

A Collusion-Resistance Privacy-Preserving Smart Metering Protocol for Operational Utility

Farid Zaredar, and Morteza Amini

Abstract—Modern grids have adopted advanced metering infrastructure (AMI) technology to facilitate bidirectional communication between smart meters and control centers. This enables smart meters to report their fine-grained consumption values at predefined intervals to utility providers for various purposes, including demand balancing, load forecasting, dynamic billing, and operational efficiency. Compared to traditional power grids, smart grids offer advantages such as enhanced reliability, improved energy efficiency, and increased security. However, utility providers can compromise user privacy by analyzing fine-grained readings and extracting individuals' daily activities from this invaluable time-series data. To address this privacy concern, we propose a collusion-resistance, privacy-preserving aggregation protocol for smart metering in operational services. Our protocol ensures privacy by leveraging strong privacy-enhancing techniques such as partially additive homomorphic encryptions, aggregation, data perturbation, and data minimization. The proposed scheme aggregates perturbed consumption readings by using the additive homomorphic property of the Paillier cryptosystem to provide the aggregated results for multiple operational purposes. We evaluate the proposed protocol in terms of both performance and privacy. The scheme's computational, memory, and communication overhead were examined. The total protocol execution time with 1024-bit key size is approximately 2.21 seconds. Furthermore, we evaluated the protocol's privacy through normalized conditional entropy (NCE) metric. Higher NCE values, closer to 1, indicate stronger privacy. We show that by increasing the noise scale, the NCE value rises. This represents perturbed value retains minimal information about the original value, thereby reducing privacy risks. Overall, evaluation results demonstrate the protocol's efficiency while employing various privacy-preserving techniques.

Index Terms—Smart Grid, Smart Meter, Privacy, Homomorphic Encryption, Data Perturbation

I. INTRODUCTION

THE digital evolution of energy infrastructure has introduced numerous advantages, including real-time monitoring and grid management, demand-supply balancing, self-healing, efficient energy generation and transmission, improved load forecasting, and rapid outage detection. Smart grids provide greater efficiency, strong reliability and sustainability, and better flexibility in electricity management compared to the traditional electrical grids, [1], [2]. Due to the inability of traditional power grids to meet electricity demand in the 21st century [1], many countries have adopted smart grids as their modernized power infrastructure. According to the U.S. Energy Information Administration (EIA), global energy consumption is projected to rise by 48% between

F. Zaredar is with the Data and Network Security Laboratory (DNSL), Department of Computer Engineering, Sharif University of Technology, Tehran, Iran (e-mail: farid.zaredar78@sharif.edu).

M. Amini is with Department of Computer Engineering, Sharif University of Technology, Tehran, Iran (e-mail: amini@sharif.edu).

2012 to 2040 [3]. Additionally, the seamless integration of smart grids with renewable energy resources reduces cost and energy waste [4]. This integration also contributes to lowering the carbon emissions [5], thereby mitigating air pollution. In Mexico and Central America, the primary objective is to generate electricity as much electricity as possible from renewable energy resources (e.g., solar panels and wind turbines) [6]. More specifically, Mexico and Central America aim to produce 50% of their electricity from renewable energy resources by 2030 [7].

Advanced metering infrastructure (AMI) enables a bidirectional flow of energy and information between smart meters and the metering data management system (MDMS) provided by the energy supplier [8]. The AMI is superior to the previous technology, automated meter reading (AMR), which only supports unidirectional communication [9]. The AMI consists of three essential components: (1) smart meters, (2) communication networks, and (3) metering data management systems [9]. Smart meters report fine-grained consumption values at high frequency (i.e., every 15 minutes) to the utility provider. Communication technologies such as power-line communication (PLC) and radio access networks (RANs) facilitate the communication between intelligent meters and the MDMS. The MDMS collects, processes, and analyzes these fine-grained readings for various purposes such as billing, operational utilities and, different types of value-added services.

Although the collection of fine-grained consumption values is useful for grid management and monitoring, it also raises significant privacy concerns. For instance, with such data, the utility provider can infer a customer's daily life patterns (e.g., presence or absence, watching TV, playing video games, sleeping patterns, and other daily activities), political orientations, types of home appliances [10], [11], [12] and even religious beliefs. Indeed, conducting an in-depth analysis of consumption values provides profound insight relevant to individuals' daily habits [13]. Consequently, this can lead to serious privacy violations. Fig. 1 presents a sample power trace derived from a smart meter's daily consumption data. It highlights overnight period, breakfast time, office hours, and evening activities (e.g., taking shower, doing laundry, and working on a computer). Furthermore, more advanced data mining techniques can reveal more detailed information about the user [14].

To address these privacy concerns, various privacy-preserving protocols have been introduced. Previous protocols have adopted different privacy-enhancing techniques, includ-

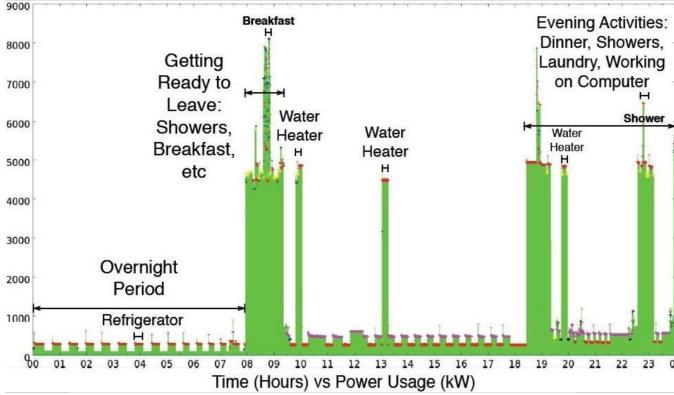


Fig. 1. An example of a full-day power consumption trace [14]

ing data masking schemes, zero-knowledge proof, the use of trusted third parties, anonymization, secure multi-party computation, and bi-homomorphic encryptions. While these schemes offer security and privacy, they also introduce challenges related to implementation and deployment complexity, high computational and communication costs, and scalability issues.

In this paper, we propose a collusion-resistance privacy-preserving aggregation smart metering protocol for operational utility. The protocol enables energy suppliers to collect aggregated consumption values at predefined intervals (e.g., every 15 minutes) without compromising consumer privacy. This scheme leverages an additive partially homomorphic encryption and a noise addition technique to preserve privacy while maintaining data utility. Furthermore, our scheme resists the collusion attack of semi-trusted entities (i.e., the aggregator and the utility provider). Based on this scheme, the utility provider can balance electricity generation and consumption in near real-time and prevent energy waste or power outages. This protocol is scalable and provides accurate aggregated consumption values for each area or district to support grid stability monitoring.

Our protocol preserves consumer privacy while supporting various operational utilities (e.g., grid management, grid monitoring, and load forecasting). Our contributions can be summarized as follows:

- 1) The protocol enables the utility provider to collect fine-grained consumption values for essential operational services while preventing inference or extraction of individual readings.
- 2) The protocol utilizes fog computing, thereby most of the computations are performed by the aggregator, reducing computational overhead on the meter side.
- 3) The protocol is resistant to collusion attacks involving semi-trusted entities.
- 4) The adopted noise mechanism enhances consumer privacy while maintaining data utility by ensuring a zero-sum noise result.

The remainder of this paper is organized as follows: Section II reviews related work. Section III presents the system model, threat model, design goals, and underlying assumptions. Section IV provides the necessary preliminaries. In Section V,

we describe our scheme. Section VI evaluates the proposed scheme. Finally, Section VII concludes the paper.

II. RELATED WORK

In the literature, various techniques and approaches have been introduced to support operational utilities. In smart grid infrastructure, the utility provider aims to precisely match customer demand with grid production to minimize grid instability and fluctuations. Achieving this objective requires access to high-frequency consumption values to control and monitor the status of the smart grid. However, such data enables utility providers to profile customers' daily activities, thereby raising significant privacy concerns. To overcome this challenge, several protocols have been proposed to preserve customer privacy against adversarial actors. To facilitate operational services while mitigating privacy risks, privacy-enhancing techniques have been employed, including transforming trackable consumption values into untrackable ones, reporting aggregated readings, adding noise to the consumption data, or leveraging cryptography mechanisms. In literature, these privacy-preserving protocols are classified into four categories: (1) Aggregation via third parties, (2) Aggregation without third parties, (3) Anonymous reporting via third parties (TPs), and (4) Reporting via anonymous overlay networks. This categorization is depicted in Fig. 2.

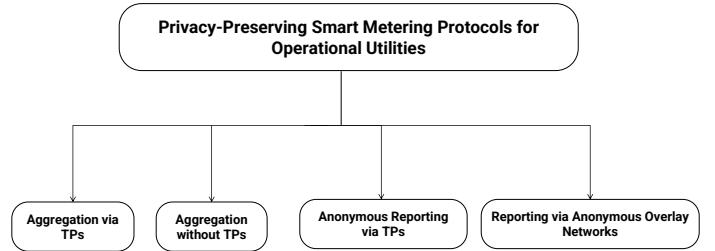


Fig. 2. Categorization of privacy-preserving smart metering protocols for operational services.

A. Aggregation via TPs

In this class, intermediary nodes (e.g., gateways or aggregators) collect and process (e.g., aggregate) consumption values using privacy-enhancing techniques such as homomorphic encryption. While these schemes reduce privacy risks by hiding individual data, they suffer from collusion attacks and a single point of trust. If the intermediary node colludes with the utility provider, it results in a user privacy violation.

Bohli et al. [15] present two privacy-preserving protocols for smart metering: the first involves a fully trusted third party (TTP) that collects individual meter readings, aggregates them, and sends only the sum to the energy supplier, achieving perfect privacy under strong encryption; in the second solution, meters apply noise to consumption readings, thereby reducing the utility provider's certainty about the collected data.

Garcia et al. [16] proposed a privacy-preserving protocol that employs additive homomorphic encryption combined with additive secret sharing to secure individual energy readings. In brief, each meter splits its consumption into random shares,

which are then encrypted and distributed so that only their aggregate—computed via the homomorphic property—is revealed for operational services, thereby preserving individual privacy. However, the protocol’s privacy guarantees are weakened if too few meters participate or if multiple devices are compromised.

Lu et al. [17] utilize the Paillier cryptosystem in conjunction with a super-increasing sequence to compress and encrypt multidimensional electricity usage data into a single ciphertext. In this approach, local gateways aggregate encrypted consumption values without decryption while employing BLS (Boneh-Lynn-Shacham) signatures for efficient batch verification. Each smart meter encrypts its multidimensional consumption data into a single compact ciphertext, preserving customer privacy during aggregation.

Molina-Markham et al. [18] utilize an aggregation scheme in conjunction with data anonymization techniques to report aggregated consumption data to the utility provider via gateways for grid management purposes. In this scheme, the aggregator is trusted and conceals meter identity from the utility provider, which is assumed to be honest but curious.

Mustafa et al. [19] proposed a privacy-enhancing smart metering protocol based on a secure multi-party computation approach. This scheme enables smart meters to split their consumption or generation measurements into secret shares, which are then aggregated by data communication company (DCC) servers using three distinguish algorithms that offer various trade-offs between privacy and efficiency. Based on these algorithms the aggregated result is computed and subsequently distributed to grid operators.

Wang et al. [20] combine differential privacy with an Elgamal-based re-encryption scheme to securely aggregate meters measurements and only the aggregated result is reported to the control center via gateways.

Wang et al. [21] employ an additive variant of Elgamal encryption combined with a binary encoding scheme to report aggregated multi-dimensional readings to the service provider for dynamic billing and grid operation.

Li et al. [22] proposed a privacy-preserving smart metering scheme based on Paillier encryption, bilinear pairing, and signature schemes. In this protocol, multi-dimensional readings, along with customer characteristics, are encoded by a binary encoding function and then encrypted using the Paillier cryptosystem. The power supply company (PSC) is fully trusted and manages the grid in conjunction with billing services. Afterward, fog servers apply aggregation rules based on customer characteristics, with these rules being defined by the outsourcing service provider (OSP). Ultimately, the aggregated results of different aggregation classes are reported to the PSC for grid management and to the OSP for tariff and pricing specification.

Zhang et al. [23] introduce a multi-channel privacy-protection metering protocol that provides three key utilities: billing, operation, and value-added services. The protocol utilizes three distinct channels for these utilities. For operational utility, readings are transmitted at a high frequency to the trusted intermediary substation. The substation aggregates individual values and sends the aggregated result to the network

operator for grid monitoring and management.

B. Aggregation without TPs

Unlike the previous category, this approach does not rely on intermediary nodes. Instead, meters collaborate with each other or leverage various privacy-preserving techniques such as homomorphic encryptions and zero-knowledge proofs to securely report readings to the utility provider. While these privacy-preserving solutions introduce no trust dependency, they raise computational and communication overhead, as well as complexity and scalability issues. For example, as the number of meters grows, management of direct interaction can become inefficient.

Vetter et al. [24] proposed a privacy-friendly smart metering protocol that employs additive homomorphic encryption alongside a homomorphic message authentication code (MAC). In this protocol, meters encrypt and transmit measurement values at predefined intervals to the energy management system (EMS). The EMS is functionally trusted, meaning it correctly responds to SQL queries. Finally, the energy provider can send SQL queries to EMS for temporal and spatial aggregation to obtain aggregated results for grid operational utilities.

Li et al. [25] leverage additive homomorphic encryption as the core component of their protocol to encrypt readings at specified intervals. In this scheme, a spanning tree is constructed over the wireless mesh network of smart meters, with a collector serving as the root node. During tree construction, meters transmit their encrypted values to their respective parent nodes. Subsequently, parent nodes aggregate the received encrypted values, add their own, and then forward the aggregated result to the next level. According to this protocol, the utility provider only accesses the aggregated result, which is primarily utilized for grid management purposes.

Mármol et al. [26] outline a privacy-enhancing protocol that employs bihomomorphic encryption, that is additive homomorphic with respect to both the plaintext and key space. In this scheme, meters encrypt their readings and then send them directly to the Energy Supplier (ES). A ring-based key update mechanism is used for key adjustment, where meters update their keys and transmit them to the key aggregator. Each time a meter is randomly selected as the key aggregator. The key aggregator transmits the aggregated decryption key to the ES. The ES cannot decrypt individual readings unless it aggregates all received values. The ES can only obtain the aggregated result.

Dimitriou et al. [27] propose two decentralized protocols for privacy-preserving, scalable aggregation: one uses symmetric cryptography with neighbor-shared randomness (secure under semi-trusted adversaries), and the other uses public-key encryption with non-interactive zero-knowledge proofs (secure under active adversaries). They extend this in [28] with implementation, simulation and evaluation demonstrating scalability and efficiency.

Nabil et al. [29] introduce a privacy-enhancing AMI scheme achieving (1) load monitoring and energy management, (2) dynamic billing, and (3) energy theft detection, where meters

record consumption every 30 minutes and mask readings via pairwise secret-sharing masks that cancel upon aggregation to reveal only the sum to the system operator.

C. Anonymous Reporting via TPs

In this group, a trusted TP is typically involved, serving as an entity that both the smart meter and the utility provider rely on. The trusted TP removes the meter's sensitive information using anonymization techniques (e.g., replacing real identities with pseudonyms) and cryptographic mechanisms before reporting consumption values to the control center. Although these protocols offer simplicity, they introduce issues such as reliance on a single point of trust and an elevated risk of attacks targeting the intermediary node.

Petrlic [30] proposed a privacy-preserving smart metering protocol using a trusted module platform, public-key cryptography, pseudonymous credentials, and anonymization. Smart meters also encrypt readings and forward encrypted value to an energy service Provider (ESP) via a collector node. The collector node anonymizes the readings before reporting them.

Efthymiou et al. [31] presents a secure protocol for smart metering privacy, where smart meters utilize two distinguished IDs: (1) Low-Frequency ID (LFID) for billing purposes, involving infrequent attributable readings, and (2) High-Frequency ID (HFID) for load management services, involving frequent anonymous readings. Their scheme incorporates a trusted third-party escrow that manages HFID setups to ensure the unlinkability of HFID and LFID. Meters report their readings using an HFID to the utility provider via a third party (or substation). The substation acts as a relay node. Finally, the utility provider collects fine-grained consumption values for operational services without knowing the identity of the source.

D. Reporting via Anonymous Overlay Networks

In this approach, meters utilize anonymous overlay networks (e.g., Tor, Freenet, and I2P) alongside cryptographic mechanisms to anonymously report readings to the utility provider. Although these schemes provide strong anonymity, their performance degrades due to the nature of anonymous overlay networks.

Finster et al. [32] introduce a privacy-preserving scheme employing anonymous overlay networks, blind digital signatures, pseudonyms, and Bloom filters: smart meters transmit high-resolution usage data (e.g., every five minutes) via an anonymous overlay network to an untrusted grid operator, replacing original IDs with pseudonyms; the GO then initializes and broadcasts a Bloom filter to let meters verify their submissions. In this scheme, the GO can access fine-grained consumption data but cannot link it to a customer's identity.

III. PROBLEM STATEMENTS AND DESIGN GOALS

As previously discussed, the transmission of fine-grained consumption data at a higher frequency can reveal customers' daily habits, presence or absence, and even results in home appliance identification, thereby compromising their privacy.

On the contrary, energy suppliers require non-attributable fine-grained consumption readings for grid status analysis. To address both data privacy and utility, a collusion-resistant, privacy-preserving, aggregation-based smart metering protocol is designed to support both privacy and operational utility requirements such as real-time grid monitoring, accurate load forecasting, and maintaining demand-supply equilibrium in the smart grid network.

In the rest of this section, we first define the architecture of the smart grid along with its key components as a system model. Additionally, we outline trust levels associated with different entities within the protocol in the threat model and assumption subsections. Lastly, we explain our scheme's design goal.

A. System Model

The protocol system model comprises three key entities: (1) smart meters, (2) aggregators, and (3) the utility provider. Entities collaborate and execute a specific part of the protocol to deliver a broad range of operational services within the AMI network. In the following sections, we provide a concise definition of each component within the scheme's system model. The smart grid's system model adopted in our protocol is depicted in Fig. 3.

- 1) **Smart Meters:** Smart meters are intelligent electronic devices that measure power usage with high precision and report it at predefined intervals (e.g., typically every 15 minutes) to an aggregator. Smart meters operate with constrained computational resources but are capable of cryptographic functions, including encryption, decryption, noise addition, digital signing, and verification.
- 2) **Aggregators:** Aggregators collect consumption values, process them (e.g., by aggregating the data), and forward the processed values to the control center. These intermediary nodes possess strong computational resources and high storage capacity. Additionally, they function as edge nodes, reducing the overall computational overhead for both smart meters and the utility provider. The energy sector specifies numerous areas for the deployment of aggregators in each city or town.
- 3) **Utility Provider:** The utility provider collects and analyzes aggregated consumption values received from aggregators for various operational purposes, such as load forecasting, demand-supply balancing, and grid management and monitoring. The utility provider benefits from powerful computing resources and extensive storage capacity.

B. Threat Model and Assumptions

As we mentioned earlier, we define trust levels for each entity, including the smart meter, the aggregator, and the utility provider. We define our scheme's threat model as follows:

- **Smart Meters:** Intelligent electricity meters are fully trusted and always adhere to the protocol. They incorporate robust tamper-proof protection mechanisms designed to detect and counteract any attempts at reverse

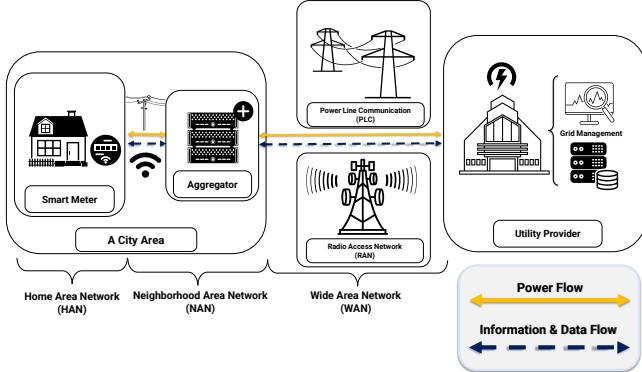


Fig. 3. The smart grid's system model comprises smart meters, aggregators, and the utility provider.

engineering or physical intrusion such as invasive, semi-invasive, and non-invasive hardware attacks. If tampering is detected, the meter triggers an alert to notify the control center about the malicious activity. Subsequently, the meter transitions to a non-operational state and erases its security credentials (e.g., cryptographic keys). Finally, the utility provider agent investigates the incident and takes appropriate action against malicious customers.

- **Aggregator:** Aggregators are considered as trusted but curious edge servers. While they adhere to the established protocol, they may attempt to gain deeper insight into the customers' energy consumption data out of curiosity. Since aggregators are configured and managed by the utility provider, the collusion of these two parties is possible.
- **Utility Provider:** Similar to aggregators, the utility provider is semi-trusted and follows the protocol. However, as it is also considered curious, it may analyze consumers' power usage data to extract additional insights for its own benefit. Both aggregators and the utility provider reside within the semi-trusted domain

The threat model is depicted in Fig. 4.

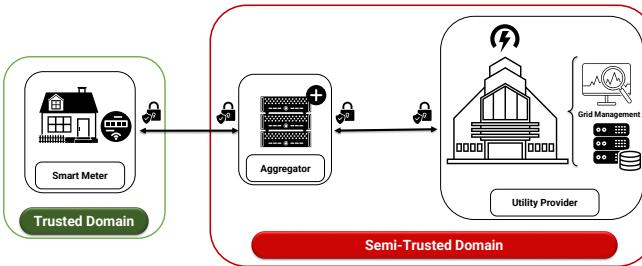


Fig. 4. The scheme's threat model

We also established several assumptions in our protocol which are summarized as follows:

- **Secure Communication Channels:** The communication channels between entities in our system model are secure, ensuring that adversarial actors cannot launch active or passive attacks on packets in transit.

- **Pre-Established Entity Authentication:** Meters, aggregators and the utility provider utilize an authentication scheme and register themselves to the AMI network, before protocol initiation.
- **Secure Public-Key Distribution:** Public keys are securely distributed (e.g., by using public key certificates) among meters, aggregators, and the utility provider. This ensures that each entity obtains the correct public keys of others for encryption and verification.

C. Design Goals

Our collusion-resistance, privacy-preserving, aggregation-based, smart metering protocol aims to achieve the following goals:

- I. **Privacy:** The scheme aims to preserve customer privacy against semi-trusted entities by leveraging different privacy-enhancing techniques. Moreover, the scheme conceals individual readings and resists collusion attacks from semi-trusted entities.
 - a) **Preserving Fine-Grained Readings Privacy:** In our proposed protocol, only the aggregated result is reported to the utility provider at each interval, ensuring that the control center cannot access the fine-grained consumption data.
 - b) **Collusion-Resistance:** Our scheme employs a data perturbation technique that prevents collusion attacks by semi-trusted entities (i.e., aggregators and the utility provider). Consequently, any collusion between the edge node and the control center results in obtaining a noisy fined-grained consumption value.
- II. **Data Utility:** The utilized data perturbation technique maintains data utility due to its zero-sum property. This characteristic ensures that the trade-off between privacy and utility is preserved and the control center can access the concise aggregated result.
- III. **Compatibility with Smart Grid Infrastructure:** The protocol's computational demands align with the processing capabilities of the current smart grid devices, ensuring seamless integration within the existing network.

IV. PRELIMINARIES

In this section, we discuss the Paillier cryptosystem, a partially additive homomorphic encryption scheme that consists three key algorithms, including (1) key generation, (2) encryption and (3) decryption. The Paillier encryption scheme is built upon decisional composite residuosity assumption [33], which asserts the intractability of distinguishing whether a given integer is an n -th power modulo n^2 . In the following, we examine each algorithm and the cryptosystem's additive homomorphic property.

1) Key Generation Algorithm:

- Choose two large prime numbers p and q such that $\gcd(pq, (p-1)(q-1)) = 1$
- Compute $n = pq$, $\lambda = \text{lcm}(p-1, q-1)$
- Choose a random value g such that $g \in \mathbb{Z}_{n^2}^*$

- Ensure n divides the order of g by checking the modular multiplicative inverse as follows:
 $\mu = (L(g^\lambda \bmod n^2))^{-1} \bmod n$
- Compute function L as: $L(x) = \frac{x-1}{x}$
- The public key is (n, g) and the secret key is (λ, μ)

2) Encryption Algorithm:

- Let m be an arbitrary message, where $0 \leq m < n$
- Choose a random value r where $0 < r < n$ and $\gcd(n, r) = 1$
- Compute ciphertext c as: $c = g^m \cdot r^n \bmod n^2$

3) Decryption Algorithm:

- Let c be the ciphertext where $c \in \mathbb{Z}_{n^2}^*$
- Compute the plaintext message m as follows:
 $m = L(c^\lambda \bmod n^2) \cdot \mu \bmod n$

4) Additive Homomorphic Property:

- The multiplication of two ciphertexts is decrypted to a value that corresponds to the sum of their respective plaintexts m_1 and m_2 .
- The formal description is as follows:
 $Dec(Enc(m_1, r_1) \cdot Enc(m_2, r_2) \bmod n^2) = m_1 + m_2 \bmod n$

Our protocol employs the additive homomorphic property of Paillier cryptosystem to aggregate fine-grained consumption values.

V. PROPOSED SCHEME

The proposed scheme introduces data privacy while maintaining data utility through various security and privacy mechanisms, including partially additive homomorphic encryption (i.e., the Paillier cryptosystem), data perturbation, and data minimization. Preserving data utility is essential for real-time grid monitoring and management. In fact, the utility provider requires concise aggregated energy usage data to maintain a balance between electricity generation and consumption, as well as for load monitoring and forecasting. This protocol consists of multiple steps to generate accurate aggregated results for the control center.

A. Protocol Overview

We briefly outline the steps of our proposed protocol in the following:

- At the beginning of the protocol, the edge node (as the aggregator of a local area) randomly chooses one of the authenticated intelligent meters in the AMI network at each interval (e.g., every 15 minutes), as the designated (selected) smart meter.
- In the next step, the utility provider broadcasts an encrypted message containing the selected smart meter's ID to the network.
- Smart meters receive the message, decrypt it, and compare the received ID with their own ID to verify whether they have been selected.
- Non-designated smart meters, generate a random value (used as noise) using a Pseudo Random Number Generator (PRNG) following a Gaussian (normal) distribution.

- Unlike other smart meters, the designated meter is not permitted to generate a random value, instead, it must wait to receive a random value generated by the non-designated meters in the network.
- Subsequently, each meter adds the generated noise to its measured power usage to create a noisy consumption value.
- Since energy usage data is utilized for operational services, an appropriate data minimization technique is applied to report only required fields (e.g., power usage in kWh)
- Each meter then encrypts its noisy consumption value using the utility provider's public key and encrypts the generated random value using the designated smart meter's public key.
- Non-designated meters forward these two encrypted messages to their aggregator. The aggregator utilizes the additive homomorphic property of the Paillier cryptosystem to aggregate both encrypted random values and encrypted noisy consumption values.
- The aggregator sends the encrypted aggregated random value back to the designated meter. The selected smart meter decrypts and computes the additive inverse of aggregated random value (to result in zero-sum noise on the whole), and adds the computed noise to its measured energy usage data. Lastly, it encrypts and sends its noisy consumption data to the edge node.
- Upon receiving encrypted noisy consumption data from the designated meter, the aggregator computes the aggregated result (where noisy values cancel out) and forwards it to the utility provider.
- Finally, the utility provider accesses the aggregated value for various operational purposes, such as grid status analysis, management, and monitoring.

The protocol overview is illustrated in Fig 5. In the subsequent section, we provide a detailed analysis of the proposed protocol.

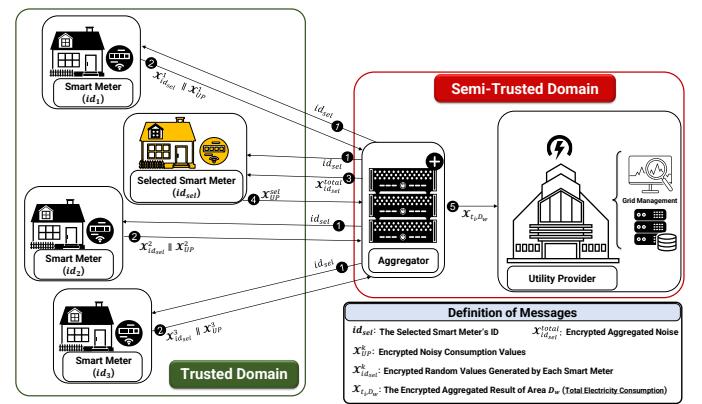


Fig. 5. The protocol overview

B. Protocol Details

In this section, we introduce the proposed scheme in more details. The notations used in the formal specification of the protocol are defined in Table I.

TABLE I
OPERATIONAL SERVICE COMPONENTS

Symbols	Description	Formal Representation
M	Number of registered, authenticated, and active smart meters in the network	$M \in \mathbb{N}$
t_i	i th time interval	$T = \langle t_1, t_2, \dots, t_i, \dots, t_n \rangle$
id_i	A meter with the identifier id_i which then can be chosen as the selected smart meter id_{sel}	$id_i \in \mathbb{ID}$
s_{t_i,id_k}	The random value generated by the smart meter with identifier id_k during interval t_i	$s_{t_i,id_k} \in \mathbb{S}_{id_k}$
pk_X, sk_X	The public-private key pair of each entity, including meters $\langle pk_{id_k}, sk_{id_k} \rangle$, the local aggregator $\langle pk_{AGG}, sk_{AGG} \rangle$, and the utility provider $\langle pk_{UP}, sk_{UP} \rangle$	$pk_X \in \mathbb{PK}, sk_X \in \mathbb{SK}$
S_{t_i}	The additive inverse (or counterpart) of the sum of random values generated by the non-designated meter at interval t_i	$S_{t_i} \in \mathbb{S}_{D_w}$
c_{t_i,id_k}	The consumption value of the meter with identifier id_k during interval t_i	$c_{t_i,id_k} \in \mathbb{C}_{id_k}$
nc_{t_i,id_k}	The noisy consumption value of the meter with identifier id_k during interval t_i	$nc_{t_i,id_k} \in \mathbb{NC}_{id_k}$
C_{t_i,D_w}	The total consumption of area D_w at interval t_i	$C_{t_i,D_w} \in \mathbb{C}_{D_w}$

As we mentioned earlier in section III-B, the protocol considers three key assumptions; existing secure communication between entities, authentication of entities before participating in the protocol, and secure distribution of entities' public keys before execution of the protocol. According to these assumptions, entities have access to other's public keys and can transmit energy data without any risk of tampering or modification.

At each interval (e.g., every 15 minutes), the local aggregator randomly selects an intelligent electricity meter from the set of available network meters as the designated smart meter. It then forwards the message id_{sel} , containing the ID of the designated meter at interval t_i through a secure communication channel to all smart meters in its network.

$$AGG \rightarrow SMs : id_{sel} || t_i \quad (1)$$

Afterward, each meter receives the message, and compares the received ID with its own ID. This allows the meter to determine the designated meter's ID and whether it has been selected for the current interval t_i .

Non-designated meters employ a noise addition algorithm, \mathcal{G} -norm, using a secure PRNG \mathcal{G} and an initial seed \mathcal{S} that follows a Gaussian distribution.

$$\mathcal{G} : \{0, 1\}^l \rightarrow \{0, 1\}^L, L \gg l \quad (2)$$

$$r_i \leftarrow \mathcal{G}(\mathcal{S}) \quad (3)$$

$$u_i \leftarrow \frac{\text{int}(r_i)}{2^L} \quad (4)$$

r_i is an L -bit string derived from \mathcal{G} (e.g., Threelfry PRNG) which is a pseudorandom sample from uniform distribution

and u_i is a real number in $[0, 1]$. Various standard transformation methods exist for mapping uniform PRNG outputs to a normal distribution, such as the Ziggurat approach, the Box-Muller algorithm, and the Gaussian inverse cumulative distribution function (CDF). For inverse transform sampling (N_i^{ITS}), the following relation can be used.

$$N_i^{ITS} = \sigma \Phi^{-1}(u_i), \quad u_i \sim U(0, 1), \quad (5)$$

The parameter $\Phi^{-1}(u_i)$ represents the Gaussian inverse cumulative distribution function. However, Box-Muller is a more efficient transformation method, that can be computed as follows:

$$\begin{aligned} R &= \sqrt{-2 \ln(u_{2i-1})}, \\ \Theta &= 2\pi u_{2i} \\ N_{2i-1}^{BM} &= \sigma R \cos(\Theta), \\ N_{2i}^{BM} &= \sigma R \sin(\Theta), \end{aligned} \quad (6)$$

The parameters u_{2i} and u_{2i-1} are two uniform pseudorandom numbers derived from \mathcal{G} and σ represents the standard deviation of normal distribution. The parameters N_{2i-1}^{BM} and N_{2i}^{BM} are two pseudorandom numbers that follow a Gaussian distribution with zero mean and standard deviation of σ . As we shown, the Box-Muller transformer generates two pseudorandom numbers. Afterward, the smart meter with id_k can select one of them randomly as a random value s_{t_i,id_k} at that interval.

$$\begin{aligned} b &\leftarrow_R \{0, 1\} \\ s_{t_i,id_k} &\leftarrow \begin{cases} N_{2i-1}^{BM} & \text{if } b = 0 \\ N_{2i}^{BM} & \text{if } b = 1 \end{cases} \end{aligned} \quad (7)$$

Finally, by combining a secure PRNG with an efficient transformation method, each non-designated meter generates a random value s_{t_i,id_k} to perturb the consumption data c_{t_i,id_k} at each interval t_i . Before applying the noise value to the consumption data, the smart meter employs a data minimization technique to report only the required fields (e.g., power usage in kWh) to the utility provider.

$$\begin{aligned} s_{t_i,id_k} &\leftarrow \mathcal{G}\text{-norm}(\mathcal{S}, \sigma) \\ nc_{t_i,id_k} &\leftarrow c_{t_i,id_k} + s_{t_i,id_k} \end{aligned} \quad (8)$$

Subsequently, each meter encrypts both the noisy consumption value nc_{t_i,id_k} and the random value s_{t_i,id_k} .

$$\begin{aligned} \mathcal{X}_{id_{sel}}^k &\leftarrow Enc(pk_{id_{sel}}, s_{t_i,id_k}) \\ \mathcal{X}_{UP}^k &\leftarrow Enc(pk_{UP}, nc_{t_i,id_k}) \end{aligned} \quad (9)$$

Each meter transmits encrypted values to the edge node.

$$\begin{aligned} \mathcal{X}^k &= \mathcal{X}_{id_{sel}}^k || \mathcal{X}_{UP}^k \\ SM \rightarrow AGG : \mathcal{X}^k \end{aligned} \quad (10)$$

The Aggregator utilizes the additive homomorphic property of the Paillier cryptosystem to aggregate encrypted random values received from non-designated meters and computes the

aggregated encrypted random value. It then forwards the result to the designated smart meter.

$$\begin{aligned} \mathcal{X}_{id_{sel}}^{total} &\leftarrow \sum_{k=1}^{M-1} \mathcal{X}_{id_{sel}}^k \\ AGG \rightarrow SM_{sel} : \mathcal{X}_{id_{sel}}^{total} \end{aligned} \quad (11)$$

The designated smart meter decrypts and computes the additive inverse of aggregated random values to counteract the noise addition of non-designated meters. Furthermore, the selected meter applies the noise to its measured power usage and encrypts it. Finally, the designated meter sends its encrypted energy usage to the aggregator.

$$\begin{aligned} S_{t_i} &\leftarrow -(Dec(sk_{id_{sel}}, \mathcal{X}_{id_{sel}}^{total})) \\ nc_{t_i,id_{sel}} &\leftarrow c_{t_i,id_{sel}} + S_{t_i} \\ \mathcal{X}_{UP}^{sel} &\leftarrow Enc(pk_{UP}, nc_{t_i,id_{sel}}) \\ SM_{sel} \rightarrow AGG : \mathcal{X}_{UP}^{sel} \end{aligned} \quad (12)$$

Upon receiving the final noisy consumption value for interval t_i , the aggregator performs the final aggregation operation on the encrypted noisy consumption values and forwards the resulting encrypted aggregated value to the utility provider.

$$\begin{aligned} \mathcal{X}_{t_i,D_w} &\leftarrow \sum_{k=1}^{M-1} (\mathcal{X}_{UP}^k) + \mathcal{X}_{UP}^{sel} \\ AGG \rightarrow UP : \mathcal{X}_{t_i,D_w} \end{aligned} \quad (13)$$

The utility provider decrypts \mathcal{X}_{t_i,D_w} and obtains the total electricity consumption of a given area. The aggregated values are collected and analyzed for various operational utilities, such as grid management and load monitoring.

$$C_{t_i,D_w} \leftarrow Dec(sk_{UP}, \mathcal{X}_{t_i,D_w}) \quad (14)$$

The protocol flow is illustrated in Fig. 6. The following equations indicate how noise values cancel out after aggregation. According to the protocol, the following relation holds:

$$S_{t_i} = - \sum_{k=1}^{M-1} s_{t_i,id_k} \quad (15)$$

In the following equation, we demonstrate the noise cancellation process.

$$\begin{aligned} C_{t_i,D_w} &= \sum_{k=1}^M (c_{t_i,id_k}) + \sum_{k=1}^{M-1} (s_{t_i,id_k}) + (S_{t_i}) \\ &= \sum_{k=1}^M (c_{t_i,id_k}) + \sum_{k=1}^{M-1} (s_{t_i,id_k}) - \sum_{k=1}^{M-1} (s_{t_i,id_k}) \\ &= \sum_{k=1}^M (c_{t_i,id_k}) \end{aligned} \quad (16)$$

VI. EVALUATION

We evaluated our scheme from two key aspects: (1) performance and (2) privacy. For performance evaluation the computational, memory, and communication overhead of the proposed protocol have been examined, whereas for privacy

evaluation, normalized conditional entropy (NCE) has been used to evaluate how effectively the protocol preserves consumer privacy against semi-trusted entities.

A. Performance Evaluation

We first conduct an in-depth analytical evaluation for computational, memory, and communication overhead. Additionally, we supplement the computational overhead with an experimental evaluation to indicate the scheme's efficiency. We further estimate protocol memory requirement based on our implementation.

In our experimental setup, the protocol is deployed on two distinct systems. The first system, simulating the smart meter, employs the Quick Emulator (QEMU) to replicate an Orange Pi One PC, which is equipped with a 1.2GHz 32-bit ARM Cortex-A7 processor and 1 GB of RAM. This emulated environment operates on Armbian, an Ubuntu-based operating system tailored for Internet of Things (IoT) applications. The second system equipped with a 1.8GHz Intel Core i7 processor (8 cores) and 32GB of RAM, represents the aggregator and the utility provider. To assess protocol feasibility under memory constraints, the RAM available on the second machine is intentionally restricted to 1GB. The second system operates on Debian 12.

We implemented our scheme in Python with optimized cryptographic libraries, such as LightPHE [34] for partially homomorphic encryptions (i.e., the Paillier cryptosystem) and randomgen [35] for generating pseudorandom values to perturb readings (i.e., the Threecry PRNG).

We utilized a dataset [36] from Slovakia's AMI infrastructure consisting of energy consumption data from 1000 anonymized residential smart meters. The dataset includes both active and reactive energy readings, recorded at 15-minute intervals over the course of a full year. This dataset is privatized using traditional techniques such as data minimization and anonymization. It is also important to note that it does not suffer from null values and data sparsity issues.

1) Computational Overhead: To analytically evaluate the protocol's computational overhead, we provide a notation table (See Table II) which represents various symbols. As different parts of the protocol are executed by different entities, we examine the execution time for each entity.

TABLE II
NOTATION TABLE FOR COMPUTATIONAL OVERHEAD

Symbol	Description
t_{arthm}	Time of performing various arithmetic operation, including addition, multiplication, comparison, and modulus.
$t_{RandGen}$	Time of generating a random value by combining a secure PRNG with a transformation method
$t_{asym-op}$	Time of performing asymmetric cryptography operations, including encryption and decryption

We can compute the computational overhead on non-designated meters as follows:

$$t_{sm} = 2t_{arthm} + t_{RandGen} + 2t_{asym-op} \quad (17)$$

Smart meters should be able to perform cryptographic primitives, including encryption, decryption, noise generation, and noise addition.

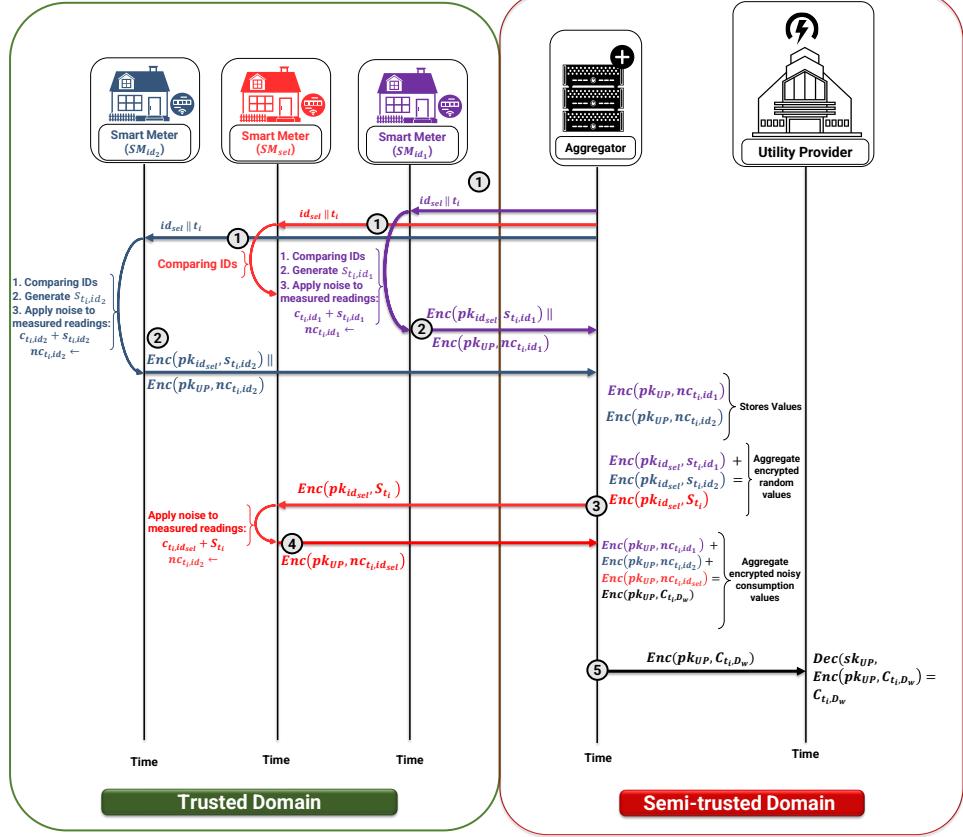


Fig. 6. The protocol flow

In our proposed protocol, the designated meter plays a pivotal role in canceling noisy values generated by the non-designated smart meters. The following relation computes the computational overhead on the designated smart meter.

$$t_{sm}^{sel} = 3t_{arthm} + 2t_{asym-op} \quad (18)$$

The designated meter is not permitted to generate random values. Instead, it receives the encrypted aggregated random value, generated by non-designated smart meters and uses this value to perturb its consumption data.

The edge node performs two essential operations during the protocol execution. First, it randomly selects one of the available smart meters, encrypts the selected meter's ID, and broadcasts it to the AMI network. Subsequently, The aggregator performs a two-step aggregation on encrypted random values and noisy consumption data. The computational overhead on the aggregator node is computed as follows where M is number of active meters that participated in the protocol.

$$t_{agg} = 2M - 1(t_{arthm}) \quad (19)$$

The Utility provider only decrypts the data and obtains the total electricity consumption of an area.

$$t_{up} = t_{asym-op} \quad (20)$$

Finally, the analytical computational overhead for the entire protocol at each time interval t_i can be computed as follows:

$$t_{op} = t_{sm} + t_{sm}^{sel} + t_{agg} + t_{up} \quad (21)$$

We also conduct an experiment to evaluate the protocol's performance and efficiency. The table III presents the protocol execution time per entity in the practical experiment. As observed, by increasing the security parameter's length (i.e., the Paillier key size), the execution time increases.

We also demonstrate the scheme's execution time based on the Paillier key size in Fig. 7.

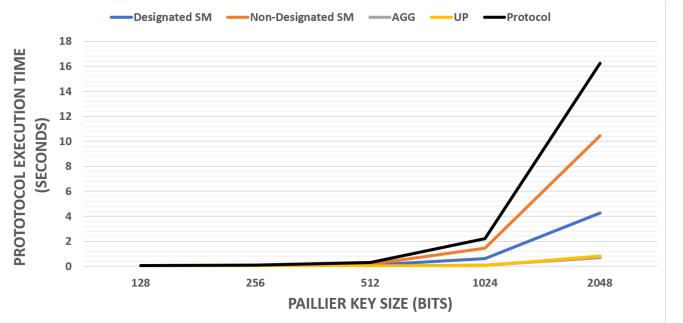


Fig. 7. The protocol flow

2) Memory Overhead: Similar to the computational overhead, we examine the memory requirement of the proposed scheme for each distinct entity. We also provide a notation table (see Table IV) for memory usage which includes formal symbols used in the analysis of memory overhead.

TABLE III
PROTOCOL EXECUTION TIME (IN SECONDS) PER EACH ENTITY IN THE OPERATIONAL SERVICE (20 METERS PARTICIPATED)

Paillier Key Size	Designated Smart Meter (t_{sm}^{sel})	Non-Designated Smart Meter (t_{sm})	Aggregator (t_{agg})	Utility Provider (t_{up})	Total Protocol Execution Time (t_{op})
128-bit	0.00658	0.04920	0.00078	0.00059	0.05717
256-bit	0.01854	0.06599	0.00206	0.00162	0.08822
512-bit	0.10286	0.18256	0.01178	0.01056	0.30777
1024-bit	0.61290	1.45342	0.08223	0.06388	2.21245
2048-bit	4.27335	10.45306	0.68692	0.81623	16.22958

TABLE IV
NOTATION TABLE FOR MEMORY OVERHEAD

Symbol	Description
S_{id_k}	Size of a meter identifier id_k
S_{cipher}	An average size of generated ciphers during protocol execution.
S_c	Size of consumption/metering data
S_{nc}	Size of noisy consumption/metering data
S_{Rand}	Size of a generated random value
$S_{<pk,sk>}$	Size of the public and private key pair
S_{pk}	Size of entities' public key

To assess the memory overhead associated with each entity in the proposed scheme, we begin by evaluating the memory consumption of non-designated smart meters. We analyze the memory overhead for these non-designated smart meters using the following relation.

$$S_{sm} = 2S_{id_k} + S_{Rand} + S_c + S_{nc} + 2S_{cipher} + S_{<pk,sk>} + S_{pk} \quad (22)$$

Relation 22 demonstrates the worst-case scenario; however, there are ways to reduce memory usage. For instance, meters can easily free up memory by erasing cipher values after transmitting them to the edge node.

After analyzing the non-designated meters, we conduct an analysis on the designated meter memory overhead. Relation 23 indicates the memory consumption for the designated meter.

$$S_{sm}^{sel} = S_{id_k} + S_{Rand} + S_c + S_{nc} + 2S_{cipher} + S_{<pk,sk>} + S_{pk} \quad (23)$$

The aggregator consumes more memory compared to other entities in the protocol. Consequently, it is necessary to assess memory overhead on this intermediary node. The following relation shows the amount of memory required by the aggregator Where M is the number active meters that participated in the protocol.

$$S_{agg} = M(S_{id_k}) + 2M(S_{cipher}) + S_{<pk,sk>} + S_{pk} \quad (24)$$

The utility provider only collects total consumption at each interval t_i ; therefore it does not consume as much memory as the aforementioned entities. The memory usage of the utility provider can be computed as follows:

$$S_{up} = S_{cipher} + S_c + S_{<pk,sk>} + S_{pk} \quad (25)$$

Finally, the memory consumption in this protocol at each time interval t_i is computed as follows:

$$S_{op} = S_{sm} + S_{sm}^{sel} + S_{agg} + S_{up} \quad (26)$$

In our experiment, we allocated 1GB of RAM for each entity. According to our results, the total memory consumed by the protocol is approximately 50.34MB which is significantly lower than 1GB.

3) *Communication Overhead*: Lastly, we assess the protocol's communication overhead. Similar to previous sections, we provide a notation table (see Table V) for scheme's the communication overhead.

TABLE V
NOTATION TABLE FOR COMMUNICATION OVERHEAD

Symbol	Description
$P_{id_{sel}}$	The data size for the designated smart meter identifier id_{sel}
$P_{\mathcal{X}_{id_{sel}}^{total}}$	The data size of the encrypted aggregated random value S_{t_i}
$P_{\mathcal{X}_{id_{sel}}^k}$	The data size of the encrypted random value s_{t_i, id_k}
$P_{\mathcal{X}_{UP}^k}$	The data size of the encrypted noisy consumption value nct_{t_i, id_k}
$P_{\mathcal{X}_{t_i, D_w}}$	The data size of the encrypted total electricity consumption C_{t_i, D_w} for the domain D_w

The maximum data size exchanged in the grid network at each interval t_i represents the minimum bandwidth required by the AMI infrastructure to successfully transmit the entire payload through the link. Consequently, the minimum required bandwidth (i.e., having no transmission delay) for the protocol execution at each time interval t_i can be computed as follows, where M is the number of smart meters, which are managed by a local aggregator.

$$P_{op} = \max(P_{id_{sel}}, P_{\mathcal{X}_{id_{sel}}^{total}}, P_{\mathcal{X}_{id_{sel}}^k}, P_{\mathcal{X}_{UP}^k}, P_{\mathcal{X}_{t_i, D_w}}) \cdot M \quad (27)$$

In this study, we assess the communication overhead in both Neighborhood Area Networks (NAN) and Wide Area Networks (WAN) by considering the link layer technologies employed. In our network architecture, smart meters transmit data to an aggregator using 6LoWPAN alongside IEEE 802.15.4g (Wi-SUN)—a standard specifically designed for NAN that connects smart meters to intermediary local aggregators [37]. The local aggregators aggregate and then relay the encrypted aggregated consumption data of an area to the utility provider via a 4G-LTE network. More specifically, the data is initially sent to the 4G-LTE Radio Access Network (RAN), also known as the Evolved Node B (eNB), then forwarded to the Packet Data Network Gateway (PGW), and finally delivered to the utility provider.

Table VI summarizes the communication overhead for each network link. Because each link utilizes a distinct network stack, the overhead differs accordingly. In particular, the available bandwidth for the Wi-SUN is about 250 kbps and for the 4G-LTE and the ethernet is 1000 kbps.

B. Privacy Evaluation

To assess the privacy aspect of the proposed scheme, we employed a privacy metric called normalized conditional entropy (NCE). The conditional entropy quantifies the amount

TABLE VI
PACKET SIZES AND PER-METER TRANSMISSION TIMES (20 SMs, 12.5 KBPS PER METER IN NAN, 50 KBPS PER METER IN WAN)

Payload	NAN		WAN					
	SM \leftrightarrow AGG		AGG \leftrightarrow eNB		eNB \leftrightarrow PGW		PGW \leftrightarrow UP	
	IEEE 802.15.4g (Wi-SUN)	4G-LTE (PDCP-LTE)	IEEE 802.3 (Ethernet)	IEEE 802.3 (Ethernet)	Packet Size	Time	Packet Size	Time
$P_{id_{sel}} = 16$ B	45 B	0.02880 s	–	–	–	–	–	–
$P_{x_{id_{sel}}^{\text{total}}} = 512$ B	661 B	0.42304 s	–	–	–	–	–	–
$P_{x_{id_{sel}}^k} = 512$ B	661 B	0.42304 s	–	–	–	–	–	–
$P_{x_{UP}^k} = 512$ B	661 B	0.42304 s	–	–	–	–	–	–
$P_{\mathcal{X}_{t_i,D_w}} = 512$ B	661 B	0.42304 s	566 B	0.09056 s	634 B	0.10144 s	578 B	0.09248 s
Available BW	250 kbps		1000 kbps		1000 kbps		1000 kbps	
BW per SM	12.5 kbps		50 kbps		50 kbps		50 kbps	

of information required to describe a random variable X when the value of another random variable Y is known [38]. Here, X represents the actual distribution while Y represents the perturbed or obfuscated distribution observed by semi-trusted entities (i.e., the aggregator and the utility provider).

In conditional entropy (CE), the output range lies in $[0, +\infty)$ which is not an ideal metric for measuring the extent to which privacy is preserved by the proposed scheme. Instead, we use NCE as a privacy metric, as it is bounded within $[0, 1]$. Higher NCE values, closer to 1, indicate stronger privacy preservation, as the perturbed data retains minimal information about the original values. Conversely, lower NCE values, approaching 0, suggest reduced privacy, implying significant data leakage and greater exposure of the actual distribution. The formulas for entropy, CE, and NCE are as follows.

$$H_{CE}(X|Y) = - \sum_{x \in X, y \in Y} p(y, x) \log_2(p(x|y)) \quad (28)$$

$$H_E(X) = - \sum_{x \in X} p(x) \log_2(p(x)) \quad (29)$$

$$H_{NCE}(X|Y) = \frac{H_{CE}(X|Y)}{H_E(X)} \quad (30)$$

In our protocol, there are two distributions: (1) the original distribution (i.e., accurate consumption values), and (2) the perturbed distributions (i.e., noisy consumption values). To ensure customer privacy, we compute the NCE metric based on original and perturbed distribution to evaluate the effectiveness of the privatization protocol in protecting customer privacy against semi-trusted entities.

In the worst-case scenario, the aggregator colludes with the utility provider and does not aggregate the encrypted noisy consumption values. Instead, it directly forwards them to the utility provider. In this situation, the utility provider receives noisy consumption data. Consequently, to ensure customer privacy, we must demonstrate that the noisy values do not reveal information about customers' daily habits. To achieve this, we compare the actual distribution with the perturbed distribution and evaluate the effectiveness of the proposed scheme by utilizing the NCE privacy metric.

According to our experiment, as illustrated in Fig. 8, increasing the noise level (or increasing the scale of the standard

deviation σ) causes the NCE metric to approach 1, indicating higher privacy preservation. This suggests that customer privacy can be enhanced by adjusting the noise level. By applying higher noise levels, we can safeguard user privacy and mitigate collusion attacks launched by semi-trusted entities. The impact of noise levels on increasing NCE values is demonstrated in Table VII, thereby illustrating the enhancement of privacy. Fig. 9 illustrates the impact of added noise on the divergence or distance between the original consumption readings and the perturbed consumption probability distributions.

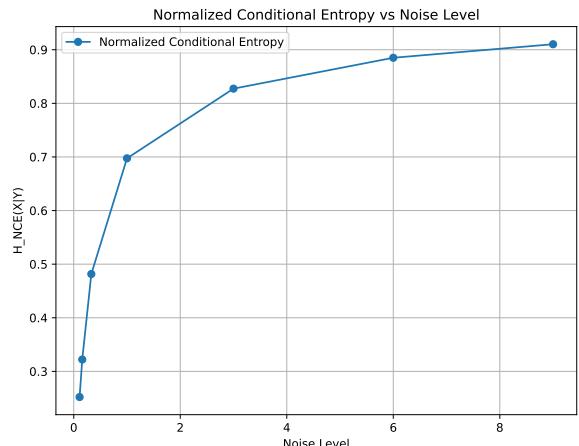


Fig. 8. The impact of noise levels on the NCE metric is significant; a higher noise level results in a dramatic increase in NCE, indicating enhanced customer privacy preservation.

VII. CONCLUSION

In this paper, we proposed a collusion-resistance privacy-preserving aggregation smart metering protocol for operational utilities, such as grid management, load forecasting, and demand-supply balancing. To safeguard customer privacy against semi-trusted entities (e.g., the aggregator and the utility provider), we employed various privacy-enhancing techniques,

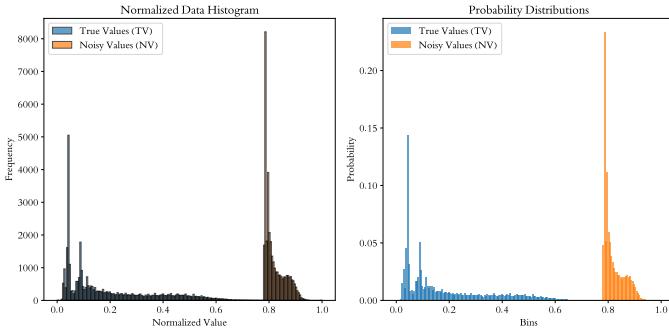


Fig. 9. The divergence between the original (i.e., actual consumption readings) and the perturbed (i.e., noisy consumption readings) probability distributions.

TABLE VII
NORMALIZED CONDITIONAL ENTROPY CALCULATION BASED ON
APPLIED NOISE LEVEL

Applied Noise Level (Standard Deviation Scale)	Normalized Conditional Entropy Values
$\frac{\sigma}{9}$	0.25216
$\frac{\sigma}{6}$	0.32217
$\frac{\sigma}{3}$	0.48160
σ	0.69743
$\sigma \times 3$	0.82740
$\sigma \times 6$	0.88500
$\sigma \times 9$	0.91025

including partially homomorphic encryption (e.g., the Paillier cryptosystem), data perturbation, and data minimization. Specifically, in our protocol, meters perturb their consumption data at each predefined interval and forward encrypted noisy consumption values to the local aggregator. The local aggregator performs an aggregation operation on encrypted noisy consumption values using the additive homomorphic property of the Paillier cryptosystem. This aggregation results in zero-sum noise, and only the aggregated consumption of each area is reported to the energy supplier.

Furthermore, We conduct an in-depth analysis on protocol's performance and privacy. According to the performance evaluation results, our scheme can provide the aggregated result for 20 smart meters of an area at each interval in about 2.21s with 1024-bit key size. Additionally, based on normalized conditional entropy metric, increasing the noise scale leads to greater divergence between original and noisy distributions, thereby enhancing customers' privacy. Our results demonstrate that the proposed scheme achieves strong privacy protection while maintaining computational efficiency through the use of a diverse set of cryptographic techniques.

REFERENCES

- [1] G. Giacconi, D. Gunduz, and H. V. Poor, "Privacy-aware smart metering: Progress and challenges," *IEEE Signal Processing Magazine*, vol. 35, no. 6, pp. 59–78, 2018.
- [2] P. Kumar, Y. Lin, G. Bai, A. Paverd, J. S. Dong, and A. Martin, "Smart grid metering networks: A survey on security, privacy and open research issues," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 3, pp. 2886–2927, 2019.
- [3] U.S. Energy Information Administration, "International energy outlook 2016," 2016, [Accessed: February 14, 2025]. [Online]. Available: <https://www.eia.gov/forecasts/ieo/>
- [4] T. Mazhar, H. M. Irfan, S. Khan, I. Haq, I. Ullah, M. Iqbal, and H. Hamam, "Analysis of cyber security attacks and its solutions for the smart grid using machine learning and blockchain methods," *Future Internet*, vol. 15, no. 2, p. 83, 2023.
- [5] J. Kua, M. B. Hossain, I. Natgunanathan, and Y. Xiang, "Privacy preservation in smart meters: current status, challenges and future directions," *Sensors*, vol. 23, no. 7, p. 3697, 2023.
- [6] J. J. Moreno Escobar, O. Morales Matamoros, R. Tejeida Padilla, I. Lina Reyes, and H. Quintana Espinosa, "A comprehensive review on smart grids: Challenges and opportunities," *Sensors*, vol. 21, no. 21, p. 6978, 2021.
- [7] R. Ferreira and L. A. Barroso, "Smart grids in latin america: Current stance of development and future perspectives," 2016, [Accessed: February 14, 2025]. [Online]. Available: <https://smartgrid.ieee.org/bulletins/november-2016/smart-grids-in-latin-america-current-stance-of-development-and-future-perspectives>.
- [8] R. Ben Romdhane, H. Hammami, M. Hamdi, and T.-H. Kim, "Privacy-preserving approaches for smart metering: A survey," in *Proceedings of the Asia Conference on Electrical, Power and Computer Engineering*, 2022, pp. 1–6.
- [9] A. A. Ansari and D. Giribabu, "State of art and comprehensive study on smart meter networking," *Flexible Electronics for Electric Vehicles: Select Proceedings of FlexEV—2021*, pp. 377–393, 2022.
- [10] C. L. Athanasiadis, D. I. Doukas, T. A. Papadopoulos, and G. A. Barzegkar-Ntovom, "Real-time non-intrusive load monitoring: A machine-learning approach for home appliance identification," in *2021 IEEE Madrid PowerTech*. IEEE, 2021, pp. 1–6.
- [11] D. de Paiva Penha and A. R. G. Castro, "Home appliance identification for nilm systems based on deep neural networks," *Int. J. Artif. Intell. Appl.*, vol. 9, no. 2, pp. 69–80, 2018.
- [12] L. Halim, R. Barthez, and N. Saputro, "Non-intrusive load monitoring: A cost-effective approach for home appliance identification utilizing machine learning," *Eastern-European Journal of Enterprise Technologies*, vol. 133, no. 8, 2025.
- [13] S. Finster and I. Baumgart, "Privacy-aware smart metering: A survey," *IEEE communications surveys & tutorials*, vol. 17, no. 2, pp. 1088–1101, 2015.
- [14] H. Souri, A. Dhraief, S. Tlili, K. Drira, and A. Belghith, "Smart metering privacy-preserving techniques in a nutshell," *Procedia Computer Science*, vol. 32, pp. 1087–1094, 2014.
- [15] J.-M. Bohli, C. Sorge, and O. Ugus, "A privacy model for smart metering," in *2010 IEEE International Conference on Communications Workshops*. IEEE, 2010, pp. 1–5.
- [16] F. D. Garcia and B. Jacobs, "Privacy-friendly energy-metering via homomorphic encryption," in *Security and Trust Management: 6th International Workshop, STM 2010, Athens, Greece, September 23–24, 2010, Revised Selected Papers*. Springer, 2011, pp. 226–238.
- [17] 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 Transactions on Parallel and Distributed Systems*, vol. 23, no. 9, pp. 1621–1631, 2012.
- [18] A. Molina-Markham, P. Shenoy, K. Fu, E. Cecchet, and D. Irwin, "Private memoirs of a smart meter," in *Proceedings of the 2nd ACM workshop on embedded sensing systems for energy-efficiency in building*, 2010, pp. 61–66.
- [19] M. A. Mustafa, S. Clempunt, A. Aly, and A. Abidin, "A secure and privacy-preserving protocol for smart metering operational data collection," *IEEE Transactions on Smart Grid*, vol. 10, no. 6, pp. 6481–6490, 2019.
- [20] T. Wang, J. Jin, X. Gu, Y. Zhang, and X. Chen, "Preen: An aggregation mechanism for privacy-preserving smart-metering communications," *IEEE Transactions on Consumer Electronics*, 2022.
- [21] H. Wang, Y. Gong, Y. Ding, S. Tang, and Y. Wang, "Privacy-preserving data aggregation with dynamic billing in fog-based smart grid," *Applied Sciences*, vol. 13, no. 2, p. 748, 2023.
- [22] H. Li, X. Li, and Q. Cheng, "A fine-grained privacy protection data aggregation scheme for outsourcing smart grid," *Frontiers of Computer Science*, vol. 17, no. 3, p. 173806, 2023.

- [23] X.-Y. Zhang, S. Kuenzel, J.-R. Córdoba-Pachón, and C. Watkins, “Privacy-functionality trade-off: A privacy-preserving multi-channel smart metering system,” *Energies*, vol. 13, no. 12, p. 3221, 2020.
- [24] B. Vetter, O. Ugus, D. Westhoff, and C. Sorge, “Homomorphic primitives for a privacy-friendly smart metering architecture.” in *SECRYPT*, 2012, pp. 102–112.
- [25] F. Li, B. Luo, and P. Liu, “Secure information aggregation for smart grids using homomorphic encryption,” in *2010 first IEEE international conference on smart grid communications*. IEEE, 2010, pp. 327–332.
- [26] F. G. Márquez, C. Sorge, O. Ugus, and G. M. Pérez, “Do not snoop my habits: preserving privacy in the smart grid,” *IEEE Communications Magazine*, vol. 50, no. 5, pp. 166–172, 2012.
- [27] T. Dimitriou, “Secure and scalable aggregation in the smart grid,” in *2014 6th International Conference on New Technologies, Mobility and Security (NTMS)*. IEEE, 2014, pp. 1–5.
- [28] T. Dimitriou and M. K. Awad, “Secure and scalable aggregation in the smart grid resilient against malicious entities,” *Ad Hoc Networks*, vol. 50, pp. 58–67, 2016.
- [29] M. Nabil, M. Ismail, M. M. Mahmoud, W. Alasmary, and E. Serpedin, “Ppetd: Privacy-preserving electricity theft detection scheme with load monitoring and billing for ami networks,” *Ieee Access*, vol. 7, pp. 96 334–96 348, 2019.
- [30] R. Petric, “A privacy-preserving concept for smart grids,” *Sicherheit in vernetzten Systemen*, vol. 18, pp. B1–B14, 2010.
- [31] C. Efthymiou and G. Kalogridis, “Smart grid privacy via anonymization of smart metering data,” in *2010 first IEEE international conference on smart grid communications*. IEEE, 2010, pp. 238–243.
- [32] S. Finster and I. Baumgart, “Pseudonymous smart metering without a trusted third party,” in *2013 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications*. IEEE, 2013, pp. 1723–1728.
- [33] P. Paillier, “Public-key cryptosystems based on composite degree residuosity classes,” in *International conference on the theory and applications of cryptographic techniques*. Springer, 1999, pp. 223–238.
- [34] S. I. Serengil and A. Ozpinar, “Lightphe: Integrating partially homomorphic encryption into python with extensive cloud environment evaluations,” *arXiv preprint arXiv:2408.05219*, 2024.
- [35] Bashtage, “Randomgen: A random number generator library,” 2025, [Accessed: February 28, 2025]. [Online]. Available: <https://github.com/bashtage/randomgen>
- [36] M. Cenky, J. Bendík, B. Cintula, P. Janiga, Ž. Eleschová, and A. Beláň, “Dataset of 15-minute values of active and reactive power consumption of 1000 households during single year,” *Data in Brief*, vol. 50, p. 109588, 2023.
- [37] K.-H. Chang and B. Mason, “The ieee 802.15. 4g standard for smart metering utility networks,” in *2012 IEEE Third international conference on smart grid communications (SmartGridComm)*. IEEE, 2012, pp. 476–480.
- [38] I. Wagner and D. Eckhoff, “Technical privacy metrics: a systematic survey,” *ACM Computing Surveys (Csur)*, vol. 51, no. 3, pp. 1–38, 2018.