Palestine Polytechnic University
Deanship of Graduate Studies and Scientific Research
Master of Informatics

Blockchained-IoT Smart Metering System

Submitted by:

Mariam Bader AlHawamdeh

Supervised by:

Dr. Radwan Tahboub

Thesis submitted in partial fulfillment of requirements of the degree of Master of Science in Informatics
April, 2024

The undersigned hereby certify that they have read, examined, and recommended to the Deanship of Graduate Studies and Scientific Research at Palestine Polytechnic University the approval of a thesis entitled: **Blockchained-IoT Smart Metering System**, submitted by **Mariam B. AlHawamdeh** in partial fulfillment of the requirements for the degree of Master in Informatics.

Graduate Advisory Committee:

Dr. Radwan Tahboub (Supervisor), Palestine Polytechnic University.

Signature:__________________________ Date:__________________

Dr. Liana Tamimi (Internal committee member), Palestine Polytechnic University.

Signature:__________________________ Date:__________________

Dr. Isam Ishaq (External committee member), Al-Quds University.

Signature:__________________________ Date:__________________

Thesis Approved

Dr. Nafez Naser Aldeen
Dean of Graduate Studies and Scientific Research
Palestine Polytechnic University

Signature:__________________________ Date:__________________

DECLARATION

I declare that the Master Thesis entitled **”Blockchained IoT Smart Metering System”** is my original work, and hereby certify that unless stated, all work contained within this thesis is my own independent research and has not been submitted for the award of any other degree at any institution, except where due acknowledgment is made in the text.

Mariam B. AlHawamdeh

Signature:________________________ Date:________________

STATEMENT OF PERMISSION TO USE

In presenting this thesis in partial fulfillment of the requirements for the master's degree in Informatics at Palestine Polytechnic University, I agree that the library shall make it available to borrowers under the rules of the library. Brief quotations from this thesis are allowable without special permission, provided that accurate acknowledgment of the source is made. Permission for extensive quotation from, reproduction, or publication of this thesis may be granted by my main supervisor, or in his absence, by the Dean of Graduate Studies and Scientific Research when, in the opinion of either, the proposed use of the material is for scholarly purposes. Any copying or use of the material in this thesis for financial gain shall not be allowed without my written permission.

Mariam B. AlHawamdeh

Signature:________________________ Date:________________

الخلاصة

نظام قياس العدادات الذكي بتقنيتي إنترنت الأشياء و سلاسل الكتل

لقد أدى نظام القياس الذكي، وهو جزء لا يتجزأ من الشبكات الذكية، إلى إحداث تحول في مراقبة وإدارة استهلاك الطاقة من خلال تمكين جمع البيانات وتحليلها في الوقت الفعلي. تشير آلية القياس الذكي إلى استخدام عداد ذكي ببنية محددة ومصممة مسبقًا لقياس وجمع وتحليل استهلاك الطاقة للمشتركين في الوقت الفعلي. يحتوي نظام القياس الذكي على العديد من المكونات المترابطة لتنفيذ التطبيق المطلوب. المكون الأساسي للنظام هو العداد الذكي المرتبط بالمشتركين المستهلكين للطاقة والمتصل بقاعدة بيانات مزود الطاقة، مما يتيح للنظام قياس ومراقبة استهلاك الطاقة عن بعد. ومع ذلك، فإن الطبيعة الحساسة للبيانات التي تم جمعها، مثل كمية الاستهلاك والبيانات التعريفية الخاصة بالمشترك، تتطلب إجراءات مراقبة وأمان صارمة.

في هذه الرسالة، اقترحنا بنية جديدة لنظام القياس الذكي تعطي الأولوية للمراقبة في الوقت الفعلي، واكتشاف فقدان الطاقة، والحفاظ على الخصوصية، من خلال الاستفادة من تقنيتي سلاسل الكتل وإنترنت الأشياء. وذلك من خلال تعزيز المراقبة في الوقت الفعلي. لا يضمن النظام المقترح أمانًا قويًا فحسب، بل يسهل أيضًا

اكتشاف فقدان الطاقة داخل النظام. تم تخفيف نقاط الضعف في أنظمة القياس التقليدية من خلال تكامل الطبيعة اللامركزية لتقنية سلاسل الكتل وميزات الاتصال بإنترنت الأشياء، مما أدى إلى تعزيز الموثوقية والكفاءة الشاملة للنظام. تؤكد هذه الرسالة أيضًا على الدور الحاسم للمراقبة والإدارة في الوقت الحقيقي في ضمان سلامة البيانات وأمن الشبكة لأنظمة القياس الذكية. بالإضافة إلى ذلك، فإنها تقترح تقنية للكشف عن فقدان الطاقة على مستويين اثنين لزيادة توفير الطاقة والتكلفة. لقد قمنا بتنفيذ النظام المقترح باستخدام لغة برمجة Python و Ethereum Blockchain بالإضافة إلى تصميم العقود الذكية الخاصة بتقنية Blockchain من خلال لغة برمجة Solidity . تم محاكاة النظام المقترح في سيناريوهين لقياس الأداء من حيث زمن الوصول ونسبة وصول الحزمة و المساحة المستهلكة للتخزين. بالإضافة إلى ذلك، تم توفير تحليل رقمي للبيانات لإظهار توفير الطاقة والتكلفة للنظام المقترح. أظهرت نتائج المحاكاة أن النظام المقترح يوفر أداء عالي مع توفير الطاقة والتكاليف من خلال تقنية الكشف عن فقدان الطاقة.

Abstract

The Smart Metering System, an integral component of Smart Grids, has transformed the monitoring and management of energy consumption by enabling real-time data collection and analysis. It refers to the usage of a smart meter with a pre-defined and designed architecture to measure, collect, and analyze the energy consumption of customers in real-time. The smart metering system contains some components that are connected to provide the desired application. The basic component is the smart meter that is related to energy customers and connected by a gateway to the utility supplier database. Which allows system measuring and monitoring of energy consumption remotely. However, the sensitive nature of the data collected, like customer energy usage and private data, necessitates stringent monitoring and security measures. In this thesis, we proposed a new Smart Metering System Architecture that prioritizes real-time monitoring, energy loss detection, and privacy-preserving, leveraging Blockchain and IoT technologies. By enhancing real-time monitoring, the proposed system not only ensures robust security but also facilitates energy loss detection within the system. Through the integration of Blockchain's decentralized nature, and IoT connectivity features, Traditional Smart Metering Systems' inherent vulnerabilities have been mitigated, enhancing the overall reliability and efficiency of the system. This thesis emphasizes the critical role of real-time monitoring and management in ensuring data integrity and network security of Smart Metering Systems. Additionally, it provides a technique to detect energy losses on two levels to increase energy and cost savings. We implemented the proposed system using Python, Ethereum Blockchain, and Solidity for Blockchain smart contracts. The proposed system is simulated in two scenarios to measure the performance in terms of latency packet delivery ratio and storage consumption overhead. In addition, a numeric data analysis is provided to show the energy and cost savings of the proposed system. The simulation results showed that the proposed system provides high performance while saving energy and costs through energy loss detection technology.

DEDICATION

To my family.

For your endless and unconditional love, support, care, and encouragement.

You were and still are my special, indispensable people.

My Great Parents,

My Gentle Husband,

My Lovely Sisters and Brothers,

And my coming Baby.

Dears, I love you forever, nothing can be done without your support.

To my beloved, patient Palestine, I believe you'll be free.

Mariam

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to my supervisor, Dr. Radwan Tahboub, for his invaluable guidance and support throughout my master's program. His expertise, patience, and encouragement helped me to complete this research and write this thesis.

I would also like to thank Dr. Liana Tamimi, Dr. Mohammad AlDasht, and Dr. Isam Ishaq for their helpful feedback, advice, and suggestions.

I appreciate my family's encouragement, especially my Great Mom. Who eagerly dreamed of completing her graduate studies, and satisfied her longing by seeing my progress as if her dreams came true by me.

Mariam B. AlHawamdeh

Table of Contents

List of Figures

List of Tables

List of Algorithms

List of Abbreviations

AMR	Automatic Meter Reading
AMI	Advanced Metering Infrastructure
DCU	Data Concentrator Unit
HAN	Home Area Network
IoT	Internet of Things
LAN	Local Area Network
ML	Machine Learning
MDMS	Meter Database Management System
NAN	Neighbored Area Network
NTL	Non-Technical Loss
P2P	Peer-to-Peer
SC	Smart Contract
SG	Smart Grid
SM	Smart Meter
TL	Technical Loss
WAN	Wide Area Network

Chapter 1

Introduction

In recent decades, the energy distribution process has been dependent on transmitting energy from the supplier to the end customers using a traditional grid system that relies on traditional electricity meters to measure the energy consumption depending on accumulative measurement [4]. After a pre-agreed period of a month, less or more, an employee from the energy supplier company came to read the energy consumption manually to produce the energy consumption monthly bill. However, with the huge increase of energy customers and rapid progress of human civilization, the traditional energy distribution systems no longer sufficed. To follow the growth, the energy grids have become smarter, more automated, and more developed [5]. A Smart Grid (SG) is an intelligent grid that uses emerging technologies to improve the efficiency, scalability, and reliability of traditional grids. Smart Metering System is a key aspect of SG. It uses digital SMs to measure a variety of things for a group of consumers. It can be used to measure energy, water, or gas consumption depending on the application that it is used for [6]. The data measured can be provided in many techniques. One of the proposed techniques to provide the data on consumption is Automatic Meter

Reading [7]. The Automatic Meter Reading (AMR) technique was proposed to solve the problem of manually reading and monitoring. AMR is a one-way directional communication technique between the energy supplier utility and an advanced energy meter to read the energy consumption of each connected meter in a predefined period over a pre-configured communication network. This technology helped in energy consumption management and analysis [7, 8]. The Fourth Industrial Revolution affected energy grids by developing the energy meters and the whole network to be more reliable, interactive, secure, and smart [3]. It paves the way for researchers to propose more developed techniques to facilitate the handling of SGs and the data provided by the Smart Metering Systems [9]. Advanced Metering Infrastructure (AMI) is a system that enables two-way communication technologies to transmit and receive data through communication media between the components of the SG. AMI offers many advantages over traditional metering systems to improve efficiency and reliability. One of the advantages is providing real-time data on energy usage. It also allows the energy supplier utility to connect remotely to read and write on the consumers' meters. Which eliminates the need for manual reading of the energy consumption of each meter. Providing that type of data allows the energy supplier utility to provide the demand of each customer by analyzing energy consumption data. For consumers, AMI can be used to provide constant knowledge about their energy consumption and pricing in real time [10, 11]. The main components of the SG are a Smart Meter(SM), Data Concentrator Unit (DCU), and Meter Database Management System (MDMS). The SMs are used to measure energy consumption and collect data related to the home that is connected. Then, it transmits this data at the requested time or periodically to the DCU that is related depending on the grid subdivision areas. The DCU is connected to many SMs in the same area. Moreover, the main aim of the DCU existence is to connect the SMs with the MDMS and transmit received data to MDMS. MDMS is responsible for data management and control as a unit of the supplier utility system, it also can take action depending on the data received. The action can be a token transmitted to one or more SMs through the communication network [11]. Data in a SG system is highly sensitive, it can be a customer's personal information, energy consumption, consumption bill, customer debts, or other data. So, some aspects should be considered when designing a SG like providing system connectivity to efficiently collect and monitor system data in addition to privacy-preserving, reliability, and scalability [10]. The researchers have proposed many solutions to provide SG connectivity, security, and reliability. The solutions depend on enabling many emerging and trending technologies like IoT, Blockchain [4, 12], and Machine Learning [13, 14] in addition to Lightweight Encryption Algorithms [15] and Digital Twin [16, 17].

1.1 Problem Statement and Analysis

The energy demand is highly increasing because of the high growth in the population of the world, which has increased the necessity for designing efficient energy grids to provide effective load balancing and distribution. However, many challenges need to be addressed to achieve that aim. Those challenges include energy losses and unauthorized access to system data. Energy losses in the SG are divided into two main forms: Technical and Non-Technical losses (TL and NTL) [18]. TL in the SG refers to power losses that occur during the transmission and distribution of electricity. TL is mainly caused due to factors such as the conductivity of the power lines, the length of the lines, and changing conditions like ambient temperature and current density in the conductor. Depending on the infrastructure and voltage levels that are involved, TL normally falls between the range of 5 and 10% [19]. On the other hand, NTL in the SG is the loss that occurs due to factors other than technical reasons. This can include losses caused by energy theft or illegal connections in some areas. For example, the percentage of energy losses in the grids of distribution utilities in Palestine varies from 17% to 39% of the total energy [20, 21]. Also, based on a survey in [22], 25% of energy generated in India is considered as losses. So, Energy loss poses a challenge and causes a high cost of service. Therefore, many techniques and technologies are integrated with SGs to treat energy loss and unauthorized energy systems data access challenges.

In addition to energy losses, data plays a very significant role in SGs. The data in SG includes information on consumers, energy consumption, bills, voltage, and frequency. So, it is very sensitive, and at risk of unauthorized access. The risk of unauthorized access to system data in SGs is crucial. Unauthorized access can compromise the confidentiality, integrity, and availability of the data. It can lead to the exposure of sensitive information, manipulation of data, and disruption of the system's operations fully or partially. Therefore, implementing strong cybersecurity measures is required to mitigate this risk and ensure the security of SGs [23, 24].

1.2 Motivation

The electricity metering infrastructure is facing several challenges that are hindering its efficiency, security, scalability, and high management costs. Manual data collecting is required for traditional power meters, which is timeconsuming, prone to errors, costly, and lacks current information about behaviors of energy usage. This outdated approach leads to inaccurate billing, energy theft, and limited control over energy usage. Nowadays, while Smart Metering Systems have come a long way in automating data collecting and billing, they still have gaps in security, efficiency, and management that are susceptible to data manipulation, energy loss, and high costs due to the energy losses not being discovered early. Thus, there is an essential need to find an effective solution to address these limitations and provide efficient and secure Smart Metering System management. In our work, we choose to use IoT and Blockchain technologies and integrate them into the infrastructure of a Smart Metering System. Integrating Blockchain technology and IoT into smart metering infrastructure offers benefits that can enhance energy management and security preservation. IoT devices collect real-time energy consumption data, which eliminates the need for manual readings. This automated data collection ensures accurate and real-time management and reduces the risk of errors in estimations. In addition, Blockchain's decentralized nature ensures the integrity of energy consumption data, so the stored data can't be changed once it is added to the block. That enhances data security and integrity, improves real-time management and monitoring, and opens the way for many applications that can be applied to Smart Metering Systems at a lower cost and higher efficiency.

1.3 Research Objectives

This research aims to develop a solution to real-time energy consumption monitoring, provide data integrity,privacy preserving and detect energy losses of SMs by integrating the structure of smart energy grid systems with Blockchain and IoT technologies. This research intends to address the following objectives:

1. To propose a four-layer architecture of a smart energy metering system that enables IoT and Blockchain technologies.
2. To develop smart contracts of each layer to improve efficiency, privacy, and integrity preservation in the IoT communication of each part of the system, in addition to enabling energy loss detection.
3. To simulate the proposed model and compare its efficiency in different scenarios.
4. To calculate the reduced (saved) cost by enabling the proposed system on SGs in different periods.

1.4 Contributions

Overall, the contributions of this thesis are as follows:

1. Survey the recent works and solutions to ensure the security and efficiency of Smart Metering Systems.
2. Enable IoT and Blockchain technologies for a four-layer architecture of the Smart Metering System.
3. Develop a Smart Metering System that covers all layers in the SG from the devices at the lowest level to the MDMS at the highest.
4. Enable two-way communication in the proposed Smart Metering System.
5. Develop three smart contracts to ensure the efficiency and privacy-preserving of the proposed Smart Metering System.
6. Help in real-time monitoring of the Smart Metering System Data.
7. Help in the detection of energy losses over the system.

8. Ensure privacy preserving and integrity of the system's data and users.
9. Simulate different scenarios to study the proposed system efficiency.
10. Measure the performance of the proposed system in terms of latency, packet delivery ratio and storage consumption overhead.
11. Experimentally measure the energy and cost savings in the simulation of a proposed system in different scopes.

1.5 Research Methodology

The research methodology of this thesis will be applied as follows:

1. Explore and Understand the Existing SGs and Smart Metering Systems in the points of their challenges, applications, and implementation methods.
2. Explore the concepts related to IoT and Blockchain technologies in terms of characteristics and implementation methods.
3. Survey some of the recent solutions of SGs and Smart Metering systems and the technologies Integrated with them.
4. Design a Blockchain-IoT enabled Smart Metering System that proposes a solution for unauthorized data access and energy loss challenges.
5. Implement the proposed Smart Metering System.
6. Simulate the proposed system in different scenarios and measure the performance in terms of latency, packet delivery ratio and storage consumption overhead.
7. Simulate the proposed system to report the energy and cost savings due to the energy loss detection proposed criteria.

1.6 Publications

The following paper has been published in IEEE Xplore after being presented at the IEIT 2023 conference [25]:

M. AlHawamdeh and R. Tahboub, "Blockchain-IoT Enabled Smart Metering System," 2023 International Conference on Electrical and Information Technology (IEIT), Malang, Indonesia, 2023, pp. 207-213.
doi: 10.1109/IEIT59852.2023.10335500.

1.7 Thesis Organization

The rest of the document is organized as follows: In Chapter 2, a background review is presented. An overview of the recent SG solutions is provided in Chapter 3, and Chapter 4 presents the proposed IoT-Blockchain Smart Metering System model. Implementation, Simulation, and Results are provided in chapter 5. Finally, the conclusions and future works are summarized in Chapter 6.

Chapter 2

Background

This chapter provides an overview of the basic concepts in this thesis like IoT, Blockchain, smart contracts, smart metering technology, SG, and the convergence between them.

2.1 Internet of Things (IoT)

The Internet of Things (IoT) is the interconnection of physical objects that can communicate, interact, and transfer data. IoT aims to connect any object to the internet depending on the used architecture and the desired application [26, 27]. IoT basic architecture consists of three layers: Perception, Networking, and Application layers as shown in Figure 2.1. First, the perception layer which is also called the Physical Layer, includes a wide variety of physical devices such as sensors and actuators. These devices are responsible for interacting with the real world and data collecting and gathering. In addition, the Network layer provides the communication channels for the devices in the perception layer to send and receive data. Different communication technologies and network protocols can be used. The last layer is the Application layer, which involves Data storage and management, and Data Analytics and Application services [26]. There are different architectures of IoT depending on the desired application. The applications of the IoT are various and vast, they are reaching into almost every field of the human world. Fields of IoT applications are Agriculture, Healthcare, Manufacturing, Transportation, Smart Cities, and Grids, in addition to Home Automation and others [27]. Examples are shown in Figure 2.2.

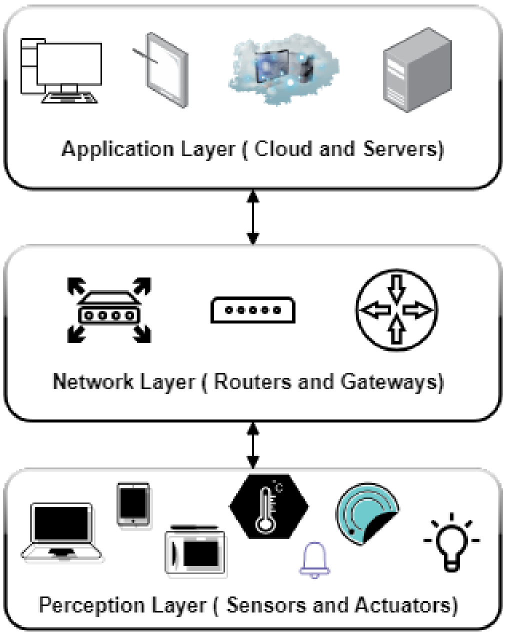

Application Layer (Cloud and Servers)
Network Layer (Routers and Gateways)
Perception Layer (Sensors and Actuators)

Figure 2.1: IoT Three Layers Architecture

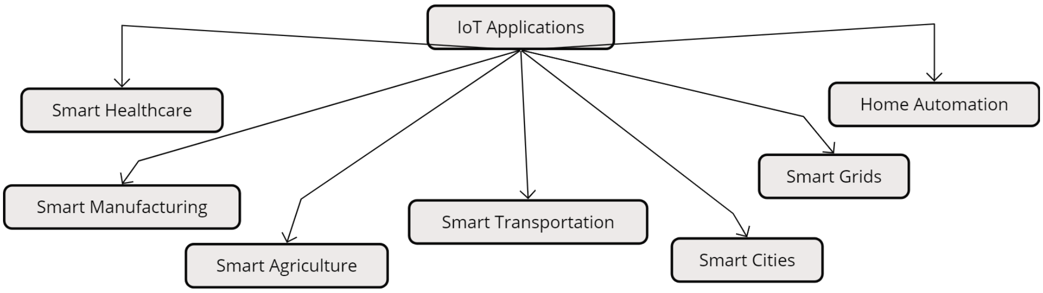


Figure 2.2: IoT Applications Examples

2.2 Blockchain

Blockchain is a decentralized distributed database technology that maintains continual growth ledger of records, called blocks. Blocks are linked and secured using cryptography techniques. Typically, Each block contains a cryptographic hash of the previous block, a timestamp, and transaction data as shown in Figure 2.3. Each block has transactions that have been verified by the Ethereum Blockchain and grouped at a specific time interval [28, 29].

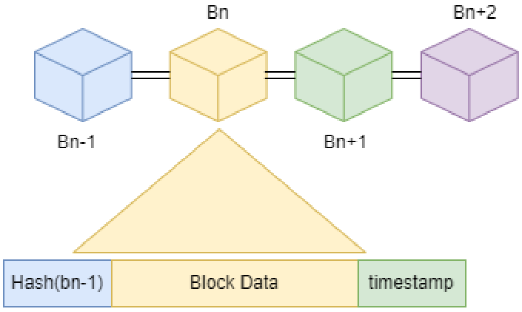


Figure 2.3: Typical Block Data

2.2.1 Blockchain Characteristics

Blockchain technology has several key characteristics [29, 30, 31], like:

- Decentralization: Blockchain operates on a decentralized network architecture. where all users are peer-to-peer (P2P) connected. Transactions in a blockchain network can occur between any two users with no need for a central authority authentication, and every user or node in the Blockchain network has a copy of the ledger.

- Persistence: Blockchain enables users to prove that their data are authenticated, where each block in the Blockchain structure is linked to the previous one and contains a hash of the previous block's data. That makes it nearly impossible to manipulate or modify the data.

- Auditability: Blockchain transactions are recorded in a digital distributed ledger and validated with a digital timestamp.
- Immutability: The new transactions in the Blockchain are made after being agreed upon by the nodes in decentralized consensus, Therefore the transactions are nearly impossible to tamper. Similarly, all previously held blocks in the blockchain are also immutable. To alter any previous block, an attacker would need to compromise a majority of the nodes involved in the blockchain network. Otherwise, any tamper in the blockchain is easy to detect.

2.2.2 Blockchain Architecture Types

There are three defined types of Blockchain Architectures, the first is Public Blockchain which is completely open and anyone can join and participate in the network. The most famous examples are Bitcoin and Ethereum. The second is Private Blockchain, where the nodes need permission to join and participate in the blockchain. So, access is restricted and controlled by the responsible entity. The last type is Consortium or Federated Blockchain. That type is semi-decentralized where more than one entity is responsible and manages the blockchain network [28].

2.2.3 A Block in Blockchain

The process of making a block in the Blockchain is batches of hashed and encrypted Merkle Tree transactions as represented in Algorithm 1 and Figure 2.4 [1, 2]. A Merkle Tree is a tree in which every leaf node is labeled with the hash of a single transaction of a block, and every non-leaf node is labeled with the cryptographic hash of the labels of its child nodes. Each block is timestamped and connected to the chain by storing the hash of the previous block.

The Algorithm 1 initializes an empty Merkle tree and partitions transactions into blocks. It then iterative hashes the blocks to construct a tree structure, ultimately generating a root hash. This root hash serves as a representation of the entire data structure, facilitating efficient verification of data integrity [1, 2].

The cryptographic hash function used by Merkle Tree is SHA256 [32, 8]. SHA-256 stands for Secure Hash Algorithm 256-bit designed by the National Security Agency (NSA). SHA-256 takes an input and returns a fixed-size, 256-bit hash value which is unique to the input; even a minor change in the input data will produce a significantly different hash value [33].

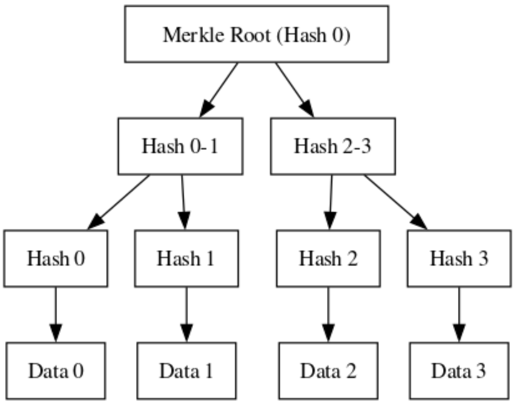


Figure 2.4: Merkle Tree Transaction Hash

Algorithm 1 Building a Merkle Tree [1, 2]

```
Initialization: Start the process.
Define $transactions$ as the list of transactions.
Define $blockSize$ as the size of each block.
Output: Root hash of the Merkle tree.
Initialize an empty list $merkleTree$.
foreach $i = 0$ to length of $transactions$ (incrementing by $blockSize$) do
    Initialize an empty list $block$.
    for $j = i$ to $\min(i + blockSize, length\ of\ transactions)$ do
        Append $transactions[j]$ to $block$.
    end
    Append $hash(block)$ to $merkleTree$.
end
while length of $merkleTree > 1$ do
    Initialize an empty list $nextLevel$.
    for $i = 0$ to length of $merkleTree - 1$ (incrementing by 2) do
        $concatenatedHash \leftarrow merkleTree[i] + merkleTree[i + 1]$.
        Append $hash(concatenatedHash)$ to $nextLevel$.
    end
    if length of $merkleTree$ is odd then
        Append $merkleTree$[length of $merkleTree - 1$] to $nextLevel$.
    end
    Update $merkleTree$ with $nextLevel$.
end
Return $merkleTree[0]$.
```

2.2.4 Ethereum Blockchain and Smart Contract

Ethereum is an open-source, blockchain-based platform that was proposed in late 2013 by programmer Vitalik Buterin and development was crowdfunded in 2014. Ethereum extends the basic blockchain principles established by Bitcoin to enable not just a platform for the exchange of digital currency, but a broader scope of applications through smart contracts. Ethereum cryptocurrency is called Ether (ETH), and it is used to reward nodes that perform computations and validate transactions on the blockchain network [34]. Ethereum features smart contract functionality, by enabling developers to build and deploy decentralized applications and smart contracts.

A smart contract is a predefined self-running program that is executed when predetermined conditions are met. These conditions are directly written into the program code. It is deployed on a blockchain platform, such as Ethereum. Smart contracts can be used to facilitate transactions depending on the designed application, such as financial transactions for financial applications without the need for a third party for authorization [35, 36]. A typical process of a smart contract is shown in Figure 2.5.

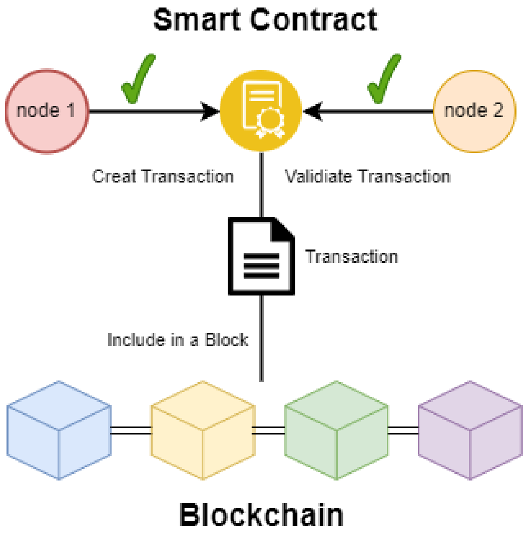


Figure 2.5: Smart Contract and Blockchain

2.3 Smart Energy Metering

Smart Energy Metering refers to the use of digital smart meters to follow and record energy consumption data for electricity systems in real-time and communicate that data to consumers and utility companies. Where traditional metering systems require manual readings, smart meters are more accurate, flexible, and efficient. They offer remote meter readings for real-time data, which can assist in different applications like loss detection and determining energy response. Smart meters typically record energy consumption hourly or on a predefined period. They enable two-way communication between the meter and the utility supplier [37, 38].

2.3.1 Smart Metering Technologies

Smart Metering has two common communication technologies depending on the structure of metering. One of them is Automatic Meter Reading (AMR) [8, 7] which is a communication technology that allows the supplier utilities to automatically read the energy consumed by each smart meter in the grid, in a one-way reading process. The other one is Advanced Metering Infrastructure (AMI) [11, 39] which is a bi-directional communication technology that allows the supplier utility to read and collect energy consumption from the metering infrastructure components. So, SMs can transmit data to the supplier utility company over a predefined network. In addition, it allows the supplier utility to send different commands to smart meters. This operation is working in or near real-time, this is deployed in a network called Smart Grid.

2.3.2 Smart Grid (SG)

A Smart Grid (SG), which is considered the advanced generation of electrical grids, uses digital technologies to monitor and manage the distribution of electricity. This includes the generation, transmission, distribution, and consumption measurement of electricity in the grid area. The basic components of SG are Smart Meters (SMs), sensors, communication networks, and other technologies that are all used in SG. These technologies allow electricity supplier utilities to monitor and control the grid in or near real-time. This can assist in the management and reduction of peak demand, detection of energy losses, prevention of outages to provide qualified service to consumers, and improving other technologies like renewable energy [9, 37].

Smart Grid Basic Components

The components of the SG architecture are Smart Meter (SM) which is the energy metering device. In addition to the Data Concentrator Unit (DCU) which is a digital device that acts as an intermediary between smart meters and the utility system, it gathers information from several connected smart meters through the communication network. Then DCU aggregates and preprocesses the received data before transmitting it to the MDMS. In addition, a DCU may perform additional tasks like meter configuration depending on the application. Also, the MDMS is one of the primary components that is responsible for processing received data from DCUs. The MDMS validates, organizes, controls, stores, and analyzes the data for efficient management of the grid and other various functionalities depending on the deployment considerations. The last component is the communication network which is the backbone of the whole system [40, 11].

Smart Grid Communication Technologies

Communication Technology is the backbone of SG. Where a communication network enables data exchange between the system components, there are many used communication techniques and they can be wired, wireless, or integrated between them like Power Line Communication (PLC), Radio Frequency (RF), Cellular Networks, Fiber Optic, Satellite Communication, and others. In table 2.1, we summarize some differences between different communication technologies that can be used as communication networks in the SG [40, 41]:

Table 2.1: Comparison of Different Communication Technologies

Technology	**Advantages**	**Disadvantages**	**Use Cases**
Power Line Communication (PLC)	• Low cost • Utilizes existing infrastructure • Short-range (up to 1km)	• Susceptible to interference • Limited bandwidth • Not supported by all meter types	Residential Areas
Radio Frequency (RF)	• Scalable and flexible • Good range (up to several kilometers)	• Higher cost than PLC • Requires line-of-sight • Complex network management	Rural areas with scattered meters
Cellular Networks (LTE-M, 3G)	• Wide-area coverage (national, global) • Low power consumption • Secure communication	• High costs • High latency • Require cellular network infrastructure	Remote areas with scattered meters

Table 2.1 – *Continued Comparison of Different Communication Technologies*

Technology	**Advantages**	**Disadvantages**	**Use Cases**
Fiber Optic	• High bandwidth • Extremely low latency • Secure transmission	• High-cost infrastructure installation	Critical infrastructure monitoring, Smart grid communication backbone
Satellite Communication	• Reaches remote and inaccessible locations	• Very high cost • High latency • Requires line-of-sight to satellite	Islands and offshore platforms

Smart Grid Benefits and Challenges

There are many benefits to working on SG from different aspects, including the following [42]:

- Improved Energy Efficiency: SG can assist in reducing energy losses in the transmission and distribution of electricity.
- Enhanced Reliability and Quality: SG can assist in preventing outages by providing utilities with real-time information about the grid.
- Demand Response and Load Balancing: SG allows for demand response programs, where consumers can be motivated to reduce their energy use during peak times, helping to balance the load on the grid.
- Integration of Renewable Energy Sources: SG can help in integrating

renewable energy sources into the grid, such as solar and wind power.

- Improved customer service: SG can help improve customer service by providing consumers with real-time information about their energy usage.

However, SG comes with several challenges that need to be addressed for enhancing the effective implementation and operation. Some of these challenges include [43]:

- Security: SG is vulnerable to cyberattacks such as Man-in-the-Middle attack [8]. Ensuring the security of the grid against attacks and data tampering is a significant challenge.
- High Initial Costs: The cost of implementing SG can be high due to installing advanced metering infrastructure, communication networks, and other technologies.
- Data Management and Privacy Concerns: SG components collect a lot of data about energy usage, which needs to be effectively managed and protected to preserve privacy.
- Reliability and Resilience: Reliability maintaining and resilience of SG in the face of natural disasters, equipment failures, and other disruptions is also a critical challenge.

The integration of SG with other emerging technologies like IoT, Blockchain, and Machine Learning can assist in enhancing the facing of SG challenges and achieving the benefits [42].

Chapter 3

Literature Review

In this chapter, an overview of some of the recently proposed works of smart metering systems and SG is presented. The chapter is divided into three sections; the first provides some of the recent solutions to preserve the security of different smart metering systems, the second presents some of the recently proposed solutions to improve energy efficiency and loss detection or SG, and the last is a conclusion of the literature review.

3.1 Security Preserving for SGs

The following are some of the recent papers that proposed solutions to preserve the security of SG.

3.1.1 Framework for Security Automatic Meter Reading Using Blockchain Technology

In [8], the authors proposed a new automated meter reading framework that enables Blockchain technology to provide security for the proposed framework. The proposed framework takes into mind the benefits of Blockchain technology to create a hybrid model of the smart metering system that consists of two components. First, smart meters are distributed in an area and related to customers. The second is the Blockchain network which consists of a cluster of servers Blockchain-enabled, and each server is connected to a group of smart meters. The cluster of servers is responsible for maintaining and reading the energy consumption of the connected smart meters. The servers receive data from the smart meters, then the data is considered input for the Blockchain network. The proposed solution utilizes the decentralized nature of the Blockchain to secure the data transmission in the network. The proposed framework also includes a set of processes that ensure the security of the proposed AMR system, including data encryption, access control, and transaction validation using a predefined smart contract. The smart contract is a set of rules that are agreed upon between the system components and used to create a valid and authorized transaction to be recorded to the ledger of the Blockchain. The proposed framework ensures data integrity, confidentiality, and availability. The framework also can provide a secure and reliable solution against different attacks like reply and DoS attacks. The proposed framework has been implemented by configuring the Ethereum Blockchain platform [34] as a Blockchain network. It also has been evaluated in terms of response time and latency on transaction creation. In addition, the proposed framework has experimented against DDoS attacks using the LoIC [44] tool. The proposed framework provides safety and security for the system data and network.

3.1.2 Implementation of a Smart Energy Meter using Blockchain and Internet of Things: A Step Towards Energy Conservation

In [45], the authors proposed a smart metering system that combines IoT and Blockchain technologies to make efficient control of renewable energy. The proposed system allows users connected to the same smart energy grid to use energy generated by neighborhood users to conserve local energy and mitigate energy loss. In this proposal, IoT and Blockchain are integrated to design an IoT-enabled energy meter that uses the Ethereum Blockchain to control and manage the authentication and authorization of the transactions between users, which follows a smart contract to make an authenticated and secure transaction for the Blockchain. One of the basic transactions is energy transferring between smart meters through an Android application. Any registered user in the SG database can download the Android application to be able to read its smart meter energy consumption and energy conserved in real-time. Also, the user can sell extra conserved power to other users in the SG and each buy and sell transaction will be recorded in the Blockchain after being authenticated by the smart contract. The proposed system has been designed using Arduino Nano and ESP8266 for IoT network implementation and Ethereum Blockchain for Blockchain and smart contracts. The proposed system works effectively compared with other previous work [46].

3.1.3 Blockchain and Secure Element, a Hybrid Approach for Secure Energy Smart Meter Gateways

In [47], the authors proposed a hybrid approach that combines Blockchain with Secure Element [48] technologies to secure data stored and transferred through the system. The proposed approach uses Blockchain for securing data immutability and Secure Element for ensuring data security at the point of generation which leads to building a trusted node at the device level before data transmission. The proposed approach aims to deliver effective and secure communication between smart meters and external market participants by relaying smart gateways in between to provide the security needed for the data exchanged. The proposed approach delivered three case studies for different three solutions, which were designed and improved to achieve the desired goal. The first is an end-to-end implementation using the Bigchain database [49] as a Blockchain database in addition to Riddle&Code Secure Element at the device level as a robust root. The designed case uses Wi-Fi as communication technology. The second uses Helium Blockchain [50] as a Blockchain database to store data and Multos Trust Core [51] as a Secure Element. In the last case, the author used UBIRCH Blockchain with 1NCE SIMs [52] and classic IoT at the robust root. The proposed three scenarios have been implemented and tested to be used as a P2P energy trading Platform. In addition, they provided a secure solution to safeguard IoT networks from cyberattacks. The three solutions are feasible, end-to-end secure, and trusted over different challenges of IoT platforms.

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

In [12], the authors proposed a secure communication scheme that uses Blockchain to secure data transactions of the AMI system. The proposed scheme uses a smart contract to substitute the trusted third party to avoid the single point of failure in the traditional AMI. Hyperledger Fabric Blockchain [53] is used to provide network distribution integrated with Practical Byzantine fault tolerance (PBFT) to ensure Blockchain transactions [54]. Users in Blockchain can communicate with each other without needing a trusted third party in a way that preserves the privacy of the communication of AMI components. The proposed scheme model works in four main phases. The first is system initialization, which aims to pre-load the main values of the system initiators of the Blockchain database. Second, is system registration and key generation, which aims to identify and register smart meters and MDMS to the defined Blockchain. In addition, it generates the private and public keys of each component which are derived from the ECC curve based on the smart contract. The third phase is to start data transactions and validate each transaction depending on the smart contract. The smart meter creates a transaction that contains electricity usage with a time of reading as a time stamp. The transaction is encrypted by the public key of MDMS then the transaction is verified by the smart contract. The fourth and last phase is to release the Blockchain. After a predefined period, the Blockchain data are transformed into the MDMS to be stored there. The proposed system has been implemented and the performance has been analyzed in terms of communication cost and time cost in comparison with other related studies.

3.1.5 Secure and Resilient Demand Side Management Engine Using Machine Learning for IoT-Enabled Smart Grid

In [14], the authors proposed a secure demand-side management (DSM) [55] engine that integrates machine learning with an IoT-enabled SG to maintain system security and optimize energy utilization. The proposed system has a new design of Home Area Network (HAN) to be associated with the secure demand side management and connected to a fixed WiFi network. Secure demand-side management performs authentication to secure all received data. Then, the data is processed into a resilient agent for the decision-making process. The resilient agent is responsible for maintaining the system's security from attacks and intrusions by applying data classification using machine learning models. The system can predict the authorized and unauthorized entities using a machine learning classifier. The proposed system has been implemented and analyzed, the results showed that the proposed system is effective and reliable in securing the SG system in addition to reducing the power utilization of DSM in SG and HAN devices.

3.1.6 Summary of Security Preserving Recent Works

Table 3.1 presents a brief definition of the security approaches used in the recent solutions for security preserving in SGs. Also, Table 3.2 shows a summary of the recent solutions to the SG security preserving challenge, their security approaches depending on Table 3.1, and some limitations.

Table 3.1: Security Approaches and Definitions

No.	Security Approach	Definition
1	Ethereum	A public, open-source, permissionless blockchain platform that utilizes smart contracts to enable developers to build and deploy decentralized applications (dApps). Ethereum operates on a proof-of-work (PoW) consensus mechanism [34].

Table 3.1 – *Security Approaches and Definitions- Continued*

No.	Security Approach	Definition
2	BigchainDB	A scalable, open-source blockchain platform designed for creating and deploying business-focused applications in decentralized environments. BigchainDB offers various consensus mechanisms, including proof-of-authority (PoA) and Byzantine Fault Tolerance (BFT), making it suitable for different use cases[49].
3	Helium	A decentralized wireless network leveraging the LoRaWAN protocol to provide internet connectivity for low-power devices. Helium motivates network participation through its Helium Token (HNT), creating a peer-to-peer (P2P) wireless infrastructure [50].
4	UBIRCH	A blockchain-based system for securing Internet of Things (IoT) data. UBIRCH utilizes self-sovereign identity (SSI) and permissioned blockchain technology to ensure the integrity, authenticity, and traceability of IoT data [52].

Table 3.1 – *Security Approaches and Definitions- Continued*

No.	**Security Approach**	**Definition**
5	Hyperledger Fabric	A modular, permissioned blockchain framework designed for enterprise solutions. Hyperledger Fabric offers privacy features, scalability, and interoperability to support various business needs. It utilizes a flexible consensus mechanism based on consensus plugins [53].
6	Demand-Side Management (DSM)	A strategy utilizing various techniques to influence consumer energy consumption patterns. DSM aims to optimize energy usage, reduce peak demand, and improve grid efficiency through incentives, price signals, and smart grid technologies. Machine learning can be applied to DSM systems for demand forecasting, dynamic pricing, and personalized recommendations [55].

Table 3.2: Recent Security Preserving Solutions Summary

Paper	Main Contribution	Security Approach	Limitations
Framework for Security Automatic Meter Reading Using Blockchain Technology,2021 [8]	Proposed a Blockchain-based framework for AMR that can improve security and reliability.	1	The framework was evaluated in a simulation environment. The results showed that the framework can improve the security and reliability of AMR systems.
Implementation of a Smart Energy Meter using Blockchain and Internet of Things: A Step Towards Energy Conservation, 2022 [45]	Implemented a smart energy meter using Blockchain and IoT to improve energy conservation.	1	The implementation was evaluated in a case study with a small number of participants. The results showed that the implementation can improve energy conservation.

Table 3.2 – *Recent Security Preserving Solutions Summary - Continued*

Paper	Main Contribution	Security Approach	Limitations
Blockchain and Secure Element, a Hybrid Approach for Secure Energy Smart Meter Gateways,2022 [47]	Proposed a hybrid approach for secure energy smart meter gateways that combines Blockchain and secure element technologies.	2, 3, 4	The approach was evaluated in a simulation environment. The results showed that the approach can improve the security of energy smart meter gateways.
A Blockchain-Enabled Distributed Advanced Metering Infrastructure Secure Communication (BC-AMI), 2022 [12]	Proposed a Blockchain-enabled distributed AMI secure communication system that can improve security and reliability.	5	The system was evaluated in a simulation environment. The results showed that the system can improve the security and reliability of AMI communication systems.

Table 3.2 – *Recent Security Preserving Solutions Summary - Continued*

Paper	Main Contribution	Security Approach	Limitations
Secure and Resilient Demand Side Management Engine Using Machine Learning for IoT-Enabled Smart Grid,2020 [14]	Proposed a secure and resilient demand side management engine using machine learning for IoT-enabled SG.	6	The engine was evaluated in a simulation environment. The results showed that the engine can improve the security and resilience of demand-side management systems.

3.2 Improving Energy Efficiency and Loss Detection of SGs

The following are some of the recent papers that proposed a solution to improve energy efficiency and energy loss detection.

3.2.1 Privacy-Preserving Electricity Theft Detection Scheme with Load Monitoring and Billing for AMI Networks

In [56], the authors proposed a privacy-preserving electricity theft detection scheme for the AMI grid and called it PPETD. PPETED aims to monitor system loads, compute bills, pre-serve customers' privacy, and identify electricity thefts using a machine learning model. The proposed model depends on the system operator (SO) which receives the periodic report of customers' energy consumption in the AMI grid where each customer has its smart meter. SO aims to learn the activities of fine-grained power consumption of smart meters to detect malicious customers in the network, which are tampering with their reports or trying to attack others. PPETED uses 4 techniques and protocols to preserve data security in the network: secret sharing cryptographic technique [57], SPDZ protocol, garbled circuits [58], and convolution neural networks (CNN) [59]. The proposed system has been mathematically modeled and implemented. The training and the privacy-preserving models have been implemented using Python 3 libraries and the OblivC framework on real smart meter data from Irish intelligent energy Trials [60]. The simulation results confirm that the proposed system imposes negligible computation and communication overhead, all the while preserving the privacy of network users.

3.2.2 Designing a Blockchain-Enabled Privacy-Preserving Energy Theft Detection System for Smart Grid NAN

In [22], The authors proposed a new system that combines blockchain technology and smart contracts to address the limitations of existing systems of SG. The proposed system aims to improve privacy preservation, authentication, and loss detection accuracy in SG neighborhood area networks (NANs). The proposed system utilizes Hyperledger Besu as the blockchain technology with smart contracts integrated to achieve the desired aim. The proposed system deployed three smart contracts, first is the energy supply contract, which is a public smart contract used by Distribution Network Operators (DNOs) to submit aggregated energy supply (AES) data for a predefined time interval. The second is the energy consumption contract, which is a private smart contract that is accessible only to the privacy group. Consumers use this contract to report their energy consumption during a predefined time interval. The last one is the energy theft detection contract which is used to analyze the data from the energy supply and consumption contracts to detect energy theft and faulty smart meters. The performance of the proposed system has been evaluated in terms of transaction throughput and latency. Various test rounds were conducted ranging from 50 to 500 transactions per second (tps). The results showed that the proposed system achieves high accuracy (98.88%) in energy theft detection with an affordable latency (less than 0.42 seconds) and a high throughput (98.37 tps). The research is part of a project titled “Developing Smart Controller for Optimum Utilization of Energy and Trustworthy Management in a Micro Grid Environment” funded by the Science and Engineering Research Board, Department of Science and Technology, Government of India.

3.2.3 An Active Learning Approach for Smart Grid-Based Energy Theft Detection

The paper [61] introduces an active learning-based machine learning model that is designed for the detection and classification of energy theft in SG. The proposed model stands out by integrating a variety of machine learning classifiers, including Random Forests (RFAL) [62], eXtreme Gradient Boosting (XGboostAL), Decision Tree (DTAL), Gradient Boosting (GBAL), K-Nearest Neighbors (KNNAL), Categorical Boosting (CatboostAL), and Light Gradient Boosting Machine (LGBMAL). These classifiers are applied to a dataset sourced from the Open Energy Data Initiative (OEDI), which is pre-processed to optimize it for analytical purposes. The performance of the proposed model has been evaluated by these classifiers using metrics such as precision, recall, and F1score. The results demonstrate varied levels of effectiveness across different classifiers, with some like GBAL #2 and KNNAL #2 achieving high precision, recall, and F1scores, while others like GBAL #6 and KNNAL #6 exhibit lower performance. Using the SG energy theft detection dataset, the proposed model outperforms competing models and obtains a detection accuracy of 70.61%.

3.2.4 Electricity Theft Detection Method Based on Ensemble Learning and Prototype Learning

In [63], the authors address the issue of electricity loss, particularly focusing on Non-technical Losses (NTLs). So, The authors propose an electricity theft detection method that leverages prototype learning and batch ensemble learning. This method is effective for imbalanced datasets. The proposed method involves constructing a prototype for each class based on feature embedding and then determining the label of each sample by finding the nearest prototype in the feature space. The authors also include experiments to validate the performance of the proposed method on imbalanced datasets. These experiments involve parameter optimization, comparison with other models, sensitivity analysis of abnormal levels, and an ablation study. The results demonstrate that the proposed method outperforms mainstream ensemble learning and deep learning models in classification ability, especially when dealing with datasets with a high level of imbalance. The paper presents various results, including the performance of different models like CNN+LSTM, CNN+LSTM+Ensemble, and the proposed method. The proposed method shows a high degree of robustness in dealing with abnormal data at low abnormal levels, as indicated by metrics like AUC (Area Under the Curve) and Diff (Difference).

3.2.5 Summary of Improving Efficiency and Loss Detection Recent Works

Table 3.3 presents a brief description and the main purpose of each technique used in the recent solutions for efficiency improvement and loss detection in SGs. Also, Table 3.4 shows a summarization of the recent solutions to the SG efficiency improvement challenges, their main contributions, used techniques depending on table 3.3, and some limitations.

Table 3.3: SG Efficiency Improving Techniques

No.	Technique	Description	Main Purpose
1	Secret Sharing	A cryptographic technique that takes a secret, divides it into multiple shares then securely distributes secret data among multiple parties without revealing the secret [64].	Security Preserving

Table 3.3 – *SG Efficiency Improving Techniques - Continued*

No.	Technique	Description	Main Purpose
2	Garbled Circuit	A cryptographic technique used in secure multi-party computation (MPC). It enables parties to compute functions on inputs without revealing them to each other. The function computed is represented as a circuit, and each gate in the circuit is ”garbled” by encoding its functionality to hide the inputs and outputs. Then, parties can evaluate the garbled circuit to compute the function without revealing each other’s inputs [65, 66].	Security Preserving.

Table 3.3 – *SG Efficiency Improving Techniques - Continued*

No.	Technique	Description	Main Purpose
3	SPDZ Protocol	A cryptographic protocol used for MPC, specifically for securing the computation of functions over private inputs from multiple parties, Garbled circuits can be used as a part of SPDZ protocol [66, 58].	Security Preserving.
4	Hyperledger Besu	An open-source Ethereum client for public and private permissioned network applications. It aims to provide a flexible and enterprise-grade platform for developing decentralized applications (DApps) and enterprise blockchain solutions [67, 68].	Security Preserving and Theft Detection.

Table 3.3 – *SG Efficiency Improving Techniques - Continued*

No.	Technique	Description	Main Purpose
5	Smart Contracts	Self-executing codes that enforce rules between the nodes of a Blockchain. They are associated with blockchain platforms such as Ethereum, which allows developers to write and deploy smart contracts using Solidity Programming Language [8, 36].	Security Preserving, Energy Consumption Monitoring and Theft Detection
6	Convolutional Neural Network (CNN)	A type of Neural Network Deep Learning that is designed to recognize and extract patterns of spatial data like images, making them well-suited for tasks such as recognition, detection, and classification [61, 69].	Energy Theft Detection

Table 3.3 – *SG Efficiency Improving Techniques - Continued*

No.	Technique	Description	Main Purpose
7	Recurrent Neural Networks (RNN)	A type of Neural Network Deep Learning that is designed to capture sequential dependencies in data, making them suitable for tasks such as time series analysis, and natural language processing. One of the RNN common architectures is Long Short-term Memory (LSTM) which is designed for long-term data dependencies [70]	Energy Theft Detection.
8	Random Forest (RFAL)	A Machine Learning Technique that constructs decision trees and is used for classification, regression, and prediction of the individual trees [62].	Energy Theft Detection

Table 3.3 – *SG Efficiency Improving Techniques - Continued*

No.	Technique	Description	Main Purpose
9	Tree Boosting	A Machine Learning Technique used for regression and classification goals by combining weak learners to create one strong learner depending on various libraries like XGBoost (XGBAL) which can provide more scalable and powerful tree boosting [71].	Energy Theft Detection

Table 3.3 – *SG Efficiency Improving Techniques - Continued*

No.	Technique	Description	Main Purpose
10	Instance-Based Learning	A Machine Learning Technique that learns directly from the training instances instead of creating a generalized model by storing all training examples to be used for predictions. K-Nearest Neighbors (KNN) is a common example of an instance-based learning algorithm used to classify data based on similarity to k nearest neighbors [72].	Energy Theft Detection

Table 3.4: Recent Works in SG Efficiency Improving and Energy Theft Detection Summary

Paper	Main Contribution	Technique Used	Limitation
Privacy-Preserving Electricity Theft Detection Scheme with Load Monitoring and Billing for AMI Networks, 2019 [56]	Proposed a privacy-preserving electricity theft detection scheme (PPETD) for the AMI grid.	1,2,3,6	Potential complexity and Computational overhead.
Designing a Blockchain-Enabled Privacy-Preserving Energy Theft Detection System for Smart Grid NAN, 2022 [22]	Developed a system combining blockchain technology and smart contracts for energy theft detection in SG NANs.	4,5	System scalability and Integration complexity.

Table 3.4 – *Recent Works Summary - Continued*

Paper	Main Contribution	Technique Used	Limitations
An Active Learning Approach for Smart Grid-Based Energy Theft Detection, 2024 [61]	Introduced an active learning-based machine learning model for energy theft detection in SG.	8,9,10	Lower performance in some classifiers.
Electricity Theft Detection Method Based on Ensemble Learning and Prototype Learning, 2024 [63]	Proposed a method combining prototype learning and batch ensemble learning for electricity theft detection.	6,7	Designed to work with imbalanced datasets.

3.3 Literature Review Conclusion

Blockchain and smart contracts have emerged as preferred solutions in recent related works due to their unique capabilities in enhancing various aspects of Smart Metering Systems as discussed in the previous sections. Blockchain technology provides a decentralized and immutable ledger that ensures transparency, integrity, and security of data. Blockchain eliminates the need for a centralized authority, reducing the risk of data manipulation and unauthorized access. This makes Blockchain technology an attractive option for Smart Metering Systems development. Furthermore, Smart Contracts automatically enforce agreements and execute Blockchain transactions when specified conditions are met, without the need for intermediaries. In the context of Smart Metering Systems, smart contracts can automate processes such as real-time monitoring, energy theft detection, and reducing administrative overhead. Moreover, the utilization of IoT devices enhances data collection and real-time monitoring capabilities across the entire Smart Metering infrastructure. These devices provide valuable insights into energy consumption, and SG performance, enabling proactive management. Combining Blockchain, Smart Contracts, and IoT in Smart Metering Systems offers several advantages. Firstly, it ensures the integrity and security of energy consumption data by leveraging Blockchain's characteristics. However, the decentralized nature of Blockchain enhances resilience and reliability. Secondly, Smart Contracts enable automation and facilitate efficient management of energy transactions and processes. Additionally, the IoT enhances system connectivity, real-time monitoring, and management. Therefore, we have opted to integrate Ethereum Blockchain, Smart Contracts, and IoT into a new Smart Metering System for enabling SG management, energy consumption monitoring, energy loss detection, and security preservation. This integration aims to create a flexible, secure, and efficient Smart Metering System for SGs. Recent efforts have been directed towards enhancing the efficiency of select components within SG, whereas the new proposed system aims to improve the efficiency and reliability of the whole SG through the designing of a new Smart Metering architecture.

Chapter 4

Proposed System Design

In this chapter, we provide an overview of the proposed system. The proposed system focuses on applying Blockchain technology to the IoT-enabled smart metering proposed architecture. The proposed system aims to monitor the Smart Metering System and preserve privacy by preventing any unauthorized party from tampering. In addition, the proposed system aims to detect the energy losses at smart meters and allows the MDMS to act on the detected problem. The proposed system combines the IoT-enabled Smart Grid system and blockchain technology by applying a decentralization network and avoiding any single point of failure. Blockchain in this system, will be used to protect data transmitted and received by the network nodes and ensure confidentiality, integrity, and availability of the nodes' data. Applying blockchain on the decentralized network is crucial in preventing data manipulation on several levels by unauthorized parties.

This chapter is organized into several sections, each addressing a distinct aspect of the proposed system. Section 4.1 introduces the System Model and Design Concept, covering design assumptions, system architecture, and communication security. Sections 4.2 and 4.3 delve into the incorporation of IoT and Blockchain technologies, respectively. Section 4.4 discusses Smart Contracts, including transaction workflows and two-way communication. Sections 4.5 and 4.6 focus on real-time monitoring and energy loss detection, respectively. Finally, Section 4.7 provides a summary of the chapter's key points and insights.

4.1 System Model: Design Concept

This section provides an overview of the proposed system architecture design and how Blockchain and IoT technologies are integrated into the proposed system.

4.1.1 Design Assumptions

For proposing this system, we assume the SG is located in a two-dimensional area of Am^2 that is divided into N neighborhoods. Each neighborhood is connected to the SG by a DCU that is connected to the MDMS. Each DCU is connected to the other DCUs. In addition, each DCU is responsible for the neighborhood SMs. The SMs within the nighaboord are interconnected with each other and with the DCU of that naighaboord. Also, the SM is connected to a smart home that has all devices connected to the internet. As shown in Figure 4.1.

From a general view, the hardware of nodes within the proposed Smart Metering System encompasses several essential components, as illustrated in the block diagram of Figure 4.2. The hardware architecture centers around the MCU, facilitating data communication and management. The energy measurement module collects real-time energy data efficiently using embedded sensors. The IoT module enables seamless communication and advanced

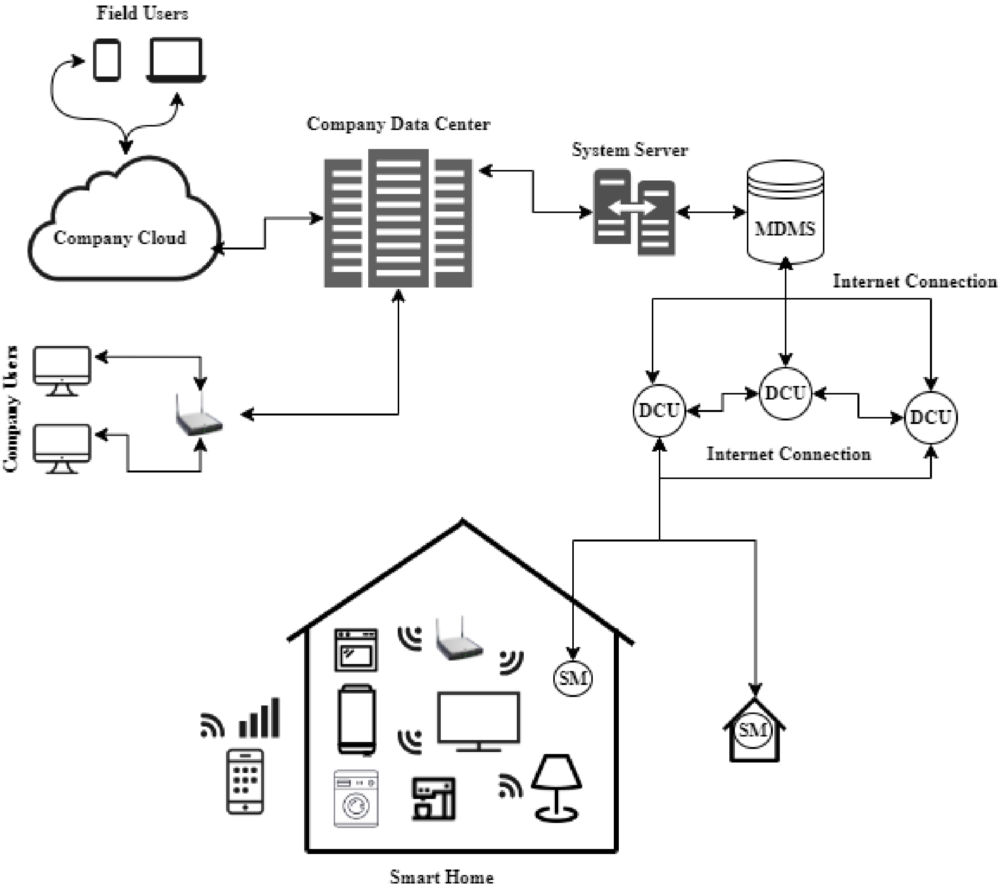


Figure 4.1: Proposed System Architecture View

functionalities, such as remote reading and connection to other nodes in the same layer of the system. Additionally, the AMI module enhances bidirectional communication, while the Blockchain module ensures data security and integrity, supported by the storage module for efficient data management.

For security, we assume that the nodes in the same layer have a pre-shared key (PSK) with the head node of the layer. Moreover, the head node has a PSK for each node belonging to it, and the key distribution is done during the physical installation. For example: The DCU that is responsible for a neighborhood of 100 smart meters, has 100 PSKs with each SM of its neighborhood, one for each. Correspondingly, each SM has the same PSK of the

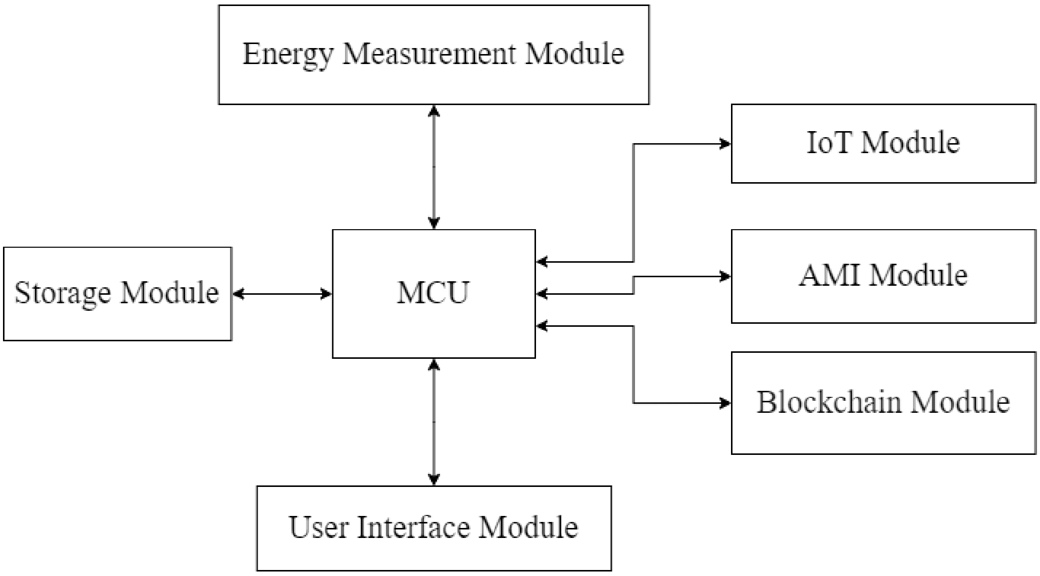


Figure 4.2: General Hardware Structure of Nodes

DCU. For connections, we assume that all smart grid nodes are connected to the internet by predefined connection infrastructure using wireless e.g.: Wi-Fi, or wired connections e.g.: Fiber optic connections. Each node in this system can transmit and receive data from the upper and lower nodes in a two-communication way. The whole system network is decentralized, so there is no single point of failure because all nodes are connected. In addition, we assume that smart home devices are IoT devices that are connected to the internet.

4.1.2 Proposed System Architecture

The proposed system has four main layers as shown in Figure 4.3. The first is the lower layer of the system architecture, the Local Area Network layer (LAN). The LAN layer is the smart home network that consists of the electrical devices in the home that are connected to the home network. This layer is connected by a two-way communication connection to the SM of that home.

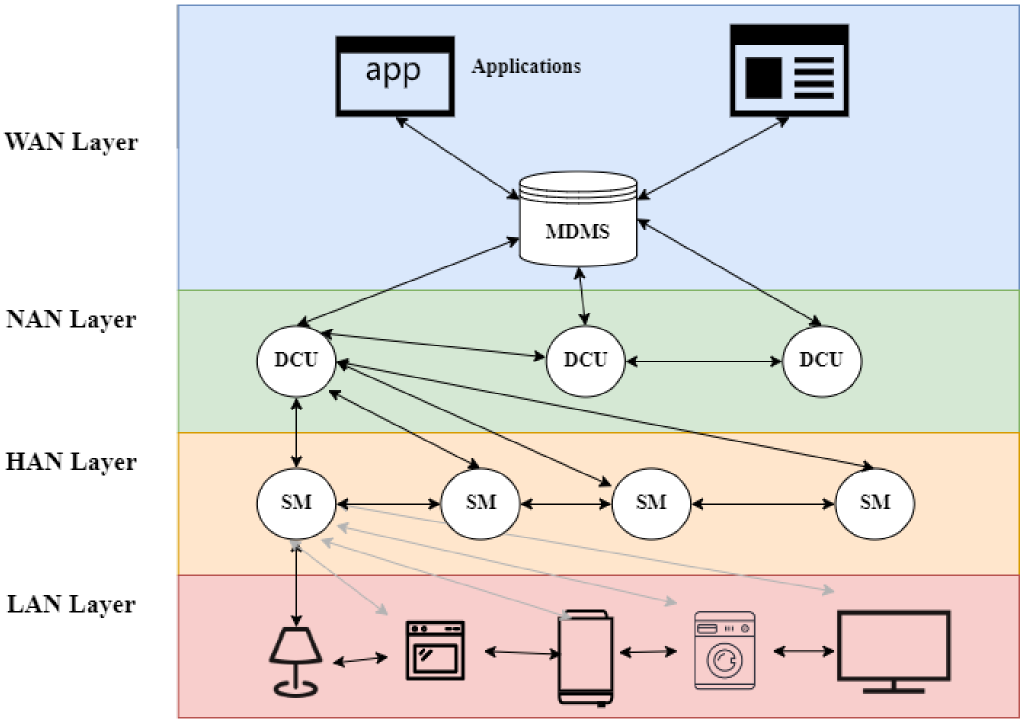


Figure 4.3: Proposed System Architecture Layers

The second is the Home Area Network layer (HAN) which consists of the neighborhood smart meters that are two-way connected to the same DCU as well they are connected. The third is the Neighborhood Area Network layer (NAN). NAN consists of the DCUs that are interconnected in a two-way connection. In addition, DCUs are two-way connected to the fourth layer. The fourth layer is the Wide Area Network layer (WAN). WAN is mainly composed of MDMS in the energy server of the Power Supply Company Data Center. WAN transmitted and received data from the lower layers by the two-way communication technologies.

Proposed System Layers

- LAN Layer: the Local Area Network layer (LAN), which is the lower layer of the proposed system architecture. The LAN layer is the smart home network that consists of all of the electrical devices in the home that are connected to the home internet network, as shown in Equation 4.1. This layer is connected by a two-way communication connection to the SM of that home.

$$LAN_i = \{SM_i, D_1, D_2, \ldots, D_n\} \tag{4.1}$$

- HAN Layer: The Home Area Network layer (HAN) consists of the neighborhood smart meters that are two-way connected to the same DCU as well they are connected, as shown in Equation 4.2.

$$HAN_i = \{DCU_i, SM_1, SM_2, \ldots, SM_n\} \tag{4.2}$$

- NAN Layer: The Neighborhood Area Network layer (NAN) consists of the DCUs that are connected in a two-way connection, as shown in Equation 4.3. In addition, DCUs are two-way connected to the Fourth layer.

$$NAN_i = \{MDMS, DCU_1, DCU_2, \ldots, DCU_n\} \tag{4.3}$$

- WAN Layer: The fourth layer is the Wide Area Network layer (WAN). WAN is mainly composed of MDMS in the energy server of the Power Supply Company Data Center. WAN transmits and receives data from the lower layers through two-way communication technologies.

4.1.3 Proposed System Communication and Security

In the proposed system each node in each layer collects data, processes it, and then transmits it to the upper layer. Any data in the system must be encrypted before it is transmitted, to ensure the security of the system. The proposed system uses the AES algorithm to encrypt the plain text data in the message before it is transmitted to the other nodes in the same layer. AES is a symmetric encryption algorithm established as an encryption standard by the U.S. National Institute of Standards and Technology (NIST) in 2001. AES is used to protect digital data and is a replacement for the older Data Encryption Standard (DES) [73, 74]. AES has widespread adoption and become a standard for symmetric encryption in various industries and applications. Its widespread adoption means that AES is well-supported by cryptographic libraries, hardware accelerators, and security protocols [75]. The plain text data is encrypted depending on the AES PSK. The AES PSK is generated and shared between each node and the head node of the same layer at the system construction. Then, when a node receives a message from another node it decrypts it using AES PSK as shown in Figure 4.4.

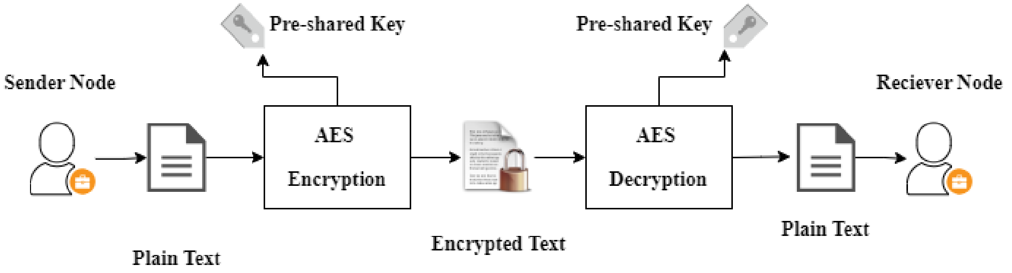


Figure 4.4: AES Algorithm Encryption and Decryption

Figure 4.5 shows an abstract view of the PSK in the HAN layer as an example.

For example, in the LAN layer, any home device d_i can read its energy consumption in Wh, encrypt it using the PSK of each device with the SM_i, and send the reading to the SM SM_i. So, the SM SM_i in the LAN layer decrypts the received reading and then it can have an instant reading of the energy consumed by each connected device. Also, the SM can read the whole en-

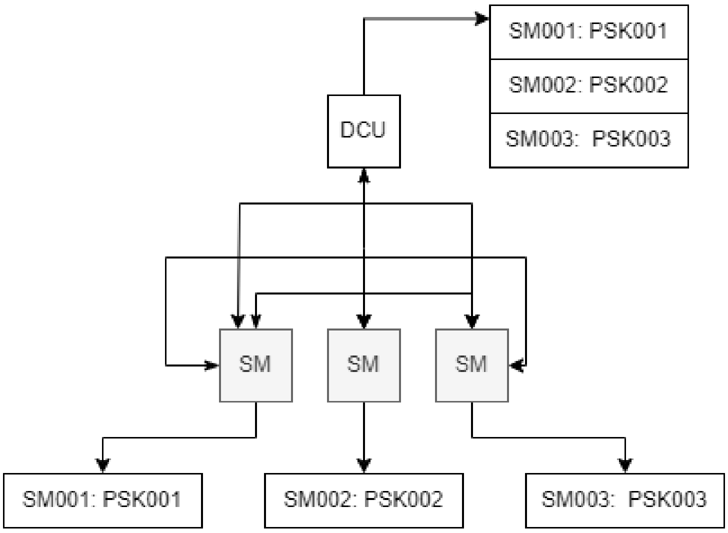


Figure 4.5: PSK in HAN

ergy consumed by the home devices which is the energy that comes out of it. The DCU_i in the HAN layer can read the neighborhood energy consumption coming from the SMs. So, all these readings will be received by the MDMS. After that the collected data can be processed by MDMS, it can take any action depending on the data readings. Also, the readings will be stored in the system database.

Utility system users will use MDMS to transmit instructions to the system's lower layers, e.g.: settings changing of a specific SM or DCU depending on a map that is defined in MDMS. Therefore, DCU will receive the instructions transmitted by the MDMS and take action depending on it, e.g., re-transmit the instructions to the required SM. All data transmitted in the system must be encrypted by the PSK of the desired layer, Then, when it is received by the desired node it must be decrypted using the same PSK.

4.2 IoT Enabled Proposed System

For enabling IoT in the proposed system, we suppose that all devices in the system are connected to the internet, and transmit and receive data through an internet connection using an IP address. Each device has an embedded IoT controller to enable use as a small computer like the Raspberry Pi [76] and memory for storage. After implementing the controller of the device to be accessible using the internet, we need to implement internet security measures to protect the device from cyber-attacks and ensure system connectivity and availability.

4.3 Blockchain Enabled Proposed System

The proposed system model focuses on secure communication between the system's layers and nodes using an authorized and reliable Blockchain platform. The use of Blockchain technology within the proposed system is to achieve security preservation of the system data by preventing different attacks, especially data tampering attacks [77]. In addition, Blockchain is used to detect energy NTL on the level of LAN and HAN layers. In the simple sense, Blockchain is used as a database to record transactions between system nodes. So each node in the system should participate in the Blockchain network to validate and preserve data transmitted and received. To create the Blockchain, first, we define a Smart Contract between nodes in the system. A Smart Contract [78] is a digital programmable contract that defines the rules and basic agreements between system nodes, to force the decentralized network nodes to adhere to these rules. Each node in the proposed system has a copy of the Blockchain and the Smart Contract of its layer. For the LAN layer, any node in LAN_i has a copy of that LAN Blockchain and Smart

Contract as shown in Figure 4.6.

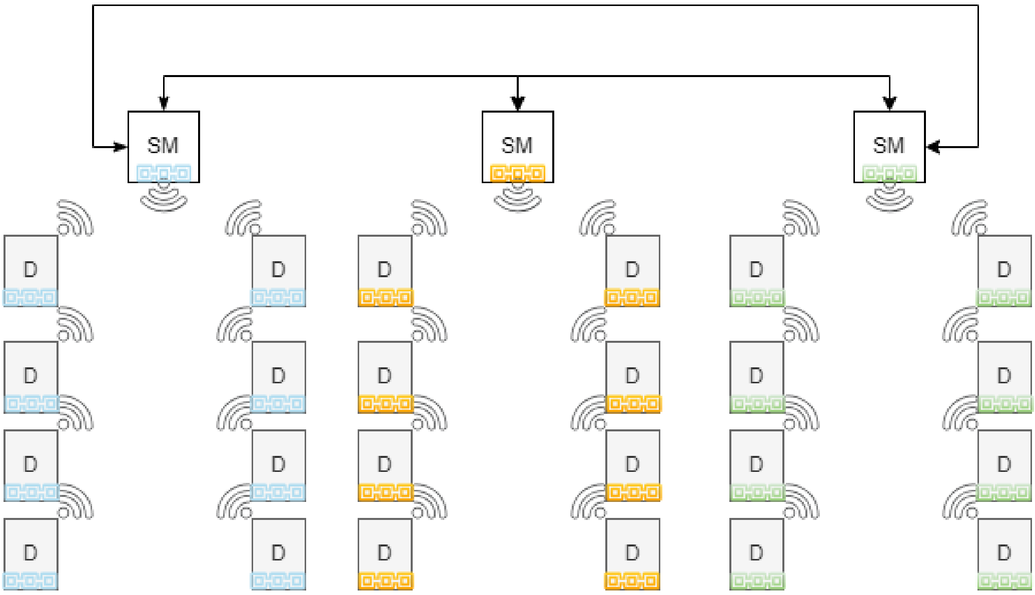


Figure 4.6: LAN Layer Blockchain

Also, for the HAN layer, any node in HAN_i has a copy of that HAN Blockchain and Smart Contract as shown in Figure 4.7.

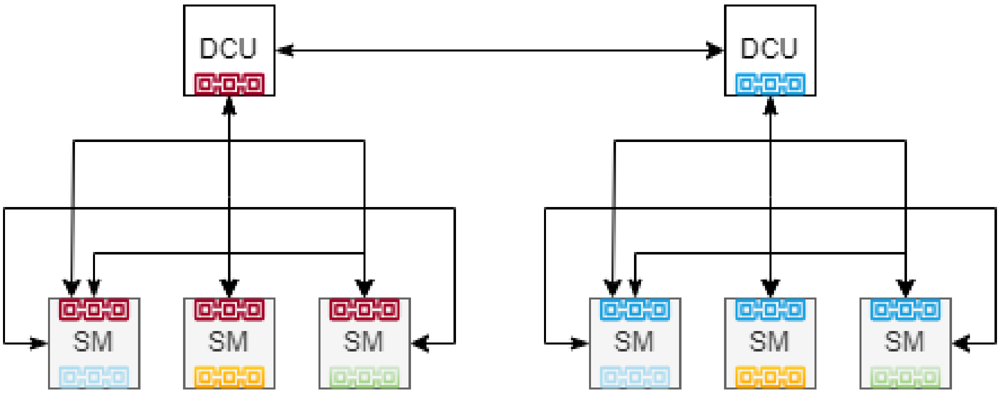


Figure 4.7: HAN Layer Blockchain

In addition, any node in NAN_i has a copy of that NAN Blockchain and Smart Contract as shown in Figure 4.8.

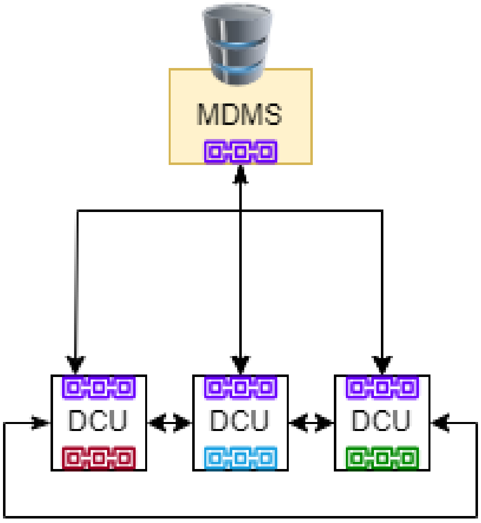


Figure 4.8: NAN Layer Blockchain

4.4 Smart Contracts

The proposed system is built on Ethereum Blockchain [34] with four layers with a Smart Contract for each layer that defines the rules between the layer's nodes in executable code. The smart Contract of the proposed system is written in solidity, a programming language derived from JavaScript, and developed for Smart Contracts [79]. Smart contracts have been created at the creation of the network. Each node in the system has access to the Smart Contract of the layer that it belongs to, these nodes are allowed to record data transactions in their layer's Blockchain after validation of the transaction. Figure 4.9 shows an abstract of the workflow of a transaction in any layer in the proposed system, and the steps are as follows.

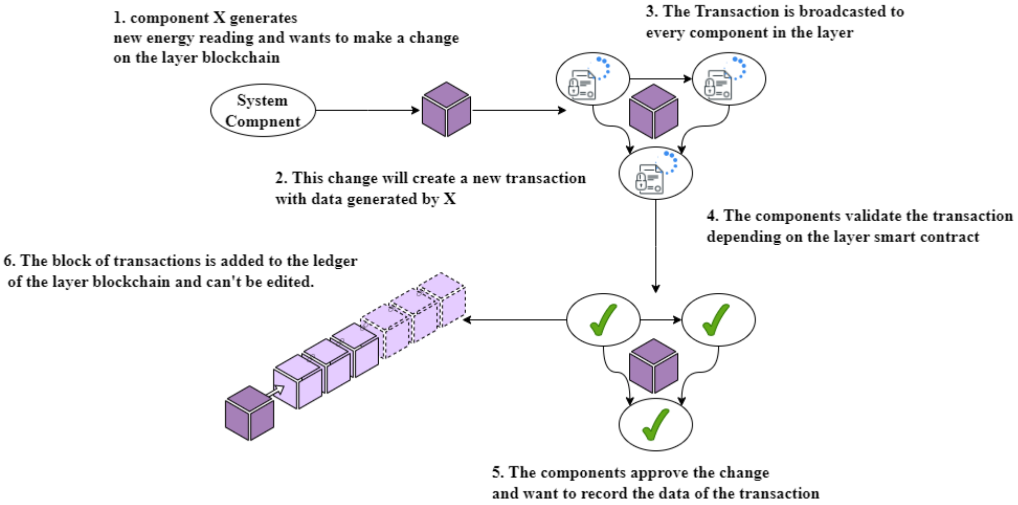

1. component X generates new energy reading and wants to make a change on the layer blockchain
System Compnent
2. This change will create a new transaction with data generated by X
3. The Transaction is broadcasted to every component in the layer
4. The components validate the transaction depending on the layer smart contract
5. The components approve the change and want to record the data of the transaction
6. The block of transactions is added to the ledger of the layer blockchain and can't be edited.

Figure 4.9: Transaction Workflow

4.4.1 Transaction Workflow

In the proposed system, the transaction goes through a workflow, intricately designed to maintain integrity, security, and efficiency. The following is a detailed exploration of each stage of the transaction workflow.

First: Key-Pair Generation

Each node in the system has its key pair (public and private keys) that are generated using an elliptic curve cryptography (ECC) scheme called ECDSA [3, 80]. ECDSA or Elliptic Curve Digital Signature Algorithm [3] is a cryptographic algorithm that is used for digital signatures that uses elliptic curve cryptography. In general, Digital Signatures use asymmetric cryptography, which involves a pair of keys: a private key and a public key. The private key is kept secret by the owning node in the network, while the public key is shared with other nodes in the same layer. The key pair of each node will be used to sign data transmitted or received to preserve data authentication and validation. The key pair are generated at the system initialization and when a new node is registered. Using ECDSA as a digital signature offers many advantages. It provides high-security levels against known attacks while requiring smaller key sizes. Furthermore, ECDSA signatures are compact, making them suitable for constrained environments like IoT devices, and their widespread adoption and standardization, Overall, the characteristics of ECDSA make it a perfect choice for digital signature applications, including Blockchain technology [81, 82]. To generate key pair (Q,d) using ECDSA an E: an Elliptic Curve has to be selected, In addition to G which is a base point on E of order n, where n is the smallest positive integer such that when G is added to itself n times, G will reach the identity element of the elliptic curve group, often denoted as the point at infinity, as seen in Equation 4.4 [3]. The proposed system uses a 256-bit elliptic curve to generate a key-pair where each r and s are 256-bit lengths.

$$n \cdot G = \mathcal{O} \tag{4.4}$$

For the key pair (Q,d), where Q is the public key of the node and d is the private key, the generation is done using the Equations 4.5 and 4.6 [3].

$$d \leftarrow \text{random integer from } [1, n-1] \tag{4.5}$$

$$Q = d \cdot G \tag{4.6}$$

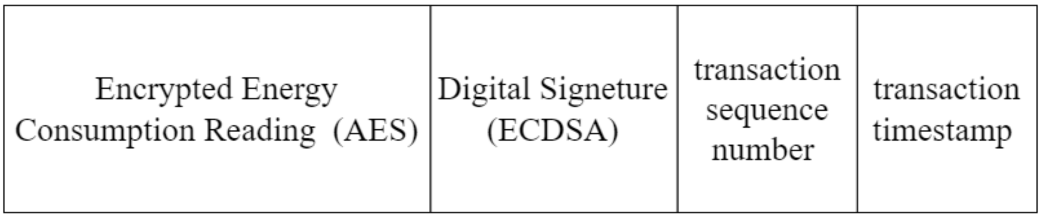
Second: Transaction Creation and Signing

Any node of LAN, HAN, and NAN reads their data and transmits it to the layer blockchain at predefined time intervals. The data could be an energy consumption reading, commands, alerts, and settings from MDMS. This data is handled as plain text, then it is encrypted using the PSK of the desired destination node. Then the source node uses its private key to sign a digital signature to the encrypted data. The digital signature of the encrypted data is attached to the data to create the transaction as shown in Algorithm 2. In addition, each node assigns the created transaction a unique sequence number. For example, in the HAN layer, The SM_i is connected to the DCU_i, and has a PSK with the DCU_i. The SM_i reads its energy consumption at time t and encrypts the data using the PSK of DCU_i. So, the energy reading is first encrypted with the PSK between the SM_i and DCU_i, then SM_i uses its private key to sign the data and create a transaction. This transaction is not validated yet, so it can't be stored in the HAN_i Blockchain or handled by the DCU_i. The signing process in ECDSA is shown in Algorithm 2 [3], where the node with a private key d wants to transmit a data message m. The Algorithm returns a pair of numbers (r,s) which constitutes the digital signature.

Algorithm 2 ECDSA Signing Algorithm [3]

```
Input: $m$ :data message,$d$,$n$,$G$
repeat
    $k \leftarrow$ random integer in $[1, n-1]$
    $(x_1, y_1) \leftarrow k \cdot G$
    $r \leftarrow x_1 \mod n$
until $r \neq 0$
repeat
    $s \leftarrow k^{-1} \cdot (m + r \cdot d) \mod n$
until $s \neq 0$
return $(r, s)$
```

Figure 4.10 shows an abstraction of the fields of the transaction created for the energy consumption reading.



Encrypted Energy Consumption Reading (AES)	Digital Signeture (ECDSA)	transaction sequence number	transaction timestamp

Figure 4.10: Transaction Abstract Fields

Third: Transaction Broadcast

The encrypted data, along with digital signature (r, s), is broadcasted to the nodes in a network of the layer as a Blockchain network. For example, for the HAN_i layer, the other smart meters in HAN_i and DCU_i received the broadcasted transaction.

Forth: Transaction Validation

Depending on the Smart Contract for each layer, the layer's nodes must validate and verify the transaction before it is recorded in the layer's blockchain and received by the destination node. Nodes in the layer, receive the transaction and validate it depending on the digital signature. For example, The nodes of the HAN_i layer will receive the transaction transmitted by SM_i. Then nodes use the SM_i public key to verify the digital signature of the transaction. This process ensures that the transaction was indeed created by the holder of the private key and has not been tampered with or altered. So, Before the data is recorded in HAN_i, its nodes should ensure that the public key associated with the SM_i is valid. This step ensures that the data being transferred to the Blockchain is coming from an authorized source. Assuming the current reading is valid, the HAN_i nodes then participate in a voting process to determine whether the transaction should be considered valid and recorded in HAN_i Blockchain or not. To store the data as a transaction in the blockchain of each node in HAN_i, The HAN_i nodes must validate the data using Ethereum Blockchain of HAN. The Algorithm 3 shows the transaction validation process using ECDSA [3] where the result is a boolean value, where the signature pair (r, s) for that transaction of a message m is valid or not. The layer nodes also check the transaction nonce to ensure the transaction's uniqueness and validity. A nonce in Blockchain is a counter that represents the number of transactions sent from a particular node in that layer. So, nonce must never duplicated. For Smart Contracts, the nonce ensures that function executions occur in the intended sequence [83].

Algorithm 3 ECDSA Signature Validation Algorithm [3]

```
Input: Public Key $Q$, Encrypted Message $m$, Signature $(r, s)$, Base Point
       $G$, Order of the Curve $n$
Output: Boolean indicating if the transaction is valid
Function validateTransaction($Q, h, r, s, G, n$) begin
    if $r < 1$ or $r > n - 1$ or $s < 1$ or $s > n - 1$ then
        return False
    end
    $u_1 \leftarrow (m \cdot \text{inverseMod}(s, n)) \mod n$  $u_2 \leftarrow (r \cdot \text{inverseMod}(s, n)) \mod n$
     $P \leftarrow u_1 \cdot G + u_2 \cdot Q$  if $P$ is at infinity then
        return False
    else
        return $r \equiv$ x-coordinate of $P \mod n$
    end
end
```

Fifth: Transaction Confirmation

Once the transaction is validated, this confirms that the transaction was indeed created by the sender and has not been altered in transit. Then, it is included in a candidate block with other validated transactions. Then the block is added to the Blockchain of that layer.

Sixth: Transaction Loss Handling

In the proposed Smart Contracts each node should keep track of the last transaction whether it's successfully added to the layer blockchain or rejected to ensure appropriate sequencing for new transactions. Before a node transmits a new transaction to the layer blockchain network, it keeps track that this transaction has been handled by the layer Blockchain network or not. So, the node verifies that the last transaction transmitted is the last transaction validated and is indeed the latest transaction for that node in the layer blockchain or it is the last transaction rejected by the layer blockchain. If the verification is successful, the node can proceed to transmit the new transaction to the layer blockchain network. If the verification fails, the node retransmits the last transaction before it transmits the new one. This process is done by defining a sequence of transactions for each node, each created transaction has a unique sequence number. If the transmitted transaction was handled by the layer blockchain network then the node recorded that the sequence number of the transaction is the last handled transaction. This ensures the sequence of transactions from each node is maintained correctly. In addition, it increases the integrity of the Blockchain for each layer.

4.4.2 Smart Contracts Algorithms

In this section, we present the algorithms used in each layer of our networked system to collect and manage energy consumption readings. These algorithms play a critical role in ensuring the integrity, security, and efficiency of data handling across the LAN, HAN, and NAN layers depending on the Smart Contract of each layer.

Algorithms Notations

The table 4.1 is the notation table for the layers of Smart Contract algorithms.

Table 4.1: Notations Used in the Algorithms

Notation	Description
D	Set of all IoT devices in the Home.
SM	Smart Meter.
DCU	Data Concentrator Unit.
dv_i	A specific IoT device in the set D.
sm_i	A specific SMin the set SM.
dcu_i	A specific IoT device in the set DCU.
LAN	Local Area Network, consisting of SM and all devices in D.
HAN	Home Area Network, consisting of DCU and all Smart Meters in SM.
NAN	Nighaboord Area Network, consisting of $MDMS$ and all DCUs in DCU.
d	private key of the node.
tr	Transaction
j	sequence no. of new transaction, for each node
k	sequence no. of last handled transaction, for each node
LAN_SC	Smart Contract associated with the LAN.
HAN_SC	Smart Contract associated with the HAN.
NAN_SC	Smart Contract associated with the NAN.
LAN_BC	LAN Blockchain.
HAN_BC	HAN Blockchain.
NAN_BC	NAN Blockchain.
T	Time interval for repeating the process.

LAN Smart Contract

The Algorithm 4 shows the flow of LAN Smart Contract, where the devices of LAN read their energy consumption at a specific time t and record the reading transaction to LAN Blockchain.

Algorithm 4 LAN Process and Smart Contract Algorithm

```
Initialization: Start the process.
Define $LAN$ as the group of $SM$ and devices in $D$.
foreach $dv_i \in D$ do
    Register $dv_i$ in $LAN$
end
while $true$ do
    // Repeat every interval $T$
    foreach $dv_i \in D$ do
        create transaction $tr_j$ where $j$ is the seq no. of $tr$ of $dv_i$  $j \leftarrow$ sequence
          no. of new transaction $tr_j$ for $dv_i$
        if $j = k + 1$ then
            Encrypt $tr_j$ with $PSK(SM)$ to get $E\_tr_j$  sign it with $(d)$
              broadcast $E\_tr_j$ within $LAN\_BC$
            if $LAN\ BC\ validates\ E\_tr_j$ then
                Proceed with the transaction, Record validated $E\_tr_j$ in
                  $LAN\_BC$
            end
            Reject transaction
            $k \leftarrow j$                                        // Update $k$
        else
            Retransmit $E\_tr_k$ to $LAN\_BC$
        end
    end
    Wait for time interval $T$
end
```

HAN Smart Contract

The Algorithm 5 shows the flow of the HAN Smart Contract, where the SMs of HAN read their energy consumption at a specific time t and record the reading transaction to HAN Blockchain.

Algorithm 5 HAN Process and Smart Contract Algorithm

```
Initialization: Start the process.
Define $HAN$ as the group of $DCU$ and Smart Meters in $SM$.
foreach $sm_i \in SM$ do
    Register $sm_i$ in $HAN$
end
while $true$ do
    // Repeat every interval $T$
    foreach $sm_i \in HAN$ do
        create transaction $tr_j$ where $j$ is the seq no. of $tr$ of $sm_i$  $j \leftarrow$ sequence
          no. of new transaction $tr_j$ for $sm_i$
          if $j = k + 1$ then
            Encrypt $tr_j$ with $PSK(DCU)$ to get $E\_tr_j$ sign it with $(d)$
              broadcast $E\_tr_j$ within $HAN\_BC$
              if $HAN\ BC\ validates\ E\_tr_j$ then
                Proceed with the transaction, Record validated $E\_tr_j$ in
                  $HAN\_BC$
            end
            Reject transaction
            $k \leftarrow j$                                    // Update $k$
        else
            Retransmit $E\_tr_k$ to $HAN\_BC$
        end
    end
    Wait for time interval $T$
end
```

NAN Smart Contract

The Algorithm 6 shows the flow of the NAN Smart Contract, where the DCUs of NAN read their energy consumption at a specific time t and record the reading transaction to NAN Blockchain.

Algorithm 6 NAN Process and Smart Contract Algorithm

```
Initialization: Start the process.
Define $NAN$ as the group of $MDMS$ and all DCUs in $DCU$.
foreach $dcu_j \in DCU$ do
    Register $dcu_j$ in $NAN$
end
while $true$ do
    // Repeat every interval $T$
    foreach $dcu_i \in NAN$ do
        create transaction $tr_j$ where $j$ is the seq no. of $tr$ of $dcu_i$  $j \leftarrow$ sequence
         no. of new transaction $tr_j$ for $dcu_i$
         if $j = k + 1$ then
            Encrypt $tr_j$ with $PSK(MDMS)$ to get $E\_tr_j$ sign it with $(d)$
             broadcast $E\_tr_j$ within $NAN\_BC$
             if $NAN$ $validates$ $E\_tr_j$ then
                Proceed with the transaction, Record validated $E\_tr_j$ in
                 $NAN\_BC$
            end
            Reject transaction
            $k \leftarrow j$                                        // Update $k$
        else
            Retransmit $E\_tr_k$ to $NAN\_BC$
        end
    end
    Wait for time interval $T$
end
```

4.4.3 Two-way Communication

The proposed system is designed to offer two-way communication between system layers. Where the energy consumption reading process is presented in the previous subsections, the writing process is not much different. However, the energy consumption is read from the lower layers to the upper layers. But, the writing is applied from the upper layers to the lower layers. The proposed system is implemented to achieve data writing from the MDMS to DCUs or SMs. The MDMS can write various types of data or commands to Data Concentrator Units (DCUs) or SMs (SMs) in a Smart Metering System. When MDMS needs to send data to a specific SM_i assigned to a DCU_i in the NAN layer, it first treats the data as any transaction. So, MDMS creates the data, encrypts it using the PSK of the DCU_i that SM_i belongs to, and signs the encrypted data with its private key to create the transaction. Then, MDMS broadcasts the transaction to connected nodes in the NAN layer. The nodes in the NAN layer validate the transaction and record it in the NAN blockchain after confirmation. The DCU_i rebroadcasts the transaction to the HAN_i nodes after it decrypts it using PSK shared with MDMS, and encrypts it again using the PSK with the desired SM_i then the DCU_i signs the transaction with its private key. HAN_i nodes receive the transaction, validate it, and record it in HAN_i blockchain. The SM_i receives the encrypted data, decrypts it using the PSK shared with DCU_i, and then takes an action on the data received after the transaction is recorded in the blockchain. Figure 4.11 shows a flowchart representing the flow of transactions from MDMS to SM_i.

4.5 Real-Time Monitoring

The proposed system is designed to monitor and record energy consumption readings of SMs at a predefined period and on demand of the utility system. The proposed system depends on collecting readings from SM in the HAN layer and transferring the readings to the NAN layer where the MDMS is a component. The energy consumption reading at a predefined period T process is done as follows:

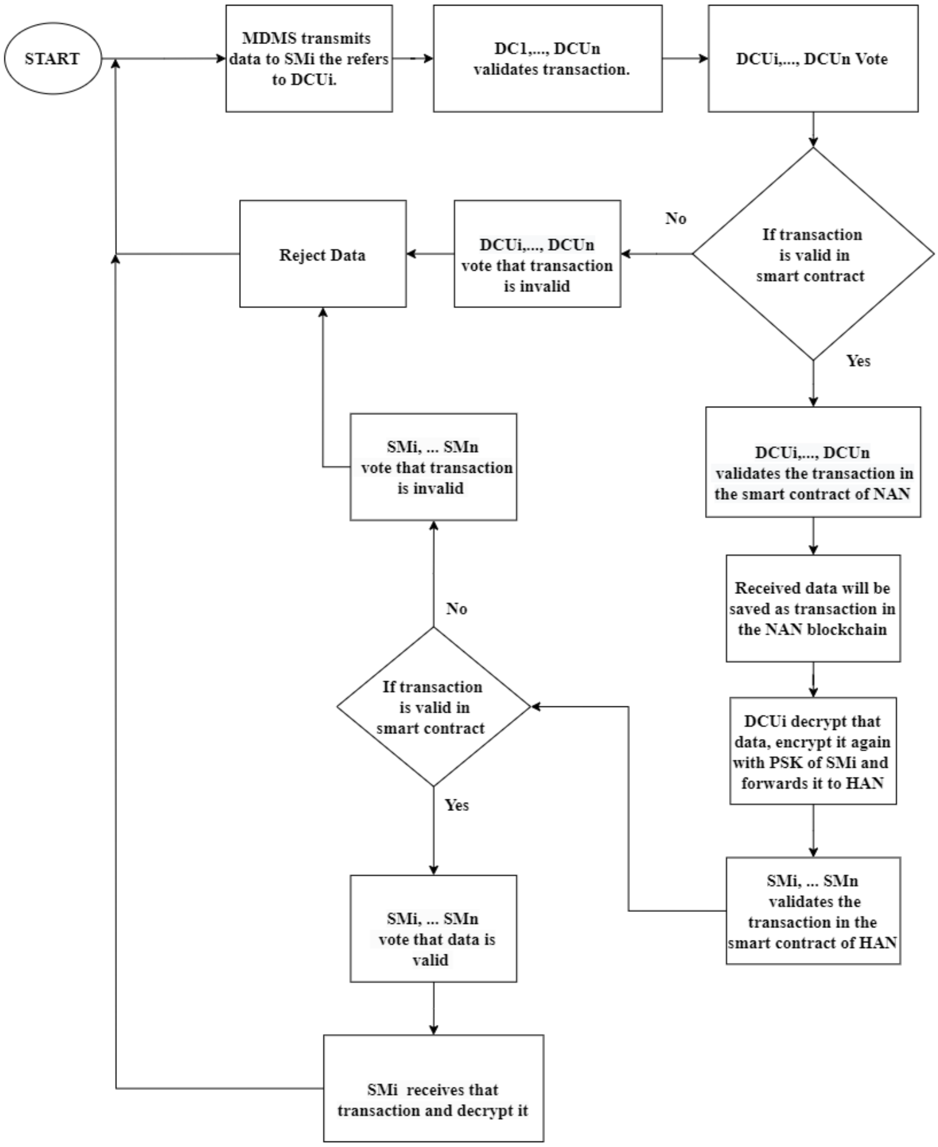

START
MDMS transmits data to SMi the refers to DCUi.
DC1,..., DCUn validates transaction.
DCUi,..., DCUn Vote
If transaction is valid in smart contract
No
Yes
DCUi,..., DCUn vote that transaction is invalid
Reject Data
DCUi,..., DCUn validates the transaction in the smart contract of NAN
Received data will be saved as transaction in the NAN blockchain
DCUi decrypt that data, encrypt it again with PSK of SMi and forwards it to HAN
SMi, ... SMn validates the transaction in the smart contract of HAN
If transaction is valid in smart contract
No
Yes
SMi, ... SMn vote that transaction is invalid
SMi, ... SMn vote that data is valid
SMi receives that transaction and decrypt it

Figure 4.11: WAN Transaction Flowchart

- Collection and Transaction Creation by Smart Meters: Every T period, each SM_j in HAN_i that belongs to DCU_i collects its readings and creates a transaction of encrypted energy reading $ECR(sm_i, t)$, signs the transaction and broadcast it in the HAN_i layer.

- Validation and Recording in HAN Blockchain: Each node in HAN_i validates the received transaction, and records it in the HAN_i Blockchain after validation.

- Aggregation by DCU: DCU_i of HAN_i decrypts the transactions delivered from SMs in HAN_i by the PSK of each one. Then DCU_i aggregates data of energy consumption readings of SMs in HAN_i encrypts that data with the PSK with MDMS creates a transaction and signs it using DCU_i's private key.

- Transmission to MDMS: DCU_i broadcast the transaction into NAN layer. The nodes of NAN validate the transaction depending on the DCU_i public key. If the transaction is validated depending on the NAN Smart Contract and recorded to the NAN Blockchain, MDMS receives the transaction, decrypts it using PSK with DCU_i, and handles energy consumption readings.

The energy consumption reading on demand is done as follows:

- MDMS Transaction: MDMS encrypts the request data, signs it, and transmits it to DCU_i that SM_i belongs to. The transaction is validated by the nodes in NAN, after the validation of the transaction, it is recorded in the NAN blockchain, as the process flow shown in Figure 4.11.

- Transferring by DCU: DCU_i of HAN_i decrypts the transactions delivered from MDMS in NAN by the PSK of MDMS. Then DCU_i transferred the transaction to the desired SM_i in HAN_i after DCU_i encrypts that data with the PSK of SM_i, creates a transaction and signs it using DCU_i's private key.

- Transmission to SM: DCU_i broadcast the transaction into HAN_i layer. The nodes of HAN_i validate the transaction depending on the DCU_i public key. If the transaction is validated depending on the HAN Smart

Contract and recorded to the HAN Blockchain, SM_i receives the transaction, decrypts it using PSK with DCU_i, and handles the data transmitted by MDMS then the flow is repeated as shown in the steps of energy consumption reading at a predefined period but is done from the moment of receiving a request data from MDMS.

4.6 Energy Loss Detection

The proposed system is designed to continuously monitor and validate energy consumption data and detect energy NTLs [18, 19]. To trigger the utility system to take an action. The proposed system is designed to detect energy losses in the LAN and HAN layers. In the LAN layer, the SM reads its energy consumption ECR_{SM_j} at time t and compares its instant reading with the sum of the devices energy consumption readings $\sum_{i=1}^{n} ECR_{device_i}$ that are validated and recorded at time t in the LAN blockchain. A threshold tl for acceptable losses (due to TL) is applied in the comparison technique as shown in Equations 4.7 4.8 and as shown in Figure 4.12.

$$\text{If } ECR_{SM}(t) - \sum_{i=1}^{n} ECR_{device_i}(t) \leq tl : \text{ Accepted Loss} \tag{4.7}$$

$$\text{If } ECR_{SM}(t) - \sum_{i=1}^{n} ECR_{device_i}(t) > tl : \text{ Loss Detected} \tag{4.8}$$

At the HAN layer level, the proposed system detects if there is an energy loss in DCU_i of HAN_i especially NTL, depending on its energy consumption aggregated of the associated SMs and the energy consumption read by the DCU. Same as in the LAN layer, the first step of energy loss detection is data collection of smart meters' energy consumption. where the DCU collects the energy consumption readings for SMs.The data is aggregated and transmitted to the layer blockchain depending on the Smart Contracts. The DCU also reads its energy consumption. The energy consumed by HAN_i should be the same as the sum of energy consumed by smart meters in HAN_i, taking into consideration the TLs. the DCU_i reads its energy consumption ECR_{DCU_i} at time t and compares its instant reading with the sum of the smart meters' energy consumption readings $\sum_{j=1}^{n} ECR_{SM_j}$ that are validated and recorded at time t in the HAN blockchain. A threshold tl for acceptable losses (due to TL) is applied in the comparison technique as shown in Equations 4.9 4.10 and as shown in Figure 4.13.

$$\text{If } ECR_{DCU}(t) - \sum_{j=1}^{n} ECR_{SM_j}(t) \leq tl : \text{ Accepted Loss} \tag{4.9}$$

$$\text{If } ECR_{DCU}(t) - \sum_{j=1}^{n} ECR_{SM_j}(t) > tl : \text{ Loss Detected} \tag{4.10}$$

The above set of equations provides a mathematical model for the energy loss detection process in the proposed system, leveraging aggregated data comparison and threshold based on TL.

The transaction of energy loss detection alert in SM_j in LAN_j is created and signed by the SM_j and broadcasted to the HAN_j layer Blockchain. The nodes in the HAN_j layer receive the transaction, validate it, and record it in the HAN blockchain. The nodes in the HAN layer receive the transaction, validate it, and record it in the HAN blockchain. The DCU_i receives the alert transaction of the SM_j which is detected as an energy loss after the transaction is recorded in the HAN blockchain and transmits it to the NAN layer Blockchain to be received by MDMS. Also, the transaction of energy loss detection alert in DCU_i in HAN_i is created and signed by the DCU_i

and broadcasted to the NAN layer Blockchain. The nodes in the NAN layer receive the transaction, validate it, and record it in the NAN blockchain. The MDMS takes the action of the DCU_i which is detected as an energy loss after the transaction is recorded in the NAN blockchain.

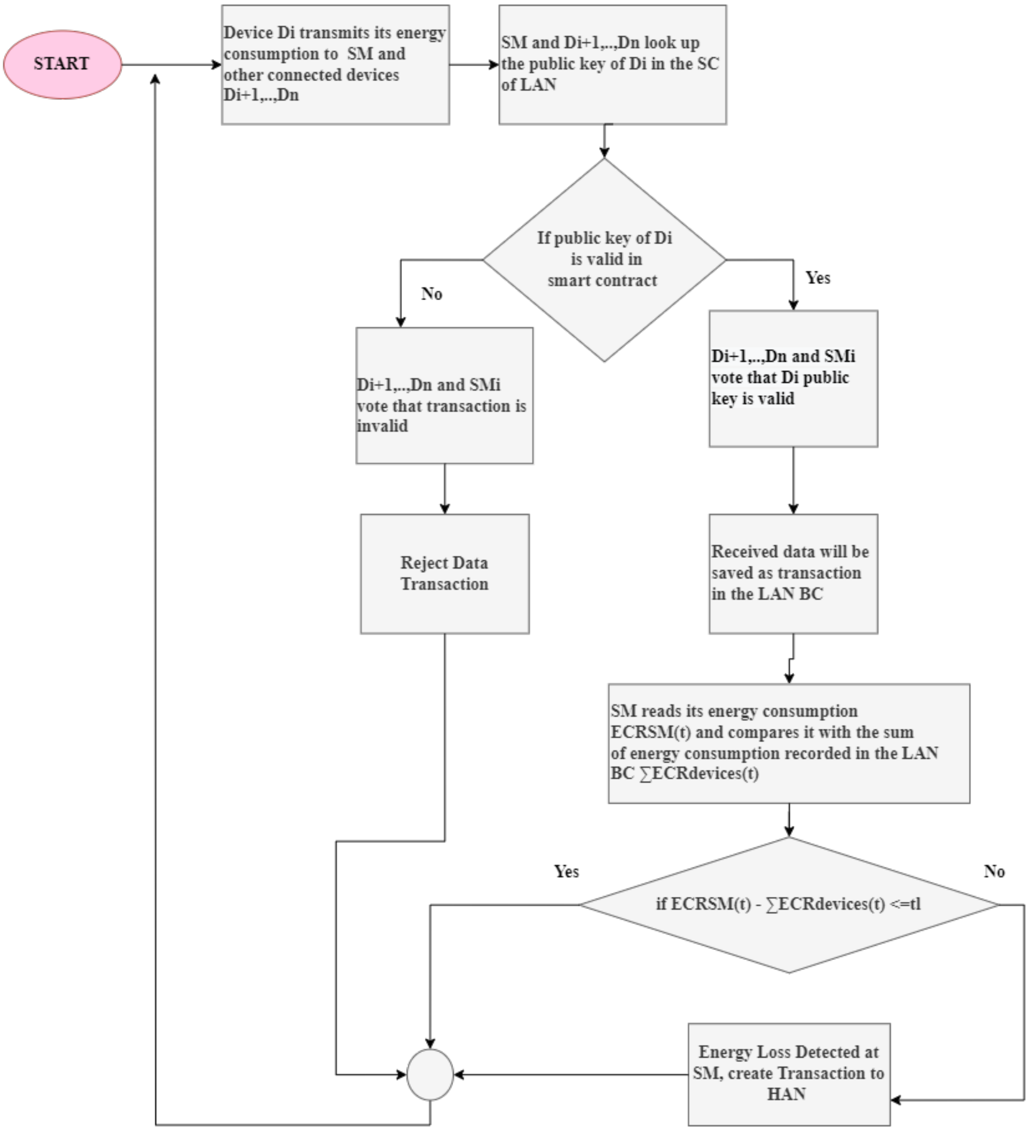

START
Device Di transmits its energy consumption to SM and other connected devices Di+1,..,Dn
SM and Di+1,..,Dn look up the public key of Di in the SC of LAN
If public key of Di is valid in smart contract
No
Yes
Di+1,..,Dn and SMi vote that transaction is invalid
Di+1,..,Dn and SMi vote that Di public key is valid
Reject Data Transaction
Received data will be saved as transaction in the LAN BC
SM reads its energy consumption ECRSM(t) and compares it with the sum of energy consumption recorded in the LAN BC ∑ECRdevices(t)
Yes
No
if ECRSM(t) - ∑ECRdevices(t) <=tl
Energy Loss Detected at SM, create Transaction to HAN

Figure 4.12: Energy Loss Detection at LAN

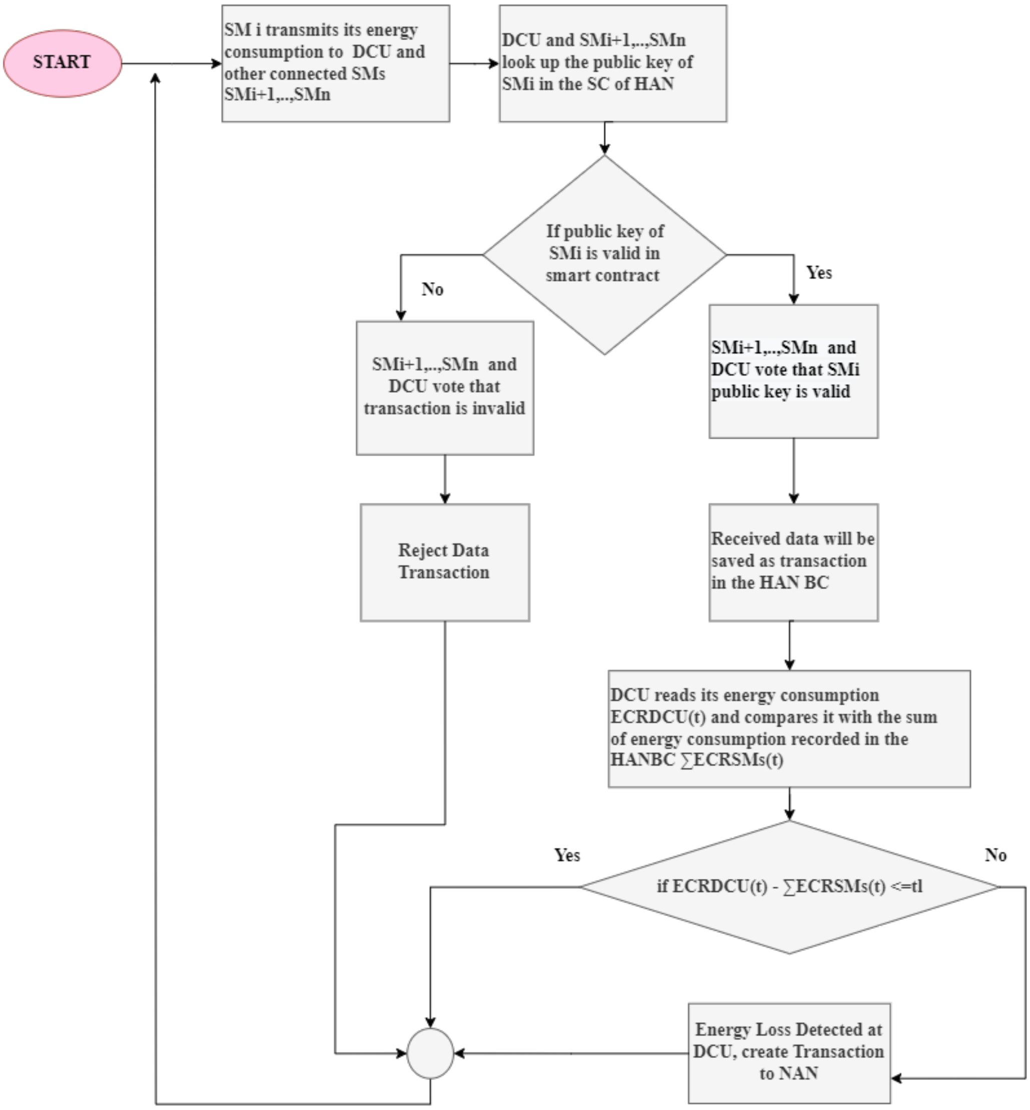

START
SM i transmits its energy consumption to DCU and other connected SMs SMi+1,..,SMn
DCU and SMi+1,..,SMn look up the public key of SMi in the SC of HAN
If public key of SMi is valid in smart contract
No
Yes
SMi+1,..,SMn and DCU vote that transaction is invalid
SMi+1,..,SMn and DCU vote that SMi public key is valid
Reject Data Transaction
Received data will be saved as transaction in the HAN BC
DCU reads its energy consumption ECRDCU(t) and compares it with the sum of energy consumption recorded in the HANBC ∑ECRSMs(t)
Yes
No
if ECRDCU(t) - ∑ECRSMs(t) <=tl
Energy Loss Detected at DCU, create Transaction to NAN

Figure 4.13: Energy Loss Detection at HAN

4.7 Summary

In this chapter, the focus lies on the design and modeling of the proposed Smart Metering System, which integrates Blockchain and IoT technologies. The primary objective is to enhance energy metering and monitoring processes by enabling real-time data collection, detecting energy losses, and ensuring security preservation throughout the system. The proposed Smart Metering System is structured into four main layers: LAN, HAN, NAN, and WAN. Each layer plays a crucial role in facilitating communication and data transfer within the system to monitor the energy consumed by system components efficiently. Also, The proposed system enables Ethereum Blockchain and Smart Contracts for LAN, HAN, and NAN layers to ensure efficient real-time monitoring and privacy preservation. Furthermore, The proposed system is designed for continuous monitoring and validation of energy consumption data, which are paramount for detecting energy NTLs and triggering corrective actions within the utility system. The system is intricately designed to identify energy losses occurring within both the LAN and HAN layers.

Chapter 5

Simulation, and Results

This chapter explains the practical approach to simulating a smart metering system by enabling Blockchain and IoT. To demonstrate the effectiveness of the proposed system, it has been implemented and its performance has been measured in terms of latency, average delivery ratio, storage consumption overhead, energy savings, and cost efficiency in different scenarios.

The scope of this thesis focuses on proposing a Blockchained-IoT Smart Metering System. Meanwhile, the scope of this chapter concentrates on real-time functionality and performance validation, contingent on the security analysis and testing of Smart Contracts and Ethereum Blockchain provided in previous works such as [8].

This chapter is organized into several sections, each dedicated to a specific aspect of implementing and analyzing the performance of the proposed Smart Metering System. Section 5.1 introduces the tools and methodologies used for implementation and simulation, detailing device specifications and simulation flow. Section 5.2 conducts a thorough performance analysis, covering metrics, scenarios, and results. In Section 5.3, energy consumption patterns and loss detection techniques are analyzed. Finally, Section 5.4 provides a concise summary, highlighting the key findings and insights gleaned from the implementation and performance analysis.

5.1 Implementation and Simulation Tools

In this section, the implementation steps and tools to simulate the proposed system are presented.

5.1.1 Device Specifications

The proposed system is simulated using a Laptop with the following specifications : CPU: intel core i7-7500U, @2.7GHz, SSD: 256 GB and RAM: 8GB. The installed OS is Ubuntu 22.04.2 LTS 64bit.

5.1.2 Simulation Flow

The steps that have been followed to simulate the proposed system are summarized as :

1. System Architecture Definition:

   - Devices: IoT devices in a smart home, are used to collect their energy readings. They belong to the LAN layer.
   - SMs: used to measure the energy consumption of a smart home and read devices energy consumption. They belong to the LAN and HAN layers.
   - DCUs: they are the head of the HAN layer, and they are used to aggregate data from SMs. They belong to the HAN and NAN layers.

- MDMS: Central system for data analysis and management. It belongs to the NAN layer.

2. Smart Contracts Development: The Smart Contract is a programmable code written in solidity to define the rules of transactions in the Blockchain, for the proposed system three Smart Contracts are implemented:

    - LAN Smart Contract: manages transactions among devices and SMs.
    - HAN Smart Contract: handles transactions among SMs and DCUs.
    - NAN Smart Contract: manages transactions among MDMS and DCUs.

    The Remix IDE [84] is used for solidity development and testing, the version of solidity is 0.8.2. To deploy and test the Smart Contracts on the local Blockchain Ganache Tool is used [85]. The Smart Contracts are deployed after switching the Remix environment to "Web3 Provider" and connecting it to the Ganache RPC URL.

3. Python Simulation : The programming language used to define the architecture is Python with various libraries and modules to perform the functionalities of the proposed system.

    - Network Topology: The network is built using classes and defined functions for transmit and receive functionalities depending on UDP protocol.
    - AES Encryption: The cryptography library implements AES encryption for secure data transmission. Two functions are defined to encrypt and decrypt data. The size PSK is 32 bytes for AES-

256. We define a unique PSK for each node in the layer to transmit the data to the head of the layer.

- Integration with Blockchain: The web3.py library is used to interact with the Ethereum Blockchain and Smart Contracts for each layer and Ethereum account creation for each node to interact as an Ethereum node using Ganache.
- Data Simulation: Artificial data is generated for devices and SMs. The Python libraries numpy and pandas are used to generate, manipulate, export, and import data.
- Energy Loss Detection: After data simulation, a loss detection function is implemented to detect the ID of the SM that has loss detected depending on the equations presented in the proposed model.
- System Modeling: The behavior of each component (devices, SMs, DCUs, MDMS) and the interactions between these components are simulated.

4. Simulation Testing: To test the simulated proposed system, the below steps are followed:

   - Unit Testing was performed to ensure that the solidity Smart Contracts and Python code were running well and handling any errors.
   - End-to-end Testing was performed to simulate the workflow from device data generation to Blockchain transaction validation and energy loss detection.

5.2 Performance Analysis

The performance of the proposed system is evaluated through two scenarios. Each scenario is designed to test and measure specific aspects of the system's capabilities and efficiency under different conditions. The performance is measured in terms of latency, packet delivery ratio (PDR), and storage consumption overhead.

5.2.1 Performance Metrics

The following are the performance metrics definitions.

1. Latency is the difference in time consumed between packet transmission from the source node and packet reception at the destination node, as shown in Equation 5.1.

$$L = t_{\text{received}} - t_{\text{sent}} \tag{5.1}$$

The latency is measured in the following situations proposed system simulation.

- Latency for packets transmitted from devices and received by SM in the LAN layer in conditions of changing N, where N is the number of devices connected to a SM.
- Latency of packets transmitted from SMs and received by DCU in the HAN layer in conditions of changing N, where N is the number of SMs associated with the SM.
- Latency of packets transmitted from DCUs and received by MDMS in the NAN layer in conditions of changing N, where N is the number of DCUs associated with the MDMS.

- Latency of a packet transmitted from SM_i and received by MDMS in conditions of changing N, where N is the number of SMs in the network.

2. Packet Delivery Ratio (PDR) is used to evaluate the quality and reliability of the system network. PDR is the ratio of the number of packets received successfully at the destination to the number of packets sent by the source. and it is expressed as a percentage. PDR is calculated as shown in the Equation 5.2.

$$PDR = \left(\frac{\text{Number of Packets Received Successfully}}{\text{Number of Packets Sent}}\right) \times 100\% \quad (5.2)$$

The average of PDR is measured in packet transmission from SM_i and received by MDMS in conditions of changing N, where N is the number of SMs.

3. Storage Consumption Overhead refers to the amount of storage or memory space required to store the data for future use and is measured in the units of Bytes. The average storage consumption overhead is recorded in the following situations proposed system simulation depending on different energy consumption reading intervals.

   - Device's storage consumption overhead to store LAN layer Blockchain: This refers to the amount of storage space required by each device to store blockchain data related to the LAN layer in conditions of changing i, where i is the number of devices associated with the SM and T where T is the pre-defined period to read the energy consumption by each device.
   - SM's storage consumption overhead to store LAN and HAN lay-

ers Blockchain: This refers to the amount of storage space required by a SM to store not only data related to the LAN layer but also data of the HAN layer in conditions of changing i and j, where i is the number of devices associated with the SM_j, j is the number of SMs associated with DCU and T where T is the pre-defined period to read the energy consumption by each SM.

- DCU's storage consumption overhead to store HAN and NAN layers Blockchain: This refers to the amount of storage space required by a DCU to store data related to both the HAN and NAN layers of the proposed system in conditions of changing i and j, where i is the number of SMs associated with the DCU_j, j is the number of DCUs associated with the MDMS and T where T is the pre-defined period to read the energy consumption by each DCU.

5.2.2 Analysis Scenarios

The proposed system is implemented and analyzed in two scenarios to test and evaluate the proposed system as an indicator of its effectiveness, reliability, and performance.

- Scenario 1: Standard Mode

  In the Standard Mode, the proposed system is implemented using simple and ideal network communication using UDP protocol without integrating Blockchain validation. This scenario aims to get an indicator of the performance of the smart metering system in an ideal IoT framework.

- Scenario 2: Blockchained Mode

  In the Blockchained Mode, the proposed system is implemented by integrating the proposed Blockchain layers with the metering system. This integration is designed to evaluate the enhancements or changes in performance brought about by Blockchain validation.

By comparing the latency, PDR, of these two scenarios, and analyzing the storage consumption overhead metric of the Blockchained mode, we can derive a comprehensive insight into how Blockchain technology impacts the performance of IoT-based smart metering systems. By taking into consideration that while there is no need to store any historical data for the Standard Mode, the Blockchained Mode is required to store a copy of the layer Blockchain in each node related to that layer.

5.2.3 Performance Analysis Results

The results of the proposed system performance are described in the following figures and description.

LAN Layer Latency :

Figure 5.1 shows the latency of the LAN layer in the simulation where transaction size is fixed to 100 bytes, and the number of devices attached to the SM is changed. As can be seen, the increasing of devices attached to the SM affects the latency of transaction delivery in the two scenarios. By increasing the number of devices, the latency in the two modes is increased, but it is larger in the Blockchained Mode due to the time that is taken in the transaction validation process.

HAN Layer Latency :

Figure 5.2 shows the latency of the HAN layer simulation where the transaction size is fixed to 100 bytes and the number of SMs attached to the DCU is

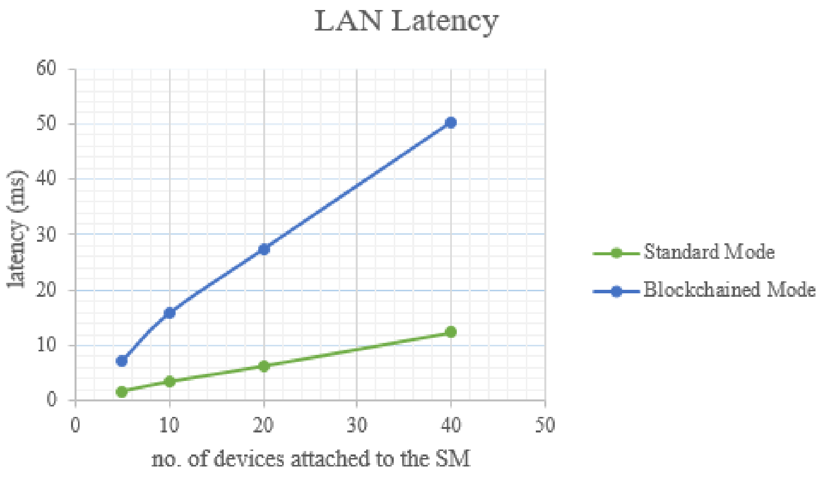


Figure 5.1: LAN Latency

increased. As can be seen, increasing SMs attached to the DCU increases the latency of transaction delivery in the two scenarios. However, it is larger in the Blockchained Mode due to the time taken in the transaction validation process.

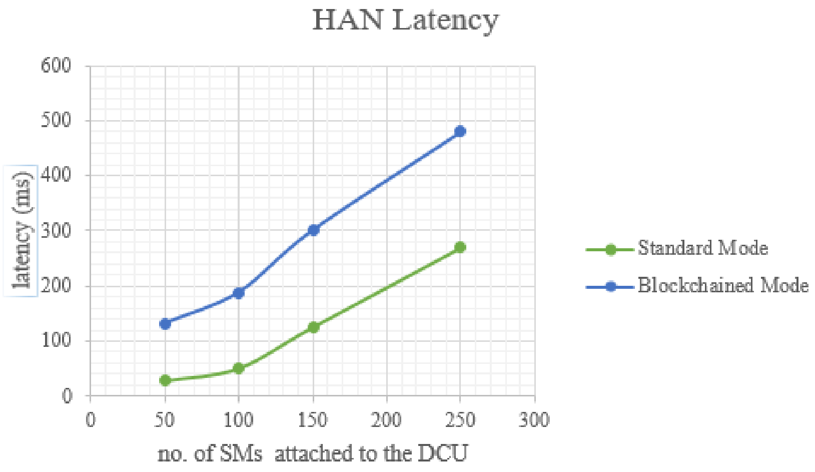


Figure 5.2: HAN Latency

NAN Layer Latency :

It is the same for the NAN layer, where figure 5.3 shows the latency of the NAN layer simulation where the number of DCUs attached to the MDMS in the metering system is increased and the number of SMs attached to the DCU is fixed to 250 SMs. As can be seen, increasing SMs attached to the DCU increases the latency of transaction delivery in the two scenarios. However, it is larger in the Blockchained Mode due to the time taken in the transaction validation process.

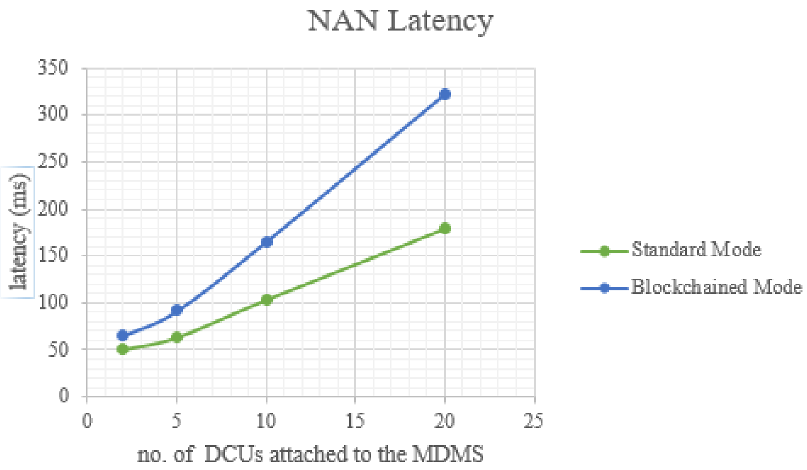


Figure 5.3: NAN Latency

Average Latency :

Finally, the average latency is measured for transactions transmitted from a SM attached to a DCU to the MDMS by increasing the number of SMs in the system passing by HAN and NAN layers. Where the maximum number of SMs attached to each DCU is 250 SM in the HAN layer, and the transaction size is fixed to 100 bytes. The number of DCUs is also increased by the increase of SMs number, for each increase of new 250 SMs, they are attached to a new DCU. Figure 5.4 shows that the average latency is increased by increasing the number of SMs in the system in the two scenarios. However, it is larger in the Blockchained Mode due to the time taken in the transaction validation process. While the latency of the validation process is increased by the increase in the number of SMs.

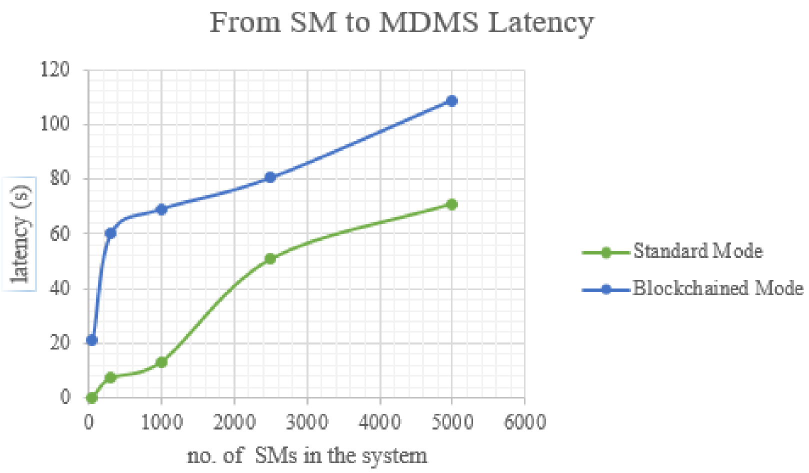


Figure 5.4: Average Latency

Packet Delivery Ratio (PDR):

The average of PDR is measured in the simulation of increasing the number of SMs, each of which sends a transaction to the DCU that the SM belongs to, and then the responsible DCU transmits it to the MDMS. The number of SMs attached to each DCU is 250 SM in the HAN layer, and the transaction size is fixed to 100 bytes. Figure 5.5 shows that the PDR of the Standard mode is highly decreased by the increasing of transactions number, so the packet loss is increasing. But for the Blockchained mode, the PDR is decreased stably, due to the transaction loss handling in the Smart Contract of each layer blockchain in the proposed system.

Storage Consumed by a Device

Table 5.1 and 5.2 present the storage consumption overhead of devices in the LAN layer. The overhead is measured in kilobytes (KB) over different periods by changing the number of devices in the LAN layer, considering a scenario where each device D_i takes 4 readings per day. Also, Table 5.2 shows the same metric measurement in a scenario where each device D_i takes

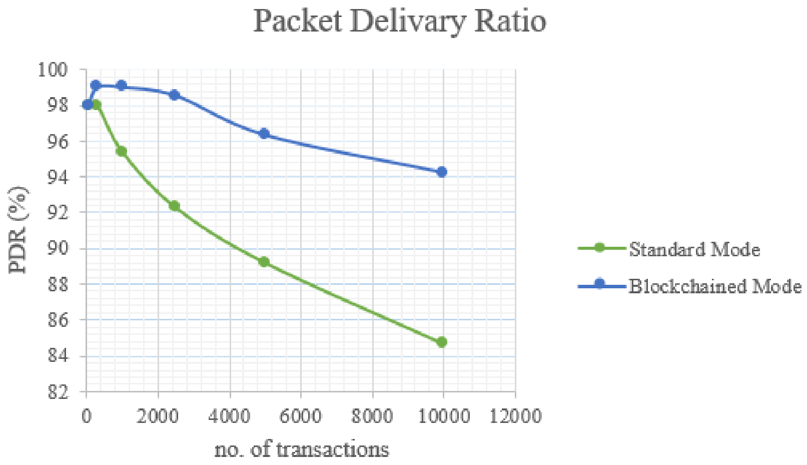


Figure 5.5: Average Packets Delivery Ratio

1 reading per day. The data in Tables 5.1 and 5.2 reflects a linear increase, as the number of devices increases linearly, the storage consumption overhead also increases linearly. Furthermore, the comparison between the two tables shows the impact of read frequency on storage consumption. Higher read frequencies result in significantly higher storage consumption overhead.

Table 5.1: Devices Storage Consumption Overhead for 4 Read/Day

# of Devices	Storage Consumption Overhead / Period (KB)		
	1 day	1 month	1 year
10	3.91	117.19	1406.25
20	7.81	234.38	2812.50
30	11.72	351.56	4218.75
40	15.63	468.75	5625.00

Table 5.2: Devices Storage Consumption Overhead for 1 Read/Day

# of Devices	Storage Consumption Overhead / Period (KB)		
	1 day	1 month	1 year
10	0.98	29.30	351.56
20	1.95	58.59	703.13
30	2.93	87.89	1054.69
40	3.91	117.19	1406.25

Storage Consumed by a SM

Table 5.3 and 5.4 present the storage consumption overhead of SMs in the LAN and HAN layers. The overhead is measured in KB over different periods by changing the number of SMs while the number of devices attached to each SM is fixed to 40 devices for simulation analysis purposes. The scenarios considered where each SM_i takes 4 readings per day for Table 5.3 and 1 reading per day for Table 5.4. The data in Tables 5.3 and 5.4 reflects the same relationship of devices storage consumption overhead with higher required storage because SM_i stores a copy of LAN_i and HAN_j Blockchains. Where as the number of SMs increases linearly, the storage consumption overhead also increases linearly. Furthermore, the comparison between the two tables shows the impact of read frequency on storage consumption. Higher read frequencies result in significantly higher storage consumption overhead.

Table 5.3: SMs Storage Consumption Overhead for 4 Read/Day

# of SMs	Storage Consumption Overhead / Period (KB)		
	1 day	1 month	1 year
50	35.16	1054.69	12656.25
100	54.69	1640.63	19687.50
150	74.22	2226.56	26718.75
200	93.75	2812.50	33750.00
250	113.28	3398.44	40781.25

Table 5.4: SMs Storage Consumption Overhead for 1 Read/Day

# of SMs	Storage Consumption Overhead / Period (KB)		
	1 day	1 month	1 year
50	8.79	263.67	3164.06
100	13.67	410.16	4921.88
150	18.55	556.64	6679.69
200	23.44	703.13	8437.50
250	28.32	849.61	10195.31

Storage Consumed by a DCU

Table 5.5 and 5.6 present the storage consumption overhead of DCUs in the HAN and NAN layers. The overhead is measured in KB over different periods by changing the number of DCUs while the number of SMS attached to each DCU is fixed to 250 SMs for simulation analysis purposes. The scenarios considered where each DCU_i takes 4 readings per day for Table 5.5 and 1 reading per day for Table 5.6. The data in Tables 5.5 and 5.6 aslo reflects a linear increase, where as the number of DCUs increases linearly, the storage consumption overhead also increases linearly. DCU_i should store a copy of HAN_i and NAN_j Blockchains. Furthermore, the comparison between the two tables shows the impact of read frequency on storage consumption. Higher read frequencies result in significantly higher storage consumption overhead.

Table 5.5: DCUs Storage Consumption Overhead for 4 Read/Day

# of DCUs	Storage Consumption Overhead / Period (KB)		
	1 day	1 month	1 year
5	99.61	2988.28	35859.38
10	101.56	3046.88	36562.50
15	103.52	3105.47	37265.63
20	105.47	3164.06	37968.75

Table 5.6: DCUs Storage Consumption Overhead for 1 Read/Day

# of DCUs	Storage Consumption Overhead / Period (KB)		
	1 day	1 month	1 year
5	24.90	747.07	8964.84
10	25.39	761.72	9140.63
15	25.88	776.37	9316.41
20	26.37	791.02	9492.19

5.3 Energy Consumption and Loss Analysis

This section presents an analysis of the simulation of energy consumed by SMs within the proposed system. The analysis focuses on scaling up the number of SMs in the network and its implication of energy consumption by changing the time interval of SM readings, both under normal conditions and in scenarios involving TL and NTL[18]. This analysis presents an indication of the energy and costs that can be saved by detecting losses using the proposed system.

5.3.1 Energy Consumption Readings

Table 5.7 presents an analysis of energy consumption in a network of SMs in normal operating conditions, based on simulated data. For simulation, the average hourly usage of home energy measured by SM is assumed to be 0.177 Kwh (177wh) as residential usage. The data reflects the total energy consumption across different reading time intervals: 6 hours, 12 hours, and 24 hours, for varying numbers of SMs, ranging from 50 to 10,000 SMs in the network as shown also in Figure 5.6.

Table 5.7: Total Energy Consumption at Different Reading Intervals

Number of SMs	Energy Consumption in Kwh		
	6 hours	12 hours	24 hours
50	37.72	75.67	150.88
300	273.39	546.74	1093.12
1000	835.95	1853.30	4404.27
5000	4152.44	6188.73	21803.96
10000	12022.48	21654.94	36756.94

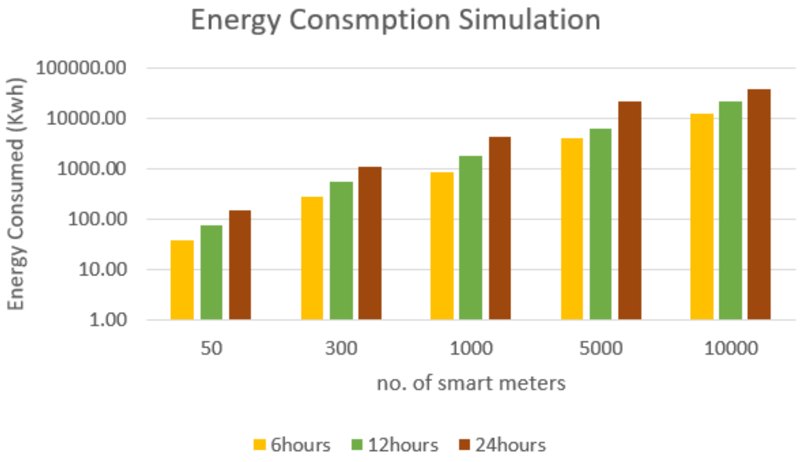


Figure 5.6: Energy Consumption Simulation

5.3.2 Experimental Energy Loss Detection

To further analyze the efficiency of the proposed system, an additional simulation was conducted by supposing that 4% of the SMs were assumed to experience energy NTL in the range of 17% to 35% of the average normal usage and 8% for TL [18]. These losses could be due to various factors, including external tampering and energy theft. The simulation aims to assess how such losses impact the total energy consumption recorded by the system.

Numeric Simulation Analysis

Table 5.8 presents a numeric simulation of energy loss detection of a smart grid consisting of 10000 SMs, by supposing that 400 SMs have energy NTL, with an average hourly energy consumption of the SM assumed to be 0.177 Kwh. The cost of NTL is calculated depending on the residential permitted tariff of Hebron Electric Power Co. [86] for 0.5065 ILS for each 1.0 Kwh. The simulation shows 5 cases of energy consumption readings in 5 predefined intervals: 120 reads per month (a read every 6 hours), 60 reads per month (a read every 12 hours), 30 reads per month, two reads per month, and one read per month. Table 5.8 also shows the total SM readings, total device readings, the considered TL of these readings the NTL detected using the proposed system, and the cost of NTL. To increase the scope of the numeric analysis of energy loss impact, Table 5.9 shows the readings in 4 cases in 4 predefined intervals: 4 reads per year (every 3 months), 3 reads per year, 2 reads and one read per year. The results show that the cost of NTL is highly increasing by widening the read period.

Table 5.8: Energy Losses Impact Month Scope

Reading Type	Reads per Month in Kwh				
	120	60	30	2	1
Total SM Readings	1326.11	2654.79	5307.28	79635.23	159142.01
Total Devices Readings	1121.61	2243.23	4486.45	67296.78	134593.56
TL	89.73	179.46	358.92	5383.74	10767.48
NTL	114.77	232.10	461.92	6954.71	13780.97
COST of NTL in ILS	58.13	117.56	233.96	3522.56	6980.06

Table 5.9: Energy Losses Impact Year Scope

Reading Type	Reads per Year in Kwh			
	4	3	2	1
Total SM Readings	476982.73	636461.12	969207.42	1935189.72
Total Devices Readings	403780.67	538374.23	818777.48	1637554.96
TL	32302.45	43069.94	65502.20	131004.40
NTL	40899.60	55016.95	84927.74	166630.37
COST of NTL in ILS	20715.65	27866.09	43015.90	84398.28

Energy and Cost Savings

The energy that can be saved when using the proposed system in the process of detecting energy losses increases as the time intervals of reading the SMs' consumption are reduced. So, depending on the numerically simulated data provided in tables 5.8 and 5.9, if the time interval of reading was 1 read per month like the traditional systems, the cost of energy losses would be much higher compared with 120 reads per month in the proposed system. The simulation analysis shows that the impact of detecting energy losses is visible in large grids, based on the percentage of the number of SMs that contain NTLs in addition to the average hourly energy consumption for each SM.

5.4 Summary

This chapter provided a simulation of the proposed system, which integrates the IoT and Blockchain technologies for enhanced energy monitoring and loss detection in smart metering systems in addition to the main purpose of Blockchain to prevent unauthorized access to the data in the system. The results from the simulation were highly promising. The system showed a significant improvement in energy consumption monitoring and a high delivery ratio of the readings. The integration of Blockchain technology played a crucial role in enhancing the reliability of the system by preventing data tampering and detecting losses in the lower layers like the LAN layer. The system encountered certain challenges, particularly in aspects such as real-time data processing and packet loss ratio. These challenges were effectively addressed through innovative Smart Contract solutions. In addition, the simulation provided an analysis for storage consumption over head and numeric study for an experiment of energy loss detection of a smart grid that contains 10000 SMs and the energy and cost savings by reduction of the read time intervals.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

This thesis is focused on developing a Smart Metering System that integrates Blockchain and IoT technologies to improve energy monitoring and management, enabling real-time data collection, energy loss detection, and security preservation. In this thesis, we studied the SGs and the challenges facing them. Then we surveyed the recent solutions for security preservation, efficiency improvement, and loss detection. Then, we proposed a new Blockchained- IoT Smart Metering System with a layers architecture by enabling two-way and real-time communication. The architecture of the proposed system contains four basic layers: LAN, HAN, NAN, and WAN, designed to monitor the energy consumed by system components efficiently. Also, The proposed system enables Ethereum Blockchain and Smart Contracts for LAN, HAN, and NAN layers to ensure efficiency and privacy preservation. Moreover, our proposal focuses on real-time monitoring and energy loss detection on the lower layers at pre-defined time intervals. Which helps in energy loss control, and increases energy and cost savings. The proposed system has been implemented and simulated in two scenarios and the system performance has been measured in terms of latency, packet delivery ratio storage consumption overhead. In addition, a numeric analysis is provided to show the proposed system's simulated energy consumption readings and the energy and cost that can be saved by enabling the proposed system and energy loss detection proposal in different scopes.

In summary, this thesis contributes to the ongoing research on Smart Metering Systems by proposing a new Smart Metering System leveraging the functionalities of Blockchain and IoT technologies to ensure the reliability, efficiency, and security of Smart Metering Systems within SGs.

6.2 Future Work

For future work improvements, the fusion of Artificial Intelligence (AI) with the proposed Blockchain and IoT Smart Metering System can hold greater potential for enhancing the detection of energy losses. By leveraging AI-driven algorithms, the proposed system can continuously monitor energy consumption patterns in real-time. Then the system can instantly identify abnormal consumption patterns, enabling prompt action to mitigate losses. Also, developing Smart Contracts to be embedded with AI-driven algorithms for autonomously analyzing energy consumption data. These Smart Contracts can execute predefined actions based on the outcomes of AI analysis, such as triggering alerts or adjusting energy parameters to control energy consumption and loss.

Bibliography

[1] GeeksforGeeks. Introduction to merkle tree. https://www.geeksforgeeks.org/introduction-to-merkle-tree/. Accessed: 2024-02-10.

[2] Masashi Sato and Shin'ichiro Matsuo. Long-term public blockchain: Resilience against compromise of underlying cryptography. In *2017 26th International Conference on Computer Communication and Networks (ICCCN)*, pages 1–8. IEEE, 2017.

[3] Don Johnson, Alfred Menezes, and Scott Vanstone. The elliptic curve digital signature algorithm (ecdsa). *International journal of information security*, 1:36–63, 2001.

[4] A. A. G. Agung and R. Handayani. Blockchain for smart grid. *Journal of King Saud University - Computer and Information Sciences*, 34(3):666–675, 2022.

[5] Chrystal R. China. Optimizing energy production with the latest smart grid technologies. https://www.ibm.com/blog/optimizing-energy-production-with-the-latest-smart-grid-technologies/, 2023. Accessed: 2023-08-02.

[6] Rehmat Ullah, Yasir Faheem, and Byung-Seo Kim. Energy and congestion-aware routing metric for smart grid ami networks in smart city. *IEEE Access*, 5:13799–13810, 2017.

[7] F. E. Abrahamsen, Y. Ai, and M. Cheffena. Communication technologies for smart grid: A comprehensive survey. *Sensors*, 21(23), 2021.

[8] E. Dbabseh and R. Tahboub. Framework for securing automatic meter reading using blockchain technology. In *ITNG 2021 18th International Conference on Information Technology-New Generations*. IEEE, 2021.

[9] M. Faheem, S. Shah, R. Butt, B. Raza, M. Anwar, M. Ashraf, M. Ngadi, and V. Gungor. Smart grid communication and information technologies in the perspective of industry 4.0: Opportunities and challenges. *Computer Science Review*, 30:1–30, 2018.

[10] H. Tian, Y. Jian, and X. Ge. Blockchain-based ami framework for data security and privacy protection. *Sustainable Energy, Grids and Networks*, 32:100807, 2022.

[11] R. Rashed Mohassel, A. Fung, F. Mohammadi, and K. Raahemifar. A survey on advanced metering infrastructure. *International Journal of Electrical Power Energy Systems*, 63:473–484, 2014.

[12] N. Islam, M. S. Rahman, I. Mahmud, M. N. A. Sifat, and Y.-Z. Cho. A blockchain-enabled distributed advanced metering infrastructure secure communication (bc-ami). *Applied Sciences*, 12(14), 2022.

[13] A. S. M. Tayeen, M. Biswal, and S. Misra. Dp-ami-fl: Secure framework for machine learning-based ami applications. In *2023 IEEE Power Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, pages 1–5, 2023.

[14] M. Babar, M. U. Tariq, and M. A. Jan. Secure and resilient demand side management engine using machine learning for iot-enabled smart grid. *Sustainable Cities and Society*, 62:102370, 2020.

[15] N. Abosata, S. Al-Rubaye, and G. Inalhan. Lightweight payload encryption-based authentication scheme for advanced metering infrastructure sensor networks. *Sensors*, 22:534, 2022.

[16] Hubbell. Leveraging ami systems for analytics and customer engagement. https://blog.hubbell.com/en/aclara/leveraging-ami-systems-for-analytics-and-customer-engagement, 2021. Accessed: 2023-10-25.

[17] Sabryna V. Fernandes, Diogo V. João, Beatriz B. Cardoso, Marcos A. I. Martins, and Edgar G. Carvalho. Digital twin concept developing on an electrical distribution systemmdash;an application case. *Energies*, 15(8), 2022.

[18] Kristina Sadovskaia, Dmitrii Bogdanov, Samuli Honkapuro, and Christian Breyer. Power transmission and distribution losses – a model based on available empirical data and future trends for all countries globally. *International Journal of Electrical Power & Energy Systems*, 107:98–109, 2019.

[19] Reinhard Günther. Technical vs non-technical losses: What you need to know. https://clouglobal.com/what-is-the-difference-between-technical-loss-and-non-technical-loss/, 2023. Accessed: 2023-11-16.

[20] NEDCO. Northern electricity is the company that loses the least electricity and is the best-selling company in palestine. https://nedco.ps/?ID=484, 2012. Accessed on 2023-09-23.

[21] WAFA. A workshop discusses the phenomenon of attacks on electricity networks. https://www.wafa.ps/Pages/Details/56020, 2022. Accessed on 2023-09-23.

[22] Ajit Muzumdar, Chirag Modi, and C. Vyjayanthi. Designing a blockchain-enabled privacy-preserving energy theft detection system for smart grid neighborhood area network. *Electric Power Systems Research*, 207:107884, 2022.

[23] Tehseen Mazhar, Hafiz Muhammad Irfan, Sunawar Khan, Inayatul Haq, Inam Ullah, Muhammad Iqbal, and Habib Hamam. Analysis of cyber security attacks and its solutions for the smart grid using machine learning and blockchain methods. *Future Internet*, 15(2), 2023.

[24] J. Kua, M. B. Hossain, I. Natgunanathan, and Y. Xiang. Privacy preservation in smart meters: Current status, challenges and future directions. *Sensors (Basel)*, 23(7):3697, 2023.

[25] M. AlHawamdeh and R. Tahboub. Blockchain-iot enabled smart metering system. In *2023 International Conference on Electrical and Information Technology (IEIT)*, pages 207–213, 2023.

[26] Ihsan Ali, Ismail Ahmedy, Abdullah Gani, Muhammad Umair Munir, and Mohammad Hossein Anisi. Data collection in studies on internet of things (iot), wireless sensor networks (wsns), and sensor cloud (sc): Similarities and differences. *IEEE Access*, 10:33909–33931, 2022.

[27] Sudip Misra, Anandarup Roy, and Arpan Mukherjee. *Introduction to Industrial Internet of Things and Industry 4.0*. Routledge, 2020.

[28] GeeksforGeeks. Blockchain structure. https://www.geeksforgeeks.org/blockchain-structure/, 2022. Accessed: 2023-11-16.

[29] Ahmed Afif Monrat, Olov Schelén, and Karl Andersson. A survey of blockchain from the perspectives of applications, challenges, and opportunities. *IEEE Access*, 7:117134–117151, 2019.

[30] Huaqun Guo and Xingjie Yu. A survey on blockchain technology and its security. *Blockchain: Research and Applications*, 3(2):100067, 2022.

[31] Muhammad Salek Ali, Massimo Vecchio, Miguel Pincheira, Koustabh Dolui, Fabio Antonelli, and Mubashir Husain Rehmani. Applications of blockchains in the internet of things: A comprehensive survey. *IEEE Communications Surveys Tutorials*, 21(2):1676–1717, 2019.

[32] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system. https://bitcoin.org/bitcoin.pdf/, 2011. Accessed 2022-12-31.

[33] Nashreen Nesa and Indrajit Banerjee. *A Lightweight Security Protocol for IoT Using Merkle Hash Tree and Chaotic Cryptography*, pages 3–16. Springer Singapore, Singapore, 2020.

[34] Vi Buterin. Ethereum white paper: A next generation smart contract & decentralized application platform. https://ethereum.org/en/whitepaper/, 2013. Accessed 2022-12-31.

[35] Yujian Zhang and Daifu Liu. Toward vulnerability detection for ethereum smart contracts using graph-matching network. *Future Internet*, 14(11), 2022.

[36] Satpal Singh Kushwaha, Sandeep Joshi, Dilbag Singh, Manjit Kaur, and Heung-No Lee. Systematic review of security vulnerabilities in ethereum blockchain smart contract. *IEEE Access*, 10:6605–6621, 2022.

[37] Kasun Weranga, Sisil Kumarawadu, and D. P. Chandima. *Smart Grid and Smart Metering*, pages 1–15. Springer Singapore, Singapore, 2014.

[38] José Manuel Carou Álvarez and Lucía Suárez Ramón. Smart meters. In Jorge García, editor, *Encyclopedia of Electrical and Electronic Power Engineering*, pages 441–447. Elsevier, Oxford, 2023.

[39] Ivan Popović, Aleksandar Rakić, and Ivan D. Petruševski. Multi-agent real-time advanced metering infrastructure based on fog computing. *Energies*, 15(1), 2022.

[40] Yasin Kabalci. A survey on smart metering and smart grid communication. *Renewable and Sustainable Energy Reviews*, 57:302–318, 2016.

[41] Dong Sik Kim, Beom Jin Chung, and Young Mo Chung. Analysis of ami communication methods in various field environments. *Energies*, 13(19), 2020.

[42] Rajini Kanth Reddy Karduri and Christo Ananth. Sustainable urban energy: Integrating smart grids into smart cities. *International Journal of Advanced Research In Basic Engineering Sciences and Technology (IJARBEST)*, 6(2):7, 2023. Posted: 19 Dec 2023.

[43] T.N. Bhattarai, S. Ghimire, B. Mainali, et al. Applications of smart grid technology in nepal: status, challenges, and opportunities. *Environmental Science and Pollution Research*, 30:25452–25476, 2023.

[44] Imperva. What is low orbit ion cannon (loic)? https://www.imperva.com/learn/ddos/low-orbit-ion-cannon/. Accessed on 2023-01-13.

[45] M. Tahir, N. Ismat, H. H. Rizvi, A. Zaffar, S. M. Nabeel Mustafa, and A. A. Khan. Implementation of a smart energy meter using blockchain and internet of things: A step toward energy conservation. *Frontiers in Energy Research*, 10, 2022.

[46] C. Pop, T. Cioara, M. Antal, I. Anghel, I. Salomie, and M. Bertoncini. Blockchain based decentralized management of demand response programs in smart energy grids. *Sensors*, 18(1), 2018.

[47] C. Zakaret, N. Peladarinos, V. Cheimaras, E. Tserepas, P. Papageorgas, M. Aillerie, D. Piromalis, and K. Agavanakis. Blockchain and secure element, a hybrid approach for secure energy smart meter gateways. *Sensors*, 22(24), 2022.

[48] T. Schlap̆fer and A. Rus̆t. Security on iot devices with secure elements. In *Embedded World Conference 2019 - Proceedings*, 2019.

[49] BigchainDB. Features of bigchaindb. https://www.bigchaindb.com/features/. Accessed on 2023-02-27.

[50] Helium. Helium technology. https://www.helium.com/technology. Accessed on 2023-03-06.

[51] MULTOS. Multos trust core developer boards. https://multos.com/support/multos-trust-anchor/developer-boards/trust-core-details/. Accessed on 2023-03-06.

[52] U. GmbH. Ubirch sim tutorial. https://ubirch.com/digital-corona-lab-certificate-1/ubirch-sim-tutorial/. Accessed on 2023-04-06.

[53] M. Graf, R. Küsters, and D. Rausch. Accountability in a permissioned blockchain: Formal analysis of hyperledger fabric. In *2020 IEEE European Symposium on Security and Privacy (EuroSP)*, pages 236–255, 2020.

[54] O. Onireti, L. Zhang, and M. A. Imran. On the viable area of wireless practical byzantine fault tolerance (pbft) blockchain networks. In *2019 IEEE Global Communications Conference (GLOBECOM)*, pages 1–6, 2019.

[55] Secure and resilient demand side management engine using machine learning for iot-enabled smart grid. *Sustainable Cities and Society*, 62:102370, 2020.

[56] M. Nabil, M. Ismail, M. M. E. A. Mahmoud, W. Alasmary, and E. Serpedin. Ppetd: Privacy-preserving electricity theft detection scheme with load monitoring and billing for ami networks. *IEEE Access*, 7:96334–96348, 2019.

[57] J. C. Choon and J. Hee Cheon. An identity-based signature from gap diffie-hellman groups. In Y. G. Desmedt, editor, *Public Key Cryptography — PKC 2003*, pages 18–30, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg.

[58] Y. Huang, D. Evans, J. Katz, and L. Malka. Faster secure two-party computation using garbled circuits. In *Proceedings of the 17th ACM conference on Computer and communications security*, pages 124–133. ACM, 2010.

[59] Y. LeCun, Y. Bengio, et al. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 3361(10):1995, 1995.

[60] A. Alsharif, M. Nabil, M. Mahmoud, and M. Abdallah. Privacy-preserving collection of power consumption data for enhanced ami networks. In *2018 25th International Conference on Telecommunications (ICT)*, pages 196–201. IEEE, 2018.

[61] Sidra Abbas, Imen Bouazzi, Stephen Ojo, Gabriel Avelino Sampedro, Ahmad S. Almadhor, Abdullah Al Hejaili, and Zuzana Stolicna. Improving smart grids security: An active learning approach for smart grid-based energy theft detection. *IEEE Access*, 12:1706–1717, 2024.

[62] S. Zidi, A. Mihoub, S. M. Qaisar, M. Krichen, and Q. A. Al-Haija. Theft detection dataset for benchmarking and machine learning based classification in a smart grid environment. *Journal of King Saud University - Computer and Information Sciences*, 35(1):13–25, Jan 2023.

[63] Xinwu Sun, Jiaxiang Hu, Zhenyuan Zhang, Di Cao, Qi Huang, Zhe Chen, and Weihao Hu. Electricity theft detection method based on ensemble learning and prototype learning. *Journal of Modern Power Systems and Clean Energy*, pages 1–18, 2024.

[64] Parsa Sarosh, Shabir A Parah, and Ghulam Mohiuddin Bhat. Utilization of secret sharing technology for secure communication: a state-of-the-art review. *Multimedia Tools and Applications*, 80:517–541, 2021.

[65] E Soria Vazquez. *Towards secure multi-party computation on the internet: Few rounds and many parties.* PhD thesis, Ph. D. thesis, Univ. Bristol, Bristol, UK, 2019.

[66] Marcel Keller, Valerio Pastro, and Dragos Rotaru. Overdrive: making spdz great again. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 158–189. Springer, 2018.

[67] Caixiang Fan, Changyuan Lin, Hamzeh Khazaei, and Petr Musilek. Performance analysis of hyperledger besu in private blockchain. In *2022 IEEE international conference on decentralized applications and infrastructures (DAPPS)*, pages 64–73. IEEE, 2022.

[68] Hyperledger Foundation. Hyperledger besu. https://www.hyperledger.org/projects/besu. Accessed: 2023-12-27.

[69] Danny Khoury and Fakheredine Keyrouz. A predictive convolutional neural network model for source-load forecasting in smart grids. *WSEAS Transactions on Power Systems*, 14:181–189, 2019.

[70] Alex Sherstinsky. Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network. *Physica D: Nmiscar Phenomena*, 404:132306, 2020.

[71] Saddam Hussain, Mohd Wazir Mustafa, Khalil Hamdi Ateyeh Al-Shqeerat, Bander Ali Saleh Al-rimy, and Faisal Saeed. Electric theft detection in advanced metering infrastructure using jaya optimized combined kernel-tree boosting classifier—a novel sequentially executed supervised machine learning approach. *IET Generation, Transmission & Distribution*, 16(6):1257–1275, 2022.

[72] SR Aishwarya, V Gayathri, R Janani, Kannan Pooja, and Mathi Senthilkumar. A framework for identifying theft detection using multiple-instance learning. In *International Conference on Soft Computing and Signal Processing*, pages 55–67. Springer, 2022.

[73] Joan Daemen and Vincent Rijmen. Aes proposal: Rijndael. *Gaithersburg, MD, USA*, 1999.

[74] Mehdi-Laurent Akkar and Christophe Giraud. An implementation of des and aes, secure against some attacks. In *Cryptographic Hardware and Embedded Systems—CHES 2001: Third International Workshop Paris, France, May 14–16, 2001 Proceedings 3*, pages 309–318. Springer, 2001.

[75] Zhengyi Lu. Analysis on aes encryption standard and safety. In *Third International Symposium on Computer Engineering and Intelligent Communications (ISCEIC 2022)*, volume 12462, pages 292–297. SPIE, 2023.

[76] Cheah Wai Zhao, Jayanand Jegatheesan, and Son Chee Loon. Exploring iot application using raspberry pi. *International Journal of Computer Networks and Applications*, 2(1):27–34, 2015.

[77] Da-Wen Huang, Wanping Liu, and Jichao Bi. Data tampering attacks diagnosis in dynamic wireless sensor networks. *Computer Communications*, 172:84–92, 2021.

[78] Fadele Ayotunde Alaba, Hakeem Adewale Sulaimon, Madu Ifeyinwa Marisa, and Owamoyo Najeem. Smart contracts security application and challenges: A review. *Cloud Computing and Data Science*, pages 15–41, 2024.

[79] Chris Dannen. *Introducing Ethereum and solidity*, volume 1. Springer, 2017.

[80] Neal Koblitz. Elliptic curve cryptosystems. *Mathematics of Computation*, 48(177):203–209, 1987.

[81] Rodrigo Román-Castro, Javier López, and Stefanos Gritzalis. Evolution and trends in iot security. *Computer*, 51(7):16–25, 2018.

[82] Johannes Bauer, Ralf C Staudemeyer, Henrich C Pöhls, and Alexandros Fragkiadakis. Ecdsa on things: Iot integrity protection in practise. In *Information and Communications Security: 18th International Conference, ICICS 2016, Singapore, Singapore, November 29–December 2, 2016, Proceedings 18*, pages 3–17. Springer, 2016.

[83] Zhen Gao, Dongbin Zhang, and Jiuzhi Zhang. A security problem in small scale private ethereum network. In Zibin Zheng, Hong-Ning Dai, Xiaodong Fu, and Benhui Chen, editors, *Blockchain and Trustworthy Systems*, pages 231–242, Singapore, 2020. Springer Singapore.

[84] Remix. Welcome to Remix's documentation! — Remix - Ethereum IDE 1 documentation. https://remix-ide.readthedocs.io/en/latest/, 2023.

[85] Truffle Suite. Ganache documentation: Quickstart. https://trufflesuite.com/docs/ganache/quickstart/. Accessed on 2023-03-17.

[86] HEPCO. Sales tariff. https://www.hepco-pal.com/en/services/sales-tariff. Accessed: 2024-01-02.

Appendix A

Energy Consumption Data Sample

Table A.1: Smart Meter Consumption Data

Smart Meter ID	Monthly AVG Consumption Kwh	Kwh/ day
00000001	84.29	2.81
00000002	57.68	1.92
00000003	11.92	0.40
00000004	23.75	0.79
00000005	59.03	1.97
00000006	5.29	0.18
00000007	131.81	4.39
00000008	59.58	1.99
00000009	39.96	1.33
00000010	141.46	4.72
00000011	49.75	1.66
00000012	36.13	1.20
00000013	127.08	4.24
00000014	9.56	0.32
00000015	8.44	0.28
00000016	24.56	0.82
00000017	35.71	1.19
00000018	66.23	2.21
00000019	410.22	13.67
00000020	94.71	3.16
00000021	89.61	2.99
00000022	124.21	4.14
00000023	52.48	1.75
00000024	156.69	5.22

Continued on next page

Table A.1 – continued from previous page

Smart Meter ID	Monthly AVG Consumption Kwh	Kwh / day
00000025	103.59	3.45
00000026	42.87	1.43
00000027	84.1	2.80
00000028	10.94	0.36
00000029	59.61	1.99
00000030	213.41	7.11
00000031	13.44	0.45
00000032	19.4	0.65
00000033	79.71	2.66
00000034	204.71	6.82
00000035	16.21	0.54
00000036	273.88	9.13
00000037	64.9	2.16
00000038	20.01	0.67
00000039	72.61	2.42
00000040	165.77	5.53
00000041	119.07	3.97
00000042	136.21	4.54
00000043	472.77	15.76
00000044	89.37	2.98
00000045	79.96	2.67
00000046	61.82	2.06
00000047	58.22	1.94
00000048	10.03	0.33
00000049	22.52	0.75
00000050	130.95	4.37
00000051	33.35	1.11
00000052	42.28	1.41
00000053	11.18	0.37
00000054	74	2.47
00000055	103.35	3.45
00000056	12.47	0.42
00000057	3.76	0.13
00000058	28.97	0.97
00000059	90.31	3.01
00000060	144.16	4.81
00000061	93.17	3.11
00000062	108.95	3.63
00000063	8.82	0.29
00000064	19.89	0.66

Continued on next page

Table A.1 – continued from previous page

Smart Meter ID	Monthly AVG Consumption Kwh	Kwh / day
00000065	29.35	0.98
00000066	29.14	0.97
00000067	5.07	0.17
00000068	175.73	5.86
00000069	12.76	0.43
00000070	32.68	1.09
00000071	1.39	0.05
00000072	42.65	1.42
00000073	47.06	1.57
00000074	6.44	0.21
00000075	15.6	0.52
00000076	28.65	0.96
00000077	7.4	0.25
00000078	1100.65	36.69
00000079	6.9	0.23
00000080	91.61	3.05
00000081	249.22	8.31
00000082	3.43	0.11
00000083	96.98	3.23
00000084	8.54	0.28
00000085	91.55	3.05
00000086	25.83	0.86
00000087	3.54	0.12
00000088	37.64	1.25
00000089	3.7	0.12
00000090	45.73	1.52
00000091	0.92	0.03
00000092	1.42	0.05
00000093	6.21	0.21
00000094	12.98	0.43
00000095	44.14	1.47
00000096	21.14	0.70
00000097	10.42	0.35
00000098	84.09	2.80
00000099	11	0.37
00000100	45.65	1.52

Table A.1: Smart Meter Consumption Data