Reliability Verification Method for Sensor Data Based on Collaboration of Heterogeneous Wireless Sensor Networks

Haoting ZHANG
Ritsumeikan University
Graduate School of Information
Osaka, Japan
is0487ip@ed.ritsumei.ac.jp

Hiroshi YAMAMOTO
Ritsumeikan University
Graduate School of Information
Osaka, Japan

***Abstract*—Efforts towards the realization of smart cities are gaining increasing attention to improve the efficiency and comfort of social activities. The smart cities are built on the IoT systems that gather and analyze various sensor data from different parts of the city. The wireless sensor networks are essential for constructing the IoT systems but have various vulnerabilities (e.g., data tampering, eavesdropping) to the integrity of the sensor data due to the openness of wireless communication. The existing study proposes a system for detecting anomalies in the sensor data by verifying the correctness of the time-series characteristics. However, it is difficult for the existing system to detect attacks that tamper with the sensor data to simulate the realistic pattern of the time series. Therefore, in this study, we assume that sensor nodes physically close in proximity can generate sensor data with similar information, and propose a new method to ensure the integrity of the sensor data by enabling nodes participating in different sensor networks to collaborate with the adjacent nodes to verify the data. Even if multiple wireless sensor networks use different communication protocols (e.g., ZigBee, Thread), nearby sensor nodes can temporarily change the mode (e.g., BLE) to form a temporal network for data verification. Within the temporal network, each sensor node shares its data with adjacent nodes on the other networks, and compares time-series characteristics to ensure the integrity of the sensor data.**

***Index Terms*—smart city, interaction of sensor networks, data reliability Verification.**

I. Introduction

Efforts towards the realization of smart cities are gaining increasing attention to improve the efficiency and comfort of social activities. The smart cities are built on the IoT systems that gather and analyze various sensors from different parts of the city. The wireless sensor networks are essential for constructing the IoT systems but have various vulnerabilities (e.g., data tampering, eavesdropping) to the integrity of the sensor data due to the openness of wireless communication. In practice, IoT devices such as smart appliances and smart cameras are exposed to various cyberattacks, and the reliability of the data generated by the sensing function of the devices cannot be guaranteed [3]. To maintain data reliability, the emergence of a technology that can detect abnormal patterns and falsify sensor data in real-time is expected.

The existing study proposes a system for detecting anomalies in the sensor data by verifying the correctness of the time-series characteristics [5]. However, it is difficult for the existing system to detect attacks that tamper with the sensor data to simulate the realistic pattern of the time series.

On the other hand, in smart cities, multiple IoT systems developed by different organizations may coexist in the same physical area. In such a situation, the stakeholders of the systems may permit the sensor nodes participating in the different systems to partially collaborate to verify the integrity of the sensor data within the limited time and space.

Therefore, in this study, we assume that sensor nodes physically close in proximity can generate sensor data with similar information, and propose a new method to ensure the integrity of the sensor data by enabling nodes participating in different sensor networks to collaborate with the adjacent nodes to verify the data. Even if multiple wireless sensor networks use different communication protocols (e.g., ZigBee, Thread), nearby sensor nodes can temporarily change the mode (e.g., BLE) to form a temporal network for data verification. Within the temporal network, each sensor node shares its data with adjacent nodes on the other networks, and compares time-series characteristics to ensure the integrity of the sensor data.

II. Related Works and Objective of our Study

A. Lifelog Mesh Sensor Network System Supporting Wake-Up Control Function Based on States of Power Generation

Our previous study proposes a new method for constructing a mesh sensor network system for offices [4]. In the proposed system, the timing of wake-up of sensor devices are controlled so that the data transmission path from each node to the data sink can be established based on the states of power generation by energy harvesting to prolong the lifetime of the sensor nodes with limited energy resources. In the demonstration experiments, the sensor device is designed as a wearable device worn by individuals, and it is confirmed that long-term collection of sensor data related to human behavior can be achieved.

However, the wireless sensor network used in this system has vulnerabilities (e.g., data tampering, eavesdropping) to the integrity of the sensor data due to the openness of wireless communication. Therefore, a new method is needed that can verify data integrity, even on low-performance computers.

B. Blockchains and Smart Contracts for the Internet of Things

The existing study by Konstantinos et al.(2016) proposes a system that combines blockchain technologies and IoT systems to ensure the integrity and protect privacy of the sensor data [6]. In this system, the devices within the IoT system construct a blockchain for decentralized data management and encryption to ensure communication security. Additionally, smart contracts are utilized when the devices exchange data to reliably record the communication logs. This mechanism ensures that data are shared only when specific conditions are satisfied to secure privacy.

However, the system requires a long time whenever the accesses to the smart contract on the blockchain occurs and hence is difficult to achieve the real-time data collection.

C. A Cyber-Physical System to Detect IoT Security Threats of a Smart Home

The existing study by Akm et al.(2020) proposes a cyber-physical system that analyzes the power consumption trends of sensor nodes to detect DDoS and man-in-the-middle attacks [5]. The system detects abnormal trends in the time-series of the power consumption through statistical signal processing and multivariate regression models.

However, it is difficult for the existing system to detect attacks that tamper with the sensor data to simulate the realistic pattern of the time series.

D. Objective of our Study

Existing studies have issues such as the long time required to access the smart contract on the blockchain making real-time data collection challenging, and the difficulty in detecting attacks that manipulate sensor data to simulate the realistic pattern of the time series.

Therefore, in this study, we assume that sensor nodes physically close in proximity can generate sensor data with similar information, and propose a new method to ensure the integrity of the sensor data by enabling nodes participating in different sensor networks to collaborate with the adjacent nodes to verify the data. Even if multiple wireless sensor networks use different communication protocols (e.g., ZigBee, Thread), nearby sensor nodes can temporarily change the mode (e.g., BLE) to form a temporal network for data verification. Within the temporal network, each sensor node shares its data with adjacent nodes on the other networks, and compares time-based characteristics to ensure the integrity of the sensor data.

III. Proposed Sensor Data Verification Method

An overview of the proposed system is shown in Fig. 1. As shown in this figure, the proposed method focuses on a situation where sensor networks employing different wireless

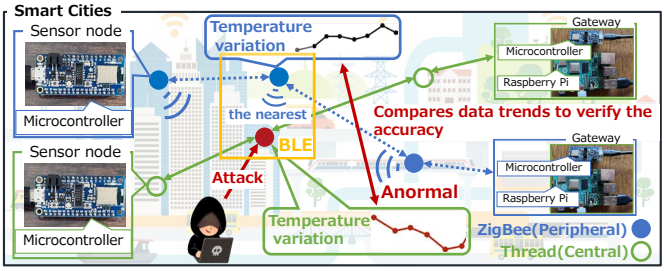

Fig. 1. Overview of the proposed method.

communication protocols coexist within a limited physical area. In the assumed scenario, one sensor network and another one are constructed based on ZigBee and Thread, respectively. In addition, sensor nodes participating in the different sensor networks can collaborate through a common wireless communication protocol (i.e., BLE) to verify the integrity of the sensor data.

A. Overview of Sensor Data Verification Method

When verifying the integrity of the sensor data before transmitting the data on the sensor network, the sensor node shares the data with the adjacent node that may be observing the shared physical space and generating similar sensor data. The data sharing between the sensor nodes belonging to the different sensor networks is facilitated by BLE (Bluetooth Low Energy). Here, the sensor data generated on the Thread-based sensor network is a verification target, and the sensor node on the ZigBee-based sensor network attempts to verify the integrity. During the verification procedure, the sensor nodes on the ZigBee-based sensor network act as peripherals of the BLE and broadcast advertising packets to announce their presence. When the sensor nodes on the Thread-based sensor network act as centrals of the BLE and receive the advertising packets. In addition, the central measures the Received Signal Strength Indicator (RSSI) when receiving the advertising packets to identify the nearest peripheral. The sensor node that requires the verification then shares the sensor data with the nearest sensor node on the ZigBee-based sensor network. After that, the sensor node on the ZigBee-based sensor network compares the received data with its own data to verify the integrity of the sensor data.

B. Hardware Configuration of the Sensor Node

The sensor node consists of a microcontroller (Adafruit Feather nRF52840 Express) supporting various wireless communication protocols (e.g., Thread, ZigBee, and BLE) [7]. Therefore, each sensor node can dynamically change the wireless communication protocol.

C. Characteristics of Wireless Communication Technologies

In this section, we introduce the wireless communication technologies used in this study. The characteristics of the wireless communication technologies are summarized in Tab. I.

TABLE I
CHARACTERISTICS OF THE WIRELESS COMMUNICATION TECHNOLOGIES.

Characteristics	BLE	Thread	ZigBee
Data Rate	Up to 2 Mbps	Up to 250 kbps	Up to 250 kbps
Communication Range	50–100 m	10–30 m	10–100 m
Power Consumption	Very low	Low	Low
Frequency Band	2.4GHz	2.4GHz	2.4GHz
Network Topology	Star, P2P	Mesh	Mesh
Protocol Stack	Bluetooth 5.x	6LoWPAN, IEEE 802.15.4	IEEE 802.15.4

1) Thread: Thread is an IPv6-based low-power wireless networking protocol designed for IoT devices [8]. The Thread provides secure and reliable communication between devices and enables the device to work with low power consumption. On the Thread, the sensor nodes support to construct a mesh network to extend the communication range, enhancing network robustness and scalability. It is commonly used in home automation, smart lighting, and industrial IoT applications.

2) ZigBee: ZigBee is a specification for high-level communication protocols using low-power digital radios based on the IEEE 802.15.4 protocol [9]. It is designed for low data rate, long battery life, and secure networking, making it suitable for home automation, data collection from medical devices and other low-bandwidth wireless applications. Devices supporting the ZigBee can also form a mesh network to extend the communication range and improve resilience.

3) BLE: BLE is a wireless communication protocol designed for short-range data exchange between devices with low power consumption, making it suitable for applications such as fitness trackers, medical devices, and proximity sensors [10]. The BLE is a part of the Bluetooth 4.0 core specification and operates on a 2.4 GHz frequency band with optimized energy efficiency for battery-powered devices. In addition, the BLE can support both point-to-point communication and broadcasting of small-size data enabling various IoT use cases.

IV. SENSOR DATA EXCHANGE ALGORITHM BETWEEN HETEROGENEOUS NETWORKS

In the proposed system, sensor networks collaborate to collect only sensor data whose integrity is verified. Specifically, sensor nodes from different networks (i.e., ZigBee and Thread) temporarily collaborate to exchange and verify sensor data using BLE. In this section, we focus on ZigBee and Thread sensor networks coexisting within the same area, and describe an algorithm for exchanging sensor data between two sensor nodes in the different networks.

A. Method for Changing Wireless Communication protocol

This section describes two methods for changing wireless communication protocols supported by the microcontroller used in this study.

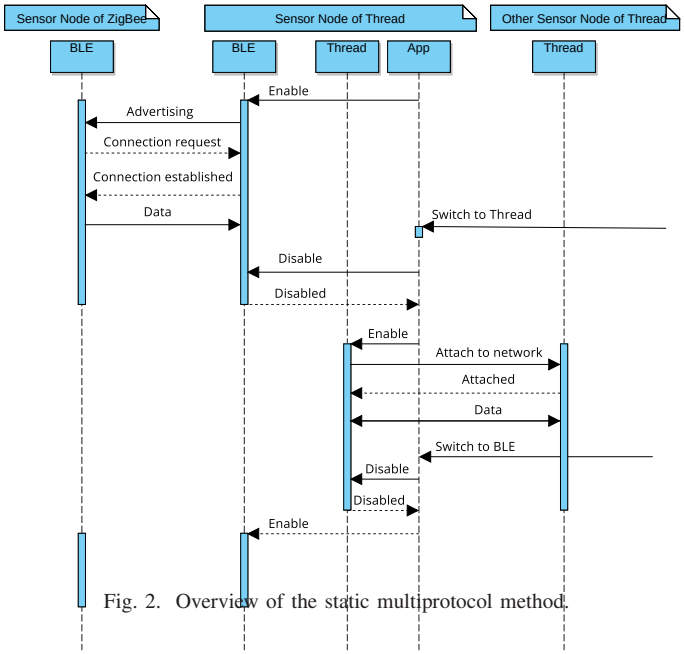

Fig. 2. Overview of the static multiprotocol method.

1) Static Multiprotocol Method: An overview of the static multiprotocol method is shown in Fig. 2. As shown in this figure, the static multiprotocol method allows the sensor node to switch to the new wireless communication protocol after disabling the currently active protocol [11].

In the static multiprotocol method, switching between the wireless communication protocols is typically performed at a timing defined by the developer of the application. For example, when a sensor node enables BLE and operates as a peripheral, it transmits advertising packets to announce its presence and then waits for a connection from a central device. Once the connection is established and data sharing between sensor nodes is completed, the sensor node disables BLE and enables Thread. When the sensor node enables Thread, it executes the necessary processes to connect with other nodes that are already part of the Thread sensor network. However, in the static multiprotocol method, it is necessary to complete the processes of disabling the current protocol and enabling the new one. This can be a challenge for sensor networks that require real-time performance, since these processes take a significant amount of time.

2) Dynamic Multiprotocol Method: An overview of the dynamic multiprotocol method is shown in Fig. 3. In the dynamic multiprotocol method, the communication functions of the microcontroller are shared among multiple wireless communication protocols in a time-division manner [11]. The sensor node in each wireless communication protocol (i.e., BLE, Thread, and ZigBee) can temporarily request communication using another protocol while maintaining connections with other sensor nodes in the current network. For example, the sensor node can participate in Thread or ZigBee communications while temporarily functioning as a BLE peripheral or central. The dynamic multiprotocol method enables the sensor node to support multiple wireless communication protocols

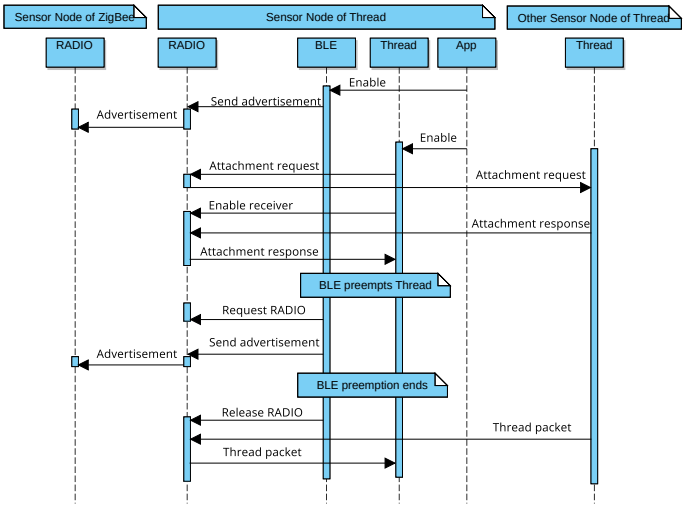

Fig. 3. Overview of the dynamic multiprotocol method.

simultaneously without the need for disabling or enabling the protocols. Therefore, the time required for switching the protocol is significantly reduced compared to the static multiprotocol method.

In the dynamic multiprotocol method, BLE always has priority over other protocols (i.e., ZigBee, Thread). According to Nordic Semiconductor, the Bluetooth packet error rate should be 0% without any external interference, which could lead to packet loss in Thread/ZigBee. [11]. To mitigate packet loss in Thread/ZigBee, it is necessary to adjust parameters related to BLE communication (i.e., the interval between advertisements, and data transmissions).

B. Algorithm for Exchanging Sensor Data Between Sensor Nodes in Different Networks

In this section, we describe the process of exchanging sensor data and verifying its integrity among sensor nodes belonging to different wireless sensor networks. A sensor node belonging to a Thread sensor network queries a neighboring sensor node in a ZigBee sensor network about the integrity of the sensor data.

Specifically, we use the dynamic multiprotocol method to temporarily change the wireless communication protocols to BLE, enabling the sharing of sensor data. The dynamic multiprotocol method allows for the simultaneous execution of multiple wireless communications without disabling and enabling operations during switching. In the BLE communication, the sensor nodes in the Thread-based sensor network are the targets for validation, while the sensor nodes in the ZigBee-based sensor network are used for validation. The sensor nodes in the ZigBee network act as peripherals and request a time slot for BLE communication every 10 minutes using the dynamic multiprotocol method to transmit advertising packets. When acting as centrals and receiving these packets, the sensor nodes in the Thread network measure the Received Signal Strength Indicator (RSSI). The sensor node that requires the verification then detects the nearest sensor node on the ZigBee-based sensor network based on the measured RSSI and shares the sensor data with that. After that, the sensor node on the ZigBee-based sensor network compares the received data with its own data to verify the integrity of the sensor data.

TABLE II
PROCESSING TIME REQUIRED FOR CHANGING BETWEEN DIFFERENT WIRELESS COMMUNICATIONS.

Step	Processing time
BLE stack disable	244.14 [μ]
Thread stack enable	33.01[ms]
Thread attaching	0.20[s]
Thread stack disable	518.80 [μ]
BLE stack enable	251.10[ms]

V. Preliminary Experiment

In the static multiprotocol method, it is necessary to disable and enable the current and new wireless communication protocols, respectively. In this preliminary experiment, we evaluate the processing time required to change between different wireless communication protocols using the static multiprotocol method to determine the most suitable method for the proposed system.

In this experiment, we measure the processing time required to enable and disable BLE, as well as to leave and join the Thread sensor network. The processing time required to join the Thread network varies depending on the scale of the network. The experiment is conducted in an environment where the Thread network consists of two sensor nodes.

The experimental results are shown in Tab. II. As shown in this table, when changing the mode from BLE to Thread, it takes a total of 233.27 ms, including the processing time required to disable BLE and join the Thread sensor network. Additionally, when changing the mode from Thread to BLE, it takes a total of 251.62 ms, including the processing time required to leave a Thread sensor network and enable BLE. From the results, it can be clarified that the static multiprotocol method can be a challenge for sensor networks that require real-time performance, as it takes a significant amount of time when changing the communication modes. Therefore, the proposed system adopts the dynamic multiprotocol method, which allows for the simultaneous execution of multiple wireless communications.

VI. Performance Evaluation

A. Impact of BLE Communication on Thread Communication

Each sensor node performs periodic and temporary communication for sharing sensor data using BLE and communicates within its own sensor network using Thread or ZigBee during the remaining time. The quality of communication using Thread and ZigBee is affected by the frequency and volume of data transmitted over BLE. In this experiment, we evaluate the impact of BLE communication settings on wireless communication quality using Thread.

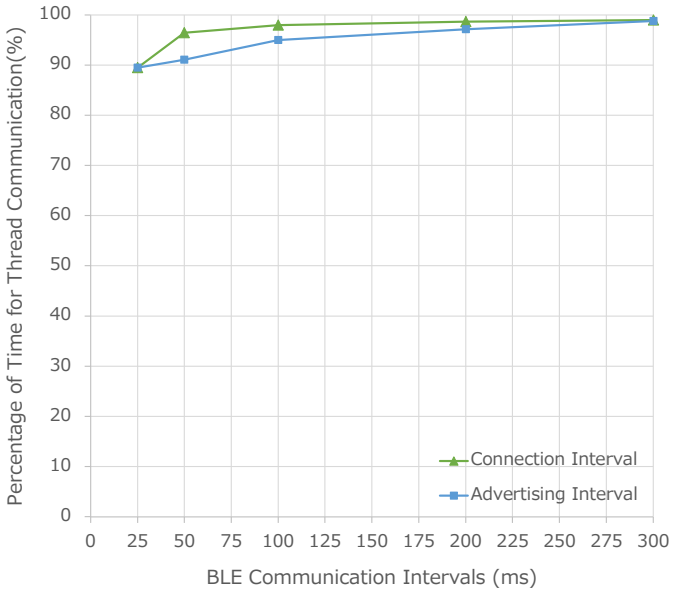

Fig. 4. The percentage of time for Thread communication.

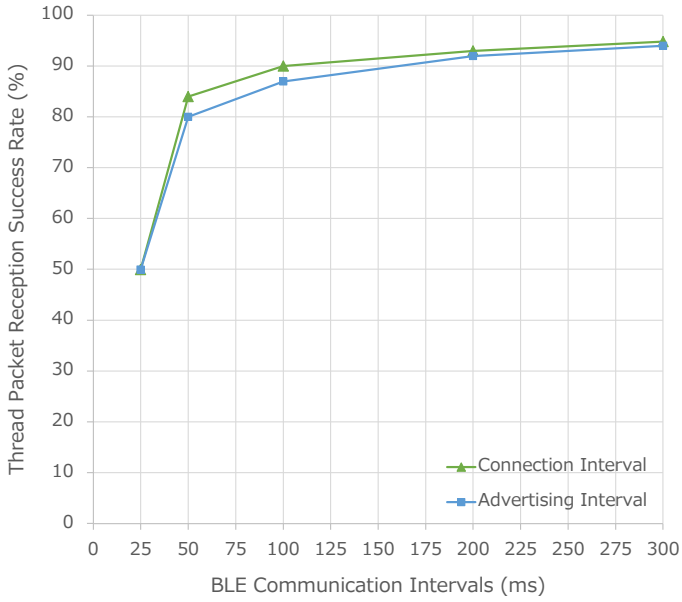

Fig. 5. The reception success rate of packets sent using Thread.

In this experiment, we evaluate the impact of changes in the interval of BLE communication on the percentage of time when the Thread communication is possible and the reception rate of data packets sent using Thread. In the Thread, the sensor node sends IEEE 802.15.4 compliant packets of 127 bytes and transmits 1000 times at a rate of 1 packet per second. The BLE communication intervals are set to 25 ms, 50 ms, 100 ms, 200 ms, and 300 ms, and each BLE packet is a size of 20 bytes.

The percentage of time when Thread communication is available is presented in Fig. 4. This figure presents the results for two cases. The first is when data is transmitted after establishing a connection between the central and peripheral (i.e., Connection Interval). The second is when data is transmitted using advertising packets sent from the peripheral (i.e., Advertising Interval). As shown in the figure, it is clear that as the BLE communication interval shortens, the possible percentage of time for Thread communication decreases. However, the Thread communication can be performed longer than 89% of total communication time even when the BLE packets are transmitted every 25 ms.

The reception success rate of packets sent using Thread is shown in Fig. 5. As shown in the figure, it is observed that as the BLE communication interval shortens, the reception success rate of Thread packets decreases. Overall, the reception success rate of Thread packets is above 80% but markedly decreases with an extremely short BLE communication interval of 25 ms. Therefore, the BLE communication interval should be set to 50 ms or longer to minimize the impact on Thread communication.

B. Processing Time for Sensor Data Exchange

In this experiment, we evaluate whether the proposed system enables real-time sensor data exchange and verification between sensor nodes from different wireless sensor networks. The experiment involves two sensor nodes equipped with sensors that can observe the same environmental data (e.g., temperature, humidity). While the specific algorithm for data verification is still under consideration, this experiment focuses only on evaluating the time required for data exchange using BLE.

The results are based on the average of 10 trials of data exchange, which showed that sensor data exchange between the nodes takes 22.16 ms. Since the sensor data can be shared in a short time, it is necessary to consider a verification method with a short processing time to enable real-time reliability verification in the future study.

VII. Considerations for Future Work

A. An Algorithm for Exchanging Sensor Data Between Multiple Sensor Nodes

In this study, we proposed an algorithm for exchanging sensor data between sensor nodes in different networks on a one-to-one basis. However, to further enhance reliability, it is necessary to consider an algorithm for exchanging sensor data among multiple sensor nodes. For example, the algorithm based on the majority vote allows for cross-verification of the sensor data where multiple readings from the different nodes can identify inconsistencies and reduce the impact of erroneous data. By integrating information from various sources, the system can ensure more accurate and trustworthy environmental assessments, ultimately leading to better decision-making based on reliable data.

When exchanging data among multiple sensor nodes, several nodes may simultaneously request sensor data verification. This requires the sensor nodes to receive and process data at the same time. In this case, if multiple sensor nodes initiate BLE communication simultaneously, interference can occur in the wireless channel and potentially reduce data reliability. Therefore, to prevent communication from occurring simultaneously, a media access control (MAC) method is necessary to determine the appropriate timing for each node to transmit data.

Additionally, the receiving sensor node requires an algorithm to buffer and sequentially verify the multiple sensor data it receives. A key challenge will be integrating and comparing data received from multiple nodes. For example, data can be weighted based on the distance to the transmitting sensor nodes, and methods such as majority voting can be used to identify the most reliable data. Furthermore, investigating and comparing time-series trends should be considered.

B. Encryption During Sensor Data Exchange

When multiple sensor nodes collaborate to verify the reliability of sensor data, there is a risk of data being intercepted during communication between nodes. Wireless sensor networks are vulnerable to attacks by third parties, potentially compromising the integrity and confidentiality of the data. Therefore, encryption is necessary when exchanging sensor data. Moreover, to ensure that the data has not been tampered with during transmission from the sender to the receiver, techniques like digital signatures should be used. The algorithm should be developed to enable encryption and digital signature techniques to run on low-performance microcontrollers in real-time.

C. Sensor Data Verification Using Machine Learning

In this study, we assume a scenario where sensor nodes simply compare whether their sensor data matches. However, more advanced methods are required to accommodate the diversity of sensor data. Therefore, it is necessary to explore approaches for advanced data analysis while maintaining real time performance and energy efficiency by selecting lightweight machine learning models suitable for microcontrollers.

Specifically, a method could be considered where features are extracted from the sensor data and a lightweight classifier is built based on those features. For example, using lightweight machine learning methods such as decision trees or random forests could enable the detection of anomalies in sensor data. These algorithms use minimal memory and have low computational overhead, allowing them to be executed on microcontroller boards. Additionally, for time-based analysis of sensor data, an LSTM (Long Short-Term Memory) model could be used to detect anomalies based on time-based trends.

VIII. Conclusion

In this study, we assume that sensor nodes physically close in proximity can generate sensor data with similar information, and propose a new method to ensure the integrity of the sensor data by enabling nodes participating in different sensor networks to collaborate with the adjacent nodes to verify the data. It has been clarified that even when data is frequently exchanged between different sensor networks, the performance of communication within each sensor network is minimally affected. In the future, we aim to achieve reliable sensor data exchange by introducing machine learning and encryption of sensor data.

Acknowledgment

This work was supported by JSPS KAKENHI Grant Number JP24K03045 and JST SPRING Grant Number JPMJSP2101.

References

[1] Cabinet Office Government of Japan, “Society 5.0”, https://www8.cao.go.jp/cstp/society5_0/index.html, (accessed 2024-8-2).
[2] Zhang Huanan, Xing Suping, Wang Jiannan, “Security and application of wireless sensor network”, Procedia Computer Science, Vol.183, pp.486-492, 2021.
[3] Tariq U, Ahmed I, Bashir AK, Shaukat K, “A Critical Cybersecurity Analysis and Future Research Directions for the Internet of Things: A Comprehensive Review”, Sensors 23, no.8, 2023.
[4] H. Zhang, H. Yamamoto, “Lifelog Mesh Sensor Network System Supporting Wake-Up Control Function Based on States of Power Generation”, 2024 IEEE 21st Consumer Communications and Networking Conference(CCNC), pp.484-489, 2024.
[5] A. J. Alam Majumder, C. B. Veilleux, J. D. Miller, “A Cyber-Physical System to Detect IoT Security Threats of a Smart Home Heterogeneous Wireless Sensor Node”, IEEE Access, vol.8, pp.205989-206002, 2020.
[6] K. Christidis, M. Devetsikiotis, “Blockchains and Smart Contracts for the Internet of Things”, IEEE Access, vol.4, pp.2292-2303, 2016.
[7] Nordic, “nRF52840”, https://www.nordicsemi.com/Products/Low-power-short-range-wireless/nRF52840, (accessed 2024-8-2).
[8] I. Unwala, Z. Taqvi, J. Lu, “Thread: An IoT Protocol”, 2018 IEEE Green Technologies Conference (GreenTech), pp.161-167, 2018.
[9] Alireza Zohourian, Sajjad Dadkhah et al., “IoT Zigbee device security: A comprehensive review”, Internet of Things, Vol.22, 2023.
[10] Matthias Cäsar, Tobias Pawelke, Jan Steffan, Gabriel Terhorst, “A survey on Bluetooth Low Energy security and privacy”, Computer Networks, Vol.205, 2022.
[11] Nordic, “Nordic Semiconductor launches a ZigBee solution for the nRF52840 multiprotocol SoC, expanding its offering for smart home applications,” https://www.threadgroup.org/What-is-Thread/Overview, (accessed 2024-8-2).
[12] Abdelmoughni Toubal, Billel Bengherbia, Mohamed Ould Zmirli, Abderrezak Guessoum, “FPGA implementation of a wireless sensor node with built-in security coprocessors for secured key exchange and data transfer”, Measurement, Vol.153, 2020.
[13] Bin Hu, Wen Tang, Qi Xie, “A two-factor security authentication scheme for wireless sensor networks in IoT environments”, Neurocomputing, Vol.500, pp.741-749, 2022.
[14] S. -J. Hsiao, W. -T. Sung, “Employing Blockchain Technology to Strengthen Security of Wireless Sensor Networks”, IEEE Access, vol.9, pp.72326-72341, 2021.
[15] Ahmad R, Wazirali R, Abu-Ain T, “Machine Learning for Wireless Sensor Networks Security: An Overview of Challenges and Issues”, Sensors 22, no.13, 2022.
[16] A. Molina Zarca et al., “Security Management Architecture for NFV/SDN-Aware IoT Systems”, IEEE Internet of Things Journal, vol.6, no.5, pp.8005-8020, 2019.
[17] F. Meneghello, M. Calore, D. Zucchetto, M. Polese, A. Zanella, “IoT: Internet of Threats? A Survey of Practical Security Vulnerabilities in Real IoT Devices”, IEEE Internet of Things Journal, vol.6, no.5, pp.8182-8201, 2019.
[18] L. D. Xu, W. He, S. Li, “Internet of Things in Industries: A Survey”, IEEE Transactions on Industrial Informatics, vol.10, no.4, pp.2233-2243, 2014.