A Decentralized Security Framework for Data Aggregation and Access Control in Smart Grids

Sushmita Ruj and Amiya Nayak

***Abstract*—We propose a decentralized security framework for smart grids that supports data aggregation and access control. Data can be aggregated by home area network (HAN), building area network (BAN), and neighboring area network (NAN) in such a way that the privacy of customers is protected. We use homomorphic encryption technique to achieve this. The consumer data that is collected is sent to the substations where it is monitored by remote terminal units (RTU). The proposed access control mechanism uses attribute-based encryption (ABE) which gives selective access to consumer data stored in data repositories and used by different smart grid users. RTUs and users have attributes and cryptographic keys distributed by several key distribution centers (KDC). RTUs send data encrypted under a set of attributes. Since RTUs are maintained in the substations they are well protected in control rooms and are assumed to be trusted. Users can decrypt information provided they have valid attributes. The access control scheme is distributed in nature and does not rely on a single KDC to distribute the keys which makes the approach robust. To the best of our knowledge, ours is the first work on smart grids, which integrates these two important security components (privacy preserving data aggregation and access control) and the first paper which addresses access control in smart grids.**

***Index Terms*—Access control, bilinear maps, decentralized attribute-based encryption, homomorphic encryption, smart grids.**

I. Introduction

SMART GRIDS are the next generation electricity grid system which will integrate power and communication networks. With the growing demand for electricity, there is a need for developing smart grids which can cope up with the demand by intelligently using different power resources and integrating different components like vehicles and wireless devices. Smart grids should have capabilities that would enable it to deal with power outages by balancing supply and demand. This can be achieved by intelligently balancing the consumption between peak and off-peak periods. One recent suggestion has been to charge electric vehicles (also incorporated into the grid) during the off-peak period and discharge it back into the grid. In this way the grid is bi-directional, energy can be used when needed and discharged back into the grid when not needed.

Manuscript received March 31, 2012; revised September 06, 2012; accepted October 02, 2012. Date of publication January 14, 2013; date of current version February 27, 2013. Paper no. TSG-00160-2012.

S. Ruj is with the CSE, Indian Institute of Technology, Indore, India (e-mail: sush@iiti.ac.in).

A. Nayak is with the School of Electrical Engineering and Computer Science (EECS), University of Ottawa, ON K1N 6N5, Canada (e-mail: anayak@site.uottawa.ca).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSG.2012.2224389

The operation of smart grid involves many aspects: generation of power using alternate sources (e.g., solar, wind, geothermal, nuclear, fossil-fuel), intelligent distribution of power by monitoring the demand of power in different regions and different customers, monitoring the power usage by customers using smart meters and intelligently deliver power when needed, building and integrating appliances, like vehicles (plugged-in electric vehicles—PHEV) and wireless devices, into the grid.

Research in smart grid is very important and involves a broad range of problems. An important problem is to design an architecture integrating all the components, which can efficiently use electricity. Smart grid architectures have been proposed and discussed by Bose [2]. It comprises of power infrastructure and information infrastructure [3]. The power infrastructure consists of power equipment like generators, transformers, transmission lines, voltage regulators, capacity banks, meters, etc., which help to deliver electricity. The power infrastructure involves generation of power from different sources and their reliable and efficient transmission. The information infrastructure helps in communication and ensures safety and reliability of the smart grid. It measures the status of the devices in the grid, balances demand and supply, helps in diagnosis of faults, helps authentication of devices, and helps in the smooth operation of plugged-in devices like vehicles. Devices might have sensors to sense different conditions and can be simple devices as smoke detectors and automatic light switches, etc.

An important problem which is associated with smart grid is the problem of security and privacy. It is very important to secure the smart grid, not only from terrorist attacks, but also from customers and building authorities who can tamper with various devices. The information from remote terminal units (RTU) at the substation is needed not only for electricity distribution purpose, but also for calculating cost, predicting future conditions, and monitoring unexpected behavior. Since the data at the substations is sensitive and needs to be well protected, the RTUs are placed in control rooms under strict supervision. The different tasks are performed by separate users; for example, the electrical and maintenance board will monitor the network, the costs calculation and analysis is done by the auditing unit, and network planners/researchers can be involved to predict future behavior. All information must be sent only to the users responsible for a specific job. Access control thus becomes a very important issue in smart grids. In future, when content distribution will also be included into the smart grid (our assumption is that future smart grids will also have cable integrated into it), it will be necessary to regulate access such that two or more users do not collude and access information they cannot individually access.

Currently, there is not much work which integrates different security components in a smart grid. Existing literature focus

either on authentication [4], [5] or privacy protection [6]or aggregation [8] separately. Therefore, there is a need for a framework where different parts of the smart grid can work securely in tandem. There is a need for privacy preserving data aggregation at the home, building and neighborhood area network, so that, smart meters are not aware of the data that they are aggregating. Such information is stored securely in the data repository. Only authorized users can access the information. Hence, there is a need for fine-grained access control.

Data aggregation proposed by Li *et al.* [8] is limited in scope. It presents privacy-protected data aggregation in a local neighborhood (typically a building area network) without focusing on large scale aggregation. Access control in power grids has been studied by Bobba *et al.* [1]. This cannot be used in our smart grid setting for the following reasons: 1) it has one centralized key distribution center (KDC) to distribute secret credentials, which is prone to single point failure. Terrorists can easily launch an attack and undermine the whole grid system, 2) the KDC has to be online all the time, which requires it to be highly efficient and energy rich center, and 3) smart grids also include a communication network, so compromising the single KDC can result in a huge loss of personal information.

For these reasons, we aim at designing a robust decentralized framework that should take into consideration the following situations. The KDCs need not be online all the time. Data and information collected by the RTUs can be shared amongst different types of users and administrators of the grid. Data sharing can depend on context, and the number of users might not be known in advance. Access control is important because sometimes there is sensitive information, which should be known only to a selected group of people. For example, if the RTU senses some abnormal readings, which indicate a major failure, the analyst or maintenance units should only know. If common people or media comes to know then there can be panic and emotional distress amongst people. It is essential that the data be sent along with some access policy, so that only certain users can decrypt.

In this paper, we propose a new decentralized security framework for smart grids, integrating simultaneously privacy preserving aggregation and access control. Aggregation of data at gateway smart meters of Home Area Network (HAN), Building Area Network (BAN), Neighborhood Area Network (NAN) is done using homomorphic encryption [10]. The aggregation is lossless, and computes the aggregated results correctly. Other encryption techniques like RSA encrypts a message with the public key of the receiver, such that the receiver can decrypt it using its secret key. The message is send and received as is. In our problem, we need to aggregate results and send the aggregated results to the destination. Thus, the message is not sent as is, but changed from time to time. We calculate the aggregated results on the ciphertexts, without disclosing the plaintext messages. This would not have been possible using RSA. This helps to process information without knowing what information is being processed, thus preserving privacy of the readings from the gateways. Our framework is decentralized and consists of multiple KDCs unlike just one in [1], thereby eliminating any single point of failure. Moreover, the KDCs need not be online all the time while the system is in operation. The access control scheme, which is based on a cryptographic primitive called attribute based encryption (ABE) [7], gives limited access to data users like audit teams, technical maintenance teams, engineers, environmentalists, research groups, policy makers, management groups, etc. Users have attributes and access policies are function of these attributes. The broadcasted message is encrypted once in such a way, that, only authorized users (with valid access policy) can decrypt it. Our scheme is collusion secure, in the sense that no two users can collude and gain access to data they alone cannot avail. Malicious and illegal users can be revoked with relatively low overhead in our security framework. Our preliminary work on this topic appears in [9]. To the best of our knowledge, this is the first work that attempts to build a unified framework, where privacy preserving data aggregation and access control of collected data have been addressed at the same time.

The paper is organized as follows. We present related work on security and privacy issues of smart grids in Section II. In Section III, we describe mathematical tools used in our work. We present our smart grid security framework in Section IV and discuss data aggregation in details in Section IV-B and access control in Section IV-C. In Section V, we analyze the security and performance issues of our scheme and compare with existing ones. We conclude in Section VI.

II. Related Work

As we noted in the introduction that security is an important aspect of smart grid, not only to protect from military threats but also protect from misbehaviors of consumers and different service providers integrated into the grid. Security issues in smart grid mainly focus on authenticating customer, operators, and service providers. There are several components in smart grids like SCADA (Supervisory Control and Data Acquisition), cellular and mobile links, fiber optic cables, etc. Security of each of these components is essential in securing the grid [12]. The cyber security requirements of smart grids have been outlined by the National Institute of Standards and Technology (NIST) [13]. The PKI infrastructure has been proposed to protect smart grids. This infrastructure should provide certification to various components and devices in the network. Specific certification policies need to be issued. Device attestation (ensuring the validity of the device) is an important requirement, since an invalid device can collect and send wrong electricity readings and can result in overloading and failure. Authentication in smart grid has been studied by Fouda *et al.* [4] and Kgwadi-Kunz [14]. Authentication is achieved using Diffie Hellman key agreement protocol, Sign-and-Mac (SIGMA) and Internet Key Exchange (IKEv2) [15].

Privacy in smart grid has been extensively studied because of its importance. Studying the details of consumed data helps deduce the behavioral pattern to a certain extent.

In order to provide anonymity to the metering data, Efthymiou and Kalogridis [16] proposed a third party key escrow policy and uses several pseudonymous IDs instead of unique identifiers. Rial and Danesiz [17] proposed a privacy preserving protocol for smart meters using zero knowledge proof [18], that ensures correct payment of fees without disclosing the details about information on electricity consumption. The protocol is implemented into smart meters and is generic enough to consider different billing settings like electronic traffic pricing, pay-as-you-drive car insurance etc.

TABLE I
NOTATIONS

Symbols	Meanings
U_u	u-th User
T_i	i-th RTU
han_i	Gateway smart meter at i-th HAN
ban_i	Gateway smart meter at i-th BAN
nan	Gateway smart meter at NAN
A_j	KDC j
$\mathcal{A}$	Set of KDCs
$\mathcal{W}$	Set of attributes
$w = \|\mathcal{W}\|$	Number of attributes
L_j	Set of attributes that KDC A_j possesses
$l_j = \|L_j\|$	Number of attributes that KDC A_j possesses
$I[j,u]$	Set of attributes that A_j gives to user U_u
I_u	Set of attributes that user U_u possesses
$PK[j]/SK[j]$	Public/secret key of entity j
$sk_{i,u}$	Secret key given by A_j corresponding to attribute i given to user U_u
S	Boolean access structure
R	Access matrix of dimension $n \times h$
$\|G\|$	Order of group G
M	Message
P_j	Power consumption by gateway at jth HAN
C, c	Ciphertext
$PKT[i]$	Packet sent by smart meter gateway i
H	Hash function, example SHA-1

Privacy preserving communication for plugged-in electrical vehicles has been studied in [19].

Bobba *et al.* [1] presented a centralized access control scheme for power grids. As mentioned before, in the introduction, this scheme has a drawback because centralized authority can be a single point of failure. It also requires that the KDC is online during data transfer. So, the system is affected when the KDC is faulty or switched off for maintenance. The access policies are implemented in XML, and the encryption mechanism uses KEM-DEM hybrid encryption paradigm introduced by Cramer and Shoup [20]. This becomes complicated when there are many KDCs that need to coordinate amongst themselves. This scheme was discussed in connection to access control in power grids in [1] and might not be feasible in smart grids. There has also been a recent proposal on applicability of ABE to smart grids in [21]; however, this work also takes a centralized approach and has only one KDC, thus prone to single point failure.

III. BACKGROUND

Table I presents the notations used throughout the paper. We also describe mathematical background used in our proposed solution.

A. Mathematical Background

We will use bilinear pairings on elliptic curves for the access control protocol. Let G be a cyclic group of prime order q generated by g. Let G_T be a group of order q. We can define the map $e : G \times G \rightarrow G_T$. The map satisfies the following properties:

1) $e(aP, bQ) = e(P, Q)^{ab}$ for all $P, Q \in G$ and $a, b \in \mathbb{Z}_q$, $\mathbb{Z}_q = \{0, 1, 2, \ldots, q-1\}$.
2) Non-degenerate: $e(g, g) \neq 1$.

We use bilinear pairing on elliptic curves groups. Weil and Tate pairings [22] are commonly used for bilinear pairing. The choice of curve is an important consideration, because it determines the complexity of pairing operations.

B. Structure of Access Policies

Access policies are functions of attributes and can be in either format: 1) Boolean functions of attributes, or 2) Linear Secret Sharing Scheme (LSSS) matrix. Any access structure can be converted into a Boolean function [7]. An example of a Boolean function is $((a_1 \wedge a_2 \wedge a_3) \vee (a_4 \wedge a_5)) \wedge (a_6 \vee a_7))$, where, $a_1, a_2, \ldots, a_7$ are attributes. Boolean functions can also be represented by access tree, with attributes at the leaves and $AND(\wedge)$ and $OR(\vee)$ as the intermediate nodes and root.

C. Homomorphic Encryption Scheme

Several encryption techniques exist which support different homomorphism, like multiplicative homomorphism (RSA [24]), additive homomorphism (Paillier [10], Boneh-Goh-Nissim [25]), or recently proposed fully homomorphic scheme [26] which can support complicated functions. Since we are only interested in aggregating information, we use additive homomorphism[10]. We will first discuss the encryption protocol and show how it can be used to support homomorphism. Let i be the receiver for whom a message is intended. The protocol consists of three algorithms:

1) Key generation: This algorithm generates the public keys and global parameters, given a security parameter. Let $N = q_1 q_2$, where q_1 and q_2 are primes. Choose $g \in \mathbb{Z}^*_{N^2}$, such that g has an order which is a multiple of N. Let $\lambda(N) = lcm(q_1 - 1, q_2 - 1)$, where lcm represents the least common multiple. Then, the public key of i is $PK[i] = (N, g)$ and secret key $SK[i] = (\lambda(N))$.
2) Encryption: Let $M \in \mathbb{Z}_N$ be a message. Select a random number $r \in Z^*_N$. The ciphertext c is given by
$$c = E(M) = g^M r^N \bmod N^2. \tag{1}$$
3) Decryption: To decrypt c, M can be calculated as
$$M = D(c) = \frac{L(c^{\lambda(N)} \bmod N^2)}{L(g^{\lambda(N)} \bmod N^2)} \bmod N, \tag{2}$$
where the $L-$ function takes input from the set $\{u < N^2 | u = 1 \bmod N\}$ and computes $L(u) = (u-1)/N$.

Additive homomorphism is demonstrated in the following way. Suppose $c_1 = E(M_1)$ and $c_2 = E(M_2)$ are two ciphertexts, for $M_1, M_2 \in \mathbb{Z}_N$. Then, $D(c_1.c_2 \bmod N^2) = M_1 + M_2 \bmod N$. Thus, the sum of the plaintexts can be obtained from the ciphertext.

We note that r^N is used only to make the homomorphic computation indeterministic; the same message can be encrypted into different ciphertexts to prevent dictionary attacks.

D. ABE Scheme

ABE is a cryptographic protocol proposed by Sahai and Waters in 2005 [27]. The main idea is to distribute attributes to receivers and attributes to senders so that only the receivers with matching attributes structure can access the data. Data is encrypted using attribute based keys, which are distributed by a central key distribution center (KDC). The protocol proposed by Sahai and Waters is restricted to only threshold access structures (t-out-of-n) that could be supported. This means that if the receiver has t attributes (out of n) in common to the sender, then it can decrypt the message. Goyal *et al.* [28] proposed a new ABE which can handle any monotonic access structure. These

schemes are known as key-policy based (KP-ABE) schemes. Another type of protocols is known as ciphertext-policy ABE (CP-ABE) [29] (proposed by Bethencourt). In these, the ciphertext is encrypted using a set of attributes under a given access structure. If a receiver has a matching set of attributes then it can decrypt the information.

All the above schemes rely on a central key and attribute distribution center, which is prone to failures. Chase [30] proposed a multi-authority (same as multi-KDC) protocol, where several KDCs generate and distribute keys and attributes. There is also a central trusted authority who coordinates the multiple KDCs. To completely do away with central authority, Chase and Chow [31] proposed a scheme where the authorities can coordinate amongst themselves, but do not require a central authority. The drawback of this protocol is that the access structure is specific and requires each user to have at least one attribute from each KDC. Both these scheme are KP-ABE. Recently, Lewko and Waters [7] proposed a multi-KDC CP-ABE which does not have trusted authority and coordination between the KDCs. It also allows any type of monotonic access structure. We use Lewko and Waters scheme modified suitably to design an access control mechanism for smart grids because it is a decentralized approach.

We will discuss Lewko and Waters [7] scheme which we will later modify to include user revocation in Section IV-D. The scheme consists of four steps: 1) System Initialization, 2) Key and attribute distribution to users by KDCs, 3) Encryption of message by sender, and 4) Decryption by receiver.

1) System Initialization: Select a prime q, generator g of G, groups G and G_T of order q, a map $e : G \times G \longrightarrow G_T$, and a hash function $H : \{0,1\}^* \longrightarrow G$ which maps the identities of users to G. The hash function used here is SHA-1 [32]. Each KDC $A_j \in \mathcal{A}$ has a set of attributes L_j. The attributes are disjoint ($L_i \bigcap L_j = \phi$ for $i \neq j$). Each KDC also chooses two random exponents $\alpha_i, y_i \in \mathbb{Z}_q$. The secret key (SK) of KDC A_j is

$$SK[j] = \{\alpha_i, y_i, i \in L_j\}. \tag{3}$$

The public key (PK) of KDC A_j is published:

$$PK[j] = \{e(g,g)^{\alpha_i}, g^{y_i}, i \in L_j\}. \tag{4}$$

2) Key Generation and Distribution by KDCs: User U_u receives a set of attributes $I[j,u]$ from KDC A_j and the corresponding secret key $sk_{i,u}$ for each $i \in I[j,u]$

$$sk_{i,u} = g^{\alpha_i} H(u)^{y_i}, \tag{5}$$

where $\alpha_i, y_i \in SK[j]$. Note that all keys are delivered to the user securely using the user's public key, such that only that user can decrypt it using its secret key.

3) Encryption by Sender: Sender decides about the access tree. LSSS matrix R can be derived as described in [7]. The algorithm $ABE.Encrypt(M, R, \pi)$ calculates ciphertext with input message M, the LSSS access matrix R, and the mapping π of the rows of R to the attributes. Sender encrypts message M as follows:

1) Choose a random seed $s \in \mathbb{Z}_q$ and a random vector $v \in \mathbb{Z}_q^h$, with s as its first entry; h is the number of leaves in the access tree (equal to the number of rows in the corresponding matrix R).
2) Calculate $\lambda_x = R_x \cdot v$, where R_x is a row of R
3) Choose a random vector $w \in \mathbb{Z}_q^h$ with 0 as the first entry.
4) Calculate $\omega_x = R_x \cdot w$
5) For each row R_x of R, choose a random $\rho_x \in \mathbb{Z}_q$.
6) The following parameters are calculated:

$$\begin{aligned} C_0 &= Me(g,g)^s \\ C_{1,x} &= e(g,g)^{\lambda_x} e(g,g)^{\alpha_{\pi(x)}\rho_x},\ \forall x \\ C_{2,x} &= g^{\rho_x} \forall x \\ C_{3,x} &= g^{y_{\pi(x)}\rho_x} g^{\omega_x} \forall x, \end{aligned} \tag{6}$$

where $\pi(x)$ is mapping from R_x to the attribute i that is located at the corresponding leaf of the access tree.

7) The ciphertext C is sent by the sender (it also includes the access tree via R matrix):

$$C = \langle R, \pi, C_0, \{C_{1,x}, C_{2,x}, C_{3,x}, \forall x\}\rangle. \tag{7}$$

4) Decryption by Receiver: Receiver U_u takes as input ciphertext C, secret keys $\{sk_{i,u}\}$, group G, and outputs message M. The function $ABE.Decrypt(C, \{sk_{u,i}\})$ is the decryption function. It obtains the access matrix R and mapping π from C. It then executes the following steps:

1) U_u calculates the set of attributes $\{\pi(x) : x \in X\} \bigcap I_u$ that are common to itself and the access matrix. X is the set of rows of R.
2) For each of these attributes, it checks if there is a subset X' of rows of R, such that the vector $(1, 0 \ldots, 0)$ is their linear combination. If not, decryption is impossible and $ABE.Decrypt(C, \{sk_{u,i}\}) = NULL$. If yes, it calculates constants $k_x \in \mathbb{Z}_q$, such that, $\sum_{x \in X'} k_x R_x = (1, 0, \ldots, 0)$. K is a vector consisting of k_x, $x \in X'$.
3) Decryption proceeds as follows:
   a) For each $x \in X'$, $dec(x) = C_{1,x} e(H(u), C_{3,x}) / e(sk_{\pi(x),u}, C_{2,x})$
   b) U_u computes $M = ABE.Decrypt(C, \{sk_{u,i}\}) = C_0 / \Pi_{x \in X'} dec(x)$.

IV. Proposed Framework for Secure Smart Grids

Our proposed framework is shown in Fig. 1. There are two types of networks involved: power network and communication network. Electricity is generated at the power plants and distributed by electric substations for use by customers. This constitutes the power network. The communication network consists several components involving smart meters, home appliances, wi-fi technologies, PHEV components. The power flow is shown in bold lines and information flow in dotted lines. We focus on two parts. The first part is to collect data from consumers and aggregate them at different levels. The aggregation network consists of home area networks (HAN), building area network (BAN), and neighborhood area network (NAN) which reports to a substation. For each home area network there is a gateway smart meter han which collects information and sends to the building area network. The gateway ban aggregates all information from smart meters in the BAN and sends to the nan at the neighborhood area. nan reports to the substation.

The second part is similar to currently deployed Control and Data Acquisition, and Energy Management System (SCADA/EMS). It consists of remote terminal units which collect information from the NAN and other sources like PHEVs and sends

to the SCADA/EMS. In our model, SCADA/EMS consists of data repository which stores the data collected by the RTU. It also has data processors to process data. The data aggregation network in Fig. 1 collects and aggregates data and sends to the RTU at the nearest substation. Aggregation at each stage uses Paillier additive homomorphic encryption [10] which ensures that data can be aggregated knowing only the ciphertext, so that the plaintext can be hidden. This will protect the privacy of individuals as well as a particular locality. There are also key distribution centers that distribute keys to RTU and users. Users can be system engineers, maintenance offices, auditors, policy makers, researchers, etc.

Access control is achieved using a recent cryptographic primitive known as decentralized attribute based encryption [7]. Each RTU and user has attributes that can be one or more of the following types (but not limited to):

1) Type of energy source: fossil fuel, solar, hydroelectricity, wind.
2) Type of consumer: Individual, corporate, PHEV.
3) Location of the consumer: City, region.
4) Type of appliances: Need based. For example, essential like light, heat, etc. Lower priority: Dryer, washing machine.
5) Type of appliances: Load based. High electricity consumption equipment like dryer, oven, etc., and low electricity consumption equipment like lights, television, etc.
6) Type of user: Electrical engineer, power engineer, environmentalists, policy makers, etc.

There are a number of key distribution centers (KDCs), which distribute secret credentials to the users and RTUs. For example, there might be a KDC who is responsible for managing power resources like Hydroone in Canada, or Statistics Canada (that keeps track of consumers within a city or a region) or Electrical Safety Authority-ESA (which regulates the safety measures for electrical appliances) who assigns secret credentials accordingly. These KDCs decide the attributes and create public and secret keys for users as given in Section III-D. KDCs are used to distribute secret credentials and need not be online during operation. The RTU builds an access policy like time of collection (peak/offpeak time), type of user who can access (like engineers, environmentalist), location of the user (region/city), etc.

Different users of data can access information stored in the databases, provided they have a valid access structure. For example, a maintenance unit might want to collect information from residential and corporate users, which run on fossil fuel and which have consumed more than a certain amount of electric power per day. Researchers, on the other hand, might be interested in predicting load due to charging and discharging of plugged in hybrid electric vehicles (PHEV) during day time. The KDCs distribute attributes and public and private keys to each RTU. The RTUs, which might have specific access policies, encrypt the data with keys (depending on the access policy) and send to the storage units.

Assumptions

We assume that each device has a unique identity (an IP address) and can authenticate itself before interacting with the network. We will not design an authentication protocol here, but rely on the authentication protocol [4], which has been designed specially for smart grid communication.

We assume that the adversaries are honest but curious. This means that they always send correct results without tampering with them. They can however read intermediate values. This is a valid assumption and has been adopted in [8], [11]. Honest-but-curious model of adversaries do not tamper with data so that they can keep the system functioning normally and remain undetected.

We also assume that the data storage center is honest but curious. This means, it can attempt to read the contents of the ciphertext and the attributes that the ciphertext might be carrying. The RTUs are also honest but curious, so we hide the privacy of individual customers. However, we must remember that when the RTUs are sending messages, they can choose their access policies according to their discretion, depending upon the data they are sending. Honest but curious model is an acceptable model in data storage applications like clouds and smart grids [8], [11].

A. Secure Aggregation by Smart Meters

In this section, we discuss how aggregation takes place at the gateway smart meters han, ban and nan before it reaches the substation. We assume that the following architecture exists: The household meters collect readings from different equipments and send to the gateway smart meter at the HAN. The gateway smart meters han send their aggregated results to the ban. The gateway smart meter ban aggregates all the readings from the gateway meters at HAN meters and sends to the NAN. The gateway smart meter nan aggregates all the readings from the gateway BANs and sends to the nearest substation. This is depicted in Fig. 1.

An RTU T_i at a substation is securely given public key $PK[i] = (N, g)$ (as in key generation step in Section III-C) and also the secret key $SK[i] = \lambda(N)$. Each smart meter in the network knows the public key $PK[i] = (N, g)$ of its nearest RTU substation T_i. Each gateway smart meter han_j sends a data packet which consists of two fields: the attributes field f and the power consumption field P_j. The power consumption field is encrypted with the public key of the substation. A packet looks like

$$PKT[han_j] = f \| c_j = f \| E(P_j), \tag{8}$$

where $E(P_j) = g^{P_j} r_j^N \bmod N^2$ ($r_j \in \mathbb{Z}_N^*$ is chosen randomly by the smart meter).

This packet is then send to the gateway BAN, ban_l which aggregates all the results. Here it checks for the attributes field. For packets which have the same set of attributes, it processes the aggregated power consumption. The aggregated result is given by $c_{ban_l} = \Pi_{j \in HAN} c_j$. The new packet looks like $PKT[ban_l] = f \| c_{ban_l}$.

The packets collected by the gateway BANs are then send to the NAN. It performs a similar operation and aggregates information from packets having same set of attributes. The aggregated result is $c_{nan} = \Pi_{ban_l \in BAN} c_{ban_l}$. The packet $PKT[nan] = f \| c_{nan}$ is then sent to the nearest substation.

The RTU T_i at the substation reads the content of the packet. It then decrypts the aggregated result because it has the secret key $SK[i]$.

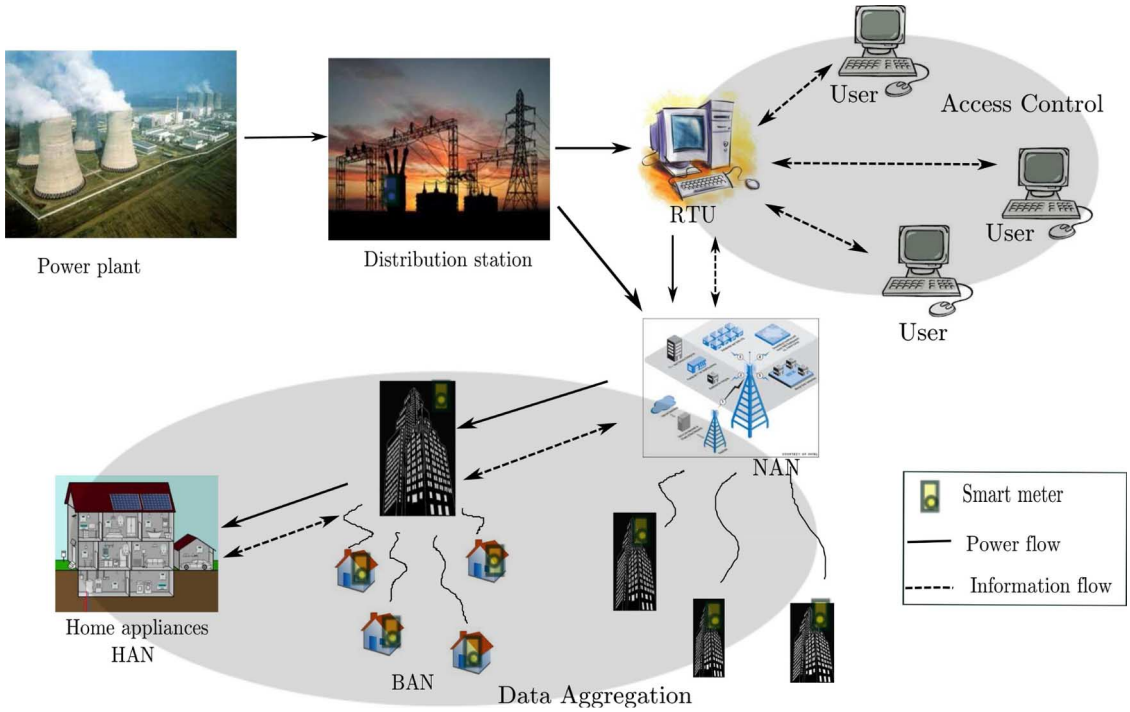

Fig. 1. Aggregation and access control framework.

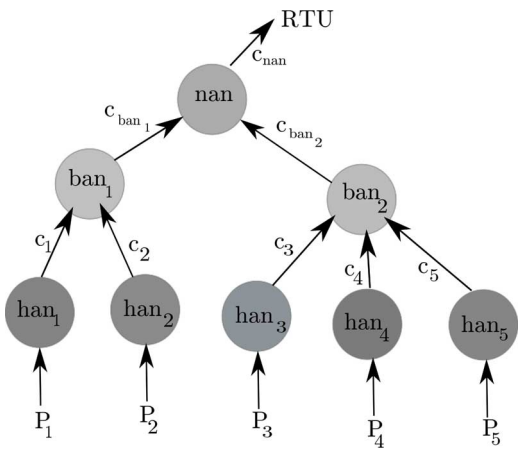

Fig. 2. Example showing data aggregation.

We note that

$$\begin{aligned} c_{nan} &= \Pi_{ban_l \in BAN}(\Pi_{j \in HAN} c_j) \\ &= \Pi_{ban_l \in BAN}(g^{\sum_{j \in HAN} P_j})(\Pi_{j \in HAN} r_j)^N \bmod N^2 \\ &= g^{\sum_j P_j}(\Pi_j r_j)^N \bmod N^2 \end{aligned}$$

Using the value of $\lambda(N)$, the aggregated message can be decrypted by the RTU (as given in Section III-C).

We next consider a very small example to show how this works in practice.

Example We show only the data having same set of attributes. The aggregation network is shown in the Fig. 2.

The HANs collect data from different devices and the encrypted data $c_1, c_2, \ldots, c_5$ to the respective BANs. Here $c_i = g^{P_i} r_i^N \bmod N^2$, for $i = \{1, 2 \ldots, 5\}$. The BAN gateways aggregate the results. ban_1 calculates

$$c_{ban_1} = c_1 c_2 = g^{P_1+P_2}(r_1 r_2)^N \bmod N^2,$$

while ban_2 calculates

$$c_{ban_2} = c_3 c_4 c_5 = g^{P_3+P_4+P_5}(r_3 r_4 r_5)^N \bmod N^2.$$

The BAN gateways then send to the NAN, which aggregates the result as

$$\begin{aligned} c_{nan} &= c_{ban_1} c_{ban_2} = c_1 c_2 c_3 c_4 c_5 \\ &= g^{P_1+P_2+P_3+P_4+P_5}(r_1 r_2 r_3 r_4 r_5)^N \bmod N^2 \end{aligned}$$

When the RTU receives ciphertext c_{nan}, it decrypts using its secret key $\lambda(N)$ as

$$\begin{aligned} D(c_{nan}) &= \frac{L(c_{nan}^{\lambda(N)}) \bmod N^2)}{L(g^{\lambda(N)}) \bmod N^2)} \bmod N \\ &= \frac{L(g^{(P_1+P_2+P_3+P_4+P_5)\lambda(N)}) \bmod N^2)}{L(g^{\lambda(N)}) \bmod N^2)} \bmod N \\ &= P_1 + P_2 + P_3 + P_4 + P_5. \end{aligned}$$

This is because $(r_1 r_2 r_3 r_4 r_5)^{N\lambda(N)} = 1 \bmod N^2$.

Thus, the RTU can aggregate information without knowing what data was send by the smart meters at the HANs.

B. Access Control Scheme

The parameters are chosen and distributed to the KDCs when they are installed. The attributes and key generation has been

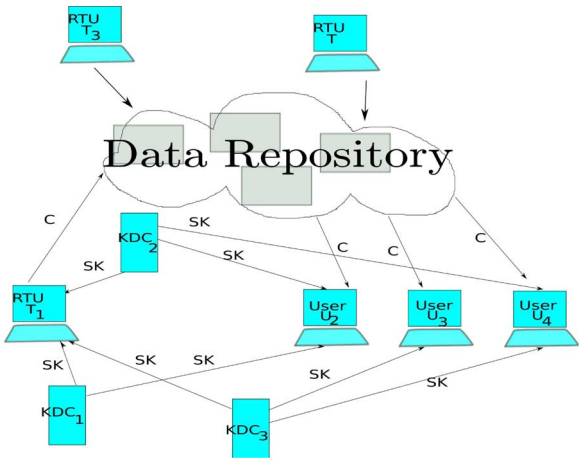

Fig. 3. Access control in presence of multiple KDCs.

presented in III-D. Fig. 3 shows the access control network, consisting of RTUs, KDCs, data repository and users. KDCs distribute keys as given in Section III-D.

Encryption proceeds in two steps. The Boolean access tree is first converted to LSSS matrix. In the second step, the message is encrypted and sent to the data storage center along with the LSSS matrix. A secure channel like ssh can be used for the transmission.

Suppose an RTU T_i wants to store a record M. T_i defines the access structure S, which helps it to decide the authorized set of users, who can access the record M. It then creates a $m \times h$ matrix R (m is the number of attributes in the access structure) and defines a mapping function π of its rows with the attributes (using algorithm below). π is a permutation, such that $\pi : \{1, 2, \ldots, m\} \rightarrow \mathcal{W}$. The RTU runs the encryption algorithm and calculates ciphertext $C = ABE.Encrypt(M, R, \pi)$. Ciphertext C is then stored in the data repository.

When a user U_u requests a ciphertext from the repository, the requested ciphertext C is transferred using ssh protocol. The decryption algorithm proceeds as in Section III-D4, and returns plaintext message $M = ABE.Decrypt(C, \{sk_{u,i}\})$, if the user has valid set of attributes or NULL otherwise. We will next consider an example to illustrate access control.

Example: Suppose an RTU sends a data record to the data repository. This data can be the amount of electricity consumed over a certain period of time by high-consumption equipments which are run by fossil fuels. There can be three KDCs which distribute the following types of attributes: 1)Type of users: D_1 (Researchers) , D_2 (policy makers), D_3 (Power engineers), D_4 (Environmentalists), etc., 2)Type of appliance: E_1 (High consumption), E_2 (Low consumption), etc., and 3)Source of power: S_1 (fossil-fuels), S_2 (solar), etc. The RTU can give access either to researchers and policy makers ($D_1 \vee D_2$) or give selective access to environmentalist working on fossil-fuels ($D_4 \wedge S_1$) or power engineers who are supervising the usage of high-consumption equipment ($D_3 \wedge E_1$).

Then the access tree is given in Fig. 4. Given the access tree, the LSSS matrix is constructed in the following way. The root has vector (1). Let $v[x]$ be parent's vector. If node $x =$ AND, then the left child is $(v[x]|1)$, and the right child is $(0, \ldots, -1)$. If $x =$ OR, then both children also have unchanged vector $v[x]$.

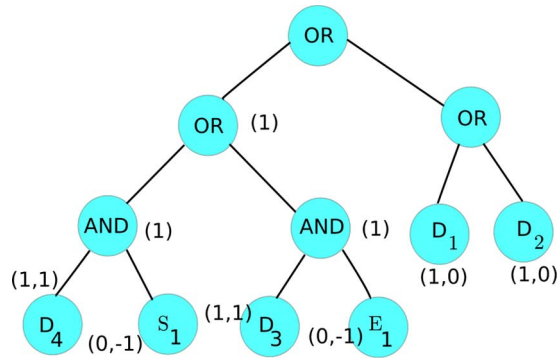

Fig. 4. Access tree structure.

Finally, pad with 0 s in front, such that all vectors are of equal length. The proof of validity of the algorithm is given in [23].

The LSSS access matrix R corresponding to Fig. 4 is given by,

$$R = \begin{pmatrix} 1 & 1 \\ 0 & -1 \\ 1 & 1 \\ 0 & -1 \\ 1 & 0 \\ 1 & 0 \end{pmatrix}.$$

An environmentalist working on fossil fuels will be able to access this data, as also a power engineer monitoring high-consumption equipment. However, an electrical engineer working on solar cells will not be able to read it.

Let there be three KDCs A_1, A_2 and A_3. The set of attributes of A_1, A_2 and A_3 are $L_1 = \{D_1, D_2, D_3, \ldots\}$, $L_2 = \{E_1, E_2, \ldots\}$, and $L_3 = \{S_1, S_2, \ldots\}$. The RTU's access tree is given by Fig. 4. Each row of the access tree denotes a leaf node (representing an attribute) of the corresponding tree. π represents the mapping from $\{1, 2, \ldots, m = 6\}$ to the set of attributes. Thus, π can be denoted as:

x	1	2	3	4	5	6
$\pi(x)$	D_4	E_1	D_3	S_1	D_1	D_2

Suppose an user (user $u = 3$) is an environmentalist studying fossil-fuels and solar energy, then he/she is given the attributes D_4, S_1 and S_2. Thus, $I[1, 3] = \{D_4\}$ and $I[2, 3] = \{S_1, S_2\}$. Next the user is given secret keys $sk_{4,1}$ from A_1 and $sk_{1,3}$ and $sk_{2,3}$ from A_3.

During encryption, the RTU sends the information $C = \langle R, \pi, C_0, \{C_{1,x}, C_{2,x}, C_{3,x}\}_{x \in \{1,2,3,4,5,6\}} \rangle$ to the data repository. $C_0 = Me(g, g)^s$, where s is chosen at random from $\mathbb{Z}_q$.

When user 3 wants to access the above information C, C is transferred securely using ssh (an inbuilt secure shell standard protocol). The user first finds out the attributes that are present from π. He/she also finds that it has the attributes D_4, S_1 in common to the attribute in data. From the matrix R, it then finds that there are two rows corresponding to D_4 and S_1, such that $(1, 1) + (0, -1) = (1, 0)$ (linear combination of rows 1 and 2 of R gives $(1, 0)$).

The user can thus calculate $e(g, g)^s$ according to Step 4 of the decryption mechanism. Once $e(g, g)^s$ is calculated, M can be obtained. The data repository does not have the secret keys,

and is unable to decrypt the message. Thus, an authorized user $u = 3$ can decrypt the information because it has valid set of keys and access rights.

C. Revocation of Users

Users can be revoked, either because they are faulty or have been tampered with. Once revoked, they should not be able to decrypt messages, even if they have valid attributes. We present a revocation mechanism to achieve this.

For each revoked user U_u, I_u is noted. Once the attributes I_u are identified, all data that possess the attributes are collected. For each such information record, the following steps are then carried out:

1) A new value of s, $s_{new} \in \mathbb{Z}_q$ is selected
2) The first entry of vector v_{new} is changed to new s_{new}
3) $\lambda_x = R_x v_{new}$ is calculated, for each $x \in I_u$
4) $C_{1,x}$ is recalculated for $x \in I_u$
5) New value of $C_{1,x}$ is securely transmitted to the storage center
6) New $C_0 = Me(g,g)^{s_{new}}$ is calculated and stored in the storage center
7) New value of $C_{1,x}$ is not stored with the data, but is transmitted to users, who wish to decrypt the data

We note here that the new value of $C_{1,x}$ is not stored in the data centers but transmitted to the non-revoked users who have attribute x. This prevents a revoked user to decrypt the new value of C_0 and get back the message.

V. Analysis and Performance

A. Security of Aggregation Mechanism

We will first show that the aggregation scheme gives correct results when the intermediate smart meter (HAN, BAN, NAN gateway) is honest. We will then prove that the privacy of not only individual customers but also that of intermediate smart meters in BAN and NAN is preserved.

Theorem 1: The aggregation scheme presented in Section IV-B gives correct results when the intermediate smart meter (HAN, BAN, NAN gateway) is honest.

Proof: We first note that the decryption step given in (2) is correct. $c^{\lambda(N)} \bmod N^2$ and $g^{\lambda(N)} \bmod N^2$ both equal 1, when raised to the power of N. This is because g has an order which is a multiple of N. Thus, $c^{\lambda(N)} \bmod N^2$ and $g^{\lambda(N)} \bmod N^2$ are both N-th roots of unity. Such roots are of the form $(1+N)^{\beta} = (1+\beta N) \bmod N^2$. Hence, the L-function can be computed as $L((g^M)^{\lambda(N)} \bmod N^2) = ML(g^{\lambda(N)} \bmod N^2) \bmod N$. (details of proof appear in [10]). From this, the value of M can be obtained.

For our aggregation scheme,

$$c_{nan} = g^{\sum_j P_j} (\Pi_j r_j)^N \bmod N^2.$$

We note that $((\Pi_j r_j)^N)^{\lambda(N)} = 1 \bmod N^2$. Thus,

$$D(c_{nan}) = \frac{L((g^{\sum_j P_j})^{\lambda(N)} \bmod N^2}{g^{\lambda(N)} \bmod N^2} \bmod N$$
$$= \sum_j P_j, (\text{by similar argument as above}).$$

■

TABLE II
Computation and Communication Costs

Operation	Cost
Encryption	$T_p + 4mT_m$
Decryption	$(2m+1)T_p + 5mT_m$
Communication Costs	$m^2 + m(\lvert G_T\rvert + 2\lvert G\rvert) + \lvert G_T\rvert + \log w + \lvert Data\rvert$
Communication Costs (Revocation)	$m^2 + m(\lvert G_T\rvert + 2\lvert G\rvert) + (m'+2)\lvert G_T\rvert + \log w + \lvert Data\rvert$

Theorem 2: Data aggregation scheme proposed in Section IV-B protects the privacy of customers and all nodes in BAN and HAN.

Proof: Pailler's cryptosystem is intractable under Decisional Composite Residuosity Assumption (DCRA) [10]. A customer sends encrypted data of its power consumption. The data is encrypted using public key of the nearest substation. As such, no user or outsider can decrypt the data unless it knows $\lambda(N)$ which is difficult to solve.

Next, we note that even the RTU at the substation cannot know the individual ciphertexts. This is because it receives encrypted aggregated results from which individual ciphertexts cannot be obtained. The use of the factor r^N while encrypting message (r chosen randomly for each message) helps to transmit the same message as two different ciphertexts and thus prevents dictionary attacks.

Thus, no user/substation can decrypt data that an individual customer sends, thus protecting privacy. ■

B. Security of our Access Control Scheme

We will show that only authorized users (possessing valid set of attributes) can decrypt the data stored in data repositories. The data center cannot change the content of the data stored in the data bases. The data center cannot collude with an user or RTU and decrypt any information it is not supposed to decrypt. No two users can share their attributes and secret keys and decrypt any information they are not supposed to decrypt alone.

Theorem 3: The proposed access control scheme is secure, collusion resistant, allows access of data only to authorized users and protects the privacy of individual consumers.

Proof: We will first show that an user can decrypt data if and only if it has a matching set of attributes. This follows from the fact that access structure S (and hence matrix R) is constructed if and only if there exists a set of rows X' in R, and linear linear constants $k_x \in \mathbb{Z}_q$, such that $\sum_{x \in X'} k_x R_x = (1, 0, \ldots, 0)$. A proof of this appear in [23, Chapter 4]. For an invalid user, there does not exists attributes x, such that $\sum_{x \in X'} k_x R_x = (1, 0, \ldots, 0)$. Thus, $e(g,g)^s$ cannot be calculated. Hence, our scheme allows access of data only to authorized users.

We next show that two or more users cannot collude and gain access to data that they are not individually supposed to access. Suppose that there exist attributes $\pi(x)$ from the colludes, such that $\sum_{x \in X} k_x R_x = (1, 0, \ldots, 0)$. However, $e(H(u), g)^{\omega_x}$ needs to be calculated in Section III-D4. Since different RTUs use different values of $e(H(u), g)$, they cannot decrypt the message even if they combine their attributes. Thus, our access control scheme is collusion secure.

The KDCs work autonomously and are not a part of the data center. We observe that the secret keys $sk_{i,u}$ are obtained from

TABLE III
COMPARISON OF OUR SCHEME WITH BOBBA *et al.* [1] AND FADLULLAH *et al.* [21]

Schemes	Type of Network	Robustness	Access policy	Revocation possible or not	Online/offline KDC
Bobba *et al.* [1]	Power Grids	Not robust (Centralized administration)	Any Boolean function	Yes	Has to remain online
Fadlullah *et al.* [21]	Smart Grids	Not robust (Centralized administration)	Any monotonic Boolean function	No	Need not be online
Our scheme	Smart Grids	Robust (Distributed control)	Any monotonic Boolean function	Yes	Need not be online

multiple KDCs depending upon the attributes in the access policy. Unless all the keys corresponding to the attributes that satisfy the policy are known, it is not possible to decrypt the information. This follows from the description of the decryption algorithm. All the KDCs that have contributed attributes that satisfy the access policy are compromised, the data cannot be decrypted. Thus, no outsider can decode data stored in the repositories, without compromising the relevant KDCs. This makes our scheme secure.

The RTUs receive aggregated results from the HAN, BAN and NAN. The consumers send encrypted data and that is never decrypted at any stage. This protects the privacy of consumer's data. ■

C. Performance Issues

We will first calculate the cost of aggregation. Encryption involves modular exponentiation of element g, which can be done using square-and-multiply technique in $O(\log N)$ time. Decryption involves calculating $L(u)$, which needs only one multiplication. Decryptions can be hastened using the technique already given in [10]. At each smart meter gateway, d values have to be multiplied (where d is the in-degree of that smart meter). So the costs are reasonable.

We will then calculate the computation and communication overhead of access control scheme with and without user revocation. In the first step of encryption, the access tree needs to be converted to an access matrix. Time taken to compute R from S is $O(m)$, where m is the number of attributes in the access structure. To check if there exists a set of rows in R, is equivalent to solving the equation $KR = (1, 0, \ldots, 0)$, for non-zero row vector K. This takes $O(mh)$. Since the list of attributes might not be too large, such overhead is very little.

The most expensive operation during encryption or decryption is pairing. During encryption, each user U_u performs only one pairing operation (to calculate $e(g, g)$). For each row x corresponding to attribute, it also performs two scalar multiplications to calculate $C_{1,x}$, one scalar multiplication to calculate $C_{2,x}$ and one to calculate $C_{3,x}$. Thus, we have a total of $4m$ scalar multiplications. During decryption, there are two pairing operations, one for $e(H(u), C_{3,x})$ and the other for $e(sk_{i,u}, C_{2,x})$, for each x. The number of pairing operations is thus $2m$ to calculate $e(H(u), C_{3,x})$. There are also at most m scalar multiplications to calculate $(e(g, g)^{\lambda_x} e(H(u), g)^{\omega_x})^{\sigma_x}$. Therefore, the computation time is $(2m + 1)T_p + 5mT_m$, where T_p and T_m are the time taken to perform pairing and scalar multiplication.

Implementation of elliptic curves and pairing functions is done using PBC library (Pairing Based Cryptography) [22]. It is a C library, which is built above GNU GMP (GNU Math Precision) library. Using PBC library (Pairing Based Cryptography) [22] on a 32-bit 3 GHz Pentium IV processor [33] and an MNT curve of embedding degree $k = 6$ and $q = 160$ bits, $T_m = 0.6$ ms and $T_p = 4.5$ ms. For an access policy consisting of 10 attributes, encryption takes $T_p + 4mT_m = 8.9$ ms and decryption time for each user is 124.5 ms. The decryption time increases linearly with the number of attributes in the access policy.

Information to be sent from RTU to data repository and from the storage centers to user requires $m \log |G_T| + 2m \log |G| + m^2 + |Data|$ bits, where $|Data|$ is the size of the data. m^2 bits are needed to transfer the matrix R, $m(|G_T| + 2|G|) + |G_T|$ to transfer C_0, $C_{1,x}$, $C_{2,x}$ and $C_{3,x}$, and $\log w$ to send π. Thus, the communication overhead is $m^2 + m(|G_T| + 2|G|) + |G_T| + \log w + |Data|$.

When revocation is required, C_0 needs to be recalculated. $e(g, g)$ is previously calculated, so only one scalar multiplication is needed. If the user is revoked, then for each x, $C_{1,x}$ has to be recomputed. $e(g, g)$ is already computed. Thus, only two scalar multiplication needs to be done, for each x. So a total of $2m' + 1$ scalar multiplications are done by the KDCs, where m' is the number of attributes belonging to all revoked users. Users need not compute any scalar multiplication or pairing operations. Additional communication overhead is $O((m' + 1)|G_T|)$. The computation and communications costs are shown in Table II. Since the encryption is done at the RTU and the decryption at the users which both have sufficient resources, this technique is feasible in smart grids.

D. Comparison With Other Schemes

In this section, we compare our access control scheme with that of Bobba *et al.* [1]. We show (in Table III) that our scheme is more robust than theirs, because ours is a decentralized scheme. The biggest drawback of Bobba *et al.* [1] is that the centralized KDC has to be online all the time to allow access of data. This is a huge restriction, because the system will completely shut off in case of fault or even maintenance.

VI. CONCLUSION

In this paper, we have presented an secure framework in smart grids which integrates aggregation and access control. Homomorphic encryption is used to preserve customer privacy, while ABE is used for achieving access control. ABE has not been used in access control in smart grids though it has been mentioned as a possibility in [1]. The access control architecture is decentralized which makes it more attractive and practical than [1]. In future, we would like to work on integrating smart grids

with clouds. This would enable users not only store information in the cloud but also avail services as and when required.

REFERENCES

[1] R. Bobba, H. Khurana, M. AlTurki, and F. Ashraf, "PBES: A policy based encryption system with application to data sharing in the power grid," in *Proc. ACM ASIACCS*, 2009, pp. 262–275.

[2] A. Bose, "Smart transmission grid applications and their supporting infrastructure," *IEEE Trans. Smart Grid*, vol. 1, no. 1, pp. 11–19, 2010.

[3] Y.-J. Kim, M. Thottan, V. Kolesnikov, and W. Lee, "A secure decentralized data-centric information infrastructure for smart grid," *IEEE Wireless Commun.*, vol. 48, no. 11, pp. 58–65, 2010.

[4] M. Fouda, Z. M. Fadlullah, N. Kato, R. Lu, and X. S. Shen, "A light-weight message authentication scheme for smart grid communications," *IEEE Trans. Smart Grid*, vol. 2, no. 4, pp. 675–685, 2011.

[5] H. Khurana, R. Bobba, T. M. Yardley, P. Agarwal, and E. Heine, "Design principles for power grid cyber-infrastructure authentication protocols," in *Proc. IEEE HICSS*, 2010, pp. 1–10.

[6] S. R. Rajagopalan, L. Sankar, S. Mohajer, and H. V. Poor, "Smart meter privacy: A utility-privacy framework," *Proc. IEEE SmartGridComm*, 2011.

[7] A. B. Lewko and B. Waters, "Decentralizing attribute-based encryption," in *Proc. EUROCRYPT*, 2011, pp. 568–588.

[8] F. Li, B. Luo, and P. Liu, "Secure information aggregation for smart grids using homomorphic encryption," in *Proc. IEEE SmartGridComm*, 2010, pp. 327–332.

[9] S. Ruj, A. Nayak, and I. Stojmenovic, "A security architecture for data aggregation and access control in smart grids," *Arxiv CoRR abs/1111.2619*, 2011.

[10] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *Proc. EUROCRYPT*, 1999, pp. 223–238.

[11] M. Li, S. Yu, K. Ren, and W. Lou, "Securing personaly health records in cloud computing: Patient-centric and fine-grained data access control in multi-owner settings," in *Proc. SecureComm*, 2010, pp. 89–106.

[12] X. Liang, R. Lu, X. Shen, X. Lin, and H. Zhu, "Securing smart grid: Cyber attacks, countermeasures, and challenges," *IEEE Commun. Mag.*, vol. 50, pp. 38–45, 2012.

[13] "Smart grid cyber security strategy and requirements," DRAFT NISTIR 7628, 2010.

[14] M. Kgwadi and T. Kunz, "Securing RDS broadcast messages for smart grid applications," in *Proc. ACM IWCMC*, 2010, pp. 1177–1181.

[15] Internet key exchange (IKEv2) [Online]. Available: http://tools.ietf.org/html/rfc4306

[16] C. Efthymiou and G. Kalogridis, "Smart grid privacy via anonymization of smart metering data," in *Proc. IEEE Int. Conf. Smart Grid Commun.*, 2010, pp. 238–243.

[17] A. Rial and G. Danezis, "Privacy-preserving smart metering," Microsoft Research, Tech. Rep. MSR-TR-2010-150, 2010.

[18] J.-J. Quisquater, M. Quisquater, M. Quisquater, M. Quisquater, L. C. Guillou, M. A. Guillou, G. Guillou, A. Guillou, G. Guillou, S. Guillou, and T. A. Berson, "How to explain zero-knowledge protocols to your children," in *Proc. CRYPTO*, 2009, pp. 628–631.

[19] Z. Yang, S. Yu, W. Lou, and C. Liu, "P^2: Privacy-preserving communication and precise reward architecture for V2G networks in smart grid," *IEEE Trans. Smart Grid*, vol. 2, no. 4, pp. 697–706, 2011.

[20] R. Cramer and V. Shoup, "Design and analysis of practical public-key encryption schemes secure against adaptive chosen ciphertext attack," *SIAM J. Comput.*, vol. 33, pp. 167–226, 2001.

[21] Z. Fadlullah, N. Kato, R. Lu, X. Shen, and Y. Nozaki, "Towards secure targeted broadcast in smart grid," *IEEE Commun. Mag.*, vol. 50, no. 5, pp. 150–156, May 2012 [Online]. Available: http://bbcr.uwaterloo.ca/h8liang/sg/Paper sg commx.pdf

[22] Pairing Based Cryptography Library [Online]. Available: http://crypto.stanford.edu/pbc/

[23] A. Beimel, "Secure schemes for secret sharing and key," Ph.D. dissertation, Technion, Haifa, Israel, 1996.

[24] R. L. Rivest, A. Shamir, and L. M. Adleman, "A method for obtaining digital signatures and public-key cryptosystems (reprint)," *Commun. ACM*, vol. 26, no. 1, pp. 96–99, 1983.

[25] D. Boneh, E.-J. Goh, and K. Nissim, "Evaluating 2-DNF formulas on ciphertexts," in *Proc. TCC*, 2005, pp. 325–341, LNCS 3378.

[26] C. Gentry, "Fully homomorphic encryption using ideal lattices," in *Proc. ACM STOC*, 2009, pp. 169–178.

[27] A. Sahai and B. Waters, "Fuzzy identity-based encryption," in *Proc. EUROCRYPT*, 2005, pp. 457–473, LNCS 3494.

[28] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *Proc. ACM CCS*, 2006, pp. 89–98.

[29] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *Proc. IEEE Symp. Security Privacy*, 2007, pp. 321–334.

[30] M. Chase, "Multi-authority attribute based encryption," in *Proc. TCC*, 2007, pp. 515–534, LNCS 4392.

[31] M. Chase and S. S. M. Chow, "Improving privacy and security in multi-authority attribute-based encryption," in *Proc. ACM CCS*, 2009, pp. 121–130.

[32] D. R. Stinson, *Cryptography: Theory and Practice, Third Edition*. Boca Raton, FL: CRC, 2006.

[33] M. Scott, "Efficient implementation of cryptographic pairings" [Online]. Available: http://ecrypt-ss07.rhul.ac.uk/Slides/Thursday/mscott-samos07.pdf

Sushmita Ruj received her B.E. degree in computer science from Bengal Engineering and Science University, Shibpur, India in 2004, and the M.S. and Ph.D. degrees in computer science from Indian Statistical Institute in 2006 and 2010, respectively.

She was a Postdoctoral Fellow at Lund University, Sweden, and University of Ottawa, Canada, during 2009–2010 and 2010–2012, respectively. She was a Visiting Researcher and Intern at the University of Wollongong, Australia; INRIA, France; and Microsoft Research Lab, India, during the summers of 2007, 2008, and 2009 respectively. She is currently an Assistant Professor, School of Computer Science & Engineering at Indian Institute of Technology, IIT, Indore. Her research interests are in security in mobile ad hoc networks, vehicular networks, cloud security, smart grids, combinatorics and cryptography. She is on the Editorial Board of *Ad Hoc and Sensor Wireless Networks*.

Amiya Nayak received his B.Math. degree in computer science and combinatorics and optimization from University of Waterloo, Canada, in 1981, and the Ph.D. degree in systems and computer engineering from Carleton University, Canada, in 1991.

He has over 17 years of industrial experience in software engineering, avionics and navigation systems, simulation and system level performance analysis. Currently, he is a Full Professor at the School of Electrical Engineering and Computer Science at the University of Ottawa, ON, Canada. His research interests are in the area of fault tolerance, distributed systems/algorithms, and mobile ad hoc networks with over 150 publications in refereed journals and conference proceedings. He is in the Editorial Board of several journals, including IEEE TRANSACTIONS ON PARALLEL & DISTRIBUTED SYSTEMS, *International Journal of Parallel, Emergent and Distributed Systems*, *International Journal of Computers and Applications*, and *EURASIP Journal of Wireless Communications and Networking*.