A Privacy-Enhancing Protocol that Provides In-Network Data Aggregation and Verifiable Smart Meter Billing

Fábio Borges, Denise Demirel, Leon Böck, Johannes Buchmann, and Max Mühlhäuser
Technische Universität Darmstadt/CASED
64293 Darmstadt, Germany
fabio.borges@cased.de, ddemirel@cdc.informatik.tu-darmstadt.de, leonboeck@gmail.com
buchmann@cdc.informatik.tu-darmstadt.de, max@informatik.tu-darmstadt.de

***Abstract*—We present an innovative protocol combining in-network data aggregation and smart meter billing for a smart grid scenario. The former enables an energy supplier to allocate and balance resources. The latter provides dynamic pricing schemes according to fine-grained consumption profiles. Moreover, smart meters and their energy supplier can prove their billing values. Since the energy supplier knows the amount of generated electricity and the consolidated consumption in a round of measurements, the energy supplier can detect energy loss and fraud. To preserve customers' privacy, we use a homomorphic commitment scheme with a homomorphic encryption scheme. All data sent from a meter to any other component in the communication network is either a commitment or an encrypted message. To provide security and privacy, we only require software modifications, leaving the hardware of the smart grid unchanged.**

***Index Terms*—Smart Grid, Privacy, Security, Homomorphic Commitment, Homomorphic Encryption.**

I. Introduction

Worldwide the interest in smart grids is increasing since their progressive approach provides remote measurements of electric energy consumptions. Frequent measurements allow an improvement of the power load by adjusting the electricity generation to a predicted power demand. Consequently, energy suppliers can avoid backup power plants and reduce the CO_2 emissions. Furthermore, fine-graded consumption data enable to trade electricity at different tariffs, so-called smart meter billing. At times of the day with high generation of electricity by renewables or less electricity consumption, energy suppliers can offer a lower tariff. Thus, customers could save money, for instance, by recharging the batteries of their electric cars at low costs.

On the opposite, fine-graded consumption data enable an adversary to draw conclusions regarding the habits of the customers. Greveler *et al.* [1] show that smart meters could even be used to identify which TV channel the customers are watching. This is possible because televisions generate a specific pattern of electric energy consumption according to the presented image. To protect customers' privacy, energy suppliers could request measurements from smart meters for larger time intervals, for instance, hourly, daily, or weekly. However, they need frequent measurements to predict the energy consumption and to improve the efficiency of the electric power network.

An approach to protect privacy is the use of batteries that allow the energy supplier to receive frequent measurements from smart meters while the customers' privacy is preserved. However, this approach is quite expensive, because the battery should be large enough to support all appliances in the household and each household needs to buy and install its own battery.

Another approach is to protect the billing information [2] and the aggregation of customers' measurements in a neighborhood [3]–[12]. Normally, smart meters are wireless and can communicate with each other using a communication protocol. Thus, they can use homomorphic encryption to aggregate the measurements and compute their total electricity consumption, namely, the consolidated consumption. This process is known as in-network data aggregation and addresses the problem of frequent measurements [3]. However, aggregation of measurements do not solve the privacy problem of smart meter billing, because the supplier has to generate a separate bill for each customer. Most publications that focus on privacy either concentrate on frequent measurements of the electricity consumption with aggregation [3]–[12] or address the dynamic pricing schemes by smart meter billing [2]. In this paper, we cover both aspects: data aggregation and smart meter billing. We introduce a protocol that offers both functions without violating customers' privacy.

Instead of sending the amount of used electricity to the supplier, the meter commits to its measurement $\mathsf{Com}(m, r)$, where m denotes the measured consumption and r some random value. A commitment scheme allows the meter to commit to a consumption in a way that it cannot change the value later on while keeping the measured consumption hidden. Further, each meter encrypts its measurement $\mathsf{Enc}(m)$ using an asymmetric encryption scheme and the public key of the supplier. The meters compute the encrypted consolidated consumption using in-network data aggregation for a set of

households $\prod \mathsf{Enc}(m_i) = \mathsf{Enc}(\sum m_i)$. Due to the homomorphic properties of some public-key encryption schemes, this is possible without allowing an adversary to draw conclusions regarding the electricity usage per household. In addition, each meter sends its encrypted decommitment value $\mathsf{Enc}(r)$ to the network that homomorphically adds all received values, resulting in the aggregated decommitment values $\prod \mathsf{Enc}(r_i) = \mathsf{Enc}(\sum r_i)$. Furthermore, each meter sends its commitment to its supplier, and the last meter in the aggregation also sends the encrypted consolidated consumption and the encrypted sum of decommitment values to the supplier. We show that using an encryption scheme and a commitment scheme, which are homomorphic over the same group, both encrypted values constitute the opening values of the product of commitments $\prod \mathsf{Com}(m_i, r_i) = \mathsf{Com}(\sum m_i, \sum r_i)$. After decrypting the consolidated consumption and the decommitment value, the supplier can use this property to verify the correctness of all received commitments before computing them for billing.

After a predefined time interval, for instance, a month or a week, the supplier computes a commitment for the total amount each customer has to pay. To generate one bill, the supplier just multiplies all commitments received from one meter, that is $\prod \mathsf{Com}(m_j, r_j)^T = \mathsf{Com}(\sum t_j \cdot m_j, \sum t_j \cdot r_j)$, where vector $T = (t_1, t_2, \ldots, t_n)$ denotes the price per round of measurements. In the meantime, each meter computes the corresponding opening values $\prod \mathsf{Enc}(m_j)^T = \mathsf{Enc}(\sum t_j \cdot m_j)$ and $\prod \mathsf{Enc}(r_j)^T = \mathsf{Enc}(\sum t_j \cdot r_j)$, respectively, and sends them to the supplier. By decrypting the received data and opening the commitments, the supplier can verify the correctness of the total amount to pay $\sum t_j \cdot m_j$.

The main contribution of this paper is the introduction of a protocol combining data aggregation and smart meter billing. The former enables the supplier to allocate and to balance resources and the latter to provide dynamic pricing schemes according to fine-graded consumption profiles. In comparison with previous smart billing schemes like [2], we do not need a privacy component per household but use in-network aggregation that processes data sent by several meters. Furthermore, running the aggregation protocol the supplier can collect enough information to verify the bills but is not able to reveal the individual measurements. Furthermore, using our protocol, it is not necessary to store and manage a record of commitments but only the latest opening values. In this paper, we give an explanation of our protocol and discuss the improvements in comparison with existing solutions.

This paper is structured as follows. In Section II, we explain two existing privacy-preserving protocols: one for data aggregation and another one for smart meter billing. In Section III, we present our improved protocol. Finally, we compare the related work in Section IV and present the conclusions in Section V.

II. Background

In this section, we provide a short description how data aggregation and smart meter billing could be performed. Due to the large variety of approaches [2]–[13], we concentrate on two specific solutions, the proposal for data aggregation by Li *et al.* [3] and the smart billing protocol introduced by Jawurek *et al.* [2]. The comprehension of these two protocols facilitates to understand our proposal.

A. Data Aggregation

Li *et al.* [3] introduce a privacy-preserving solution for in-network data aggregation. They use the Paillier Cryptosystem [14], which is an additive homomorphic public-key encryption scheme. Furthermore, to solve the malleability introduced by the homomorphic property, Li *et al.* [4] propose the additional use of homomorphic signatures allowing non-repudiation. In both papers, in-network data aggregation is performed, i.e., all meters participate in the aggregation process by carrying out the following steps.

1) Each meter i encrypts its measurement with the supplier's public key $\mathsf{Enc}(m_i)$.
2) The meters aggregate their encrypted measurements with the encrypted measurements received from other meters. More precisely, the meter i aggregates its measurement with the measurements received from meters $i-1$ and $i-2$, by computing $\mathsf{Enc}(m_i) \cdot \mathsf{Enc}(m_{i-1}) \cdot \mathsf{Enc}(m_{i-2})$. Due to the homomorphic property of the used encryption scheme, the encrypted measurements are added up, resulting in $\mathsf{Enc}(m_i + m_{i-1} + m_{i-2})$.
3) The meter i sends the encrypted sum of the measurements by meters i, $i-1$, and $i-2$ to the next meter.
4) After all meters aggregated their measurements, the last meter sends the result to the supplier.

Having a set of K meters, the supplier receives

$$\mathsf{Enc}(m_1) \cdot \mathsf{Enc}(m_2) \cdot \cdots \cdot \mathsf{Enc}(m_K) = \mathsf{Enc}(m_1 + m_2 + \cdots + m_K),$$

and decrypts

$$\mathsf{Dec}(\mathsf{Enc}(m_1 + m_2 + \cdots + m_K)) = m_1 + m_2 + \cdots + m_K,$$

the sum of the consumption of all K meters.

B. Smart Meter Billing

Jawurek *et al.* [2] present a privacy-preserving smart meter billing protocol. They use Pedersen Commitments [15] to allow the supplier to verify the correctness of the bill while a Privacy Component (PC) computes the invoice. Each meter is connected to its own PC that communicates with the supplier.

To generate a bill, a meter computes and signs a commitment $Comm_j = \mathsf{Com}(m_j, r_j)$ for the consumption m_j measured at round j. Its respective PC receives the tariff per round t_j from the supplier and receives the signed commitment $Comm_j$ and the opening values m_j and r_j from its meter. After $j \in [1 \ldots L]$ rounds of measurements, the PC has the signed vector of commitments $COMM = (Comm_1, Comm_2, \ldots, Comm_L)$, the measured values $(m_1, m_2, \ldots, m_L)$, the random values used in the commitments $(r_1, r_2, \ldots, r_L)$, and the tariff vector $T = (t_1, t_2, \ldots, t_L)$. Then, the PC computes

$$P = \sum_{j=1}^{L} m_j \cdot t_j$$

and

$$r' = \sum_{j=1}^{L} r_j \cdot t_j.$$

Afterward, the PC sends P, r', and $COMM$ to the supplier. Due to the homomorphic property of Pedersen Commitments, the supplier can generate the commitment to the bill by computing

$$COMM_{Tariff} = \prod_{j=1}^{L} COMM_j^{t_j}.$$

The invoice P computed by the PC is correct, if the supplier can open the commitment $COMM_{Tariff}$ using the opening values P and r'. It proves that $\mathsf{Com}(P, r') = COMM_{Tariff}$. Furthermore, it can detect a cheating PC by verifying the signatures of the commitments.

III. Protocol Providing Data Aggregation and Smart Meter Billing

In this section, we describe our protocol with the requirements, assumptions, and cryptographic primitives. At the end of this section, we have a short discussion about alternative cryptographic primitives and fraud detection.

A. Requirements for Smart Grids

Our protocol addresses the two basic requirements: data aggregation and smart meter billing.

1) Data aggregation: To provide electricity consumption forecast and new services in the power grid, the supplier needs to receive the consolidated consumption from a set of smart meters on demand.

2) Smart meter billing: Customers should be billed accordingly to a dynamic pricing scheme based on supply and demand. It implies that the energy price floats over the time. Furthermore, the values used in the billing should be the same as used in the aggregation.

B. Assumptions

We assume that each meter has a small memory to store data but is able to compute commitments, encryptions, and signatures. This assumption is fairly standard [2], [3]. A signature scheme may be used to provide integrity of the messages sent. Furthermore, we assume that each meter can communicate with at least one other meter and can receive a measurement request from the supplier. Moreover, the initial parameters, such as cryptographic keys, should be installed in a tamper-resistant meter.

C. Cryptographic Primitives

Our protocol uses three cryptographic primitives.

- A homomorphic **commitment scheme** used by meters to commit to their measurements.
- A homomorphic public key **encryption scheme** that allows meters to communicate the corresponding opening values privately to the supplier.
- A **signature scheme** that ensures the integrity and authenticity of the data sent.

A commitment scheme has the functions Com and Unv such that $\mathsf{Com}(m, r) = c \in \mathcal{C}$ commits to a measurement $m \in \mathcal{M}$ using a randomly chosen decommitment value $r \in \mathcal{R}$. The function $\mathsf{Unv}(c, m, r)$ returns the measurement m, if c is a correct commitment to m and r, and false ($\perp$), if not. Further, commitment schemes provide the following security properties.

- **Correctness**, i.e., $\mathsf{Unv}(\mathsf{Com}(m, r), m, r) = m$ for any $m \in \mathcal{M}$ and $r \in \mathcal{R}$.
- **Computational Bindingness**, i.e., given a commitment $c = \mathsf{Com}(m, r)$, the probability to find a second pair $m' \in \mathcal{M}$ and $r' \in \mathcal{R}$ with $m \neq m'$ such that $\mathsf{Com}(m, r) = \mathsf{Com}(m', r')$ is negligible.

In our protocol, we need the commitment scheme to have the following two additional properties.

- **Homomorphic**, i.e., the commitment scheme has to be additively homomorphic such that

  $$\mathsf{Com}(m, r) \cdot_{\mathcal{C}} \mathsf{Com}(m', r') = \mathsf{Com}(m +_{\mathcal{M}} m', r +_{\mathcal{R}} r')$$

  for all $m, m' \in \mathcal{M}$ and $r, r' \in \mathcal{R}$, where $\cdot_{\mathcal{C}}$, $+_{\mathcal{M}}$, and $+_{\mathcal{R}}$ define operations in $\mathcal{C}$, $\mathcal{M}$, and $\mathcal{R}$, respectively. Note that

  $$\mathsf{Com}(m, r)^n = \mathsf{Com}(n \cdot_{\mathcal{M}} m, n \cdot_{\mathcal{R}} r),$$

  for all $n \in \mathbb{N}^*$, $m \in \mathcal{M}$, and $r \in \mathcal{R}$, where $\cdot_{\mathcal{M}}$ and $\cdot_{\mathcal{R}}$ define operations in $\mathcal{M}$ and $\mathcal{R}$, respectively.
- **Unconditional Hidingness**, i.e., having a pair of measurements $m, m' \in \mathcal{M}$, the distribution of the corresponding commitments $\mathsf{Com}(m, r)$ and $\mathsf{Com}(m', r')$ must be identical when $r, r' \in \mathcal{R}$ are chosen uniformly at random.

A homomorphic public key encryption scheme provides the two algorithms Enc and Dec with message space $\mathcal{M}'$. The function $\mathsf{Enc}(m, s)$ encrypts a message $m \in \mathcal{M}'$ using a public key and randomness $s \in \mathcal{R}'$. The function $\mathsf{Dec}(c)$ denotes the decryption of a ciphertext $c = \mathsf{Enc}(m, s)$ to a message $m \in \mathcal{M}'$ with the corresponding secret key. Furthermore, the used scheme should provide semantic security and be homomorphic in $\mathcal{M}'$ such that

$$\mathsf{Enc}(m) \cdot \mathsf{Enc}(m') = \mathsf{Enc}(m +_{\mathcal{M}'} m')$$

and

$$\mathsf{Enc}(m)^n = \mathsf{Enc}(n \cdot_{\mathcal{M}'} m)$$

for all $m, m' \in \mathcal{M}'$ and $n \in \mathbb{N}^*$, where $+_{\mathcal{M}'}$ and $\cdot_{\mathcal{M}'}$ define operations in $\mathcal{M}'$. For our protocol to work, we need two instances of the encryption scheme. One is $\mathsf{Enc}_{\mathcal{M}}$ that is homomorphic over message space $\mathcal{M}$, and another is $\mathsf{Enc}_{\mathcal{R}}$ that is homomorphic over randomization group $\mathcal{R}$ of the used commitment scheme.

Note that having a commitment scheme and a matching encryption scheme providing the requirements above, two measurements $m, m' \in \mathcal{M}$ and two random values $r, r' \in \mathcal{R}$ can be processed as follows: assume that we have two

commitments $c = \mathsf{Com}(m, r)$ and $c' = \mathsf{Com}(m', r')$ and their encrypted opening values ($\mathsf{Enc}_{\mathcal{M}}(m)$, $\mathsf{Enc}_{\mathcal{R}}(r)$) and ($\mathsf{Enc}_{\mathcal{M}}(m'), \mathsf{Enc}_{\mathcal{R}}(r')$). We can compute the encrypted opening values for the commitment $c \cdot c' = \mathsf{Com}(m + m', r + r')$ by

$$\mathsf{Enc}_{\mathcal{M}}(m) \cdot \mathsf{Enc}_{\mathcal{M}}(m') = \mathsf{Enc}_{\mathcal{M}}(m + m')$$

and

$$\mathsf{Enc}_{\mathcal{R}}(r) \cdot \mathsf{Enc}_{\mathcal{R}}(r') = \mathsf{Enc}_{\mathcal{R}}(r + r').$$

As instantiation of the commitment and encryption scheme, we use the construction proposed by Moran and Noar for their Split-Ballot voting system [16, Appendix A]. Their approach uses the Paillier Cryptosystem [14] in combination with slightly adapted Pedersen Commitments [15] such that the measurement and randomization space, $\mathcal{M}$ and $\mathcal{R}$, of the commitment scheme are equal and correspond to the Paillier Cryptosystem over message space $\mathcal{M}'$. In this paper, we just provide high-level information about both primitives. More information regarding their construction and security analysis may be found in [14] and [15].

Using the Paillier Cryptosystem, the measurement $m \in \mathcal{M}' = \mathbb{Z}_N$ is encrypted by $\mathsf{Enc}(m) = \mathsf{Enc}(m, s) = \gamma^m \cdot s^N \mathsf{mod} N^2$, where $s \in \mathcal{R}' = \mathbb{Z}_N^*$ is a random value, $N = p_1 p_2$ is the product of two safe primes, and $\gamma \in \mathbb{Z}_{N^2}^*$ is a randomly chosen generator. The public key is (N, γ), and the corresponding private key is generated using the safe primes p_1 and p_2.

Using Pedersen Commitments, a commitment to measurement $m \in \mathbb{Z}_p^*$ with random value $r \in \mathbb{Z}_p^*$ is generated by $\mathsf{Com}(m, r) = g^m \cdot h^r$, where g and h are two randomly chosen generators in $\mathbb{Z}_p^*$. Like proposed in Split-Ballot, we adapt the Pedersen Commitment scheme such that it takes place in the order N subgroup of $\mathbb{Z}_{4N+1}^*$, where $4N + 1$ is a prime number. Note that computing discrete logarithm is an intractable problem, if p_1 and p_2 are sufficiently large primes.

A signature scheme provides the functions Sig and Ver and uses a key pair consisting of a private key sk to sign messages and a public key pk to verify signatures. More precisely, the algorithm $\mathsf{Sig}_{sk}(W) = u$ generates a signature to message W and the function $\mathsf{Ver}(u, W, pk)$ returns W, if u is a correct signature for W, and $\perp$, if not. In this paper, we do not provide details regarding the used signature scheme, because any secure signature scheme that is currently deployed for smart meters can be used together with our protocol.

D. Protocol Specification

We divided our protocol in three main parts: initialization, data aggregation, and billing verification.

1) Initialization: To set up the protocol, the parameters for the cryptographic primitives have to be chosen and the public parameters have to be stored in the meters. First, the supplier generates two safe primes, p_1 and p_2, and computes its private key and its public key for the Paillier Cryptosystem. Then, the supplier generates the corresponding parameters for the Pedersen Commitment scheme. Note that the meters can verify, whether $N = p_1 p_2$ is a product of safe primes or not, without knowing the primes [17].

2) Data Aggregation: Assume that we have a set of $i \in [1 \dots K]$ meters and $j \in [1 \dots L]$ rounds of measurements, for each round j the following steps are performed.

i) Each meter i
   a) commits to the measurement $m_{i,j}$ in the round j using a random value $r_{i,j}$ by computing $\mathsf{Com}(m_{i,j}, r_{i,j}) = c_{i,j}$;
   b) signs the commitment $\mathsf{Sig}_{sk_i}(c_{i,j}, j) = s_{i,j}$ and sends the result directly to the supplier;
   c) sends the encrypted measurement $\mathsf{Enc}_{\mathcal{M}}(m_{i,j})$, and the encrypted random value $\mathsf{Enc}_{\mathcal{R}}(r_{i,j})$ to the next meter.

ii) Throughout in-network aggregation, the meters
   a) compute the product of all encrypted opening values

$$\prod_{i=1}^{K} \mathsf{Enc}_{\mathcal{M}}(m_{i,j}) = \mathsf{Enc}_{\mathcal{M}}\left(\sum_{i=1}^{K} m_{i,j}\right) = \mathsf{Enc}_{\mathcal{M}}(M_j)$$

   and

$$\prod_{i=1}^{K} \mathsf{Enc}_{\mathcal{R}}(r_{i,j}) = \mathsf{Enc}_{\mathcal{R}}\left(\sum_{i=1}^{K} r_{i,j}\right) = \mathsf{Enc}_{\mathcal{R}}(R_j);$$

   b) send $\mathsf{Enc}_{\mathcal{M}}(M_j)$ and $\mathsf{Enc}_{\mathcal{R}}(R_j)$ to the supplier throughout the last meter.

iii) The supplier
   a) verifies the signature of all received commitments and computes their product

$$\begin{aligned}\prod_{i=1}^{K} c_{i,j} &= \prod_{i=1}^{K} \mathsf{Com}(m_{i,j}, r_{i,j}) \\ &= \mathsf{Com}\left(\sum_{i=1}^{K} m_{i,j}, \sum_{i=1}^{K} r_{i,j}\right) = C_j;\end{aligned}$$

   b) decrypts $\mathsf{Enc}_{\mathcal{M}}(M_j)$ and $\mathsf{Enc}_{\mathcal{R}}(R_j)$ revealing the consolidated consumption of the K households, $M_j = \sum_{i=1}^{K} m_{i,j}$, and the sum of the used decommitment values, $R_j = \sum_{i=1}^{K} r_{i,j}$;
   c) verifies whether M_j and R_j open C_j, more precisely, it checks whether

$$\mathsf{Unv}(C_j, M_j, R_j) \neq \perp;$$

   d) updates the commitments stored from each meter for billing by computing $U_{i,j} = U_{i,j-1} \cdot (c_{i,j})^{t_j}$ for $i \in [1..K]$, where t_j denotes the tariff for electricity in round j. Note that when a new billing interval begins, $U_{i,0}$ is initialized with 1.

iv) The meter i updates the stored billing data by computing

$$\mathsf{Enc}_{\mathcal{M}}(M'_{i,j}) = \mathsf{Enc}_{\mathcal{M}}(M'_{i,j-1}) \cdot \mathsf{Enc}_{\mathcal{M}}(m_{i,j})^{t_j}$$

and

$$\mathsf{Enc}_{\mathcal{R}}(R'_{i,j}) = \mathsf{Enc}_{\mathcal{R}}(R'_{i,j-1}) \cdot \mathsf{Enc}_{\mathcal{R}}(r_{i,j})^{t_j}.$$

At the beginning of a new billing interval, the values $\mathsf{Enc}_{\mathcal{M}}(M'_{i,0})$ and $\mathsf{Enc}_{\mathcal{R}}(R'_{i,0})$ are initialized with 1.

Note that if the supplier can open the commitment, it proves that the consolidated consumption M_j decrypted by the supplier corresponds to the sum of committed measurements.

3) Billing Verification: Smart billing can be carried out similar to the protocol described by Jawurek *et al* [2]. Assume that the supplier collected the commitments sent by a meter i over L rounds. In the following, we describe the billing protocol for meter i where t_j denotes the electricity tariff in round j.

i) The meter i sends the encrypted billing data directly to its supplier, i.e.,

$$\begin{aligned}\mathsf{Enc}_{\mathcal{M}}(M'_i) &= \prod_{j=1}^{L} \mathsf{Enc}_{\mathcal{M}}(m_{i,j})^{t_j} \\ &= \mathsf{Enc}_{\mathcal{M}}\left(\sum_{j=1}^{L} t_j \cdot m_{i,j}\right)\end{aligned}$$

and

$$\begin{aligned}\mathsf{Enc}_{\mathcal{R}}(R'_i) &= \prod_{j=1}^{L} \mathsf{Enc}_{\mathcal{R}}(r_{i,j})^{t_j} \\ &= \mathsf{Enc}_{\mathcal{R}}\left(\sum_{j=1}^{L} t_j \cdot r_{i,j}\right).\end{aligned}$$

ii) The supplier decrypts the data sent by the meters. Following, using the stored commitment

$$U_{i,L} = \prod_{j=1}^{L} (c_{i,j})^{t_j} = \mathsf{Com}\left(\sum_{j=1}^{L} t_j \cdot m_{i,j}, \sum_{j=1}^{L} t_j \cdot r_{i,j}\right).$$

The supplier verifies whether

$$\mathsf{Unv}(U_{i,L}, M'_i, R'_i) \neq \perp .$$

If the supplier can open the commitment, then M'_i is the total amount that the household of meter i has to pay for the rounds 1 to L.

iii) The supplier initializes the commitments stored for billing, i.e.,

$$U_{i,0} = 1, \quad \text{for} \quad i \in [1..K].$$

iv) Each meter i resets its opening values, i.e.,

$$\mathsf{Enc}_{\mathcal{M}}(M'_{i,0}) = \mathsf{Enc}_{\mathcal{R}}(R'_{i,0}) = 1.$$

E. Alternative Cryptographic Primitives

Our protocol requires a homomorphic commitment scheme with a matching homomorphic encryption scheme. The solution by Moran and Naor [16] is currently the best-evaluated instantiation and uses primitives proposed for smart meters before. However, this solution has a disadvantage. It uses Paillier Cryptosystem that is time-consuming due to the modular exponentiation over integer numbers [18]. Another approach to speed up the processing time is to use the pairing based scheme proposed by Cuvelier *et al*. [19].

F. Fraud Detection

The supplier can compare the amount of generated electricity with the consolidated consumption computed throughout in-network data aggregation. Thus, the supplier can detect energy loss and fraud. The property of billing verification allows the customers to ensure that the supplier is charging them correctly. Furthermore, the supplier is assured that the meters cannot change the value of a measurement after they sent the commitment. Note that with Step 2-iii-c the supplier can verify the consistency between the received commitments and its measured consumption for a set of meters. This allows to detect measurement errors before the bill is computed.

IV. Comparison with Related Work

Security and privacy of smart grids are well-studied topics, e.g., [2]–[12]. The first paper in this list addresses the problem of smart meter billing, while the others address the problem of data aggregation.

To the best of our knowledge, only [13] addresses both problems. They use anonymous identifiers for the data aggregation and non-anonymous identifiers for billing. However, their protocol relies on a Trusted Third Party (TTP) and has an undesirable trade-off between privacy and efficiency [20]. Therefore, protocols based on homomorphic encryption, e.g., [3], [5], [6], which use the Paillier Cryptosystem, are more interesting, because they do not have such a trade-off and do not require a TTP. However, whoever has the key of the supplier can decrypt single measurements. In addition, protocols based on homomorphic encryption need to protect the encrypted measurement against malleability, for instance, by using homomorphic signatures as proposed by [4].

Protocols based on DC-Nets are non-scalable with respect to the number of meters, e.g., [7]–[9]. Some protocols are also non-scalable with respect to the bit length of the measurements, e.g., [10]. Motivated by such drawback, [11] describes a better solution to the problem of data aggregation. However, [11] as well as [10], [12] are based on TTP to set up the initial phase.

Our protocol is scalable in the number of meters and their measurements. We address the problem of data aggregation and smart meter billing without a TTP. We use a very specific cryptographic component, a homomorphic commitment scheme with matching homomorphic encryption scheme, which allows to privately process the data and publicly verify the outcome. We summarize the comparison of this paper with related work in Table I.

V. Conclusions

Smart grids are a very important research topic, because they allow to use resources more efficiently and to reduce the negative impact on the environment. Since smart grids would be deployed on a large scale, the costs are an important factor. However, while searching for efficient solutions, privacy should not only take a back seat. Since the implementation is still in progress, there is the opportunity to introduce privacy

Table I
COMPARISON BETWEEN PRIVACY-ENHANCING PROTOCOLS FOR SMART GRIDS.

Protocol	Data Aggregation	Smart Meter Billing	Initialization without TTP	Based on
[2]	No	Yes	Yes	Commitment
[3]	Yes	No	Yes	Paillier Cryptosystem
[4]	Yes	No	Yes	Homomorphic Signature
[5]	Yes	No	Yes	Paillier Cryptosystem
[6]	Yes	No	Yes	Paillier Cryptosystem
[7]	Yes	No	Yes	DC-Nets and Paillier Cryptosystem
[8]	Yes	No	Yes	DC-Nets
[9]	Yes	No	Yes	DC-Nets
[10]	Yes	No	No	Discrete Logarithm
[11]	Yes	No	No	Variation of Paillier Cryptosystem
[12]	Yes	No	No	Variation of Paillier Cryptosystem
[13]	Yes	Yes	No	Pseudonymous
Our model	Yes	Yes	Yes	Homomorphic Commitment with Homomorphic Encryption

by design and to prevent that established solutions must be rebuilt later in order to protect customers' privacy.

Therefore, we have presented a protocol that describes how privacy-preserving data aggregation and smart billing can be performed. The provided protocol grants the benefits of data aggregation of the measurements and allows billing with time-based pricing. In addition, customers' privacy is guaranteed by a homomorphic encryption with a homomorphic commitment scheme. All data sent are encrypted to prevent that the components in the communication network can violate customers' privacy and protect against eavesdropping on the communication channels. The only data that the supplier receives by each meter individually are unconditional hiding commitments providing even everlasting privacy. Moreover, we only require changes on the software. Therefore, our protocol can be used in conjunction with many existing smart grid communication models. However, there is still room for improvements. One example is to find more efficient cryptographic primitives that can be used to securely process the measurements in a privacy-preserving way. Another example is to improve the protocol with respect to the communication complexity to allow for more fine-grained consumption profiles. We plan to work on these matters in the future.

REFERENCES

[1] U. Greveler, B. Justus, and D. Löhr, "Multimedia content identification through smart meter power usage profiles," in *Computers, Privacy and Data Protection (CPDP 2012)*, 2012.

[2] M. Jawurek, M. Johns, and F. Kerschbaum, "Plug-in privacy for smart metering billing," in *Privacy Enhancing Technologies*, ser. Lecture Notes in Computer Science. Springer, 2011, vol. 6794, pp. 192–210.

[3] F. Li, B. Luo, and P. Liu, "Secure information aggregation for smart grids using homomorphic encryption," in *Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference on*, Oct 2010, pp. 327–332.

[4] F. Li and B. Luo, "Preserving data integrity for smart grid data aggregation," in *Smart Grid Communications (SmartGridComm), 2012 IEEE Third International Conference on*, 2012, pp. 366–371.

[5] S. Ruj and A. Nayak, "A decentralized security framework for data aggregation and access control in smart grids," *Smart Grid, IEEE Transactions on*, vol. 4, no. 1, pp. 196–205, March 2013.

[6] D. Seo, H. Lee, and A. Perrig, "Secure and efficient capability-based power management in the smart grid," in *Parallel and Distributed Processing with Applications Workshops (ISPAW), 2011 9^{th} IEEE International Symposium on*, May 2011, pp. 119–126.

[7] Z. Erkin and G. Tsudik, "Private computation of spatial and temporal power consumption with smart meters," in *ACNS*, ser. Lecture Notes in Computer Science, F. Bao, P. Samarati, and J. Zhou, Eds., vol. 7341. Springer, 2012, pp. 561–577.

[8] K. Kursawe, G. Danezis, and M. Kohlweiss, "Privacy-friendly aggregation for the smart-grid," in *Privacy Enhancing Technologies*, ser. Lecture Notes in Computer Science. Springer, 2011, vol. 6794, pp. 175–191.

[9] G. Ács and C. Castelluccia, "I have a dream! (differentially private smart metering)," in *Information Hiding*, ser. Lecture Notes in Computer Science. Springer, 2011, vol. 6958, pp. 118–132.

[10] E. Shi, T.-H. H. Chan, E. G. Rieffel, R. Chow, and D. Song, "Privacy-preserving aggregation of time-series data," in *NDSS*. The Internet Society, 2011.

[11] M. Joye and B. Libert, Eds., *A Scalable Scheme for Privacy-Preserving Aggregation of Time-Series Data*. Springer, 2013.

[12] R. Lu, X. Liang, X. Li, X. Lin, and X. Shen, "Eppa: An efficient and privacy-preserving aggregation scheme for secure smart grid communications," *Parallel and Distributed Systems, IEEE Transactions on*, vol. PP, no. 99, p. 1, 2012.

[13] C. Efthymiou and G. Kalogridis, "Smart grid privacy via anonymization of smart metering data," in *Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference on*, Oct 2010, pp. 238–243.

[14] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *Advances in Cryptology - EUROCRYPT 1999*, ser. Lecture Notes in Computer Science, 1999, vol. 1592, pp. 223–238.

[15] T. P. Pedersen, "Non-interactive and information-theoretic secure verifiable secret sharing," in *Proceedings of the 11th Annual International Cryptology Conference on Advances in Cryptology*, ser. CRYPTO '91. London, UK, UK: Springer-Verlag, 1992, pp. 129–140.

[16] T. Moran and M. Naor, "Split-ballot voting: Everlasting privacy with distributed trust," *ACM Trans. Inf. Syst. Secur.*, vol. 13, no. 2, 2010.

[17] J. Camenisch and M. Michels, "Proving in zero-knowledge that a number is the product of two safe primes," in *Proceedings of the 17th international conference on Theory and application of cryptographic techniques*, ser. EUROCRYPT'99. Berlin, Heidelberg: Springer-Verlag, 1999, pp. 107–122.

[18] P. Lara, F. Borges, R. Portugal, and N. Nedjah, "Parallel modular exponentiation using load balancing without precomputation," *Journal of Computer and System Sciences*, vol. 78, no. 2, pp. 575–582, 2012.

[19] E. Cuvelier, O. Pereira, and T. Peters, "Election verifiability or ballot privacy: Do we need to choose?" in *ESORICS*, ser. Lecture Notes in Computer Science, J. Crampton, S. Jajodia, and K. Mayes, Eds., vol. 8134. Springer, 2013, pp. 481–498.

[20] F. Borges, L. Martucci, and M. Muhlhauser, "Analysis of privacy-enhancing protocols based on anonymity networks," in *Smart Grid Communications (SmartGridComm), 2012 IEEE Third International Conference on*, Nov 2012, pp. 378–383.

[21] D. Demirel, J. Van De Graaf, and R. Araújo, "Improving helios with everlasting privacy towards the public," in *Proceedings of the 2012 international conference on Electronic Voting Technology/Workshop on Trustworthy Elections*, ser. EVT/WOTE'12. Berkeley, CA, USA: USENIX Association, 2012, pp. 8–8.