Distributed Blockchain Wireless Sensor Network Architecture for Malicious Node and Sensor Data Detection

Aswadul Fitri Saiful Rahman, Andani Achmad*, and Wardi
Electrical Engineering, Faculty of Engineering, Hasanuddin University, Makassar, Indonesia
Email: rahmanafs23d@student.unhas.ac.id (A.F.S.R.), andani@unhas.ac.id (A.A.), wardi@unhas.ac.id (W.)

Manuscript received February 27, 2024; revised May 29, 2024; accepted June 18, 2024
*Corresponding author

***Abstract*—Wireless Sensor Network (WSN) is a private network with many sensors. It has the ability acquire sensor data with reduced energy consumption but considered vulnerable, primarily due to centralization. Therefore, disruptive blockchain technology capable of hashing block-based security was recommended to overcome the problem. This research proposes an appropriate architectural model for integrating WSN and distributed blockchain networks to detect suspicious sensor nodes and data. The model is implemented using WSN hardware with a mesh topology and integrated software components, including a blockchain system. The results showed that with 10–500 block node iterations and three distributed Peer servers, the Scrypt hash had the fastest execution time of 469.49 seconds compared to the others. The combination of two consensus mechanism methods, including smart contract (Mac Address and SHA-256 on the microcontroller) and Proof of Information (PoI) to the three sensor parameters (Temperature or Temp, Humidity or Humd, and gas concentration or MQ2) led to detection and maintenance of sensor nodes and data reaching 97.37% validity. Moreover, sensor control consisting of two Radio Frequency Identification (RFID) nodes and one Quick Response (QR) code node applied to 500 block node iterations led to 592.47 seconds execution time longer than the monitoring sensors. This showed that the separation of monitoring and controlling sensors was the best for the architecture model proposed. The major reason was that the monitoring sensors prioritized automation, speed, and data accuracy in real time. Meanwhile, the controlling sensors focused on the interaction between users and devices, specifically when integrated into a transaction system.**

***Index Terms*—blockchain, hash algorithms, proof of information, smart contract, Wireless Sensor Network (WSN)**

I. Introduction

Wireless Sensor Network (WSN) and the Internet of Things (IoT) are interconnected technologies with different protocol standards. Both technologies rely on centralized control, albeit in different ways. For example, IoT is a technology developed primarily for interoperability, security, and scalability [1]. Meanwhile, WSN is a private network designed to ensure low energy consumption in the process of acquiring sensor data [2, 3]. Both technologies are often combined to access and monitor data from outside the WSN despite its private status [4]. However, the network is vulnerable to attacks on data and sensor nodes, specifically in relation to centralization issues. This has led to the recommendation of disruptive blockchain technology that is capable of hashing block-based security to overcome the issues through consensus mechanisms and hashing algorithms.

Blockchain is a technology developed based on inter-blocks to store data and has hash encryption for block sign which is important for subsequent connection [5]. It is also defined as distributed and structured digital ledger widely used for transaction management. Some of the concepts developed through blockchain include peer-to-peer network, distributed ledgers, consensus mechanisms, smart contracts, and the use of other applications considered necessary [6]. Moreover, Bitcoin and Ethereum use blockchain for cryptocurrency transactions through the security algorithms known as SHA-256 and Ethash respectively [7]. This application allows the transfer of assets and distribution of information through smart contracts to network [8]. For example, blockchain version 3.0 is often used specifically to decentralize governance and regulation [9]. It has also been widely used to improve the security of WSN and IoT [10].

WSN is an embedded system for the interconnection of different sensor points into network and is designed with low power consumption based on the IEEE 802.15.4 communication standard [11]. It has been applied in fields such as military, forest fires, vehicle movement detection, environmental monitoring, industry, agriculture, healthcare, smart homes, transportation, and smart cities [12]. However, the security is a significant concern due to the complex structure and vulnerability of the technology to internal and external attacks [13]. This is possible because open wireless communications lead to the vulnerability of data and sensor nodes to both active and passive attacks. Therefore, network security is implemented in the form of policies, mechanisms, and services that prevent unauthorized access and illegal network use. It is important to state that the conceptualization of security mechanisms requires considering the limitations of network infrastructure [14].

Blockchain with interconnected hash capabilities can be

doi: 10.18178/ijeetc.14.4.219-232

used to detect the existence of interference or manipulation in stored sensor data. This is achievable through the previous and current hash of each block. For example, the manipulation of a block out of 1000 can provide a sign to show error in the next block data. This capability can be used to overcome or detect manipulation of WSN sensor data. In addition to hash, blockchain is capable of validating data through a proof-based consensus mechanism. The consensus develops constantly based on technological needs and the examples include Proof of Work (PoW), Proof of Stake (PoS), Proof of Location (PoL), smart contract, and others [15]. Research has been conducted in recent years regarding the use of blockchain to maintain WSN data security. The benefits and challenges have also been discussed, specifically the decentralized systems with sha256 hashing [16]. The report showed that the strengths included the consensus and decentralized mechanism for managing the integrity of data stored in the ledger in addition to the prioritization of security and reliability over efficiency [17]. Fig. 1 shows the centralization, decentralization, and distribution of WSN and the combination with blockchain data flow.

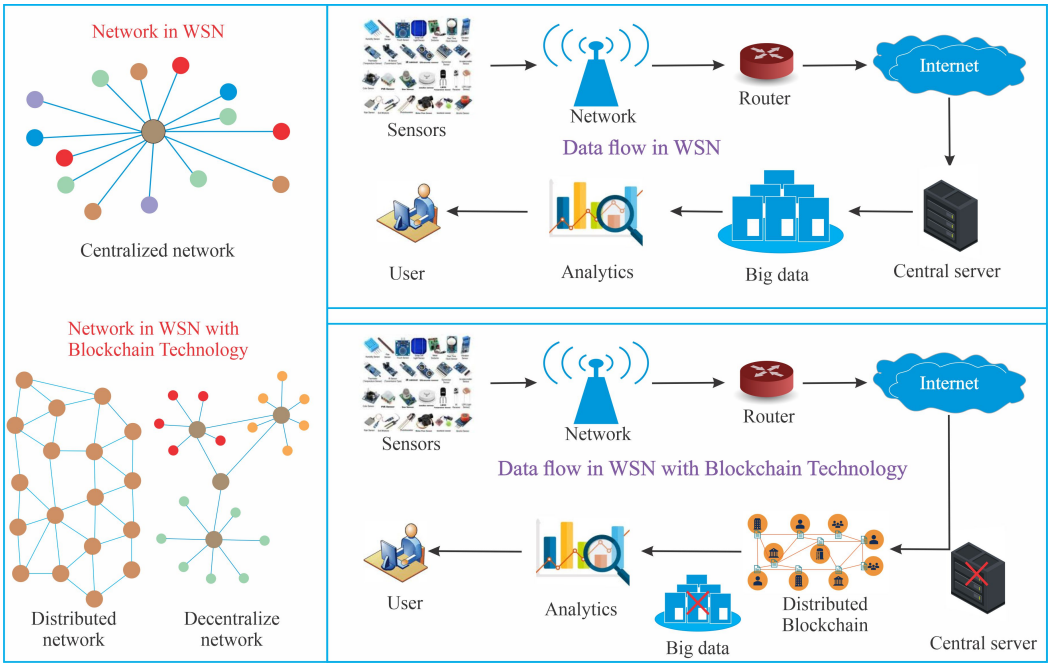

Fig. 1. Centralization, decentralization, and distribution of WSN and BWSN data flows [16].

Blockchain serves not only as a data recording mechanism but also as a means to ensure the integrity and secure transmission of sensor data across multiple points in the network. To enhance efficiency and automation in the sensor data validation process, smart contracts are introduced. Smart contracts are digital scripts that run on top of the blockchain and automatically execute predefined rules. Smart contracts were applied in detecting malicious nodes [18]. However, due to the dynamic and high-volume nature of sensor data, not all data needs to be recorded on the blockchain. Therefore, we propose a validation approach based on PoI, which serves as a selection mechanism to filter sensor data based on information quality using predefined thresholds [19]. The integration of blockchain, smart contracts, and PoI enables a more efficient, reliable, and tamper-resistant system within distributed WSN environments.

Several related articles used SHA-256 as blockchain hash and the application of the algorithm was observed to be very numerous and diverse. However, when applied to large volumes of sensor data, SHA-256 can significantly impact execution time and storage requirements in distributed systems. Therefore, this study investigates the use of alternative hashing algorithms for blockchain, with the results used to develop a WSN-blockchain architectural model aimed at detecting sensor data and node-level attacks with improved efficiency in terms of time complexity and storage utilization.

The objective of this study is to integrate smart contract consensus mechanisms with PoI to protect both sensor nodes and sensor data from unauthorized or malicious attacks within a distributed system.

The sensor data framework employs two main functional layers. The first is Blockchain Sensor Monitoring (BSM), which manages real-time monitoring of sensor data, including the validation of data authenticity and integrity before it is recorded on the blockchain. This layer is responsible for detecting anomalies or data manipulation at an early stage, as well as filtering information based on predefined thresholds.

The second layer, Blockchain Sensor Control (BSC), is responsible for managing access control, node authorization, and transaction handling between sensors or with the central system. At this layer, mechanisms including smart contracts, MAC address verification, and UID authentication are employed to guarantee that only

authorized entities are permitted to interact with or perform updates on the system. Smart contracts and PoI play a vital role in strengthening sensor node identity parameters and related data attributes, including personal area network identifiers (PAN IDs), MAC addresses, threshold values, and key hashing algorithms.

II. Related Works

Previous research focused on WSN and blockchain models using different methods and mechanisms. For example, [18] performed a malicious node detection using a trust model and smart contracts to track ID and location. The model was applied to sensor nodes but was not tested with data. Moreover, [17] provided data protection model with blockchain using Raspberry Python (Raspberry Pi) as a server. The weakness identified was the insufficiency of the memory for sensor data.

The other researches by [20] and [21] developed a model with smart contract to filter cluster headers as well as monitor and select good data quality for subsequent conversion into a block. The research depended on external components, such as solidity and zookeeper. Furthermore, [22] and [23] used PoW for data security but the method consumed significant resources which led to some consequences when the sensor data increased. Another research by [24] applied UID authentication to protect nodes through a tangle mechanism. The method could produce a new block but the weakness was the lack of authentication in blockchain mechanism [19]. Implemented the PoI concept to develop a block when valid data was met but the weakness was the absence of sensor nodes protection. The trend observed from several articles showed the existence of weaknesses and advantages in modeling sensor data security. Therefore, a new architectural model was proposed through the combination of several methods to detect and protect sensor nodes and data from dangerous attacks such as manipulation. The proposed model was compared with previous research in Table I.

Table I: State-of-the-Art and Proposed Model

WSN dan Blockchain	**Wei She (2019) [18]**	**Sung-Jung Hsiao (2021) [17]**	**Hafsa Benaddi (2021) [20]**	**Hoang T. Tran (2022) [22]**	**Huanhuan Feng (2022) [21]**	**Sanjeev Kumar Dwivedi (2023) [23]**	**Yuling Chen (2022) [24]**	**Weiwei Qi (2023) [19]**	**Proposed New Model**
Smart Contract	√		√	√	√				√
PoI/PoW				PoW		PoW		PoI	PoI
Authentifikasi Mac/UID							√		√
Authentifikasi ID	√					√	√		√
Sensor Monitoring (BSM)		√	√		√	√	√	√	√
Sensor Control (BSC)									√
Hash256	√	√	√	√	√	√	√	√	√
Scrypt, Bcrypt, Argon2									√
Other GAP	Location tracking	Raspberry Pi as server (limited memory)	Solidity	Smart contract description, PoW simulation	Kafka dependency on external zookeeper	Audit storage data with blockchain	Focus on node registration and tangle mechanism	PoI for valid data amount	Device identity authentication, smart contract, PoI three sensors, testing the use of several hash algorithms, using BSC and BSM.
Test/ validation	OPNET	Prototype	Simulation	PoW simulation does not match the concept	Simulation	Simulasi Scyther and platform Ethereum	Matlab simulation	Matlab simulation	Prototype (microcontroller, zigbee, multisensor) and real peer server distributed.

A column was provided in the table for the proposed new model which was WSN architecture and distributed blockchain developed to detect dangerous nodes and sensor data. The purpose was to compare the model with state-of-the-art research. This was necessary because attention was not on the security of sensor nodes and data using dual smart contract consensus methods with a focus on security and detection as well as the application of PoI to ensure protection from the multi-sensor. Therefore, the novelty was the division of the architecture into two sensor models or multi-sensors, including BSM and BSC. The two architectural models had several different parts due to the variation in their functions. The first step for sensor nodes detection was registering the MAC address, PAN ID, and Key Hashing. The second step was to set sensor threshold limit to suspect sensor data forgery. The validation of the first and second steps led to the development of a block by the system.

The key hashing embedded in the microcontroller was tested using several hashing algorithms such as SHA-256, Bcrypt, Scrypt, and Argon2. The aim was to determine an efficient algorithm model that could not exhaust the memory. Subsequently, a consensus mechanism that matched the two sensor models with Smart contract and PoI was proposed. For BSC, smart contract mac address and UI RFID mechanism was suggested. The new proposal

was based on the validation of control or transactions, including Proof of 2FA which was not discussed in this research. Moreover, appropriate algorithm model was tested for hashing in the blockchain.

Table I compares recent approaches in integrating blockchain with WSNs, highlighting limitations such as reliance on PoW consensus [18, 21], limited smart contract usage [17, 20], and basic identity authentication mechanisms. Most prior works do not incorporate comprehensive sensor control or monitoring frameworks, nor do they utilize advanced validation techniques. In contrast, the proposed model integrates smart contracts with a PoI-based validation mechanism for multiple sensor inputs, enhanced authentication (e.g., Scrypt, Bcrypt), and is implemented in a real-world distributed setup, offering a more complete and scalable solution compared to earlier simulations and prototypes.

Blockchain decentralization is part of the concepts often used to duplicate data. The summary of research conducted using decentralization with SHA-256 is presented in Table II. It was observed that all research that focused on integrating WSN into blockchain used SHA-256 hashing algorithm with different types of consensus and sensors. Therefore, this research was conducted to evaluate both centralized and decentralized systems using three peer storage setups, including Peer A, Peer B (Online), and Peer C (Online). Several hashing algorithms were also tested with the distributed systems.

TABLE II: DATA PAPER WITH HASH256 FROM THE LAST FILTER

Ref.	Storage	Sensor	Consensus
[25]	Decentralization	Sensor Monitoring	PoA
[26]	Decentralization	Node Sensor Identification	Credit System
[27]	Decentralization	Light Sensor	Time windowing method
[28]	Decentralization	Sensor Monitoring	Consensus of miner nodes
[29]	Decentralization	Node Sensor Identification	NA
[30]	Decentralization	Soil and Temperature, etc.	NA
[23]	Decentralization	Health sensor	NA
[31]	Decentralization	NA	PoW
[32]	Decentralization	Monitoring Sports sensor	NA
[33]	Decentralization	Monitoring Sensor	NA
[34]	Decentralization	Body Sensors	PoS
[35]	Decentralization	NA	PoAh
[36]	Decentralization	NA	PoW
[37]	Decentralization	Node Sensor	PoW, PoA
[38]	Decentralization	NA	PoW

As presented in Table I and Table II, various approaches have been proposed to integrate security into WSNs using blockchain technology. However, several limitations persist in previous works, including memory constraints in sensor data processing [17], lack of device identity authentication [23], dependency on external components [20, 21], and limited flexibility of consensus mechanisms in handling multi-sensor data [22, 23]. Therefore, a new model is required to address these challenges with greater efficiency and security. The model proposed in this study introduces a distributed blockchain-based WSN architecture comprising a dual-layer design: BSM and BSC. These layers are designed to independently yet cohesively manage sensor data security and sensor transaction control. The system employs a PoI consensus mechanism to validate only legitimate data, while smart contracts ensure node authentication through MAC addresses, PAN IDs, and UID-based key hashing (SHA-256). Furthermore, lightweight hashing algorithms such as SHA-256, Bcrypt, Scrypt, and Argon2 are evaluated to optimize memory usage and reduce execution time on resource-constrained devices, without compromising the reliability of data integrity and node security.

III. THEORETICAL BACKGROUND

A. WSN

WSN is network of sensor nodes that connect different devices such as end devices, routers, sink nodes, and coordinators. It has the capacity to communicate through hop-to-hop or multi-hop mechanisms [39]. Fig. 2 is an example of network design with the Zigbee protocol from sensor to the central server, which consists of a coordinator, router, and end device.

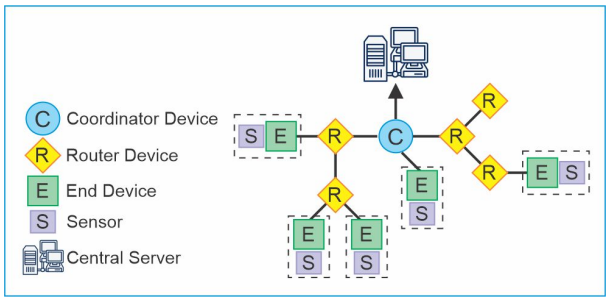

Fig. 2. Network design with Zigbee protocol [40].

Sensor network consists of several stations called sensor nodes which are small in size, portable, and integrated into transducers, microcontrollers, and power sources. The WSN system is a collection of sensor nodes in several places interconnected to form network. Data on sensor nodes are collected at the center via a wireless RF module. Moreover, network system focuses on low power consumption at a low price [41]. This shows that the main concerns for sensor nodes architecture are energy efficiency, size reduction, and minimum cost [13].

B. Security WSN

Network security can be defined as a set of policies, mechanisms, and services preventing unauthorized access and illegal use of network [14]. A significant concern is on WSN security due to its complex structure and vulnerability to internal and external attacks [13]. This is a challenge to be resolved in order to ensure the protection of WSN information. Some of the possible attack methods include jamming, eavesdropping, packet replays, packet changes or spoofing, and node replication. Some others

include Sybil, wormhole, sinkhole, DoS (denial-of-service), node breaches, and injections of bogus messages [42]. The attacks are further classified into active and passive based on the nature of the transmission media [43]. In an active attack, the attacker or hacker actively searches for and destroys information, whereas a passive attack involves the unauthorized acquisition of valuable information, such as passwords or confidential data [4].

Security in WSNs encompasses protection against inherent vulnerabilities such as limited computational power, open data transmission, and ad-hoc network infrastructure that is highly susceptible to attacks. These challenges make WSNs particularly vulnerable to node impersonation, data manipulation, as well as physical and network-based attacks. Therefore, WSN security systems must be lightweight yet effective in addressing threats such as sensor data tampering, node forgery, and communication disruption. The key security challenges in WSNs include node authentication, data confidentiality, data integrity, availability, data authenticity, and inter-node consistency. Due to the lack of persistent identities, sensor nodes are vulnerable to impersonation attacks such as Sybil and node replication [42]. Limited processing and memory capabilities often prevent the use of encryption, exposing data to potential eavesdropping . Furthermore, sensor data can be modified during transmission, leading to replay and injection attacks. The energy-constrained nature of sensor nodes makes them susceptible to DoS attacks like flooding and jamming [42]. Even when data appears valid, it may originate from malicious nodes, resulting in false data injection . Finally, the distributed structure of WSNs introduces synchronization and consistency issues, particularly evident in problems such as forks in distributed ledgers.

C. Blockchain Technology

Blockchain is a disruptive technology currently being developed to ensure security, specifically for financial transactions. It was first produced in 1991 by Stuart and Harber and continued by Satoshi Nakamoto in 2009 [10]. The development process led to cryptocurrencies such as bitcoin. Moreover, blockchain was explained as a ledger developed based on blocks used to store transactions. The blocks were encrypted using hash algorithm containing the transaction, timestamp, and previous hash [44]. Another research defined blockchain as a giant sequential database or spreadsheet that extended beyond the classic financial ledger. This was due to the ability to record transactional information securely using cryptography and was governed by a consensus mechanism. It is a combination of technologies (P2P networking, cryptography, and distributed ledger) [45].

The technology enables secure data transmission based on a very complex encryption system. This is possible because each block contains creation time information linked to the previous block as shown in Fig. 3. The trend shows that blockchain is designed to combat fraud and data alteration.

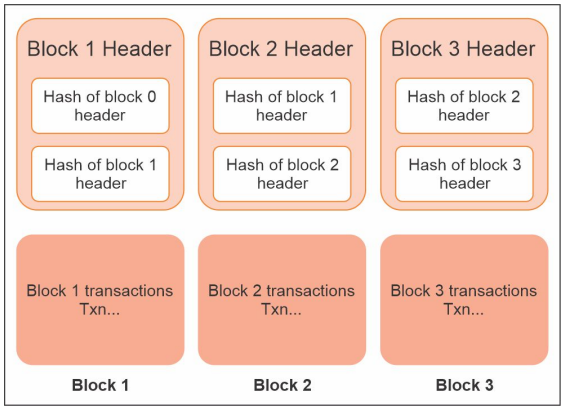

Fig. 3. Block structure in blockchain [44].

Blockchain is inseparable from smart contracts and proof-based consensus mechanisms. The smart contracts are used during transactions between two parties and validated transactions included in blockchain. Meanwhile, the PoW consensus algorithm was developed to validate the transactions between blockchain nodes (users). There are other consensus algorithms such as PoS, PoL, practical byzantine fault tolerance (PBFT), and others. The implementation of the proof-based algorithm is an indication that nodes in blockchain network show sufficient evidence of having the privilege to add new blocks to the main chain and collect rewards [15].

Blockchain is particularly well-suited for WSNs due to its ability to provide decentralized trust, immutability, and data integrity without relying on centralized authority. Its distributed consensus mechanisms mitigate single points of failure, while cryptographic techniques such as hash chaining and digital signatures protect sensor data from tampering, impersonation, and replay attacks. Moreover, the inherent transparency and auditability of blockchain enhance trust and accountability among distributed sensor nodes. These characteristics establish blockchain as a robust foundation for strengthening WSN security, especially when combined with lightweight consensus mechanisms such as PoI, which are tailored to the limitations of resource-constrained sensor devices.

D. Encryption and Hashing

Encryption and hashing functions are important concepts in cryptography. The main difference is the possibility of retrieving the input value after being changed. This is possible because encryption hides a message by changing the form value to ensure unauthorized parties cannot read the content. The hidden message is called plaintext and the result of the encryption is ciphertext. Moreover, authorized parties need a secret key to hide and read the message. Encryption can be written through the formula $E(P)=C$ while Decryption is $D(C)=P$.

Hashing is a scheme that allows plain text password input for subsequent conversion into hash value by considering the function, iteration, and salt. The iteration parameter is optional and often applied to determine the number of consecutive executions for the hash used. Moreover, the number of iterations can be adjusted to

provide a consideration computation time or key stretching for hash value calculation. Some of the functions used for password hashing include MD5, SHA1, SHA-256-SHA512, PBKDF, Bcrypt, Scrypt, and Argon2. This research was conducted using 4 hashing, including SHA-256 which was the default for Blockchain as well as Bcrypt, Scrypt, and Argon2.

E. Microcontroller

A microcontroller is a chip used for electronic circuits with inputs, outputs, and programs that can be written and erased. A microcontroller company, Atmel, has an AVR chip and also develops a small physical computing device, Arduino, which is widely used on microcontroller boards. This device runs at 16 MHz with an 8-bit core and has limited memory of 32 kilobytes storage and 2 kilobytes of RAM [46]. Arduino Uno is a microcontroller board developed based on the Atmega328 which has a high-performance Atmel 8-bit AVR RISC (little instruction code and addressing). The microcontroller board also has 14 digital I/O pins, 6 analog pins, a 16 MHz crystal oscillator, 32 KB flash memory, 2 KB SRAM, a power jack, and a reset button [47].

IV. Research Method

A. System Design

This research proposes an appropriate architectural model for integrating WSN and blockchain networks to detect suspicious sensor nodes and data. This model is implemented using WSN hardware with a mesh topology and integrated software components, including a blockchain system, rather than relying on conventional simulation tools. The software utilized in this research includes Python for smart contract execution, PoI implementation, and block creation; an SQL database for storing registration and block data; and a web interface for displaying distributed blockchain data. This was achieved by dividing the WSN sensor model into BSM and BSC with several different parts due to the variations in their functions. BSM is responsible for real-time sensor data monitoring, validating data authenticity and integrity before it is recorded on the blockchain, detecting anomalies at an early stage, and filtering information based on predefined threshold values. In contrast, BSC manages access control, node authorization, and transaction handling between sensors. Mechanisms such as smart contracts, MAC address verification, and UID authentication are implemented in the BSC layer to ensure that only authorized entities can interact with the system.

The first step for sensor nodes detection was to register the MAC address, PAN ID, and Key Hashing. The second step was to set sensor threshold limit required to detect sensor data forgery. The validation of the first and second steps allowed the system to develop a block. Moreover, the key hashing embedded in the microcontroller was tested using several hashing algorithms, including SHA-256, Bcrypt, Scrypt, and Argon2 to determine the most efficient or fastest. A consensus mechanism that matched the two sensor models such as smart contract and PoI was further proposed. For the BSC model, smart contract consensus mechanism, including Mac, SHA-256 key, and UID was proposed. UID was registered first due to its application for control or transactions required to verify data and form blocks. The mechanism is widely applied in control processes such as room access or financial transactions with the aim of protecting the users. It was applied to sensors designed for control or transactions such as fingerprint, RFID, and QR codes. The trend shows the capacity of the model developed to protect data integrity through smart contracts, PoIs, or other blockchain consensus mechanisms in addition to sensor nodes. Furthermore, distributed blockchain system was experimented through the application of three peer servers. One of the peer servers served as a master database for authentication and filtering of sensor nodes or dangerous data. Meanwhile, the other two were integrated into the online database to store duplicate valid data.

B. Procedural Steps for Sensor Data Security and Validation

1) Sensor node initialization

Each sensor node is first registered into the system through an initial authentication process. The information submitted into a smart contract includes:

- MAC Address (a unique device identifier),
- PAN ID (Personal Area Network Identifier),
- Key Hashing derived from the device's UID or unique token, encrypted using a lightweight hashing algorithm.

2) Sensor data monitoring and acquisition (BSM layer)

Active nodes transmit data in real-time. Each incoming data packet is subjected to:

- Authenticity verification using the UID hash and node identity,
- Integrity checks to ensure proper format and acceptable value ranges,
- Threshold-based filtering to remove data that falls outside of predefined criteria.

3) Consensus validation and block formation (PoI layer)

If the data passes validation, PoI consensus algorithm determines whether the information holds sufficient significance and integrity (e.g., it originates from a trusted node and meets all threshold criteria).

- Only data deemed valid and relevant are eligible for processing.
- This data is then formed into a new block and added to the blockchain by a designated trusted node.

4) Access control and transactions (BSC layer)

All control activities, inter-node transactions, and data verifications are managed in the BSC layer. Security mechanisms implemented at this stage include:

- Smart Contracts that validate access rights based on node roles and permissions. Only nodes that meet the smart contract rules may initiate or respond to transactions.
- Multi-Factor Identity Verification, including:
  RFID numbers mapped to specific nodes or users,

Unique QR codes linked to individual sensor identities or data sets.

These mechanisms strengthen entity authentication prior to accepting or executing data interactions.

- Incoming Data Audits:

All sensor data must undergo visual or digital validation via QR code or RFID scanning before being deemed legitimate for blockchain entry. This additional control layer prevents unauthorized data injection from unregistered nodes.

C. Smart Contract and PoI Scenarios in Attack Testing

The smart contract variable is utilized to protect sensor node data, which has been registered in the database and includes parameters such as PAN ID, MAC address, and a SHA-256 hash key. In contrast, sensor data integrity is safeguarded using PoI method. PoI is a consensus approach in distributed systems designed to verify the validity of information recorded on the blockchain. In this context, sensor data is only committed to the blockchain if it meets specific validity criteria, such as: 1) the data falls within a predefined normal threshold range, 2) the data does not exhibit anomalous behavior, and 3) the sending node has a reliable trust record. The formal model of PoI can be formulated as an integration of these three key aspects:

$$\text{PoI}_{\text{score}} = \alpha\times\text{Entropy(Data)} + \beta\times\text{Trust(Node)} + \gamma\times\text{Relevance}$$

Here Entropy(Data) measures the degree of uncertainty or uniqueness in the data, Trust(Node) evaluates the credibility of a node based on its history of transmitting valid data, Relevance assesses the relevance of the data in relation to the system's context or requirements, and α, β, γ represent the weights assigned to each parameter, adjusted according to the system design, with the constraint that $\alpha + \beta + \gamma = 1$.

In this study, PoI approach is simplified using a threshold-based filtering method. The validity of sensor data is determined by whether the data falls within a predefined normal range. For each sensor x_i the validation can be expressed as an indicator function:

$$\delta_i(x_i) = \begin{cases} 1 \text{ if } L_i < x_i < U_i \\ 0 \text{ if } x_i \le L_i \text{ or } x_i \ge U_i \end{cases}$$

where x_i is the value of the ith sensor (e.g., temperature, humidity, MQ2), L_i and U_i are the lower and upper thresholds defining the valid range for the sensor value, and $\delta_i(x_i)$ is the indicator function for the ith sensor.

If $\sum_{i=1}^{n} \delta_i(x_i) = n$ then create a new block since all sensor readings are valid.

Alternatively, in Boolean logic form:

$$\text{Block\_Valid} \begin{cases} 1 \text{ if } \wedge_{i=1}^{n} \delta_i(x_i) = 1 \\ 0 \text{ otherwise} \end{cases}$$

Example for three sensors:

$$\delta_1(\text{temp}) \wedge \delta_2(\text{hum}) \wedge \delta_3(\text{MQ2}) = 1 \Rightarrow \text{Valid} \rightarrow \text{Store to Block}$$

The pseudocode of the PoI method is as follows:

```
Input:
    - Sensor readings: temp, hum, MQ2
    - Thresholds: temp_range (L1, U1), hum_range (L2, U2), MQ2_range (L3, U3)
Function IsValid(x, L, U):
    Return 1 if (L < x < U) else 0
Begin:
    V_temp = IsValid(temp, L1, U1)
    V_hum  = IsValid(hum, L2, U2)
    V_MQ2  = IsValid(MQ2, L3, U3)
      If V_temp AND V_hum AND V_MQ2 == 1:
          CreateNewBlock(temp, hum, MQ2)
      Else:
          RejectDataOrMarkAsSuspicious()
```

This study applies a threshold-based filtering method to ensure sensor data validity before blockchain recording. Data is considered valid within the following ranges: 20°C to 40 °C for temperature, 40% to 70 % RH for humidity, and 200 ppm to 600 ppm for gas concentration (MQ2). These thresholds are selected to represent normal environmental conditions and to exclude anomalous readings from the distributed system.

The details of the attacks and mitigation strategies used for testing are comprehensively outlined in Table III.

TABLE III: SECURITY THREATS AND MITIGATION STRATEGIES IN WSN-BLOCKCHAIN SYSTEMS USING SMART CONTRACTS AND POI

Threat	Mitigation Strategy	Role of Smart Contract	Role of PoI
Sybil Attack	- Node identity verification using unique credentials and sensor data validation	Automatically verifies nodes before approving transactions	Accepts only verified data within valid threshold ranges
Replay Attack	- Timestamping - Unique nonce in each transaction	Rejects transactions with outdated or duplicate timestamps	Accepts only verified data within valid threshold ranges
Sensor Data Manipulation	- Threshold validation	Rejects transactions deemed invalid by contract rules	Indicator function: only data within valid ranges is stored
Malicious/ Untrustworthy Node	- Automatic blacklisting	Locks or disables nodes with poor reputation	Accepts only verified data within valid threshold ranges
Data Tampering	- Data encryption and hashing prior to storage	Contracts accept only data with matching node key hashes	Checks sensor data integrity and validates threshold ranges
Eavesdropping	- End-to-end encryption	Manages encryption and authentication within smart contracts	Indirectly involved by maintaining validity of verified data
Consensus Failure/ Blockchain Data Inconsistency	- Periodic synchronization among nodes	Ensures consistent block order and hash validation	Invalid data is excluded from blocks, maintaining uniformity

D. Model Design

The WSN sensor was divided into BSM and BSC as discussed in the previous subsection. The BSM model used smart contract mac address consensus mechanism and UID. The application of UID was to protect sensor data which were also used as the controller. Moreover, the BSC sensors such as fingerprint, RFID, and QR Codes allowed the model to protect data integrity using smart contracts, PoIs, or other blockchain consensus mechanisms in addition to sensor nodes.

The proposed architecture model is presented in Fig. 4 with the BSM and BSC sensors observed to be integrated into one server point or centralization. The model did not eliminate centralization but added decentralization to blockchain. This was achieved through the inclusion of a WSN network with a Mesh topology.

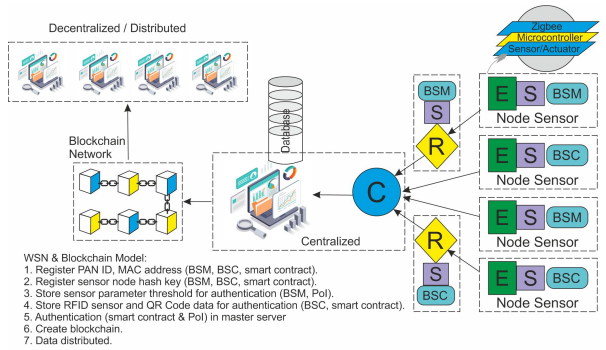

Fig. 4. BSC and BSM flow diagram.

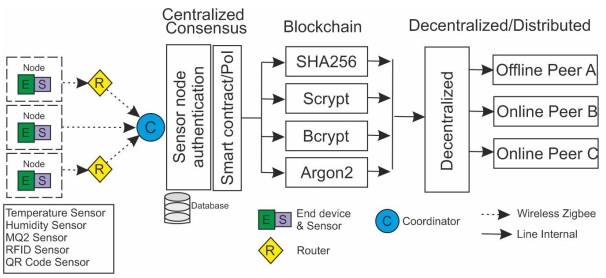

Fig. 5. First scenario on WSN data flow to blockchain.

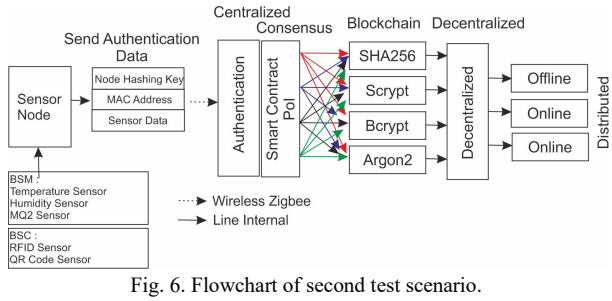

Fig. 6. Flowchart of second test scenario.

The first test scenario was conducted using only smart contracts integrated into centralized and decentralized three distributed peer nodes, including A and B (online) as well as C (online). The test concentrated on hashing algorithms used to construct blockchain by comparing SHA-256, Scrypt, Bcrypt, and Argon2 for each peer server. The results were evaluated to provide recommendations related to the application of hashing algorithms in WSN network containing thousands of nodes and vast amounts of sensor data. The focus was on the execution time challenges and blockchain programs depending on hash algorithms selected. The test scenario is presented in the following Fig. 5.

Each of the four hashing algorithms showed unique characteristics, specifically in the generation process. Moreover, the second test scenario presented in Fig. 6 shows the efficiency of hashing algorithm and consensus mechanism for BSM and BSC. Sensor and wireless transmission module on node integrated into the microcontroller were tested for several algorithms as identities. This was followed by the assessment of the authentication through the consensus mechanism in the centralized WSN. After the validation process, blockchain with an algorithm was formed through the decentralized system with access provided offline, online, or hybrid. The algorithms on the microcontroller and blockchain could be different.

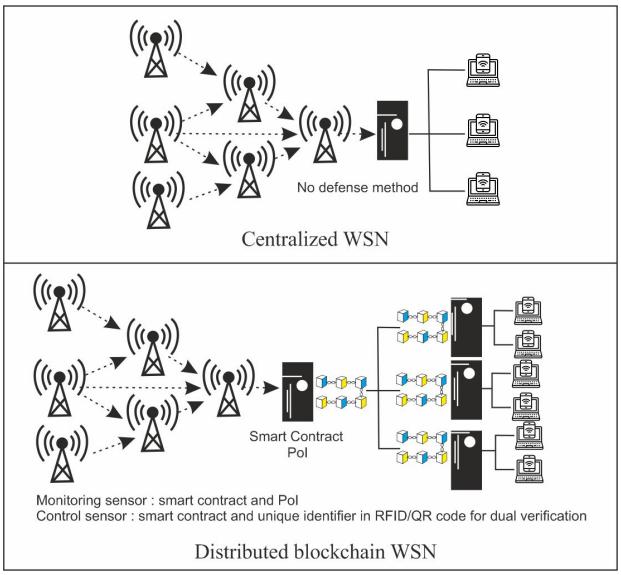

Fig. 7. Comparative overview diagram illustrating the differences between a centralized WSN and the proposed distributed WSN based on blockchain architecture.

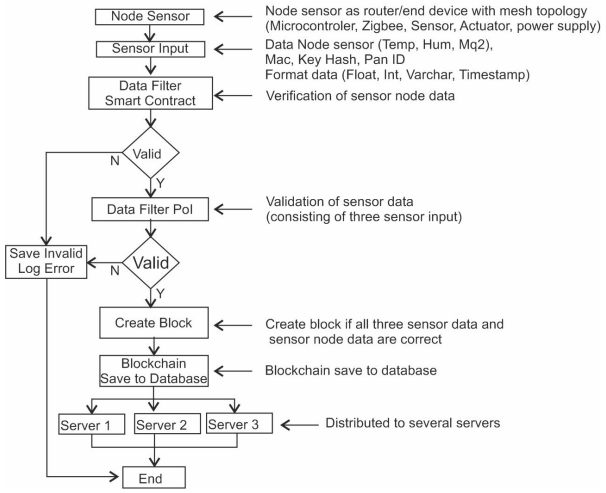

Fig. 8. Data capture to blockchain process.

Centralized WSN and distributed blockchain-based WSN exhibit fundamental differences in data flow and the security protection of both sensor nodes and sensor data. As illustrated in Fig. 7, we propose an enhanced security defense mechanism tailored to the monitoring and control layers of sensor operation prior to the creation of distributed blocks. For sensor monitoring, we employ a consensus approach based on smart contracts and PoI mechanism. Meanwhile, for sensor control, we integrate smart contracts with a unique identifier embedded in RFID or QR codes to enable dual-layer verification. The data capture process for forming a distributed blockchain is illustrated in Fig. 8.

V. Results and Discussion

A. WSN Node with Distributed Blockchain

WSN node with distributed blockchain in the form of SHA-256, Scrypt, Bcrypt, and Argon2 was tested to determine the fastest execution time and memory capacity to store data. This was achieved using a WSN prototype with 3-5 Zigbee xbee nodes and three monitoring sensors. Moreover, 10 – 500 blockchain iterations were conducted using three peer servers distributed including one offline and two online.

1) WSN node with SHA-256

Fig. 9 shows that the time required to send data to the three peer servers using SHA-256 is approximately 506.62 seconds.

2) WSN node with Scrypt

Fig. 10 shows that the time required by Scrypt to iterate 500 blocks of three peer servers is 469.49 seconds.

3) WSN node with Bcrypt

Fig. 11 shows that the time required for Bcrypt to iterate 500 blocks of three peer servers is 705.87 seconds.

4) Node WSN with Argon2

Argon2 completed 500 blocks of three peer servers in 602.84 seconds as shown in Fig. 12.

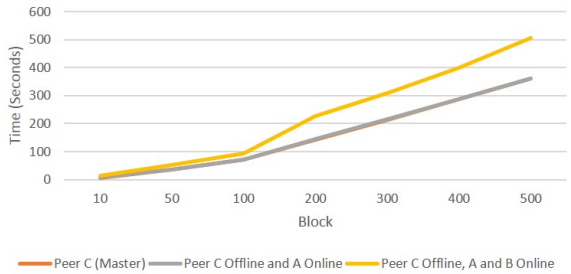

Fig. 9. WSN node with SHA-256.

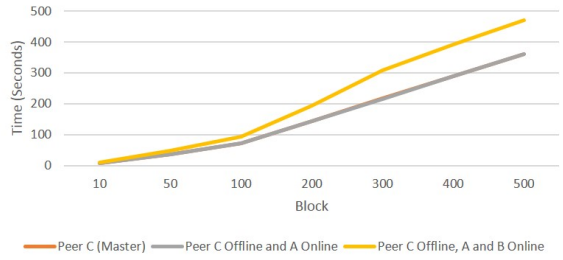

Fig. 10. WSN node with Scrypt.

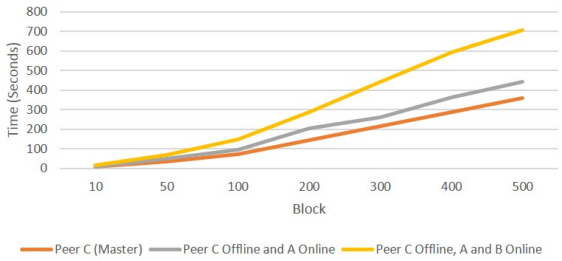

Fig. 11. WSN node with Bcrypt.

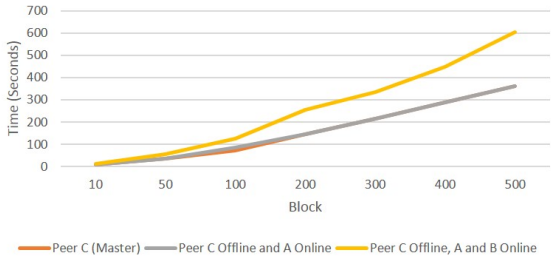

Fig. 12. WSN node with Argon2.

5) Comparison of execution time

Fig. 13 compares the time of execution for the four hashing algorithms applied to the three peer servers. The results showed that Scrypt completed the process for 500 blocks at 469.49 seconds faster than SHA-256, Bcrypt, and Argon2. This led to the application of Scrypt for subsequent analysis of the proposed model. Argon2 was observed to have the largest storage capacity while Bcrypt had the smallest.

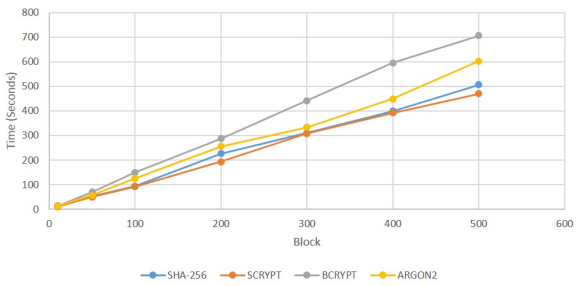

Fig. 13. Execution time comparison.

B. Comparison of WSN Blockchain Using Only Scrypt with the Inclusion of SHA-256

This research compared the WSN blockchain using only Scrypt with the SHA-256 embedded in the microcontroller as dual authentication on sensor nodes. This is further explained in the following subsections.

1) Comparison of the Execution Time between Scrypt and the Inclusion of SHA-256 in Node Sensor in the WSN Blockchain

The results showed that the addition of security authentication or smart contract to sensor nodes or microcontroller led to a longer execution time as presented in Fig. 14. The difference was estimated at 63.54 seconds, showing the advantage of adding SHA-256 to sensor nodes for double security authentication. Therefore, SHA-256 was used for further testing and applied to the architecture

for the purpose of adding smart contract parameters.

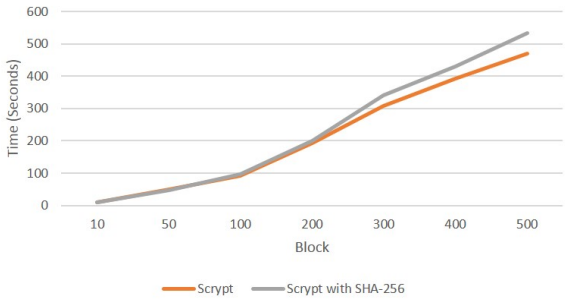

Fig. 14. Comparison of execution time between WSN blockchain Scrypt and the inclusion of SHA-256 in node sensor.

2) Comparison of the memory size between Scrypt and the inclusion of SHA-256 in node sensor in the WSN blockchain

Fig. 15 shows that the addition of security authentication or smart contract to sensor nodes or microcontroller increases the memory size to 132 Kb. This was because hash data sent was stored in each block in blockchain.

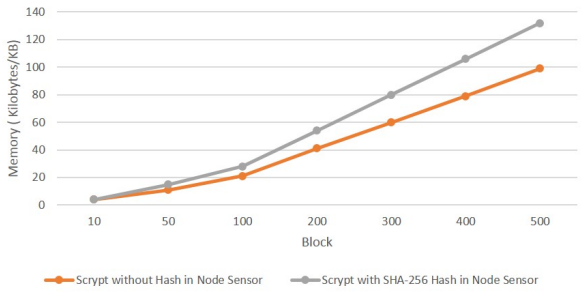

Fig. 15. Comparison of blockchain memory size in WSN (Scrypt vs. Scrypt with SHA-256 hashing in sensor nodes).

C. Detection Capacity of WSN with Blockchain

WSN prototype was tested using 3-5 Zigbee xbee nodes with three monitoring sensors. The scenario focused on applying the Scrypt hash algorithm with the fastest time to 10 to 500 iterations of three peer servers distributed online, including one master and two online. The primary objective of this experiment was to detect valid and invalid (including malicious) sensor nodes and data, which may result from manipulation or hardware faults. Below is an explanation of what constitutes valid and invalid data:

- Valid data: Sensor data originating from nodes whose identities were authenticated via smart contracts, using MAC address, PAN ID, key hashing (based on UID), RFID, and QR Code. These data were further validated using PoI mechanism, based on threshold criteria for temperature, humidity, and MQ2 gas levels.
- Invalid data: Data from unregistered nodes, tampered or manipulated sensor readings, values outside predefined thresholds, or those affected by Sybil attacks, replay attacks, and other types of malicious intrusions.

1) Detection without node registration

Table IV shows the results of the detection test conducted using three sensor nodes, three distributed peer servers, and 10-500 blockchain node iterations. It was observed that the external nodes with the same PAN ID not recorded entered distributed server. For example, only 304 of the 400 block nodes iterated in the server were found to be valid after check and analysis while 96 were from outside.

2) Detection using MAC address registration

Detection test conducted using three sensor nodes, three distributed peer servers, and 10-500 blockchain node iterations, showed that the data executed and distributed were only the registered MAC nodes. Illegal nodes were stored in the designated data log. However, the method had a weakness which was observed from the storage of the manipulated sensor data, including temp, humd, and MQ2, on the server even though node was registered. Table V shows the results of detection test conducted with a registered mac address. It was observed that 264 registered and 36 unregistered nodes were detected. Moreover, 84 out of the 264 registered had manipulated or invalid data, leading to a total of 201 considered valid.

TABLE IV: DATA DETECTION WITHOUT REGISTRATION

No	Block	Outer Nodes	Number Sent to Server
1	10	2	10
2	50	12	50
3	100	22	100
4	200	45	200
5	300	67	300
6	400	96	400
7	500	138	500

TABLE V: DATA DETECTION USING NODE REGISTRATION

Block	Registered MAC Address Nodes	Unregistered MAC Address Nodes	Registered MAC Address Nodes with Manipulated Data	Actual Number of Valid Nodes
10	8	2	3	5
50	41	9	14	27
100	81	19	23	58
200	158	42	36	122
300	264	36	63	201
400	358	42	84	274
500	460	40	106	354

TABLE VI: DATA DETECTION USING SMART CONTRACT (MAC AND SHA-256 HASH)

Block	MAC: OK, SHA-256: empty, data sensor: invalid or valid	MAC: OK, SHA-256: available but not registered, data sensor: valid	MAC: OK, SHA-256: OK, data sensor: invalid (Temp, Humd, MQ2)	Actual valid nodes, MAC: OK, SHA-256: OK, data sensor: valid (Temp, Humd, MQ2)
100	27	13	7	53

3) Detection using smart contract (MAC addresses and SHA-256)

Table VI shows the result of the detection test for nodes using MAC registration and the inclusion of the SHA-256 hash algorithm in the microcontroller. The result of 100 iterations conducted on block nodes showed that 60 were

valid based on registration identity, 27 were detected without SHA-256 hash, and 13 were identified with different hash algorithms. After rechecking, 7 block nodes had invalid or manipulated sensor data and this led to the validation of only 53 block nodes. The trend showed the need for a more accurate method in the form of a PoI consensus mechanism.

4) Detection using smart contract (MAC and SHA-256 Hashing) and PoI (Temp)

Fig. 16 shows the application of smart contract and PoI Temp to 100 block node iterations. It was observed that 57% of data entered distributed servers or were considered valid while the rest were detected as invalid according to the information. Reanalysis showed that 8% out of the 57% were invalid sensor data 2 (humd) and 11% were invalid sensor data 3 (MQ2). The method also had a weakness which was the inability to detect other sensors.

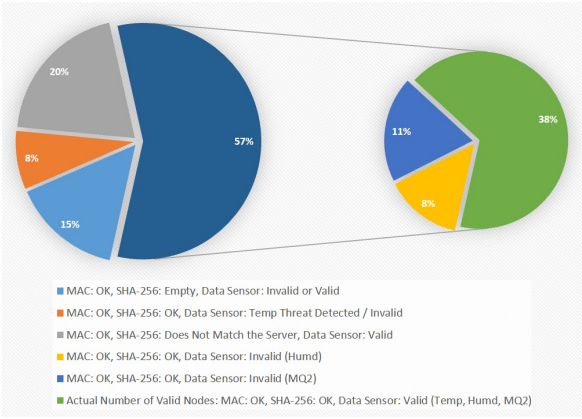

Fig. 16. Detection using smart contract (MAC and SHA-256 hash) and PoI (Temp).

5) Detection using smart contract (MAC and SHA-256 Hash) and PoI (Temp, Humd)

Fig. 17 shows the application of smart contracts and PoI (Temp, Humd) to detect sensor nodes and data. The results showed that 53% of data in 100 block node iterations in distributed servers were considered valid while the rest were invalid. It was further observed from the reanalysis that 13% of the 53% were MQ2 invalid, thereby indicating approximately 40% were valid. The method also had a weakness of not detecting the 3rd sensor due to the presence of manipulated or damaged data.

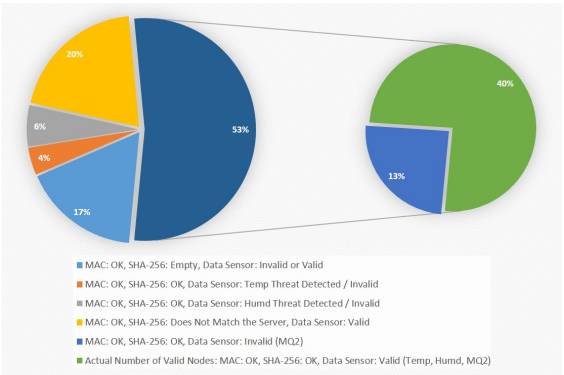

Fig. 17. Detection using smart contract (MAC and Hash SHA-256) and PoI (Temp, Humd).

6) Detection using smart contract (MAC and SHA-256 Hash) and PoI (Temp, Humd, MQ2)

Fig. 18 shows the application of smart contract and PoI (Temp, Humd, MQ2) to 100 block node iterations. The results showed that 38% of data entered distributed servers or was considered valid while the rest were detected as invalid. According to the information from PoI, 1% of the 38% had an error caused by the influence of network. This combined method was able to detect valid and invalid nodes as well as the types of sensors. The percentages shown represent the number of validated blocks out of a total of 100 evaluated nodes.

- The value of 38% in the section "Actual number of valid nodes..." indicates that 38 blocks were successfully verified as valid (i.e., MAC and SHA-256 matched, and sensor values were within acceptable thresholds).
- Similarly, other values such as 15%, 8%, 4%, and so on each represent the number of blocks out of a total of 100 blocks.

In total, 100 blocks were analyzed, with 38 confirmed as fully valid and eligible for inclusion in the blockchain. The information presented in Table VII showed that the application of two consensuses, including smart contract and PoI increased the valid data in distributed servers to 97.37%. The remaining 2.63% was associated with the inability of network to distribute data to all servers. Although numerically small, this residual error has critical implications for the resilience of the WSN. In security-sensitive applications, even a small percentage of undelivered or unvalidated data can lead to system inconsistencies, decision delays, or exposure to replay or injection attacks.

Category A represents the initial category, which relies solely on MAC registration and is capable of detecting only 17 invalid blocks. However, 83 blocks (T) were not identified as invalid and were mistakenly considered valid. After conducting a data audit, it was found that 45 invalid blocks (I) were embedded within 38 valid blocks (V). The formulas for calculating the percentages of valid and invalid data are as follows:

$$\text{Valid } (\%) = \left(\frac{V}{T}\right) \times 100 \quad \text{and} \quad \text{Invalid } (\%) = \left(\frac{I}{T}\right) \times 100$$

Thus, Category A has an undetected invalid data percentage of 54.22%. The formula is intended for categories B and C.

In Category B, the system successfully detected 29 invalid data blocks. Further analysis and verification revealed that 71 blocks (T) were initially considered valid, comprising 33 undetected invalid blocks (I) and 38 valid blocks (V). The results indicate that 53.52% of the data were valid, while 46.48% were invalid.

In Category C, the system successfully detected 15 invalid MQ2 sensor data entries. However, subsequent analysis revealed that 18 invalid entries had been included in the blocks and merged with 38 verified entries. This occurred because SHA-256 identity verification was disabled, allowing both legitimate and manipulated nodes to be accepted into the system. The percentages of valid (V)

and invalid (I) data were recalculated based on the affected group, where $V = 38$ and $I = 18$, resulting in a total of 56 blocks (T).

In Category D, the system initially detected 38 valid data blocks (100%). However, after a revalidation or data audit, it was found that 1 block was problematic (for example, one block may have been missing on a server or the total number of blocks was not synchronized). The first valid data (V_a) and the new invalid data (I_b) are then calculated to determine the value of the new valid data (V_b), where: $V_b = V_a - I_b$. Therefore, the total block (T) = $V_b + I_b$. The validation was then updated as follows: V_a: 38, I_b:1. Then, the percentage of valid and invalid data (after the audit) can be calculated as follows:

$$\text{Valid (\%)} = \left(\frac{V_b}{T}\right) \times 100 \quad \text{and} \quad \text{Invalid (\%)} = \left(\frac{I_b}{T}\right) \times 100$$

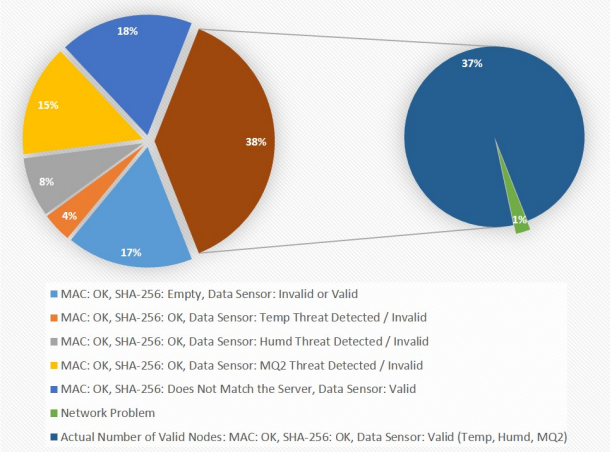

Fig. 18. Detection using smart contract (MAC and SHA-256) and PoI (Temp, Humd, MQ2).

TABLE VII: VALID NODE DATA USING SMART CONTRACT AND POI

Cat.	Category A MAC: OK, SHA-256: empty, data sensor: invalid or valid	Category B MAC: OK, SHA-256: OK, data sensor: Humd threat detected/ invalid	Category C MAC: OK, SHA-256: OK, data sensor: MQ2 threat detected/ invalid	Category D MAC: OK, SHA-256: OK, data sensor: valid (Temp, Humd, MQ2), 1 Block lost/Network Problem
Valid	45.78%	53.52%	67.86%	97.37%
Invalid	54.22%	46.49%	32.14%	2.63%

D. Comparison of Execution Time for the Monitoring and Controlling Sensors

A test was conducted to compare the execution time of the monitoring, controlling, and hybrid sensors. The monitoring sensors consisted of temperature, humidity, and MQ2 in one node. Meanwhile, the controlling sensors were RFID and QR code nodes. The scenarios focused on comparing the three monitoring sensor nodes, two monitoring sensors and one RFID node, two RFID and one monitoring sensor nodes, as well as two RFID and one QR Code node. This was achieved using 10-500 block node iterations with three distributed peer servers.

The results presented in Fig. 19 showed that the execution time was approximately 592.47 seconds when the WSN node was used for the controlling sensors. This value was much longer than the application of only the monitoring sensors which requires 533.03 seconds. The controlling sensors required longer periods due to the manual control operations between the user and the device compared to the fully automatic activities of the monitoring sensors. Meanwhile, the combination of both led to an unstable execution time. This was possibly due to network problems which were identified as part of the factors affecting delivery or transaction process in distributed network.

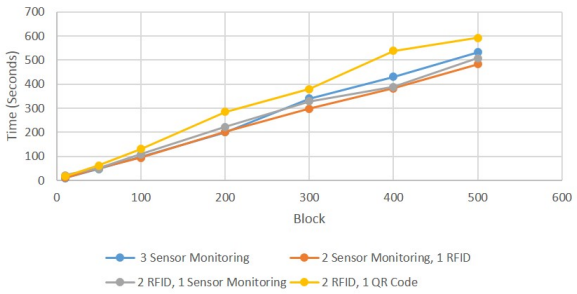

Fig. 19. Comparison of blockchain execution time WSN (sensor monitoring and controlling).

E. Discussion

System scalability in mesh-topology peer networks presents challenges related to latency and bandwidth consumption, which can impact synchronization performance. The proposed architecture demonstrates reliable performance under increased workloads, maintaining minimal error rates even with up to 1,000 blocks. To ensure continued resilience and efficiency at scale, it is recommended to enhance inter-node communication, optimize bandwidth usage, and implement region-based segmentation.

This study specifically focuses on memory efficiency, execution speed, data security, and sensor validation within a distributed blockchain-based WSN architecture—particularly emphasizing node identity verification and data filtering through the consensus mechanisms of smart contracts and PoI. As such, evaluation parameters including latency distribution, energy consumption, network overhead, and error rate were not prioritized during the current experimental phase. Nonetheless, these parameters are recognized as critical for assessing the overall performance of WSN systems, particularly in the context of large-scale and long-term deployments. The evaluation of these aspects is planned for future work, enabling the proposed solution to be assessed not only in terms of data security and validity, but also in terms of network efficiency and resource utilization

VI. CONCLUSION

In conclusion, this study presents a novel approach through a dual-layer architecture (BSM and BSC) that separates monitoring and control functions within

blockchain-based WSNs. By integrating PoI consensus mechanism, smart contracts utilizing UID-based hashing, and evaluations of lightweight hashing algorithms, the model enhances data validity, node security, and processing efficiency on resource-constrained devices. The results showed that the fastest blockchain algorithm for WSN architecture and distributed blockchain in detecting dangerous nodes and sensor data was Scrypt with 469.49 seconds. The addition of the SHA-256 algorithm as smart contract on sensor nodes or microcontroller led to a longer execution time than without hash as observed from the difference of approximately 63.54 seconds. However, the inclusion led to more security for sensor nodes. The combination of two consensus mechanisms, smart contracts (MAC and SHA-256) and PoI three sensor parameters (Temp, Humd, MQ2), detected and maintained 97.37% valid sensor nodes and data. Furthermore, the implementation of the controller sensor comprising two RFID nodes and one QR code node over 500 block node iterations across three distributed peer servers resulted in a total execution time of 592.47 seconds, which is longer than the monitoring sensor's execution time of 533.03 seconds. This was because the process was based on the usage of manual control between the user and the device. The combination of both controlling and monitoring sensors showed an unstable execution time. Therefore, the model architecture was required to separate the monitoring and controlling sensors. This was necessary because the monitoring sensors prioritized automatic operation, speed, and data accuracy in real-time. Future work could focus on optimizing PoI algorithm for real-time deployment in large-scale sensor networks by addressing challenges such as network latency and energy efficiency. Additionally, exploring the integration of edge computing and adaptive threshold mechanisms may enhance system responsiveness and broaden applicability in dynamic real-world environments.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

The code program and installation of prototype components were conducted by Aswadul Fitri Saiful Rahman. Tests, data validation, methodology, and review of manuscripts were done through the efforts of Andani Achmad. Moreover, data analysis, presentation, and management of the research were conducted by Wardi.

ACKNOWLEDGMENT

This research was funded by the 2024 FISCAL YEAR Research Grant KEMDIKBUDRISTEK.

REFERENCES

[1] S. Kumar, P. Tiwari, and M. Zymbler, "Internet of things is a revolutionary approach for future technology enhancement: A review," *J. Big Data*, vol. 6, no. 1, 2019. doi: 10.1186/s40537-019-0268-2

[2] B. A. Begum and S. V. Nandury, "Data aggregation protocols for WSN and IoT applications – A comprehensive survey," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 35, no. 2, pp. 651–681, 2023.

[3] A. Jangra, "Wireless Sensor Network (WSN): Architectural design issues and challenges," *International Journal on Computer Science and Engineering.,* vol. 2, no. 9, pp. 3089–3094, 2010.

[4] M. Keerthika and D. Shanmugapriya, "Wireless sensor networks: Active and passive attacks - vulnerabilities and countermeasures," *Glob. Transitions Proc.*, vol. 2, no. 2, pp. 362–367, 2021.

[5] G.-T. Nguyen and K. Kim, "A survey about consensus algorithms used in blockchain," *Journal of Information Processing System*, vol. 14, no. 1, pp. 101–128, 2018.

[6] E. M. Adere, "Blockchain in healthcare and IoT: A systematic literature review," *Array*, vol. 14, 2022. doi: 10.1016/j.array.2022.100139

[7] R. M. Jaya, V. D. Rakkhitta, P. Sembiring, I. S. Edbert, and D. Suhartono, "Blockchain applications in drug data records," *Procedia Comput. Sci.*, vol. 216, pp. 739–748, 2023. doi: 10.1016/j.procs.2022.12.191

[8] A. E. Onjewu, N. Walton, and I. Koliousis, "Blockchain agency theory," *Technological Forecasting & Social Change*, vol. 191, pp. 1-10, 2023, Apr. 2022.

[9] M. Xu, X. Chen, and G. Kou, "A systematic review of blockchain," *Financ. Innov.*, vol. 5, no. 1, 2019. doi: 10.1186/s40854-019-0147-z

[10] S. Alsaedy, S. Alraddadi, and A. Owais, "A review on using blockchain in wireless sensor networks," *J. Theor. Appl. Inf. Technol.*, vol. 98, no. 23, pp. 3879–3886, 2020.

[11] N. Patel, H. Kathiriya, and A. Bavarva, "Wireless sensor network using zigbee," *International Journal of Research in Engineering and Technology*, vol. 2, no. 6, pp. 1038–1042, 2013.

[12] F. Taieb, *Wireless Sensor Networks : Technology, Protocols, and Applications,* New Jersey: John Wiley & Sons, Inc*,* 2007.

[13] Z. Huanan, X. Suping, and W. Jiannan, "Security and application of wireless sensor network," *Procedia Comput. Sci.*, vol. 183, pp. 486–492, 2021. doi: 10.1016/j.procs.2021.02.088

[14] S. A. Khah, A. Barati, and H. Barati, "A dynamic and multi-level key management method in Wireless Sensor Networks (WSNs)," *Comput. Networks*, vol. 236, Jun., 2023. doi: 10.1016/j.comnet.2023.109997

[15] B. Wen, Y. Wang, Y. Ding, H. Zheng, B. Qin, and C. Yang, "Security and privacy protection technologies in securing blockchain applications," *Inf. Sci. (Ny).*, vol. 645, 2023. doi: 10.1016/j.ins.2023.119322

[16] C. V. Nguyen, M. T. Nguyen, T. T. H. Le, T. A. Tran, and D. T. Nguyen, "Blockchain technology in wireless sensor network : Benefits and challenges," *Trans. on Computer Networks and Communications*, vol. X, no. Y, pp. 1–4, 2021.

[17] S. J. Hsiao and W. T. Sung, "Utilizing blockchain technology to improve WSN security for sensor data transmission," *Comput. Mater. Contin.*, vol. 68, no. 2, pp. 1899–1918, 2021.

[18] W. She, Q. Liu, Z. Tian, J. Sen Chen, B. Wang, and W. Liu, "Blockchain trust model for malicious node detection in wireless sensor networks," *IEEE Access*, vol. 7, pp. 38947–38956, 2019. doi: 10.1109/ACCESS.2019.2902811

[19] W. Qi, Y. Xia, P. Zhu, S. Zhang, L. Zhu, and S. Zhang, "Secure and efficient blockchain-based consensus scheme for MWSNs with clustered architecture," *Pervasive Mob. Comput.*, vol. 94, p. 101830, 2023. doi: 10.1016/j.pmcj.2023.101830

[20] H. Benaddi, K. Ibrahimi, and H. Dahri, "Toward safety of wireless sensor network based on blockchain," *Advances in Dynamical Systems and Applications*., vol. 16, no. 2, pp. 1705–1723, 2021.

[21] H. Feng, M. Zhang, V. Gecevska, B. Chen, R. Saeed, and X. Zhang, "Modeling and evaluation of quality monitoring based on wireless sensor and blockchain technology for live fish waterless transportation," *Comput. Electron. Agric.*, vol. 193, 106642, 2022. doi: 10.1016/j.compag.2021.106642

[22] H. T. Tran, C. V. Nguyen, and M. T. Nguyen, "A framework of deploying blockchain inwireless sensor networks," *EAI Endorsed Trans. Ind. Networks Intell. Syst.*, vol. 9, no. 32, pp. 1–8, 2022. doi: 10.4108/eetinis.v9i32.1125

[23] S. K. Dwivedi, R. Amin, and S. Vollala, "Design of secured blockchain based decentralized authentication protocol for sensor networks with auditing and accountability," *Comput. Commun.*, vol.

197, pp. 124–140, 2023. doi: 10.1016/j.comcom.2022.10.016
[24] Y. Chen, X. Yang, T. Li, Y. Ren, and Y. Long, "A blockchain-empowered authentication scheme for worm detection in wireless sensor network," *Digit. Commun. Networks*, 2022. doi: 10.1016/j.dcan.2022.04.007
[25] B. N. Sudheer and K. Sujatha, "A brief survey on data aggregation and data compression models using blockchain model in wireless sensor network," in *Proc. of Int. Conf. Innov. Data Commun. Technol. Appl.*, pp. 406–413, 2023.
[26] Z. Cui, F. Xue, S. Zhang *et al.*, "A Hybrid BlockChain-Based Identity Authentication Scheme for Multi-WSN," *IEEE Trans. Serv. Comput.*, vol. 13, no. 2, pp. 241–251, 2020.
[27] S. Verma, S. Kaur, R. Manchanda, and D. Pant, "Essence of blockchain technology in wireless sensor network: A brief study," in *Proc. of 2020 Int. Conf. Adv. Comput. Commun. Mater.*, 2020, pp. 394–398.
[28] T. Jagannadha Swamy, B. Pallavi, V. Amaraveni, Y. Sireesha, and S. Siddarth, "Secure data dissemination in wireless sensor networks with the help of module based blockchain technology," in *Proc. 2023 3rd Int. Conf. Intell. Technol.*, 2023. doi: 10.1109/CONIT59222.2023.10205841
[29] H. Kaschel, S. Cordero, P. Adasme, and C. Ahumada, "Smart agriculture 4.0: Technology recommendations and interoperability of devices, sensors and data management using blockchain," in *Proc. of 2022 IEEE Int. Conf. Autom. Congr. Chil. Assoc. Autom. Control Dev. Sustain. Agric. Syst.*, 2022. doi: 10.1109/ICA-ACCA56767.2022.10006132
[30] K. Godawatte, P. Branch, and J. But, "Use of blockchain in health sensor networks to secure information integrity and accountability," *Procedia Comput. Sci.*, vol. 210, no. C, pp. 124–132, 2022.
[31] J. Zhu, "Real-time monitoring for sport and mental health prevention of college student based on wireless sensor network," *Prev. Med. (Baltim).*, vol. 173, May 2023. doi: 10.1016/j.ypmed.2023.107581
[32] M. Abdussami, R. Amin, P. Saravanan, and S. Vollala, "BSAPM: Blockchain based secured authentication protocol for large scale WSN with FPGA implementation," *Comput. Commun.*, vol. 209, pp. 63–77, Apr. 2023.
[33] K. Hasan, M. J. M. Chowdhury, K. Biswas, K. Ahmed, M. S. Islam, and M. Usman, "A blockchain-based secure data-sharing framework for Software Defined Wireless Body Area Networks," *Comput. Networks*, vol. 211, 109004, Apr. 2022.
[34] Y. F. Ebobissé Djéné, M. S. El Idrissi, P. M. Tardif, A. Jorio, B. El Bhiri, and Y. Fakhri, "A formal energy consumption analysis to secure cluster-based WSN: A case study of multi-hop clustering algorithm based on spectral classification using lightweight blockchain," *Sensors*, vol. 22, no. 20, 2022. doi: 10.3390/s22207730
[35] J. Lee, "Patch transporter: Incentivized, decentralized software patch system for WSN and IoT environments," *Sensors (Switzerland).*, vol. 18, no. 2, pp. 1–35, 2018.
[36] V. L. Vinya, Y. Anuradha, H. R. Karimi, P. B. Divakarachari, and V. Sunkari, "A novel blockchain approach for improving the security and reliability of wireless sensor networks using jellyfish search optimizer," *Electron.*, vol. 11, no. 21, 2022. doi: 10.3390/electronics11213449
[37] H. Zhang, M. Zaman, B. Stacey, and S. Sampalli, "A novel distributed ledger technology structure for wireless sensor networks based on IOTA tangle," *Electron.*, vol. 11, no. 15, 2022. doi: 10.3390/electronics11152403
[38] K. Matusiewicz, J. Pieprzyk, N. Pramstaller, C. Rechberger, and V. Rijmen, "Analysis of simplified variants of SHA-256," in *Proc. Western European Workshop on Research in Cryptology*, Leuven, 2005. doi: 10.1634/theoncologist.2013-0294
[39] A. El Zawawi and A. Ibrahim, "Using Zigbee to build a web-based DCS system," in *Proc. of 2012 IEEE Power and Energy Society General Meeting*, 2012. doi: 10.1109/PESGM.2012.6343966
[40] A. F. S. Rahman, I. W. Mustika, and S. S. Kusumawardani, "Pengelolaan sistem informasi data presensi dengan media transmisi menggunakan sistem wireless sensor network," *SENIATI Proc.,* pp. 1–7, 2016.
[41] X. Liu, "Atypical hierarchical routing protocols for wireless sensor networks : A review," *IEEE Sensors Journal*., vol. 15, no. 10, pp. 5372–5383, 2015.
[42] P. Ramadevi, S. Ayyasamy, Y. Suryaprakash, C. Anilkumar, S. Vijayakumar, and R. Sudha, "Security for wireless sensor networks using cryptography," *Meas. Sensors*, vol. 29, 100874, Aug. 2023. doi: 10.1016/j.measen.2023.100874
[43] I. Butun, P. Osterberg, and H. Song, "Security of the internet of things: Vulnerabilities, attacks, and countermeasures," *IEEE Commun. Surv. Tutorials*, vol. 22, no. 1, pp. 616–644, 2020.
[44] L. Nasraoui, L. Nasraoui, and L. A. Saidane, "Blockchain for WSN and IoT applications," in *Proc. of 2022 IEEE 9th Int. Conf. on Sciences of Electronics, Technologies of Information and Telecommunications (SETIT).,* 2022. doi: 10.1109/SETIT54465.2022.9875746
[45] H. Han, R. K. Shiwakoti, R. Jarvis, C. Mordi, and D. Botchie, "Accounting and auditing with blockchain technology and artificial intelligence : A literature review," *International Journal of Accounting Information Systems.,* vol. 48, 100598 Mar., 2023.
[46] W. Durfee. Arduino Microcontroller Guide. (Oct. 2011). [Online]. Availble: www.me.umn.edu/courses/me2011/arduino/
[47] Y. A. Badamasi, "The working principle of an Arduino," in *Proc. of 2014 11th Int. Conf. Electron. Comput. Comput.*, Sep. 2014. doi: 10.1109/ICECCO.2014.6997578

Aswadul Fitri Saiful Rahman received the bachelor degree in electrical engineering from Muhammadiyah University of Surakarta, central java, Indonesia in 2006 and the master of engineering (M.Eng.) in electrical engineering and information technology from Gadjah Mada University, Yogyakarta, Indonesia in 2016. He is currently a doctoral student in electrical engineering at Hasanuddin University, Makassar, Indonesia. His research areas of interest include wireless sensor network, smart system, blockchain technology and information technology.

Andani Achmad is a lecturer in electrical engineering at Hasanuddin University, he received the bachelor of engineering in 1986 and the master of engineering in 2000 in electrical engineering from Hasanuddin University, Makassar, Indonesia. He received the doctoral in 2010 in electrical engineering from Hasanuddin University. He is currently professor at the Department of Electrical Engineering, Hasanuddin University. His current research interests include electrical engineering, telecommunications, control, computer system, digital system and wireless technology.

Wardi is a lecturer in electrical engineering at Hasanuddin University, he received the bachelor of engineering in 1997 in electrical engineering from Hasanuddin University, Makassar, Indonesia. He received the master of engineering (M.Eng.) in 2006 from University of South Australia and Doctor of Engineering (D.Eng.) in 2012 from Ehime University. He is currently an associate professor at the Department of Electrical Engineering, Hasanuddin University. His current research interests include electrical engineering, telecommunications engineering, cellular communication, data communication and wireless communication network.