A Lightweight Privacy-Preserved Spatial and Temporal Aggregation of Energy Data

Sye Loong Keoh[†], Yi Han Ang[†*] and Zhaohui Tang[‡]
[†]School of Computing Science, University of Glasgow
[*]Singapore Institute of Technology
[‡]School of Infocomm, Republic Polytechnic Singapore
Email: SyeLoong.Keoh@glasgow.ac.uk, angyihan@hotmail.com, Linda_Tang@rp.edu.sg

***Abstract*—Smart grid provides fine-grained real time energy consumption, and it is able to improve the efficiency of energy management. It enables the collection of energy consumption data from consumer and hence has raised serious privacy concerns. Energy consumption data, a form of personal information that reveals behavioral patterns can be used to identify electrical appliances being used by the user through the electricity load signature, thus making it possible to further reveal the residency pattern of a consumer's household or appliances usage habit.**

This paper proposes to enhance the privacy of energy consumption data by enabling the utility to retrieve the aggregated *spatial* and *temporal* consumption without revealing individual energy consumption. We use a lightweight cryptographic mechanism to mask the energy consumption data by adding random noises to each energy reading and use Paillier's additive homomorphic encryption to protect the noises. When summing up the masked energy consumption data for both *Spatial* and *Temporal* aggregation, the noises cancel out each other, hence resulting in either the total sum of energy consumed in a neighbourhood at a particular time, or the total sum of energy consumed by a household in a day. No third party is able to derive the energy consumption pattern of a household in real time. A proof-of-concept was implemented to demonstrate the feasibility of the system, and the results show that the system can be efficiently deployed on a low-cost computing platform.

I. Introduction

Smart grid is a two way communication electrical grid that can link an array of energy sources [17]. It is capable of recording fine-grained real time consumption data, and handling bi-directional flow of energy, thus increasing the efficiency of energy management. Energy consumption data is usually aggregated by a concentrator, and then sent to the utility for monthly electrical billing, energy distribution automation, load monitoring and forecasting [5], [12]. With this, the utility will have a better understanding in electrical demand during daily peaks in order to keep up with the uprising trend of electrical energy demand. This can help to prevent insufficient allocation of electricity that could lead to blackouts and power outage, and at the same time better integrate the renewable energy into the energy grid.

The ability to collect energy consumption data from consumer in real time has raised serious privacy concerns. Energy consumption data can be seen as a form of personal information that reveals behavioral patterns and it can be used to identify electrical appliances [17], [6] being used in a household. A number of research studies have shown that through the analysis of electrical consumption data, we can easily determine electrical events, identify electrical appliances and eventually know how these appliances are been used [13], [3], [19], [4]. Each electrical appliance has its unique load signature, and through these consumption data, attackers are able to visualize consumption behavior, such as the sleep and wake habit in a consumer's household. With a long period of monitoring, it is possible to further reveal the residency pattern of a consumer's household or appliances usage habit. To a certain extent, appliance load monitoring can be misused by attackers to forecast a consumer's household activities, posing possible risks and threats to consumers.

On the other hand, the energy data is useful for marketing purposes. Many industries use these data to *profile* their users, identify potential customers as part of their marketing strategy. The problem is further exacerbated by the fact that smart grid is not protected against eavesdropping, making it possible for the public or a third party to get hold of these personal information [2], [3], [22].

This paper proposes a lightweight cryptographic mechanism to mask the energy consumption data by adding random noises to each energy consumption data before it is transmitted to the utility. When summing up the masked energy consumption data for both *Spatial* and *Temporal* aggregation, the noises cancel out each other, hence resulting in either the total sum of energy consumed in a neighbourhood at a particular time, or the total sum of energy consumed by a household in a day. We further protect the leakage of individual random noise by using homomorphic encryption such that the utility only has access to the sum of all random noises in order to derive the actual energy consumed. Hence, no third party is able to derive the energy consumption pattern of a household in real time.

This paper is organized as follows: Section II reviews the literature and related work. Section III describes a smart grid use case and requirements to preserve the user's privacy. Section IV presents the proposed lightweight *Spatial* and *Temporal* data aggregation scheme. In Section V, we present the implementation details, while Section VI describes the evaluation results. Finally, we present an informal security analysis in Section VII and conclude the paper with future work in Section VIII.

II. Background and Related Work

There are various privacy enhancing techniques to protect the user's privacy in smart grid [20], and we briefly describe them in this section.

A. *Obfuscating*

Obfuscating energy consumption can be done using alternative energy sources [11], or if there's a energy storage through flattening or reshaping strategies . Flattening energy consumption requires a rechargeable battery that will charge and discharge to modify the energy consumption at strategic timing. The purpose is to prevent the identification of electrical events or usage patterns of electrical appliances from the energy consumption. Energy consumption will be modified to a constant flat energy consumption. The rechargeable battery can choose to charge or discharge for each level in net demand [7].

Reshaping of energy consumption can be used to hide characteristics of electrical appliances. It modifies the energy consumption such that appliances load signature cannot be easily discerned. Similar to flattening of energy consumption, this method requires a rechargeable battery or other power generation devices to be used. There are three different reshaping strategies, namely (1) *Hiding*, it will provide all the required energy from the battery and recharged later. (2) *Obfuscating*, it will provide all the required energy from the battery but the battery will recharge itself at different short interval. (3) *Smoothing*, it will use part of the required energy from the battery while the smart grid will provide the remaining energy required [14]. However, these techniques have its limitations and drawbacks. Flattening of energy consumption is limited by the maximum charging and discharging rate of the rechargeable battery or the capacity of the battery [7]. The cost for this implementation is very high, furthermore rechargeable battery requires maintenance and have a limited lifetime [14].

B. *Homomorphic Encryption*

Homomorphic encryption allows aggregation of encrypted energy consumption data without the need of decrypting them [21], [9], [10]. In a simpler term, energy consumption data from the smart meters are first encrypted using homomorphic encryption, when they reach the concentrator, the total sum of the energy consumption is computed through the aggregation of the encrypted readings, in which addition is performed over encrypted data. When the aggregated data is decrypted by the utility, it would be the total sum of energy consumption. The purpose is to allow utility to retrieve aggregation of the energy consumption for the necessary tasks, such as billing and load monitoring, without revealing the original energy consumption data.

Garcia and Jacobs [8] proposed to integrate secret sharing and homomorphic encryption for smart metering. They proposed that every smart meter picks a random amount of smart meters and prepares a share of its own measurement with each selected smart meter. Among all the shares, one of the shares is to be kept locally and the other shares will be encrypted with the public key of the smart meters involved. A substation connected to the smart meters will then collect all encrypted shares and multiply the encrypted shares that belong to the same smart meter recipient before sending the results to the smart meter. The smart meter will decrypt the results and adds the locally stored share to the decrypted results. The substation will collect all aggregated results and sum them up to compute the total energy consumption. The proposed approach can provide confidentiality to all the consumption from the smart meter, however, the smart meter would have to transfer data four times before the substation can receive the total energy consumption, which makes the proposed approach inefficient and incurs additional communication overheads.

C. *Masking*

Kursawe et al. [16] proposed to mask energy consumption with secret sharing. In a group of the smart meters, one will be elected as the leader. Before sending the energy consumption, all smart meters, except the leader, will generate a random secret. The random secret will be encrypted and sent to the leader. The leader will then compute its secret using all random secrets from the smart meters, so that the sum of all the secrets will become zero. The smart meters will mask the energy consumption using the secret and send it to the aggregator. The aggregator will sum up all the masked energy consumption to retrieve the aggregated energy consumption from all smart meters. The secrets will cancel out each other during the summing process. This scheme requires all smart meters, except for the leader, to transfer data at least twice to send out the energy consumption. In addition, the scheme is an expensive protocol due to the requirement of exchanging a new Diffie Hellman public key among all the smart meters for every transmission.

III. Smart Grid Use Case and Requirements

In a typical deployment, a smart meter is installed in each household to measure energy use of electrical appliances at home. Within a neighborhood, e.g., a HDB Block in Singapore, or a residential area, a wireless mesh network is set up to interconnect all the smart meters from each home to a concentrator, so that the collected energy usage data can be aggregated before they are forwarded to the Network Operating Centre (called a Head-end) through a WAN interface. The smart meter from every household then periodically reports energy usage to a concentrator, which subsequently aggregate the data in order to minimize bandwidth consumption and to reduce the number of messages to be transmitted, therefore improving efficiency. The aggregated data are then forwarded to the utility data management center for billing and analysis.

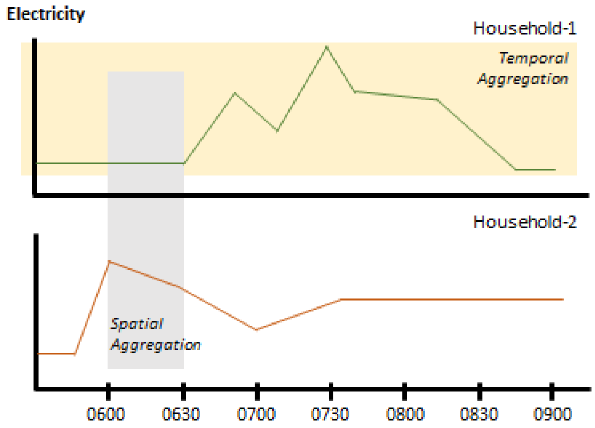

Fig. 1. Spatial and Temporal Aggregation of Energy Consumption

In this setting, we are interested in achieving two goals, namely (1) The utility has an overview of the energy consumption of each neighbourhood on an hourly basis so that energy distribution and planning can be achieved more accurately. However, the utility should not have any knowledge of the energy consumed by an individual household hourly. (2) The utility is able to obtain a 24-hour energy consumption of each household, so that billing can be done accurately. As illustrated in Figure 1, the first goal is known as *Spatial Aggregation*, it allows the utility to know the total energy consumption a group of consumers in a neighbourhood has used at a fixed time. Utility uses these data to adjust the energy distribution to different groups of consumers. Countries with renewable energy can use these data to balance out the usage of renewable energy and non-renewable energy, which are more expensive and less environment friendly. While *Temporal Aggregation* allows the utility to know the total energy consumption a consumer has used for a fixed period of time, and uses these data for billing purposes.

IV. A Lightweight Privacy-Preserved Spatial and Temporal Data Aggregation

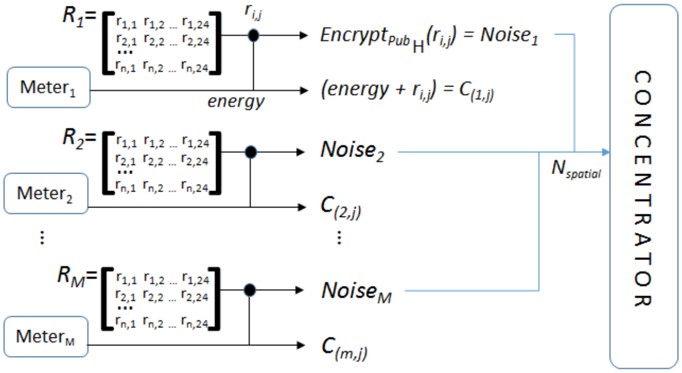
R_1= [$r_{1,1}$ $r_{1,2}$ ... $r_{1,24}$ / $r_{2,1}$ $r_{2,2}$ ... $r_{2,24}$ / ... / $r_{n,1}$ $r_{n,2}$ ... $r_{n,24}$]
$r_{i,j}$
$Encrypt_{Pub_H}(r_{i,j})$ = $Noise_1$
Meter$_1$
energy
(energy + $r_{i,j}$) = $C_{(1,j)}$
R_2= [$r_{1,1}$ $r_{1,2}$ ... $r_{1,24}$ / $r_{2,1}$ $r_{2,2}$ ... $r_{2,24}$ / ... / $r_{n,1}$ $r_{n,2}$ ... $r_{n,24}$]
$Noise_2$
$N_{spatial}$
Meter$_2$
$C_{(2,j)}$
R_M= [$r_{1,1}$ $r_{1,2}$ ... $r_{1,24}$ / $r_{2,1}$ $r_{2,2}$ ... $r_{2,24}$ / ... / $r_{n,1}$ $r_{n,2}$ ... $r_{n,24}$]
$Noise_M$
Meter$_M$
$C_{(m,j)}$
CONCENTRATOR

Fig. 2. Overview of the privacy-preserved data aggregation

We propose a security protocol to commission a privacy-preserved smart metering infrastructure and to protect the user's privacy based on additive data masking [16] and homomorphic encryption [18]. As shown in Figure 2, the idea is similar to adding a random noise to each energy usage data reported by the smart meter. Each smart meter generates a random noise matrix with a special property that the sum all elements in a row is equivalent to zero. Each random noise in the matrix is for one time use only. The smart meter reports energy data periodically, by first adding a random noise to mask the energy reading, by computing C. At the same time, the random noise is encrypted using homomorphic scheme, i.e., $Noise$, and this allows the concentrator to perform additive homomorphic encryption on all the $Noises$ to compute $N_{spatial}$ at a given time interval.

For *Spatial Aggregation*, the utility upon receiving all the masked readings (C_1, ..., C_m) from the smart meters at a given time interval, sums them up and deducts the decrypted sum of the random noises produced by the concentrator, ($Decrypt_{Priv_H}(N_{spatial})$), thus knowing the energy load in the neighbourhood in real time. As for *Temporal Aggregation*, all the readings from a household ($C_{1,1}, ..., C_{1,24}$) are summed up at the end of the day to obtain the total energy consumed for the day. When summing up the energy data, the noises added by smart meters will cancel out each other, thus summing up to zero (0), enabling the utility to obtain the aggregated *Temporal* readings.

A. Commissioning and System Setup

Prior to deployment, while commissioning the smart metering infrastructure, it is assumed that each smart meter is configured and provisioned with a secret-key, K_{sm}. This secret-key is unique to each smart meter and is shared with the utility. This secret-key is used for identification purpose, allowing the utility to identify each household uniquely. Additionally, Paillier homomorphic cryptosystem [18] is set up such that a keypair is generated and the public-key, Pub_H is distributed to all smart meters, while the $Priv_H$ is kept by the utility. The concentrator is assumed to have no knowledge of the $Priv_H$, and its main responsiblity is to perform data aggregation.

Each smart meter is responsible for generating a random noise matrix to be used to mask the energy usage, and this matrix should be kept secret by the smart meter itself in order to prevent leakage. The following shows the random noise matrix, R_i generated dynamically by each smart meter, where i denotes the smart meter id. The row denotes the random noises to be used to mask the energy data per day (Day $1, ..., n$), while the column denotes the time interval in a day $(1, 2, 3, ..., 24)$ assuming that the smart meter reports the energy consumption on an hourly basis. Therefore, $r_{1,1}$ is used on Day 1 at 0000 AM, $r_{1,2}$ on Day 1 at 0100 AM, and so on and so forth. While $r_{2,1}$ is used on Day 2 at 0000 AM, etc. The random noise is for one-time use, and a new random noise matrix is generated once all the rows in R_i have been used up.

$$R_i = \begin{bmatrix} r_{1,1} & r_{1,2} & r_{1,3} & \dots & r_{1,24} \\ r_{2,1} & r_{2,2} & r_{2,3} & \dots & r_{2,24} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ r_{n,1} & r_{n,2} & r_{n,3} & \dots & r_{n,24} \end{bmatrix}$$

The property of this random noise matrix, R_i is that the sum of each row and each column is zero respectively, i.e., sum of ($r_{1,1}$, $r_{1,2}$, ..., $r_{1,24}$) is zero, and sum of ($r_{1,1}$, $r_{2,1}$, ..., $r_{n,1}$) is zero as well. R is generated column by column. Since it is assumed that the energy consumption is to be reported on an hourly basis, the first 23 columns are generated using the procedure below and it is iterated for each column, i.e., 23 iterations:

Step 1:
A random number matrix of N X N is first generated, where N is the number of days.

$$\begin{bmatrix} p_{1,1} & p_{1,2} & \dots & p_{1,n} \\ p_{2,1} & p_{2,2} & \dots & p_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ p_{n,1} & p_{n,2} & \dots & p_{n,n} \end{bmatrix}$$

Step 2:
To compute each element $r_{i,p}$ in R where p = 1, 2,... N and i = 1, 2,... N where N is the number of days, Equation 1 is used to generate all the random noises for column p, which

contains N random noises. The value β in Equation 1 is a large number generated randomly.

$$r_{i,p} = \beta + \sum_{j=1} p_{(i,j \to N)} - \sum_{j=1} p_{(j \to N,i)} \tag{1}$$

Step 3:
At the end of the 23 iterations, the first 23 columns in R have been computed, Equation 2 is used to compute the last column of the random noise matrix. Thus completing the random noise matrix generation.

$$r_{i,24} = \beta - \sum_{j=1}^{k=23} r_{(i,j \to k)} \tag{2}$$

B. Operation: Energy Usage Reporting

1) Temporal Aggregation: As each smart meter has its own R_i, it uses this random noise matrix to protect its privacy. When reporting the energy data at each interval, say j, the smart meter adds the corresponding noise to the energy data. For example, on Day 1, the first row of random noise matrix is used to mask the energy data. The message is formatted as (*Source* | *Timestamp* | *Masked Energy Data*), and then encrypted with its secret key, K_{sm} before it is sent to the utility for processing and billing. The masked energy data, $C_{k,j}$ is computed by smart meter k at each hour j as follows:

$$C_{(k,j)} = (energy + r_{i,j}) \mod \beta \tag{3}$$

where $j = 1, 2, ..., 24$.

Subsequently, $C_{(k,j)}$ is encrypted using the secret key, K_{sm-k} that is only known between the smart meter k and the utility.

2) Spatial Aggregation: In order to facilitate *Spatial Aggregation* of energy data, while reporting the energy reading, each smart meter k also uses homomorphic encryption to encrypt the random noise, $r_{i,j}$ it added to its energy reading at each interval j, and then sends the encrypted noise to the concentrator.

$$Noise_{(k)} = Encrypt_{Pub_H}(r_{i,j} \mod \beta) \tag{4}$$

where i = 1, 2, 3,..., n and $j = 1, 2, ..., 24$.

The concentrator then computes the sum of all the $Noise_{(k)}$ received from all smart meters in a neighbourhood. In the Paillier's scheme, the summation of encrypted values can be computed using the dot product of two encrypted data, i.e., $Encrypt_{Pub_H}(r_{1,1}$ + $r_{1,2})$ = $Encrypt_{Pub_H}(r_{1,1})$ $\cdot$ $Encrypt_{Pub_H}(r_{1,1})$. This is also equivalent to $(Noise_1 \cdot Noise_2)$. In this way, the concentrator does not have access to the value of the individual random noise, but it is able to sum them up by performing the following:

$$N_{spatial} = \prod_{k=1}^{M} Noise_{(k)} \tag{5}$$

where M denotes number of smart meters

The encrypted sum of the noises from all smart meters denoted as $N_{spatial}$ at a time interval is sent to the utility.

C. Verification and Billing

1) Spatial Aggregation: With this scheme, in real time, the utility is able to monitor the energy load of a neighbourhood on an hourly basis through the data received from *Spatial Aggregation*, which means that the utillity will receive all the masked readings, $C_{(k,j)}$ from all smart meters, and $N_{spatial}$ from the concentrator. For each neighbourhood, the masked readings are added up together, hence yielding the total sum of masked consumption. As the utility has the private key of the Pailler's homomorphic encryption scheme, it is able to retrieve the sum of all random noises at hour j by decrypting $N_{spatial}$.

$$SUM_j = Decrypt_{Priv_H}(N_{spatial}) \tag{6}$$

In order to derive the total energy in a neighbourhood, the utility deducts the total masked consumption with SUM_j as this would cancel out the random noises added by the smart meters, thus leaving a multiple of β in the total sum of energy consumption. The result must then be modulus by β in order to retrieve the original total sum of energy consumption data. In essense, the utility computes the sum of all masked energy data from smart meter k where k = 1,2,..., M for hour j as follows:

$$Spatial_{(j)} = \sum_{k=1}^{k=M} C_{(k,j)} - SUM_j \mod \beta \tag{7}$$

2) Temporal Aggregation: The utility is also able to determine the total energy usage for a household by computing the sum of all masked data recorded by a smart meter k in a day, i.e., 24 hours. With that, the utility is able to correctly bill the customer, without the ability to know the enegy consumption of a particular household on an hourly basis.

$$Temporal_{(k)} = \sum_{j=1}^{j=24} C_{(k,j)} \mod \beta \tag{8}$$

3) Cross-check and Verification: It is important to ensure that the random noises used for all the smart meters have not been tampered with or modified in a malicious manner. As each row of random noises in the matrix sums up to zero, the summation of all random noises of M smart meters must yield zero as well. A verification can be performed by the utility to ensure the following:

$$\sum_{j=1}^{j=24} SUM_j \mod \beta = 0 \tag{9}$$

D. Renewal of Random Noise Matrix, R_i

As the random noise matrix is generated by the smart meter itself, whenever all the random noises have been used up, a new random noise matrix should be generated using the method and procedures as described in Section IV-A.

V. Implementation

A proof-of-concept (PoC) had been implemented to demonstrate the feasiblity of the proposed lightweight privacy-preserved *spatial* and *temporal* data aggregation. A typical smart meter does not have high computing power or flash memory. Thus, we developed our PoC on a Raspberry Pi Model B, in order to simulate the smart meter. Raspberry Pi is a low-end computer with low computing capability, allowing this prototype to have the closest test results when deploying the application. Similarly, we employed a Raspberry Pi as the concentrator.

The module for generating the random noise matrix was implemented in Java, and the communication between the smart meters and the concentrator was via ethernet, other communication means such as Zigbee, Wi-Fi, IEEE 802.15.4 should be used in the real deployment. Energy data are sent using UDP, and we implemented a simple reliable UDP protocol to ensure that the energy consumption data is sent and received using *request-reply-ack* protocol.

As for the communication between the concentrator and the utility. A TCP connection was established between them. This communication channel could optionally be secured using Transport Layer Security (TLS). However, in this prototype, since the masked energy data is encrypted, it is not necessary to deploy TLS. Encryption of masked energy consumption data was based on AES with a key length of 128-bit.

As for the Paillier Homomorphic Encryption scheme, we plan to integrate the existing implementation provided in [15] with the proposed energy masking scheme. A homomorphic encryption keypair can be generated by computing $n = pq$ and $\lambda = lcm(p-1, q-1)$ where two large prime numbers, p and q are choosen randomly and independently from each other such that $gcd(pq, (p-1)(q-1) = 1$. A random integer g is generated where $g \in \mathbb{Z}^*_{n^2}$, and hence the public-key is derived as (n, g) and the private-key is (λ, μ) where $\mu = (L(g^\lambda \mod n^2))^{-1}$ mod n, and function L is defined as $L(u) = \frac{u-1}{n}$.

Additive homomorphic encryption is performed (on both the smart meter and the concentrator) by first selecting a random number $r \in \mathbb{Z}^*_n$ and computing the ciphertext, c as follows:

$$c = g^m \cdot r^n \mod n^2 \tag{10}$$

Decryption is done (at the utility) to compute the plaintext as follows:

$$m = L(c^\lambda \mod n^2) \cdot \mu \mod n \tag{11}$$

VI. Evaluation and Results

This section discusses the evaluation of the system developed as well as the performance of the protocol.

To ensure that the developed system does not excessively overload the smart meter, we monitored the CPU usage and memory consumption on the Raspberry PI for generating the random noise matrices. Table I shows the usage of CPU for both the smart meter and the concentrator on a Raspberry Pi.

	Memory	CPU
Energy reporting on a smart meter	2.8%	60 - 70%
Random Noise Matrix Generation	3.3%	80 - 90%

TABLE I. Usage of CPU and RAM on Raspberry Pi

While performing the energy data masking (excludes the homomorphic encryption operation), the memory usage in smart meter is approximately 2.8%, while the memory usage for generating the random noise matrix is 3.3%. The values in the table are the mean average results from a series of monitoring while the application was running. However, the CPU usage for the process of generating random noise matrix on each smart meter seems to be rather high, using 80 to 90% of the CPU most of the time.

Number of rows in R_i	Time	No of random noises per row
2	130 ms	24
25	1206 ms	24
50	2638 ms	24
50	6197 ms	30

TABLE II. Time Taken to Compute Random Noise Matrix with Different Number of Smart Meters

The time taken to generate the random noise matrix with varing dimensions is shown in Table II. When generating the random noise matrix for two days consumption (i.e., 2 rows), the system took 130 ms. However, when the number of rows is increased to 25, the time taken to generate the noise increased to approximately 1200 ms. The time taken would be further increased when there were 50 rows.

VII. Security Analysis and Discussion

A. Privacy Protection

From a smart grid perspective, it is the consumers' requirement to protect their own privacy from the utility provider, hence it is difficult to persuade the utility provider to deploy a privacy-enhanced security protocol in this context. Additionally, the utlity provider is reluctant to use privacy-enhanced technologies as it reduces the data that can be collected to profile their customers for marketing purposes.

As a result, the best place to deploy privacy-enhanced technology is at the smart meter itself where it is located at the consumer domain. Since the consumers would like their energy readings to be protected, they can exploit the proposed system to generate the random noise matrix locally on the smart meter. This ensures that no other entities in the smart grid infrastructure have knowledge of the random noises added to the energy readings reported periodically. The concentrator only has access to the encrypted random noises and performs homomorphic operations on the encrypted noises. The utility only has access to the sum of the random noises from all smart meters at a particular time interval, and it would be difficult to derive the individual value of each random noise.

B. Protection of Random Noises

The privacy of energy reading is protected through masking using a random noise. The random noise is then protected using homomorphic encryption, so that only the sum of the random noises can be computed and then deducted by the utility to derive *Spatial* energy data. One can argue that there is a leak of information in that the sum of random noises is known to the utility, however we argue that as the noise added to the energy reading is random, the sum of these noises is considered as a random number that is unpredictable to the utility as well.

The reason that homomorphic encryption was used to protect the random noises instead of the energy readings is because we can use the masking scheme to obsfucate energy data both *Spatial*ly and *Temporal*ly in which case it requires less computation. As compared to the scheme where only homomorphic encryption is used to protect *spatial aggregation* and *temporal aggregation* of energy data, it would incur a lot of multiplication operations of encrypted data on the concentrator, if not the smart meters.

C. Homomorphic Operation on the Concentrator

The current architecture requires that the additive homomorphic encryption of random noises is performed by the concentrator with the assumption that it does not have the knowledge of the homomorphic private key. As the concentrator device belongs to the utility provider, it is possible that the utility provider may deploy its private-key onto the concentrator so that it is able to decrypt the individual noise.

A mitigation plan is only allows the smart meters to perform the additive homomorphic encryption operation by relaying the encrypted noise from one smart meter to another without involving the concentrator. In this way, the random noises generated by each smart meter will only be known to itself.

VIII. Conclusions and Future Works

This paper proposed a lightweight privacy-preserved aggregation of energy usage data using a combination of masking method and homomorphic encryption in order to allow for *Spatial* and *Temporal* aggregation. We have developed the masking technique based on random noises that allows utility to retrieve aggregated spatial consumption for monitoring purposes as well as aggregated temporal consumption for billing purposes while hiding individual energy consumption from any parties. The benefits of the proposed system are mainly (1) enabling the utility to accurately monitor the energy load of an area, e.g., a neighbourhood in real time, although it has no knowledge of the energy load of each individual household. (2) enabling the utility to correctly bill its customers, such that the system can accurately report the total sum of energy consumption of a household for a day, without revealing the electricity load signatures. Through our PoC, the commissioning process and the generation of random noise matrix R_i as well as the data masking process is sufficiently lightweight to be deployed.

The next step is to further integrate the Paillier's homomorphic implementation to protect the random noises especially for *Spatial Aggregation*. Furthermore, we would like to further evaluate our privacy protection protocol using the real energy load dataset available from the Singapore Energy Market Authority (EMA) [1]. It provides half-hourly system demand data and this will enable us to further evaluate our system. Having a large β while performing the masking is extremely important and the system must be able to pick an appropriate value for β in this case.

Other fault tolereance and reliable communicaiton issues should be investigated further as whenever a smart meter fails to report its energy usage data to the utility, the total of the masked energy data will not sum up to zero, and hence introducing discrepancies in the energy consumption data. This has to be dealt with carefully in order to prevent the attackers from attempting to cripple the smart grid infrastructure. Mechanisms to recover from failures and data logging could be introduced to mitigate these issues.

References

[1] Singapore Energy Market Authority. http://http://www.ema.gov.sg/.

[2] Z. Baig and A.-R. Amoudi. An analysis of smart grid attacks and countermeasures. *Journal of Communications*, 8(8), August 2013.

[3] N. Beety. Analysis: Smart meter and smart grid problems. https://smartmeterharm.files.wordpress.com/2012/12/1-smart-meter-problems-dec-2012-final.pdf, December 2012.

[4] M. Berenguer, M. Giordani, F. Giraud-By, and N. Noury. Automatic detection of activities of daily living from detecting and classifying electrical events on the residential power line. In *e-health Networking, Applications and Services, 2008. HealthCom 2008. 10th International Conference on*, 2008.

[5] B. Defend and K. Kursawe. Implementation of privacy-friendly aggregation for the smart grid. In *Proceedings of the First ACM Workshop on Smart Energy Grid Security*, 2013.

[6] 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*, 2010.

[7] D. Egarter, C. Prokop, and W. Elmenreich. Load hiding of household's power demand. *CoRR*, 2014.

[8] F. Garcia and B. Jacobs. Privacy-friendly energy-metering via homomorphic encryption. In *Security and Trust Management*. Springer Berlin Heidelberg, 2011.

[9] C. Gentry. A fully homomorphic encryption scheme. Technical report, Stanford University, September 2009.

[10] C. Gentry and S. Halevi. Implementing gentrys fully-homomorphic encryption scheme. In *Advances in Cryptology - EUROCRYPT 2011*. Springer Berlin Heidelberg, 2011.

[11] J. Gomez-Vilardebo and D. Gunduz. Smart Meter Privacy for Multiple Users in the Presence of an Alternative Energy Source. *IEEE Transactions on Information Forensics and Security*, Jan 2015.

[12] M. Jawurek, F. Kerschbaum, and G. Danezis. Privacy technologies for smart grids - a survey of options. Technical report, 2012.

[13] G. Kalogridis, C. Efthymiou, S. Denic, T. Lewis, and R. Cepeda. Privacy for smart meters: Towards undetectable appliance load signatures. In *Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference on*, 2010.

[14] A. Kirmse. Privacy in smart homes. http://kirmandi.rumeln.net/data/paper-Privacy.in.Smart.Homes.pdf.

[15] Kun Liu. Paillier Homomorphic Cryptosystem (Java Implementation). www.csee.umbc.edu/ kunliu1/research/Paillier.html.

[16] K. Kursawe, G. Danezis, and M. Kohlweiss. Privacy-friendly aggregation for the smart-grid. In *Proceedings of the 11th International Conference on Privacy Enhancing Technologies*, 2011.

[17] NIST. Guidelines for smart grid cyber security: Vol. 2, privacy and the smart grid. http://csrc.nist.gov/publications/PubsNISTIRs.html#NIST-IR-7628, August 2010.

[18] P. Paillier. Public-key cryptosystems based on composite degree residuosity classes. In *Proceedings of the Annual Conference on the Theory and Applications of Cryptographic Techniques, EUROCRYPT*, LCNS, 1999.

[19] S. Patel, T. Robertson, J. Kientz, M. Reynolds, and G. Abowd. At the flick of a switch: Detecting and classifying unique electrical events on the residential power line (nominated for the best paper award). In *UbiComp 2007: Ubiquitous Computing*. Springer Berlin Heidelberg, 2007.

[20] L. Sankar, S. Rajagopalan, S. Mohajer, and H. Poor. Smart Meter privacy: A Theoretical Framework. *IEEE Transactions on Smart Grid*, Jun 2013.

[21] D. Stehl and R. Steinfeld. Faster fully homomorphic encryption. In *Advances in Cryptology - ASIACRYPT 2010*. Springer Berlin Heidelberg, 2010.

[22] C. Valli, A. Woodward, C. Carpene, P. Hannaya, M. Brand, R. Karvinen, and C. Holme. Eavesdropping on the smart grid. In *Proceedings of the 10th Australian Digital Forensics Conference, Novotel Langley Hotel, Perth, Western Australia, 3rd-5th*, 2012.