applied sciences

MDPI

Article

A Blockchain-Enabled Distributed Advanced Metering Infrastructure Secure Communication (BC-AMI)

Nahida Islam [1,†], Md. Sazzadur Rahman [1,*,†], Imtiaz Mahmud [2], Md. Nur Amin Sifat [3] and You-Ze Cho [2,*]

[1] Institute of Information Technology, Jahangirnagar University, Savar, Dhaka 1342, Bangladesh; nahida11.islam@gmail.com
[2] School of Electronics Engineering, Kyungpook National University, Daegu 41566, Korea; imtiaz@knu.ac.kr
[3] Department of Computer Science & Engineering, Daffodil International University, Daffodil Smart City 1207, Bangladesh; mdnuraminsifat380@gmail.com
* Correspondence: sazzad@juniv.edu (M.S.R.); yzcho@ee.knu.ac.kr (Y.-Z.C.)
† These authors contributed equally to this work.

Abstract: The world is facing an urgent need to provide secure communication and data access control in advanced metering infrastructure (AMI) because conventional cryptographic key management and authentication protocols are at stake. The cryptography schemes entirely rely on trusted third parties (TTPs), leading to a single point of failure and increasing network overhead. In response to this inefficiency and security compromise, this study proposes a blockchain-enabled distributed AMI secure communication scheme. In the proposed work, smart contract (SC), an integrated part of the blockchain, is programmed to substitute traditional TTP-based transaction systems, which operate in a distributed, immutable, and trustworthy manner. In this paper, we implemented practical Byzantine fault tolerance (PBFT) consensus algorithm and Hyperledger Fabric (HLF) blockchain platform to ensure Byzantine fault tolerance in the blockchain transaction. Performance analysis shows that the proposed BC-AMI scheme has the advantage of incurring the least amount of communication and time costs compared with similar studies while ensuring security against some common cyber-attacks.

Keywords: advanced metering infrastructure secure communication; smart contract; hyperledger fabric; practical byzantine fault tolerance

Citation: Islam, N.; Rahman, M.S.; Mahmud, I.; Sifat, M.N.A.; Cho, Y.-Z. A Blockchain-Enabled Distributed Advanced Metering Infrastructure Secure Communication (BC-AMI). *Appl. Sci.* **2022**, *12*, 7274. https://doi.org/10.3390/app12147274

Academic Editor: Chi-Wai Chow

Received: 9 June 2022
Accepted: 18 July 2022
Published: 20 July 2022

1. Introduction

Smart grid (SG) is a two-way bidirectional intelligent network that regulates electricity generation, transmission, distribution, and consumption in an automated way for the modern power infrastructure. Nearly 200 million smart meters were installed in Europe until 2020 [1]. As of the end of 2021, approximately 23.6 million smart and advanced meters are installed in the UK residential and non-residential buildings, almost having increased by 20 times since 2014 [2]. According to another statistical report, the U.S. smart meter data management market was valued at 176.56 U.S. million in 2018, and it is projected to reach 556.94 U.S. million by 2026, growing at a CAGR of 15.54% between 2019 and 2026 [3]. As per another study, it is expected for SG networking to gain potential growth by 2027, witnessing market growth at a rate of 10.9% in the forecast period from 2020 to 2027 [4]. In this way, this statistical visualization helps us to realize that SG goes beyond customizing today's demand, but also represents the future of electric grids.

Advanced metering infrastructure (AMI), also known as smart meters, is a technology that integrates automatic fault detection with self-healing capabilities and facilitates two-way communication. Smart meters (SMs) collect, analyze, and send electricity usage data via data collectors (DCs) to meter data management system (MDMS) for use in real-time pricing, billing, outage management, leak detection, demand forecasting, etc. [5]. This dynamic process is carried out through public communication channels. Therefore, it is highly susceptible to attacks such as data theft, tampering and modification of usage

reports, impersonation, unauthorized access, the disclosure of customer information, meter compromise, false data injection, etc., that damage infrastructures and steal energy.

Blockchain technology provides a promising solution to overcome these security issues. In a blockchain, users can interact with each other in a verifiable manner without needing a trusted intermediary. All network participants maintain the public ledger instead of central managers. Moreover, blockchain is comparably more robust considering a single point of failure. Thus, being a decentralized, tamper-proof, and trustworthy technology, blockchain has become an incredible innovation for the key management of smart grid systems. It paves the way for privacy-preserving communication between SMs and MDMS too. Especially, the Merkle trees or hash trees, which are well-known for secure and efficient transaction validation and integration, can also be used to provide authentication between SMs and MDMS [6,7]. Therefore, we propose a blockchain-enabled distributed AMI secure communication scheme combining these two technologies to achieve an efficient, secure, and low-cost security protocol in this work. The main contribution of this work is as follows:

- We integrate decentralized blockchain technology in smart meter secure communication to remove the need for conventional trusted authorities;
- We eliminate single point of failure and regulate communication in a flexible, trustworthy, and automated way;
- The proposed scheme takes less time and expends less communication bit than other existing works;
- Finally, by an informal security discussion, we confirm that the proposed scheme is secure enough against the most common types of attacks.

The rest of the paper is organized as follows: Section 2 discusses the research works regarding smart meter secure communication and blockchain-based security frameworks; and Section 3 provides details on blockchain, smart contracts, and the consensus algorithm. Section 4 provides the system overview and a detailed workflow of the proposal. Section 5 includes the implementation and performance analysis of the proposed scheme along with a comparative study. Section 6 concludes the paper.

2. Literature Survey

Recently, a significant amount of work has addressed the unique security and privacy challenges faced by IoT-aided SG. According to George et al. [1], a key management scheme (KMS) is proposed for AMI secure communication that implements the hybridization of Advanced Encryption Standard (AES) and Rivest, Shamir, Adleman (RSA) algorithms. The proposed scheme's execution time is very high without accounting for storage or network overhead. Tsai et al. [8] proposed an identity-based encryption scheme in order to ensure secure key distribution in SG. The system's protocol allows for anonymous access to a smart meter using only a single private key without the help of a trusted anchor. However, it implemented bilinear pairing, which is computationally expensive. In addition, the system is vulnerable to leakage attacks and lacks strong session key security. Uludag et al. [9] suggested a pairwise key distribution scheme among power operators, SMs, and DCs when setting up a secure and scalable smart meter data collection scheme. Even though the proposed scheme seeks to minimize data collection time, it lacks anonymity. Aside from that, the identities are sent in plaintext, opening the door for a variety of cyber-attacks.

Alishahi et al. [10] also presented a key agreement protocol based on elliptic curve cryptography (ECC). This protocol can withstand sybil and reply attacks and needs fewer communication messages. However, no performance analysis of the scheme has been conducted to evaluate the computational and communicational overhead. Later, in the next year, Khasawneh et al. [11] implemented a hybrid scheme using AES and Elliptic Curve Integrated Encryption Scheme (ECIES) algorithms, with the addition of a precomputation of point scalar multiplication. This scheme requires a robust formal security analysis under any widely accepted threat model to prove its security strength than the suggested one. Olelu et al. [12] designed a provably secure authenticated scheme that reduces the computational

overhead both for smart meters and service providers. Despite securing formal security properties, the system is vulnerable to impersonation, MITM, and traceability attacks.

N. Kumar et al. [13] introduced a novel ECC-based secure authentication protocol in SG for preserving demand-response management. The scheme is secure against several known attacks and verified under formal and informal attack analysis. However, the total communication cost is 1376 bits, with a time cost of about 266 ms, which is computationally costly. Later, in the same year, Lili Yan et al. [14] proposed a key agreement framework to decrease the computational overhead of batch authentication using a binary tree. The drawback of this framework is the high computational cost of multiplication and pairing rather than hash operation. Intending to provide a key agreement protocol using two pass-based authentication methods, M. Qi et al. [15], proposed an ECC and Qu-Vanstone (ECQV)-based implicit certificates. Though it provides better communicational and computational costs, it consumes double the time as the original Diffie–Hellman (DH) protocol.

Olivares et al. [16] developed a novel multi-tier blockchain security system using a lightweight proof-of-efficiency (PoEf) algorithm to protect smart meter data. This layer-wise protection is specially designed to mitigate database tampering and claims to be scalable up to 2000 smart nodes. As the implementation is hardware-based, it lacks AMI components interoperability. Zhang et al. [17] designed a secure keyless signature scheme using a decentralized Go Ethereum (Geth) for SG data protection. The decentralization is achieved through PBFT consensus while maintaining transaction accuracy, effectiveness, and efficiency. According to the analysis, the execution of the designed functionalities of each smart meter takes 46.21 ms per meter. Even so, the chosen consensus algorithm needs to be optimized for efficient and reliable message distribution. A flexible and trustworthy data access control scheme for AMI is proposed by Abou et al. [18] to mitigate DDoS attacks and single point of failure. The access permission is controlled by deploying Ethereum's smart contract and warranties the users' pseudonymity, system's flexibility, and secure transaction with a lower gas price. Nevertheless, there has been no analysis of the bit and time costs involved in implementing the proposed scheme.

In 2020, Zhang et al. [19] introduced another smart grid data access control scheme, which supports decentralization with a three-password verification program for information security and integrity. Moreover, it analyzes the signature correctness and safety certificate based on DH difficulties. Despite the system's resistance to key substitution and key generation center (KGC) attacks, the system provides only a theoretically designed model. It does not include a blockchain platform implementation and computational cost analysis. Melo et al. [20] developed a blockchain-based ECDSA public key infrastructure (PKI) smart metering system using Hyperledger Fabric. The infrastructure is designed using smart contracts and legal metrology to measure trust in physical applications. It eliminates the dependencies of TTP, simplifies the signature verification, and provides solid security, including stolen device attack mitigation. Recently, another blockchain-based smart grid protocol was proposed by Zhong et al. [21] to authenticate smart meter's identity and required resource authorization. InterPlanetary File System (IPFS) is implemented using PBFT consensus to exploit the privacy properties of the FISCO BCOS blockchain platform. Furthermore, the system guarantees mutual authentication, message integrity, identification, undeniability, scalability, and security from some well-known heinous cyber-attacks.

3. Background and Preliminaries

3.1. Blockchain

Over the past decade, blockchain has become a buzzword in the cryptocurrency, finance, and industrial world due to its immutable, decentralized, trustworthy, and secure digital transitions. In 2008, blockchain was conceptualized by Satoshi Nakamoto. The very first successful release of the concept of "blockchain" was in 2009, integrated with cryptocurrency applications combining public-key cryptography and the newly developed consensus algorithm, proof-of-work (PoW).

Blockchain is a decentralized and distributed digital ledger that records innumerable information in an encoded format so that the data cannot be tampered with or forged. A blockchain can also be termed as a "chain of blocks", a distributed data structure replicated on numerous chained nodes using a cryptographic hash where each node is called a block. Each block of the chain is connected to its previous blocks, hence the data stored in a specific block is replicated all over the blocks. This replication provides originality of the block information even if a block is imitated. A node in the blockchain accepts a transaction from an outsider if and only if the transaction is validated by other nodes using a consensus algorithm. After that, the new block is generated and appended to the blockchain. A detailed overview is illustrated in Figure 1 [22,23].

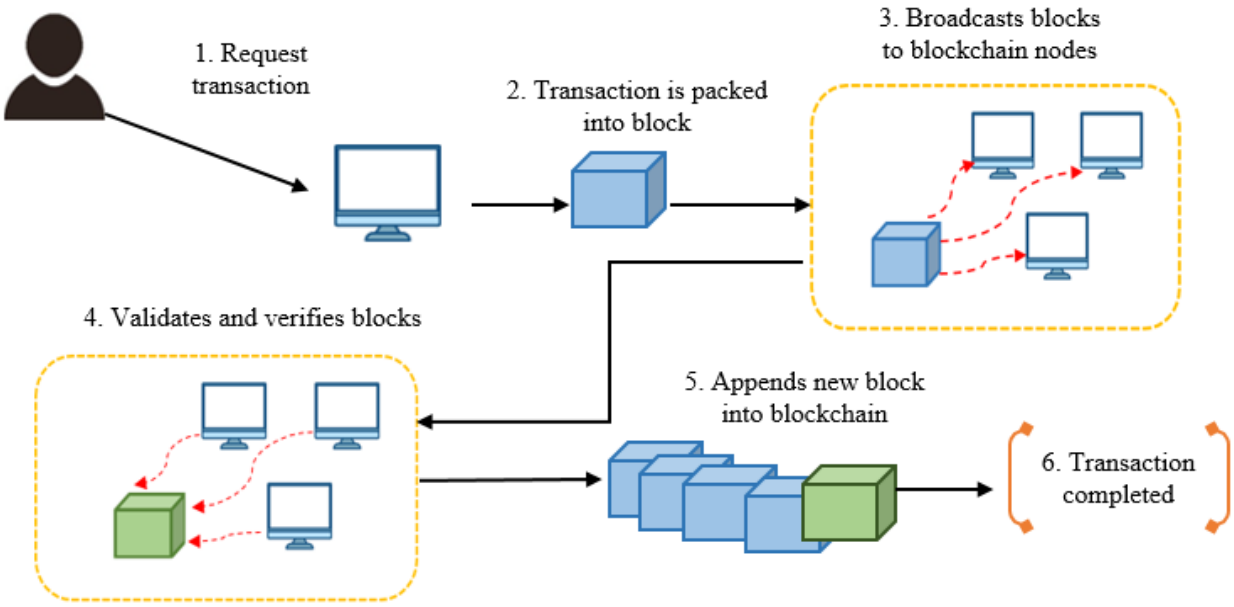

Figure 1. An overview of blockchain transaction process.

Blockchain can be broadly classified into public or permissionless and private or permissioned blockchain. Public or permissionless blockchains are open to all to participate in the verification and consensus process. Any public node can access any transaction without permission and have read or write access (example: Ethereum, Litecoin). In a private or permissioned blockchain, only pre-selected nodes can participate in the verification and consensus process. Only the permissioned nodes have read or write access (example: Hyperledger, R3, Ripple).

3.2. Smart Contract

Smart contracts are small computer programs integrated into hardware or software that can execute triggers, conditions, or any specified logic to enable transactions between users and the blockchain network. Though the concept of smart contracts drew attention in the late nineties, the execution of smart contract platforms without any third-party interference has only been possible after the introduction of blockchain [24]. Figure 2 exemplifies a smart contract's basic operational structure.

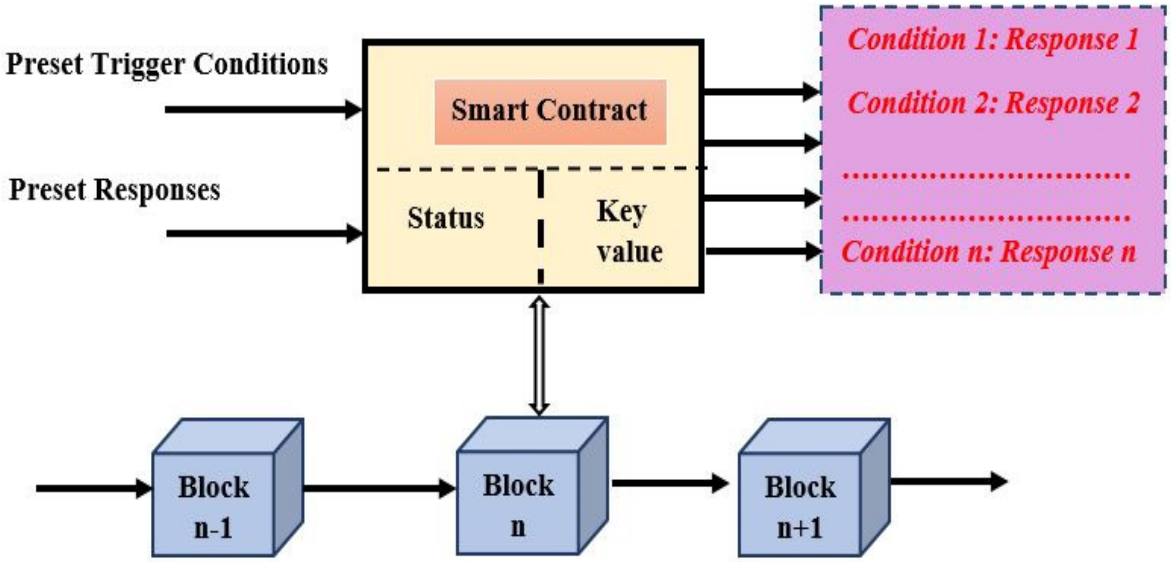

Figure 2. Basic structure of smart contract.

A smart contract is flexible and programmable, executed automatically in a blockchain platform without any third party, if and only the users' conditions and terms are satisfied. Blockchain enables an automated and decentralized platform to execute smart contracts and introduces a secure mechanism where anonymous participants can execute trusted and irreversible transactions between them governed by the rules "if/when . . . then . . . ". Thus, this mechanism is a promising replacement for the conventional transaction procedure requiring trusted third parties. The advantages of blockchain-based smart contracts include immutability, cost-efficiency, self-execution, accuracy, trustworthiness, and inspectability [25].

3.3. Hyperledger Fabric with Practical Byzantine Fault Tolerance

IBM has recently proposed Hyperledger Fabric (HLF) as an open source blockchain platform that is highly scalable and designed for distributing applications on an extensive network. All the other BC technologies rely on the order-execute paradigm, but Fabric introduces an execute-order paradigm, which executes transactions at the first stage before they are finally ordered. This significantly reduces performance overhead and increases confidentiality. Transactions in Fabric are divided into three phases: execution, ordering, and validation. Traditionally, HLF implements reliable, replicated, redundant, and fault-tolerant (Raft) as a consensus algorithm in the ordering service phase to validate the transactions. Though the organizational design of HLF makes it resistant against 51% attack, it is somewhat susceptible to the attack due to the mechanism of Raft [26].

Both Raft and PBFT are leader-based but only Raft is fault-tolerant while PBFT is crash-tolerant. Therefore, Raft does not guarantee the liveliness and security of the network. In a blockchain network, a 50% fraction of the computational power is sufficient to materialize 51% attack. A successful 51% attack can eventually lead to double-spending and DoS attacks. PBFT ensures Byzantine fault tolerance, it has 33% fault tolerance ($^1/_3$ of total network nodes) and hence, it avoids synchronous consumption such as other traditional consensus algorithms (PoW or PoS). Moreover, in PBFT, each node of the network must be pre-validated before participating in transaction validation [27,28]. On that account, we designed our proposed work in the HLF platform using PBFT to ensure network liveliness and prevent 51% attack. A schematic representation of HLF organization integrated with PBFT consensus algorithm is presented in Figure 3.

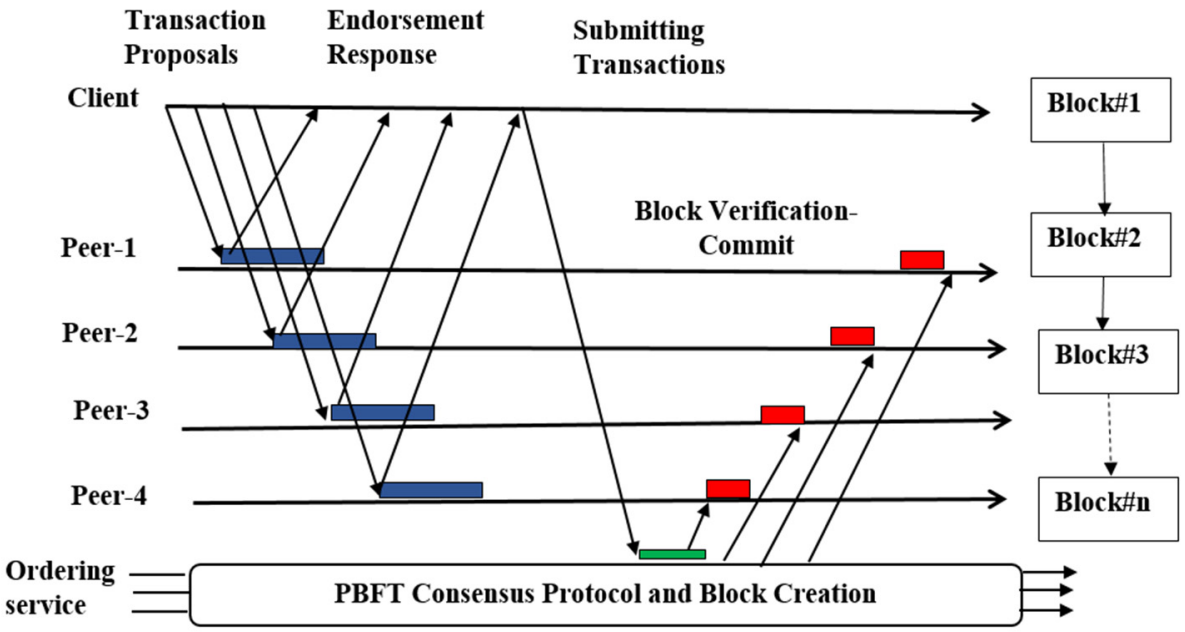

Figure 3. Hyperledger Fabric organization structure integrated with PBFT consensus algorithm.

4. System Methodology

4.1. System Overview

The proposed system model focuses on secure communication between a set of SMs and MDMS using a permissioned blockchain platform. The proposed scheme is composed of the following components: (1) a set of smart meters (SM_i), (2) a set of pre-selected data concentrators (DC_j), (3) Smart Contract (SC), and (4) Meter Data Management System (MDMS). Figure 4 depicts the abstract view of the proposed system model architecture.

SM_i and MDMS represent the ith smart meter and meter data management system in the proposed system model, respectively. These are registered in the permissioned blockchain regulated by DC_j. For permissioned blockchain, we used Hyperledger Fabric (HLF); DC_j is considered as the client nodes, and SC plays the role of registration and key generation instead of any trusted third party or registration authority. DC only relays the SM and MDMS's registration to the deployed SC, and verifies transaction requests to the HLF ordering service. The ordering service follows the PBFT consensus mechanism. The ordering service performs a proper transaction validation. The transactions are integrated into a block and added to the blockchain. After a certain period, the blocks are released to the database, which the pre-registered MDMS retrieves.

4.2. Proposed System Workflow

Figure 5 portrays the proposed system model workflow, and Table 1 indexes the used notations in the proposed work. A brief explanation is as follows:

4.2.1. Phase 1: System Initialization

Preloaded values: $E_p(a, b)$, p, Z_p, G, $H_i(\ )$, ID_{SMi}, ID_{MS}, HID_{SMi}', and HID_{MS}'. Initially, the system initiator initializes the permissioned blockchain which is constructed with the DC_j of AMI infrastructure. As the blockchain is permissioned, the DC_j are pre-selected to ensure decentralization. Before the registration, ID_{SMi} and ID_{MS} are assigned to the corresponding entities, and HID_{SMi} and HID_{MS} are pre-stored in the blockchain.

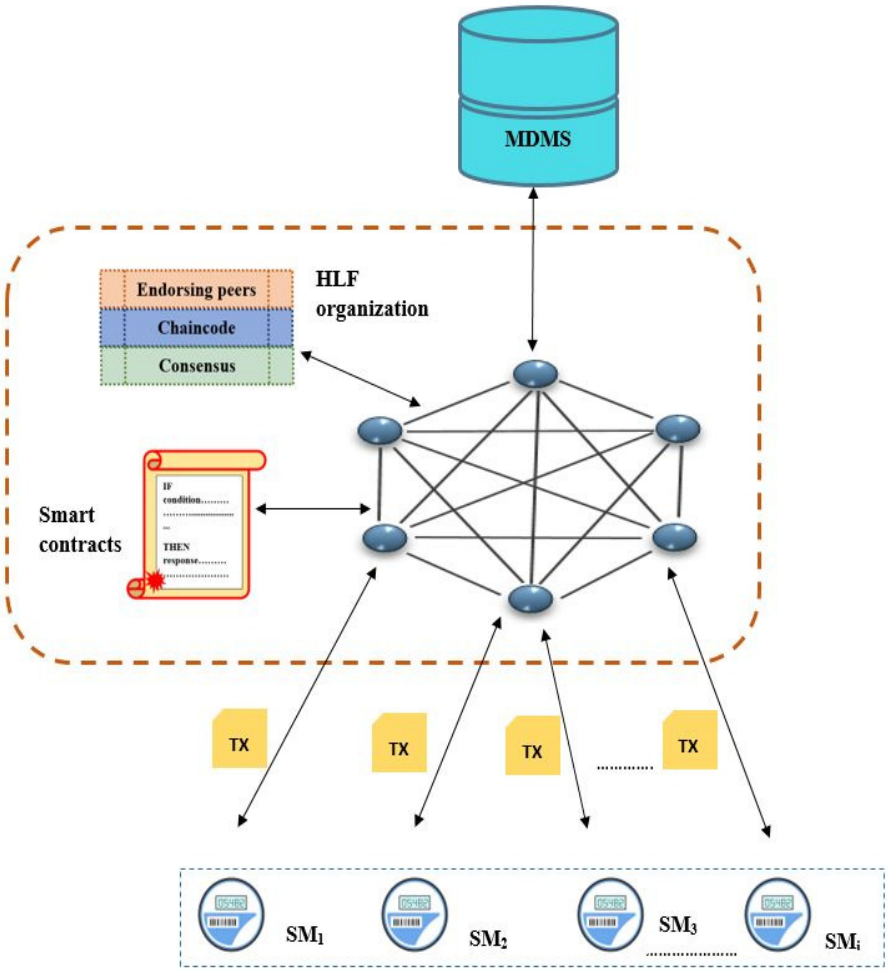

Figure 4. Proposed system model architecture designed with a set of smart meters, smart contract (aka chain code) integrated Hyperledger Fabric blockchain, and MDMS.

Table 1. Index of notations used in this paper and their definition.

Notations	Definition
DB	Blockchain database
SM_i	ith smart meter where $i = 1, 2, 3 \ldots$
DC_j	jth data concentrator where $j = 1, 2, 3 \ldots\ldots$
$E_p(a, b)$	A non-singular elliptic curve $y^2 = x^3 + ax + b \ (mod \ p)$ over finite field Z_p with $4a^3 + 27b^2 \neq 0$
p	A large prime number
Z_p	$0, 1, 2, 3, \ldots . p - 1$, a finite field
G	A base point on $E_p(a, b)$
$x.G$	Point multiplication where $x \in Z_p$
τ	Timestamp
$H_i()$	Collision resistant one-way hash function, where $i = 1, 2, 3 \ldots$
HID	Hashed identity
P_r, P_u	Private and public key pair
Sig_{SMi}	$SM_i's$ ECDSA digital signature
L_{Pr}, L_{Pu}	Length of private and public keys
L_{sign}	Length of ECDSA signature
L_τ	Length of timestamp
T_τ	Timestamp generation time
T_{tx}	Transaction encryption time
T_{sign}	Signature generation time

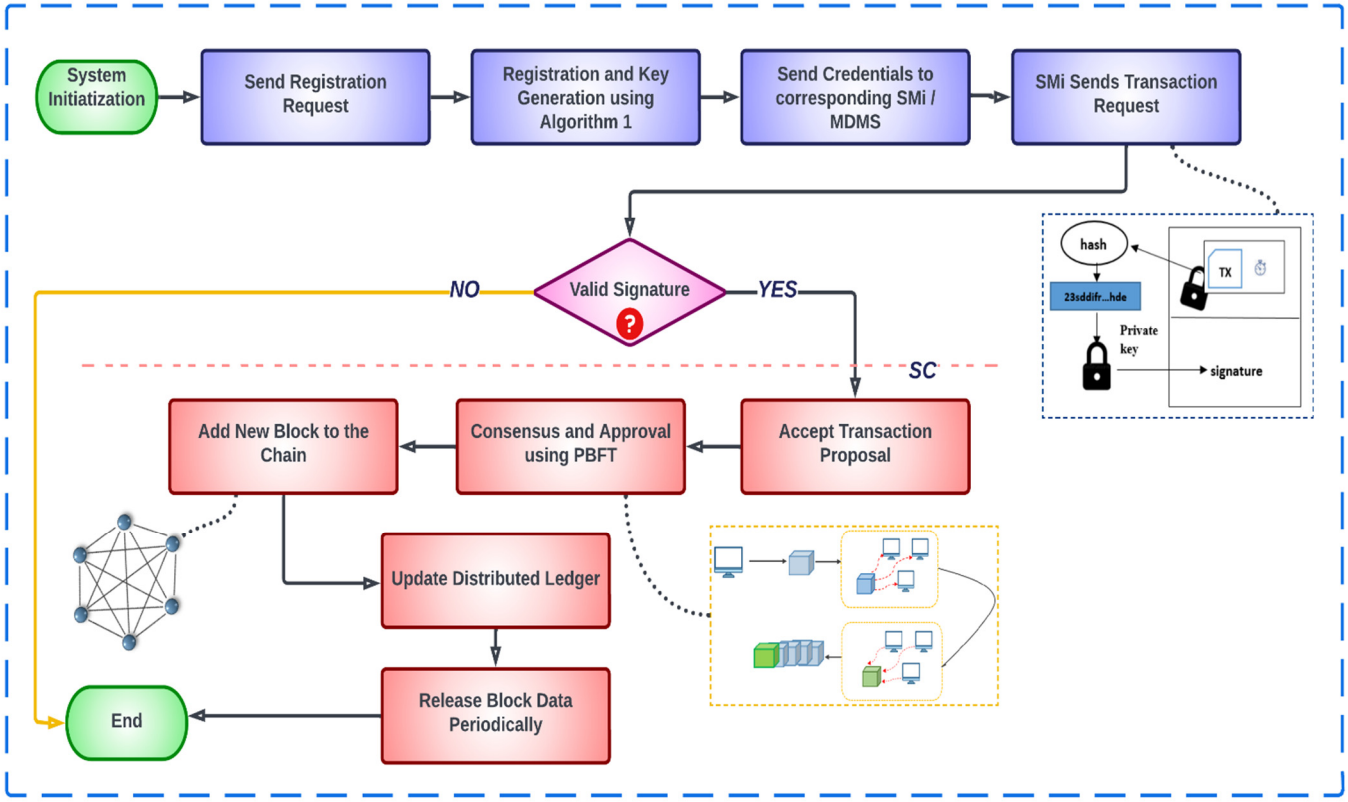

Figure 5. Workflow of proposed system model to illustrate the information transaction and validity from smart meter to permissioned blockchain via smart contract.

4.2.2. Phase 2: Registration and Key Generation

Registration and key generation process is performed as follows:

- Both SM_i and MDMS are registered in the blockchain using their hash identities in this phase. SM_i computes its HID_{SMi} and sends a registration request message to DC_j to obtain its public and private key pair. Here, request message = (HID_{SMi}, reg_req);
- DC_j relays the request message to the smart contract. Smart contract executes a query in the database to verify if $HID_{SMi} = HID_{SMi}'$. If the HID_{SMi} exists, then the registration is executed to generate the corresponding key pair (Pr_{SMi}, Pu_{SMi}). Otherwise, the request is declined. The key pair is derived from ECC curve [29] as described in Algorithm 1;
- The registration process of MDMS is similar to SM_i. After successful registration, the HID_{SMi} is provided to the MDMS along with the key pair (Pr_{MS}, Pu_{MS}).

Algorithm 1: Smart contract-based registration and key generation

Input: HID of SM_i/MDMS
Requires: Pre-stored HID' of SM_i/MDMS in DB
Output: (P_r, P_u) for registered SM_i/MDMS
1: For each SM_i/MDMS do
2: Query for HID' records in DB
3: If $HID == HID'$ then
4: Execute registration () {select random number $x \in Z_p$//using ECC curve
5: $P_r = x$
6: $P_u = x.G$
7: Return (P_r, P_u)}
8: Else
9: Decline

The whole process is conducted using a secure private channel. Before starting each new session, the devices generate a new key pair while they are offline and not interacting with each other. In this state, before taking part in any session, each of them are registered (if already registered, then smart contract checks the pre-registration) and the corresponding generated key pair are distributed over a secure private channel. The generated public keys of SM_i and MDMS are broadcasted throughout the full metering system and blockchain. Hence, anyone can avail these public keys.

4.2.3. Phase 3: Data Transaction and Validation

The data transaction and validation process follow the following steps:

- Create Transaction: Each SM_i collects electricity usage data (UD) and creates an electricity usage report with the timestamp, τ. The formatted usage report is encrypted using the public key of MDMS. The final transaction is created using the encrypted usage report E(UD), its hash, timestamp, HID_{SMi}, and the ECDSA signature of the sender SM_i. Table 2 provides the final transaction format of the collected usage data;
- Transaction Request Validation: DC_j passes the message again to SC after receiving the transaction request message. SC checks the HID_{SMi} to ensure the sender's pre-registration and verifies the signature. If the signature is verified, *DCj* sends the encrypted message to the peers for endorsement. Peers respond to the proposal, append signature as an endorsement, and send it back to the DC_j. After receiving the response, DC_j sends the encrypted transaction to the ordering service with the peer endorsement. The ordering service validates the transaction using PBFT consensus algorithm;
- In the ordering phase, after receiving a transaction request, the leader node generates a new block with the transaction information, which is considered a candidate block. For verification as well as auditing, the leader node broadcasts the block to other nodes. Nodes audit the block data after receiving it and broadcast the results with a hash to the rest of the network. Each node compares the audit results with the others. Nodes reach a consensus on the candidate block and send the audit and comparison results back to the leader. If the leader obtains 2f + 1 responses, then the candidate block is finalized and appended as a new block in the blockchain.

Table 2. Transaction format of smart meter's electricity usage data.

Transaction Header
Encrypted usage report, E(UD) = {Pr_{SMi} (UD || τ)};
Hash of E(UD);
Sender's hash identity, HID_{SMi};
ECDSA signature, Sig_{SMi};
Timestamp, τ;

4.2.4. Phase 4: Release Block Data

In the blockchain network, the nodes keep a distributed ledger in which meters' public keys and hashed identities are recorded. Each data entry in the ledger represents the association between meters and their hash values. In addition, nodes on the blockchain replicate the ledger among themselves so that any node can verify if a public key belongs to a specific meter. With the continuous transaction, the shared ledger becomes progressively larger. Hence, after a specific period, the block data are transferred to the MDMS along with the corresponding HID_{SMi}. MDMS verifies HID_{SMi} and decrypts it using its private key Pr_{MS} to obtain the original meter data.

5. Implementation and Performance Analysis

The purpose of this section is to assess the effectiveness and improvements of the proposed blockchain-enabled distributed AMI secure communication scheme, in comparison with other cryptography and blockchain-based studies. The performance is analyzed numerically by computing the communication cost (bits) and the time cost (ms). Section 5.3

discusses in detail the estimating of these two parameters, which are essential to determining whether the proposed system is lightweight or not. Additionally, a discussion of some informal security properties was also provided to assess the proposed system's security risk.

5.1. Experimental Environment

We conducted the experiments on an HP ProBook 440G7 with 1.60 GHz Intel Core i5-102100U processor, Intel HD UHD Graphics, 8 GB RAM, and Windows 10 operating system. We set up the blockchain platform HyperLedger Fabric using Docker Engine and Docker Compose. The smart contract code snippets are written using JavaScript programming language in Visual Studio Code.

5.2. Dataset Description

We used the "Smart Meter in London" dataset from Kaggle [30] to evaluate the performance. This dataset contains electricity usage records collected from 5567 households in London from November 2011 to February 2014.

5.3. Communication and Time Cost Analysis

5.3.1. Communication Cost

Each smart meter only stores its corresponding asymmetric ECC key pairs, public key of MDMS, and generates ECDSA signature in the proposed system. For this purpose, the secp256k1 curve was chosen to produce ECC keys, which have a length of the 256-bit private key (L_{Pr}) and 257-bit public key (L_{Pu}). Similarly, the generated ECDSA signature has a size of 512-bits (L_{sign}) [31]. Moreover, a timestamp of 64-bits ($L\tau$) was also appended while performing the corresponding operations. Therefore, the total communication bit cost (Bit_{tot}) for each smart meter is:

$$Bit_{tot} = L_{sign} + (L_{PuSMi} + L_{PrSMi}) + L_{PuMS} + L\tau$$

$$= [512 + (257 + 256) + 257 + 64] \text{ bits}$$

$$= 1346 \text{ bits } (\equiv 0.16825 \text{ Kb})$$

Generally, a smart meter has a maximum storage of 3 Kb [17], where the proposed scheme requires only 0.168 Kb, which is acceptable.

5.3.2. Time Cost

To analyze the time cost of each smart meter, from the collected dataset archive, "halfhourly_dataset" was chosen. This file has 112 blocks of electricity usage data of 112 unique smart meters. Each of these blocks contains a total record of 10,48,576 rows and three columns, namely, LCLid (meter ID), tstp (timestamp), and energy (measured in kWh/hh). For convenience, we took the first five blocks: Block1, Block2, Block3, Block4, and Block5 for the experiment, which collected a total record of 424,552 kWh, 365,857 kWh 252,237.9 kWh, 244,846.1 kwH, and 290,359.9 kWh, respectively, over a period of 30 min.

In our proposed system, each smart meter needs to perform only three operations, namely, timestamp generation ($T\tau$), transaction encryption (T_{tx}), and signature generation (T_{sign}). The time required to perform these operations are measured in milliseconds (ms). Table 3 lists the individual time of each SM for these operations and Figure 6 illustrates the graphical view; therefore, as per the calculation, the total computational time cost for SM1, SM2, SM3, SM4, and SM5 are 20.59 ms, 8.02 ms, 13.34 ms, 13.52 ms, and 8.96 ms, respectively.

Table 3. Time cost analysis of each smart meter for half hourly energy consumption records.

Cryptography Operations	SM1	SM2	SM3	SM4	SM5
Timestamp generation ($T\tau$)	0.02	0.01	0.01	0.01	0.01
Transaction encryption (T_{tx})	13.18	4.38	5.83	5.79	4.01
Signature generation (T_{sign})	7.39	3.63	7.50	7.72	4.94
Total time cost for each SM (ms)	20.59	8.023	13.34	13.52	8.96

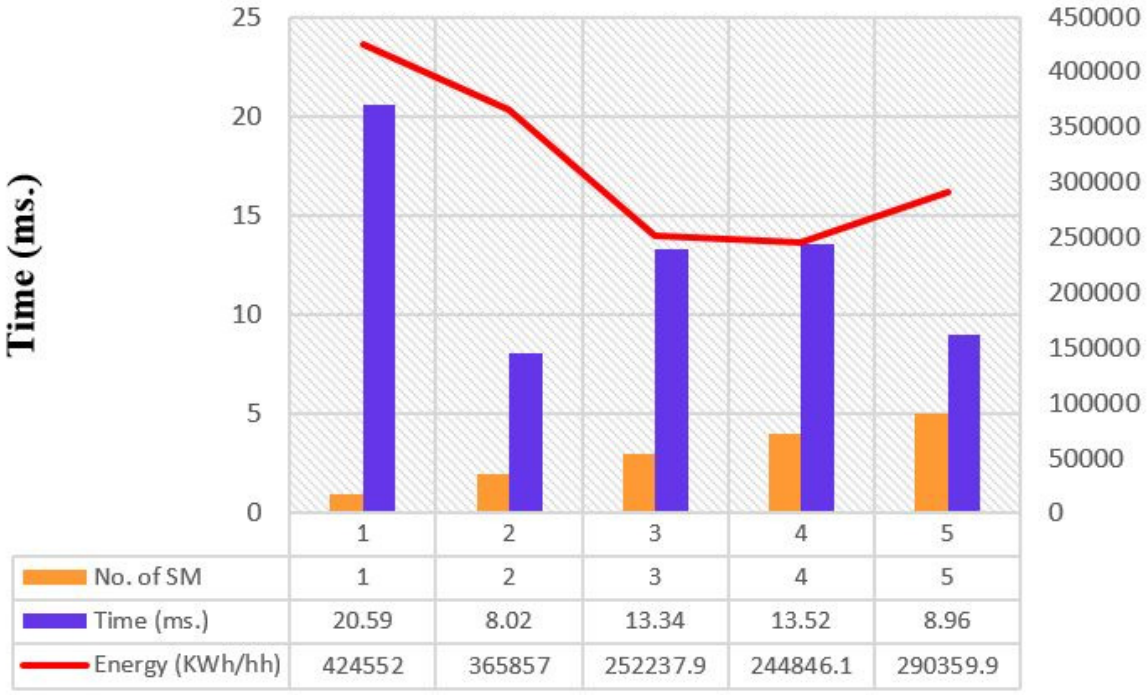

Figure 6. Time cost analysis of each smart meter for half hourly energy consumption records.

5.4. Informal Security Discussion

5.4.1. Perfect Forward Secrecy (PFS) and Key Replacement Attack

In the proposed system, the key pair is generated using ECC curves, and the strength of these curves is based on the computational difficulty of the ECDLP problem. Moreover, these keys have an ephemeral property that guarantees the auto-refreshing of keys (i.e., new key pair is generated for each session). As a result, PFS is ensured, and the key replacement attack is prevented.

5.4.2. Message Integrity, Authentication, and Signature Correctness

To obtain the key pair to conduct a transaction requires the verification of SM_i's hash ID, which is irreversible; therefore, if the recovered public key does not match the provided one, the signature verification fails, which means that the encrypted transaction content was tampered with or the signature was altered. Consequently, as SC of blockchain checks these criteria before accepting the transaction proposal; thus message integrity, authentication, and signature correctness are ensured.

5.4.3. Man-in-the-Middle (MITM), Impersonation, and Replay Attack

Smart meters are registered using their hash ID, which prevents impersonation attacks. Timestamps avoid replay attacks. The public key is derived from point multiplication of the private key by using an efficient elliptic curve which prevents an adversary from discovering the computed private key for a MITM attack.

5.4.4. Byzantine Fault Tolerance and 51% Attack

The proposed system is designed using PBFT consensus based on the HLF platform, where nodes are pre-selected. PBFT only validates a transaction if it obtains >2f+1 responses, where f is the number of faulty nodes. Therefore, it has low Byzantine fault tolerance and prevents 51% attack.

5.5. Comparative Study

From the summary of the above discussion and based on previous studies and performance analysis, we can outline that the main novelty of this research work is the concept of security without a trusted third party (TTP), which is a potential vulnerability for cryptography-based smart meters. Blockchain provides the most effective security solution in a decentralized way. In our proposed scheme, SC performs all the functionalities of TTP in a secure blockchain environment.

In Hyperledger Fabric, Certificate Authority (CA) authenticates the participants, an internal functionality fully handled and secured by blockchain mechanism. It ensures no interference from any outsider such as TTP or trusted anchor. In the proposed system, the smart meters are authenticated via smart contracts (an integrated part of the blockchain), and the transactions from the authenticated smart meters are validated and stored by the blockchain. In the traditional cryptography-based security systems, these responsibilities are performed by a trusted assumed third party or CA, which was replaced in the proposed work by blockchain with the integration of smart contract. Therefore, the proposed solution can be outlined as a trusted third party-free solution.

As shown in Figures 7 and 8, the proposed scheme's total time and communication cost is lower than that of the others. Tsai et al. [8], Uludag et al. [9], Odelu et al. [12], and Kumar et al. [13] proposed cryptography-based AMI security models required a time cost of 55.98 ms, 158.06 ms, 32.16 ms, and 266 ms, respectively, and communication cost of 1376 bits, 2240 bits, 6144 bits, 1538 bits, and 1376 bits, respectively. In contrast, our proposed blockchain-based scheme incurs the least time of 20.59 ms and a communication cost of 1346 bits.

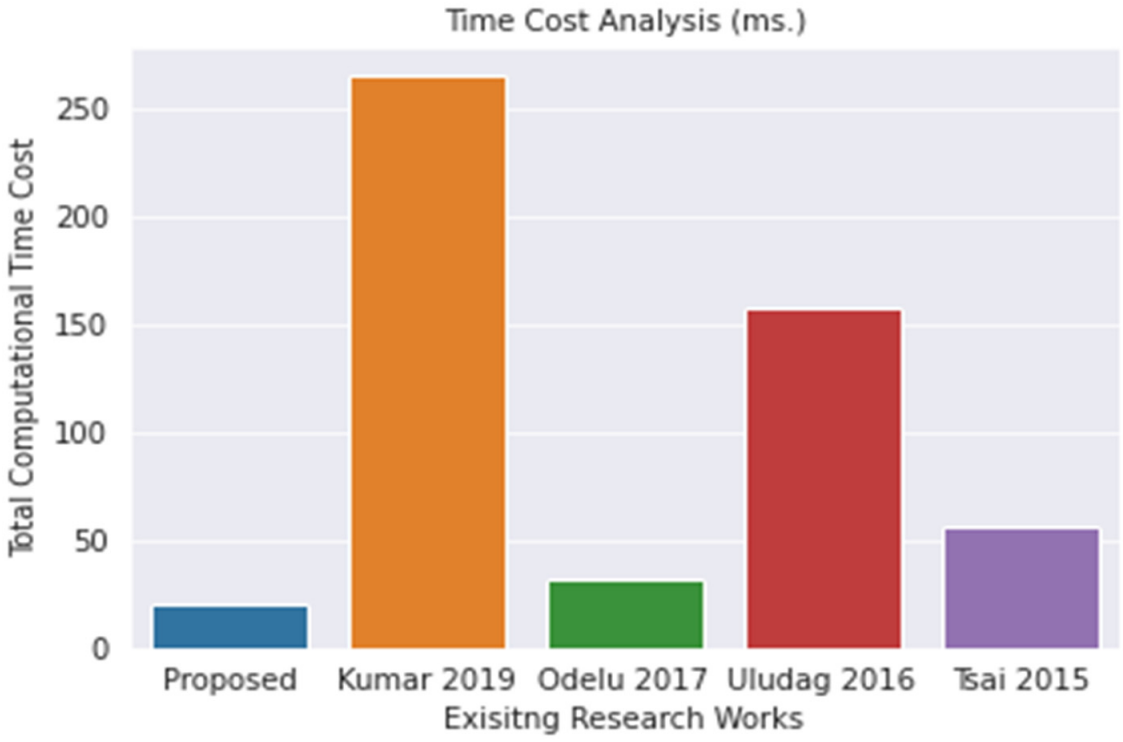

Figure 7. Comparative performance analysis between proposed and existing works in terms of time cost analysis [8,9,12,13].

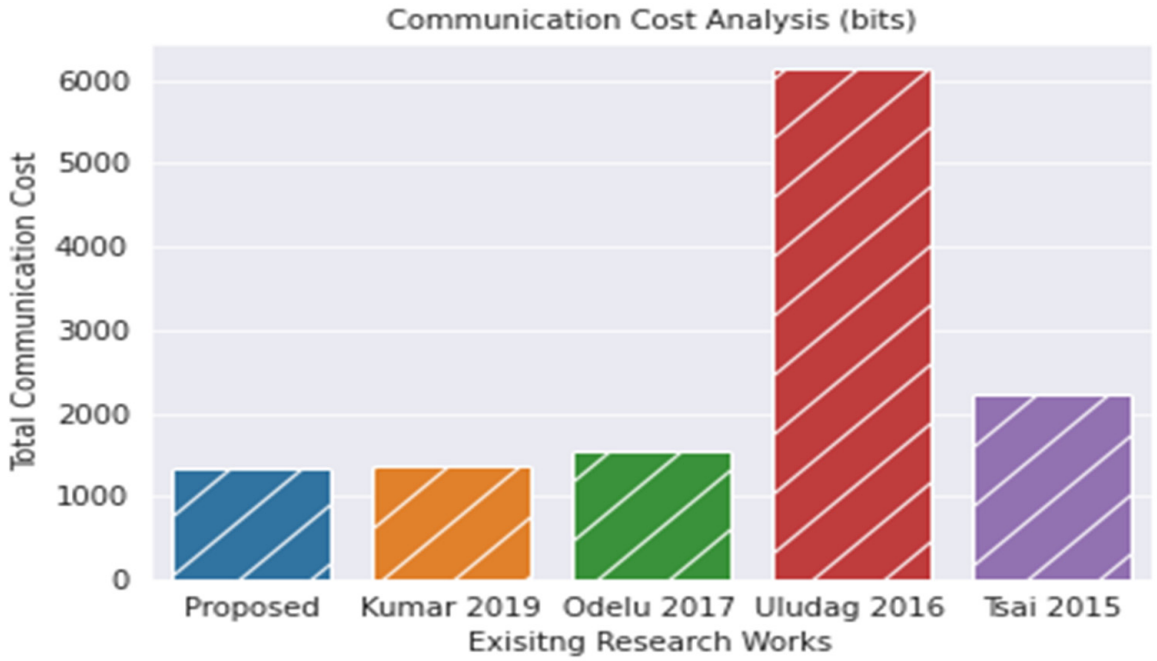

Figure 8. Comparative performance analysis between proposed and existing works in terms of communication cost analysis [8,9,12,13].

After the progression of blockchain, some other works in smart metering security have designed only theoretical frameworks under some informal security features analysis. Zhang et al. [19] and Melo et al. [20] both designed TTP-free smart meter security schemes that include informal security analysis but their implementation, required time, or cost analysis is unknown. Table 4 compares the informal security features between proposed and existing works. The security protocols designed by Zhang et al. [19] and Melo et al. [20] do not ensure PFS feature, therefore the systems are vulnerable to session key replacement attack and neglect the auto-refresh of the key. Moreover, the consensus algorithms implemented by these security protocols do not have BFT property, and eventually it leads to 51% attack in their proposed blockchain platform. With comparison to these studies, our proposed BC-AMI scheme ensures PFS and BFT security properties to prevent key replacement and 51% attacks, respectively.

Table 4. Comparison based on informal security features between proposed and existing works.

Study	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
[19]	×	√	√	√	√	√	√	√	×	×
[20]	×	√	√	√	√	√	√	√	×	×
Proposed	√	√	√	√	√	√	√	√	√	√

Legend: F1: PFS; F2: key replacement attack; F3: message integrity; F4: authentication; F5: signature correctness; F6: MITM; F7: impersonation; F8: replay; F9: BFT; F10: 51% attack.

6. Conclusions

In this work, we address the problem of conventional TTP-based cryptography AMI security models that eventually lead to single point of failure with an increasing amount of network overhead. We deprecate the role of traditional TTP with smart contract functionalities based on blockchain implemented on the HLF platform. This platform implements PBFT as their ordering service to avoid 51% attacks and ensures network fault tolerance. According to the informal security discussion, the proposed scheme has perfect forward secrecy, message integrity, authentication, signature correctness properties, resistance to key replacement, MITM, impersonation, and replay attacks. Furthermore, the proposed scheme takes the least time and communication cost among the compared schemes.

The proposed BC-AMI security scheme was designed to concentrate on permissioned consortium blockchain platforms. Since the consortium blockchain implementation has not yet been appropriately documented, we implemented the proposed scheme using a private HLF blockchain platform. The consortium implemented blockchain platform is currently in development. Thus, the future scope of this research work lies in the proper implementation of consortium blockchain to analyze the security performance. Furthermore, our proposed scheme is limited only to informal discussions of security features. In the future, rigorous and formal security analysis using an appropriate threat model and verification tool will be performed.

Author Contributions: Conceptualization, M.S.R.; Data curation, N.I.; Formal analysis, N.I., M.S.R. and M.N.A.S.; Funding acquisition, Y.-Z.C.; Methodology, N.I. and M.S.R.; Resources, I.M.; Supervision, M.S.R.; Validation, N.I.; Visualization, M.N.A.S.; Writing—original draft, N.I. and M.S.R.; Writing—review & editing, I.M. and Y.-Z.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported in part by the Basic Science Research Program through the National Research Foundation of Korea (NRF), funded by the Ministry of Education (No. NRF-2018R1A6A1A03025109), and by the National Research Foundation of Korea (NRF) grant funded by the Korean government (No. NRF-2019R1A2C1006249).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare that there is no conflict of interest in this paper.

References

1. Ghosal, A.; Conti, M. Key Management Systems for Smart Grid Advanced Metering Infrastructure: A Survey. *IEEE Commun. Surv. Tutor.* **2019**, *21*, 2831–2848. [CrossRef]
2. Alves, B. Domestic Electricity Meter Types in Operation Great Britain. Available online: https://www.statista.com/statistics/791967/domestic-electricity-meter-types-in-operation-great-britain/ (accessed on 5 June 2022).
3. U.S. Smart Meter Data Management Market. Available online: https://www.alliedmarketresearch.com/us-smart-meter-data-management-market (accessed on 5 June 2022).
4. Global Smart Electricity Meters Market Trends Report. 2020. Available online: https://www.grandviewresearch.com/industry-analysis/smart-meters-market (accessed on 7 December 2021).
5. Islam, N.; Sultana, I.; Rahman, M.S. HKMS-AMI: A Hybrid Key Management Scheme for AMI Secure Communication. In Proceedings of the International Conference on Trends in Computational and Cognitive Engineering, Online, 21–22 October 2021; pp. 383–392.
6. Merkle Charles, R. *Secrecy, Authentication, and Public Key Systems*; Stanford University: Stanford, CA, USA, 1979.
7. Haber, S.; Stornetta, W.S. How to time-stamp a digital document. In Proceedings of the Conference on the Theory and Application of Cryptography, Aarhus, Denmark, 21–24 May 1990; pp. 437–455.
8. Tsai, J.L.; Lo, N.W. Secure anonymous key distribution scheme for smart grid. *IEEE Trans. Smart Grid* **2015**, *7*, 906–914. [CrossRef]
9. Uludag, S.; Lui, K.-S.; Ren, W.; Nahrstedt, K. Secure and scalable data collection with time minimization in the smart grid. *IEEE Trans. Smart Grid* **2016**, *7*, 43–54. [CrossRef]
10. Alishahi, M.; Farhadi, M.; Jafari, S.; Taghavi, M.; Moosavi, H.; Mohajerzadeh, A. An efficient and light asymmetric cryptography to secure communication in smart grid. In Proceedings of the 2017 IEEE International Conference on Smart Energy Grid Engineering (SEGE), Oshawa, ON, Canada, 14–17 August 2017; pp. 248–252.
11. Khasawneh, S.; Kadoch, M. Hybrid Cryptography Algorithm with Precomputation for Advanced Metering Infrastructure Networks. *Mob. Netw. Appl.* **2018**, *23*, 982–993. [CrossRef]
12. Odelu, V.; Das, A.K.; Kumari, S.; Huang, X.; Wazid, M. Provably secure authenticated key agreement scheme for distributed mobile cloud computing services. *Future Gener. Comput. Syst.* **2017**, *68*, 74–88. [CrossRef]
13. Kumar, N.; Aujla, G.S.; Das, A.K.; Conti, M. ECCAuth: A Secure Authentication Protocol for Demand Response Management in a Smart Grid System. *IEEE Trans. Ind. Inform.* **2019**, *15*, 6572–6582. [CrossRef]
14. Yan, L.; Chang, Y.; Zhang, S. An efficiency batch authentication scheme for smart grid using binary authentication tree. *Int. Arab J. Inf. Technol.* **2019**, *16*, 435–441.
15. Qi, M.; Chen, J. Two-Pass Privacy Preserving Authenticated Key Agreement Scheme for Smart Grid. *IEEE Syst. J.* **2020**, *15*, 3201–32077. [CrossRef]

16. Olivares-Rojas, J.C.; Reyes-Archundia, E.; Gutiérrez-Gnecchi, J.A.; Cerda-Jacobo, J.; González-Murueta, J.W. A novel multitier blockchain architecture to protect data in smart metering systems. *IEEE Trans. Eng. Manag.* **2019**, *67*, 1271–1284. [CrossRef]
17. Zhang, H.; Wang, J.; Ding, Y. Blockchain-based decentralized and secure keyless signature scheme for smart grid. *Energy* **2019**, *180*, 955–967. [CrossRef]
18. Abou El Houda, Z.; Hafid, A.; Khoukhi, L. Blockchain meets AMI: Towards secure advanced metering infrastructures. In Proceedings of the ICC 2020-2020 IEEE International Conference on Communications (ICC), Dublin, Ireland, 7–11 June 2020.
19. Zhang, L.; Li, J.; Hu, F.; Huang, Y.; Bai, J. Smart grid data access control scheme based on blockchain. *Comput. Intell.* **2020**, *36*, 1773–1784. [CrossRef]
20. Melo, W.; Machado, R.C.S.; Peters, D.; Moni, M. Public-Key Infrastructure for Smart Meters using Blockchains. In Proceedings of the 2020 IEEE International Workshop on Metrology for Industry 4.0 & IoT, Roma, Italy, 3–5 June 2020.
21. Zhong, Y.; Zhou, M.; Li, J.; Chen, J.; Liu, Y.; Zhao, Y.; Hu, M. Distributed Blockchain-Based Authentication and Authorization Protocol for Smart Grid. *Wirel. Commun. Mob. Comput.* **2021**, *2021*, 5560621. [CrossRef]
22. Aggarwal, S.; Chaudhary, R.; Aujla, G.S.; Kumar, N.; Choo, K.K.R.; Zomaya, A.Y. Blockchain for smart communities: Applications, challenges and opportunities. *J. Netw. Comput. Appl.* **2019**, *144*, 13–48. [CrossRef]
23. Liu, Z.; Luong, N.C.; Wang, W.; Niyato, D.; Wang, P.; Liang, Y.-C.; Kim, D.I. A survey on applications of game theory in blockchain. *arXiv* **2019**, arXiv:1902.10865.
24. Wang, S.; Ouyang, L.; Yuan, Y.; Ni, X.; Han, X.; Wang, F.Y. Blockchain-enabled smart contracts: Architecture, applications, and future trends. *IEEE Trans. Syst. Man Cybern. Syst.* **2019**, *49*, 2266–2277. [CrossRef]
25. Gamage, H.T.M.; Weerasinghe, H.D.; Dias, N.G.J. Survey on blockchain technology concepts, applications, and issues. *SN Comput. Sci.* **2020**, *1*, 114. [CrossRef]
26. Graf, M.; Küsters, R.; Rausch, D. Accountability in a permissioned blockchain: Formal analysis of hyperledger fabric. In Proceedings of the 2020 IEEE European Symposium on Security and Privacy (EuroS&P), Genoa, Italy, 7–11 September 2020.
27. Saad, M.; Spaulding, J.; Njilla, L.; Kamhoua, C.; Shetty, S.; Nyang, D.; Mohaisen, D. Exploring the attack surface of blockchain: A comprehensive survey. *IEEE Commun. Surv. Tutor.* **2020**, *22*, 1977–2008. [CrossRef]
28. Zhang, S.; Lee, J.-H. Analysis of the main consensus protocols of blockchain. *ICT Express* **2020**, *6*, 93–97. [CrossRef]
29. Nino, C.A.L.; Perez, A.D.; Sandoval, M.M. Elliptic curve lightweight cryptography: A survey. *IEEE Access* **2018**, *6*, 72514–72550. [CrossRef]
30. Smart Meters in London. Available online: https://www.kaggle.com/datasets/jeanmidev/smart-meters-in-london/discussion?select=halfhourly_dataset.zip (accessed on 15 February 2022).
31. NIST Special Publication 1108r3. Available online: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.1108r3.pdf (accessed on 7 November 2021).