Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 4401809, 23 pages
https://doi.org/10.1155/2021/4401809

Research Article

Data Integrity Time Optimization of a Blockchain IoT Smart Home Network Using Different Consensus and Hash Algorithms

Ammar Riadh Kairaldeen, Nor Fadzilah Abdullah, Asma Abu-Samah, and Rosdiadee Nordin

Department of Electrical, Electronic and Systems Engineering, Faculty of Engineering & Built Environment, Universiti Kebangsaan Malaysia, 43600 UKM Bangi, Selangor, Malaysia

Correspondence should be addressed to Ammar Riadh Kairaldeen; aaltotanje@gmail.com

Received 27 June 2021; Accepted 5 October 2021; Published 9 November 2021

Academic Editor: Ruhui Ma

Data security is a major issue for smart home networks. Yet, different existing tools and techniques have not been proven highly effective for home networks' data security. Blockchain is a promising technology because of the distributed computing infrastructure network that makes it difficult for hackers to intrude into the systems through the use of cryptographic signatures and smart contracts. In this paper, an architecture for smart home networks that could guarantee data integrity, robust security, and the ability to protect the validity of the blockchain transactions has been investigated. The system model is tested using various sizes of realistic datasets (30, 3 k, and 30 k to represent a small, medium, and large number of transactions, respectively). Four different consensus algorithms were considered, the conventional schemes concatenated hash transactions (CHT) and Merkle hash tree (MHT), as well as the newly proposed odd and even modified MHT (O&E MHT) and modified MHT (MMHT). Moreover, 15 hash functions were also examined and compared to understand the effects of each consensus algorithms on the data integrity verification check execution time and the time optimization provided by the proposed MMHT algorithm. The results show that even though the CHT algorithm gives the lowest execution time, it is impractical for a blockchain implementation due to the requirement to copy the entire blockchain ledger in real time. Meanwhile, the O&E MHT does not give any tangible benefit in the execution time. However, the proposed MMHT offers a minimum of 30% gain in time optimization than the conventional MHT algorithm typically used in blockchains. This work shows that the proposed MMHT consensus algorithm not only can identify malicious codes but has an improved data integrity check performance in smart homes, all while ensuring network stability.

1. Introduction

A large communication network of smart devices, sensors, and other consumer electronics such as a TV, refrigerator, and air conditioner in a home area network (HAN) has made communication and interaction among themselves very complicated and complex. Therefore, the communication between these devices in the network needs to ensure high security of data so that these systems' users can be provided with a high degree of privacy [1].

The changes brought by the Fourth Industrial Revolution have enabled the Internet of things (IoT) to play a significant role in bridging the gap between each of the physical industrial environments and cyberspace of computing systems. This requires multiple interconnected systems with unique identities that can communicate and interact with each another and transfer data over the network without requiring human-to-human or human-to-computer interaction, unlike the current case of physical industrial environments. In addition to this, the use of other advanced technologies such as artificial intelligence (AI), big data analytics (BDA), machine learning (ML), and other emerging tools helped in utilizing collected data effectively through different sources in the network. Therefore, through this practice, the processed data can be used to improve system efficiency and performance [2, 3]. To accomplish a highly

interactive, efficient but secure network, various elements and factors such as data privacy, authentication, ease of use and maintenance, and high security standards against possible attacks are needed. These robust and advanced features are possible using blockchain technology in the IoT system.

Various types of blockchain are used depending on the different elements and features of the system in consideration. One of the core features of blockchain-based IoT is authentication. The use of this feature helps strengthen the network against potential attacks from hackers. Also, the accessibility of data is another factor that defines the safety of data used by the system. Thus, different blockchain types are discussed further to examine their characteristics and effectiveness against data integrity threats. In general, there are two types of blockchain, namely, *permissionless* and *permissioned* [4].

A *permissionless* blockchain is a popular type of blockchain that allows anyone to participate in the network. This type of blockchain does not require participators to be authorized to be active in the network. Anyone can join the network and use their computational powers to contribute to the system. An example of a permissionless blockchain is a Bitcoin network that allows everyone to enter the system without any prior authorization. Therefore, participation is encouraged by granting entry into the network without the need for authorization. A participant's task is to ensure performance by verifying the network operation. These verifiers are important for the network as they enhance its operation. Hence, different algorithms are used by the verifiers.

The second type of blockchain is the *permissioned* blockchain, which requires the verifiers to gain authorization before taking part in the network. Verification nodes are set by a central authority, ensuring that the verifier should ask for permission before becoming involved in the blockchain. Permissioned blockchain can be further classified into *private* and *public* blockchains. Public blockchain allows anyone to read and submit the transactions, while private blockchain restricts the right to users of the organization to become involved in the network.

There are several fundamental differences between the permissionless and permissioned blockchains. These include their way of operation and the range of activities that they can perform [5]. Each of these blockchains has different benefits and limitations. One of the benefits of permissioned blockchains is scalability, as they allow the verification of all the transactions performed by the nodes. On the other hand, the permissionless blockchain provides the benefit of being highly resistant to the changes in the blocks using a single verifier. This capacity of the blockchain is highly beneficial in keeping data safe and secure. Therefore, this type of blockchain is very applicable to the HANs, making them safer for use. The large user base of HANs can find their data safe using permissionless blockchain that will prevent transactions without their consensus. Nevertheless, permissioned blockchains can also be used by the service providers who can act as a central authority to provide authorization to the users based on their requirements. Thus, through this practice, all the transactions can be monitored and controlled by the central authority. However, it has a disadvantage in that it can be very challenging to monitor and to control many transactions, which can make the networks less efficient.

Nowadays, the use of second-generation blockchains, including smart contracts, is on the rise in the industry [6, 7]. However, before the wide-scale adoption of smart contracts occurs, various factors must be considered in keeping the smart home applications private and secure. One of these factors includes the blockchain system that continuously monitors all activities from any intrusion [8]. Scalability can be very challenging as every node needs to process every transaction in the blockchain [9], resulting in a higher execution time cost required for the validator due to enormous computational power [10, 11]. Secondly, code correctness is an issue as both the developers and users must be confident about smart contracts and must not employ high fees for unnecessary computations [12]. In addition to this, the concern is about using a suitable authentication algorithm with a suitable hash function in the blockchain system [13].

In this work, the proof-of-work (PoW) is the selected algorithm to provide data integrity for smart homes [14]. The consensus algorithm is based on PoW that secures the network via the validator, which can be one or more participant nodes to verify the accomplished and submitted work, allowing them to add new transactions to the blockchain as it does not allow a single verifier to make changes to the entire network. The concept of consensus among most of the verifiers keeps the entire system safe from hackers' activity. Therefore, hackers must put in a large amount of time and money due to the massive computational efforts required. This effort has to be greater than all the power spent from the reference point that the hacker aims to change at a specific time [15]. The permissioned blockchains use different consensus protocols [16] to permit users to become authorized verifiers. Besides, this type of blockchain also uses a set of trusted parties to perform verification so that additional verifiers can become a part of the network. This can be achieved through the consensus of a current member and central authority of the blockchain. Financial settings are more likely to include this type of blockchain, which operates a Know Your Business (KYB) or Know Your Client (KYC) procedure to differentiate users that are allowed to undertake operations in a particular space [14].

This article has the following major contributions:

(i) Design of a smart home architecture using a blockchain-based technology on specific criteria to ensure performance and security

(ii) Selection of optimal consensus algorithms and authentication algorithms for smart homes, considering the security standards

(iii) Comparison between different hash functions to select the most suitable for adoption into the authentication algorithm

(iv) Propose a modified Merkle hash tree (MMHT) authentication algorithm to reduce the validation execution time

The rest of the paper is structured as follows. In Section 2, a brief background on IoT in a smart home area network,

data integrity in the blockchain system, highlighting standardizations used in security and policy, explanation on smart contract, consensus algorithms, authentication algorithms, hash functions, attacks, and smart home security attacks has been outlined. Section 3 presents the architecture components in the smart home ecosystem such as the server, hardware and software requirements, applications, smart contract structure, and design, illustrating the proposed design and architecture workflow. Section 4 describes the dataset used and the data preparation procedures. Section 5 explains the architecture of conventional and proposed consensus algorithms, the implementation steps of the proposed algorithms, and findings on the network's performance data integrity regarding transaction scalability and validation execution time. Finally, the paper is concluded in Section 6 with a detailed discussion on different issues involved in the proposed MMHT algorithm.

2. Related Works

2.1. IoT in Smart Home Area Networks. Over the architecture revolution, 5G networks promised to give credible schemes such as a high quality of service, ultralow latency, and high level of security demand [17]. Home area networks (HANs) are home-based networks that connect all the devices including computers, laptops, and smart appliances. This network is aimed at achieving energy-efficient smart homes by efficiently managing appliances and energy usage [18]. Therefore, the concept of smart homes relies on the application of HANs. HANs comprise various appliances integrated with the network and different sensing devices such as thermostats and smart meters. These sensors' primary objective is to collect data from these appliances and communicate it to the homeowners, utility providers, and other service providers. Therefore, this data flow is of key importance for HANs as it allows the homeowners and service providers to monitor and control the operation of the appliances and energy consumption. It is also important to note that most of this data communication occurs through different communication protocols such as Wi-Fi, RFID, and Zigbee [19]. However, most contemporary smart HANs are based on the Internet and Wi-Fi as they consume a large bandwidth of data transmission. Therefore, the high speed of Internet connections has made HANs very efficient and quick.

The integration of the smart devices and sensors with these networks has enabled the controllers to gain real-time information about various parameters including energy used and traffic load. Thus, the use of IoT in smart homes provides the stakeholders with a great opportunity to automate most appliances using smart systems. IoT is defined as the system of smart appliances and devices that can collect data and transfer it over the network without human interaction. Thus, the use of IoT in smart homes has reduced human interaction substantially as the appliances have become smart [20]. Smart appliances and devices are connected to the network in smart homes, which also comprise microcontrollers programmed to enable them to make decisions without human support. Thus, the entire IoT concept in smart HANs is based on collections and effective data use. The ability of IoT-based HANs to collect, transfer, and process data has proved to be beneficial for homeowners and service providers as energy consumption has been made significantly efficient [21]. However, these systems have also increased the threat of people's data security and privacy.

Data security lies at the core of smart home networks as it is very important to keep the information being sent over the network safe [22]. For this reason, data security experts have been using different security protocols and standards to make smart home networks safer and strong against cyberattacks. Irrespective of these measures, smart HANs are still prone to the threat of data security. Various issues including credit card fraud, identity theft, and virus attacks are common for smart homes [23]. Therefore, the service providers have been using security methods such as encryption and authentication to protect people's privacy and avoid any data theft and information leak. Nevertheless, intruders have been able to decrypt the data and communication over the HANs, which requires data security experts to look for new methods to mitigate this issue. Challenges such as the complexity of the network structure in HANs and the devices' heterogeneity are major barriers to applying highly efficient and effective security standards [24]. IoT for data security has enabled researchers and experts to develop smart data security protocols, which enable high protection of the data and privacy of smart homeowners.

The use of IoT in smart HANs for data security is based on monitoring and control mechanisms. IoT's major role in smart home networks is the monitoring and data control through the application of security protocols that monitor data and prevent any suspicious activity.

Previously, blockchain is normally implemented in the public network [25]. However, with IoT there is a tendency for adoption of blockchains in the private network. Using Blockchain in the private network makes it easy to connect different systems horizontally rather than work on vertical compatibility; this will have the advantage of being easily scalable and adding more applications while considering security requirements.

IoT-based networks provide the benefit that they do not require human support in case any malicious activity is encountered [26]. These systems are smart enough to stop intruders from injecting the virus or malicious code into the network. Sensors are the major player in IoT that work to monitor the data once the network is live. The real-time monitoring of data through IoT on smart networks enables high security of all the network gateways and communication media. The data sent and received from all the sources is effectively monitored to prevent any data security attack. Besides, IoT also ensures data storage safety and manages the devices' operation status to enforce high-security standards. Through these practices, IoT has proved to be very efficient in maintaining data integrity and keeping it safe from cyber criminals [20]. Therefore, due to these security protocols offered by IoT, it is gaining high popularity among security experts.

The use of IoT in smart HANs also ensures the high availability of data and network systems. The efficient

monitoring and control processes employed by IoT have enabled the systems to reduce their downtimes as any encountered issues are handled automatically or communicated instantly to the human operators. Besides, IoT maintains a high focus on data integrity and confidentiality by using different security methods and data encryption protocols [27]. IoT-based data encryption is safer in smart home networks as the systems are continuously monitored. Thus, any activity of third-party intrusion into the network can be immediately detected. Hence, IoT in smart home networks effectively increases the data and network systems' security. The increase in IoT devices and their broad-spectrum applications in houses provides advanced ways to keep data safe. The risks from lack of transparency, auditability, and accountability in HANs are being catered using IoT through efficient application in critical areas by staying within the legal domains [28].

Attacks are one of the major threats to information systems and networks. They harm the integrity and security of data, leading to negative effects for various stakeholders of the systems [29]. Therefore, the systems' vulnerability to potential attacks must be managed so that the network systems can work with high data security standards. Different types of attacks exist to target the information systems and network, as explained by [30]. Some of these critical and most popular security attacks are examined as follows:

2.1.1. Data Availability Attack. This type of attack will be defeated by the data validity algorithm [31]. Different types of data availability attacks, the response of smart contract, and explanation of data validity check algorithm are as follows:

(i) Malicious block attack: when a malicious block producer publishes a block to the blockchain, data validity will check the block inclusion to the blockchain and flag invalid transactions hash and show the fraud-proof status (attack status) to the system administrator

(ii) Denial-of-service attacks: when a system is aware of data unavailability, it will flag an alarm without needing any kind of proof in the blockchain

2.1.2. Access Control Impersonation Attack. In HAN impersonation attack comprises both user and device impersonation, which is a form of fraud caused by replay, message modification, etc. [15]. The malicious attackers pose as a known or trusted person and gain admin privileges before using the smart home IoT ecosystem to share sensitive information or liability of any vicious activities. For instance, attackers abnormally manipulate IoT devices (home appliances) and increase Sauna's temperature, which is connected to a HAN, thus risking people's lives at home using the facility.

In HAN impersonation attack, it is necessary to follow the standardization protocol of communication to avoid the lack of service, using various data communication standards to eliminate impersonation attack over a HAN (e.g., ZigBee and Wi-Fi) which does not affect the blockchains efficiency [32]. The main challenge for both efficiency and speed of ensuring is that all nodes are not involved in potential impersonation attacks or fraudulent behavior. Validator nodes usually work to check consensus in the blockchain network. However, this work requires enormous computational power from those validators, and hence, in this research, we are trying to efficiently consensus algorithms that reduce computational power by decreasing the execution time and enhancing speed.

2.1.3. Double-Spending Attack. Double-spending attacks are one of the most popularly used threats by hackers in PoW algorithms. This type of attack occurs when the user controls more than 50% of the computing power [33]. Therefore, they can send a fraudulent transaction log to the network, enabling them to perform the same transaction multiple times by removing the record of previous transactions. However, this attack is not very easy to execute; but, it can be very harmful if hackers accomplish it.

2.1.4. Side-Channel Attack. The Merkle tree-based algorithm is also vulnerable to side-channel attacks, which reduces the integrity of data. A side-channel attack targets the authentication process, which reduces its reliability and effectiveness [5]. This type of attack introduces a malicious code that intrudes the authentication process, making it ineffective for testing the data's credibility. Therefore, this is one of Merkle tree-based networks' most critical threats as it takes away their ability to validate the datasets.

Attacks can be defeated by different parts of the system based on the type of attack and its effect. Table 1 summarizes the attacks considered in the implementation and design.

2.2. Blockchain and Data Integrity. Blockchain is one of the most advanced technologies for data security as it allows the data to be stored in blocks linked through the chain. This chain is complex enough to avoid the intrusion of cybercriminals. Blockchain is becoming popular in various parts of the world as it is a highly secure method of keeping data safe. The blocks are assigned hash values along with timestamps that depend on the data stored in each block and their link with the neighbor blocks [24]. Therefore, it is almost impossible for the hackers to intrude into the system and steal the data as it would require changing the hash value of a block, which would take high cost and time due to the dependency on other blocks. In addition, it is important to note that such an activity cannot go unnoticed as the network managers and cybersecurity experts have a close monitoring and control system over the blockchain [34]. Therefore, breaching the security of the blockchain is the most challenging task for any cybercriminals. Also, blockchain technology in data safety is based on a consensus algorithm, which prevents malicious activities from becoming successful. Hence, most organizations are shifting their database to blockchain to ensure high security and integrity of data.

Moreover, blockchain is based on cryptography that uses encryption through advanced algorithms to hide the real data [10]. The use of encryption in the blockchain is very important for data integrity as it keeps the data safe during communication, processing, and storage. Thus, the use of blockchain in IoT systems adds a major benefit to the latter

Table 1: Summary of attacks on HANs.

Attack name	Effect	Defended by
Malicious block (cite)	Attackers produce a malicious block	Authentication algorithm and smart contract
Denial-of-service (cite)	Data unavailability	Access authority and management
Access control impersonation (cite)	Fraudulent behavior	Consensus algorithm
Double-spending (cite)	Same transaction multiple times	Consensus algorithm
Side-channel (cite)	Authentication process	Authentication algorithm

as it substantially improves the data security standards. However, it is one of the riskiest areas in IoT due to a large volume of data [11]. IoT-based smart home networks are highly vulnerable to data security threats as they involve collecting, transferring, and storing household users' personal information. Blockchain can prevent both data tampering and spoofing [35], recording all node transactions in the blockchain, which is a complete managing and securing of the industrial IoT and operational technology (OT) devices, in which the transacted data of the sensor, device, or controller after it is deployed and starts working cannot be changed [36]. Another important benefit of using blockchain in IoT-based systems is that it will take intruders a large amount of time to break into the system and data.

The researchers conducted a study over blockchain for the security of data in smart home networks. It was noted that the increasing use of smart home networks is raising different challenges of privacy and authentication. Therefore, the authors have proposed an IoT-based blockchain for smart homes to ensure the safety and integrity of data. The authors proposed network architecture based on key blockchain elements such as smart contracts and tokens to perform strong verification checks. These verification checks are the primary function of the blockchain that enables them to perform authentication checks. Using these security protocols, smart home networks can be strongly secured by blockchain technology. The authors of [37] have used an IoT-based network to conduct tests on the use of blockchain to apply highly secure standards for the transaction of information and data. Thus, the result found the model to be highly effective in preventing any attack from external forces.

2.3. Blockchain Standardizations. Security standards are a set of policies and methods to keep the smart home system protected. Security standards allow the systems to become safer as the standards have been tried and tested before. Thus, the use of the developed standards is very effective in making secure networks and data systems. These standards are developed by internationally acclaimed organizations such as ISO, NIST, IEEE, ITU, and W3C, which work to introduce standardization at all levels and areas of the firms. Therefore, organizations and industries can keep themselves protected from the existing threats and challenges [38].

The use of network security standards is aimed at preventing, detecting, and rectifying network challenges and threats. The use of these network standards is critical in ensuring the security and integrity of data. Hence, in our research, we aim to comply with the standards as they are the factors that ensure our research's security to be applicable [39]. There are several standards available in the market. Most of them are authorized by the governments and used in different private and public sectors.

Some of these standards focusing on blockchain and distributed ledger technologies are as follows:

(i) Permissioned distributed ledger (PDL) provides the foundations for the operation of permissioned distributed ledgers, which is not limited to standardization activities. Furthermore, it includes research activities and initiatives concerning blockchain and the distributed ledger [40]

(ii) Focus group on application of distributed ledger technologies (FG DLT) provides the process and technologies to synchronize the distributed ledger across the network's nodes to undertake the updates and validate the network's nodes securely

Other standards play an important role in facilitating business interaction, communication, measurement, and manufacturing [41]. For example, ISO 27001 is an information security management standard based on the assessment of organization management of its data security systems. The implementation and control of data security are the major requirements of this standard [10].

Thus, its use is significant for home networks as it can help the service providers to maintain standardized security systems that can be applied universally to all households. Similarly, in ISO 7498-2, there are seven layers in the security architecture:

(1) The authentication layer

(2) The access control layer

(3) The nonrepudiation layer

(4) The data integrity layer

(5) The confidentiality layer

(6) The assurance/availability layer

(7) Notarization/signature layer

ISO 7498-2 standard is based on using a reference model to secure different basic layers of the information system model. This standard has different security protocols for each of the layers. ISO 7498-2 focuses on the communication used by the information systems and ensures that each layer communicates the data in a very secure manner.

Therefore, this standard provides highly efficient security of networks involving various clients of smart home networks. Hence, this standard is examined in detail in this research to ensure data security for IoT smart home networks based on blockchain technology.

2.4. Smart Contract. Smart contracts are digital contracts that are self-executable without the need of a third party. The use of smart contracts is very crucial in decentralized networks such as blockchain that do not have a central authority, allowing all the parties to interact with each other without any third party. Therefore, smart contracts ensure that all the transactions between the nodes are credible and reliable [42]. A smart contract is very efficient in avoiding conflicts and keeping the cost of transactions minimal. The safety of blockchain is highly dependent on smart contracts because they allow the users to share information, money, shares, etc., without any middleman [43]. The effectiveness of a blockchain is incomplete without smart contracts, as it is the most important tool to ensure that all the transactions are carried out fairly.

Smart contracts are a major part of the second-generation blockchains. Previously, the experts found that the blockchains are less effective due to the tools' inability to handle the conflicts [44]. Recently, the industry's application interest that focuses on digital assets has increasingly moved to the second generation of blockchain applications, including digitizing asset ownership, smart contract, and intellectual property. It must be noted that smart contracts act like real business contracts, which ensure that all the parties comply with the contract terms. Therefore, the increasing trend of blockchain applications worldwide is enabling organizations to use smart contracts. This practice has the benefit that smart contracts are strong enough to control the behavior of parties that are part of the blockchain. They are more efficient at keeping the transactions fair as compared to physical contracts. A smart contract is also advantageous because it can be encoded as computer code rules, which can then be replicated and executed across all the blockchain nodes. Therefore, this saves the time and cost of making an individual contract for all the blockchain nodes.

IoT devices generate a huge amount of sensitive data with limited resources. Using blockchain technology with a decentralized smart contract, the network will be more secured and efficient by improving different factors like error handling, monitoring, analysis, and data and identity issues. However, this is outside the scope of this paper, where our work focuses on the design and configuration of the smart contract as a part of the blockchain.

Moreover, another benefit of smart contracts is that they are self-enforcing. They do not require any external authority to manage the contractual terms and monitor external inputs from trusted sources, such as a financial exchange or meteorological service [14]. The complexity of the network due to a large number of users and the structure of the conventional contracts makes them very ineffective. Hence, smart contracts are very critical as it allows the blockchain users to ensure their safety. Smart contracts are incumbent upon all the nodes to perform according to their responsibilities. Through this practice, blockchains have become very reliable for making transactions. Bitcoin and Ethereum are common blockchain examples that use a smart contract for keeping all the money transactions safe and fair [45]. Hence, through this practice, blockchains are becoming a regular part of various organizations.

Smart contracts can also be effectively used for smart home networks that use blockchain. In smart home networks, data fairness and safety management are one of the major challenges that can be overcome by using smart contracts in blockchains [46]. Also, in smart home networks, the users are preferably not to be managed manually. Therefore, smart contracts allow the entire network to be managed digitally without the need for any external or third-party authority. The smart contracts work based on verification as they ensure that the terms are fully satisfied by the users. This practice can accomplish high data safety and integrity for the data producers and consumers [47].

There are various types of smart contracts that are used in blockchain systems. The following are the fundamental types of smart contracts [48]:

(1) Decentralized autonomous organizations (DAOs): this type is based on the general rules that are made by the developers who designed the blockchain. These rules apply to all the participants of the chain. Therefore, all the users have to comply with the rules to ensure that they can work effectively in the blockchain community. One of the key points about DAOs is that they are usually handled manually as the programmers and developers have a strong influence over the control of the smart contracts

(2) Application logic contracts (ALCs): this type is usually engraved in the program's code. These contracts work with the program code and other smart contracts to enforce the rules for blockchain users. Therefore, they are completely independent of the external forces as they can make decisions based on the programmed code. ALCs are highly effective in handling the entire network transactions independently as they can be replicated at all the nodes automatically without human assistance. Hence, such smart contracts are gaining high popularity among network designers and developers

2.5. Consensus Algorithms. Consensus algorithms are the core part of the blockchain network as they allow the network to function without any central authority. Consensus algorithms are based on the mechanism of consensus, in which all peers need to arrive at a common agreement about the states of a ledger [49].

Therefore, this mechanism of the blockchain networks ends the need for any centralized power. Through this practice, the peers in the network can build trust and carry out secured transactions among themselves. Various types of consensus algorithms exist in the blockchain network, which is applied according to the network's objectives and the

users' needs. Different consensus algorithms are used in blockchain; they are generally categorized as proof-based and voting-based consensus algorithms. The classification is shown in Figure 1. The proof-based consensus algorithms require the nodes joining the verifying network to show that they are more qualified before having the right to append a new block to the chain. Meanwhile, voting-based consensus algorithms consider the number of votes cast by nodes on the network to achieve consensus on transactions and key network decisions.

The comparison of these consensus algorithms is shown in Table 2 [50]. The overview for each consensus algorithms is further explained as follows.

(i) *Proof of work (PoW)*: the concept of this algorithm is to produce a new block to the blockchain and confirm the transaction. This process responsibility bears special nodes called miners, and a process is called mining. In PoW, miners compete against each other to complete transactions on the network and get rewarded [44]. This algorithm offered multisignature transactions and multichannel payments over an address to enhance blockchain security. It also has strong support to increasing numbers of nodes in the network. This type of consensus algorithm is highly effective in an environment where there is a lack of trust among the nodes (e.g., public networks of home networks) that involves multiple data collectors and communicators, where each user sends each other digital tokens [51]

(ii) *Proof-of-stake (PoS)*: this algorithm works on an incentive mechanism, where every block is validated through a betting system [53]. If a consensus is made between the majority of the blocks about adding another block, the stakes of all the blocks are raised, which is a strong support to increasing numbers of nodes in the network. Therefore, this algorithm has a drawback that it can cause major stakeholders' monopoly, influencing the transactions' efficiency. While PoS solved various issues earlier associated with PoW, two popular PoS variations are DPoS and LPoS

(iii) *Delegated proof of stake (DPoS)*: this type of algorithm is based on a positive relationship; the more the coins and vote to invest, the more the weightage to receive, providing the semicentral control benefits to the network. For instance, to increase the speed while maintaining the features of the decentralization network and enhance the efficiency [59], this algorithm has strong support to increasing numbers of nodes in the network

(iv) *Leased proof of stake (LPoS)*: this type of algorithm operates on the wave platform, which is considered as an advanced version of the traditional PoS. The users will add the next block to the blockchain by releasing a larger amount of a cryptocurrency to the full node; as a part of the process, the leaser will gain a percent of a transaction fee [54]. Also, this algorithm has strong support to increasing numbers of nodes in the network

(v) *Proof of activity (PoA)*: this algorithm is a hybrid approach of PoW and PoS blockchain consensus models designed through the convergence of both of them. The validator races to solve cryptographic mathematical challenges at the earliest using special hardware and electric energy, similar to PoW. The mechanism could end up to Prove of PoS when the blocks added the network and hold only the information about block winner and reward the transaction identity. [30]. This algorithm has strong support to increasing numbers of nodes in the network

(vi) *Proof of identity (PoL)*: the concept in this consensus algorithm is just the same as that of the authorized identity. To ensure the authenticity and integrity of the created data by network users using cryptographic confirmation of the private key attached in any transaction, any identified user in the network works under a consensus that the PoI algorithm can create and then manage a block of data that can be presented to others in the network [54]. This algorithm has strong support to increasing numbers of nodes in the network

(vii) *Proof of importance (PoI)*: this algorithm developed NEM; PoI is a variation of the PoS consensus algorithm that considers the mechanisms of validators for its operation. However, this algorithm is used to determine which network nodes are eligible to manage, when adding a new block to the blockchain and which are not affected by the size and balance only but also other factors similar to the number of transactions between network nodes; reputation and overall balance also play a role in it [30]. In this algorithm, scalability is considered and support increasing numbers of nodes in the network

(viii) *Proof-of-capacity (PoC)*: this algorithm is a less popular consensus algorithm [13], where the validators invest their hard disk space into the network, rather than adding any money or hardware. The more space a validator has in the network, the bigger his chance to mine the next block and get rewarded. Therefore, this system also promotes monopoly, making the smart home network less effective and unsafe. It relies on the computing power of the miners to their disk capacity [43], which is significantly energy efficient. The miners store huge data to mine the next block. The drawback of this technique is high latency, especially with the smart home network in which the devices have limited storage capacity. Hence, PoW is the most suitable consensus algorithm for the home network [60]. In this research, the PoW has been used in the system model. Scalability?

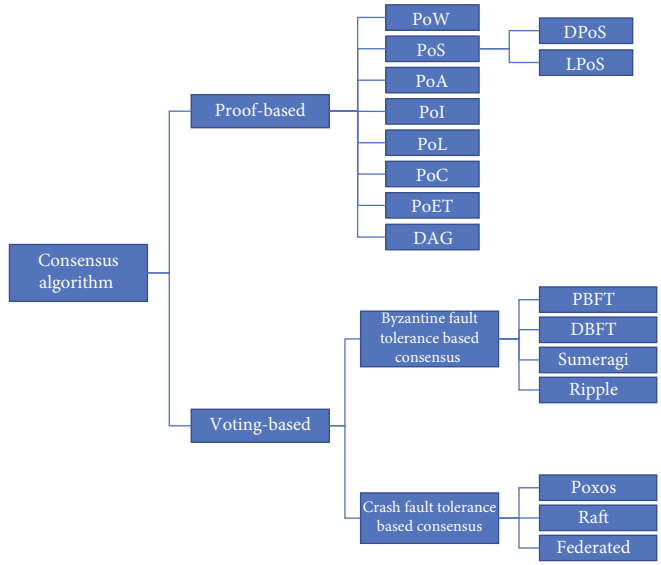

Figure 1: A summary of the consensus algorithms.

Table 2: Consensus algorithm comparison.

Algorithms	Designing goal	Advantages	Disadvantages	Scalability
PoW [44, 51]	Sybil-proof	(i) Security improvements (ii) Minimize the attacks up to 50% or less [52]	(i) More power consumption (ii) Centralized miners	Strong
PoS [53]	Energy efficiency	(i) Energy efficient (ii) More decentralized	(i) Nothing-at-stake problem	Strong
DPoS [53]	Organize PoS effectively	(i) Energy efficient (ii) Scalable (iii) Increased security	(i) Partially centralized (ii) Double spend attack	Strong
LPoS [54]	Distributed PoS	(i) Fair usage (ii) Lease coins	(i) Decentralization issue	Strong
PoA [29]	Benefits of both Pos and PoW	(i) Reduces the probability of the 51% attack (ii) Equal contribution	(i) Greater energy consumption (ii) Double signing	Strong
PoL [54]	Improve PoS	(i) Vesting (ii) Transaction partnership	(i) Decentralization issue	Strong
PoC [13]	Less energy than PoW	(i) Cheap (ii) Efficient (iii) Distributed	(i) Favoring bigger fishes (ii) Decentralization issue	Strong
PoET [55]	Decide the mining rights	(i) Cheap participation	(i) Need for specialized hardware (ii) Not good for public blockchain	Low
DAG [44, 56]	Speed and scalability	(i) Low-cost network (ii) Scalability	(i) Implementation gaps (ii) Not suited for smart contracts	Strong
BFT [30]	Failures of system	(i) Energy efficiency (ii) Transaction finality	(i) Number of replicas in the network (ii) Message complexity	Low
PBFT [30]	Remove software errors	(i) No need for confirmation (ii) Reduction in energy	(i) Communication gap (ii) Sybil attack	Low
DBFT [30]	Faster PBFT	(i) Scalable (ii) Fast	(i) Conflictions in the chain	Medium
Sumeragi [57]	Reputation system.	(i) Distributed across many clusters	(i) The more nodes that exist on the network, the more time it takes to reach consensus	Medium
Ripple [58]	Same FBFT	(i) Reduce the latency	(i) Few nodes required to vote, not really distributed network	Strong

(ix) *Proof of elapsed time (PoET)*: this algorithm developed by Intel enhances the PoW mechanism in view that the CPU architecture and the validator hardware identify when and at what frequency a validator deserves the reward block. It is based on fair network distribution and expanding the odds for a bigger fraction of participant nodes in the blockchain. The task for every participating node is to wait for a particular time to participate in the following mining process. In this case, the miner is randomly chosen to solve the hash problem based on a random wait time [55]. Network validator nodes with the shortest hold-up time have the authority to offer a block. Simultaneously, every network node similarly comes up with its own waiting time. After the sleep mode, the node gets active and a block is available. This network node is considered as a validator. In the end, the validator can spread the information throughout the blockchain network, even though maintaining the property of decentralization in the network and then receiving the shared reward. This technique is aimed at reducing energy consumption

(x) *Direct acyclic graph (DAG)*: this algorithm is familiar to every blockchain mobile app development service company. In this model, all nodes in the network can be a "miner" and validate transactions by themselves and reduce fees to zero, making the process easy, faster, and secure. This algorithm is used in Tangle [56] that reduces the computational time and does not use blocks to store transactions. With DAG, each transaction must approve two older transactions to provide a fast, scalable supports and no transaction fee framework for data integrity [44]. The drawback of this technique is the storage caused by imposing the rule to choose two-order transactions for approval. It can be solved by running a node named "coordinator" to perform this rule. However, this can be in conflict with the decentralization's basics of the blockchain architecture

(xi) *Byzantine fault tolerance (BFT)*: (or called Byzantine Generals Problem) this algorithm is used to deal with the Byzantine fault in situations where the system's actors have to agree on an effective strategy to circumvent catastrophic failure of the system, but some of them are dubious. The two variations of BFT models that are prime in the blockchain arena are PBFT and DBFT [30]. This algorithm has low support to increasing numbers of nodes in the network

(xii) *Practical Byzantine fault tolerance (PBFT)*: this algorithm is a lightweight algorithm that solves the Byzantine Generals Problems by enabling the user to confirm the messages delivered to them by performing a computation to evaluate the decision about the message's validity. This algorithm has low support to increasing numbers of nodes in the network [30]

(xiii) *Delegated Byzantine fault tolerance (DBFT)*: (presented by NEO) this algorithm is similar to the DPoS mode, besides being more effective by countering unreliable or untrustworthy participants to the blockchain. This algorithm has medium support to increasing numbers of nodes in the network. Moreover, the NEO token holders can vote for the delegates [30]

According to [41], the consensus algorithm is used in the smart home network to prevent anybody from playing the network and to secure the communicated devise and stored data in this network against malicious acts that desire to wreak havoc with someone's home. This work is focused on the PoW based on optimizing data integrity for the time required to secure the blockchain of the smart home network, taking into consideration the total storage required and executing several hash algorithms.

Moreover, the benefits gained from PoW, not covered in this work, are solving the cooperation tracking, collective behavior, and discrete opinion [61]. These are based on using statistical methods and mathematical calculation. Furthermore, this work is not attempting to change the structure of the PoW and disrupt the core characteristic of the blockchain, which should be always secure, decentralized, and peer to peer. The proposed system [62] works on improving the transaction speed and scalability by modifying the structure of PoW and introducing a parallel PoW in which the miners work together in such task to validate the transactions.

Additionally, it is suitable when working with the mathematical challenges in the digital ledger, such as recording and maintaining the unalterable transactions [63]. Typically, PoW is a computationally expensive consensus approach. However, such techniques reduce the computation requirements to achieve data integrity of the network without harming the data protection criteria [28]. Hence, the smart home networks' high security can be ensured by the PoW consensus algorithm.

2.6. Verification Algorithms. The data structure verification algorithm is also known as the hash tree [64], which includes transaction blocks. Every node connected to IoT devices in a smart home network, including miners in a blockchain network, has a memory pool (Mempool) that contains all current transactions that are waiting to be added to the blockchain to produce a new block. This memory pool contains all the current transactions in wait, which must be added to the blockchain to create a new block.

The verification and summarization of all the transactions after hashing are performed by the different algorithms [34]. In consideration of the particularity and complexity of the chained hash database, the concatenated hash transaction (CHT) strong component is collision resistance but not more than collision resistance because it is feasible for an attacker to find two messages with the same hash

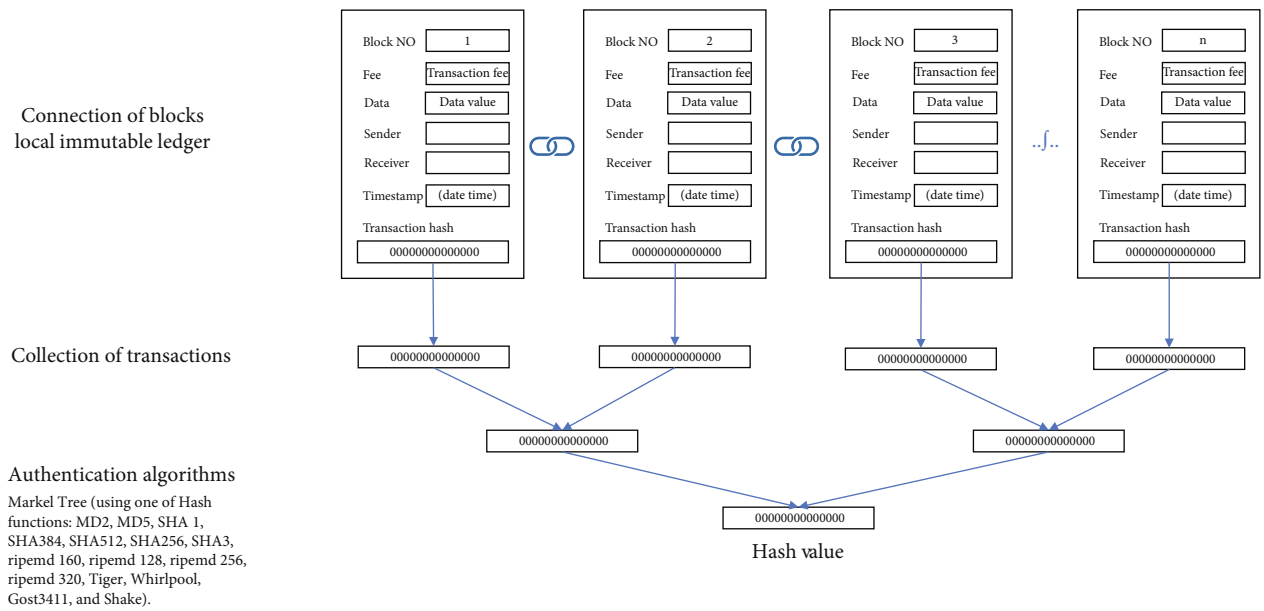

Figure 2: Merkle tree algorithm visualization.

functions. They can find as many additional messages with that same hash function as they desire, with no greater difficulty. Using Merkle hash tree (MHT) in blockchain is more effective than CHT by increasing complexity, and validators can calculate hashes progressively as they receive transactions from their peers.

MHT summarizes all the transactions in a block by using a mathematical data structure composed of hashes of different data blocks. "?>. It allows an efficient and secure verification in the blockchain of a large transaction hash. As a fundamental part of blockchain, MHT benefits by providing a means to maintain the integrity and validity of data and helping in saving the memory or disk space as the proof, computationally easy and fast, and their proofs and management require tiny amounts of information to be transmitted across networks.

Figure 2 shows the MHT block connection and the algorithm structure. If the transactions are valid, they are included in the new block that miners on the home network can mine. The miners produce a hash of a block by the process of changing the nonce and time stamp. In the blockchain, the system then compares the generated hash with the target. As soon as validating of the new block is finished, this block becomes part of the chain. After checking the hash's value below the target value, the PoW is verified as a successful transaction and then added to the block [65]. Subsequently, an update notification concerning this change of the blockchain size is broadcasted to the whole network for the ledger to inform every connected nodes [66].

Similarly, another research is conducted in [67] by the researchers over the existing surveillance systems. It was noted that the current surveillance system has a high risk of data security, which demands more safety protocols to protect the privacy of the users. The current use of cloud and big data-based surveillance systems has not been very efficient at handling the data and information as they are vulnerable to possible attacks by intruders. Therefore, [67] has proposed blockchain technology in the network of surveillance systems to ensure high security of the data. The results noted that using a MHT algorithm made the transactions safer as it imposed effective monitoring and control of the security. This proposed method is also applicable to the home networks that have a high number of users. The MHT algorithm enabled the blockchain to conduct verification checks at all the nodes to prevent any data attacks [67]. Therefore, the proposed model is very efficient at handling data security due to its high convenience of design and implementation.

2.7. Hash Functions. One-way hash functions are used to map any data of the arbitrary size to fixed-size values, also referred to as message integrity check (MIC), message digest, contraction functions, compression functions, cryptographic checksum, fingerprint, and manipulation detection code (MDC). In the Merkle tree, the hash value is used as a Merkle root as the tree is created bottom-up using the individual transaction hashes. 15 popular hash functions were used in the implementation, which includes the following: MD2, MD5, SHA 1, SHA384, SHA512, SHA256, SHA3, RIPEMD-160, RIPEMD-128, RIPEMD-256, RIPEMD-320, Tiger, Whirlpool, GOST3411, and SHAKE (SHA with KEccak) [68].

3. Home Area Network Blockchain-Based Security System Model

3.1. Architecture Components. Node.js version (v12.16.2) and NPM version (6.14.7) have been used at the server side, where the local blockchain operates. At this place, the smart contract is also developed and deployed with the help of NPM. In addition to this, various tools and plugins such as the Truffle framework version (5.1.37) for deployment, migration, and management of smart contracts are used.

MetaMask add-on in the Chrome browser has been used for visiting the distributed web in the browser. Besides, the Solidity programming language has been used to develop smart contracts. Figure 3 and Table 3 show the components used in the local blockchain along with the network and server elements and versions.

After setting up all the server nodes, all the network components are connected in a local blockchain provided by Ganache. The Ganache version (2.5.4) has been used in this study. The accounts provided by this blockchain network are added to MetaMask to start transactions and make the blockchain network fully operational.

The simulation is implemented on a laptop with specifications given in Table 4. An i7 8th generation processor has been used along with 16 GB RAM to enhance the network's performance. Also, SSD storage is used to ensure that the data can be processed and stored at high speed.

3.2. Smart Home Ecosystem. The smart home IoT ecosystem data is majorly generated from the sensors connected to different devices and electronics. The computing nodes with the central processing unit have to process these data collected from the sensors. These nodes then send the processed data to the transceivers, allowing the information's transfer to other nodes or other associated devices. In addition to this, the actuators work according to the decisions made by the competing nodes. These actuators may be electromechanical in nature that allow them to receive data from the nodes and use it to operate different devices on the network. Therefore, through this practice, the data collected from the sensors can be used to trigger different devices' function based on the decision made by the computational nodes [15].

On the other hand, the IoT smart home control systems can be messaging based, such as Telegram, Blynk, and web or they can work as a voice command, for instance, Google Assistant, Apple kit, and Amazon Echo [69]. A decentralized structure is implemented in the current study along with smart contracts using Solidity in a Truffle framework. This structure is then deployed into Ganache, a blockchain network. The nodes in this network are designed to perform a transaction through web-based services on NodeJS server, the frontend of web3js with a combination of HTML, CSS, and JavaScript. The network's backend is designed using a combination of Truffle environment for blockchain development and management with Ethereum Virtual Machine (EVM), smart contract, deployment, and binary management, as well as NodeJS web servers with a node package manager (NPM). The NPM manages the users' requests and calls for transactions outside the context of a frontend in the migration of ABI bytecode for the compiler to deploy the smart contract. Figure 4 shows a smart home ecosystem, where all the devices are connected to a single network that controls all the devices' operations.

The nature of blockchain technology is usually decentralized. However, in smart homes, the blockchain has a central authority to ensure efficient control and monitoring of all the devices on the network [7]. The decentralized nature of blockchain is based on the distributed transactions between

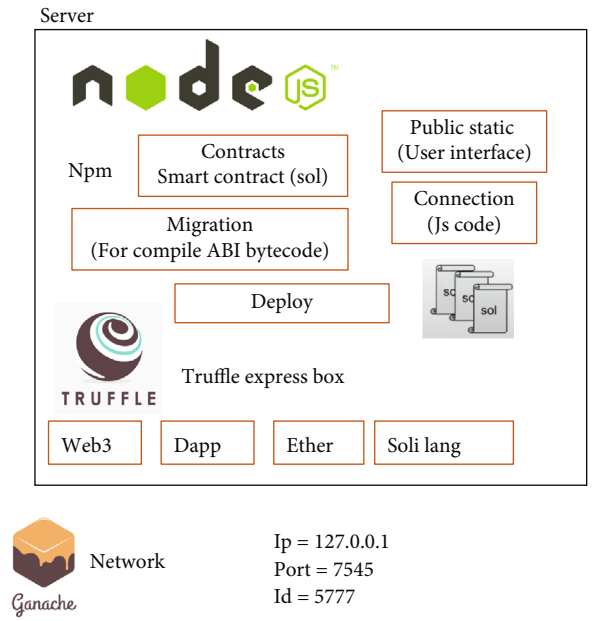

Figure 3: Local server blockchain components.

Table 3: Summary of server component versions.

Component	Version	Role
Node.js	12.16.2	Blockchain backend
NPM	6.14.7	Package management
Truffle	5.1.37	Smart contract development tools
Ganache	2.5.4	Distributed local network

Table 4: Summary of local server specifications.

Component	Description
CPU	Intel(R) Core (TM) i7-8550U CPU @ 1.80 GHz 1.99 GHz
RAM	16.0 GB Speed 2133 MHz
OS	Windows 10 Pro, version 20H2, 64-bit operating system, x64-based processor
Disk type	SSD SAMSUNG MZVLB512HAJQ-000L7

the blockchain network-participating nodes. These transactions are not stored in a single node or a storage device. Also, there is no central authority to approve the transactions as they are assessed according to the blockchain algorithm's specific rules. Therefore, this removes the need for a central authority in a smart home network to carry out transactions or reach a consensus in the network. In addition to this, blockchains allow only new verified blocks to be added to the old chain. The existing blocks in the blockchain are already public and distributed, which makes them openly verifiable. Hence, they cannot be changed or revised. Thus, the blockchain security is another major benefit for the home networks as they allow the data to be kept safe [70].

3.3. Smart Contract Design. A smart contract is an essential part of the blockchain network used to ensure fair and

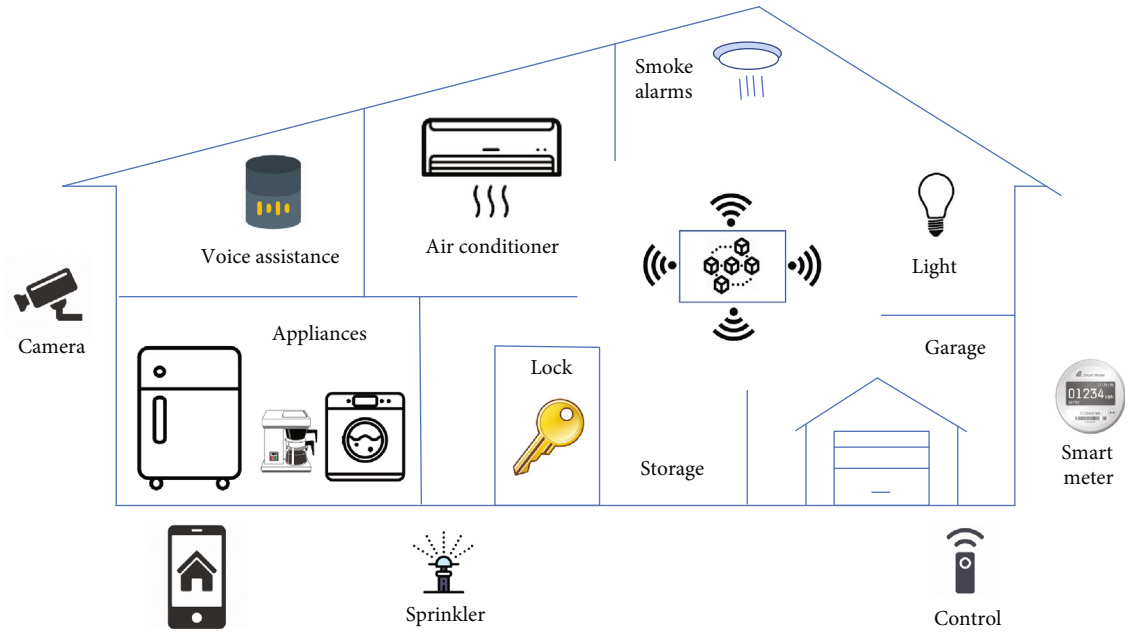

Figure 4: Example of a smart home network.

secure transactions among users. Therefore, a smart contract is also used in this research to make the network highly efficient. Smart networks are usually written in a domain-specific language such as Solidity so that they can be used to reach a consensus among all the peers in the network [42]. The smart contract serves a variety of functions in the blockchain, including registering all network and node components, receiving queries and transactions from various peers and applications, and allowing them to access the blockchain network's private ledger and update the ledger database. The different functions in the smart contract system is shown in Figure 5.

The hash value of the transactions is the key to the smart contract's verification process as the latter uses it to ensure all the terms have been complied. The smart contract includes the functions with a specific requirement of the input parameters [43]. The peers must fulfil these requirements on the network to make changes to the private ledger. In the nonfulfillment of the input parameters, the smart contract automatically rejects the peer's query. The smart contract takes decisions based on the code stored in it, which are also referred to as the rule of the network [38]. Therefore, a similar smart contract has also been used in our proposed research work to gain highly efficient results. Figure 5 shows the smart contract design used in this study used to communicate with the IoT blockchain network private ledger and client application.

The smart contract mechanism is illustrated in Figure 6 as an executable code stored in the blockchain that defines and manages the operation between smart devices and the identity of all kinds of users, from homeowners to local miners and normal users.

3.4. Process Flowchart. Figure 7 shows the process workflow of the PoW consensus algorithm. The workflow consists of five layers: the authentication layer, access control layer, data integrity layer, availability layer, and signature layer. According to the authentication layer, the registration of all the devices on the IoT network is necessary to ensure that all the devices are part of the system. This practice can ensure that all the devices are ready to track and perform the transactions.

The access control layer contains the core system components explained previously in the HAN system model section, which determines the process workflow into different scenarios and is linked to other model layers. The registration process takes place at the sponsor, which is usually a server or many servers in the case of a distributed system. In addition to this, a smart contract is the core of the blockchain system to achieve data integrity for all transferred data between the nodes in the network. The transacted data is managed by a data integrity algorithm, which will be explained in Section 5.

Subsequently, validity check is made to approve the transaction handled in the data integrity layer. Next, in the availability layer, the private ledger is managed and updated based on the consensus algorithm's results. Finally, the users and validators can use and verify the performance based on the framework and the used technology in the signature layer.

4. Dataset

The dataset is a collection of various transactions between ten nodes in the network, sorted in tables. These datasets include transactions along with data about transaction hash, block no., timestamp, date and time, data value, and transaction fee, as shown in Figure 8. The total number of hashes for implementation is (30,703), compiled in ten groups.

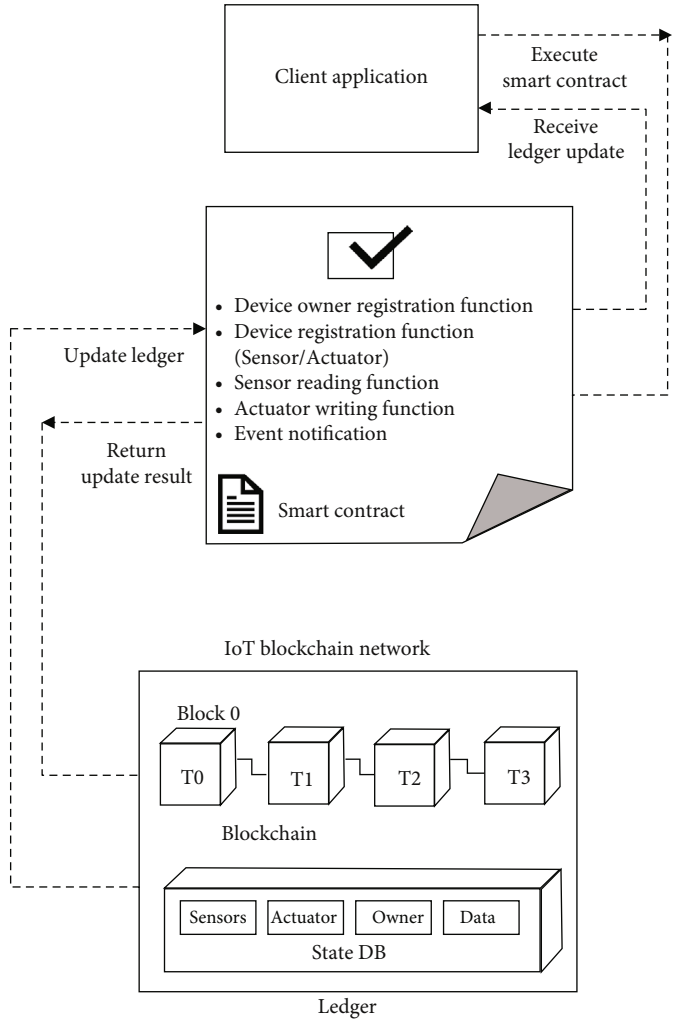

Figure 5: General smart contract mechanism.

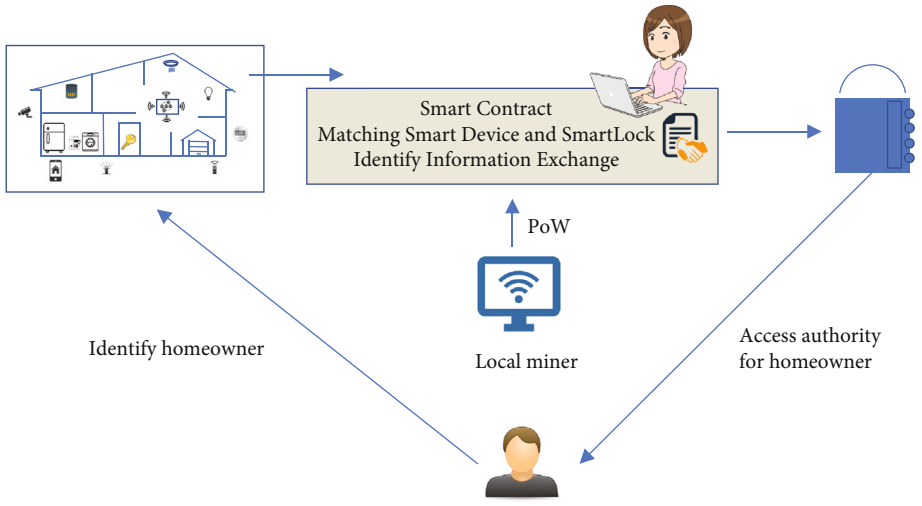

Figure 6: Smart contract mechanism for HAN.

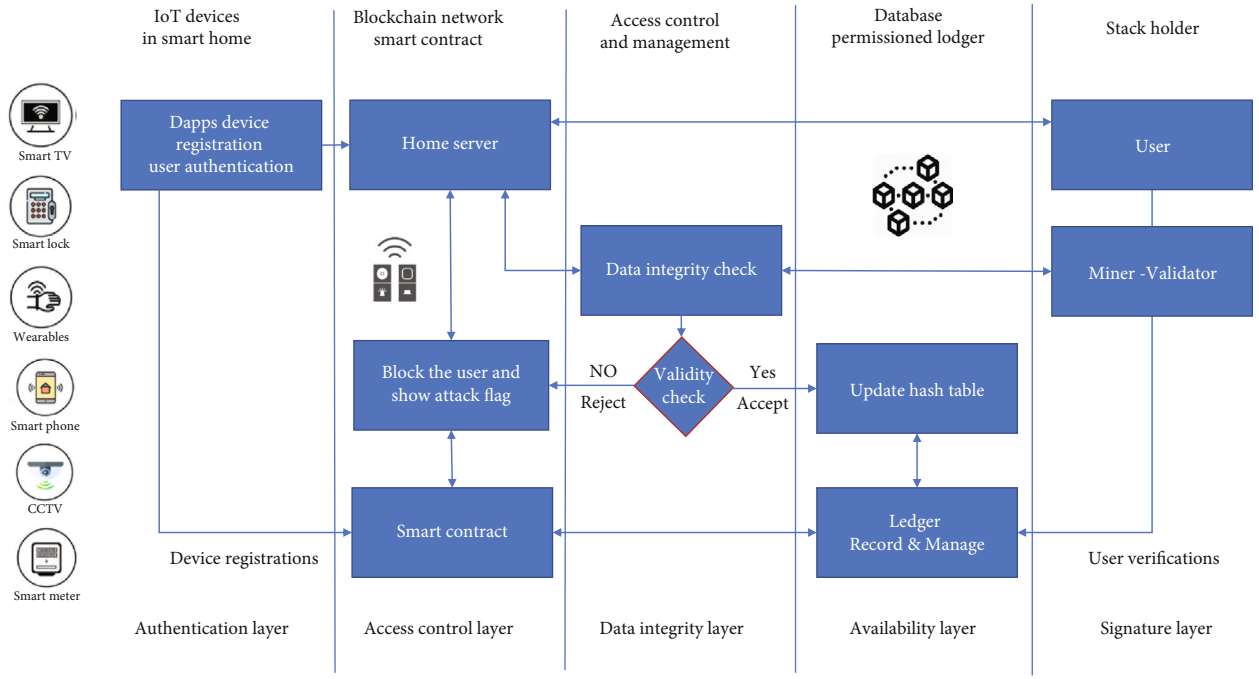

Figure 7: Process workflow of the PoW consensus algorithm.

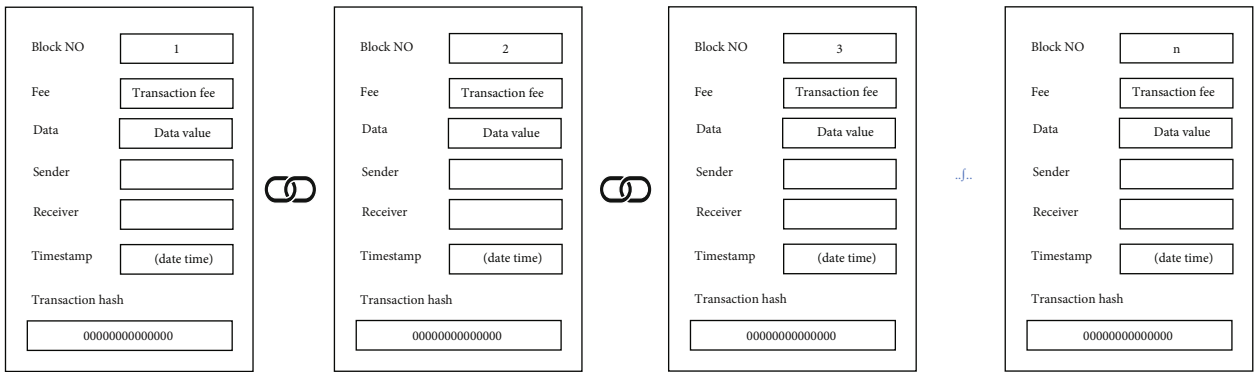

Figure 8: Dataset block structure.

The blockchain dataset structure and size of transactions are shown in Figures 8 and 9 [71].

Data cleaning and normalization have been performed in this work for removing unwanted records such as double spending and errors. Through this practice, an effective evaluation of the proposed system model has been conducted to produce accurate and realistic results.

In the initial stage after obtaining the selected dataset, the data is preprocessed to remove invalid data. Next, they are stored data of blockchain network transactions across a distributed architecture. One of the major objectives includes testing the system's data integrity procedures to examine the effect of each consensus algorithms based on applying different hash functions.

Next, the dataset was split into three different sizes to check the network's performance in data integrity and transaction scalability. Hence, the datasets were based on 30, 3000, and 30000 transactions.

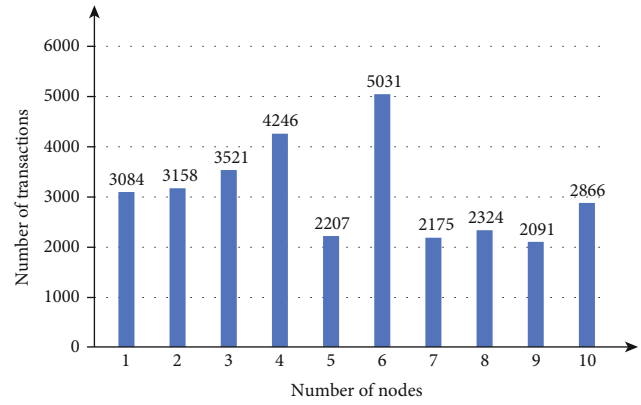

Figure 9: Number of transactions in the dataset.

The researchers conducted a similar evaluation in [17], which performed a similar test on the blockchain network using real home situations. Two smart home networks were

set up in different rooms, and data was collected from three different sensors in each of the rooms. The data was collected and processed through an IoT-based blockchain system that used web3.js to interact, while the smart contract was designed on Solidity. Their work is different from our current framework because it used a small volume of data and reliance on a single hash function and PoW consensus algorithm. Therefore, their results are significantly different from our current findings. The work in [17] tested the network's ability to prevent any attack on the network. It has experimented on a potential attack on the network by inserting correct and incorrect data parameters for the smart contract. The results found that the blockchain network peers were able to make effective transactions only when the correct parameters were provided. However, in the case of wrong parameters, the smart contract blocked the query of the peers. Therefore, it can be noted from [17] that smart contracts can be highly efficient in handling the security of smart home networks.

Moreover, another previous work conducted by the researchers in [18] used the Merkle tree to analyze security data of a blockchain network. The analysis was based on a network of surveillance cameras connected through a blockchain network. The work mainly relied on using the Merkle tree algorithm for authorization of the transactions between nodes rather than smart contracts. They tested the efficiency of the Merkle tree by comparing it with a similar SM tree. The results of both the trees were used to identify the network's ability to identify possible attacks on the blockchain. Some of the tested critical aspects included testing the network against data falsification attacks, message tapping attacks, and privacy masking. The results found that the use of blockchain technology in the surveillance network can make it very secure. It was noted that using the Merkle tree algorithm, the blockchain could perform rigorous authorization checks, which help in the protection against different data security threats. However, they did not use different consensus algorithms for testing the systems and relied on a single hash function and consensus algorithm. Therefore, the results of [18] are less reliable than our current work that uses multiple consensus algorithms and hash functions for testing the effectiveness of blockchain against potential data security threats.

5. Performance Evaluation of Proposed Consensus Algorithm

Different consensus algorithms were used in our proposed system model to compare the results and test the effectiveness and efficiency. Hence, this helped identify the most efficient consensus algorithm for the blockchain network and the ability to enhance data security and integrity. Some modifications were also made to the algorithms to increase their performance and overcome their complexity.

In the first scenario, the transactions' values were hashed while the chain of transactions was concatenated to form the concatenated hash transactions (CHT). Through this method, the final transaction value was obtained, as shown in Figure 10.

In the second scenario, the Merkle hash tree (MHT) [72] algorithm shown in Figure 11 was used for hashing the transaction values. MHT is also referred as a hash tree because it is used in data verification and synchronization. Therefore, it is one of the most used methods for hashing the data. MHT has a tree-like structure in which all the nodes are of the same depth and as far left as possible. The input of the tree is mapped onto the fixed output, which is known as a hash. Thus, through this mechanism, MHT can hash large and complex data with high efficiency. MHT has also been used by previous researchers in [67] to enhance the authentication procedure of the blockchain network. MHT can be represented mathematically for n blocks as follows:

$$H_{0-n} = \text{Hash}(H_{0-1} \| H_{2-3} \| H_{4-5} \| \cdots \| H_n). \tag{1}$$

Next, we attempt to modify the sequence of the values of the MHT algorithm to add complexity to the traditional MHT algorithm by considering the odd and even value sequences together, as shown in Figure 12.

Odd and even modified Merkle hash tree (O&E MHT) algorithm can be represented mathematically for n blocks as follows:

$$H_{0-n} = \text{Hash}(H_{0-2} \| H_{1-3} \| H_{4-6} \| \cdots \| H_n). \tag{2}$$

Finally, a modified MHT algorithm (MMHT) shown in Figure 13 is applied to the transaction chain by combining it with two algorithms (CHT & MHT). The blockchain was divided into two groups during the analysis. The first group of the blockchain was given a specific size as it included the initial blocks until block $(X-1)$. The second group began from block number (X) and ended at the last transaction block (n). The value of X that achieves the best performance will be selected. The first group of the chain used the CHT algorithm. When a new block was added to the chain during the experiment, then block number 1 was removed from the second group and added to the first group. Throughout the test, this process was performed during the transactions; the blocks were removed from the second group and added to the first group. MHT algorithm has been used in the second group of the chain. At the end of the test, groups one and two of the chain were combined and a final hash value was used to validate the transaction as explained in Figure 13.

The main purpose of dividing the original blockchain into two groups was to increase the network's efficiency by accelerating the validation execution time required to find the total hash of the blockchain and detect any partial changes. Through this practice, the process of hashing the transactions was performed with high efficiency and speed, allowing the tests to be conducted accurately.

MMHT can be represented mathematically for n blocks as follows:

$$\begin{aligned} H_{0\longrightarrow n} &= \text{CHT}\, H_{(0\longrightarrow(n-(x+1)))} \Big\| \text{MHT}\, H_{((n-x)\longrightarrow n)}, \\ H_{0\longrightarrow n} &= \left(H_{0\longrightarrow 1} \| H_{2\longrightarrow 3} \| \cdots \Big\| H_{(n\longrightarrow x-2)-(n\longrightarrow x-1)}\right), \\ &\Big\| H_{(n\longrightarrow x)-(n\longrightarrow x+1)} \| \cdots \| H_n. \end{aligned} \tag{3}$$

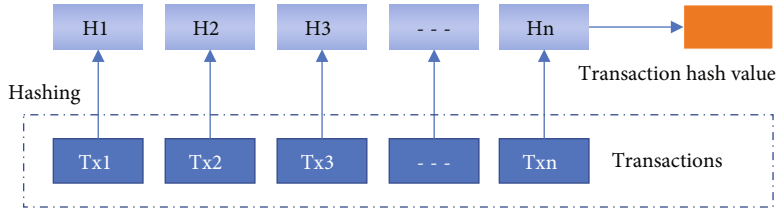

Figure 10: Concatenated hash transaction (CHT) algorithm.

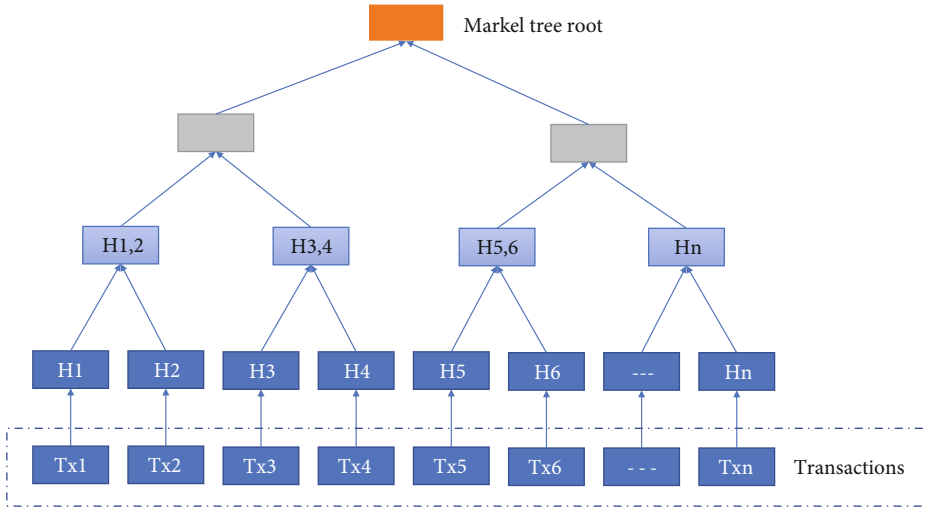

Figure 11: Merkle hash tree (MHT) algorithm.

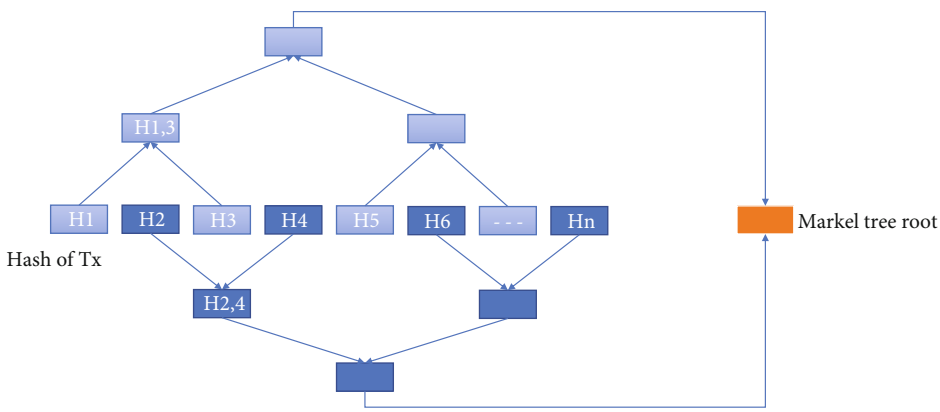

Figure 12: Odd and even modified Merkle hash tree (O&E MHT) algorithm.

Furthermore, another major benefit of using two different blockchain groups was that it allowed the use of a modified algorithm. This helped update the private ledger after knowing the new block's index that was newly added to the blockchain. Therefore, only the new blocks were used to update the hash table. The benefit of this method can be seen in testing the algorithms' effectiveness for the security of data and help in comparing the results of the two algorithms. The trigger number (n) is used to define the first group size by knowing the significant number of changes in the execution time used in the MHT algorithm. In Figure 14, the flow of the MMHT algorithm is presented, and then, Algorithm 1 describes the process steps towards the development of the MMHT algorithm.

In this work, the evaluation of all the above consensus algorithms (CHT, MHT, O&E MHT, and MMHT) with 15 different hash functions was conducted. Furthermore, three different dataset sizes (30, 3 k, and 30 k) to check the data integrity performance of the network at different transaction scalability were investigated. This represents different models of blockchain transactions for a specific system. The results on the validation execution time using different consensus algorithms and different transaction sizes are presented in Tables 5 and 6, considering various hash functions with a length of 128 bits to 512 bits.

The analysis is performed based on an average of 11 simulation runs to obtain an accurate result with a significant

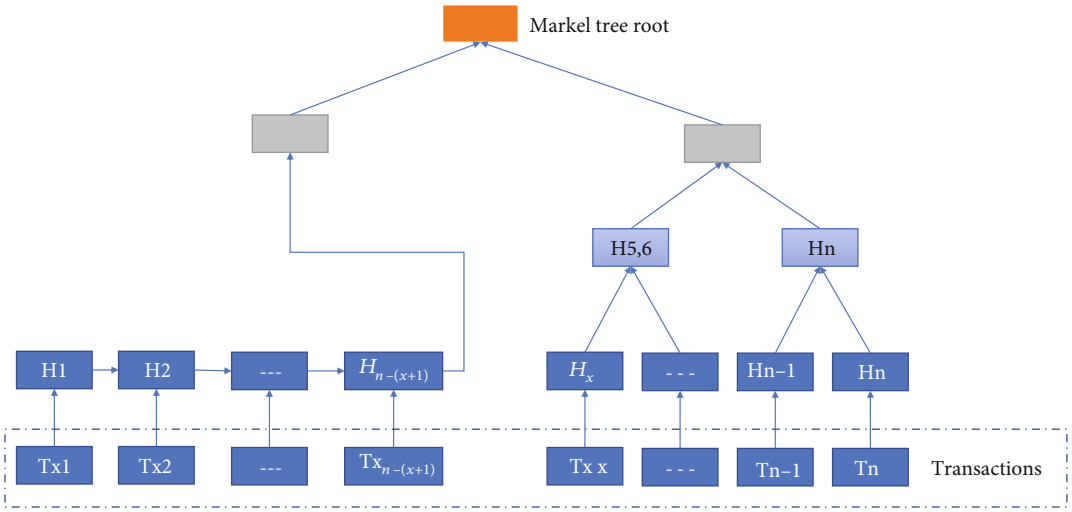

Figure 13: Modified Merkle hash tree (MMHT) algorithm.

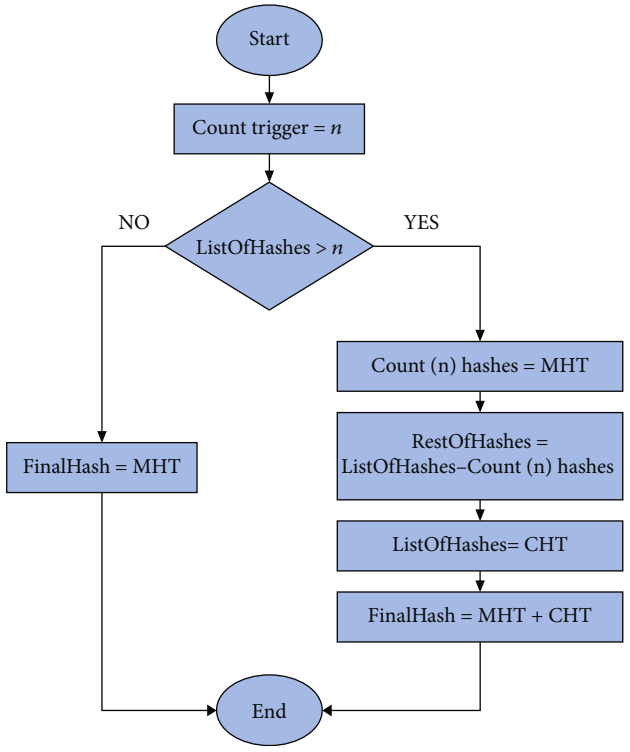

Figure 14: MMHT algorithm flowchart.

confidence interval of 90%. The method of averaging the results of the last ten runs was used to ensure the results' high accuracy and credibility. This practice was used as it was noted that all the execution gave different results and were not fixed. Therefore, to gain more reliable results, the program runs were averaged to produce efficient results.

For the n dataset size of transactions, it is shown that the CHT algorithm has the lowest execution time compared to any tree structure MHT, O&E MHT, and MMHT. However, it is impractical for a blockchain implementation due to the requirement of the entire copy of the blockchain ledger in real time [47]. By using any hash functions, the test runs sequentially from one block to the other.

The results from 30 transactions recorded in Table 5 show the execution time in milliseconds (ms) and the improvement percentage mentioned in the column (%) compared to the proposed MMHT algorithm against the conventional MHT algorithm. Note that the table is categorized into hash length because it has a direct impact on the execution time of the authentication consensus algorithm and for a fair comparison between different hash functions of the same length. The higher the hash length, the higher the theoretical execution time. Then, the same hash length can be compared between hash families. Values in green highlight the best execution time in each family, while the grey ones show the overall best value.

Under the category of a 128-bit hash length, MD5 gives the best performance for all consensus algorithms. Meanwhile, SHA1 for CHT and MMHT has the best execution time compared to other hash functions regardless of hash length. However, the newer SHA2 family (SHA256, SHA384, and SHA512) do not have good execution time performance. This is due to the increase in the computational processing and number of rounds applied in the more complex hash algorithm. RIPEMD-256 gives the best performance for the 256-bit hash length category. However, most improvement (65%) in time optimization between the proposed MMHT and a conventional MHT is observed for GOST3411 hash function. The higher processing time required by the GOST3411 is based on the HMAC (hashed message authentication code) protocol. The results showed a significant benefit of using the proposed MMHT algorithm compared to the MHT algorithm. However, the proposed O&E MHT does not offer much advantage over the conventional MHT.

The analysis also considered the storage size, which is an important factor affecting the blockchain network's performance. It was noted that the storage size is fully dependent on the hash function length used in the chain. Therefore, there is a trade-off between security robustness and the low-capacity smart home IoT devices.

Input:
1. List of Hashes (listOfHashes)
2. Count trigger = n
3. Selected algorithm

Algorithm Steps:
1. Start Process, Initialize Count (n) hashes
2. Condition, listOfHashes > Count (n) hashes, if YES go to step 3, if NO go to step 7
3. Set Count (n) hashes by taking top count(n) transactions from listOfHashes. The remaining transactions are set to restOfHashes.
4. Use Merkel Root to hash Count (n) hashes based on selected algorithm and add it to the restOfHashes.
5. Use CHT to hash the restOfHashes based on selected algorithm, getting the final hash. Go to Step 8.
7. Use MHT to hash listOfHashes based on selected algorithm, getting the final hash. Go to Step 8.
8. End of Process

Algorithm 1: MMHT algorithm process steps

Table 5: Consensus algorithm execution time using a dataset size of 30 transactions.

Hash length	Algorithm	Total storage (bits)	CHT (ms)	O&E MHT (ms)	MHT (ms)	MMHT (ms)	MMHT/MHT (%)
128 bits	MD5	3840	0.05006	0.46151	0.45941	0.20045	56.4
	RIPEMD-128	3840	0.05277	0.50815	0.50605	0.24363	51.9
	SHAKE	3840	0.2185	1.50799	1.50589	0.72076	52.1
160 bits	MD2	3840	0.44345	2.7373	2.7352	1.18904	56.5
	RIPEMD-160	4800	0.07338	0.62818	0.62608	0.36012	42.5
	SHA1	4800	0.04245	0.501	0.4989	0.18341	63.2
192 bits	Tiger	5760	0.05415	0.5399	0.5378	0.29054	46.0
256 bits	RIPEMD-256	7680	0.05246	0.48365	0.48155	0.24179	49.8
	SHA256	7680	0.06199	0.57151	0.56941	0.25043	56.0
	SHA3	7680	0.26925	1.65981	1.65771	0.66007	60.2
320 bits	GOST3411	7680	1.18739	8.26976	8.26766	2.8954	65.0
	RIPEMD-320	9600	0.07626	0.70223	0.70013	0.30498	56.4
384 bits	SHA384	11520	0.04286	0.48769	0.48559	0.30443	37.3
512 bits	SHA512	15360	0.04588	0.46818	0.46608	0.22264	52.2
	Whirlpool	15360	0.37904	2.64535	2.64325	1.1103	58.0

Table 6: Consensus algorithm execution time using a dataset size of 30 k transactions.

Hash length	Algorithm	Total storage (bits)	CHT (ms)	O&E MHT (ms)	MHT (ms)	MMHT (ms)	MMHT/MHT (%)
128 bits	MD5	3840000	12.21466	183.1979	180.5979	103.8136	42.5
	RIPEMD-128	3840000	15.93768	211.553	208.953	101.2142	51.6
	SHAKE	3840000	105.5813	823.7975	821.1975	517.4801	37.0
160 bits	MD2	3840000	256.6967	1635.536	1632.936	1074.465	34.2
	RIPEMD-160	4800000	27.38025	281.9626	279.3626	168.3616	39.7
	SHA1	4800000	18.12102	210.1011	207.5011	123.1716	40.6
192 bits	Tiger	5760000	14.70333	191.2121	188.6121	110.3094	41.5
256 bits	RIPEMD-256	7680000	17.44287	208.334	205.734	121.6099	40.9
	SHA256	7680000	23.85505	274.2761	271.6761	160.9931	40.7
	SHA3	7680000	130.8481	825.7562	823.1562	543.7262	33.9
320 bits	GOST3411	7680000	512.3987	3650.143	3647.543	2337.47	35.9
	RIPEMD-320	9600000	26.5973	288.7437	286.1437	170.9692	40.3
384 bits	SHA384	11520000	17.62926	249.641	247.041	142.4498	42.3
512 bits	SHA512	15360000	18.74704	264.5122	261.9122	151.0031	42.3
	Whirlpool	15360000	178.0506	1251.731	1249.131	803.9161	35.6

Table 7: Consensus algorithm execution time using a dataset size of 3 k transactions.

Hash length	Algorithm	Total storage (bits)	CHT (ms)	O&E MHT (ms)	MHT (ms)	MMHT (ms)	MMHT/MHT (%)
128 bits	MD5	384000	1.94123	34.92615	34.10615	22.11161	35.2
	RIPEMD-128	384000	2.93489	44.16066	43.34066	20.65914	52.3
	SHAKE	384000	18.99856	152.072	151.252	18.78511	87.6
	MD2	384000	47.86639	329.0667	328.2467	20.32121	93.8
160 bits	RIPEMD-160	480000	4.69918	54.79416	53.97416	18.92513	64.9
	SHA1	480000	3.53118	38.42909	37.60909	19.87732	47.1
192 bits	Tiger	576000	2.3327	29.43644	28.61644	19.47202	32.0
256 bits	RIPEMD-256	768000	3.49019	38.84966	38.02966	17.26856	54.6
	SHA256	768000	4.26395	48.45686	47.63686	18.63771	60.9
	SHA3	768000	25.84644	161.3161	160.4961	20.6792	87.1
	GOST3411	768000	84.64851	652.9856	652.1656	19.39326	97.0
320 bits	RIPEMD-320	960000	5.27818	45.66796	44.84796	18.72902	58.2
384 bits	SHA384	1152000	3.24072	44.06398	43.24398	21.01735	51.4
512 bits	SHA512	1536000	2.94544	46.2301	45.4101	19.64456	56.7
	Whirlpool	1536000	31.0449	237.9127	237.0927	18.40894	92.2

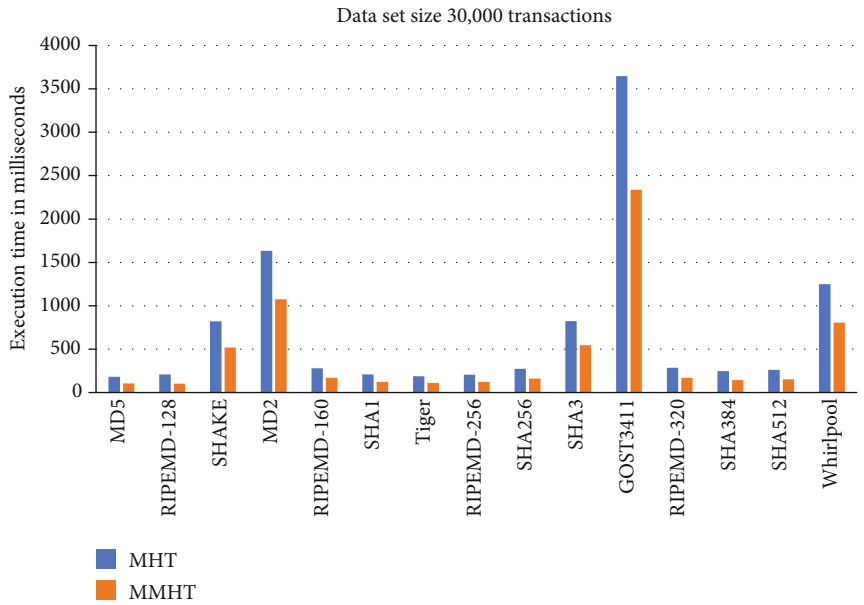

Figure 15: Execution time comparison graph of MHT & MMHT.

The next scenario was then conducted by increasing the size of the dataset to 3000 transactions to represent a medium-sized blockchain smart home network. The MD5 hash function was used along with the CHT consensus algorithm due to the low execution time compared to other algorithms and hash functions. The results of this medium-sized transactions have been recorded in Table 7. Similar to small transaction results in Table 5, the GOST3411 hash function gives the most time optimization gain (97%) against conventional MHT. A new trend to note, the Tiger hash function gives the best conventional MHT performance and RIPEMD-256 giving the best proposed MMHT performance. These results prove that the relationship between the number of transactions, consensus algorithm, and hash function is not straightforward. The blockchain network needs to be designed so that it can be adaptive to different conditions in the network.

The third and final scenario was performed using 30000 transaction dataset. The results of these large-sized transactions are shown in Table 6. In the table, a more expected and stable result where the smallest 128-bit MD5 and RIPEMD-128 hash functions have the best MHT and MMHT execution time, respectively. The proposed MMHT consensus algorithm using RIPEMD-128 hash function also gives the highest time optimization gain of 51.6% compared to conventional MHT. Meanwhile, SHA1 offers the lowest execution time for the 128-bit category and RIPEMD-256 for the 256-bit category and SHA512 category. SHA3, the

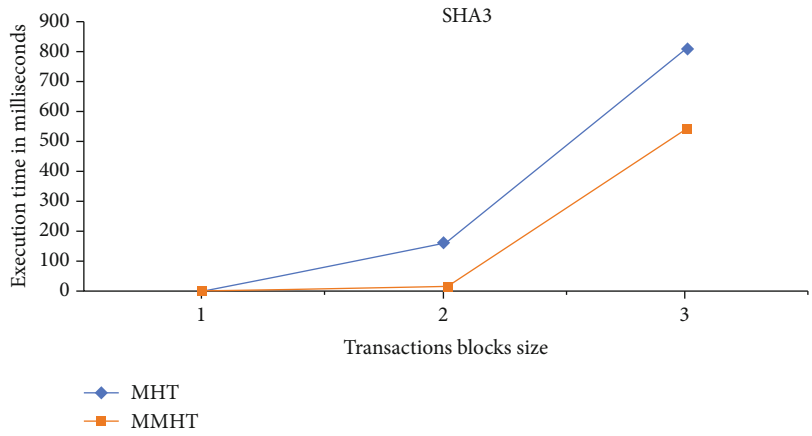

Figure 16: An example of execution time comparison using SHA3 with three levels of datasets.

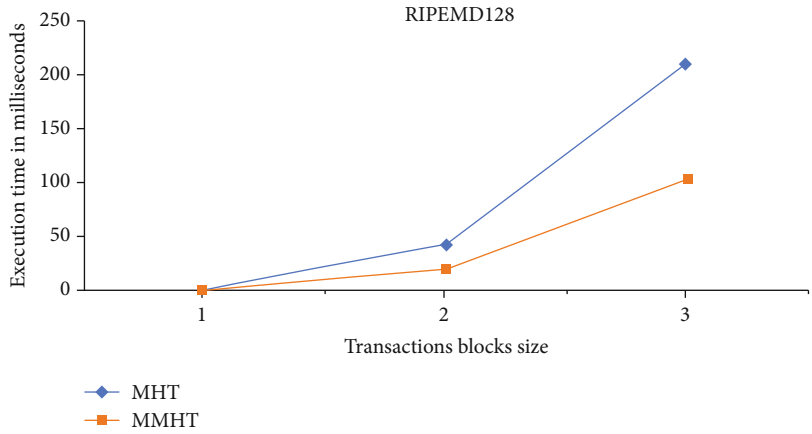

Figure 17: An example of execution time comparison using RIPEMD128 with three levels of datasets.

most recent version of the hash function that is frequently debated and proposed for usage presently, is slower in software implementation of an algorithm but more suitable for hardware implementation [5]; hence, it is not ideal for blockchain architecture.

As a result, we used 15 different hash functions with three different dataset sizes to show that the consensus authentication process can determine which Hash function is the best in terms of performance.

To illustrate the results, one of the comparisons of MHT and MMHT consensus algorithm execution time using 30000 transactions is shown in Figure 15.

The results shows that using the SHA3 hash function with the CHT consensus algorithm has a low execution time compared to other algorithms and hash functions, but not low enough; the use of these hash functions and consensus algorithms has a lower execution time than those of MHT and O&E MHT using the same hash function, which is not suitable for blockchain because of its complexity. Figure 16 shows an example of execution time with MMT and MMHT when comparing three different datasets using SHA3.

The RIPEMD-128 hash function with the MMHT consensus algorithm also has a low execution time, whereas the improvement percentage is 51.6% compared with that using MHT, shown in Figure 17. It is noted that we did not consider the data maintenance functions [25] when manipulating the structure of the transaction chain. While changing of the block order in the proposed MMHT algorithm comes with an advantage of reduced execution time, it also increases the complexity of executing data maintenance functions, especially data recovery compared with the original MHT.

6. Conclusion

The use of blockchain-based IoT systems for smart homes can provide high security against possible data security threats. Recent studies have found that blockchain networks are highly effective and secure due to their advanced features like smart contracts, which keep a strong check over the activities and transactions over the network. The consensus algorithm based on PoW secures the network via a validator responsible for handling all communication verifications between the blockchain network nodes within the smart homes. This work is unique because to the best of our knowledge, there are no previous studies that has attempted an investigation of multiple hash functions for a consensus algorithm, as well as different data sizes for testing

blockchain performance. In terms of data integrity verification check, the results show that the proposed modified Merkle hash tree (MMHT) consensus algorithm used in the blockchain has a very efficient execution time.

This system can be effectively used by smart homes to provide the safest systems for high data security and integrity.

However, the results have shown that the relationship between the number of transactions, consensus algorithms, and hash functions is not straightforward. The blockchain network needs to be designed so that it can be adaptive to different conditions in the network. Even though the proposed MMHT algorithm gives a significant advantage in the simulation, the current study also has limitations in terms of the lack of testing of the proposed system in a real environment. In addition to this, the concern about the relationship between a smart electronic contract and its legal counterpart can cause inefficiencies and barriers to the networks' operation. The lack of a legal status for smart contracts in the current laws is a significant issue that needs further investigation.

Data Availability

The dataset used in the implementation of this study is included and explained within the article and referenced in reference [71].

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

Part of this work is supported by the Malaysian Ministry of Higher Education and Universiti Kebangsaan Malaysia (Grant number: GUP-2021-023).

References

[1] P. Ii, *Part II Cryptocurrencies and Blockchain Applications Applications with Blockchain*, Scrivener Publ. LLC, 2020.

[2] M. Dopico, A. Gomez, D. De la Fuente, N. García, R. Rosillo, and J. Puche, "A vision of industry 4.0 from an artificial intelligence point of view," in *Proc. 2016 Int. Conf. Artif. Intell. ICAI 2016- WORLDCOMP 2016*, pp. 407–413, Athens, 2016.

[3] Z. Mubeen, M. Afzal, Z. Ali, S. Khan, and M. Imran, "Detection of impostor and tampered segments in audio by using an intelligent system," *Computers and Electrical Engineering*, vol. 91, article 107122, 2021.

[4] H. N. Dai, Z. Zheng, and Y. Zhang, "Blockchain for Internet of things: a survey," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8076–8094, 2019.

[5] H. Cho, "ASIC-resistance of multi-hash proof-of-work mechanisms for blockchain consensus protocols," *IEEE Access*, vol. 6, no. c, pp. 66210–66222, 2018.

[6] M. Maslin, M. Watt, and C. Yong, "Research methodologies to supportthe development of blockchain standards," *Journal of ICT Standardization*, vol. 7, no. 3, pp. 249–268, 2019.

[7] U. Bodkhe, S. Tanwar, K. Parekh et al., "Blockchain for industry 4.0: a comprehensive review," *IEEE Access*, vol. 8, pp. 79764–79800, 2020.

[8] D. Guha Roy, P. Das, D. De, and R. Buyya, "QoS-aware secure transaction framework for Internet of things using blockchain mechanism," *Journal of Network and Computer Applications*, vol. 144, pp. 59–78, 2019.

[9] Q. Nasir, I. A. Qasse, M. Abu Talib, and A. B. Nassif, "Performance analysis of hyperledger fabric platforms," *Security and Communication Networks*, vol. 2018, 14 pages, 2018.

[10] J. Cynthia, H. P. Sultana, M. N. Saroja, and J. Senthil, *Security Protocols for IoT, no. 2020*, Springer International Publishing, 2019.

[11] K. Hao, J. Xin, Z. Wang, and G. Wang, "Outsourced data integrity verification based on blockchain in untrusted environment," *World Wide Web*, vol. 23, no. 4, pp. 2215–2238, 2020.

[12] E. Androulaki, A. Barger, V. Bortnikov et al., "Hyperledger fabric: a distributed operating system for permissioned blockchains," in *Proc. 13th EuroSys '18: Thirteenth EuroSys Conference 2018*, Porto Portugal, 2018.

[13] S. Alsaqqa and S. Almajali, "Blockchain technology consensus algorithms and applications: a survey," *International Journal of Interactive Mobile Technologies (iJIM)*, vol. 14, no. 15, pp. 142–156, 2020.

[14] G. W. Peters and E. Panayi, "Understanding modern banking ledgers through blockchain technologies: future of transaction processing and smart contracts on the Internet of money," in *Banking Beyond Banks and Money*, Springer, Cham, 2016.

[15] M. Moniruzzaman, S. Khezr, A. Yassine, and R. Benlamri, "Blockchain for smart homes: review of current trends and research challenges," *Computers and Electrical Engineering*, vol. 83, article 106585, 2020.

[16] M. Salimitari, M. Chatterjee, and Y. P. Fallah, "A survey on consensus methods in blockchain for resource-constrained IoT networks," *Internet of Things*, vol. 11, article 100212, 2020.

[17] P. M, M. Malviya, M. Hamdi et al., "5G based Blockchain network for authentic and ethical keyword search engine," *IET Communications*, vol. 2021, 2021.

[18] S. N. Makhadmeh, M. A. al-Betar, Z. A. A. Alyasseri et al., "Smart home battery for the multi-objective power scheduling problem in a smart home using grey wolf optimizer," *Electronics*, vol. 10, no. 4, p. 447, 2021.

[19] S. A. Maghdid, H. S. Maghdid, S. R. HmaSalah, K. Z. Ghafoor, A. S. Sadiq, and S. Khan, "Indoor human tracking mechanism using integrated onboard smartphones Wi-Fi device and inertial sensors," *Telecommunication Systems*, vol. 71, no. 3, pp. 447–458, 2019.

[20] H. M. Kim, H. Turesson, M. Laskowski, and A. F. Bahreini, "Permissionless and permissioned, technology-focused and business needs-driven: understanding the hybrid opportunity in blockchain through a case study of insolar," *IEEE Transactions on Engineering Management*, pp. 1–16, 2020.

[21] S. Beg, A. Anjum, M. Ahmad et al., "A privacy-preserving protocol for continuous and dynamic data collection in IoT enabled mobile app recommendation system (MARS)," *Journal of Network and Computer Applications*, vol. 174, article 102874, 2021.

[22] H. Hosseinian, H. Shahinzadeh, G. B. Gharehpetian, Z. Azani, and M. Shaneh, "Blockchain outlook for deployment of IoT in distribution networks and smart homes," *International*

Journal of Electrical and Computer Engineering (IJECE), vol. 10, no. 3, pp. 2787–2796, 2020.

[23] P. Sandner, J. Gross, and R. Richter, "Convergence of blockchain, IoT, and AI," *Frontiers in Blockchain*, vol. 3, 2020.

[24] B. Ali and A. I. Awad, "Cyber and physical security vulnerability assessment for IoT-based smart homes," *Sensors*, vol. 18, no. 3, pp. 817–817, 2018.

[25] M. U. Hassan, M. H. Rehmani, and J. Chen, "Privacy preservation in blockchain based IoT systems: integration issues, prospects, challenges, and future research directions," *Future Generation Computer Systems*, vol. 97, pp. 512–529, 2019.

[26] S. Shetty, C. Kamhoua, and L. Njilla, *Blockchain for Distributed Systems Security*, John Wiley & Sons, Inc., 2019.

[27] M. Liu, K. Wu, and J. J. Xu, "How will blockchain technology impact auditing and accounting: permissionless versus permissioned blockchain," *Current Issues in Auditing*, vol. 13, no. 2, pp. A19–A29, 2019.

[28] D. Minoli, "Positioning of blockchain mechanisms in IOT-powered smart home systems: a gateway-based approach," *Internet of Things*, vol. 10, article 100147, 2020.

[29] M. Yahuza, M. Y. I. B. Idris, A. W. B. A. Wahab et al., "Systematic review on security and privacy requirements in edge computing: state of the art and future research opportunities," *IEEE Access*, vol. 8, pp. 76541–76567, 2020.

[30] M. S. Ferdous, M. J. M. Chowdhury, M. A. Hoque, and A. Colman, "Blockchain consensus algorithms: a survey," 2020, http://arxiv.org/abs/2001.07091.

[31] Brilliant, *Merkle Tree*, Springer, 2016.

[32] S. Khan, A. Gani, A. W. Abdul Wahab et al., "Towards an applicability of current network forensics for cloud networks: a SWOT analysis," *IEEE Access*, vol. 4, pp. 9800–9820, 2016.

[33] A. Niakanlahiji and J. H. Jafarian, "WebMTD: defeating cross-site scripting attacks using moving target defense," *Security and Communication Networks*, vol. 2019, 13 pages, 2019.

[34] H. Wang and J. Zhang, "Blockchain based data integrity verification for large-scale IoT data," *IEEE Access*, vol. 7, pp. 164996–165006, 2019.

[35] A. M. Sagheer, M. S. Al-Ani, and O. A. Mahdi, "Ensure security of compressed data transmission," in *2013 Sixth International Conference on Developments in eSystems Engineering*, pp. 270–275, Abu Dhabi, United Arab Emirates, 2013.

[36] A. Basil Ghazi, O. Adil Mahdi, and W. Badee Abdulaziz, "Lightweight route adjustment strategy for mobile sink wireless sensor networks," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 21, no. 1, pp. 313–320, 2021.

[37] T. L. N. Dang and M. S. Nguyen, "An approach to data privacy in smart home using blockchain technology," in *Proc. -2018 Int. Conf. Adv. Comput. Appl. ACOMP 2018*, pp. 58–64, Ho Chi Minh City, Vietnam, 2018.

[38] L. Hang and D. H. Kim, "Design and implementation of an integrated iot blockchain platform for sensing data integrity," *Sensors*, vol. 19, no. 10, p. 2228, 2019.

[39] S. Sicari, A. Rizzardi, L. A. Grieco, and A. Coen-Porisini, "Security, privacy and trust in Internet of things: the road ahead," *Computer Networks*, vol. 76, pp. 146–164, 2015.

[40] L. König, Y. Korobeinikova, S. Tjoa, and P. Kieseberg, "Comparing blockchain standards and recommendations," *Future Internet*, vol. 12, no. 12, p. 222, 2020.

[41] R. Leszczyna, *Cybersecurity in the Electricity Sector*, Springer, 2019.

[42] I. Karamitsos, M. Papadaki, and N. B. Al Barghuthi, "Design of the blockchain smart contract: a use case for real estate," *Journal of Information Security*, vol. 9, no. 3, pp. 177–190, 2018.

[43] B. K. Mohanta and D. Jena, "An overview of smart contract and use cases in blockchain technology," in *2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, pp. 1–4, Bengaluru, India, 2018.

[44] M. Maximilien, A. Vallecillo, J. Wang, and M. Oriol, "Service-oriented computing," in *15th International Conference, ICSOC 2017*, pp. 229–237, Malaga, Spain, November, 2017.

[45] F. Daniel and L. Guida, "A service-oriented perspective on blockchain smart contracts," *IEEE Internet Computing*, vol. 23, no. 1, pp. 46–53, 2019.

[46] H. Wang, X. A. Wang, S. Xiao, and J. S. Liu, "Decentralized data outsourcing auditing protocol based on blockchain," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 2, pp. 2703–2714, 2021.

[47] R. Kalis and A. Belloum, "Validating data integrity with blockchain," in *2018 IEEE International Conference on Cloud Computing Technology and Science (CloudCom)*, pp. 272–277, Nicosia, Cyprus, August, 2018.

[48] R. Zambrano, "Taming the beast: harnessing blockchains in developing country governments," *Frontiers in Blockchain*, vol. 2, pp. 1–15, 2020.

[49] B. Bhushan, A. Khamparia, K. M. Sagayam, S. K. Sharma, M. A. Ahad, and N. C. Debnath, "Blockchain for smart cities: a review of architectures, integration trends and future research directions," *Sustainable Cities and Society*, vol. 61, article 102360, 2020.

[50] S. J. Alsunaidi and F. A. Alhaidari, "A survey of consensus algorithms for blockchain technology," in *2019 International Conference on Computer and Information Sciences (ICCIS)*, pp. 1–6, Italy, 2019.

[51] A. Meneghetti, M. Sala, and D. Taufer, "A survey on pow-based consensus," *Annals of Emerging Technologies in Computing*, vol. 4, no. 1, pp. 8–18, 2020.

[52] T. Alam, "IoT-Fog: a communication framework using blockchain in the Internet of things," 2019, http://arxiv.org/abs/1904.00226.

[53] K. S. Gorniak and A. M. Kudin, "Aspects of blockchain reliability considering its consensus algorithms," *Theoretical and Applied Cybersecurity*, vol. 2, no. 1, pp. 5–9, 2020.

[54] S. M. H. Bamakan, A. Motavali, and A. Babaei Bondarti, "A survey of blockchain consensus algorithms performance evaluation criteria," *Expert Systems with Applications*, vol. 154, article 113385, 2020.

[55] L. Chen, L. Xu, N. Shah, Z. Gao, Y. Lu, and W. Shi, "On security analysis of proof-of-elapsed-time (PoET)," in *Stabilization, Safety, and Security of Distributed Systems*, pp. 282–297, Springer, Cham, 2017.

[56] W. F. Silvano and R. Marcelino, "Iota Tangle: a cryptocurrency to communicate Internet-of-things data," *Future Generation Computer Systems*, vol. 112, pp. 307–319, 2020.

[57] M. Salimitari and M. Chatterjee, "A survey on consensus protocols in blockchain for IoT networks," 2018, http://arxiv.org/abs/1809.05613.

[58] M. Salimitari, M. Chatterjee, and Y. P. Fallah, "A survey on consensus methods in blockchain for resource-constrained IoT networks," *Internet of Things*, vol. 11, article 100212, 2020.

[59] M. Bartoletti, S. Lande, and A. S. Podda, "A proof-of-stake protocol for consensus on bitcoin subchains," in *Financial*

Cryptography and Data Security, pp. 568–584, Springer, Cham, 2017.

[60] X. Fu, H. Wang, and P. Shi, "A survey of Blockchain consensus algorithms: mechanism, design and applications," *Science China Information Sciences*, vol. 64, no. 2, pp. 1–15, 2021.

[61] G. Kumar, R. Saha, M. K. Rai, R. Thomas, and T. H. Kim, "Proof-of-work consensus approach in blockchain technology for cloud and fog computing using maximization-factorization statistics," *IEEE Internet of Things Journal*, vol. 6, no. 4, pp. 6835–6842, 2019.

[62] S. S. Hazari and Q. H. Mahmoud, "A parallel proof of work to improve transaction speed and scalability in blockchain systems," in *2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC)*, pp. 916–921, Las Vegas, NV, USA, 2019.

[63] A. Gervais, G. O. Karame, K. Wüst, V. Glykantzis, H. Ritzdorf, and S. Čapkun, "On the security and performance of proof of work blockchains," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pp. 3–16, Vienna Austria, 2016.

[64] Y. Ren, Q. Zhao, H. Guan, and Z. Lin, "A novel authentication scheme based on edge computing for blockchain-based distributed energy trading system," *EURASIP Journal on Wireless Communications and Networking*, vol. 2020, no. 1, 2020.

[65] Y. Zhao, "Research on personal credit evaluation of Internet finance based on blockchain and decision tree algorithm," *EURASIP Journal on Wireless Communications and Networking*, vol. 2020, no. 1, 2020.

[66] A. Ju, Y. Guo, Z. Ye, T. Li, and J. Ma, "HeteMSD: a big data analytics framework for targeted cyber-attacks detection using heterogeneous multisource data," *Security and Communication Networks*, vol. 2019, 9 pages, 2019.

[67] D. Lee and N. Park, "Blockchain based privacy preserving multimedia intelligent video surveillance using secure Merkle tree," Multimedia Tools and Applications, 2020.

[68] B. Schneier and B. Schneier, "Cryptography in context," in *Secrets and Lies: Digital Security in a Networked World*, pp. 102–119, Wiley Publishing, Inc., 2015.

[69] H. Isyanto, A. S. Arifin, and M. Suryanegara, "Design and implementation of IoT-based smart home voice commands for disabled people using Google Assistant," in *2020 International Conference on Smart Technology and Applications (ICoSTA)*, Las Vegas, NV, USA, 2020.

[70] M. H. Miraz and M. Ali, "Integration of blockchain and IoT: an enhanced security perspective," *Annals of Emerging Technologies in Computing*, vol. 4, no. 4, pp. 52–63, 2020.

[71] B. Podgorelec, *Dataset of Transactions of 10 Ethereum Addresses Controlled by a Private Key, Each Has At Least 2000 Output Transactions, Which Include a Transfer of Cryptocurrency, and All Transactions Are Performed within No Longer Than Three Months Period*, 2019.

[72] M. S. Niaz and G. Saake, "Merkle hash tree based techniques for data integrity of outsourced data," *CEUR Workshop Proceedings*, vol. 1366, pp. 66–71, 2015.