Human-Factor-Aware Privacy-Preserving Aggregation in Smart Grid

Weiwei Jia, Haojin Zhu, *Member, IEEE*, Zhenfu Cao, *Senior Member, IEEE*, Xiaolei Dong, and Chengxin Xiao

***Abstract*—Privacy-preserving metering aggregation is regarded as an important research topic in securing a smart grid. In this paper, we first identify and formalize a new attack, in which the attacker could exploit the information about the presence or absence of a specific person to infer his meter readings. This attack, coined as human-factor-aware differential aggregation (HDA) attack, cannot be addressed in existing privacy-preserving aggregation protocols proposed for smart grids. We give a formal definition on it and propose two novel protocols, including basic scheme and advanced scheme, to achieve privacy-preserving smart metering data aggregation and to resist the HDA attack. Our protocol ensures that smart meters periodically upload encrypted measurements to a (electricity) supplier/aggregator such that the aggregator is able to derive the aggregated statistics of all meter measurements but is unable to learn any information about the human activities. We present the formal security analysis for the proposed protocol to guarantee the strong privacy. Moreover, we evaluate the performance of our protocol in a Java-based implementation under different parameters. The performance and utility analysis shows that our protocol is simple, efficient, and practical.**

***Index Terms*—Aggregation, differential privacy, privacy preserving, smart grid.**

I. Introduction

SMART grid is the next-generation electricity grid system which is expected to integrate power and communication network together (see Fig. 1). Compared with the traditional grid featured with centralized one-way transmission (from generation plants to customers) and demand-driven response, the envisioned smart grid allows decentralized two-way transmission and reliability- and efficiency-driven response, and aims to provide improved reliability (e.g., self-healing, self-activating, and automated outage management), efficiency (e.g., cost-effective power generation, transmission, and distribution), sustainability(e.g., accommodation of future alternative and renewable power sources), consumer involvement, and security [1].

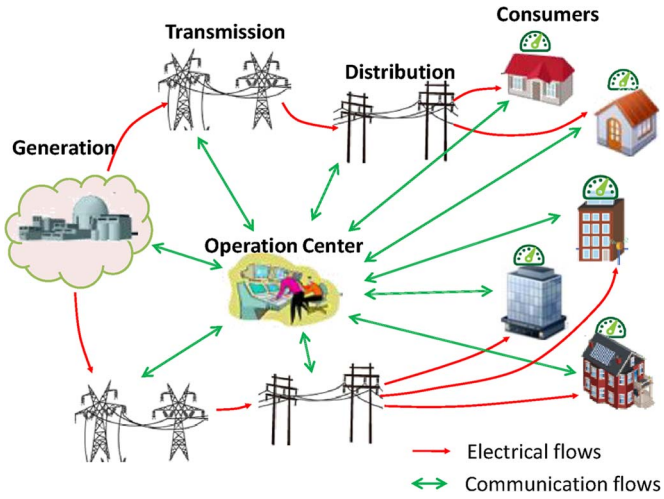


Fig. 1. Structure of a smart grid.

Smart metering is a vital part of the smart grid and has received an increasing attention from both academia and industry. By equipping with smart meters, the smart grid is able to collect real-time information about grid operations and status, thus performing intelligent balancing of the consumption between peak and off-peak periods. For example, it is suggested to charge electric vehicles (also incorporated into the grid) during the off-peak period and to discharge it back into the grid. Due to the obvious advantages of smart meters, the deployments of smart meters are actively promoted by many governments, including the U.S. and the European Union. Smart meters are expected to cover 80% of all households in 2020 [2]. In the U.S., the largest single electric grid modernization investment in U.S. history was launched on October 27, 2009, with the DOE tapping 3.4 billion in American Reinvestment and Recovery Act funds for 100 projects. China is another most active investor in the smart grid infrastructure today, and it will install over 300 million smart meters by the end of 2015, according to the research report of Zpryme.

Security, especially on the aspect of privacy, is regarded as one of the major challenges of the large-scale practical deployment of smart meters. Current smart metering technologies send all personal detailed consumption information to the utilities or a centralized database. As a result, the detailed consumption information will leak the user's behavior information

Manuscript received April 15, 2012; revised October 1, 2012; accepted January 23, 2013. Date of publication June 17, 2013; date of current version April 22, 2014. This work was supported by the National Natural Science Foundation of China under Grants 61033014, 60970110, 60972034, 61003218, and 61272444, by the National Natural Science Foundation of China A3 Foresight Program under Grant 61161140320, by the Doctoral Fund of the Ministry of Education of China under Grant 20100073120065, and by the National Fundamental Research Development Program of China (973) under Grant 2012CB723401.

W. Jia is with the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China, and is also with the College of Sciences, Hohai University, Nanjing 210098, China (e-mail: jlss@sjtu.edu.cn; jiaweiwei@hhu.edu.cn).

H. Zhu, Z. Cao, X. Dong, and C. Xiao are with the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: zhu-hj@cs.sjtu.edu.cn; zfcao@cs.sjtu.edu.cn; dong-xl@cs.sjtu.edu.cn; xcjack@cs.sjtu.edu.cn).

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/JSYST.2013.2260937

[3]. A remarkable number of large electric appliances (e.g., water heaters, well pumps, furnace blowers, refrigerators, and air conditioners) can be identified by appliance load monitoring [4] by analyzing high-resolution consumption data. Further researchers are now focusing on a myriad of small electric devices around the home such as personal computers, laser printers, and light bulbs [5]. In [3], it shows that, from the high-resolution consumption information, complex usage patterns can be extracted with the simple off-the-shelf statistical tools, and then, the extracted information can be used to profile and monitor users for various purposes, creating serious privacy risks and concerns. It could also be demonstrated by the fact that, in 2009, the Senate in Netherlands has stopped a law aimed to make the usage of smart meters compulsory because of the privacy and human rights issues [6]. Therefore, hiding the data privacy of the individual meter readings and, at the same time, allowing the operation centers to obtain overall information in a specific region are two important tasks for securing the smart grid.

There are several existing researches which work on the privacy-preserving aggregation in smart grids. However, the existing works may face the following challenges. First, the existing works are based on the computational expensive operations such as Paillier cryptosystem, which may not be desirable for smart grids, which typically has limited resources, both in terms of bandwidth and computation. Furthermore, the current solutions only prevent leakage of the smart meter's reading privacy to others and seldom take the human factor into consideration. In particular, whether the user is at home or not will have a direct impact on the meter readings. Therefore, even with the privacy-preserving aggregation protocol in place, the attacker can still infer the metering information of the user by comparing the aggregation results before and after the user leaves the home or comes back. After that, the attacker could launch any further attacks by using the estimated meter readings to infer users' behavior patterns. We denote such kind of attack as human-factor-aware differential aggregation (HDA) attack.

To address the aforementioned two challenges, in this paper, we propose a novel human-factor-aware privacy-preserving aggregation protocol to achieve efficient data aggregation without leaking the individual value of the meter readings and, at the same time, to thwart differential aggregation attacks. The contribution of this paper could be summarized as follows.

1) *Formulation of a New Attack in a Smart Grid*. We formulate a new HDA attack in a smart grid. Different from the existing works on privacy-preserving aggregation in a smart grid, we consider the effect of the human factor on the data aggregate. As in Fig. 2, we assume that the electricity consumption aggregation is implemented on five smart meters $(id_1, id_2, \ldots, id_5)$ at two time slots T_1 and T_2 and that smart meter id_5 is the attack target. Assume that the house owner, who is equipped with smart meter id_5, is absent in T_1 while is at home in T_2, and others' electricity consumption is kept relatively stable in these two time slots. Even though each meter sends itself readings in an encrypted form, the attacker can still obtain smart meter id_5's readings by comparing two aggregation results $R_5 = \tilde{R} - \hat{R}$. We give a formal privacy definition on such kind of attack.
2) *High Efficiency and Strong Security Guarantee*. We propose a novel approach, which not only enables efficient privacy-preserving data aggregation but also thwarts human-factor-attack differential privacy-preserving aggregation. In particular, the proposed basic protocol enables the aggregator to obtain the aggregation results without obtaining the individual value. Furthermore, the advanced protocol could provide the HDA attack resisting privacy-preserving aggregation on meters' readings, even under the Byzantine attack (1/3 colluding participants).
3) *Formal Security Proof and Extensive Performance Analysis*. We present the formal security analysis of the proposed protocol to guarantee the strong privacy. Moreover, we evaluate the efficiency of our protocol via extensive performance analysis.

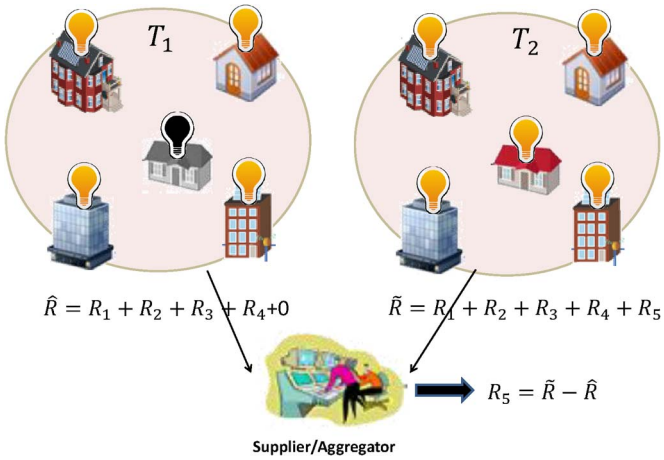


Fig. 2. HDA attack in a smart grid.

The remainder of this paper is organized as follows. In Section II, we review the related works. In Section III, we introduce the preliminaries of our work, including the system model, attack model, and formal security definition. We introduce our basic privacy-preserving metering aggregation protocol in Section IV. In Section V, we propose our advanced protocol to resist the HDA attack. Our performance analysis and experimental evaluation are given in Section VI, which is followed by the conclusion in Section VII.

II. Related Work

Leaking of real-time metering consumption is regarded as a serious privacy risk and concern in the development of the smart grid. Several papers addressed the privacy problems of smart metering in the recent past [3], [5], [7]–[11]. They point out that the accurate high-resolution consumption uploads to the supplier would create serious privacy risks and concerns. It is urgent to develop privacy-preserving smart grid systems that provide strong and provable guarantees.

In traditional cryptographic domain, the related researches on privacy-preserving aggregation include homomorphic encryption and secure multiparty computation (SMPC). Homomorphic encryption [12] allows one to implement homomorphic

operations on ciphertexts under the same key without being able to decrypt. Rather than possibly producing high cost in the implementation of homomorphic encryption, the existing homomorphic encryption schemes could not be directly applied to the case which requires additive operations on the ciphertexts that are encrypted with different users' secret keys. Yang *et al.* [13] proposed an encryption scheme that allows an aggregator to compute the sum over encrypted data from multiple participants. As pointed out by Magkos *et al.* [14], their construction only supports a single time step rather than multiple time steps. SMPC [15], [16] as a well-known cryptographic technique can provide a general method to realize privacy-preserving computation. It addresses the issue on how to privately compute functions $(f_1(x), f_2(x), \ldots, f_n(x))$ on n parties' inputs $x = (x_1, x_2, \ldots, x_n)$. Most of the SMPC protocols require the interaction among the participants. However, such requirements may not be adapted to the smart grid where meters ideally act independently.

Garcia and Jacobs first introduce the concept of privacy-preserving metering aggregation in smart grid [17]. They exploit the homomorphic properties of Paillier encryption on the additive sharing of each meter's readings. By encrypting each share on n meter Pallier public key, every meter helps to compute the share summation. Thus, their protocol requires $O(n^2)$ bytes of interaction between the individual meters as well as relatively expensive cryptography on the meters themselves. Reference [1] uses a superincreasing sequence to structure multidimensional data and to encrypt the structured data by the homomorphic Paillier cryptosystem technique. In [18], Kursawe *et al.* proposes the protocol based on Diffe–Hellman key exchange to add secret value on each meter's readings for each round such that they all add up to zero. However, none of them take the human factor into consideration in terms of privacy preservation in the context of smart grid. In our work, we consider the HDA attack to leak the privacy of users, which is similar to the differential attacks in the database.

The concept of differential privacy is first proposed by Dwork *et al.* [19] and defines the tradeoff between the utility of the statistical data and the privacy of participants when implementing statistical analysis in the statistical database. The differential privacy can guarantee that, even if the participant removes his data from the data set, the released results would not likely become significantly more or less. They achieve the differential privacy by adding appropriate noise which obeys some certain random distribution. Moreover, the previous work on differential privacy considers a trusted data aggregator who has capability to access to all users' data. Reference [20] proposes a distributed differentially private protocol, while it needs the interaction among the participants, which makes it impractical in the smart grid.

Some closely related works to ours are [21]–[23]. In [21], the authors propose the protocol to achieve differential privately aggregate sums on time-series data when the aggregator is untrusted. However, as [23] pointed out, the threshold Paillier cryptosystem [24] brings more expensive cost in each meter. Reference [22] proposes another technique to implement the same problem. They use a Diffe–Hellman-based applied cryptographic technique to conceal each meter's measurement and obtain the result through a brute-force search. The cost of their proposed protocol is related to the number of participants and the input range, whereas our construction is based on a more efficient construction that only uses modular additions. Reference [23] provides a more efficient technique based on modular addition of a one-time secret key. However, the key establishment is expensive because it requires the establishment of pairwise keys among smart meters. Moreover, [23] uses a different noise generation method from ours. We explore a discrete distribution to add noise, achieving strong privacy.

III. Preliminaries

In this section, we will present the problem definition by introducing the considered system model and adversary model.

A. System Model

In our system model, we mainly focus on aggregating multiple smart meter real-time consumptions by the supplier. We uniformly call the electricity producers or the grid operators who need cumulative or statistical information of usage patterns as the *aggregators*. In particular, our system is composed of one aggregator and $nusers$ (customers). Every user is equipped with an electricity smart meter, which measures the electricity consumption of the user in every T_p long period, and the smart meter sends the measurement to the aggregator at the end of every slot. In practice, high-frequency meter reading uploading is suggested, and T_p has about 15-min intervals. For notational and description convenience, we do not distinguish between the meters and the users, and we number the meters $1, 2, \ldots, n$ and also the aggregator 0. In every time period $t \in \mathbb{N}$, the readings of user i are denoted by $R_i^{(t)}$. When the context is clear, we simply write R_i instead of $R_i^{(t)}$. The supplier is interested in the aggregate statistics of all measurements in every slot.

As in [17], we assume that the smart meters have secure components which are responsible for providing a trusted computing capability for secure storage and basic cryptographic functionality. The meters should ideally act independently, without requiring interaction with other meters. We also assume that the communication channel between the aggregator and the meter is an authenticated communication channel.

B. General Attack Model

We use three progressive attack types to straightforwardly depict the possible attacks in the smart grid.

1) *External Attack*: the external attacker tries to compromise the data privacy of the users by eavesdropping the transmitted data from the user side to the aggregator side.
2) *Inside Attack*: the insider attacker (e.g., untrusted aggregator) tries to compromise the meter's privacy by accessing or even misusing the metering data of the users, which may introduce serious privacy-leaking threats.
3) *Malicious Data Mining Attack*: the attacker corrupts the aggregator and collaborates with some compromised meters to infer the users' meter readings by maliciously mining the aggregation results, e.g., comparing the aggregate results in the presence and absence of a specific user.

C. Formal Attack Model and Privacy Definition

We employ a game and the *HDA attack* to formally represent the adversary's capabilities.

1) *Game on Leaking Individual Meter's Measurements*: in this game, we use some queries and responses between the adversary and the challenger to define the adversary's capability. The success probability of the adversary is defined as the probability that the adversary successfully determines the correspondence between a ciphertext and the given true readings.

Setup. The challenger runs the system initialization algorithm to make public the resulting common parameter to the adversary $\mathcal{A}$.

Queries. In this phase, the adversary not only captures the meters' encrypted reports but also initiatively issues *Compromise* and *Encrypt* queries for some meters.

a) *Encrypt*. The adversary $\mathcal{A}$ chooses a meter i given the readings R in time slot t and asks for the ciphertext readings. We use (i, t, R) to denote meter i's readings R in time slot T. The challenger returns the ciphertext $Enc(sk_i, t, R)$ to the adversary by implementing the protocol. Moreover, only less than q_c queries for one meter i are allowed in the *Encrypt* phase.

b) *Compromise*. The adversary specifies an integer $i \in \{1, \ldots, n\}$. If $i = 0$, the challenger returns the aggregator capability sk_0 to the adversary. If $i \neq 0$, the challenger returns sk_i, the secret key for the ith meter, to $\mathcal{A}$.

Challenge. $\mathcal{A}$ issues two plaintext readings R_0 and R_1, a group of meters $i_1, i_2, \ldots, i_l$, and a time slot t^*. The challenger picks a bit $b \in \{0, 1\}$ uniformly and returns the challenger text $\mathrm{Enc}(sk_i, t^*, R_b)$. Any i must not have been compromised at the end of the game.

Guess. The adversary outputs a guess $b' \in \{0, 1\}$. $\mathcal{A}$ wins if $b = b'$.

The advantage of $\mathcal{A}$ in attacking the scheme is defined as follows:

$$Adv_{\mathcal{A}} = \left| \Pr[b = b'] - \frac{1}{2} \right|. \tag{1}$$

2) *HDA Attack*: in this attack model, the adversary maliciously infers the statistical information from the aggregate results as in Fig. 2. The adversary may collude with a set of corrupted meters to reveal the honest meters' measurements. The corrupted meters can reveal their readings, even providing false measurements to the aggregator for the adversary to statistically mine the honest meters' measurements.

We give an example to illustrate the risks of HDA attack. We assume that the adversary knows the exact aggregate results on metering set D before Alice comes back to consume electricity and successfully infers Alice's house activities by comparing two exact aggregate outputs on metering set $\{D \cup \mathrm{Alice}\}$. We design a perturbation algorithm to reduce the effects on aggregate results by modifying any single meter's readings. Our method guarantees the indistinguishable aggregate results for similar inputs (more precisely, differing by a single meter readings), and thus, the modification of any single meter's readings changes the probability of any output only up to a multiplicative factor e^{ϵ} and a constant factor δ.

In terms of the aforementioned possible attacks, we define our security definitions from two aspects.

Definition 1: Individual measurement security: if $Adv_{\mathcal{A}} < \epsilon$ is true for any polynomial time adversary $\mathcal{A}$ that makes at most q_e encrypt queries for one meter and at most q_c private key extraction queries, we say that the protocol is (q_e, q_c, ϵ) secure.

Similar to the differential privacy in database, we give the HDA-attack secure definition.

Definition 2: HDA attack security: a randomized function $\mathcal{K}$ gives δ-approximate ϵ-indistinguishability differential privacy if, for all collected readings, sets D_1 and D_2 differ on at most one element, and all $S \subseteq Range(\mathcal{K})$

$$\Pr[\mathcal{K}(D_1) \in S] \leq \exp(\epsilon) \cdot \Pr[\mathcal{K}(D_2) \in S] + \delta.$$

IV. Privacy-Preserving Metering Aggregation Scheme in Smart Grid

In this section, we present our basic protocol which allows the aggregator to make the data aggregation without leaking individual meter's measurements.

A. Basic Idea

One challenge that we face when designing the mechanism in smart grid is how to minimize the computation and communication overheads of the meters. Traditional SMPC techniques can be used to ensure that the aggregator learns only the sum [15], [16]. However, rather than the high cost of the computation, the SMPC requires the participants to interact with each other in the phase of computation, which makes the SMPC impractical in the smart grid. In contrast, in our basic solution, after a trusted system initialization phase between all meters and aggregator, no further interactions are required, except for uploading encrypted readings to the aggregator for each time slot.

We now explain the intuition behind our construction. Similar to the traditional cryptographic techniques of secret sharing, our basic idea is to permit the meter itself to randomly divide its readings into m shares. That is, meter i's readings R_i are divided into $(r_{i1}, r_{i2}, \ldots, r_{im})$ such that $R_i = r_{i1} + r_{i2} + \cdots + r_{im}$. To clearly describe our idea, we use a matrix (2) to represent the n meters' divided reading shares. Each row represents one meter's data shares. For example, R_i is represented by the ith row $(r_{i1}, r_{i2}, \ldots, r_{im})$

$$\begin{pmatrix} r_{11} & r_{12} & \cdots & r_{1m} \\ r_{21} & r_{22} & \cdots & r_{2m} \\ \cdots & \cdots & \cdots & \cdots \\ r_{n1} & r_{n2} & \cdots & r_{nm} \end{pmatrix}. \tag{2}$$

Thus, the desired aggregate is $\sum_{i=1}^{n} R_i = \sum_{i=1}^{n}\sum_{j=1}^{m} r_{ij} = \sum_{j=1}^{m}\sum_{i=1}^{n} r_{ij}$. We exploit the interactive masking among the meters to protect the individual meter's readings. An aggregator only can reveal the sum of shares $\sum_{i=1}^{n} r_{ij}$ for each column in matrix (2). Shares are added together to conceal the individual ones. In this paper, we will propose a novel mechanism in which the sum of shares in the same column $\sum_{i=1}^{n} r_{ij}$, $j = 1, 2, \ldots, m$, is a unique solution of the linear equations.

B. Details of the Basic Protocol

Our basic protocol mainly includes three parts: system initialization, meter's reading report, and privacy-preserving aggregation.

1) System Initialization: Let H be a hash function $H: \{0,1\}^* \to \mathbb{Z}_p$ and $\alpha_1, \alpha_2, \ldots, \alpha_m$ be the possible seeds for H. The later computations all are performed in $\mathbb{Z}_p$. Suppose each meter's readings are in the range $\{0, 1, \ldots, \triangle\}$. Then, the sum of n meter readings is in the range $\{0, 1, \ldots, n\triangle\}$. We claim $p > n\triangle$. The public parameters in the system are

$$params = (H, p, \alpha_1, \ldots, \alpha_m).$$

For n meters, the trusted server generates $m\cdot(n+1)$ random numbers $(s_{01}, s_{02}, \ldots, s_{0m}; s_{11}, s_{12}, \ldots, s_{1m}; \ldots; s_{n1}, s_{n2}, \ldots, s_{nm}) \in \mathbb{Z}_p$ such that $\sum_{i=0}^{n} s_{ij} = 0$ for $j = 1, 2, \ldots, m$. Here, m is the security parameter, which refers to the number of shares in the protocol. The aggregator obtains $sk_0 = (s_{01}, s_{02}, \ldots, s_{0m})$, and smart meter i is distributed with its corresponding secret keys $sk_i = (s_{i1}, s_{i2}, \ldots, s_{im})$.

2) Meter's Reading Report: In order to conceal the meter readings R_i in the time slot t, smart meter i performs the following steps.

Step 1): meter i divides its readings R_i into m shares

$$R_i = r_{i1} + r_{i2} + r_{i3} + \cdots + r_{im} \bmod p. \tag{3}$$

Moreover, meter i computes the time-series information for each time slot t by performing the following:

$$(H(\alpha_1 \| t), H(\alpha_2 \| t), \ldots, H(\alpha_m \| t)).$$

We use $(x_1, x_2, \ldots, x_m)$ to denote these values, i.e.,

$$(x_1, x_2, \ldots, x_m) = (H(\alpha_1 \| t), H(\alpha_2 \| t), \ldots, H(\alpha_m \| t)). \tag{4}$$

Step 2): using the private key sk_i, meter i encrypts each shares into $(y_{i1}, y_{i2}, \ldots, y_{im})$ satisfying

$$\begin{cases} y_{i1} = r_{i1} + r_{i2}x_1 + \cdots + r_{im}x_1^{m-1} + s_{i1}x_1^m \bmod p \\ y_{i2} = r_{i1} + r_{i2}x_2 + \cdots + r_{im}x_2^{m-1} + s_{i2}x_2^m \bmod p \\ \cdots \quad \cdots \quad \cdots \\ y_{im} = r_{i1} + r_{i2}x_m + \cdots + r_{im}x_m^{m-1} + s_{im}x_m^m \bmod p \end{cases} \tag{5}$$

3) Privacy-Preserving Aggregation:

Step 1): to aggregate the time slot t's readings, the aggregator also computes the time-series information $(x_1, x_2, \ldots, x_m)$ as the aforementioned meter i's operations.

Step 2): using the private key sk_0, $(x_1, x_2, \ldots, x_m)$, and the collected data $(y_{i1}, y_{i2}, \ldots, y_{im})$ for all meter $i (i = 1, 2, \ldots, n)$, the aggregator constructs the linear equations with the $(A_1, A_2, \ldots, A_m)$ unknown variables

$$\begin{cases} \sum_{i=1}^{n} y_{i1} + s_{01}x_1^m = A_1 + A_2x_1 + \cdots + A_mx_1^{m-1} \bmod p \\ \sum_{i=1}^{n} y_{i2} + s_{02}x_2^m = A_1 + A_2x_2 + \cdots + A_mx_2^{m-1} \bmod p \\ \cdots \quad \cdots \quad \cdots \\ \sum_{i=1}^{n} y_{im} + s_{0m}x_m^m = A_1 + A_2x_m + \cdots + A_mx_m^{m-1} \bmod p \end{cases} \tag{6}$$

Step 3): the aggregator computes the solutions of the aforementioned linear equation set and outputs the aggregated readings by adding the solutions together

$$Result = A_1 + A_2 + \cdots + A_m. \tag{7}$$

C. Correctness Analysis

The data aggregator can correctly get the sum of the meters' measurements if each user honestly follows the protocol. For jth equation in the linear equations, we have

$$\begin{aligned} &\sum_{i=1}^{n} y_{ij} + s_{0j}x_j^m \\ &\quad = \sum_{i=1}^{n} r_{i1} + \sum_{i=1}^{n} r_{i2}x_j + \cdots + \sum_{i=1}^{n} r_{im}x_j^{m-1} \\ &\qquad + \sum_{i=1}^{n} s_{ij}x_j^m + s_{0j}x_j^m \\ &\quad = \sum_{i=1}^{n} r_{i1} + \sum_{i=1}^{n} r_{i2}x_j + \cdots + \sum_{i=1}^{n} r_{im}x_j^{m-1} \\ &\qquad \times \left(\because \sum_{i=0}^{n} s_{ij} = 0 \right). \end{aligned}$$

Since $x_1, x_2, \ldots, x_m$ are different values, the aggregator obtains the unique solution $A_1 = \sum_{i=1}^{n} r_{i1}, \ldots, A_m = \sum_{i=1}^{n} r_{i1m}$ with m equations and m unknown variables. The sum of these equation solutions is the desired result $Result = A_1 + A_2 + \cdots + A_m = \sum_{i=1}^{n}\sum_{j=1}^{m} r_{ij}$. Therefore, the aggregator could obtain the aggregation results of meter readings.

D. Security Analysis

Theorem 1: Our basic protocol is secure under the previously defined formal attack model, and the adversary's success probability against the basic protocol is at most $1/p^m$ if there are at most $m - 1$ queries for one meter in encrypt query phase.

We present the proof of Theorem 1 in Appendix A.

V. Advanced Protocol Resisting HDA Attack

The basic scheme presented in Section IV ensures that the aggregator learns nothing about the individual value of the metering but the overall aggregation result. However, the basic scheme is still vulnerable to the HDA attack. The dishonest meter may even collude with the aggregator to provide some

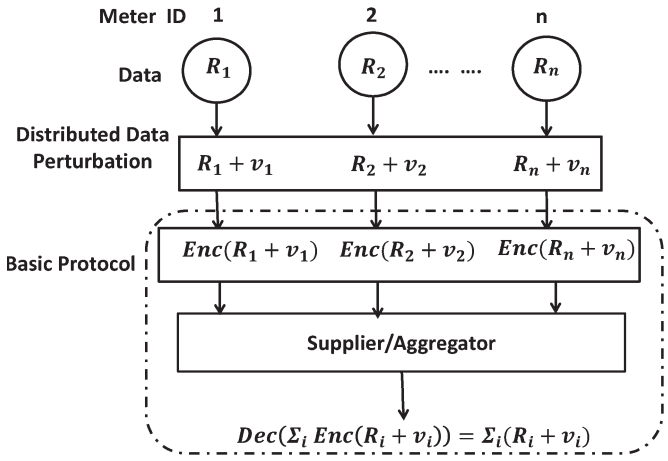


Fig. 3. Our advanced protocol to resist the HDA attack. In every time slot, each meter independently adds noise v_i to its measurements before encrypting. The following encryption is executed on the perturbed data as in the basic protocol. The aggregator uses capability sk_0 to decrypt the noisy sum but learns nothing more.

false information to manipulate the collected data. In this section, we will propose a novel approach to thwart HDA attack by introducing some randomness (or introducing some noise) without interfering the aggregation results.

To realize the HDA attack security, we use the idea of differential privacy in the database. Differential private algorithm in the database produces indistinguishable outputs for similar inputs (differing by a single entry), and thus, the modification of any single meter's measurements changes the probability of any output only up to a multiplicative factor e^{ϵ} and a small nonzero shift δ. The parameters ϵ and δ allow us to control the level of privacy. Moreover, the common methods for achieving differential privacy let the aggregator add a random noise to perturb the outputs of f, where the noise distribution is carefully calibrated to the global sensitivity of f.

However, in our case, the aggregator is untrusted. The perturbation does not rely on any trusted aggregator. Therefore, the noise should be added by each meter itself on its own measurements and should be encrypted in such a way that the aggregator can only compute the (noisy) aggregate. In particular, each meter i should add noise to its measurements before encrypting them (Fig. 3).

A. Differential Privacy Preliminaries

Traditionally, the differential privacy adds noise sampled from a Laplace distribution to the published statistics to ensure individual privacy. However, because the basic protocol is based on finite fields $\mathbb{Z}_p$, we need to choose a discrete distribution instead. A fact in probability states that the Laplace distribution is divisible and can be constructed as the sum of i.i.d. gamma distributions. We use binomial distribution, which can be regarded as an approximation for the gamma distribution. The use of binomial distribution first appeared in Dwork *et al.* [20]. We now provide some backgrounds on the binomial distribution.

Definition 3: Unbiased binomial distribution: let the bias of the binomial distribution be $p = 1/2$. Then, the mass function at $n/2 + x$ is $\binom{n}{2/n+x}(1/2)^n$. We use $B(n, p)$ to denote the binomial distribution with parameters n and p. The unbiased binomial distribution is denoted as $B(n, 1/2)$.

Fact 1: Let $\tilde{V}_1, \ldots, \tilde{V}_m$ be i.i.d., and $\tilde{V}_i \sim B(n_i, p)$, $i = 1, 2, \ldots, m$. Then, $\tilde{V} = \tilde{V}_1 + \cdots + \tilde{V}_m \sim B(n, p)$, in which $n = n_1 + \cdots + n_m$.

Theorem 2: Let $\epsilon > 0$ and $\delta > 0$. Suppose u and v are two integers such that $|u - v| \leq \triangle$. Let r be a random variable having the *unbiased binomial distribution*$B(n, 1/2)$. Then, for any integer k, $\Pr[u + r = k] \leq exp(\epsilon) \cdot \Pr[v + r = k] + \delta$, as long as parameter n is chosen to be at least $\hbar$, $\hbar = 64\triangle^2 log(2/\delta)/\epsilon^2$.

B. Distributed Data Perturbation: Achieving HDA Privacy

In our data perturbation phase, we exploit the classical assumption in the Byzantine literature that at least 2/3 of the participants will survive. With the $2N/3$ honest smart meters cooperatively generating noise, we ensure that the deviation between the released data and the true results is small and provides the differential privacy for the HDA attack. We present our protocol as follows. Assume that each meter's readings are in the range of $\{1, 2, \ldots, \triangle\}$.

Our distributed sanitization algorithm is simple; meter i calculates the value $\hat{R}_i$ for every reading R_i in time slot t such that $\hat{R}_i = R_i + v_i \mod p$ and sends it to the aggregator, where $v_i \sim B(3\hbar/2N, 1/2)$ and $\hbar = 64\triangle^2 log(2/\delta)/\epsilon^2$. Now, if the aggregator sums up all values received from n meters, then $\sum_{i=1}^{n} \hat{R}_i = \sum_{i=1}^{n} R_i + \sum_{i=1}^{n} v_i$ is differential private based on Theorem 2. Since the mean of the unbiased binomial distribution $B(n, 1/2)$ is $n/2$, the releasing output is $\sum_{i=1}^{n} \hat{R}_i - n/2$.

C. Utility Analysis

In order to measure the utility of our perturbation, we quantify the difference between the noisy statistic $\mathrm{sum}(\hat{R})$ and the true output $\mathrm{sum}(R)$. A common error measure is the mean absolute error, which is the expectation $\mathbb{E}|\mathrm{sum}(\hat{R}) - \mathrm{sum}(R)|$. However, if the true output $\mathrm{sum}(R)$ is greater, the added noise becomes small compared to $\mathrm{sum}(R)$, which intuitively produces better utility. For better description of the added noise compared to the output result, we define the relative error $\gamma = |\mathrm{sum}(\hat{R}) - \mathrm{sum}(R)|/\mathrm{sum}(R) + 1$, and thus, the error function is defined as $\mathbb{E}(\gamma)$. Therefore, the utility of our distributed perturbation algorithm is defined as $\mathbb{E}(\gamma) = (1/\mathrm{sum}(R) + 1)\mathbb{E}|\mathrm{sum}(R) - \mathrm{sum}(R) + \sum_{i=1}^{n} v_i| = p/\mathrm{sum}(R) + 1$.

Fig. 4 shows the error with different numbers of users. In the simulation, we assume that each meter's readings are in the range $\{0, 1, \ldots, 5\}$. We vary the number of users from 3000 to 15 000 and compare the utility of our protocol under the privacy parameter ($\epsilon = 0.5$ and $\delta = 0.01$). Simulation shows that our advanced protocol produces little error with different numbers of users n.

VI. Performance Evaluation

In this section, we present the detailed implementation and evaluation of the proposed protocol with different security

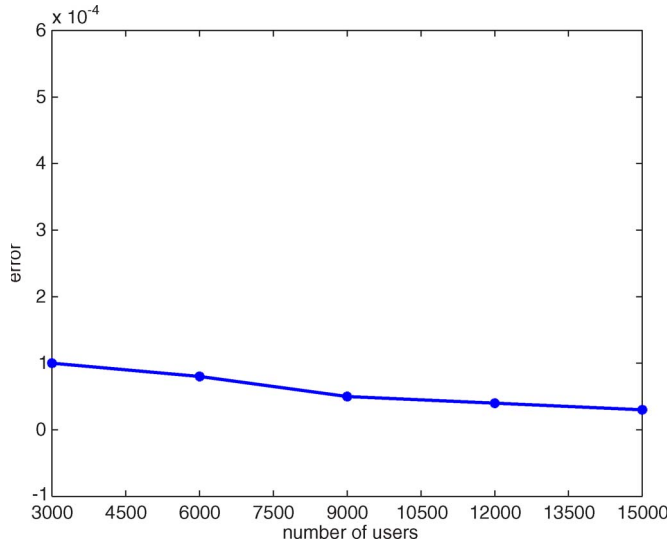


Fig. 4. Utility analysis of the advanced protocol.

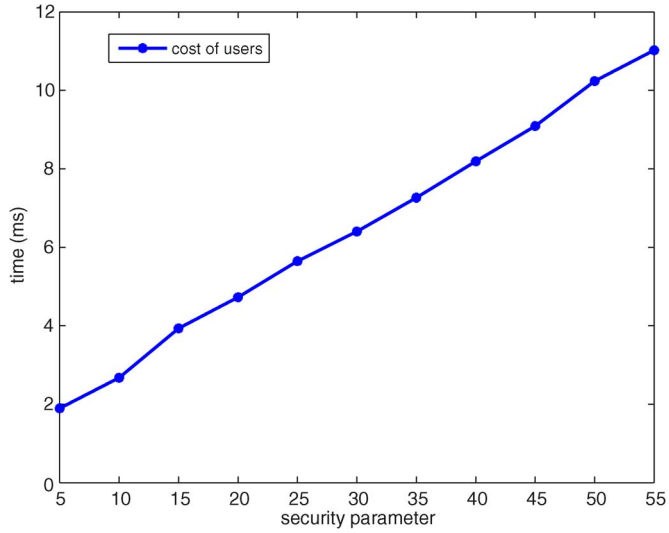


Fig. 5. Computation cost of each user.

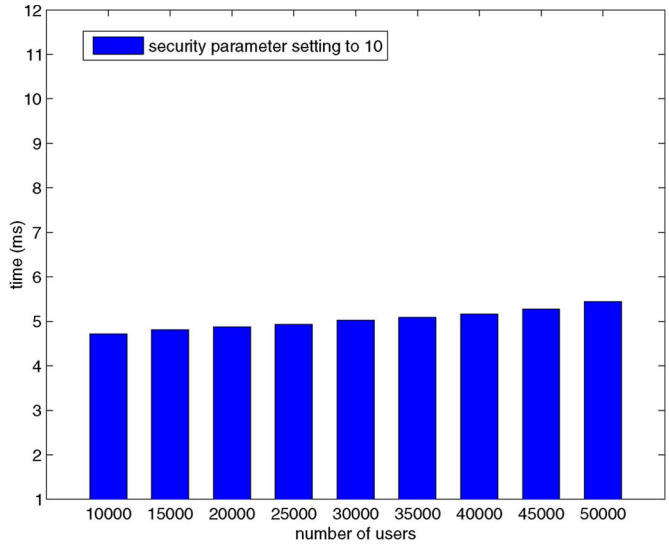


Fig. 6. Computation cost of the aggregator with a security parameter setting of 10.

parameters and different numbers of users (smart meters). The overheads of the keys and hash seed generation and distribution by the trusted server in the system initialization phase are not evaluated because the system initialization occurs rarely only when adding or removing meter in the grid.

A. Computation Cost Analysis

In the experiment, we implement the proposed scheme in Java and run it in a computer with 2.26 GHz, dual-core CPU and 2-G memory, and Windows XP OS. We use SHA1 as the hash function to generate the time-series information $(H(\alpha_1\|t), H(\alpha_2\|t), \ldots, H(\alpha_m\|t))$ in the implement. As we have proved in the section of security analysis of the basic protocol, the security of our scheme is related with the value of m (we call m as security parameter). The cost of each meter is mainly on perturbing the real readings R, dividing the perturbed readings into m shares r_i randomly, calculating the public time-series data $(x_1, x_2, \ldots, x_m)$, and outputting m values y_i with its private keys. Because the calculation does not include complex and expensive computation except for hash function, the computation cost of the meter is low. In our experiment, it takes about 5 ms when the security parameter m is 20. The experiment shows that the computation cost of the meter is linearly increased with the increase of the security parameter m (see Fig. 5) but is kept always in a low level. Moreover, the smart grid does not need to store any long public key like some other schemes. For example, in the scheme [17], each smart grid has to store N public keys. Here, N is the number of smart grids in one local substation, and the number of N usually is a few hundred. Therefore, our scheme can be implemented easily on the cheap Java-enable smart card.

Fig. 6 shows that the computation cost of the aggregator is increased slowly with the increase of the number of users when the security parameter is set to 10. Figs. 7 and 8 show that, when the security parameter is 20 and 30, respectively, the computation cost of the aggregator has a similar trend with the one where the security parameter is 10. Therefore, we can say that the number of user has little effect on the cost of the aggregator. For ten thousand users, the computation cost is only a few millisecond when the security parameter is 20 in our experiment. This result shows that our scheme has little request on the computation performance of the aggregator and is suitable for the real word that the grid has a lot of meters.

Fig. 9 shows the relation between the computation cost and the security parameter with the different numbers of meters. From Fig. 9, we can find that the computation cost of the aggregator is linearly increased with the increase of the security parameter. The grid manager can estimate the computation cost easily with this feature.

B. Communication Overhead Analysis

The communications between the smart meters and the aggregator are bidirectional in the smart grid. It includes the aggregator-to-meter communications and meter-to-aggregator communications. During the system initialization phase, the

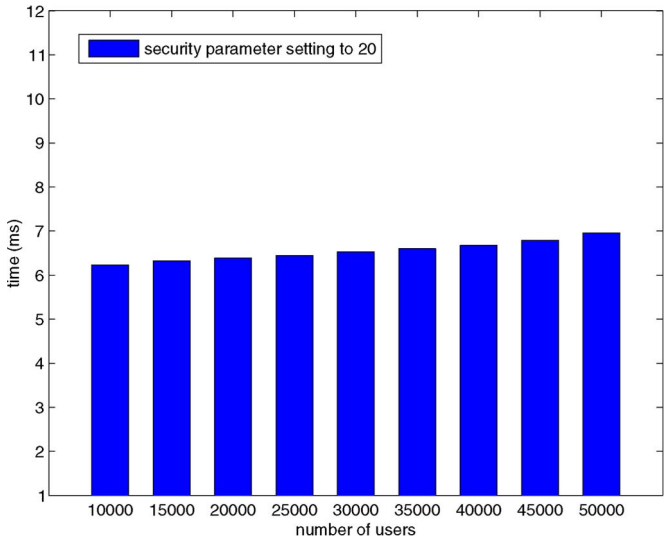


Fig. 7. Computation cost of the aggregator with a security parameter setting of 20.

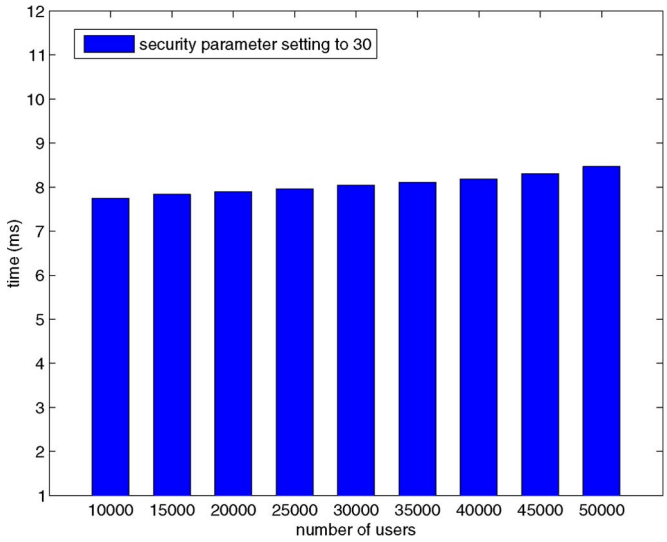


Fig. 8. Computation cost of the aggregator with a security parameter setting of 30.

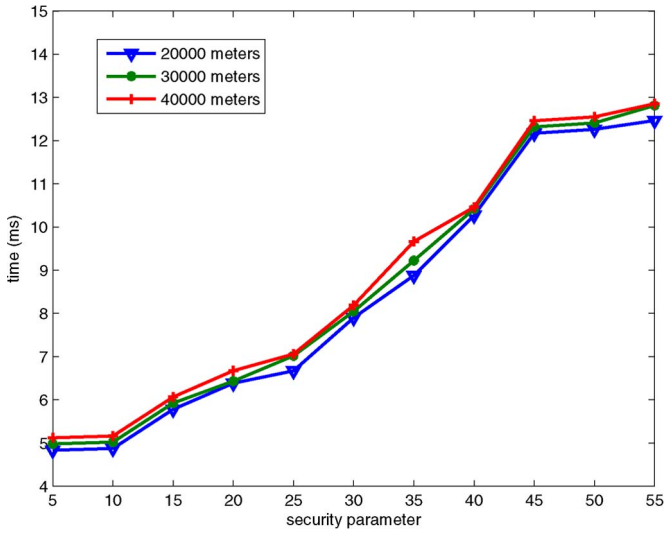


Fig. 9. Computation cost of the aggregator with different security parameters.

trusted server will send the secret keys to the user. After that phase, the trust server and aggregator do not need to send additional information to the meters. In one reading time slot, each user calculates m results y_i and sends them to the aggregator with the format $(y_1 \| y_2 \| \cdots \| y_m \| t)$. Suppose each y_i is an integer with 32 b. Therefore, the total length of the uploaded result is $32 \cdot (m+1)$. If the secure parameter m is chosen to be less than 100, the length of the communicated message is only a few thousand bits, which is very shorter. Therefore, the overhead of communication in our protocol is very low and very suitable for the limited bandwidth of the smart meter.

From the aforementioned analysis, the proposed scheme is indeed efficient in terms of computation and communication cost, which is suitable for the real-time high-frequency data collection in the smart grid.

VII. Conclusion

In this paper, we have identified and formulated a new security threat toward privacy-preserving data aggregation in smart grid. To thwart this attack, we have proposed a novel privacy-preserving data aggregation scheme which could support efficient data aggregation for time-series metering data without leaking the individual value. It could also thwart HDA attack by introducing some randomness to the aggregation result without affecting the aggregation utility. We have given a formal proof to the security of the proposed scheme and a detailed performance evaluation of the proposed scheme by the Java implementation. The extensive analysis and experimental results demonstrate the effectiveness, efficiency, and security of the proposed scheme. Our future research will focus on the other security aspects of the smart grid.

Appendix A
Basic Individual-Measurement-Secure Protocol Security Analysis

In this section, we present the formal security proof of Theorem 1.

Setup: The challenger builds the system in the initialization phase and picks random seeds $\alpha_1, \alpha_2, \ldots, \alpha_m$, hash function H, and random $(s_{01}, s_{02}, \ldots, s_{0m}; s_{11}, s_{12}, \ldots, s_{1m}; \ldots; s_{n1}, s_{n2}, \ldots, s_{nm}) \in \mathbb{Z}_p$ such that $\sum_{i=0}^{n} s_{ij} = 0$ for $j = 1, 2, \ldots, m$. The challenger gives the adversary the params $(H, \alpha_1, \alpha_2, \ldots, \alpha_m, p)$.

Queries:

1) *Compromise*. The adversary $\mathcal{A}$ can make *Compromise* queries adaptively and can ask for the compromised meter i's secret key $sk_i = (s_{i1}, s_{i2}, \ldots, s_{im})$. The challenger returns $(s_{i1}, s_{i2}, \ldots, s_{im})$ to the adversary when asked. Moreover, if the adversary asks the value sk_0, then the aggregator capability is compromised.
2) *Encrypt*. The adversary $\mathcal{A}$ submits an *Encrypt* query to the challenger. We use (i, t, R_i) to denote the meter i's readings R_i in the time slot t. The challenger randomly divides $R_i = R_{i1} + R_{i2} + \cdots + R_{im}$ and computes $y_{i1}, y_{i2}, \ldots, y_{im}$ as the protocol. $\mathcal{A}$ obtains $(y_{i1}, y_{i2}, \ldots, y_{im})$. We require that, for meter i, the *Encrypt* queries (i, t, R_i) are at most $m - 1$.

Challenge: The adversary $\mathcal{A}$ specifies a group of meters V and a time slot t^*. For each $i \in V$, the adversary chooses (i, R_{i0}, t^*) or (i, R_{i1}, t^*) to the challenger. The challenger picks a bit $b \in \{0, 1\}$ uniformly and returns the challenger text $(y_{ib_1}, y_{ib_2}, \ldots, y_{ib_m}) = \mathrm{Enc}(\{s_{ij_{1\le j\le m}}\}, t^*, R_{ib})$. Because the challenger randomly divides R_{ib}, the adversary has no other choice but to solve the linear equations with given $(y_{ib_1}, y_{ib_2}, \ldots, y_{ib_m})$. Moreover, by launching *Compromise* and *Encrypt* queries, the adversary can get some additional information on solving the linear equations. However, we analyze that the additional information is not sufficient in solving the equation set.

For ease of discussing, we make a small modification of the game. In the *Encrypt* queries, if the adversary issues a request for some tuples (i, x, t^*), where t^* is the time step specified in the challenge phase, the challenger treats this as a *Compromise* query and simply returns sk_i to the adversary. From now on, we will assume that the adversary does not make any *Encrypt* queries for time t^*. We use W to denote the *Compromise* secret key set and U to denote the *Encrypt* meter's ID. Let $\bar{W} \triangleq [n] \setminus W$ denote the set of uncompromised meters.

We consider three cases for the attack target V.

Case 1. $sk_0 \in W$ (i.e., the aggregator capability has been compromised), and $V = \bar{W}$. We assume the number of W as w. The adversary can construct $(n - w) \cdot m + 1$ linear equations with $(n - k) \cdot (m + m)$ unknowns. This means that the adversary cannot determine the dividing shares or the share sum for every $i \in V$.

Case 2. $V \neq \bar{W}$, or the aggregator capability has not been compromised. Compared to case 1, the adversary has much less equation number with the same number unknowns. Therefore, the adversary cannot determine the dividing shares or the share sum for every $i \in V$.

Case 3. For some meter i, the adversary $\mathcal{A}$ queries the *Encrypt* at most q_e times for time slot $t \neq t^*$. In this case, the adversary get $m \cdot q_e + q_e$ linear equations with $m \cdot q_e + m$ unknowns. If $q_e < m$, then the equation set has undetermined solutions.

Therefore, the success probability of the adversary is at most $1/p^m$.

APPENDIX B
PROOF OF THEOREM 2

This section presents the proof of Theorem 2.

a) Proof: We consider the extreme case when $|u - v| = \triangle$. Then

$$\frac{\Pr[u + r = k]}{\Pr[v + r = k]} = \frac{\Pr[r = k - u]}{\Pr[r = k - u + \triangle]}.$$

Use $n/2 + x$ to denote $k - u$. We determine the range of x which makes

$$\frac{\Pr[r = n/2 + x]}{\Pr[r = n/2 + x + \triangle]} = \frac{\binom{n}{n/2+x}}{\binom{n}{n/2+x+\triangle}} < e^{\epsilon}. \tag{8}$$

By computing the aforementioned inequality, we say that, as long as $x < n\epsilon/8\triangle$, the inequality holds.

In the case $x > n\epsilon/8\triangle$, we let δ limit the relative shift. That is, $\Pr[y > n/2 + \epsilon n/8\triangle] \le \delta/2$. According to the Chernoff bound, we get $\Pr[y > n/2 + \epsilon n/8\triangle] \le \exp(-\epsilon^2 n/64\triangle^2)$. Thus, we get δ-approximate ϵ-indistinguishability as long as n is chosen to be at least $64\triangle^2 \log(2/\delta)/\epsilon^2$. ■

REFERENCES

[1] R. Lu, X. Liang, X. Li, X. Lin, and X. Shen, "EPPA: An efficient and privacy-preserving aggregation scheme for secure smart grid communications," *IEEE Trans. Parallel Distrib. Syst.*, vol. 23, no. 9, pp. 1621–1631, Sep. 2012.

[2] *Directive 2009/72/ec*, European Parliament, Brussels, Belgium, 2009.

[3] R. Anderson and S. Fuloria, "Who controls the off switch?" in *Proc. IEEE SmartGridComm*, 2010, pp. 96–101.

[4] H. Lam, G. Fung, and W. Lee, "A novel method to construct taxonomy electrical appliances based on load signatures," *IEEE Trans. Consum. Electron.*, vol. 53, no. 2, pp. 653–660, May 2007.

[5] A. Molina-Markham, P. Shenoy, K. Fu, E. Cecchet, and D. Irwin, "Private memoirs of a smart meter," in *Proc. 2nd ACM Workshop Embedded Sens. Syst. Energy-Efficiency Build.*, pp. 61–66.

[6] C. Cuipers and B. Koops, "Het wetsvoorstel slimme meters: Een privacytoets op basis van art. 8 evrm," Universiteit van Tilburg, Tilburg, The Netherlands, 2008.

[7] C. Efthymiou and G. Kalogridis, "Smart grid privacy via anonymization of smart metering data," in *Proc. 1st IEEE Int. Conf. SmartGridComm*, 2010, pp. 238–243.

[8] Y. Z. H. Liu, H. Ning, and L. Yang, "Aggregated-proofs based privacy-preserving authentication for V2G networks in the smart grid," *IEEE Trans. Smart Grid*, vol. 3, no. 4, pp. 1722–1733, Dec. 2012.

[9] J. Li, Z. Li, K. Ren, and X. Liu, "Towards optimal electric demand management for internet data centers," *IEEE Trans. Smart Grid*, vol. 3, no. 1, pp. 183–192, Mar. 2010.

[10] Q. Li and G. Cao, "Multicast authentication in the smart grid with one-time signature," *IEEE Trans. Smart Grid*, vol. 2, no. 4, pp. 686–696, Dec. 2011.

[11] D. He, C. Chen, J. Bu, S. Chan, Y. Zhang, and M. Guizani, "Secure service provision in smart grid communications," *IEEE Commun. Mag.*, vol. 50, no. 8, pp. 53–61, Aug. 2012.

[12] C. Gentry, "Fully homomorphic encryption using ideal lattices," in *Proc. 41st Annu. ACM Symp. Theor. Comput.*, 2009, pp. 169–178.

[13] Z. Zhong and R. Wright, "Privacy-preserving classification of customer data without loss of accuracy," in *Proc. SIAM Int. Conf. SDM*, Newport Beach, CA, USA, 2005, pp. 603–610.

[14] E. Magkos, M. Maragoudakis, V. Chrissikopoulos, and S. Gritzalis, "Accurate and large-scale privacy-preserving data mining using the election paradigm," *Data Knowl. Eng.*, vol. 68, no. 11, pp. 1224–1236, Nov. 2009.

[15] O. Goldreich, "Secure multi-party computation," 1998. (Available at http://citeseerx.ist.psu.edu).

[16] W. Du and M. Atallah, "Secure multi-party computation problems and their applications: A review and open problems," in *Proc. Workshop New Sec. Paradigms*, 2001, pp. 13–22.

[17] F. Garcia and B. Jacobs, "Privacy-friendly energy-metering via homomorphic encryption," in *Proc. Sec. Trust Manage.*, 2011, pp. 226–238.

[18] K. Kursawe, G. Danezis, and M. Kohlweiss, "Privacy-friendly aggregation for the smart-grid," in *Privacy Enhancing Technologies*. New York, NY, USA: Springer-Verlag, 2011, pp. 175–191.

[19] C. Dwork, "An ad omnia approach to defining and achieving private data analysis," in *Proc. 1st ACM SIGKDD Int. Conf. Privacy, Sec., Trust KDD*, 2007, pp. 1–13.

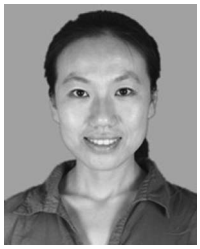
[20] C. Dwork, K. Kenthapadi, F. McSherry, I. Mironov, and M. Naor, "Our data, ourselves: Privacy via distributed noise generation," in *Proc. Adv. Cryptology EUROCRYPT*, 2006, pp. 486–503.

[21] V. Rastogi and S. Nath, "Differentially private aggregation of distributed time-series with transformation and encryption," in *Proc. 2010 Int. Conf. Manage. Data*, 2010, pp. 735–746.

[22] E. Shi, T. Chan, E. Rieffel, R. Chow, and D. Song, "Privacy-preserving aggregation of time-series data," in *Proc. NDSS*, 2011, vol. 17, pp. 1–17.

[23] G. Acs and C. Castelluccia, "I have a dream! (differentially private smart metering)," in *Information Hiding*. Berlin, Germany: Springer-Verlag, 2011, pp. 118–132.

[24] P.-A. Fouque, G. Poupard, and J. Stern, "Sharing decryption in the context of voting or lotteries," in *Proc. Financial Cryptography*, 2000, pp. 90–104.



Weiwei Jia received the B.Sc. and M.Sc. degrees in mathematics from Shanxi Normal University, Xi'an, China, in 2002 and 2005, respectively. She is currently working toward the Ph.D. degree at Shanghai Jiao Tong University, Shanghai, China.

She was a teacher with the College of Sciences, Hohai University, Nanjing, China. Her research interests include applied cryptography and network security, particularly security on cloud computing and smart grid.

Haojin Zhu (M'09) received the B.Sc. degree in computer science from Wuhan University, Wuhan, China, in 2002, the M.Sc. degree in computer science from Shanghai Jiao Tong University (SJTU), Shanghai, China, in 2005, and the Ph.D. degree in electrical and computer engineering from the University of Waterloo, Waterloo, ON, Canada, in 2009.

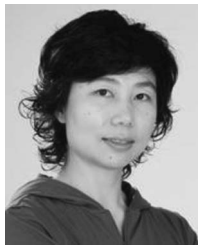
He is currently an Associate Professor with the Department of Computer Science and Engineering, SJTU. His current research interests include wireless network security and distributed system security.

Dr. Zhu was a corecipient of the best paper awards of the IEEE ICC 2007—Computer and Communications Security Symposium and Chinacom 2008—Wireless Communication Symposium. He served as a Guest Editor of IEEE NETWORKS and Associate Editor of *KSII Transactions on Internet and Information Systems* and *Ad Hoc & Sensor Wireless Networks*. He currently serves in the Technical Program Committee of international conferences such as IEEE International Conference on Computer Communications (INFOCOM), IEEE Global Telecommunications (GLOBECOM), IEEE International Conference on Communications (ICC), IEEE Wireless Communications and Networking Conference (WCNC), etc.

Zhenfu Cao (SM'10) received the B.Sc. degree in computer science and technology and the Ph.D. degree in mathematics from Harbin Institute of Technology, Harbin, China, in 1983 and 1999, respectively.

He is currently a Distinguished Professor and the Director of the Trusted Digital Technology Laboratory, Shanghai Jiao Tong University, Shanghai, China, where he was exceptionally promoted as Associate Professor in 1987 and became a Professor in 1991. He is an Associate Editor of *Computers and Security* (Elsevier), a member of the Editorial Board of *Fundamenta Informaticae* (IOS) and *Peer-to-Peer Networking and Applications* (Springer-Verlag), a Guest Editor of the *Special Issue: Wireless Network Security* of *Wireless Communications and Mobile Computing* (Wiley), etc.

Dr. Cao is a member of the expert panel of the National Nature Science Fund of China. He is actively involved in the academic community, serving as committee/session chair and program committee member of several international conference committees, which are as follows: the IEEE Global Communications Conference (since 2008), the IEEE International Conference on Communications (since 2008), the International Conference on Communications and Networking in China (since 2007), etc. He has received a number of awards, including the Youth Research Fund Award of the Chinese Academy of Sciences in 1986, the Ying-Tung Fok Young Teacher Award in 1989, the National Outstanding Youth Fund of China in 2002, the Special Allowance by the State Council in 2005, and a corecipient of the 2007 IEEE International Conference on Communications (Computer and Communications Security Symposium) Best Paper Award in 2007. He also received seven awards granted by the National Ministry and governments of provinces such as the first prize of the Natural Science Award from the Ministry of Education.



Xiaolei Dong received the Ph.D. degree in mathematics from Harbin Institute of Technology, Harbin, China, in 2001.

She was a Postdoctoral Researcher with Shanghai Jiao Tong University (SJTU), Shanghai, China, from September 2001 to July 2003. She then joined the Department of Computer Science and Engineering, SJTU, where she is currently a Professor. She has published more than 50 academic papers. Her primary research interests include number theory, cryptography, trusted computing, etc.

Dr. Dong is an Associate Editor of *Security and Communication Networks* (John Wiley). Her project on "Number Theory and Modern Cryptographic Algorithms" won the first prize of the Science and Technology Progress Award in Universities of China in 2002. Her project on "New Theory of Cryptography and Other fundamental Problems" won the second prize of Shanghai Natural Science Award in 2007. Her project on "Formalized Security Theories on Complex Cryptosystems and Their Applications" won the second prize of the Natural Science Award of the Ministry of Education in 2008. She won the first prize of the "Outstanding Teachers on the School Award" of SJTU in 2009.

Chengxin Xiao received the B.Sc. degree from Shanghai Jiao Tong University, Shanghai, China, in 2010, where he is currently working toward the M.Sc. degree.

His research interests include mobile social network, delay tolerant networks (DTN), and cloud computing.