{q}_k X_1 f_k \hat{r}_k \\
&\quad + p''_a \ddot{q}_k X_2 g_k \ddot{r}_k + p''_a \ddot{q}_k X_2 h_k \ddot{r}_k \\
&= p'_a \hat{q}_k X_1 (e_k + f_k) \hat{r}_k + p''_a \ddot{q}_k X_2 (g_k + h_k) \ddot{r}_k \\
&= p'_a \hat{q}_k \hat{r}_k + p''_a \ddot{q}_k \ddot{r}_k \\
&\equiv P_{agg} \cdot Q_k \cdot R_k
\end{aligned}
$$

The result of the decryption process, $P_{agg}^{\mathbb{D},k}$ as shown in Figure 3, is a vector in which the N_d locations intended for S_k contain the APCD $R^{j,k}$ for each area j, $1 \leq j \leq N_d$ which satisfies the functional requirement (F2a). In addition, S_k can easily compute the total power consumption by all its customer in all areas $\sum r_i^{\mathbb{D},k}$ as $\sum r_i^{\mathbb{D},k} = \sum_{j=1}^{N_d} \sum r_i^{j,k}$ which satisfies the functional requirement (F2b).

In a similar process, each DNO D_j uses its decryption key $\mathcal{DK}_j$ to obtain $P_{agg}^{j,\mathbb{S}}$ which is a vector in which the N_s locations intended for D_j contain the aggregated data $\sum_i r_i^{j,k}$ for each supplier S_k, $1 \leq k \leq N_s$ which satisfies the functional requirement (F1a). Also, D_j can compute the total power consumption in its region $R^{j,\mathbb{S}}$ as $R^{j,\mathbb{S}} = \sum_{k=1}^{N_s} R^{j,k}$ which satisfies the functional requirement (F1b).

Finally, the TSO should receive $R^{j,\mathbb{S}} = \sum r_i^{j,\mathbb{S}}$ from each DNO D_j which satisfies the functional requirement (F3a) and then, the TSO can compute the total power consumption of all users in all areas $\sum r_i^{\mathbb{D},\mathbb{S}} = \sum_{j=1}^{N_d} \sum r_i^{j,\mathbb{S}}$ which satisfies the functional requirement (F3b).

VI. SECURITY ANALYSIS

A. PRIVACY PROTECTION OF USER'S POWER CONSUMPTION DATA

As shown in subsection V-C, the plaintext vector P_i is encrypted to generate the ciphertext $C_i = \{p'_i a_i N_1, p'_i b_i N_2, p''_i c_i N_3, p''_i d_i N_4\}$. The vector S is used as splitting indicator to split P_i into p'_i, p''_i. The secret key is $\mathcal{EK}_i = \{a_i N_1, b_i N_2, c_i N_3, d_i N_4\}$ is used to encrypt p'_i, p''_i. The security of this encryption algorithm has been proven in the known ciphertext model [46]. Thus, the content of ciphertext cannot be identified. Therefore, privacy protection of P_i can be achieved. In addition, users receive unique secret encryption keys from the TA generated from the master key set $\mathcal{MK}_1$. Thus, a user u_j who has an encryption key $\mathcal{EK}_j = \{a_j N_1, b_j N_2, c_j N_3, d_j N_4\}$ cannot decrypt the ciphertext C_i generated by another user u_i [49]. Therefore, EPDA satisfies the security requirement (S1).

B. ACCESS CONTROL AND CONFIDENTIALITY OF EACH SUPPLIER/DNO'S APCD

As shown in subsection V-E, the aggregated ciphertext C_{agg} contains all the APCD intended for all DNOs and suppliers. C_{agg} is still a ciphertext encrypted under the master key set $\mathcal{MK}_1$, which has been shown to be secured as discussed in the previous subsection. Thus, the DCC cannot learn any information about the APCD intended for any DNO or supplier.

Access control is achieved through the re-encryption process presented in subsection V-E. The re-encryption process transforms the ciphertext encrypted under $\mathcal{MK}_1$ into a cipher text encrypted under $\mathcal{MK}_2$. Since each DNO and supplier has a decryption key derived from $\mathcal{MK}_2$, they may try to decrypt the re-encrypted ciphertext intended for other competitor DNOs/suppliers. However, the re-encryption process also limits the access only to the intended DNO/supplier through the secure multiplication of the aggregated vector P_{agg} by the access control vector Q_k of a supplier S_k. Thus, a re-encrypted ciphertext $C_{agg}^{\mathbb{D},k}$ intended for S_k cannot be decrypted by another supplier ($S_\ell, k \neq \ell$) because the element-wise product of the access control vector Q_k that was used in the re-encryption and the decryption vector R_ℓ that will be used in the decryption process will result in a vector of zeros. Therefore, EPDA can satisfy the security requirement (S2).

C. CONFIDENTIALITY OF CUSTOMER DISTRIBUTION OF EACH SUPPLIER

The scheme in [29] reveals the customer distribution of each supplier for functionality. This is because the access policy that identifies which supplier should access the APCD is sent in clear. Therefore, internal and external entities can learn the customer distribution of each supplier in the system. For [28], external entities cannot learn the customers' distribution of any supplier since suppliers' IDs are sent from users to BAN-GWs in an encrypted form. However, BAN-GWs, controlled by the DCC, must know the supplier of each user so that the ciphertexts of each group of users under a specific supplier can be aggregated together. On the other hand, in EPDA, this information is hidden by the encryption of the plaintext vector and the scheme allows BAN-GWs to aggregate all the ciphertexts for all users, even if they are supplied by different suppliers, into a single ciphertext without revealing the supplier of each user. Therefore, EPDA can satisfy (S3), whereas [29] cannot.

D. COLLUSION BETWEEN DCC AND DNO

In [28], the electricity supplier receives a ciphertext of the APCD in an area j from the DCC. It also receives the APCD and a random number from the DNO. Electricity suppliers do not trust the DNOs, therefore, in order to ensure the correctness of the APCD received from the DNO, the suppliers should first encrypt the received APCD from the DNO using its public key and the provided random number, and then compare the result with the received ciphertext from the trusted DCC. If they are equal, then the supplier accepts the APCD, otherwise, the DNO has sent higher APCD to charge the supplier more distribution network fees. If the DCC is an active adversaries, it can collude with the DNO to deceive a supplier as follows. First, the DNO encrypts a false APCD using its public key and send the result to the DCC. The DCC will forward this false ciphertext to the supplier instead of sending the correct one. Finally, the DNO will send the false APCD to the supplier. In this way, the correctness of the false

APCD will be ensured and thus the supplier pays the DNO higher network distribution fees.

This type of collusion is not possible in EPDA. As shown in Figure 2, a supplier receives only a single ciphertext from the DCC and can decrypt it directly using its decryption key. Neither the DNO nor the DCC can modify this ciphertext since the modification of the re-encrypted ciphertext requires the knowledge of the master secret key $\mathcal{MK}_2$. Thus, EPDA can resist the collusion attack between the DCC and the DNO if they are active adversaries, whereas [28] can resist it only if the DCC is trusted. It should be noted that [29] assumes this type of collusion cannot happen since the DCC has to be a trusted entity and can access all the APCD of all recipients.

VII. PERFORMANCE EVALUATION

In this section we evaluate the performance of EPDA in terms of the computation cost required by each entity and the communication overhead incurred between each two entities in the network. We also compare the performance of EPDA with [28] and [29].

A. COMPUTATION COST

To evaluate the computation cost of EPDA, we implemented number-theoretic based methods of cryptography using Python charm cryptographic library [56] running on an Intel Core i7-4765T 2.00 GHz and 8 GB RAM. We used supersingular elliptic curve (SS512 curve) with the symmetric Type 1 pairing to realize the bilinear pairing operation [57]. The size of the parameter q is 512 bits. The measurement of the cryptographic operations required in EPDA are reported in the upper part of Table 3, those needed in [28] and [29] are reported in the lower part of the table, and the operations used in all the schemes are reported in the middle part of the table.

TABLE 3. Computation times for cryptographic operations.

	Cryptographic Operation	Time
T_{ve}	Encrypt data vector	0.317 ms*
T_{va}	Aggregate two encrypted vectors	0.002 ms*
T_{vr}	Re-encrypt one encrypted vector	0.605 ms*
T_{vd}	Decrypt one encrypted vector	0.168 ms*
T_h	Hash operation $H : \{0,1\}^* \rightarrow \mathbb{G}$	0.02 ms
T_e	Exponentiation $g^x \in \mathbb{G}$	0.209 ms
T_p	Bilinear Pairing $e(g^a, g^b)$	1.043 ms
T_{as}	Aggregate two signatures $\sigma_1\sigma_2 \in \mathbb{G}$	0.0038 ms
T_{mp}	Multiply two pairings $e_1e_2 \in \mathbb{G}_T$	0.0032 ms
T_{pe}	Paillier Encryption	20.125 ms
T_{pa}	Paillier Aggregation	0.012 ms
T_{pd}	Paillier Decryption	20.933 ms
T_{pr}	Paillier random number recovery	5.932 ms
T_{ae}	Encryption using ABE	15.606 ms‡
T_{ad}	Decrypting ABE ciphertext	8.674 ms‡

*: At vector size of 280 element, $N_d = 14$ and $N_s = 20$
‡: Two attributes per the ABE access policy

As explained in section V, for each reporting period in EPDA, each SM should: (1) perform one vector encryption operation which requires T_{ve}, and (2) compute one signature which requires $T_h + T_e$. A BAN-GW should: (1) verify the received N_α signatures which requires $(N_\alpha + 1)T_p + (N_\alpha - 1)T_{as} + N_\alpha T_h$, (2) aggregate N_α encrypted vectors which requires $(N_\alpha - 1)T_{va}$, and (3) compute one signature which requires $T_h + T_e$. The NAN-GWs, the WAN-GWs, and the DCC perform the same operations as a BAN-GW for the received N_β, N_γ, and N messages respectively. In addition, in EPDA the DCC (1) performs one re-encryption operation for each DNO and each supplier in the system which requires $(N_d + N_s)T_{vr}$, and (2) signs $(N_d + N_s)$ messages, one for each recipient, which requires $(N_d + N_s)(T_h + T_e)$. Finally, each DNO and each supplier (1) verifies one received signature which requires $2T_p + T_h$, and (2) decrypts the re-encrypted vector which requires T_{vd}.

In [28], each SM performs (1) one Paillier cryptosystem encryption operation which requires T_{pe}, and (2) computes one signature which requires $T_h + T_e$. A BAN-GW should (1) verify the received signatures as in EPDA, (2) classify the received N_α Paillier ciphertexts into N_s groups and aggregates each group together which requires $(N_\alpha - N_s)T_{pa}$, and (3) compute one signature as in EPDA. For NAN-GWs, they receive N_β messages from N_β BAN-GWs and perform the same signing/verification operations as a BAN-GW. However, since each message sent from a BAN-GW to a NAN-GW contains N_s Paillier ciphertexts, the computational cost required for aggregation becomes $N_s(N_\beta - 1)T_{pa}$. The WAN-GWs and the DCC perform the same operations as the NAN-GW for the received N_γ and N messages, respectively. However, since the DCC sends messages to $(N_d + N_s)$ recipients, it requires $(N_d + N_s)(T_h + T_e)$ to compute the signatures. Each DNO receives N_s ciphertexts from the DCC. Each DNO should (1) verify the DCC signature which requires $2T_p + T_h$, (2) decrypt N_s Paillier ciphertexts which requires N_sT_{pd}, and (3) recover N_s random numbers from the N_s ciphertexts which requires N_sT_{pr}. Each supplier receives N_d ciphertexts from the DCC, and N_d APCD with N_d random numbers from N_d DNOs through secure N_d secure channels, i.e., signatures are not needed during this communication. However, the supplier should (1) verify the DCC signature which requires $2T_p + T_h$, and (2) perform N_d Paillier encryption operations to ensure the correctness of the received APCD which requires N_dT_{pe}.

For [29], the operations done by each SM, BAN-GW, NAN-GW, and WAN-GW are exactly the same as in [28] with the difference that [29] sends an access policy with the ciphertext instead of sending the supplier IDs as in [28]. However, in [29] the DCC is a trusted node that decrypts all the received ciphertexts and encrypts them using ABE to ensure fine-grained access control. Specifically, the DCC receives N_dN_s Paillier ciphertexts. Therefore, N_dN_s Paillier decryption operations are needed which requires $N_dN_sT_{pd}$. Then, N_dN_s ABE encryption operations are needed and the access policy is constructed as (supplier attribute OR DNO attribute) which requires $N_dN_sT_{ae}$. Each DNO should decrypt N_s ABE ciphertext, one for each supplier, while each supplier should decrypt N_d ciphertexts, one for each DNO, to meet the functional requirements mentioned in subsection III-C.

In Table 4, we summarize the computation cost needed for EPDA, [28] and [29]. In Figure 4, we compare the computation cost of EPDA against [28] and [29] for each entity in the AMI network. We set $N_d = 14$ as in [28] to model an AMI network that can cover the entire grid in UK [58].

TABLE 4. Computation cost comparison. (Times in ms).

	EPDA*
SM	0.546
BAN-GW	$1.068N_\alpha + 1.266$
NAN-GW	$1.068N_\beta + 1.266$
WAN-GW	$1.068N_\gamma + 1.266$
DCC	$1.068N + 0.834(N_d + N_s) + 1.037$
DNO	2.274
Supplier	2.274
Total	$0.229(N_d + N_s) + 1.07(N_\alpha+N_\beta+N_\gamma+N)+9.93$
	[28]
SM	20.354
BAN-GW	$1.078N_\alpha - 0.012N_s + 1.268$
NAN-GW	$1.066N_\beta + 0.012N_\beta N_s - 0.012N_s + 1.268$
WAN-GW	$1.066N_\gamma + 0.012N_\gamma N_s - 0.012N_s + 1.268$
DCC	$0.012NN_s + 1.066N - 0.012N_s+$ $+0.229(N_d + N_s) + 1.039$
DNO	$26.865N_s + 2.106$
Supplier	$20.125N_d + 2.106$
Total	$26.817N_s + 20.125N_d + 0.229(N_d + N_s) + 1.08N_\alpha$ $+1.07(N_\beta+N_\gamma+N)+0.012N_s(N_\beta+N_\gamma+N)+29.41$
	[29]‡
SM	20.354
BAN-GW	$1.078N_\alpha - 0.012N_s + 1.268$
NAN-GW	$1.066N_\beta + 0.012N_\beta N_s - 0.012N_s + 1.26$
WAN-GW	$1.066N_\gamma + 0.012N_\gamma N_s - 0.012N_s + 1.26$
DCC	$36.539N_dN_s + 1.066 + 0.012NN_s - 0.012N_s$
DNO	$8.674N_s$
Supplier	$8.674N_d$
Total	$36.539N_dN_s + 8.674N_d + 8.626N_s + 1.078N_\alpha$ $+1.07(N_\beta+N_\gamma)+0.012(N_\beta+N_\gamma+N)N_s+25.23$

*: At vector size of 280 element, $N_d = 14$ and $N_s = 20$
‡: Two attributes per the ABE access policy

Figure 4a shows the computation cost for each SM versus the number of suppliers in the network. As shown in the figure, the computation cost required by each SM in EPDA increases slightly as the number of suppliers increases. This is because the data vector size depends on the number of suppliers, and thus more arithmetic addition and multiplication operations are needed during the vector encryption process. Also, the figure shows that EPDA is much more efficient than [28], [29]. This is because the encryption process in EPDA requires only efficient arithmetic addition and multiplication operations compared to the computationally expensive Paillier encryption time, T_{pe}, required in other schemes. Therefore, EPDA is more suitable than other schemes for the resource-constrained SMs due to its lower computational cost.

Figure 4b shows the computation cost for each BAN-GW versus the number of user under each BAN-GW at $N_s = 15$ and $N_s = 30$. The computation cost of all the schemes are almost the same. Although the schemes require different times to aggregate the received ciphertexts from the users, the signature verification process, which is the same in all schemes, consumes the most time. On contrary, Figures 4c, and 4d show that EPDA has better performance than [28] and [29] for the NAN-GWs and WNA-GWs computations. In [28] and [29], each NAN-GW/WAN-GW receives N_s ciphertexts from each of its BAN-GW/NAN-GW children. Thus, as the number of suppliers increases, more aggregation operations are needed. Although the vector size in EPDA also increases linearly with the number of suppliers, which means more arithmetic operations are need for aggregation, EPDA has lower aggregation time than other schemes that aggregates Paillier ciphertexts.

Figure 4e shows the computation cost required by the DCC versus the number of suppliers in the network. It is clear that the computation cost of the DCC in [29] is worst compared to EPDA and [28]. This is because in [29] the DCC is a trusted node that decrypts N_dN_s Paillier ciphertexts and encrypts them using ABE as explained earlier. In Figure 4f, we excluded [29] to compare EPDA against [28] and the figure shows that the DCC in EPDA has lower computation time than [28].

In Figure 4g we compare the computation cost required by each DNO versus the number of suppliers in the network while Figure 4h shows the computations required by each supplier versus the number of DNOs. It can be seen in the figure that EPDA is the most efficient while [28] is the worst. This is because in EPDA only one vector decryption operation is requires which is much more efficient than either Paillier decryption operation or the ABE decryption operations.

To sum up, EPDA outperforms other schemes in terms of computation cost at each entity in the network except at BAN-GWs where all the schemes require almost the same computation cost.

B. COMMUNICATION OVERHEAD

The communication overhead is measured by the size of transmitted messages between the network entities. In the multi-recipient AMI network, the communication overheads that will be measured are for the messages sent from SM to BAN-GW, BAN-GW to NAN-GW, NAN-GW to WAN-GW, WAN-GW to DCC, DCC to DNO, and DCC to supplier. Since the overhead between the intermediate gateways are exactly the same in all schemes, we can evaluate only SM-to-GW, GW-to-DCC, DCC-to-DNO, and DCC-to-Supplier overheads.

In EPDA, each SM sends a message on the form $C_i \parallel \mathrm{ID}_{u_i} \parallel \mathrm{ID}_{bg_\alpha} \parallel \mathrm{TS} \parallel \sigma_i$ to its BAN-GW. Thus, the SM-to-GW overhead is $|C^v| + 2|\mathrm{ID}| + |\mathrm{TS}| + |\sigma|$, where $|C^v|$ is the size of an encrypted vector which is $64N_dN_s$ bits. Since all ciphertext are aggregated in a single ciphertext in EPDA,

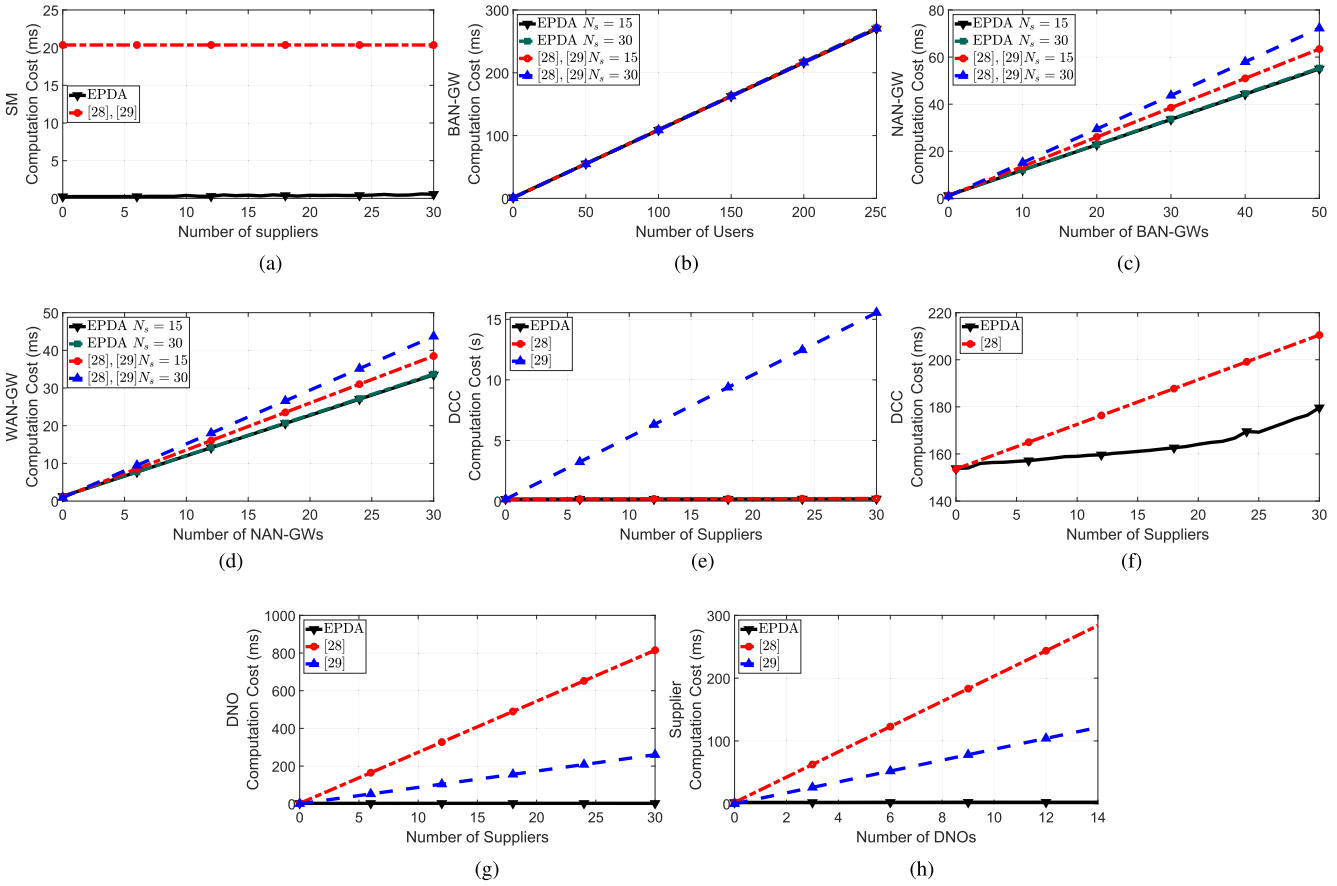

FIGURE 4. Computation cost comparison. (a) Computation Cost for each SM. (b) Computation Cost for each BAN-GW. (c) Computation Cost for each NAN-GW. (d) Computation Cost for each WAN-GW. (e) Computation Cost for the DCC. (f) Computation Cost for the DCC, [29] is excluded. (g) Computation Cost for each DNO. (h) Computation Cost for each supplier.

the GW-to-DCC overhead is the same as that of SM-to-GW overhead. Similarly, the DCC-to-DNO and the DCC-to-supplier overheads are the same since single vector is sent in either case.

In [28], each SM sends a message on the form $\mathrm{ID}_{u_i} \parallel \mathrm{ID}_{bg_\alpha} \parallel \mathrm{ID}_{d_j} \parallel E(\mathrm{ID}_{S_k} \parallel C_i^p) \parallel \mathrm{TS} \parallel \sigma_i$. Thus, the SM to GW overhead is $3|\mathrm{ID}| + (|C^p| + |\mathrm{ID}|) + |\mathrm{TS}| + |\sigma|$, where $|C^p|$ is the size of Paillier ciphertext. The messages from the BAN-GW to NAN-GW, NAN-GW to WAN-GW, and WAN-GW to the DCC have the same form and are represented as GW to DCC overhead. These message is on the form of $\mathrm{ID}_{wg_\gamma} \parallel \mathrm{ID}_{dcc} \parallel \mathrm{ID}_{d_j} \parallel (\mathrm{ID}_{s_1} \parallel C_1^p) \cdots \parallel (\mathrm{ID}_{s_{N_s}} \parallel C_{N_s}^p) \parallel \mathrm{TS} \parallel \sigma_\gamma$. Since N_s ciphertexts corresponding to N_s suppliers in the network should be sent to meet the functional requirements, the GW-to-DCC overhead is $3|\mathrm{ID}| + N_s(|C^p| + |\mathrm{ID}|) + |\mathrm{TS}| + |\sigma|$. Finally, the DCC sends to each DNO in the network a message containing N_s ciphertexts with an overhead of $(2|\mathrm{ID}| + N_s(|C^p| + |\mathrm{ID}|) + |\mathrm{TS}| + |\sigma|$. Also, the DCC sends to each supplier in the network a message containing N_d ciphertexts with an overhead of $2|\mathrm{ID}| + N_d(|C^p| + |\mathrm{ID}|) + |\mathrm{TS}| + |\sigma|$. It should be noted that, to meet the functional requirements, [28] requires each DNO to send N_s ciphertexts to N_s suppliers in the network through secure channels. We have excluded this additional overhead from our comparison since EPDA and [29] do not require any overhead between DNOs and suppliers.

In [29], the SM-to-GW overhead and the GW-to-DCC overhead are the same as [28] except that access policy of size |AP| is used instead of supplier IDs to classify Paillier ciphertexts. The DCC-to-DNO overhead is a set of N_s ABE ciphertexts with a set of N_s access policies. Therefore, the DCC-to-DNO overhead is $N_s((1+3L)|C^a| + |AP|)$, where $L = 2$ is the number of attributes in the access policy, and $|C^a|$ is the size of one point in the Elliptic curve group used for ABE. Similarly, the DCC-to-Supplier overhead is $N_d((1+3L)|C^a| + |AP|)$ since each supplier receives N_d ABE ciphertexts with access policies for N_d DNOs in the network.

Using $|C^v| = 64N_dN_s$ bits, $|\mathrm{ID}| = 16$ bits, $|\mathrm{TS}| = 32$ bits, $|\sigma| = 1,024$ bits, $|C^p| = 2,048$ bits, $|AP| = 16$ bits, $L = 2$ attributes and $|C^a| = 1,024$ bits, we summarize the communication overhead for all the schemes in Table 5 and compare all the overheads of all the schemes in Figure 5.

In Figure 5a, we compare the SM-to-GW overhead of EPDA against other schemes. Since AMI networks are typically deployed using wireless mesh networks [59], we consider two cases in our comparison. In one-hop

TABLE 5. Communication overhead comparison (bits).

	SM-to-GW	GW-to-DCC	DCC-to-DNO	DCC-to-Supplier
EPDA	$64N_dN_s + 1088$	$64N_dN_s + 1088$	$64N_dN_s + 1088$	$64N_dN_s + 1088$
[28]	$3168N_c$	$2064N_s + 1104$	$2064N_s + 1088$	$2064N_d + 1088$
[29]	$3168N_c$	$2064N_s + 1104$	$14352N_s$	$14352N_d$

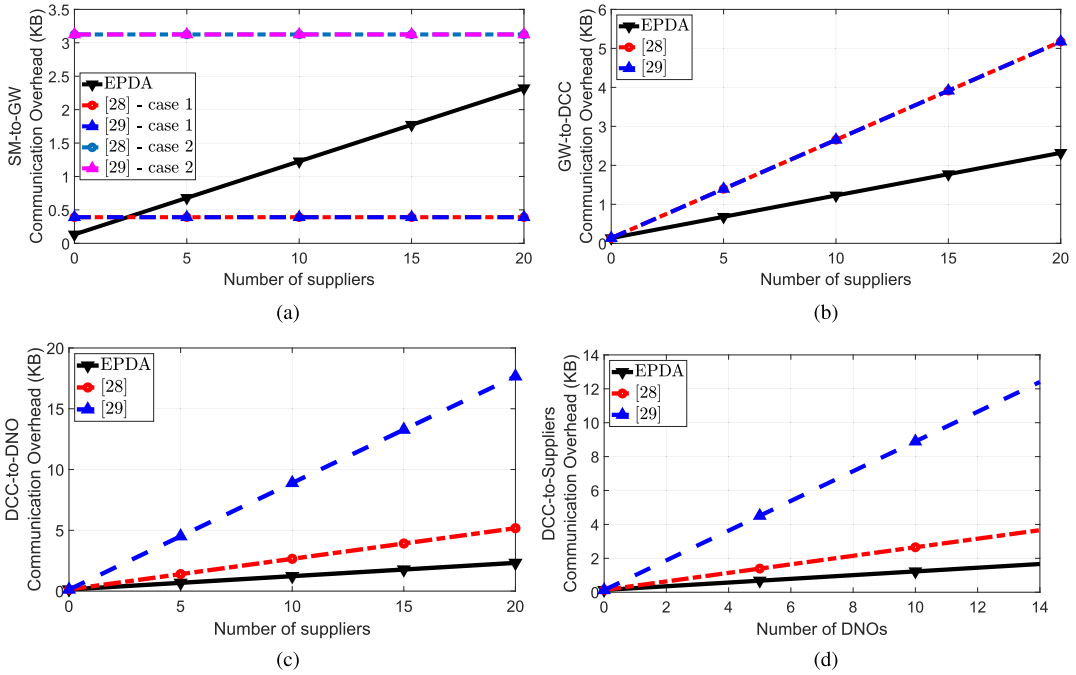

FIGURE 5. Communication overhead comparison. (a) SM-to-GW overhead. (b) GW-to-DCC overhead. (c) DCC-to-DNO overhead. (d) DCC-to-Supplier overhead.

communication case (Case 1), the SMs and the BAN-GW communicate directly, whereas in multi-hop communication case (Case 2), an SM acts as a router to relay other SMs' messages to the BAN-GW. In the latter case, we assume that each SM relays eight messages for eight children SMs. We denote the number of relayed messages for the children meter as N_c as given in Table 5. As shown in the figure, as the number of suppliers increase "horizontal axis", the SM-to-GW overhead in EPDA increases linearly because the vector size in EPDA increases linearly with the number of suppliers, while [28] and [29] have constant overhead. On the other hand, as the number of relayed messages increases in case 2, the SM-to-GW overhead increases in other schemes since the rely nodes cannot classify their children SMs' ciphertexts and thus cannot aggregate them, instead, they have to forward all their ciphertexts to the BAN-GW. Unlike that, in EPDA any relay node can aggregate their children SMs' ciphertexts since EPDA does not require ciphertext classification for data aggregation, i.e., the overhead in both cases will be the same for EPDA. In addition, it should be noted that, the linear increment in overhead in EPDA is the cost needed to hide the customers distribution of each recipient in the network to satisfy the security requirement (S3) which cannot be achieved in [28] and [29].

In Figure 5b, we plot the GW-to-DCC overhead versus the number of suppliers in the network. In EPDA, the GW-to-DCC overhead is exactly the same as the SM-to-GW overhead since all the encrypted vectors are aggregated in one ciphertext before sending them to the DCC. On contrary, [28] and [29] send N_s Paillier ciphertext for N_s suppliers in the network to meet the functional requirements, and thus, they require increased overhead as compared to EPDA.

In Figure 5c, and Figure 5d, we plot the DCC-to-DNO overhead versus the number of suppliers in the network, and the DCC-to-supplier overhead versus the number of DNOs in the network, respectively. As shown in the figures, EPDA has the least overhead needed. This is because the ABE ciphertext in [29] and the Paillier ciphertext in [28] are larger than the encrypted vector transmitted in EPDA.

Based on the security analysis presented in section VI and the performance evaluation presented in section VII, we summarize in Table 6 the comparison of EPDA against [28], [29].

It should be noted that, although EPDA has better computation and communication overheads when compared

TABLE 6. Comparison between EPDA and similar existing schemes.

	EPDA	[28]	[29]
Multi-Recipient	√	√*	√*‡
Efficiency			
Encryption	√	×	×
Aggregation	√	√	√
Decryption	√	×	×
Privacy Preservation	√	√	√
Communication Overhead	Low	High	High
Confidentiality of Customers' Distributioin	√	√	×

‡: Requires an online trusted central node to ensure access control.
* : Allows aggregators to (1) classify the individual ciphertext, (2) send several aggregated ciphertexts, one for each classified group.

to [28] and [29], it uses larger key sizes. The key size in EPDA is 39.48 kB while it is 0.384 kB in [28] and [29]. This is because the key size in EPDA increases with both the number of DNOs and suppliers in the system. Note that, key distribution does not occur very frequently unlike the periodic data collection of PCD. Thus, the increased key size in EPDA does not have a marked impact on the communication overhead.

VIII. CONCLUSION

In this paper, we proposed EPDA, an efficient and privacy-preserving data collection and access control scheme for multi-recipient AMI networks. Based on, but not limited to, the kNN encryption technique, we developed a novel proxy re-encryption scheme that allows data aggregation before re-encryption and can allow either full or partial access to the aggregated data after re-encryption as needed. Thus, each recipient can access only the aggregated data intended to it and cannot access the aggregated data intended to other recipients to achieve the functional and security requirements for the multi-recipients AMI networks. Our security analysis demonstrated that EPDA is secure and can ensure better security compared to other data collection schemes. Specifically, EPDA can resist collusion attacks and hide customers's distribution that is needed for fair electricity trade market. Moreover, our performance evaluations showed that EPDA is both computationally and bandwidth-wise efficient compared to similar schemes.

ACKNOWLEDGMENT

This work was supported by the Qatar National Research Fund (a member of Qatar Foundation) through NPRP under Grant NPRP10-1223-160045. The statements made herein are solely the responsibility of the authors.

REFERENCES

[1] U.S. Departmrnt of Energy. (2008). *The Smart Grid: An Introduction*. [Online]. Available: https://www.energy.gov/oe/downloads/smart-grid-introduction-0

[2] D. Mosher and A. Kiersz. (2016). *A 100-Year Solar Storm Could Fry Our Power Grids—These are the Places Most at Risk*. Accessed: Nov. 30, 2018. [Online]. Available: https://www.businessinsider.com/solar-storm-risk-map-united-states-2016-9

[3] U. S. Canada Power System Outage Task Force. (2014). *Final Report on the August 14, 2003 Blackout in the United States and Canada: Causes and recommendations*. Accessed: Nov. 30, 2018. [Online]. Available: https://energy.gov/sites/prod/files/oeprod/DocumentsandMedia/BlackoutFinal-Web.pdf

[4] NIST. (2014). *NIST Framework and Roadmap for Smart Grid Interoperability Standards, Release 3.0*. Accessed: Nov. 30, 2018. [Online]. Available: https://www.nist.gov/sites/default/files/documents/smartgrid/NIST-SP-1108r3.pdf

[5] M. I. Baza, M. M. Fouda, A. S. T. Eldien, and H. A. K. Mansour, "An efficient distributed approach for key management in microgrids," in *Proc. 11th Int. Comput. Eng. Conf. (ICENCO)*, 2015, pp. 19–24.

[6] H. Farhangi, "The path of the smart grid," *IEEE Power Energy Mag.*, vol. 8, no. 1, pp. 18–28, Jan./Feb. 2010.

[7] D. G. Hart, "Using AMI to realize the smart grid," in *Proc. IEEE Power Energy Soc. Gen. Meeting-Convers. Del. Electr. Energy 21st Century*, Jul. 2008, pp. 1–2.

[8] F. M. Cleveland, "Cyber security issues for advanced metering infrasttructure (AMI)," in *Proc. IEEE Power Energy Soc. Gen. Meeting-Convers. Del. Electr. Energy 21st Century*, Jul. 2008, pp. 1–5.

[9] B. Karimi, V. Namboodiri, and M. Jadliwala, "Scalable meter data collection in smart grids through message concatenation," *IEEE Trans. Smart Grid*, vol. 6, no. 4, pp. 1697–1706, Jul. 2015.

[10] E. J. Palacios-Garcia *et al.*, "Using smart meters data for energy management operations and power quality monitoring in a microgrid," in *Proc. IEEE 26th Int. Symp. Ind. Electron. (ISIE)*, Jun. 2017, pp. 1725–1731.

[11] A.-H. Mohsenian-Rad, V. W. S. Wong, J. Jatskevich, R. Schober, and A. Leon-Garcia, "Autonomous demand-side management based on game-theoretic energy consumption scheduling for the future smart grid," *IEEE Trans. Smart Grid*, vol. 1, no. 3, pp. 320–331, Dec. 2010.

[12] M. Pazos-Revilla, A. Alsharif, S. Gunukula, T. N. Guo, M. Mahmoud, and X. Shen, "Secure and privacy-preserving physical-layer-assisted scheme for EV dynamic charging system," *IEEE Trans. Veh. Technol.*, vol. 67, no. 4, pp. 3304–3318, Apr. 2018.

[13] M. Nabil *et al.*, "Priority-based and privacy-preserving electric vehicle dynamic charging system with divisible e-payment," in *Smart Cities Cybersecurity and Privacy*, D. B. Rawat and K. Z. Ghafoor, Eds. Amsterdam, The Netherlands: Elsevier, 2019, ch. 12, pp. 165–186. [Online]. Available: http://www.sciencedirect.com/science/article/pii/B9780128150320000123

[14] C. Douris. (2017). *Balancing Smart Grid Data and Consumer Privacy*. Accessed: Nov. 30, 2018. [Online]. Available: http://www.lexingtoninstitute.org/wp-content/uploads/2017/07/Lexington_Smart_Grid_Data_Privacy-2017.pdf

[15] A. Paverd, A. Martin, and I. Brown, "Security and privacy in smart grid demand response systems," in *Proc. Int. Workshop Smart Grid Secur.*, 2014, pp. 1–15.

[16] Y. Gong, Y. Cai, Y. Guo, and Y. Fang, "A privacy-preserving scheme for incentive-based demand response in the smart grid," *IEEE Trans. Smart Grid*, vol. 7, no. 3, pp. 1304–1313, May 2016.

[17] G. W. Hart, "Nonintrusive appliance load monitoring," *Proc. IEEE*, vol. 80, no. 12, pp. 1870–1891, Dec. 1992.

[18] C. Laughman *et al.*, "Power signature analysis," *IEEE Power Energy Mag.*, vol. 99, no. 2, pp. 56–63, Mar./Apr. 2003.

[19] X. Li, X. Liang, R. Lu, X. Shen, X. Lin, and H. Zhu, "Securing smart grid: Cyber attacks, countermeasures, and challenges," *IEEE Commun. Mag.*, vol. 50, no. 8, pp. 38–45, Aug. 2012.

[20] Electronic Privacy Information Center (EPIC). *The Smart Grid and Privacy*. Accessed: Nov. 30, 2018. [Online]. Available: https://epic.org/privacy/smartgrid/smartgrid.html

[21] H. Mohammed, S. Tonyali, K. Rabieh, M. Mahmoud, and K. Akkaya, "Efficient privacy-preserving data collection scheme for smart grid AMI networks," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2016, pp. 1–6.

[22] G. Karopoulos, C. Ntantogian, and C. Xenakis, "MASKER: Masking for privacy-preserving aggregation in the smart grid ecosystem," *Comput. Secur.*, vol. 73, pp. 307–325, Mar. 2018.

[23] F. Knirsch, G. Eibl, and D. Engel, "Error-resilient masking approaches for privacy preserving data aggregation," *IEEE Trans. Smart Grid*, vol. 9, no. 4, pp. 3351–3361, Jul. 2018.

[24] A. Alsharif, M. Nabil, S. Tonyali, H. Mohammed, M. Mahmoud, and K. Akkaya, "EPIC: Efficient privacy-preserving scheme with E2E data integrity and authenticity for AMI networks," *CoRR*, vol. abs/1810.01851, 2018. [Online]. Available: http://arxiv.org/abs/1810.01851
[25] S. Li, K. Xue, Q. Yang, and P. Hong, "PPMA: Privacy-preserving multi-subset data aggregation in smart grid," *IEEE Trans. Ind. Informat.*, vol. 14, no. 2, pp. 462–471, Feb. 2018.
[26] A. Alsharif, M. Nabil, M. Mahmoud, and M. Abdallah, "Privacy-preserving collection of power consumption data for enhanced AMI networks," in *Proc. 25th Int. Conf. Telecommun. (ICT)*, Jun. 2018, pp. 196–201.
[27] N. Saxena, B. J. Choi, and S. Grijalva, "Secure and privacy-preserving concentration of metering data in AMI networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2017, pp. 1–7.
[28] M. A. Mustafa, N. Zhang, G. Kalogridis, and Z. Fan, "DEP2SA: A decentralized efficient privacy-preserving and selective aggregation scheme in advanced metering infrastructure," *IEEE Access*, vol. 3, pp. 2828–2846, 2015.
[29] S. Ruj and A. Nayak, "A decentralized security framework for data aggregation and access control in smart grids," *IEEE Trans. Smart Grid*, vol. 4, no. 1, pp. 196–205, Mar. 2013.
[30] M. A. Mustafa, S. Cleemput, A. Aly, and A. Abidin. (2018). "A secure and privacy-preserving protocol for smart metering operational data collection." [Online]. Available: https://arxiv.org/abs/1801.08353
[31] G. Erbach and EPRS | European Parliamentary Research Service. (2016). *Understanding Electricity Markets in the EU*. Accessed: Nov. 30, 2018. [Online]. Available: http://www.europarl.europa.eu/thinktank/en/document.html?reference=EPRS_BRI(2016)593519
[32] Quantum Gas & Power Services, Ltd. *Energy Deregulated States in the United States*. Accessed: Nov. 30, 2018. [Online]. Available: http://www.quantumgas.com/list_of_energy_deregulated_states_in_united_states.html
[33] N. Granados, A. Gupta, and R. J. Kauffman, "Online and offline demand and price elasticities: Evidence from the air travel industry," *Inf. Syst. Res.*, vol. 23, no. 1, pp. 164–181, 2012.
[34] M. Fisher, S. Gallino, and J. Li, "Competition-based dynamic pricing in online retailing: A methodology validated with field experiments," *Manage. Sci.*, vol. 64, no. 6, pp. 2496–2514, 2017.
[35] P. Paillier *et al.*, "Public-key cryptosystems based on composite degree residuosity classes," in *Eurocrypt*, vol. 99. Berlin, Germany: Springer, 1999, pp. 223–238.
[36] G. Danezis, C. Fournet, M. Kohlweiss, and S. Zanella-Béguelin, "Smart meter aggregation via secret-sharing," in *Proc. 1st ACM Workshop Smart Energy Grid Secur.*, 2013, pp. 75–80.
[37] F. D. Garcia and B. Jacobs, "Privacy-friendly energy-metering via homomorphic encryption," in *Security and Trust Management*. Berlin, Germany: Springer, 2010, pp. 226–238.
[38] S. Finster and I. Baumgart, "Elderberry: A peer-to-peer, privacy-aware smart metering protocol," in *Proc. IEEE INFOCOM*, Apr. 2013, pp. 3411–3416.
[39] A. C. Yao, "Protocols for secure computations," in *Proc. IEEE 23rd Annu. Symp. Found. Comput. Sci.*, Nov. 1982, pp. 160–164.
[40] A. Lewko and B. Waters, "Decentralizing attribute-based encryption," in *Proc. Annu. Int. Conf. Theory Appl. Cryptograph. Techn.* Berlin, Germany: Springer, 2011, pp. 568–588.
[41] Office of Gas and Electricity Markets, Ofgem. *The GB Electricity Transmission Network*. Accessed: Nov. 30, 2018. [Online]. Available: https://www.ofgem.gov.uk/electricity/transmission-networks/gb-electricity-transmission-network/
[42] M. Nabil, A. Alsharif, A. Sherif, M. Mahmoud, and M. Younis, "Efficient multi-keyword ranked search over encrypted data for multi-data-owner settings," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2018, pp. 1–6.
[43] N. Cao, C. Wang, M. Li, K. Ren, and W. Lou, "Privacy-preserving multi-keyword ranked search over encrypted cloud data," in *Proc. IEEE INFOCOM*, Apr. 2011, pp. 829–837.
[44] N. Cao, C. Wang, M. Li, K. Ren, and W. Lou, "Privacy-preserving multi-keyword ranked search over encrypted cloud data," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 1, pp. 222–233, Jan. 2014.
[45] C. Yang, W. Zhang, J. Xu, J. Xu, and N. Yu, "A fast privacy-preserving multi-keyword search scheme on cloud data," in *Proc. Int. Conf. Cloud Service Comput.*, Nov. 2012, pp. 104–110.
[46] W. K. Wong, D. W.-L. Cheung, B. Kao, and N. Mamoulis, "Secure kNN computation on encrypted databases," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, 2009, pp. 139–152.
[47] H. Li, Y. Yang, T. H. Luan, X. Liang, L. Zhou, and X. S. Shen, "Enabling fine-grained multi-keyword search supporting classified sub-dictionaries over encrypted cloud data," *IEEE Trans. Dependable Secure Comput.*, vol. 13, no. 3, pp. 312–325, May/Jun. 2016.
[48] H. Li, D. Liu, K. Jia, and X. Lin, "Achieving authorized and ranked multi-keyword search over encrypted cloud data," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2015, pp. 7450–7455.
[49] H. Li, D. Liu, Y. Dai, and T. H. Luan, "Engineering searchable encryption of mobile cloud networks: When QoE meets QoP," *IEEE Wireless Commun.*, vol. 22, no. 4, pp. 74–80, Aug. 2015.
[50] A. B. T. Sherif, K. Rabieh, M. M. E. A Mahmoud, and X. Liang, "Privacy-preserving ride sharing scheme for autonomous vehicles in big data era," *IEEE Internet Things J.*, vol. 4, no. 2, pp. 611–618, Apr. 2017.
[51] A. Sherif, A. Alsharif, J. Moran, and M. Mahmoud, "Privacy-preserving ride sharing organization scheme for autonomous vehicles in large cities," in *Proc. IEEE 86th Veh. Technol. Conf. (IEEE VTC-Fall)*, Sep. 2017, pp. 1–5.
[52] A. Sherif, A. Alsharif, M. Mahmoud, M. Abdallah, and M. Song, "Efficient privacy-preserving aggregation scheme for data sets," in *Proc. 25th Int. Conf. Telecommun. (ICT)*, Jun. 2018, pp. 191–195.
[53] A. Sherif, A. Alsharif, J. Moran, and M. Mahmoud, "Privacy-preserving autonomous cab service management scheme," in *Proc. 3rd Africa Middle East Conf. Softw. Eng.*, Dec. 2017, pp. 19–24.
[54] M. Nabil, M. Mahmoud, A. Sherif, A. Alsharif, and M. Abdallah. (2018). "Efficient and privacy-preserving ride sharing organization for transferable and non-transferable services." [Online]. Available: https://arxiv.org/abs/1809.07314
[55] D. Boneh, C. Gentry, B. Lynn, and H. Shacham, "Aggregate and verifiably encrypted signatures from bilinear maps," in *Proc. Int. Conf. Theory Appl. Cryptograph. Techn.* Berlin, Germany: Springer, 2003, pp. 416–432.
[56] J. A. Akinyele *et al.*, "Charm: A framework for rapidly prototyping cryptosystems," *J. Cryptograph. Eng.*, vol. 3, no. 2, pp. 111–128, 2013.
[57] *SEC 2: Recommended Elliptic Curve Domain Parameters*, Standard, The Standards for Efficient Cryptography Group (SECG), 2000. [Online]. Available: http://www.secg.org/
[58] Office of Gas and Electricity Markets, Ofgem. *The GB Electricity Distribution Network*. Accessed: Nov. 30, 2018. [Online]. Available: https://www.ofgem.gov.uk/electricity/distribution-networks/gb-electricity-distribution-network
[59] G. Leon. (2011). *Smart Planning for Smart Grid AMI Mesh Networks*. Accessed: Nov. 30, 2018. [Online]. Available: https://www.smartgrid.gov/document/smart_planning_smart_grid_ami_mesh_networks

AHMAD ALSHARIF (M'18) received the B.Sc. and M.Sc. degrees in electrical engineering from Benha University, Egypt, in 2009 and 2015, respectively. He is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering, Tennessee Tech University, USA. He is also a Cybersecurity Instructor with the Computer Science Department, University of Central Arkansas, USA. His research interests include security and privacy in smart grid, cyber-physical systems, vehicular ad hoc networks, and multihop cellular networks. In 2009, he was one of the recipients of the Young Innovator Award from the Egyptian Industrial Modernization Centre.

MAHMOUD NABIL received the B.S. and M.S. degrees in computer engineering from Cairo University, Cairo, Egypt, in 2012 and 2016, respectively. He is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering, Tennessee Tech University, USA, where he is a Graduate Research Assistant. His research interests include machine learning, cryptography and network security, smart-grid and AMI networks, and vehicular ad-hoc networks.

MOHAMED M. E. A. MAHMOUD received the Ph.D. degree from the University of Waterloo, in 2011. From 2011 to 2012, he was a Postdoctoral Fellow with the Broadband Communications Research Group, University of Waterloo. From 2012 to 2013, he was a Visiting Scholar with the University of Waterloo, and a Postdoctoral Fellow with Ryerson University. He is currently an Associate Professor with the Department of Electrical and Computer Engineering, Tennessee Tech University, USA. He has authored more than 23 papers published in major IEEE conferences and journals, such as INFOCOM conference and IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, the IEEE TRANSACTIONS ON MOBILE COMPUTING, and the IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS. His research interests include security and privacy preserving schemes for smart grid communication networks, mobile ad hoc networks, sensor networks, and delay-tolerant networks. He served as a Technical Program Committee Member for several IEEE conferences and as a Reviewer for several journals and conferences, such as the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, the IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, and the Journal of *Peer-to-Peer Networking and Applications*. He was a recipient of the NSERC-PDF Award and the Best Paper Award from IEEE International Conference on Communications (ICC'09), Dresden, Germany, in 2009. He serves as an Associate Editor with the Springer Journal of *Peer-to-Peer Networking and Applications*.

MOHAMED ABDALLAH (S'94–M'08–SM'13) received the B.Sc. degree from Cairo University, in 1996, and the M.Sc. and Ph.D. degrees from the University of Maryland at College Park, in 2001 and 2006, respectively. From 2006 to 2016, he held academic and research positions with Cairo University and Texas A&M University at Qatar. He is currently a Founding Faculty Member having the rank of Assistant Professor with the Information and Computing Technology Division, Hamad bin Khalifa University. He was a Technical Program Committee Member of several major IEEE conferences. He was a Technical Program Chair of the 10th International Conference on Cognitive Radio Oriented Wireless Networks. He was an Associate Editor for the IEEE TRANSACTIONS ON COMMUNICATIONS.