Privacy-Friendly Planning of Energy Distribution in Smart Grids

Tassos Dimitriou
Computer Technology Institute
Athens, Greece
Kuwait University, Kuwait
tassos.dimitriou@ieee.org

Ghassan O. Karame
NEC Laboratories Europe
Heidelberg 69115
Germany
ghassan.karame@neclab.eu

ABSTRACT

The smart-grid is gaining increasing attention nowadays, owing to its premise to offer increased reliability, performance, and a balanced utilization of energy. However, the current design of smart-grids raises serious concerns with respect to the privacy and anonymity of users. Thus far, the literature has solely focused on the problem of privately aggregating energy reports and has not addressed the privacy threats that can occur through other intelligent operations which take place in the smart grid, such as planning the energy distribution.

In this paper, we propose a novel solution that enables the planning of energy distribution in the grid without leaking any information about the energy requests of individual smart meters. We also implement a prototype based on our proposal and we evaluate its performance in realistic deployment settings.

1. INTRODUCTION

The *smart grid* leverages infrastructural support to achieve fine-grained power consumption monitoring, integration of new sources of renewable energy in an attempt to offer higher efficiency, reliability, and security [1]. At the heart of the smart grid are *smart meters*, i.e., devices that monitor/record electricity consumption and support a two-way flow of electricity between households and the utility provider.

The widespread deployment of smart meters introduces serious privacy risks since the frequent collection of power data may reveal considerable information about residential appliance usage. For example, previous studies [2] have shown that energy signatures of home appliances can be used to remotely eavesdrop at activities within homes, thus exposing a wealth of private information to anyone with access to such usage data. Currently, a handful of solutions exist to protect smart grid privacy; most of these solutions rely on the use of anonymization/escrow [3] or privacy-preserving aggregation techniques [7, 8]. Other solutions require users to prove in zero-knowledge the correctness of computations based on readings on their own devices [5] or rely on statistical tools to minimize the risks of information leakage while retaining the benefits of the transmitted information (see [6] and the references therein). However, most of these contributions do not address privacy implications that can arise from additional operations that are envisioned by the smart grid, such as energy planning.

Indeed, the premise behind smart grids goes beyond the simple collection of measurements as smart grids were designed to support the integration of the (surplus) energy originating from home owners within the smart grid; home owners can produce energy (e.g., from solar or wind power) and sell their surplus back to the utility provider at a price of their choice. This is especially important for small rural areas that are remote from the utility provider. By planning energy distribution, the utility provider can efficiently reintegrate the surplus of energy generated by households back to the grid. Here, it is important to ensure that such functionality does not come at odds with the privacy of users, and does not leak information about their energy requests, pricing, etc.

In this paper, we address this problem and present a novel solution that enables households to participate in the distribution planning of energy in the smart grid without leaking sensitive information about their energy consumption and/or pricing. More specifically, our solution enables a central entity to solve, in a privacy-preserving manner, a linear program in order to derive a cost effective solution to distribute energy between consumers and producers in the grid—without learning any information about the energy consumption or pricing particular to each household. Finally, we implement a prototype based on our proposal and we evaluate its performance; our findings show that our proposal is viable, scales with the number of smart meters, and can be easily integrated within smart grids.

The remainder of this paper is organized as follows. In Section 2, we detail our model. In Section 3, we propose and analyze a novel solution that enables a privacy-preserving distribution planning of energy in the smart grid. In Section 4, we implement and evaluate our proposal. In Section 5, we overview related work in the area, and we conclude the paper in Section 6.

2. MODEL

We assume a smart grid network which connects N Smart Meters (SMs) that are installed at the premises of N different households and that are connected to a Utility Provider (UP). We also assume that SMs feature secure storage and

autonomous cryptographic functionality. This can be achieved, for example, by using tamper-resistant meters or TPM chips (see [5] for a similar assumption).

We envision a setting in which the UP does not only regularly receive energy requests from the associated metering devices, but can also *purchase* energy from a given SM. For instance, we assume that each household can be a producer or consumer of energy; energy producers could, e.g., correspond to households equipped with renewable energy sources such as solar and/or wind energy. We assume that users are interested in acquiring their deficit of energy from the grid, and in selling their energy surplus back to the grid while maximizing their profits.

Conforming with existing energy distribution models, we assume the existence of *repeaters* in the grid who are responsible for connecting a local neighborhood (featuring a number of households and SMs) to the UP. Since households can be either consumers or producers of energy, the repeaters take the responsibility of locally distributing energy among households of the same neighborhood. In case of a deficiency of energy in their neighborhoods, the repeaters will seek to acquire energy from the UP or from other repeaters in the grid. Similarly, in case of a surplus of energy, the repeaters will dissipate the excess energy back to the rest of the grid. This ensures minimum energy distribution costs (since each neighborhood seeks to gather energy from local sources). Here, the UP orchestrates the energy distribution among/to repeaters in the grid; we assume that the UP and repeaters are mainly involved in the distribution of energy in the neighborhood, and are not necessarily concerned with energy pricing.

Note that renewable energy generation could be highly dynamic, and non-dispatchable. To avoid the need for constant re-planning of energy distribution in the network, we assume that users can rely on rechargeable batteries to temporarily store their generated energy[1] (see [9] for a similar assumption). This enables the UP and repeaters to plan energy distribution in the network less frequently and within pre-agreed timeslots.

Clearly, energy distribution planning comes at the expense of information leakage of sensitive user data [2]; this includes the required energy consumption/production of users, and their purchasing/selling price. Users (or their corresponding SMs)—and rightly so—should control the release of their sensitive data throughout their participation in the distribution planning. In this respect, we assume that the UP and the repeaters are *honest but curious*, and as such correctly execute the protocol but are curious to acquire information about the energy consumption/preferences of the smart grid users.

We also consider in our analysis protection against an external eavesdropper (who may collude with the UP/repeaters). We, however, assume that the adversary is computationally bounded, and cannot break cryptographic primitives, forge signatures, etc. Note that we do not consider information leakage that arises from a mobile adversary who can eavesdrop on the energy transmission links in order to measure the energy consumption of households; as far as we are aware, there are no efficient mechanisms to deter this misbehavior. In our analysis, we do not assume any particular network topology. The SMs, repeaters, and the UP may be connected using bi-directional or uni-directional links. Furthermore, our analysis is not restricted to a network featuring a single UP and extends to any other topology which might include other big energy providers—besides the UP. Figure 1 summarizes our system model; here, we designate the repeaters with letters a to e.

[1]We stress that households which only consume (and do not generate) energy do not need to rely on batteries.

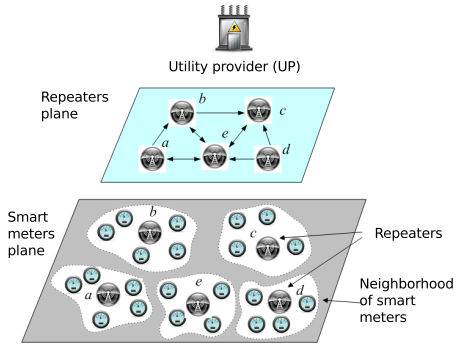


Figure 1: Sketch of a smart grid network. We assume a model where the SMs are connected to neighborhood repeaters which can sell/acquire energy to/from the UP or other repeaters.

3. PRIVACY-PRESERVING DISTRIBUTION PLANNING IN SMART GRIDS

In this section, we outline a novel solution that enables the privacy-preserving energy distribution planning.

3.1 Distribution Planning within Local Neighborhoods

We first start by outlining a scheme that allows the repeaters to plan energy distribution within their local neighborhood. In Section 3.2, we show how this scheme can be integrated within the entire smart grid. Here, we assume that each household can set a price to sell the energy that it produces.

Given the aforementioned model, we can safely model energy distribution among the repeaters and the smart meters as a network flow problem. By casting the various aggregated energy demands and constraints of the households/SMs (as reported by each repeater) into a linear program, efficient energy distribution in the neighborhood can be derived by the repeaters. As these repeaters might be curious, it is however crucial that the prices and energy demands of the various smart meters are not disclosed during this process to the repeaters.

3.1.1 Optimization Problem

For simplicity and without loss of generality, we assume that the capacity of the links is large enough to accommodate the energy requests of smart meters/repeaters. Moreover, we assume that each smart meter N_j is associated with a triplet (e_j, p_j, P_j). e_j denotes the energy required/produced by the household (and as such can be positive/negative). Similarly, p_j denotes the price that the node is willing to receive in exchange for the produced energy, and P_j is the total budget allocated by the node to purchase additional energy if needed (cf. Figure 2).

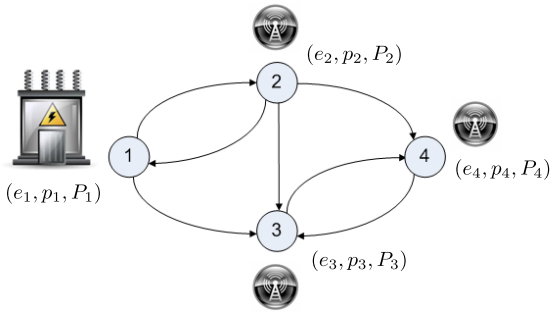


Figure 2: Example of a neighborhood network. Smart meters sell/purchase energy to/from the grid.

Let $\mathcal{N}_i$ be the set of all neighbors of a node N_i, $\mathcal{L}_i$ denote the set of links connecting to node N_i, and $\mathbf{x} \leftarrow \{x_{ij}, \dots\}$ be the vector of energy transmitted over links in the network (here, x_{ij} corresponds to the amount of energy transmitted over link l_{ij}). We denote by c_{ij} the *costs* of transferring energy over link l_{ij}. For example, c_{ij} captures the costs in relation with the wear level of the link, or the distance between households, etc. Clearly, not all links have to be active since some nodes may choose not to send/receive energy to/from specific nodes. We use a positive sign for every outgoing link and a negative sign for every incoming link. Thus, for Node 1 in Figure 2, the constraint should read $x_{12} + x_{13} - x_{21} = e_1$. The set of constraints corresponding to the example of Figure 2 are shown in Table 1.

We then formulate the following optimization problem:

Minimize $\sum_{i,j} c_{ij} x_{ij}$, subject to:

$$\sum_{k \in \mathcal{L}_j} x_{kj} = e_j \quad (1)$$

$$\sum_{k \in \mathcal{N}_j} p_k \cdot x_{kj} \leq P_j \quad (2)$$

By solving the above linear program, the repeater minimizes the costs of transporting energy across its neighborhood. We point out that our analysis described hereafter is not restricted to a specific objective function, and applies to any objective function.

Clearly, the repeaters need to know the values (e_i, p_i, P_i) owned by each node N_i in order to solve the aforementioned optimization problem. However, triplets (e_i, p_i, P_i) reveal considerable information about the households (i.e., the required energy, pricing). To remedy this problem, we present in what follows a novel solution that enables the households to anonymize their triplets (e_i, p_i, P_i) before sharing them with the repeaters, while ensuring that any solution to the LP constructed using the anonymized triplets can be used to derive the correct solution of the LP corresponding to the original triplets. Note that our scheme ensures that the solution of the LP is not revealed to the repeaters. Finally, our scheme does not require changes to the LP solver and enables the repeater to use existing off-the-shelf LP solvers in order to derive the optimal distribution solution in each neighborhood.

3.1.2 Energy Aggregation in Neighborhoods

Table 1: Partitioning of columns according to private data of nodes.

A										b
Node 1		Node 2			Node 3		Node 4			
x_{12}	$+x_{13}$	$-x_{21}$								$= e_1$
$-x_{12}$		$+x_{21}$	$+x_{23}$	$+x_{24}$						$= e_2$
	$-x_{13}$		$-x_{23}$		$+x_{34}$		$-x_{43}$			$= e_3$
				$-x_{24}$	$-x_{34}$		$+x_{43}$			$= e_4$
		$p_2 x_{21}$								$\leq P_1$
$p_1 x_{12}$										$\leq P_2$
	$p_1 x_{13}$		$+p_2 x_{23}$				$+p_4 x_{43}$			$\leq P_3$
				$+p_2 x_{24}$	$+p_3 x_{34}$					$\leq P_4$

Before the repeater is able to derive the optimal energy distribution plan within its neighborhood, it needs to know how much energy is required to be fetched/released from/to the neighborhood (e.g., from the UP or other neighborhoods). By fetching/selling the total deficit/excess of energy, E_r, within its local neighborhood, the repeater can be sure that there is *enough* energy within its neighborhood to be distributed among the participant SMs; this increases the likelihood that the LP will find a distribution solution within its local neighborhood.

The main advantages of this approach are as follows:

1. The repeater interacts with parties located outside the neighborhood only in case of energy excess/deficit inside its local neighborhood. In this way, energy produced within one neighborhood is likely to be dissipated within that neighborhood—thus minimizing the total energy transportation costs within the grid.
2. Since the LP is solved locally within each neighborhood, energy distribution planning in the entire grid becomes a fast and scalable process. Note that a neighborhood typically comprises a modest number of SMs (e.g., < 100).
3. To distribute the energy among repeaters and the UP (repeater/UP plane in Figure 1), the UP can also compute an LP among repeaters. This hierarchical approach ensures that energy is also optimally distributed among repeaters.

We start by showing how the repeater can compute, in a privacy-preserving manner, the total energy demands in its neighborhood. In Sections 3.1.3 and 3.1.4, we then show how the nodes can appropriately anonymize the values that they own, in such a way to enable the repeater to correctly solve the LP in their neighborhood, without learning any information about (e_i, p_i, P_i).

To compute $E_r = -\sum_i e_i$, the total energy aggregate within a neighborhood, the repeater needs to aggregate the values e_i for all neighborhood SMs. However, since the repeater might be curious, we need to ensure that it does not acquire information about the energy demands of SMs and their prices, even when computing the total energy produced/consumed within its neighborhood.

We adapt the protocols in [4, 8] to aggregate the energy demands in a privacy-preserving manner. This ensures that only when the repeater aggregates *all of the* SM energy demands, then it will be able to acquire the correct sum E_r; at all times, the repeater will not be able to acquire any meaningful information about the individual energy values e_i. Following [8], let x_i be a pre-shared secret for meter i such that $\sum_i x_i = 0$. Each SM i sends to the repeater the quantity $g_i = e_i + x_i$. Since x_i is random and secret, g_i

can be seen as a one-time encryption of e_i[2]. The repeater collects all g_i and computes $g_r = \sum_i g_i$. By construction, this is equal to $\sum_i g_i = \sum_i e_i$. Given this, the repeater can recover $\sum_i e_i$, the aggregate of all energy values.

Once the repeater computes the necessary E_r and p_r for its neighborhood, the repeater can formulate an LP for energy distribution within its neighborhood. Here, besides constructing and solving the LP, we assume that the repeater participates as a node in the network with the tuple (E_r, p_r, P_r) in order to fetch/sell the deficit/surplus of energy outside of the neighborhood. Note that the repeater chooses a price p_r for selling energy to SMs, and a total budget P_r (e.g., p_r and P_r could be a unified budget for all repeaters, or could be a function of the number of SMs in the neighborhood).

3.1.3 Hiding the Coefficient Matrix

For clarity of presentation, we will consider an optimization problem of the form $\min cX$, where $X = \{x|Ax \leq b\}$. Here, A refers to the coefficient matrix captured by Equations (1) and (2) (i.e., A is an $m \times n$ matrix which contains p_j), and b refers to the rightmost column of the linear program (see also Table 1).

As shown in Table 1, A is divided in p vertical blocks of $n_1, n_2, \ldots, n_p$ columns such that $n_1 + n_2 + \cdots + n_p = n$. This partitioning is explained by the fact that node N_i "owns" all the variables corresponding to its outgoing links; the coefficients of these variables correspond to a vertical partitioning of the coefficient matrix, thus each group of columns and its corresponding cost vector are owned by a distinct entity. To hide p_j in the coefficient matrix A, we adapt and extend the scheme of [14] as follows[3].

1. Each node N_j randomly generates a new matrix $B_{.j} \in R^{k \times n_j}$, $j = 1, \ldots p$, where n_j is the number of columns held by entity j and $k \geq n$. This results in a joint matrix $B = [B_{.1} B_{.2} \cdots B_{.n}] \in R^{k \times n}$.
2. Each node N_j makes public *only* its matrix product $A_{.j}B_{.j}^T$ as well as its cost coefficient product $B_{.j}c_j$. These products do not reveal either $A_{.j}$ or c_j but enable the public computation of the full constraint matrix needed for the secure version of the linear program:
$$AB^T = A_{.1}B_{.1}^T + A_{.2}B_{.2}^T + \cdots + A_{.p}B_{.p}^T,$$
as well as the full coefficient vector:
$$c^T B^T = c_1 B_{.1}^T + c_2 B_{.2}^T + \cdots + c_p B_{.p}^T.$$
3. By setting $x = B^T u$, the original LP is transformed to the following privacy-preserving variant:
$$\min_{u \in U} c^T B^T u, \text{ where } U = \{u|\ AB^T u \leq b\} \qquad (3)$$
that can be solved by any entity. The optimal value of this LP equals the optimal objective function of the original linear program.
4. Finally, each node N_j computes its optimal x_j component group by setting $x_j = B_{.j}^T u$, for $j = 1, \ldots, p$.

The security of this transformation follows from the observation that for any entity (that is not j, e.g., SM, repeater, or UP), it is impossible to compute either c_j from the revealed product $c_j^T B_{.j}^T$, or $A_{.j}$ from the revealed product $A_{.j}B_{.j}^T$. This is the case since $B_{.j}^T$ is a random group element that is not revealed to the adversary (see also [14]). Here, we also point out that an honest but curious repeater cannot acquire the solution x_j from u_j.

3.1.4 Hiding the (e_j, P_j) Vector

Clearly, the aforementioned solution can only hide p_j; it cannot hide any variable which appears in the rightmost column (i.e., vector b) depicted in Table 1. As such, this solution cannot be used alone to solve the smart grid flow problem since the nodes are not willing to share e_i's and P_i's to the repeater. In what follows, we propose a novel technique to anonymize e_i and P_i (i.e., hiding vector b) before sending them to the repeater while enabling the latter to correctly solve the LP for distribution planning.

Step 1. Let $A_i \in R^{m \times n_i}$ be the matrix of coefficients owned by node N_i, x_i the set of corresponding variables and $b_i \in R^{m \times 1}$ the sensitive part that needs to be hidden. For every b_{ij}, $j = 1 \ldots m$, we create a new variable y_{ij} and a new coefficient α_{ij}. Then, we expand matrix A to a new matrix $A' \in R^{m \times (n_i + m)}$, where $A' = A|D$, and D is a diagonal matrix whose diagonal entries correspond to the coefficients α_{ij} of the new variables. By doing so, *every row* of the sub-LP, $A_i x_i \leq b_i$, is expanded by the term $\alpha_{ij} y_{ij}$, where α_{ij} is a random coefficient only known to node N_i.

If we ensure that $\alpha_{ij} y_{ij}$ is equal to a random value r_{ij}, then the entry b_{ij} can also be updated to $b'_{ij} = b_{ij} + r_{ij}$ and safely made public since the value r_{ij} is only known to node i. To this end, we need to ensure that every new variable y_{ij}, upon solution, obtains the value α_{ij}/r_{ij}, as shown in the following Step 2.

Step 2. We introduce a new set of *constraints* for variables $\alpha_{ij} y_{ij}$. In particular, we create a random coefficient matrix $Q_i \in R^{m \times m}$ and we set $Q_i Y_i = \beta_i$, where Y_i is the new set of variables corresponding to $\alpha_{ij} y_{ij}$. Here, each β_{ij} is given by the product of each row of Q_i with the set of variables *instantiated* with the values $\frac{r_{ij}}{\alpha_{ij}}$. This new set of constraints provides a unique solution to the variables we introduced in Step 1, thus ensuring the correctness of our proposal.

Steps (1) and (2), combined, enable each node to keep private its (enhanced) coefficient matrix and publicize b'_i. *Once this is done, the resulting LP can be solved using the solution in Section 3.1.3 to hide all sensitive variables in the columns "owned" by nodes.*

3.2 Distribution Planning in the Grid

In the previous paragraphs, we outlined a solution to optimally distribute energy within local neighborhoods of the smart grid. As shown in Figure 1, the repeater interacts with other repeaters in the network as well as the UP in order to sell/buy energy required for its neighborhood.

To distribute the energy among repeaters, the UP–in turn–can also compute an LP among the repeaters. This hierarchical approach ensures that energy is also (near-)optimally distributed within the entire grid, thus ensuring that our solution scales with the number of SMs populating the grid. Note that this LP does not have to hide the values reported by the repeaters from the UP. This is the case since these values already correspond to the aggregation of all energy

[2]Note that x_i is used once per each smart meter; we refer the readers to [8] for more details.

[3]The notation $A_{.j}$ denotes the j-th column component of A and A^T denotes the transpose of matrix A.

demands within each neighborhood. Nevertheless, depending on the application scenario, the UP can construct and solve a privacy-preserving LP to distribute energy among the repeaters similar to that described in the previous section.

3.3 Security Analysis

In what follows, we show that an adversary who has access to all the "anonymized" inputs of the nodes cannot acquire any meaningful information about the triplets (e_i, p_i, P_i). Note that the security of aggregating the energy values e_i to compute E_r within a local neighborhood follows from the correctness of the protocols in [8]. As this is a preprocessing step only, we now focus on the security of our proposed transformations.

In order to learn P_i and e_i (the values appearing in the rightmost column of the LP), the adversary needs to acquire the random variable r_{ij}. This is only possible if the adversary *guesses* the rows corresponding to the additional constraints $Q_i Y_i = \beta_i$. Note that the probability of identifying the y_{ij} variables and the associated constraints is proportional to $\frac{m! n_i!}{(m+n_i)!}$. By doing so, the adversary could first solve for the y_{ij}'s $(= \alpha_{ij}/r_{ij})$, and then obtain the r_{ij}'s that could be used to acquire P_i and e_i.

However, our scheme ensures that the adversary cannot learn whether her guesses are correct. Recall, here, that it is computationally infeasible to acquire y_{ij}, from the inputs sent by the nodes; this is the case since those inputs are multiplied with random group elements, which are kept secret from the adversary (cf. Section 3.1.3). Furthermore, y_{ij} and r_{ij} are chosen at random by the nodes and as such act as fresh and unpredictable nonces in the protocol. This ensures that no meaningful information about (e_i, p_i, P_i) is leaked in the process. Additionally, the repeater cannot learn the actual solution of the original LP problem, since it does not know the random matrix B.

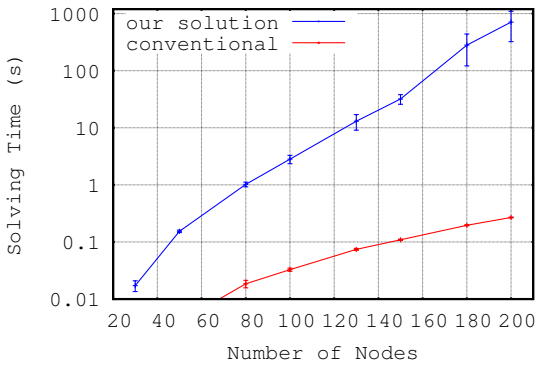
4. IMPLEMENTATION & EVALUATION

We now describe an implementation of our privacy-preserving energy distribution planning solution within a neighborhood.

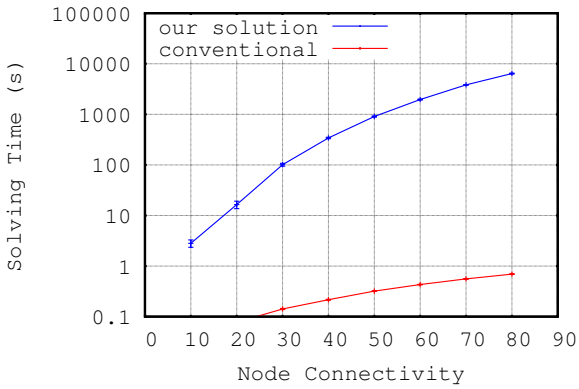
4.1 Implementation Setup

Our system is implemented in C++ and interfaces with the COIN Linear Programming (CLP) [11] solver. CLP is a well-known open-source optimization solver based on the simplex algorithm. In our implementation, we assume that the repeater executes the (unmodified) CLP algorithm to compute the optimal energy distribution in the neighborhood. As described previously, the CLP algorithm takes as input the anonymized inputs of the various SMs. Note that anonymization of inputs incurs negligible overhead on the smart meters since this process only requires the smart meter to perform addition and/or multiplication of small matrices. In our evaluation, we therefore focus on assessing the overhead incurred by our solution on the repeaters—which are computing the optimal solution in their neighborhoods.

To evaluate our solution within a realistic neighborhood, we created a C++ based network generator for smart grids. Our network generator takes as input the number of nodes N in the neighborhood, and the average connection per household C, and randomly generates the appropriate smart grid network featuring N smart meters—each connected to $C < N$ other households on average. Our generator also emulates random input constraints by each smart meter; as



(a) Time to solve w.r.t number of smart meters.



(b) Time to solve w.r.t network connectivity.

Figure 3: Performance evaluation of our privacy-preserving energy distribution solution.

described previously, these consist of the amount of energy that each smart meter is willing to purchase/sell, the price of energy for which the smart meter is willing to sell, and the total price each smart meter is willing to pay in exchange for energy. Although these inputs are randomly generated for the entire network, our generator ensures that the corresponding LP problem features a unique solution. This is done by ensuring that the rightmost column in the LP problem is randomly chosen from the set of *feasible* solutions.

We evaluated the LP solving performance using a repeater running a 64-bit Windows7 OS on a Quad-Core i5-3470 with 3.2 GHz and 8 GB RAM. Each data point is averaged over 20 independent runs of the simulator; for each data point, we also compute the corresponding 95% confidence intervals.

4.2 Evaluation Results

Our results are depicted in Figure 3. In Figure 3(a), we show the time required by the repeater to solve the linear program as a function of the smart grid size. Here, we assume a network topology whereby each smart meter acquires energy from at most 10% of the households (i.e., $C = 0.1N$). As expected, our findings show that our solution incurs additional overhead in time in order to solve the LP, when compared to the conventional non-privacy-preserving solution. This is mainly due to the fact that our solution introduces a new set of variables and constraints per smart meter, which

translates into a bigger linear program to solve. Notably, our solution converts a $(2N + NC) \times NC$ LP matrix into an $(4N + 2NC) \times (2CN^2 + 2N^2)$ anonymized matrix.

In Figure 3(b), we evaluate the time to solve the linear program with respect to the network connectivity. Here, we assume a network comprising of 100 nodes, and we vary C. Our results indicate that C considerably impacts the solving time of our privacy-preserving LP. The more connected is the network, the larger are the numbers of required variables and constraints; this translates into longer LP solving times.

However, our results show that the overhead incurred by our privacy-preserving solution can be largely tolerated. For instance, our solution requires around 5 seconds of computations by a repeater equipped with a single-threaded non-optimized open-source LP solver, in case the neighborhood features 100 moderately connected SMs. Note that this corresponds to a rather large neighborhood. Clearly, we expect a drastic reduction in LP solving times in an optimized multi-threaded LP solver implementation. Nevertheless, even when using an off-the-shelf solver such as CLP, we point out that this overhead can be largely tolerated—especially since the timescale for planning energy distribution in the smart grid (e.g., on an hourly basis) exceeds by far the cost of computing the optimal energy distribution solution by the repeaters.

5. RELATED WORK

As far as we are aware, this is the first contribution that addresses information leakage due to energy distribution planning in smart grids.

The concept of privacy-preserving aggregation in smart grids was first proposed in [7,8]. In [3], Efthymiou *et al.* propose an escrow scheme that relies on a trusted party to protect SMs' reports. In [4, 8], different methods are discussed for aggregating energy measurements through the use of randomness to blind the power traces of users. In [5], Rial *et al.* propose a method to calculate energy fees while protecting meter data using zero-knowledge proofs and commitment schemes. In [16], Lu *et al.* propose a technique based on the Paillier cryptosystem to secure data aggregation in smart In [10], Dimitriou and Karame present a solution which enables anonymous payments in smart grids.

In [12], Vaidya develop a framework to transform any LP into one that leaks no information about the input data in a two-party setting where one party owns the objective function while the other owns the constraints of the LP. In the multi-party case, Mangasarian propose similar privacy-preserving transformations for either horizontally partitioned linear programs [13] or vertically partitioned linear programs [14]. This latter contribution was extended in [15] to capture non-negativity constraints for the variables x. In this paper, our privacy-preserving energy planning solution extends the scheme in [14] to hide all the inputs of the LP in the case where the constraints are owned by several nodes in the network. In [17], He *et al.* propose to enhance the privacy of maximum flow computations in distributed graphs by obfuscating the underlying graph and the node identities.

6. CONCLUSION

Within existing smart grids, smart meters undergo a set of essential operations, which range from the collection of measurements to the distribution of energy in the smart grid. In this paper, we presented a novel solution that protects the privacy of home owners when trading energy with the grid. More specifically, our solution enables households to participate in the distribution planning of energy in the grid without leaking any information about their energy consumption and pricing. We implemented and evaluated a preliminary prototype based on our proposal in a smart grid setting; our findings show that our solution scales well with the number of smart meters in the grid.

Acknowledgements

The authors thank Christopher Zech for the help in the implementation and evaluation of the solution as well as the reviewers for their useful comments. The first author would like to acknowledge support by Kuwait University, Research Grant No. QE 02/13.

7. REFERENCES

[1] P. McDaniel and S. McLaughlin, "Security and Privacy Challenges in the Smart Grid", In IEEE Security and Privacy, Vol. 7, No. 3., 2009.

[2] H. Y. Lam, G. S. K. Fung, and W. K. Lee, "A Novel Method to Construct Taxonomy Electrical Appliances Based on Load Signature", In IEEE Transactions on Consumer Electronics, vol. 53, 2007.

[3] C. Efthymiou and G. Kalogridis, "Smart Grid Privacy via Anonymization of Smart Metering Data", In IEEE SmartGridComm, 2010.

[4] T. Dimitriou, "Secure and Scalable Aggregation in the Smart Grid", In the 6th IFIP/IEEE International Conference on New Technologies, Mobility and Security (NTMS), 2014.

[5] A. Rial, G. Danezis, "Privacy-Preserving Smart Metering", In WPES, 2011.

[6] S. Raj Rajagopalan, L. Sankar, S. Mohajer, H. Vincent Poor, "Smart Meter Privacy: A Utility-Privacy Framework", In IEEE SmartGridComm, 2011.

[7] F.D. Garcia, B. Jacobs, "Privacy-Friendly Energy-Metering via Homomorphic Encryption", In Proceedings of the 6th Workshop on Security and Trust Management (STM), 2010.

[8] K. Kursawe, G. Danezis, and M Kohlweiss, "Privacy-Friendly Aggregation for the Smart-Grid", In Proceedings of PETS, 2011.

[9] G. Kalogridis, C. Efthymiou, S.Z. Denic, T.A. Lewis, and R. Cepeda, "Privacy for Smart Meters: Towards Undetectable Appliance Load Signatures", In IEEE SmartGridComm 2010.

[10] T. Dimitriou, G. Karame, "Privacy-Friendly Tasking and Trading of Energy in Smart Grids", In Proceedings of ACM SAC, 2013.

[11] COIN-OR Linear Programming Solver, Available from `https://projects.coin-or.org/Clp`

[12] J. Vaidya, "Privacy-preserving linear programming", in ACM SAC, 2009.

[13] O.L. Mangasarian, "Privacy-preserving horizontally partitioned linear programs", In Optim. Letters, 2012.

[14] O. L. Mangasarian, "Privacy-Preserving Linear Programming", In Optimization Letters, 1–7, 2010.

[15] H. Li, Z. Tan and W. Li , “Privacy-preserving vertically partitioned linear program with nonnegativity constraints”, In Optim. Letters, 2012.

[16] R. Lu, X. Liang, X. Li, X. Lin, X. Shen, “EPPA: An Efficient and Privacy-Preserving Aggregation Scheme for Secure Smart Grid Communication”, In IEEE TPDS, 2011.

[17] X. He, J. Vaidya, B. Shafiq, and N. Adam, “Privacy Preserving Maximum-Flow Computation in Distributed Graphs”, In Proceedings of SOCIALCOM-PASSAT, 2012.