Privacy Preservation for Smart Grid Multicast via Hybrid Group Key Scheme

Depeng Li and Zeyar Aung
Masdar Institute Science and Technology
Abu Dhabi, UAE,
{dli, zaung}@masdar.ac.ae

Srinivas Sampalli
Dalhousie University
Halifax, Canada
srini@cs.dal.ca

John Williams and Abel Sanchez
Massachusetts Institute of Technology (MIT)
Cambridge, Massachusetts, USA
{jrw, doval}@mit.edu

Abstract—Privacy preservation is a crucial issue for smart grid security. With more and more group applications, data and appliances proliferated in smart grid, electricity customers are faced with extended privacy risks. Meanwhile, the natures of smart grid—such as limitation in computation power of smart meters, requirement to be highly reliable, and transformation from a private system to an open one—which makes privacy preservation a challenging task. In this paper, we propose a group key scheme to safeguard privacy with the supply of availability, efficiency and fault-tolerance. In particular, hybrid architecture is constructed to achieve both fault-tolerance and efficiency with only one set of group key installed. In addition, key trees are sophisticatedly managed in order to reduce the number of exponentiation operations. Experimental results show that our scheme is able to provide significant gains in performance over other privacy preservation methods while effectively preserving the participant’s privacy.

Keywords—group key, multicast, privacy, reliability, smart grid

I. INTRODUCTION AND MOTIVATION

Smart grids, or the intelligent electricity grid that utilize modern IT/communication/control technologies, become a global trend nowadays. It introduces substantial benefits and opportunities to our society, but also raises challenges concerning with privacy as a side effect. For example, fine-grained smart metering data provides customers and utility companies with the information about ongoing electricity status and consumption patterns. Nevertheless, this also significantly increases possibilities of leaking customers’ privacy including personal information, daily activities, individual behaviors, etc., more frequently [8], [4].

The digitizing move to replace normal meters with smart meters allures more privacy invasions. Some pioneer studies, e.g. [22] (depicted in Fig. 1), explore means to monitor major appliances in a dwelling by examining the power usage data collected every 15 minutes within 24 hours. In [9], power customer’s privacy is far more disclosed: it can recognize which program is currently being displayed on a standard TV set in a dwelling through matching its power consumption signature with that of each TV channel. Therefore, all the players: government policy makers, utility companies, and customers now realize that smart grid’s privacy is a critical issue cannot be overlooked. In 2009, the Netherland senate questioned about the potential risk of privacy exposure and therefore turned down the bill which was aiming at compulsory installations of smart meters [6]. Customers can choose to install the meter with remote reading facility disabled or even to refuse the meter.

To address privacy risks and concerns, most of pioneer researches focused on shielding data recorded at smart meters via extra battery [18] or protecting data unicasted out of smart meters via perturbation [15] or encryption [14], [26]. However, not all smart data has been protected in terms of privacy: 1) Smart messages, e.g., power control commands sent toward smart meters are not covered. 2) Multicast messages are not safeguarded efficiently. They both contain sensitive messages. Privacy preservation is like the proverbial chain which is as strong as its weakest link. Therefore, privacy solutions should extend their scope to ensure no smart data is leaked.

Relying on bi-directional communication and group management technologies, group communication is extensively deployed in smart grid because of its scalability, its efficiency and its functionality across network segments. In the IEC 61850 [10], the multicast technology is utilized in the substation to collect power consumption data and status as well as to deliver power control commands and alarms. The Generic Object Oriented Substation Events (GOOSE), for example, a multicast protocol, is designed for group services.

Unsurprisingly, the deployment of group communications incurs a particular concern for privacy loss. A number of group applications/communication within a customer premise, meter monitoring, demand-respond system, smart devices’ renew services, etc. are more likely to be probed due to their popularity. Multicast messages controlling air-conditioning units’ temperatures, for instance, may be sent in customer premise in high population density, e.g., high-rise towers. The adversary can recognize a specific group’s purpose such as saving energy consumption in demand-respond system. More importantly, as comparing the current control commands with that of the history, adversary can easily infer a specific group member’s privacy: if the group member always receives commands turning off the air conditioning at a fixed time frame, it is highly probable that the customer routinely leaves at that time. This may introduce a privacy breach.
Sensitive data in multicast messages should not be leaked to non-group participants. Meanwhile, most of the times, these data should be shared with other group members even it is private. One reason, for instance, is because they belong to the same customer. Here, we focus on privacy preservation for scenario aforementioned via group key scheme [11], [25], a popular cryptographic primitive. Each group member reaches a symmetric group key to encrypt multicast data such as power control commands, advanced metering data, etc. Therefore, adversaries can neither eavesdrop usage/status contents nor analyze activity models.

Unfortunately, previous group key schemes cannot be deployed directly in smart grid since they cannot satisfy the following smart grids’ requirements: 1) The nature of smart grid ranks reliability, safety and availability as the highest priorities [4], [24]. Therefore, group key schemes should heavily emphasize the built-in availability as well as minimize its down time during smart grid’s operation. 2) Most smart meters are equipped with low-capacity devices and limited memory which tend to be restricted in their computation and storage capability.

To achieve these goals, we propose an efficient and fault-tolerant privacy preservation solution via group key schemes for such scenarios that multicast messages can be shared among group members. The following contributions are made:

1. **Hybrid architecture with reliability and self-healing:**
   We present a hybrid architecture that incorporates the centralized and the contributory group key scheme. Both efficiency and fault-tolerance are ensured.

2. **Efficiency:**
   To be more efficient, a few novel algorithms are designed:
   a) Individual rekeying immediately after a member joins and batch rekeying at the end of an interval as members leave;
   b) Leaving members are treated differently based on their attributes or intentions to return;
   c) The key tree is organized using special constructs like the fixed key tree, the child key tree, etc.

II. BACKGROUND

A. Smart Grid Multicast and Privacy Concerns

There are a number of privacy concerns regarding multicast communications in smart grid. Due to page limit, we discuss only one multicast communication case within a customer’s building as an example:

In a customer building such as a dwelling, Smart Grid Nodes (SGN) can be installed at or embedded within a number of smart appliances, e.g., washing machines, dryers, heating, ventilations, refrigerators, air conditioners and even solar panels as well as wind turbine generators [1]. The data concentrator/substation gateway/smart meters communicate with these smart appliances to collect power status and usage data. Meanwhile, the data concentrator etc. also plays a key role in conducting incentive pricing by decreasing and smoothing out power usage peak. The data concentrator etc. relays and multicasts SGN settings, signals, or commands received from control center/substations. Messages are used to switch SGNs’ profiles e.g. power-saving mode, power devices on/off, etc. or to amend significant parameters e.g. temperatures for air conditioners. Privacy in multicast is critical especially when the customer premise is in a high population density area such as high-rise towers. Multicast messages are prone to be eavesdropped which leads to the result that privacy is likely to be disclosed. Meanwhile, multicast participants share sensitive messages with each other since they belong to the same customers or message sharing is mandatory.

B. Selection of Group Key Scheme

To satisfy smart grids’ privacy-preserving requirement for multicast communication, a common and efficient solution is to deploy a symmetric group key shared by all multicast participants, e.g. smart meters, data concentrators, Intelligent Electronic Devices (IED), etc. With the support of this shared key (group key), multicast communication data can be encrypted and decrypted. Outsiders cannot peek. Therefore, a group key management protocol that computes the symmetric group key and forwards the partial keys to all participants is required.
legitimate multicast members is central to the privacy preservation of the multicast communication in smart grids.

When one or more member leaves or joins the group, the group key should be updated so that only current group members comprehend the group key. This procedure is called rekey. There are two kinds of rekey strategies: individual rekey and periodical batch rekey. The former rekeys the group key for every group membership update such as joining/leaving. The later processes the joining and leaving requests in a batch at the end of each rekey interval. In this paper, we utilize individual rekeying to process join request and periodical rekeying to process leaving requests because: 1) In smart grids, most smart devices e.g. smart meters playing the group member role have stationary membership. The group membership change events e.g. joining/leaving are rare; 2) Periodical rekeying introduces vulnerability window but also leads efficiency. Considering that some group members, e.g. smart meters show low-end processing capacity, the tradeoff between performance and security is affordable. 3) Periodic rekeying introduces group key refresh at the end of time interval even there are no membership changes. This promotes the security level.

Furthermore, in view of architecture, group key management schemes can be broadly classified into two categories, namely, centralized and contributory: In a typical centralized group key management scheme e.g. Logical Key Hierarchy (LKH) [25], a trusted third party, known as the key server, is responsible to generate, to encrypt and to distribute the symmetric group key, partial keys and individual keys to all other group members. It has the advantages of efficiency of the symmetric key encryption/decryption. However, it suffers from the following drawbacks. 1) Since all group secrets are generated and stored in one place, the key server could present itself as an attractive attack target for adversaries. 2) The key server can become the single point of failure/bottleneck.

In contrast, in contributory group key management schemes e.g. Tree-based Group Diffie-Hellman (TGDH) [11], every group member contributes to the group key generation. It has the advantage of fault-tolerance. However, for group membership changes, it lacks scalability in terms of computational cost. For example, TGDH has the following drawbacks. 1) Every group member performs the expensive Diffie-Hellman key exchange with \(1, O(\log_2 n)\) times exponentiation operations for every group membership update where \(n\) is the group size. 2) Every sponsor should sign and forward a large number of rekeying multicast messages to update a group key. It results in expensive communication overload and computational costs. In this paper, we are willing to propose hybrid architecture which combines both centralized and contributory group key schemes to protect the privacy of smart grid multicast service.

C. TGDH

The crux of the group key management scheme in TGDH is to use a binary key tree for group key updates. Let \(T\) be a binary tree in which every node is represented by \(<h, i>\) where \(h\) is its height (level) and \(i\) is its index. Each node in the binary tree, has two keys, node key \((K)\) and blinded key \((BK)\). The node key associated with node \((l, v)\) is \(K_{<l,v>}\) and its blinded key \(BK_{<l,v>} = a^{K_{<l,v>}}\).

Each node in the tree is either a leaf or a parent of two nodes. Each leaf represents a group member \(M_i\) which generates \(r_i\), a random integer. It can be treated as the leaf node’s node key. The node key of an internal node \(<l, v>\), is derived from keys of its children, \(<l + 1, 2v>\) and \(<l + 1, 2v + 1>:\)

\[
K_{<l,v>} = BK_{<l+1,2v>}^{K_{<l+1,2v+1>}} = BK_{<l+1,2v+1>}^{K_{<l+1,2v>}} = a^{K_{<l+1,2v>}} = a^{K_{<l+1,2v+1>}}
\]  

(1)

The node key of the root is the group key. While a group member joins, the shallowest leftmost leaf node in the key tree is selected as the sponsor and acts as the sibling for the new group member. When a group member leaves, the sponsor is the shallowest leftmost leaf node of the sub-tree rooted as the leaving member’s sibling node. The sponsor is responsible for updating its secret random integer \(r_i\), and all keys on its key path. Then, the sponsor multicasts all updated blinded keys, based on which, other members update keys on their key paths and compute the new group key. Refer to [11] for details.

D. ID-based Key agreement based on Diffie-Hellman

We adapt an ID-based key agreement [21] to mitigate the leakage of ID values. It is noticeable that the leakage of ID values associated with partial keys cannot uncover any smart data’s privacy.

1. Set-up

According to the RSA algorithm, TC (trusted key generation center) generates and publishes \((n, g, e)\) but keeps \((p, q, d)\) secret.

2. Key generation

For an authorized user A, TC assigns it randomly generated ID, \(ID_a\) and computes \(s_a = ID_a^{-d} (\mod n)\);

For an authorized user B, TC assigns it randomly generated ID, \(ID_b\), and computes \(s_b = ID_b^{-d} (\mod n)\);

Then TC stores \((n, g, e, ID_a, s_a)\) into a smart card and issues it to user A.

TC stores \((n, g, e, ID_b, s_b)\) into a smart card and issues it to user B.

3. Key Agreement

Step 1: A and B respectively input/insert these secret value to their devices via secure channel e.g. physically touch methods via smart cards.

Step 2: A randomly generates an integer \(r_a\), and, computes

\[
t_a = g^{r_a+ID_e} s_a
\]  

(2)
B randomly generates an integer $r_b$, and computes

$$t_b = g^{r_b}D_a \times s_b$$ (3)

**Step 3:** A and B exchange $(ID_a, t_a)$ and $(ID_b, t_b)$

**Step 4:** A and B computes formula (3) and (4) respectively:

$$k_a = ((g^{-ID_a} \times t_b)^e \times ID_b)^r_a (mod n)$$

$$= g^{\sigma a r_a b} (mod n)$$ (4)

$$k_b = ((g^{-ID_b} \times t_a)^e \times ID_a)^r_b (mod n)$$

$$= g^{\sigma b r_a a} (mod n)$$ (5)

III. OUR PROPOSED SCHEME

A. Hybrid Architecture

In a centralized group key scheme, the data concentrator/substation gateway is equipped with powerful computation capacity. They play the key server’s role to generate the symmetric group key, partial keys, individual keys, key path and key tree which are forwarded to the corresponding group member—smart devices, e.g. smart meters. It takes advantages of symmetric key encryption/decryption’s efficiency but the key server is the single-point failure: when it is out of control, as demonstrated in Fig. 3, the centralized group key scheme ceases therefore reliability cannot be guaranteed. In contrast, contributory group key scheme is fault-tolerant. But it is expensive in both computation and communication. Consequently, low-capacity smart meters and limited wireless communication channels, e.g., Zigbee, cannot afford the cost.

A naïve/straightforward method to fix this problem is to install one set of centralized scheme and one set of contributory scheme simultaneously on every smart device to guarantee both fault-tolerance and efficiency. However, deployments of two sets of schemes not only make the system more complicated but require more resources such as storages.

In this paper, we propose a hybrid protocol which combines the advantages of the centralized approach’s efficiency and the contributory scheme’s fault tolerance. The basic idea behind the hybrid architecture is that when the key server is off-line, then group key management will utilize a contributory scheme. If the key server is on-line, there will be two possibilities. If all the group members are able to access the key server (no partitioning of the group), a centralized scheme e.g. LKH is used in which the key server is responsible for calculating and delivering the intermediate keys associated with the binary key tree since the key server is deemed to have a high processing capability. On the other hand, if the group is partitioned (some of the members are not able to access the key server), then a combination of the two schemes is used—the members with access to the key server use the centralized scheme while the others use the contributory scheme. Both of them follow the TGDH key tree to update the node keys and blinded keys associated with the nodes on the binary key tree.
In both centralized and contributory group key scheme, the key tree structure maintenance components within the key server or the group members modify their key tree structures according to the group member joining or leaving.

Generally, the centralized scheme is the primary mechanism: the key server, e.g. substation gateway generates, maintains and distributes the key tree. Every group member, e.g. smart meter receives and stores its key path. Once the key server fails, a contributory scheme takes in charge automatically—every smart meter cooperates with each other to manage the group key and the key tree as well. When the gateway restores, every meter delivers its latest key path to the gateway via secure channels. The scheme is controlled in centralized way again. Thus, self-healing is ensured. Since both centralized and contributory key management use the same key tree structure, no rekey operations are processed and no rekey messages are forwarded to implement the switch between the centralized scheme and the contributory scheme. Due to space limit, we will not address each component in detail.

B. Efficient Group Rekeying Scheme

In this subsection, we address our key tree structure, how to arrange nodes in a key tree while members join/leave and how to compute the shared group key for key server and each group member as well.

Like [11], [13], [25], our scheme uses a binary key tree in which every node is associated with a node key and a blinded key. Each leaf node represents one and only one group member. The node key of a leaf node is also called individual key as depicted in Fig. 4 (d).

The layout of the key tree is demonstrated in Fig. 4 (a), in which the fixed key tree, $T_{\text{fixed}}$, contains all stationary smart devices e.g. smart meters. The rest group members are located in the subtree $T_{\text{main}}$. Notice that $T_{\text{main}}$ is $T_{\text{fixed}}$’s sibling and at the same time, the child key tree, $T_{\text{child}}$ is part of $T_{\text{main}}$. $T_{\text{child}}$ stores the new incoming smart devices e.g. Plug-in Electrical Vehicles (PEV).

In our scheme, to lessen member’s waiting time, a joining request is processed immediately. All leaving requests are handled at the end of the rekeying time interval for the sake of efficiency. This may introduce trivial vulnerability but it is affordable. The reason is that the nature of smart grids informs us that almost all smart devices are stationary and membership changes rarely happen. Therefore, the rekeying interval (e.g. 15 – 60 minutes) defined in our scheme is sufficiently secure.

### Individual Rekeying for Joining Member:

New joining group members should contain its attributes: stationary or dynamic. The former is associated with nodes representing a fixed smart meter or customer’s own PEV. It will be inserted into the fixed key tree, $T_{\text{fixed}}$. The later is associated with nodes representing trusted nodes e.g. friend’s PEV. It will be inserted into the child key tree, $T_{\text{child}}$. The insertion point for $T_{\text{child}}$ is its corresponding shallowest leftmost nodes. Meanwhile, every current group member calculates the new group key via one way hash function $G’ = Hash(G)$ where $G$ is the current group key. The new member receives the new group key $G’$ from the sponsor via secure channels. Refer to TGDH about how the sponsor node delivers partial keys to the new member.

### Batch Rekeying for Leaving Member:

When a group member is going to leave as it is malfunctioning or legacy, the group key need to be updated at the end of the rekeying interval. The child key tree, $T_{\text{child}}$, may be moved to replace a leaving leaf node’s place if any, for sake of computation efficiency. For details, suppose that the group member $M_i$, is represented by the leaf $< h, i >$ which leaves the group. Four cases follow:

**Case 1:** If $T_{\text{child}}$ is not available, our leave protocol is the same as that of TGDH.

**Case 2:** If $T_{\text{child}}$ is available and $< h, i >$ is within $T_{\text{child}}$, the key tree structure stays the same.

**Case 3 and 4:** If $T_{\text{child}}$ is available and $< h, i >$ is not within $T_{\text{child}}$, there are two possibilities, moving $T_{\text{child}}$ which is shown as Fig. 4(a), or not. The leaf node $< h, i >$’s position and computational cost decide whether $T_{\text{child}}$ should be moved.
If moving $T_{child}$ cannot result in the performance gain, $T_{child}$ should stay. This scenario is called case 3. Otherwise, $T_{child}$ should be moved to take $<h,i>$’s position and $<h,i>$ is cut off. It is called case 4.

For example, Fig. 4(b) is the original key tree. Fig. 4(c) shows that $M_2$ leaves. Since $M_2$ is not within $T_{child}$ and moving $T_{child}$ costs less, $T_{child}$ rooted at $<2,2>$ is moved to replace the position of node $<2,1>$. The former node $<2,1>$ is cut off. As its left child node is removed, node $<1,1>$ is deleted. $<1,1>$’s right node $<2,3>$ is renamed as $<1,1>$ and promoted to its parent’s position. Fig. 4(d) demonstrates that, when $M_5$ leaves, $T_{child}$ need not be moved since $M_5$ is within $T_{child}$.

According to our observation, we find that the fixed smart devices will not leave as long as it is legal, works in a good condition and no plan to be updated. Therefore, even they appear not online sometimes because of device out-of-control or its jammed communication channel, we still do not mark it as leaving. Furthermore, since the customer’s own PEV intends to return, its node still stays in the key tree and no leaving request should be sent out. Otherwise, the nodes e.g. friends’ PEV, expired/broken smart meters, etc. are marked as leaving and will finally be deleted in rekeying process.

IV. IMPLEMENTATION AND EXPERIMENTS

A. Implementation

During our implementation, we utilize ID-based Diffie-Hellman key exchange agreement introduced in section II-D to calculate the node key for intermediate nodes in the key tree. It is against the Man-in-the-Middle attack. Node keys of any two nodes $a$ and $b$ will be $r_a$ and $r_b$. And their corresponding blinded keys are $t_a$ and $t_b$. Finally, their parent node’s node key will be $k = k_a = k_b$. As described in section II-D, ID issued to smart devices can be anonymous, random strings to cover smart device’s privacy.

We implement the adapted ID-based key agreement by C language based on Pairing-Based Cryptography (PBC) library [16] built on the GNU Multiple Precision arithmetic (GMP) library [7]; GMP library provides arbitrary precision arithmetic APIs which are invoked by PBC to support pairing-based cryptosystem. Our implementation has been executed on Virtual Machine hosted by Oracle’s VirtualBox. Here is the detailed configuration of VM – OS: Ubuntu 11.10; Memory: 496MB; Processor: Intel Core i5-M560; CPU 2.67GHz; Disk 7.9 GB. The developed ID-based key agreement is used as a test bed for experimental evaluation. The test is executed for 10 repetitions (randomly selected number), the average of which is utilized to represent the running time to accomplish the two party key agreement. Our test result shows that it is 37.625 ms for each party.

B. Experimental results

The goals of our experiments are to estimate the performance of our group key scheme specifically focusing on the computational cost and communication overhead. These goals enable us to showcase the feasibility of our schemes. Furthermore, it can help us determine the performance gains or additional cost introduced by our scheme at different scenarios.

**Computational Cost:**

In this paper, we use the group membership change data set (duration: 3 months; group size: 250 nodes) collected in MBone [2], a famous and practicing multicast services on Internet. We demonstrate the times used to generate/update a group key for different group size in Fig. 5 (a) based on the number of exponentiation operations required to accomplish our group key scheme. It also includes the time requires by GDOI [19] as a comparison. Therefore, as shown in Fig. 5 (a), our proposed group key scheme is significantly efficient than GDOI concerning with the overall execution times on our test bed. While the key server is out of control, in terms of computational cost, the hybrid architecture requires no computational operations and the straightforward solution needs a number of operations to generate a new contributory group key.
Communication Overhead:

In this sub-section, hybrid architecture and straightforward methods are simulated via Network Simulation-2 (ns-2) [20], a widely used simulation tool. In this test, to achieve the routing function, AODV, a routing protocol, is deployed to connect wireless nodes and forward packets from one node to another. A multicast AODV (MAODV) module is extended at ns-2 to multicast packets for this project. Specially, this simulation utilized the test scenario components listed below:

- NS2 version: ns-2.27
- Network: Mobile Ad Hoc Network (MANET)
- Routing Protocol: AODV
- Multicast Protocol: MAODV
- Area: 1500 x 300 meters
- Number of nodes: 50
- Physical/Mac layer: IEEE 802.11 at 2 Mbps, 250 meter transmission range

Mobility model: random waypoint model with no pause time, maximum speed 20m/s (high mobility scenarios).

This test was developed to simulate a scenario which last 10 minutes. In the middle of the test, the key server is out-of-service. The purpose of this case study is to compare the communication overhead manage a group key for straightforward method and for the proposed hybrid architecture. Specially, in the hybrid architecture, the messages multicast to notify other group members that the key server is out of service if the key server cannot be detected with heartbeats. Then, a notation to let all group members switch from the centralized method to the contributory one is forwarded. In contrast, in the straightforward method, a new group key will be generated via contributory solutions. Fig. 5 (b) shows experimental results which indicate that our hybrid scheme is efficient in terms of communication overhead. Notice that background messages e.g. routing data, keep-alive data are also counted in Fig. 5(b).

V. RELATED WORKS

We now summarize the privacy preservations of smart grids. Main approaches such as cryptographic primitives [5], [12], [14], [23], [26] battery [18], ID anonymization [3], and disturbance [15] are reviewed.

Cryptographic primitives: J. Zhang and C. A. Gunter [26] propose an approach to secure multicast in smart grid via deploying IPSec protocol and Group Internet Key Exchange (GIKE)/Group Domain of Interpretation (GDOI). Both IPSec and GIKE/GDOI [19] are standards for multicast applications: video broadcast and multicast file transfer; nevertheless, it is less efficient as not specifically designed for multicast in smart grid. Furthermore, its goal is confidentiality instead of privacy.

F. Li, B. Luo and P. Liu [14] focus on privacy preservation for smart metering data aggregation in which all messages are encrypted by using homomorphic encryption algorithm. F. D. Garcia and B. Jacobs [5] proposed a privacy-friendly protocol by using homomorphic (Paillier) encryption and additive secret sharing to realize tasks such as billing, leakage detections, etc. A. Rial and G. Danezis [23] use zero knowledge proofs and commitments to preserve smart meters’ privacy. This protocol facilitates the accurate bill payment and derives the correctness of power bills without exposing any message about customers’ power usage. In [12], K. Kursawe, G. Danezis, M. Kohlweiss proposed four different protocols, e.g. Diffie-Hellman (DH) Key-exchange based protocol, DH and Bilinear-map based protocol etc. for metering data aggregation services with no privacy disclosure.

These privacy preservation solutions relying on cryptographic schemes (except GDOI) cannot secure group communication. The reason is because they are designed for privacy of an individual device or unicast communications. Invoking them in multicast settings cannot be scalable.

Battery: S. McLaughlin, P. McDaniel and W. Aiello [18] propose the Non-Intrusive Load Leveling (NILL), a new class of algorithms and systems to mask the appliance’s fine-grained power usage signature. Ten particular deep-cycle batteries are deployed. However, there is still a few privacy leakages after deploying extra batteries. Furthermore, they show expensive installment and maintenance cost.

ID Anonymization: in [3], C. Efthymiou and G. Kallogridis proposed a trusted key escrow service to enhance privacy during services such as smart metering data collection, billing service, etc. The approach anonymizes frequent readings with pseudonymous IDs along with randomized time intervals.

Disturbance H. Li, R. Mao, L. Lai, and R. C. Qiu [15] proposed a compressed meter reading approach that enhances its privacy through the use of random sequence. This solution integrates with pseudo-random spreading codes and channel gains from smart meters to the Access Points (AP). However, AP is assumed never to be compromised.

VI. CONCLUSION

The privacy preservation in smart grids over multicast communication is a challenging task. The well-known group key scheme cannot be directly used to protect the privacy of multicast service. The central issues are the reliability and efficiency considering that the prompt restoration and the minimum overall fault times are highly demanded in smart grids. Previous solutions are not appropriate for smart grid settings because of heavyweight rekeying operations, poor scalability or single-point failure architecture.

This paper presents the design and specification of a fault-tolerant and efficient group key agreement to safeguard the privacy of multicast messages in smart grids. The performance result demonstrates that our scheme is the efficient and acceptable. It satisfies smart grid system’s reliability and efficiency requirement.

REFERENCES


GNU, 2012. The GNU Multiple Precision Arithmetic Library http://gmplib.org/


